Consensus + Innovations Approach for Distributed Multi-Agent Coordination in a Microgrid

Gabriela Hug, *IEEE Member*, Soummya Kar, *IEEE Member*

***Abstract*—Distributed energy resources and demand side management are expected to become more prevalent in the future electric power system. Coordinating the increased number of grid participants in an efficient and reliable way is going to be a major challenge. A potential solution is the employment of a distributed energy management approach which uses intelligence distributed over the grid to balance supply and demand. In this paper, we specifically consider the situation in which distributed resources and loads form microgrids within the bulk power system in which load is supplied by local generation. A distributed energy management approach based on the consensus + innovations method is presented and used to coordinate local generation, flexible load and storage devices within the microgrid. The approach takes advantage of the fact that in the optimum the marginal costs given as a function of the power output/consumption needs to be equal for all the network entities (agents). Solutions for single time step as well as multi time step optimization including inter-temporal constraints are presented.**

***Index Terms*—Economic Dispatch, Consensus + Innovations Algorithm, Distributed Optimization, Multi-Step Optimization**

I. Introduction

The trend in the electric power system is to move towards more and more distributed generation resources, distributed storage capabilities and participation of the load in the generation/demand balancing process. This leads to a significant increase in the number of entities in the system which need to be coordinated and to electric energy often being generated closer to the loads. Hence, a possible structure of the future electric power system could consist of a number of self-sufficient cells of various sizes which internally coordinate their generation and load but also exchange or trade some amount of power with neighboring cells.

Such a self-sufficient cell is referred to as a microgrid [1]–[3]. Microgrids have the capability to disconnect from the main grid if needed and locally supply their loads. Prominent examples for microgrids include university campuses, military microgrids and islands. Similar to the bulk power system, the key questions are how to ensure the balance between generation and demand and how to achieve this in the most cost-effective way. Hence, the main focus of this paper is to design an efficient energy management system for a microgrid.

There are two fundamentally different approaches for the design of such an energy management system. One is to assign the responsibility of coordinating generation, demand, storage and main grid connection to a central entity, e.g. [4], [5]. An optimization problem is solved at the central location and signals are sent to the individual components. Another approach is based on multi-agent systems in which the decisions are made in a distributed way, e.g. [6], [7]. While any distributed approach can be implemented at a centralized location for the purpose of being able to parallelize computation and therefore improve computation speed, two situations in which physically distributed computations make sense include (1) when the participating entities do not want to share all of their operational information with any other entity and (2) when it is of importance to ensure that a failure of a single computational entity, i.e. the central coordinator, will not lead to an inability to control the system.

For distributed approaches in microgrids, research has mostly focused on setting up the multi-agent structure, ensuring interoperability to allow for plug-and-play capability and defining the communication structure. In this paper, we present an algorithm to be implemented in such a multi-agent structure and by which the participants in the microgrid coordinate their control settings in a distributed way. The proposed approach is based on the consensus + innovations method [8] and does not require any central coordinator or master agent. Such an algorithm forms the basis for realizing the plug-and-play capability of a microgrid. Agents are assigned to nodes to which generators, loads and/or storage devices are connected. These agents define incremental cost/demand functions and constraints for the local energy production and consumption. The consensus portion of the algorithm facilitates the agreement on an incremental price for the energy provided and the innovation portion ensures that total generation matches total demand [9].

Prior distributed approaches to schedule generation and/or load are mostly based on Lagrangian and Augmented Lagrangian Relaxation [10]. Applications of these methods to model predictive control in electric power systems include [11]–[13]. The approach presented in this paper is conceptually very different from these decomposition theory based approaches. It is based on obtaining a distributed iterative solution of the system of first order optimality equations (KKT conditions) associated with the constrained optimization problem. Specifically, by exploiting the special structure of the optimality equations, we show that the problem of obtaining optimal generator allocations can be reduced to a *distributed restricted agreement* problem – at any given stage the optimal generator allocations are uniquely determined by a single parameter which coincides with the marginal price of generation at the non-binding generators, i.e., the generators which do not reach their capacity limits at the optimal allocation. We propose a consensus + innovations approach to ensure that the generators reach an agreement on this parameter. Another key difference, as far as implementation is concerned, is that in the decomposition based approaches, the coordinating entities actually need to solve a local optimization problem whereas in

The authors are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, 15213, USA, e-mail: ghug@ece.cmu.edu, soummyak@andrew.cmu.edu

the presented approach the computational effort of each entity is limited to evaluating simple algebraic variable updates and projecting the values into the feasible space for these variables.

