On the influence of inter-agent variation on multi-agent algorithms solving a dynamic task allocation problem under uncertainty

Gerrit Anders, Christian Hinrichs, Florian Siefert, Pascal Behrmann, Wolfgang Reif, Michael Sonnenschein

Angaben zur Veröffentlichung / Publication details:

Anders, Gerrit, Christian Hinrichs, Florian Siefert, Pascal Behrmann, Wolfgang Reif, and Michael Sonnenschein. 2012. "On the influence of inter-agent variation on multi-agent algorithms solving a dynamic task allocation problem under uncertainty." In *2012 IEEE Sixth International Conference on Self-Adaptive and Self-Organizing Systems, 10-14 September 2012, Lyon, France*, edited by Salima Hassas, Paul Robertson, Anwitaman Datta, Marie-Pierre Gleizes, and Ingo Scholtes, 29–38. Piscataway, NJ: IEEE. https://doi.org/10.1109/saso.2012.16.

On the Influence of Inter-Agent Variation on Multi-Agent Algorithms Solving a Dynamic Task Allocation Problem under Uncertainty

Gerrit Anders*, Christian Hinrichs†, Florian Siefert*, Pascal Behrmann*, Wolfgang Reif*, and Michael Sonnenschein†
*Institute for Software & Systems Engineering, Augsburg University, Germany
E-Mail: {anders, siefert, reif}@informatik.uni-augsburg.de, PascalBehrmann@gmx.de
†Department of Environmental Informatics, University of Oldenburg, Germany
E-Mail: {hinrichs, sonnenschein}@informatik.uni-oldenburg.de

***Abstract*—Multi-agent systems often consist of heterogeneous agents with different capabilities and objectives. While some agents might try to maximize their system's utility, others might be self-interested and thus only act for their own good. However, because of their limited capabilities and resources, it is often necessary that agents cooperate to be able to satisfy given tasks. To work together on such a task, the agents have to solve a task allocation problem, e.g., by teaming up in groups like coalitions or distributing the task among themselves on electronic markets.**

In this paper, we introduce two algorithms that allow agents to cooperatively solve a dynamic task allocation problem in uncertain environments. Based on these algorithms, we investigate the influence of inter-agent variation on the system's behavior. One of these algorithms explicitly exploits inter-agent variation to solve the task without communication between the agents, while the other builds upon a fixed overlay network in which agents exchange information. Throughout the paper, the frequency stabilization problem from the domain of decentralized power management serves as a running example to illustrate our algorithms and results.

***Keywords*-Inter-Agent Variation; Dynamic Task Allocation Problem; Distributed Problem Solving; Frequency Stabilization Problem**

I. The Dynamic Task Allocation Problem

Multi-agent systems (MAS) are usually composed of various agents with different capabilities and limited resources. Because complex tasks often require several capabilities and resources that exceed those of a single agent, they have to be solved in a cooperative manner. In many cases, solving a given task allocation problem (TAP) is accompanied by the formation of explicit organizational structures such as coalitions [1] which allow agents to satisfy tasks cooperatively in an optimized way, even though agents might be self-interested [2] or untrustworthy [3]. One step in coalition formation is thus to decide which agents should work together on which task.

The dynamic task allocation problem (DTAP) that we address in this paper consists of a single task that is dividable into arbitrary partitions and to be solved by a set of cooperative agents $\mathcal{A} = \{a_1, \ldots, a_n\}$. While the task does not require different capabilities, it can exceed the resources available to a single agent. Therefore, instead of having to decide which agent should work on which task, the main problem in our DTAP is to distribute the task among agents.

In our DTAP, the single task of the agents $\mathcal{A}$ is to hold a global value at a constant, predefined target value $V_T \in \mathbb{R}$ at all times. To this end, all agents $\mathcal{A}$ cooperatively work on this task. Each agent $a_i \in \mathcal{A}$ can autonomously decide whether or not to change the actual, current global value $V(t) \in \mathbb{R}$ in time step t by changing its own contribution $v_{a_i}(t)$. To be able to make informed decisions, each agent knows $V(t)$ and thus the current deviation $\Delta V(t) = V(t) - V_T$ from the target value. However, the agents are differently sensitive to the current deviation from the target value. So only if $|\Delta V(t)|$ reaches or exceeds an agent a_i's threshold $\phi_{a_i} \in \mathbb{R}_0^+$, it can decide whether or not to change its own contribution. If it decides to contribute to change the global value, it can either increase or decrease its current contribution and thus increase or decrease the global value. Since each agent a_i has limited resources, a_i has a minimum contribution of $v_{a_i,min} \in \mathbb{R}$ and a maximum contribution of $v_{a_i,max} \in \mathbb{R}$ ($v_{a_i,min} \leq v_{a_i,max}$). If $0 \in [v_{a_i,min}, v_{a_i,max}]$, agents are at liberty to decide against a contribution (i.e., $v_{a_i}(t) = 0$). Moreover, the contribution $v_{a_i}(t)$ of an agent in time step t depends on its contribution $v_{a_i}(t-1)$ in time step $t-1$, thus introducing a kind of inertia. In detail, an agent a_i can increase or decrease its contribution by at most $\Delta v_{a_i,max} \in [0, v_{a_i,max} - v_{a_i,min}]$ from one time step to another. Because of this inertia as well as the minimum and maximum contribution, an agent a_i can actually increase its contribution by at most $\Delta v_{a_i,+}(t)$ and decrease it by at most $\Delta v_{a_i,-}(t)$ in time step t:

$$\begin{aligned}\Delta v_{a_i,+}(t) &= \min\left(\Delta v_{a_i,max}, v_{a_i,max} - v_{a_i}(t)\right)\\ \Delta v_{a_i,-}(t) &= \min\left(\Delta v_{a_i,max}, v_{a_i}(t) - v_{a_i,min}\right)\end{aligned} \tag{1}$$

Therefore, $\Delta v_{a_i}(t) \in [-\Delta v_{a_i,-}(t), \Delta v_{a_i,+}(t)]$ for the change in contribution $\Delta v_{a_i}(t) = v_{a_i}(t+1) - v_{a_i}(t)$ of an agent a_i in time step t. Such properties can be found in control systems that regulate physical devices like generators. Consequently, if the state of an agent is not known, it is uncertain to what extent it can contribute to a given deviation from V_T.

As it is the aim to hold the global value at the target value V_T, the change in contribution $\Delta v_{a_i}(t)$ of all agents $a_i \in \mathcal{A}$ in time step t should compensate for $\Delta V(t)$, i.e., the current deviation from V_T. Thus, the global problem to be solved can be formulated as follows:

$$\begin{aligned} \text{minimize} \quad & \left| \Delta V(t) - \sum_{a_i \in \mathcal{A}} \Delta v_{a_i}(t) \right| \\ \text{subject to} \quad & \forall a_i \in \mathcal{A} : \Delta v_{a_i}(t) \in [-\Delta v_{a_i,-}(t), \Delta v_{a_i,+}(t)] \end{aligned}$$

However, as stated above, each agent has a different threshold ϕ_{a_i} of when to adjust its contribution. Consequently not all agents necessarily change their contribution. The agents' thresholds are given by a probability distribution with a mean threshold $\mu_\phi > 0$. We assume that this form of inter-agent variation as well as the agents' other properties are attributes of the system and cannot be changed from outside.

Further, uncertainty is introduced by the environment, which can change the global value $V(t-1)$ by an arbitrary value $\delta(t) \in [-\delta_{max}, \delta_{max}]$ in each time step t:

$$V(t) = V(t-1) + \sum_{a_i \in \mathcal{A}} \Delta v_{a_i}(t-1) + \delta(t)$$

$\delta(t)$ has to be in $[-\delta_{max}, \delta_{max}]$, because the agents have no chance of holding the global value at a constant level if they cannot change their contribution by $\delta(t)$.

Usually, a given TAP is to be solved (nearly) optimally with regard to a specific cost function. MAS algorithms for TAPs are therefore often based on electronic markets which use auctions (e.g., [4]) or variants of the contract net protocol (e.g., [5]) to distribute a given task among agents in a nearly optimal way. In contrast, the algorithms that we present in this paper serve to investigate the influence of inter-agent variation on the system's behavior when solving our DTAP in a distributed manner. Inter-agent variation is a property that has been studied in the context of distributed task allocation by Campbell et al. [6] and applied to various problems (e.g., [7], [8]). In Section III, we propose an algorithm that solves our DTAP without any communication between agents. This is achieved by utilizing the inter-agent variation available within the system. However, because the algorithm relies on an adequate amount of inter-agent variation, it particularly allows to examine how this property affects the system's behavior. The second algorithm, which is presented in Section IV, is based on neighborhood structures in which agents negotiate about their contribution in an iterative process. While inter-agent variation is still present, agents can coordinate their actions. We evaluate these two algorithms in Section V before we discuss related work in Section VI. In Section VII, we finally conclude the paper and give an outlook on future work. Throughout the paper, the frequency stabilization problem in the power grid, which is introduced in Section II, serves as a running example.

II. Running Example: Frequency Stabilization in the Power Grid

An instance of the DTAP outlined in Section I is the stabilization of the utility frequency in the power grid. However, before we go into details concerning this matter, we show which factors influence the utility frequency and how it is stabilized in the power grid nowadays.

A. State of the Art

In the European UCTE synchronous area, which consists of several independent control areas, the power grid is operated at a nominal frequency $f_T = 50$ Hz. A frequency deviation $\Delta f(t) = f(t) - f_T$ at time step t from f_T can be traced to a mismatch between power supply and demand. If too much energy is fed into the system, the frequency $f(t)$ is above 50 Hz. If more energy is demanded than produced, $f(t)$ is below 50 Hz. While $\Delta f(t)$ can be measured at every socket, the actual power deviation $\Delta P(t)$ between power supply and demand at time step t can be determined by means of $\Delta f(t)$ and κ, the *network frequency characteristic*, which puts the mismatch between supply and demand into relation to the frequency deviation:

$$\Delta P(t) = \Delta f(t) \cdot \kappa \tag{2}$$

Further, κ depends on the size of the power grid and defines its behavior and reaction to disturbances. For the European transmission system, κ is 15000 MW/Hz [9].

Most power generators as well as many consumers rely on the nominal frequency f_T for proper operation and may disconnect from the grid or even take damage if $\Delta f(t)$ rises above a critical value. Since $f(t)$ is the same for all utilities connected to the grid, the task is to cooperatively hold the global utility frequency within a small corridor around 50 Hz, optimally at exactly 50 Hz. As mentioned in Section I, in each point in time, the environment can change $\Delta P(t)$ by an arbitrary value. This uncertainty is introduced by stochastic power plants like weather-dependent generators and unpredictable power demand.

The necessary control actions for maintaining this balance are divided into successive interdependent steps [9]. *Primary control*, the first step in the series of control actions, aims at stabilizing an increasing frequency deviation at a stationary value in order to ensure the operational reliability of the grid. This step is implemented as a joint action of all connected utilities, regardless of the control area they are located in. They simultaneously increase or decrease generation in a time frame of seconds after a disturbance. Primary control is activated in each participating power plant at the latest when the frequency deviation reaches ± 20 mHz, dependent on the local frequency measurement accuracy ($\leq \pm 10$ mHz) and insensitivity of the controller ($\leq \pm 10$ mHz), thus inducing inter-agent variation with different thresholds. On the other hand, *secondary control*, which is activated subsequent to primary control, pursues the goal to restore the frequency to its nominal value. It is carried out independently in each control area so that a disturbance is compensated for in the area where it occurred. While primary control underlies a globally standardized control mechanism implemented in the individual power plant's governors, the semi-automatic secondary control may be realized differently in each control area. We therefore focus on secondary control in a single control area throughout the rest of this paper.

B. Dynamic Task Allocation Problem

Usually, the capacity for secondary control is issued by controllable, fast-responding generators like hydro, biofuel, or gas power plants and is activated by a central scheduler within each control area. More formally, such a scheduler has to solve an instance of the DTAP introduced in Section I.

In order to counter the mismatch between energy supply and demand, the set $\mathcal{A}$ of power plants contributing secondary control power in a specific control area have to adjust their output according to $\Delta P(t)$. Optimally, they would provide an aggregated reactive adjustment $\sum_{a_i \in \mathcal{A}} r_{a_i}(t)$, where $r_{a_i}(t)$ is the reactive adjustment of power plant $a_i \in \mathcal{A}$, that is equal to $\Delta P(t)$ and takes effect in the next time step $t+1$. Hence, a_i's output in the next time step is $p_{a_i}(t+1) = p_{a_i}(t) + r_{a_i}(t)$. Under the assumption that the environment does not change $\Delta P(t+1)$, the mismatch would be dissolved.

However, the process of modifying a power plant's output is bound to physical constraints which limit the power plant's output range and how fast it can increase or decrease its output. The output range of a power plant a_i is defined by a_i's minimum output $p_{a_i,min}$[1] and maximum output $p_{a_i,max}$. $\Delta p_{a_i,max}$ is the absolute maximum value by which a_i can change its output.[2] Consequently, for the given time step t, power plant a_i can increase its output by at most $\Delta p_{a_i,+}(t)$ and decrease it by at most $\Delta p_{a_i,-}(t)$ akin to Equation 1.

Further, the system has to deal with inter-agent variation as the power plants are differently sensitive to power deviations. The threshold ϕ_{a_i} states at which minimum power deviation ϕ_{a_i} power plant a_i takes measures to stabilize and restore the frequency by adjusting its output. As stated in Section I, the degree of inter-agent variation is defined by a probability distribution with a mean threshold μ_ϕ.

Because of these physical constraints, it is often not possible to provide the reactive adjustment necessary to balance energy supply and load within a single time step. The objective is therefore to compensate for $\Delta P(t)$ as effectively as possible (we deliberately abstract from economic criteria that might influence optimal solutions):

$$\text{minimize} \quad \left| \Delta P(t) - \sum_{a_i \in \mathcal{A}} r_{a_i}(t) \right| \tag{3}$$
$$\text{subject to} \quad \forall a_i \in \mathcal{A} : r_{a_i}(t) \in [-\Delta p_{a_i,-}(t); \Delta p_{a_i,+}(t)]$$

If this optimization problem is solved by a central scheduler, one can easily define an optimal solution according to Equation 3. However, centralized solutions do not scale with the number of power plants that participate in the frequency stabilization problem. As the number of distributed energy resources is steadily increasing, a centralized approach is thus not suitable in future energy management systems [10]. Further, future energy systems will develop to a more and more "open access" system in which any component becomes an active decision maker [11]. Therefore, decentralized approaches have to be investigated. However, if the frequency stabilization problem is solved in a decentralized way by making use of local knowledge and local decisions, it is not obvious how to compensate for a given power deviation. For example, if a power plant does not know the state and properties of others, it does not know how they will react to a given power deviation.

For the two distributed utility frequency stabilization approaches that are introduced in Section III and Section IV, there is no central instance that calculates appropriate reactive adjustments for all power plants. Instead, each power plant tries to adjust its output on its own by making use of local knowledge. The view of an individual power plant on the frequency stabilization problem is thus the following: Each power plant a_i determines the current power deviation $\Delta P(t)$ as explained in Equation 2. If $|\Delta P(t)| \geq \phi_{a_i}$, its goal is to autonomously adjust its current output by $r_{a_i}(t)$ such that together with the reaction of the other power plants $\mathcal{A} \setminus \{a_i\}$ the deviation is reduced and, optimally, dissolved.