The most relevant related work with regards to consensus based methods has been presented in [14]–[17] where a decentralized economic dispatch approach based on the consensus algorithm has been introduced. The two key differences between our approach and this work are that we do not assign any leading role (for achieving global coordination) to any of the generators and we consider a multi-step optimization including inter-temporal constraints allowing for optimal integration of storage devices and consideration of generation ramp rates. Usage of the consensus algorithm for the purpose of ancillary service provision is presented in [18]. However, the focus is on ensuring resilience against potential packet drops and it also uses a coordinator which determines the total amount of required power. Compared to our earlier work [9], where we employed the consensus + innovations approach to derive a distributed economic dispatch algorithm, we extend the approach to optimize over multiple time steps enabling distributed Model Predictive Control and we include flexible loads and storage devices as controllable components.

The remaining part of the paper is structured as follows: Sect. II introduces the problem formulation for single and multi-step economic dispatch. In Sect. III, the proposed distributed approach based on the consensus + innovations algorithm is derived. Sect. IV discusses the robustness of the algorithm. Sect. V provides simulation results and Sect. VI concludes the paper.

II. Problem Formulation

We consider a microgrid which includes dispatchable and non-dispatchable generators, critical/inflexible and non-critical/flexible loads and storage devices. A storage could be a battery but also plug-in electric vehicles which are available only intermittently. In this section, we provide the mathematical problem formulation which we will use subsequently in the next section to derive the distributed algorithm. First, we focus on a single time step formulation and then extend it to include inter-temporal constraints in a multi-step formulation.

A. Single Time Step

We assign a quadratic cost/demand function to each component n given by

$$C_n(P_n) = \frac{1}{2} a_n P_n^2 + b_n P_n + c_n \tag{1}$$

with $a_n, b_n, c_n \geq 0$ and $P_n > 0$ if the power is generated or injected into the system and $P_n < 0$ if it is consumed or drawn from the system. For generators, the function reflects the costs of producing the power P_n whereas for loads, it is the (negative) cost the load is willing to pay for power $|P_n|$. For a storage device, P_n is positive whenever the storage is discharging, i.e. the function corresponds to the amount the storage is willing to accept for the provision of P_n, and it is negative whenever the storage is charging which corresponds to acting like a load. The power P_n is upper and lower bounded

$$\underline{P}_n \leq P_n \leq \overline{P}_n \tag{2}$$

with $\underline{P}_n, \overline{P}_n \geq 0$ for generators, $\underline{P}_n, \overline{P}_n \leq 0$ for loads and $\underline{P}_n \leq 0, \overline{P}_n \geq 0$ for storage devices. Inflexible loads and "must take" generation such as from PV or wind generation can be modeled by setting upper and lower limits equal to each other resulting in the corresponding P_n variables becoming constants instead of optimization variables.

The objective is to determine the settings of the components, i.e. P_n's, that maximize the (concave) social welfare given by

$$SW = -\sum_{n=1}^{N} C_n(P_n) \tag{3}$$

In addition, the power balance

$$\sum_{n=1}^{N} P_n = 0 \tag{4}$$

and the upper and lower bounds on the power injected/consumed as defined in (2) for $n = 1, \ldots, N$ need to be fulfilled where N is the total number of components to be coordinated.

It can be derived from the first order optimality conditions of this optimization problem that the following conditions need to hold for the optimal solution: the marginal costs at the solution

$$dC_n(P_n)/dP_n = a_n P_n + b_n \doteq \lambda_n \tag{5}$$

for all system components n for which P_n has not reached the upper or lower limit have to be equal to the same value, namely the *system price* λ^* at the optimal solution, and the power balance (4) needs to be fulfilled. Components for which the optimal setting is $\overline{P}_n$, yield a marginal cost as defined in (5) that is lower than λ^* and components for which the optimal setting is $\underline{P}_n$, yield a marginal cost that is greater then λ^*. The system price λ^* is also the Lagrange Multiplier associated with the power balance equation (4).

Nonetheless, assuming that the primal problem admits a feasible solution, it may be shown (see [9]) that the optimal settings at all the system entities can be uniquely parameterized in terms of the quantity λ^*, in that, the optimal setting P_n^* at an entity n is given by

$$P_n^* = \mathcal{P}_n \left[\frac{\lambda^* - b_n}{a_n} \right], \tag{6}$$

where $\mathcal{P}_n[\cdot]$ denotes the projection operator associated with entity n, i.e., it projects the argument into the feasible solution space $[\underline{P}_n, \overline{P}_n]$.