In such a setting, each power plant a_i might only have knowledge about how a limited number of other power plants in its neighborhood $\mathcal{N}_{a_i}$ will react.[3] Because we assume that $\mathcal{N}_{a_i} \neq \mathcal{A}$, $\Delta P(t)$ is the only information that is available to all power plants. Thus, the system could either tend to insufficiently react to a given power deviation or to overreact. The latter can result in undesired oscillations.

In the following, we present two bio-inspired distributed algorithms that solve the frequency stabilization problem by autonomous decisions based on local knowledge.

III. The Honey Bee Algorithm

Variation and error have been identified as important properties of complex technical (e.g., [12]) and biological systems (e.g., [13]) to allow them to reach their objectives. Inspired by these insights, Campbell et al. [6] hypothesized and deduced that inter-agent variation is also an essential property to enable effective self-organization and a stable system behavior in cooperative MAS. Based on the results of Campbell et al. (see Section VI for a more detailed discussion), we regard the way honey bees regulate the temperature of their nest to solve the DTAP given in this paper and investigate the impact of inter-agent variation on the system's behavior.

In order that the brood develops normally, honey bees have to hold the temperature of their nest between 32 °C and 36 °C, optimally at 35 °C [13]. Worker bees manage to regulate the temperature by fanning hot air out of the nest if the nest temperature is too high or by generating metabolic heat if it is too low. Genetic variance makes sure that the bees have different threshold temperatures, i.e., sensitivities, when to start cooling or heating the nest so that the optimal nest temperature can be maintained. Interestingly, this property is

[1]We assume that $p_{a_i,min}$ is a value ≥ 0 kW. We thus do not regard storage power plants or controllable consumers.

[2]For simplicity, we assume that $\Delta p_{a_i,max}$ does not depend on $p_{a_i}(t)$ and that $p_{a_i}(t)$ can take arbitrary values between $p_{a_i,min}$ and $p_{a_i,max}$.

[3]If $\forall a_i \in \mathcal{A} : \mathcal{N}_{a_i} = \mathcal{A}$, each power plant a_i would have full knowledge about the system.

achieved without any communication between the individuals. Bees solely perceive and influence the temperature in their environment. The property of variance is of great importance. If each bee equally perceived temperature, all bees would start to cool or heat the nest simultaneously. Thus, it is likely that the nest temperature would oscillate between two temperature values (see Section V), which is not desired from an evolutionary as well as a technical perspective. Because of these characteristics, the nest thermoregulation mechanism is an auspicious example from nature illustrating how inter-agent variation allows complex self-organizing systems to achieve their goals without communication.

With regard to the definition of our DTAP, we mapped the honey bee nest thermoregulation mechanism onto the frequency stabilization problem in power systems as follows (see Figure 1): As each bee can feel the temperature of the nest, each power plant can determine the current power deviation $\Delta P(t) = \Delta f(t) \cdot \kappa$ which has to be compensated for as stated in Section II-B. Similarly to bees, power plants do not react simultaneously to this power deviation. Their different sensitivity values introduce inter-agent variation.

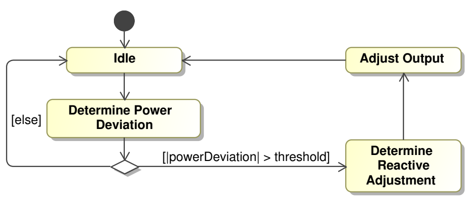

Figure 1. Control loop of the Honey Bee Algorithm

To be able to investigate how inter-agent variation influences the system's behavior, the function used by power plants to determine their reactive adjustment should have two properties: 1) In case of either too little or too much inter-agent variation, the power plants should either overreact to a given power deviation, which results in oscillations, or insufficiently adjust their output to compensate for a given power deviation $\Delta P(t)$. 2) In case of a suitable degree of inter-agent variation, the power plants should manage to compensate for $\Delta P(t)$.

$r^*_{a_i}(t)$ meets these properties. In detail, independent of $\Delta p_{a_i,-}(t)$ and $\Delta p_{a_i,+}(t)$, $r^*_{a_i}(t)$ specifies the preliminary reactive adjustment of power plant a_i on the basis of the power deviation $\Delta P(t)$, a_i's threshold ϕ_{a_i}, and the maximum value $\Delta p_{a_i,max}$ by which a_i can adjust its output:

$$r^*_{a_i}(t) = \alpha \cdot \Delta p_{a_i,max} \cdot \sqrt[\beta]{\frac{\phi_{a_i}}{K}} \cdot \left(1 - \sqrt[\gamma]{\frac{K}{|\Delta P(t)| + (K - \phi_{a_i})}}\right), \quad \text{with } K = 1 \text{ Hz} \cdot \kappa$$

$r^*_{a_i}(t)$ has some important characteristics. The greater the power deviation $\Delta P(t)$, the greater $r^*_{a_i}(t)$. Further, the slope of $r^*_{a_i}(t)$ depends on the power plant's threshold ϕ_{a_i}. While sensitive power plants contribute earlier than less sensitive ones, they do not adjust their contribution as much as less sensitive power plants. Obviously, the definition of $r^*_{a_i}(t)$ seems to be rather complex and other functions might even yield better results. However, it is well-suited to examine the impact of inter-agent variation on the system's behavior.

The crucial property that $r^*_{a_i}(t)$ has to meet to solve our DTAP is that the aggregated reactive adjustment of all power plants equals $\Delta P(t)$ as accurately as possible. To reach this objective, the constants α, β, and γ allow to calibrate the system so that the reactive adjustments coincide with the actual surplus of or deficit in power. Useful values for α, β, and γ depend on the properties of the underlying system (e.g., the number of power plants), the mean threshold μ_ϕ, as well as the degree of inter-agent variation.

Figure 2 depicts the reactive adjustment $r_{a_i}(t)^*$ dependent on the power deviation $\Delta P(t)$ and different sensitivity values for a power plant a_i with fix values for α, β, and γ. In Section V, we identify suitable values for these constants and analyze the role of inter-agent variation.

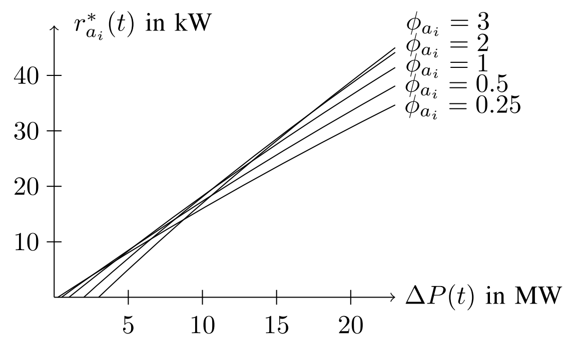

Figure 2. Reactive adjustment $r^*_{a_i}(t)$ in MW with different thresholds ϕ_{a_i} dependent on the power deviation $\Delta P(t)$, with $\Delta p_{a_i,max} = 1$ MW, $K = 150$ MW, $\alpha = 0.8$, $\frac{1}{\beta} = 0.15$, and $\frac{1}{\gamma} = 0.85$.

However, whenever a_i determines its reactive adjustment, it also has to take its current output $p_{a_i}(t)$, its minimum output $p_{a_i,min}$, and its maximum output $p_{a_i,max}$ into account. Therefore, a_i actually alters its output $p_{a_i}(t)$ by $r_{a_i}(t)$:

$$r_{a_i}(t) = \begin{cases} -\min\left(r^*_{a_i}(t), \Delta p_{a_i,-}(t)\right) & \text{if } \Delta P(t) \geq \phi_{a_i} \\ \min\left(r^*_{a_i}(t), \Delta p_{a_i,+}(t)\right) & \text{if } \Delta P(t) \leq -\phi_{a_i} \\ 0 & \text{otherwise} \end{cases}$$

If $\Delta P(t)$ is equal to or greater than a_i's threshold ϕ_{a_i}, it reduces its output by at most $\Delta p_{a_i,-}(t)$, or, if $\Delta P(t)$ is equal to or less than $-\phi_{a_i}$, a_i increases its output by at most $\Delta p_{a_i,+}(t)$. Otherwise, a_i does not adjust its output and thus does not contribute to neutralize $\Delta P(t)$. Consequently, given a power deviation $\Delta P(t)$, all power plants a_i for which $r_{a_i}(t) \neq 0$ either increase or decrease their output.

Summarizing, the Honey Bee Algorithm allows the power plants to compensate for $\Delta P(t)$ without any communication or knowledge about the other power plants' state or their reaction to the given deviation. In terms of Section II-B, this corresponds to an empty neighborhood for each power plant ($\forall a_i \in A : \mathcal{N}_{a_i} = \emptyset$). We thus expect the algorithm to scale well with the number of power plants. Importantly, as the algorithm exploits the system's inherent inter-agent variation, the quality of the algorithm's results heavily depends on the available amount of inter-agent variation. We investigate this essential property in Section V.

IV. The Schooling Fish Algorithm

The Honey Bee Algorithm introduced in the previous section does not include any communication between power plants so that the DTAP is solved on the basis of probabilistic local decisions and the observation of the environment. Hence, the algorithm is very sensitive to the amount of inter-agent variation present in the system, and has to be configured accordingly. In contrast, social interaction of individuals has been identified as another interesting model for the design of distributed task solving systems [14], which might be less dependent on the parameters of the underlying system. In the following, we present an algorithm for the DTAP given in this paper that is inspired by the schooling behavior of fish. Interaction is carried out indirectly, i.e., by observation of other individuals rather than direct negotiations. The algorithm is based on the preliminary work in [15], where a self-organizing coordination of dynamic electrical loads has been proposed.

Just like flocking birds, swarming insects, or other herding animals, the schooling of fish is an emergent behavior that arises from local interactions between individuals, without central control. Fish observe their environment and react to motions of other fish. These reactions can easily be described by few simple behavioral rules that take the distance, velocity, and direction of other fish into account. Based on these observations, a fish determines its own velocity and swimming direction (see [16] for a detailed description).

This principle can be applied to the frequency stabilization problem in power systems as follows: Each power plant determines the current power deviation $\Delta P(t) = \Delta f(t) \cdot \kappa$ as described in Section II. The reactive adjustment that, in average, each single power plant has to perform can be determined by normalizing the negative current power deviation to the amount of participating power plants:

$$r_{avg}(t) = \frac{-\Delta P(t)}{|\mathcal{A}|} \quad (4)$$

But as not every power plant is physically able to perform this average adjustment, a more equitable partitioning of the globally required reactive adjustment has to be found so that each power plant contributes preferably the same fraction of reactive adjustment relative to the amount it is able to contribute in total (i.e., power plants with a large reserve contribute more than power plants with a small reserve). We assume that a power plant a_i has a neighborhood $\mathcal{N}_{a_i}$ of other power plants it can observe, and a threshold ϕ_{a_i}, as stated in Section II-B. If $|\Delta P(t)| \geq \phi_{a_i}$, the task of a power plant a_i is to adapt its own reactive adjustment $r_{a_i}(t)$ based on its local view on the system. This is done iteratively, starting with $r_{a_i}(t) = 0$. The local view comprises observed properties of each neighbor $a_j \in \mathcal{N}_{a_i}$, namely the currently chosen $r_{a_j}(t)$ and the remaining possible adjustment $\Delta r_{a_j,max}(t)$ a power plant a_j is physically still capable of. For each power plant a_i, the latter is defined as:

$$\Delta r_{a_i,max}(t) = \begin{cases} \Delta p_{a_i,-}(t) - |r_{a_i}(t)| & \text{if } \Delta P(t) \geq \phi_{a_i} \\ \Delta p_{a_i,+}(t) - |r_{a_i}(t)| & \text{if } \Delta P(t) \leq -\phi_{a_i} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Note that $\Delta r_{a_i,max}(t)$ is always a positive value. Given these observed values, a power plant a_i is able to calculate the locally still required power adjustment as

$$\Delta r_{\mathcal{N}_{a_i}}(t) = r_{avg}(t) \cdot (|\mathcal{N}_{a_i}| + 1) - \sum_{a_j \in (\mathcal{N}_{a_i} \cup \{a_i\})} r_{a_j}(t)$$

where the term "locally" refers to the limited view of a_i on the system (i.e., its neighborhood). Further, the locally still *possible* absolute reactive adjustment is given by

$$\Delta r_{\mathcal{N}_{a_i},max}(t) = \sum_{a_j \in (\mathcal{N}_{a_i} \cup \{a_i\})} \Delta r_{a_j,max}(t).$$

Since the power plant a_i knows the situation in its neighborhood at this point in time, it is able to calculate an equitable partitioning of $\Delta r_{\mathcal{N}_{a_i}}(t)$ and therefore the estimated fraction of the reactive adjustment it should perform itself:

$$\hat{r}_{a_i}(t) = \begin{cases} \Delta r_{\mathcal{N}_{a_i}}(t) \cdot \frac{\Delta r_{a_i,max}(t)}{\Delta r_{\mathcal{N}_{a_i},max}(t)} & \text{if } \Delta r_{\mathcal{N}_{a_i},max}(t) > 0 \\ 0 & \text{otherwise} \end{cases}$$

Afterwards, the physical constraints of the power plant have to be applied to this calculated value:

$$r^*_{a_i}(t) = \begin{cases} \min(\hat{r}_{a_i}(t), \quad \Delta p_{a_i,+}(t)) & \text{if } \hat{r}_{a_i}(t) \geq 0 \\ \max(\hat{r}_{a_i}(t), \ -\Delta p_{a_i,-}(t)) & \text{if } \hat{r}_{a_i}(t) < 0 \end{cases} \quad (6)$$

However, due to varying neighborhood configurations, it may occur that, from a_i's perspective, the desired local reactive adjustment $r_{\mathcal{N}_{a_i}}(t)$ in its neighborhood $\mathcal{N}_{a_i}$ has been already exceeded. Given the above calculation, the power plant would now choose an adaptation in the opposite direction of the overall desired reactive adjustment to counterbalance the local overshooting. But as such a situation arises solely from a lack of global knowledge, this behavior would not be beneficial for the common goal. Thus, as a last step, the power plant compares the sign of the above calculated value $r^*_{a_i}(t)$ to the sign of the overall desired reactive adjustment (see Equation 4). If the signs are equal, the value $r^*_{a_i}(t)$ is added to the reactive adjustment $r_{a_i}(t)$ for the current time step t. If the signs are unequal, the calculated values are discarded. This ensures that every power plant in the population makes adjustments in the same direction so that no operating reserves are wasted. If the value of $r_{a_i}(t)$ has been changed during this process, $r_{a_i}(t)$ and $\Delta r_{a_i,max}(t)$ are finally made known to the environment.

Just like in the movement correction of fish, this adaptation is performed each time a value change is observed in the neighborhood. Overall, this mechanism yields an infinite

Listing 1. The Schooling Fish Algorithm

```
while True:
  wait for input: ΔP(t)
  r_{a_i}(t) ← 0
  publish r_{a_i}(t) and Δr_{a_i,max}(t)     // Equation 5
  if Δr_{a_i,max}(t) > 0:
    for c in range(0, k_{a_i}):
      wait for updates from neighbors,
        on timeout: break
      calculate r*_{a_i}(t)                 // Equation 6
      if sgn(r*_{a_i}(t)) == sgn(r_{avg}(t)):
        r_{a_i}(t) ← r_{a_i}(t) + r*_{a_i}(t)
        publish r_{a_i}(t) and Δr_{a_i,max}(t)
  establish r_{a_i}(t) as reactive adjustment
```

iterative adaptation to the changing environment by dividing the remaining required power adjustment into smaller and smaller pieces. While such a behavior is appropriate in a school of fish where each movement adaptation can be carried out instantly, it is not feasible for a technical system like the power network. Therefore, this process is meant as a planning phase in which the desired reactive adjustment is determined iteratively. We included an upper bound k_{a_i} on the number of iterations a power plant is allowed to perform for a time step t. This leads to a rapid termination of the planning phase. Only then each power plant a_i establishes its final value $r_{a_i}(t)$ as reactive adjustment for the current time step t. The whole process is summarized for a single power plant a_i in Listing 1.