Now, given the power balance constraint (4), the goal of a distributed algorithm can be formalized as a *restricted agreement problem*, in which the entities seek to reach an agreement on the quantity λ^* that satisfies (see also [9]):

$$\sum_{n=1}^{N} P_n^* = \sum_{n=1}^{N} \mathcal{P}_n \left[\frac{\lambda^* - b_n}{a_n} \right] = 0. \tag{7}$$

Furthermore, note that each entity n is only aware of its local marginal cost/demand function parameters and capacity constraints, and hence, cannot directly solve (7). Hence, the need for collaboration through inter-entity information exchange arises, which motivates our distributed algorithm in Sect. III to determine λ^* satisfying (7) at each entity.

For illustrative purposes, characteristic marginal cost curves for generation, load and storage are shown in Fig. 1.

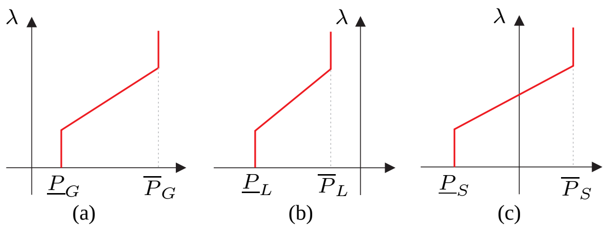

Fig. 1. Marginal cost functions for (a) generator, (b) load and (c) storage.

B. Multiple Time Steps

When introducing storage into the system, it is indispensable to optimize over multiple time steps concurrently because not only instantaneous power is limited but also energy. In order to take into account these inter-temporal constraints and to make efficient use of the available storage capacity, we extend the single step problem to a multi-step optimization problem. This allows us to also include other inter-temporal constraints such as ramping limitations on generation output. Hence, the problem formulation is given by

$$\max_{P_{n,k}} \sum_{k=0}^{K-1} \sum_{n=1}^{N} (-C_{n,k}(P_{n,k})) \tag{8}$$

$$s.t. \qquad \sum_{n=1}^{N} P_{n,k} = 0, \qquad k = 0, \ldots, K-1 \tag{9}$$

$$\underline{P}_{n,k} \leq \quad P_{n,k} \quad \leq \overline{P}_{n,k}, \; k = 0, \ldots, K-1 \tag{10}$$

$$\Delta\underline{P}_n \leq \; P_{n,k} - P_{n,k-1} \leq \Delta\overline{P}_n, \; k = 0, \ldots, K-1 \tag{11}$$

and for storage devices additionally

$$\underline{E}_n \leq \; E_{n,0} - \sum_{l=0}^{k} P_{n,l} \leq \overline{E}_n, \quad k = 0, \ldots, K-1 \tag{12}$$

where $P_{n,k}$ is the power output of component n at time step k and K is the number of steps considered in the multi-step optimization. Constraint (9) corresponds to the power balance at each step, (10) is the limit on instantaneous power injection/consumption, (11) corresponds to ramping limits with lower and upper ramping limits $\Delta\underline{P}_n, \Delta\overline{P}_n$ and (12) incorporates the upper and lower limits $\underline{E}_n, \overline{E}_n$ on stored energy $E_{n,k}$. The initial values for power injection/withdrawal are denoted by $P_{n,-1}$ and $E_{n,0}$, respectively.

Furthermore, we add a terminal constraint

$$E_{n,N} = E_{n,0} - \sum_{l=0}^{K-1} P_{n,l} = E_n^f \tag{13}$$

for the storage which ensures that the energy level $E_{n,N}$ at the end of the optimization horizon is equal to a fixed value E_n^f, e.g. $E_n^f = 0.5 \cdot \overline{E}_n$. Adding such a terminal constraint in essence corresponds to adjusting the cost function parameter b_n. This value b_n is equal to the price below which the storage charges and above which it discharges.

The optimal solution to the above problem is given by

$$\sum_{n=1}^{N} P_{n,k} = 0, \tag{14}$$

and marginal costs $\lambda_{n,k}, \; k = 1, \ldots, K$ being equal to the optimal system prices, namely $\lambda_k^*, \; k = 1, \ldots, K$. Power outputs are given by

$$P_{n,k} = \frac{\lambda_k^* - b_n}{a_n} \tag{15}$$

and projecting it into the feasible solution space defined by constraints (10) – (13) applicable to component n. Here, λ_k^* is the marginal cost / price at time step k of the horizon at the optimum which is also equal to the Lagrange Multiplier associated with the power balance equation at time step k in the optimization problem and $\lambda_{n,k}$ is a local copy of this variable of component n.