In summary, the Schooling Fish Algorithm allows power plants to compensate for $\Delta P(t)$ using limited, local knowledge about their neighboring power plants. Due to its iterative, adaptive nature, we expect the algorithm to be able to cope with different amounts of inter-agent variation available in the system, and thus to be rather independent of the thresholds ϕ_{a_i}. This is evaluated in the following.

V. Evaluation

For the evaluation of the Honey Bee and the Schooling Fish Algorithm, we set up a system with a set $\mathcal{A}$ of 222 simulated controllable power plants of different types (hydro, biofuel, and gas power plants) with no other power plants contributing to the energy supply. The power plants had different physical properties based on real data. More precisely, $p_{a_i,min} \in [0.0 \text{ kW}, 414.0 \text{ kW}]$, $p_{a_i,max} \in [2.0 \text{ kW}, 45500.0 \text{ kW}]$, and $\Delta p_{a_i,max} \in [2.0 \text{ kW}, 45500.0 \text{ kW}]$. As $\forall a_i \in \mathcal{A} : p_{a_i,min} \geq 0$, we did not regard storage power plants or controllable consumers with a negative contribution (see Section II-B).

The power plants had to satisfy a load curve $\mathcal{L}(t)$ for a set of discrete time steps T, i.e., they had to hold $|\Delta P(t)| = |\mathcal{L}(t) - P(t)|$ as small as possible $\forall t \in T$. The energy demand was within the power plants' output range so that all points in the load curve could be theoretically reached by the power plants. Moreover, the load could only change by at most $\Delta P_{max} = \sum_{a_i \in \mathcal{A}} \Delta p_{a_i,max}$ from one time step to another so that $\forall t \in T \setminus \{0\} : |\mathcal{L}(t) - \mathcal{L}(t-1)| \leq \Delta P_{max}$. Thus, the maximum uncertainty δ_{max} introduced by the environment was limited by ΔP_{max} (see Section I).

In all simulation runs, the power plants' sensitivity $\phi_{a_i} \in [0 \text{ MW}, 150 \text{ MW}]$ was initialized by a central random number generator using a beta distribution with a mean threshold of $\mu_\phi = 3$ MW (we assume that our system models one per cent of the UCTE network). Different degrees of inter-agent variation were modelled by different values for the standard deviation $\sigma_\phi \in \{0 \text{ kW}, 75 \text{ kW}, 750 \text{ kW}, 7500 \text{ kW}, 15000 \text{ kW}\}$ of the power plants' sensitivity.

In order to properly evaluate both algorithms, the power plants only reacted to a power deviation, i.e., the power plants were not controlled in any other way and did not know the future power demand, there were no malfunctioning power plants, and all messages were processed correctly.

The evaluation was implemented in a sequential, round-based execution model. At the beginning of each round, the current load was updated, whereupon the current frequency deviation based on the current demand and the previous supply was determined. Afterwards, each power plant could determine the power deviation by means of the frequency deviation and adjust its output accordingly.

The Fitness Function: The goal of our algorithms is to compensate for a given power deviation $\Delta P(t)$. With respect to a single time step t, the fitness of a frequency stabilization algorithm thus depends on the absolute deviation $\mathcal{D}_s(t)$ of the power plants' reactive adjustment from $\Delta P(t)$ for a specific simulation run s (see Equation 3):

$$\mathcal{D}_s(t) = \left| \Delta P(t) - \sum_{a_i \in \mathcal{A}} r_{a_i}(t) \right|$$

The total deviation $\mathcal{D}_{s,total}$ for a single simulation run s is calculated by summing up $\mathcal{D}_s(t)$ for all time steps $t \in T$:

$$\mathcal{D}_{s,total} = \sum_{t \in T} \mathcal{D}_s(t)$$

As we perform several simulation runs S for each parametrization, we define an algorithm's fitness for this parametrization on the basis of the mean total deviation $\mathcal{D}_{total} = \sum_{s \in S} \frac{\mathcal{D}_{s,total}}{|S|}$:

$$\mathcal{F} = 1.0 - \frac{\mathcal{D}_{total} - \mathcal{D}_{best}}{\mathcal{D}_{worst} - \mathcal{D}_{best}}$$

The actual total deviation $\mathcal{D}_{total}$ is normalized to the difference between the highest $\mathcal{D}_{worst}$ and the lowest $\mathcal{D}_{best}$ mean total deviation that occurred in our simulation runs in one scenario, regardless of a specific parametrization. Thus, $\mathcal{F} = 0.0$ is the worst and $\mathcal{F} = 1.0$ is the best rating.

Scenarios: We evaluated the algorithms in two different scenarios. On the one hand, the load was set to a constant value $\mathcal{L}(t) = 56885$ kW for all time steps (scenario *CL*). The overall production of the power plants was initialized with $P(0) = 24678$ kW. Thus, the power plants' output should converge towards the demand as fast as possible and with as less oscillations as possible. On the other hand, the load was based on a real load curve over six days with a resolution of 15 minutes per time step (scenario *RL*). The initial production of the power plants was initialized with $P(0) = \mathcal{L}(0) = 56885$ kW. The power plants' task was to follow the load curve as close as possible.

In the following, we present our evaluation results. For each parametrization, we performed $|S| = 200$ simulation runs, each starting with another random initialization of a power plant's initial output $p_{a_i}(0) \in [p_{a_i,min}, p_{a_i,max}]$ and $\sum_{a_i \in \mathcal{A}} p_{a_i}(0) = P(0)$.

A. The Honey Bee Algorithm

First of all, our goal was to identify suitable parameters for the Honey Bee Algorithm that allow to examine how inter-agent variation influences the system's behavior when solving our DTAP. To identify these parameters, we used the scenario RL and evaluated different combinations of $\alpha \in \{0.3, 0.8, 1.2\}$, $\frac{1}{\beta} \in]0.00, 0.50]$, and $\frac{1}{\gamma} \in [0.50, 1.00]$, each with an increment of 0.05. Since we obtained promising results with $\alpha = 0.8$, $\frac{1}{\beta} = 0.15$, and $\frac{1}{\gamma} = 0.85$, we used these parameters for further investigations.

To analyze to what extent inter-agent variation influences oscillations and convergence, we employed the scenario CL and compared the results for the five different values for the standard deviation σ_ϕ of the power plants' sensitivity. As can be seen in Figure 3, inter-agent variation significantly influenced the agents' behavior and thus the algorithm's results. Too much inter-agent variation ($\sigma_\phi = 15000$ kW) made the algorithm too sluggish. Since too few agents contributed to the task, the system needed much time to compensate for the given power deviation. On the other hand, if the system featured no or insufficient inter-agent variation (e.g., $\sigma_\phi = 0$ kW, $\sigma_\phi = 75$ kW, or $\sigma_\phi = 750$ kW), the system tended to overreact and was prone to oscillations. The smaller σ_ϕ, the higher the amplitudes and the lower the damping. Regarding the results depicted in Figure 3, an inter-agent variation of $\sigma_\phi = 7500$ kW allowed the system to compensate for the given power deviation within approximately 4 time steps – almost without oscillations. Interestingly, the agents compensated for the majority of the power deviation in the first time step.

Table I lists the total deviation $\mathcal{D}_{total}$, the standard deviation $\sigma_{\mathcal{D}_{total}}$ of $\mathcal{D}_{total}$, as well as the fitness $\mathcal{F}$ for each parametrization. Like Figure 3, the table mirrors that the system's behavior was very sensitive to the amount of inter-agent variation. For scenario CL and $\sigma_\phi = 7500$ kW, $\mathcal{D}_{total}$ was only 13% or 19% of $\mathcal{D}_{total}$ for $\sigma_\phi = 750$ kW or $\sigma_\phi = 15000$ kW.

The results for the scenario RL are shown in Figure 4. For the sake of clarity, we only depict the results for three different values of σ_ϕ, representing a low, medium, and high amount of inter-agent variation. If an adequate degree of inter-agent variation was available, the power plants could compensate for power deviations very accurately. However, in case of too little inter-agent variation, the power plants did not sufficiently adjust their output and were not sensitive enough to react to small power deviations (the output was sometimes kept at a constant level while the consumption changed). Similarly, if the system featured too much inter-agent variation, the output seemed to lag behind the consumption and load peaks were not satisfied either.

Table I quantifies these observations. For scenario RL and $\sigma_\phi = 7500$ kW, the Honey Bee Algorithm yielded results for which $\mathcal{D}_{total}$ was 38% or 46% of $\mathcal{D}_{total}$ for $\sigma_\phi = 0$ kW or $\sigma_\phi = 15000$ kW. However, the Honey Bee Algorithm only achieved a maximum fitness of 0.677, indicating that the Schooling Fish Algorithm solved our DTAP much better.

	σ_ϕ	$\mathcal{D}_{total}$	$\sigma_{\mathcal{D}_{total}}$	$\mathcal{F}$
CL	0 kW	402484.68 kW	3046.73	0.000
	75 kW	401804.38 kW	3981.39	0.002
	750 kW	335255.37 kW	30041.63	0.187
	7500 kW	43232.44 kW	7981.76	1.000
	15000 kW	223861.35 kW	139629.91	0.497
RL	0 kW	1197486.89 kW	11186.26	0.197
	75 kW	1193843.26 kW	12429.11	0.200
	750 kW	1153631.58 kW	91626.88	0.229
	7500 kW	551714.40 kW	155030.11	0.677
	15000 kW	1462001.22 kW	576019.70	0.000

Table I
HONEY BEE ALGORITHM: STATISTICAL OVERVIEW OF DIFFERENT DEGREES OF INTER-AGENT VARIATION IN SCENARIOS CL AND RL

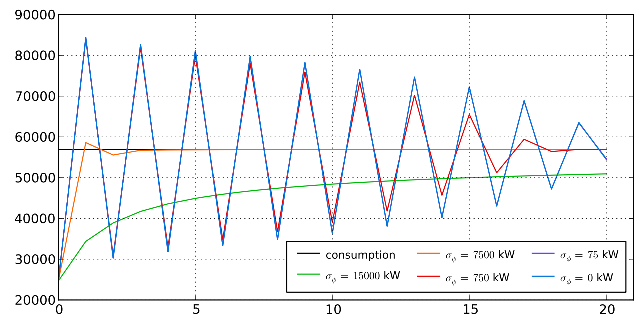

Figure 3. Honey Bee Algorithm: output adjustment to compensate for a given power deviation caused by a constant load for different degrees of inter-agent variation (standard deviations) in scenario CL.

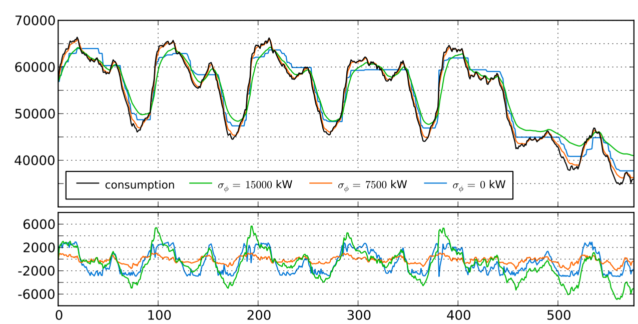

Figure 4. Honey Bee Algorithm: following a given load curve with different degrees of inter-agent variation (standard deviations) in scenario RL (top); remaining difference to the load curve (bottom).

B. The Schooling Fish Algorithm

The Schooling Fish Algorithm uses local knowledge and interaction in order to solve the given DTAP. We expected that the underlying communication network, which defines the neighborhood structure, would have an influence on the solution quality. So, first of all, we examined different network topologies: an ordered ring, random graphs with different densities, small worlds with varying shortcut probabilities (cf. [17]), and a regular mesh-shaped graph. We found that, in general, there is a trade-off between solution quality and the amount of communication in the system. But as there were no obvious dependencies between the network topology and the amount of inter-agent-variation in the system, we chose a single topology for further investigations, namely a random graph with a mean neighborhood size $\overline{|\mathcal{N}_{a_i}|} = 5$. This topology produced rather much communication overhead, but in turn yielded very good results.

To investigate the influence of inter-agent variation on the Schooling Fish Algorithm, we evaluated the same five values for the standard deviation σ_ϕ of the power plants' sensitivity as in the Honey Bee Algorithm. The simulation results for scenario CL are shown in Figure 5. As expected, the power

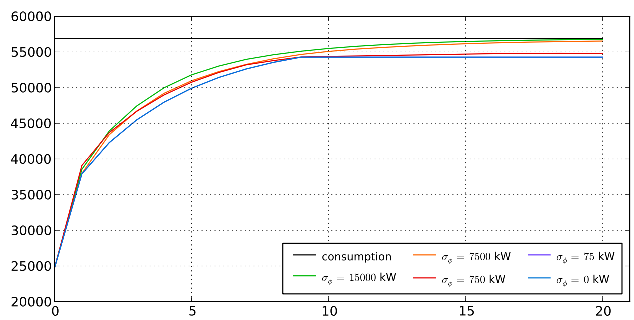

Figure 5. Schooling Fish Algorithm: output adjustment to compensate for a given power deviation caused by a constant load for different degrees of inter-agent variation (standard deviations) in scenario CL.

plants divided the deviation from the target load value iteratively into smaller and smaller pieces. Thus, a near-optimal partition of reactive adjustments was approached asymptotically without inducing oscillations. Because the number of iterations in each time step t was limited to $k_{a_i} = |\mathcal{N}_{a_i}|$ for each power plant a_i, the convergence advanced over several time steps.[4] More importantly, the results show that, regarding the speed of convergence, the inter-agent variation did not have an influence on the Schooling Fish Algorithm. All examined threshold distributions show a similar slope in the first nine time steps.[5] Afterwards however, parametrizations with a very low standard deviation σ_ϕ leveled at a overall production $\sum_{a_i \in \mathcal{A}} p_{a_i}(t) \approx (\mathcal{L}(t) - \mu_\phi)$. This indicates that the algorithm was not able to find an optimal partition of reactive adjustments if the system featured very little inter-agent variation. Table II shows the total deviation $\mathcal{D}_{total}$, the standard deviation $\sigma_{\mathcal{D}_{total}}$ of $\mathcal{D}_{total}$, the fitness $\mathcal{F}$, and the number of messages $\overline{msg}$ which were sent on average

[4]Note that the single iterations during the planning phase are not shown here; only the resulting overall production per time step is depicted.