III. Distributed Scheduling

It is assumed that generators, loads and storage devices are connected to nodes in the microgrid and an agent is assigned to each node. The proposed algorithm by which the agents coordinate is based on the consensus + innovations approach. In this section, we first give a general introduction to this approach, describe how it is used in the single step optimization and then extend it to the multi-step case.

A. Distributed Decision-Making: Consensus + Innovations

We briefly review the consensus + innovations method and its variants, a generic approach for solving distributed decision-making problems in multi-agent networks, e.g. [8]. The decision-making setups that fall under the purview of consensus + innovations typically involve collaborative distributed information processing such as estimation, optimization and control in agent networks, in which each network agent has a priori access to only local information, such as knowledge of model parameters and sensed data, and inter-agent communication (interaction) is restricted to a pre-assigned sparse communication graph. Broadly speaking, in the consensus + innovations architecture, the autonomous network agents or decision-makers engage in local information processing and neighborhood communication to achieve or optimize the global decision-making task of interest.

For definiteness in this paper we restrict the discussion of the consensus + innovations method to the distributed *restricted agreement problem* in multi-agent distributed networks. Formally, in an information processing network, of J agents, the restricted agreement problem consists of having the J agents agree on a common value ν subject to the equality constraint (restriction)

$$g(\nu) = \sum_{j=1}^{J} h_j(\nu) = \sum_{j=1}^{J} \sum_{n \in \Omega_j} d_n(\nu) = 0 \tag{16}$$

and inequality constraints

$$\underline{d}_n \leq d_n(\nu) \leq \overline{d}_n, \quad n \in \Omega_j, \; j = 1, \cdots, J, \tag{17}$$

where $\sum_{j=1}^{J} |\Omega_j| = N$ for some positive integer N, i.e., the sets $\{\Omega_j\}_{j \in J}$ constitute a partition of $[1, \cdots, N]$, $d_n(\cdot), \; n = 1, \cdots, N$ are certain real-valued functions, and $\underline{d}_n, \overline{d}_n \in [-\infty, \infty]$, for $n = 1, \cdots, N$, are constants. Moreover, we are interested in a distributed solution of the restricted agreement problem in which, (i) to start with, each agent j is only aware of its set of *local* functions $d_n(\cdot),\ n \in \Omega_j$ and the corresponding inequality constraints (17), and (ii) inter-agent communication for information exchange is restricted to a pre-assigned communication graph. Under broad assumptions on the local functions $d_n(\cdot)$'s and the inter-agent communication topology (to be made precise later) an iterative algorithm of the consensus + innovations type may be applied to solve the above distributed restricted agreement problem.

Before proceeding to the general consensus + innovations solution, as a side remark, we comment on a specific instance of the above distributed restricted agreement problem in which the (aggregate) local functions $h_j(\cdot)$'s are affine functions of the form $h_j(\nu) = \nu - x_j$ where the x_j's are real constants and $\underline{d}_n = -\infty$ and $\overline{d}_n = \infty$ for all $n \in \Omega_j$ and $j = 1, \cdots, J$, i.e., the inequality constraints are relaxed. In other words, the agents want to agree on the average value $(1/J)\sum_{j=0}^{J} x_j$. Historically, this version of the restricted agreement problem is referred to as the average consensus problem or simply the agreement problem and has been studied extensively over the last few years, e.g. see the review papers [19], [20]. The problem with generic functions $d_n(\cdot)$'s and inequality constraints is more involved and in the following we describe the consensus + innovations method [8] for a general solution methodology.

In the consensus + innovations method, each agent j maintains a local copy $\nu_j(i)$ of the variable ν which is iteratively updated, with i denoting the iteration index, as follows:

1) Update local copy of common variable according to

$$\begin{aligned}\nu_j(i+1) &= \nu_j(i) - \beta_i \sum_{l\in\omega_j} (\nu_j(i) - \nu_l(i)) \\ &\quad - \alpha_i \sum_{n\in\Omega_j} \hat{d}_n(i) \end{aligned} \quad (18)$$

where α_i and β_i are weight parameters, ω_j is the communication neighborhood of agent j as prescribed by the given inter-agent communication topology, i.e. the subset of network agents with which agent j can exchange information directly, and

$$\hat{d}_n(i) = \mathcal{P}_n\left[d_n(\nu_j(i))\right], \quad n \in \Omega_j, \quad (19)$$

where $\mathcal{P}_n[\cdot]$ denotes the projection operator onto the interval $[\underline{d}_n, \overline{d}_n]$;

2) Update dependent variables according to (19) to obtain $\hat{d}_n(i+1),\ n \in \Omega_j$;
3) Exchange local value $\nu_j(i+1)$ with agents in the communication neighborhood ω_j and increase iteration counter i.