[5]The graphs of $\sigma_\phi = 0$ kW and $\sigma_\phi = 75$ kW are almost identical and thus are visually indistinguishable in the figure.

in each time step per agent for each parametrization. These

	σ_ϕ	$\mathcal{D}_{total}$	$\sigma_{\mathcal{D}_{total}}$	$\mathcal{F}$	$\overline{msg}$
CL	0 kW	342029.00 kW	22568.91 kW	0.168	5.15
	75 kW	341011.44 kW	19163.09 kW	0.171	5.15
	750 kW	278791.92 kW	28809.23 kW	0.344	5.16
	7500 kW	123273.04 kW	17188.02 kW	0.777	5.25
	15000 kW	117251.98 kW	17340.99 kW	0.794	5.31
RL	0 kW	784578.48 kW	22565.43 kW	0.504	6.26
	75 kW	787419.68 kW	23818.22 kW	0.502	6.03
	750 kW	786352.64 kW	29336.63 kW	0.503	5.42
	7500 kW	182570.04 kW	37941.59 kW	0.952	11.66
	15000 kW	117896.81 kW	18960.42 kW	1.000	16.98

Table II
SCHOOLING FISH ALGORITHM: STATISTICAL OVERVIEW OF DIFFERENT DEGREES OF INTER-AGENT VARIATION IN SCENARIOS CL AND RL

values confirm that, regarding convergence in scenario CL, the algorithm performs better the more inter-agent variation is present in the system.

Figure 6 (upper subplot) shows the simulation results for scenario RL. For a better visualization, we included only the two best and the worst performing parametrizations in the diagram. The lower subplot depicts the remaining difference to the load curve. The statistical properties of

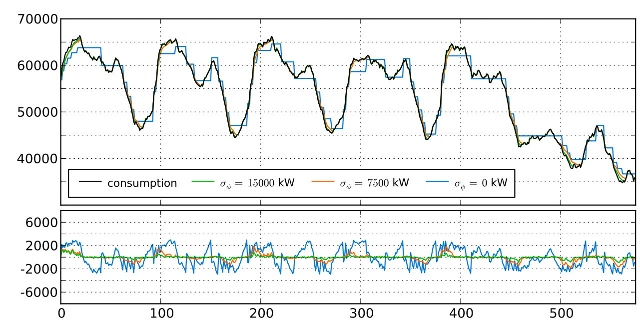

Figure 6. Schooling Fish Algorithm: following a given load curve with different degrees of inter-agent variation (standard deviations) in scenario RL (top); remaining difference to the load curve (bottom).

these simulations are shown in Table II as well. Again, it is obvious that the algorithm performs better with an increasing amount of inter-agent variation. If we look at the values more closely, however, we see that $\mathcal{D}_{total}$ dropped dramatically by 77% from $\sigma_\phi = 750$ kW to $\sigma_\phi = 7500$ kW. So there seems to be a critical threshold for the amount of inter-agent variation, which is needed by the Schooling Fish Algorithm to find a near-optimal solution.

VI. RELATED WORK

As stated in Section III, Campbell et al. investigate in [6] the role and benefit of inter-agent variation in self-organizing systems on the basis of a TAP that is solved by a variant of our Honey Bee Algorithm ([6] refers to our sensitivity as error). Similar to our DTAP, their agents try to maintain a task's value at a predefined target value by contributing to the task. As in our DTAP, the task's value is not fix but the value can change from one time step to another. Despite a changing environment, the authors of [6] are able to compute how much inter-agent variation is necessary so that the system reaches and is kept in a stable state. In this state, the task's value equals the target value in each future time step. However, there are several differences between the TAP given in [6] and our DTAP introduced in Section I. 1) [6] only regards negative deviations, whereas our agents try to compensate for negative as well as positive deviations. 2) In [6], each agent contributes with the same, fixed value. With respect to our Honey Bee Algorithm, an agent's contribution depends on the deviation from the target value as well as the agent's sensitivity, its properties, and its resources. 3) In [6], the current value of the task is decreased by a constant value so that the agents can anticipate their behavior. In our DTAP, the task's current value can change randomly within certain bounds from one time step to another. Therefore, power plants cannot anticipate how much to contribute. Consequently, as our system features a higher degree of inter-agent variation and is situated in an uncertain environment, we could not apply the formulas given in [6] to our Honey Bee Algorithm. Apart from that, [6] served as a theoretical foundation for our investigations.

In the context of frequency stabilization, there are approaches that suggest to introduce [18] or to make use of inherent [19] inter-agent variation as a means to avoid oscillations. [18] proposed that the hardware used to detect and react to an impending frequency instability should use different frequency-response thresholds. In [19], a mechanism for frequency stabilization is presented that is based on a great quantity of devices equipped with thermal storage capacity whose temperature should be held within certain bounds (e.g., refrigerators). For frequency stabilization, the control system that governs the cooling is modified such that it allows temperatures that are linearly dependent on the utility frequency. Interestingly, inter-agent variation is not in place as a result of different sensitivity thresholds like in our DTAP. Instead, [19] assumes that the devices are in different states with respect to their temperature and thus react differently to a given frequency deviation. A similar source of inter-agent variation is also present in our DTAP. Like the Honey Bee Algorithm, the mechanism in [19] gets by with local knowledge and without any communication.

[20], [21] present an approach for balancing energy supply and load by grouping small generators and consumers into pools. Within each pool, dynamic groups of generators and consumers immediately balance small power deviations, while the other devices compensate for significant deviations. The devices within a dynamic group successively adjust their output and communicate the remaining power deviation to a neighbor. If the dynamic groups cannot compensate for the deviation, the other devices within the pool adjust their output. If a device adjusts its output, it requests its neighbors to what extent they can adjust their output and subsequently selects those devices that are best suited to compensate for the power deviation. While these devices thus have to solve a knapsack problem to decide which devices should adjust their output, our agents only decide how to adjust their own output. When using the Honey Bee Algorithm, no communication is needed. With regard to the Schooling Fish Algorithm, information is exchanged within an agent's neighborhood, similar to [20], [21].

A related approach to the Schooling Fish Algorithm can be found in the domain of distributed constraint satisfaction problems (DCSPs). [22] introduces DCSPs as a formalism for cooperative distributed problem solving. Similar to our approach, the Asynchronous Backtracking algorithm (ABT) [22] for DCSPs harnesses the partial knowledge of agents induced by neighborhood relations to speed up the search for an optimal solution. However, ABT is limited to satisfaction-based problems and cannot easily be transferred to optimization problems as discussed in the paper at hand. In [23], an extension for ABT for optimization is presented which can be applied only to limited types of optimization problems.

In [24], a self-organizing system of cooperative energy resource agents (primarily flexible loads) is proposed. The agents communicate indirectly by using a black-board like medium called "stigspace". The goal is to satisfy a globally known power supply cap. Similar to our Schooling Fish Algorithm, the agents adapt their load adjustments according to published values of other agents. Inter-agent variation is introduced by using randomized load shifts as an adjustment strategy. Our approach differs in that we equitably distribute the required power adjustments rather than making random choices. Also, in the DTAP given in this paper, inter-agent variation is inherent in the system rather than introduced by the algorithm. It is exploited by the Honey Bee Algorithm on the one hand, and coped with by the Schooling Fish Algorithm on the other hand.

A bottom-up approach to a supply and demand matching is proposed in [25]. Agents are organized in a dynamically built tree hierarchy using an overlay network on top of a decentralized peer-to-peer communication network. Here, each agent receives the set of possible energy utilization plans for a specific period of time (i.e., the feasible load schedules) from its child nodes in the tree hierarchy. From these sets, it selects the best combination of plans subject to a fitness function and informs the child nodes about its choice. These plans are then sent to the parent node together with the set of possible own plans, which in turn applies the same procedure. Eventually, the root node is reached where the resulting global plan emerges. This is similar to our Schooling Fish Algorithm since it also propagates information on the basis of neighborhood relations. Our approach, however, allows arbitrary topologies and does not rely on a hierarchy. Hence, we do not have a single point of failure and no performance bottleneck. We also make use of iterative adaptation to gradually improve an initial solution, whereas [25] uses a greedy bottom-up mechanism.

VII. Discussion and Future Work

In this paper, we presented two algorithms that solve a dynamic task allocation problem (DTAP) under uncertainty. On their basis, we examined how inter-agent variation affects the system's behavior. In our DTAP, a group of agents has the task to hold a global value at a fixed target value by adjusting their contribution. While the agents solve the task cooperatively, inter-agent variation is introduced by different thresholds of when agents adjust their contribution to satisfy the task. Uncertainty is introduced as the global value can change almost arbitrarily from one time step to another.

The Honey Bee Algorithm solves the DTAP by exploiting the system's inter-agent variation. It copes without communication between the agents or knowledge about the agents' state or contribution to the task. However, while our evaluations showed that the algorithm can achieve appealing results, their quality heavily depends on the amount of inter-agent variation available within the system. If the amount of inter-agent variation is inadequate, i.e., either too high or too low, either oscillations occur or the agents insufficiently change their contribution. Regarding the Honey Bee Algorithm, future work is to examine to what extent other execution models influence the algorithm's results. Furthermore, we will investigate if we can improve the results by allowing the agents to self-configure their parameters on the basis of a feedback mechanism.

In the Schooling Fish Algorithm, each agent knows the number of agents solving the DTAP. Moreover, agents can coordinate their actions by communicating within overlapping neighborhoods. The evaluations showed that the algorithm can solve the DTAP very accurately. In contrast to the Honey Bee Algorithm, the Schooling Fish Algorithm is not prone to oscillations because of its iterative annealing nature which is almost independent of the amount of inter-agent variation. Interestingly, the more inter-agent variation was available, the better the algorithm's results. Simulation results indicated that there even is a critical amount of inter-agent variation which helps the algorithm to find near-optimal solutions. Conversely, if the amount of inter-agent variation exceeded a certain point in the Honey Bee Algorithm, higher inter-agent variation led to a more sluggish behavior. Future work is to examine these critical amounts as well as the impact of different probability distributions, i.e., characteristics of inter-agent variation, on the system's behavior in more detail. Regarding the Schooling Fish Algorithm, different topologies should be analyzed, especially with respect to these properties, in order to find a possibly hidden dependency between network structure and the ability to cope with less inter-agent variation.

Acknowledgment

This research is partly sponsored by the research unit *OC-Trust* (FOR 1085) of the German Research Foundation. The authors would like to thank Jan-Philipp Steghöfer and Sebastian Lehnhoff for their feedback and the discussions.

References

[1] O. Shehory and S. Kraus, "Methods for task allocation via agent coalition formation," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 165–200, 1998.

[2] ——, "Feasible formation of coalitions among autonomous agents in nonsuperadditive environments," *Computational Intelligence*, vol. 15, no. 3, pp. 218–251, 1999.

[3] N. Griffiths and M. Luck, "Coalition formation through motivation and trust," in *Proc. of the 2nd Int. Joint Conf. on Autonomous Agents and Multiagent Systems*. ACM, 2003, pp. 17–24.

[4] W. Walsh and M. Wellman, "A market protocol for decentralized task allocation," in *Proc. of the International Conference on Multi Agent Systems, 1998*, jul 1998, pp. 325 –332.

[5] R. Smith, "The contract net protocol: High-level communication and control in a distributed problem solver," *IEEE Trans. on Computers*, vol. 100, no. 12, pp. 1104–1113, 1980.

[6] A. Campbell, C. Riggs, and A. Wu, "On the Impact of Variation on Self-Organizing Systems," in *Proc. of the Fifth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO), 2011*, oct. 2011, pp. 119 –128.

[7] M. Krieger, J. Billeter, and L. Keller, "Ant-like task allocation and recruitment in cooperative robots," *Nature*, vol. 406, no. 6799, pp. 992–995, 2000.

[8] S. Nouyan, R. Ghizzioli, M. Birattari, and M. Dorigo, "An insect-based algorithm for the dynamic task allocation problem," *Künstliche Intelligenz*, vol. 4, no. 05, pp. 25–31, 2005.

[9] UCTE, "UCTE Operation Handbook – Policy 1: Load-Frequency Control and Performance," Union for the Co-ordination of Transmission of Electricity, Tech. Rep. UCTE OH P1, 2009.

[10] S. McArthur, E. Davidson, V. Catterson, A. Dimeas, N. Hatziargyriou, F. Ponci, and T. Funabashi, "Multi-agent systems for power engineering applications–Part I: concepts, approaches, and technical challenges," *IEEE Trans. on Power Systems*, vol. 22, no. 4, pp. 1743–1752, 2007.

[11] M. Ilic, "From hierarchical to open access electric power systems," *Proc. of the IEEE*, vol. 95, no. 5, pp. 1060–1084, 2007.

[12] W. Ashby, "Requisite variety and its implications for the control of complex systems," *Cybernetica*, vol. 1, no. 2, pp. 83–99, 1958.

[13] J. C. Jones, M. R. Myerscough, S. Graham, and B. P. Oldroyd, "Honey Bee Nest Thermoregulation: Diversity Promotes Stability," *Science*, vol. 305, no. 5682, pp. 402–404, 2004.

[14] J.-P. Mano, C. Bourjot, L. Gabriel, and P. Glize, "Bio-inspired mechanisms for artificial self-organised systems," *Informatica*, vol. 30, pp. 55–62, 2006.

[15] C. Hinrichs, U. Vogel, and M. Sonnenschein, "Approaching Decentralized Demand Side Management via Self-Organizing Agents," in *Proc. of the 2nd Int. Workshop on Agent Technologies for Energy Systems (ATES 2011)*, 2011, pp. 31–38.

[16] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 25–34, 1987.

[17] S. H. Strogatz, "Exploring Complex Networks," *Nature*, vol. 410, no. March, pp. 268–276, 2001.

[18] D. Hammerstrom, J. Brous, D. Chassin, G. Horst, R. Kajfasz, P. Michie, T. Oliver, T. Carlon, C. Eustis, O. Jarvegren, W. Marek, R. Munson, and R. Pratt, "Pacific Northwest GridWise^{TM} Testbed Demonstration Projects: Part II. Grid Friendly^{TM} Appliance Project," Pacific Northwest National Laboratory, Richland, WA, Tech. Rep. PNNL-17079, 2007.

[19] J. Short, D. Infield, and L. Freris, "Stabilization of Grid Frequency Through Dynamic Demand Control," *IEEE Trans. on Power Systems*, vol. 22, no. 3, pp. 1284 –1293, aug. 2007.

[20] A. Kamper and A. Eßer, "Strategies for Decentralised Balancing Power," in *Biologically-inspired Optimisation Methods – Parallel Algorithms, Systems and Applications*, ser. Studies in Computational Intelligence, A. Lewis, S. Mostaghim, and M. Randall, Eds. Springer, 2009, vol. 210, pp. 261–289.

[21] A. Kamper and H. Schmeck, "Adaptives verteiltes Lastmanagement in Bilanzkreisen," *Informatik-Spektrum*, vol. 35, pp. 102–111, 2012.

[22] M. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara, "Distributed constraint satisfaction for formalizing distributed problem solving," in *Proc. of the Twelfth IEEE Int. Conf. on Distributed Computing Systems*, 1992, pp. 614–621.

[23] K. Hirayama and M. Yokoo, "An approach to over-constrained distributed constraint satisfaction problems: Distributed hierarchical constraint satisfaction," in *Proc. of Int. Conf. on Multiagent Systems*, 2000, pp. 135–142.

[24] J. Li, G. Poulton, and G. James, "Coordination of distributed energy resource agents," *Applied Artificial Intelligence*, vol. 24, no. 5, pp. 351–380, 2010.

[25] E. Pournaras, M. Warnier, and F. Brazier, "Local agent-based self-stabilisation in global resource utilisation," *Int. Journal of Autonomic Computing*, vol. 1, no. 4, pp. 350–373, 2010.