Typical conditions that ensure convergence, i.e., $\nu_j(i) \to \nu$ as $i \to \infty$ for all j with ν satisfying (16)-(17), are as follows (see, e.g. [8]):

- The local functions $d_n(\cdot)$'s are sufficiently regular[1].
- The inter-agent communication network needs to be connected[2].
- The weight parameters α_i, β_i are positive and satisfy the following conditions:
  - As $i \to \infty$, the sequences $\{\alpha_i\}$ and $\{\beta_i\}$ are decaying, i.e., $\alpha_i \to 0, \beta_i \to 0$.
  - The excitations are persistent, i.e.,

$$\sum_{i\geq 0} \alpha_i = \sum_{i\geq 0} \beta_i = \infty \quad (20)$$

  - The consensus potential asymptotically dominates the innovation potential, i.e., $\beta_i/\alpha_i \to \infty$ as $i \to \infty$.

[1]Several regularity conditions of the form of Lipschitz continuity, monotonicity etc. are presented in [8] that ensure convergence.

[2]By connectivity, we mean that there exists a path, possibly multi-hop, between any pair of agents. As such, the communication network may be quite sparse and in fact, much sparser, than the physical grid topology which is typically dense.

B. Single Step

In the single step application, only the scheduling for one time step is considered. To this end, let $j = 1, \cdots, J$ denote the nodes (agents) in the microgrid and let $n \in \Omega_j$ index the components (generators, loads and storage devices) connected to node j. Also, let $N = \sum_{n\in\Omega_j} |\Omega_j|$ be the total number of microgrid components. The common variable ν that the agents need to agree on corresponds to the marginal cost of supply λ for that time step whereas the constraint $g(\nu)$ which needs to be fulfilled is the power balance between supply and demand. Specifically, according to (7) and the development in Sect. III-A, the local component function $\hat{d}_n(\lambda)$ is the power P_n injected/drawn by component n which can be given as a function of λ by

$$\hat{d}_n(\lambda) = P_n(\lambda) = \mathcal{P}_n\left[\frac{\lambda - b_n}{a_n}\right], \quad (21)$$

where $\mathcal{P}_n[\cdot]$ is the projection operator corresponding to the local capacity constraints which as explained later are adjusted from (7) to also include generator ramp rate and energy storage constraints for this single time step.

According to (18), each agent j carries out the following iterative calculations

$$\begin{aligned}\lambda_j(i+1) &= \lambda_j(i) - \beta_i \sum_{l\in\omega_j} (\lambda_j(i) - \lambda_l(i)) \\ &\quad - \alpha_i \sum_{n\in\Omega_j} P_n(i) \end{aligned} \quad (22)$$

Hence, the part of the innovation term assigned to agent j corresponds to the sum of power injections/consumptions of the components connected to node j, i.e. $n \in \Omega_j$.

Then, the power injected/consumed $P_n(i)$ for $n \in \Omega_j$ is updated according to

$$P_n(i+1) = \begin{cases} \frac{\lambda_j(i+1)-b_n}{a_n}, & P_n^{min} < \frac{\lambda_j(i+1)-b_n}{a_n} < P_n^{max} \\ P_n^{max}, & \frac{\lambda_j(i+1)-b_n}{a_n} \geq P_n^{max} \\ P_n^{min}, & \frac{\lambda_j(i+1)-b_n}{a_n} \leq P_n^{min} \end{cases} \quad (23)$$

where P_n^{min} and P_n^{max} are defined as in Table I. Hence, the upper and lower capacity constraints are adjusted to take into account generation ramp rate and energy level constraints for this one single time step. The initial generator output and storage energy level are denoted by $P_{n,-1}$ and $E_{n,0}$, respectively, and T is the length of one time step.

TABLE I
INCORPORATION OF INTER-TEMPORAL CONSTRAINTS INTO UPPER AND LOWER BOUND FOR POWER INJECTION/WITHDRAWAL.

	P_n^{min}	P_n^{max}
Gen.	$\max(\underline{P}_n, P_{n,-1} + \Delta\underline{P}_n)$	$\min(\overline{P}_n, P_{n,-1} + \Delta\overline{P}_n)$
Load	$\underline{P}_n$	$\overline{P}_n$
Stor.	$\max(\underline{P}_n, \frac{1}{T}(E_{n,0} - \overline{E}_n))$	$\min(\overline{P}_n, \frac{1}{T}(E_{n,0} - \underline{E}_n))$

Note that the inter-agent communication topology (i.e., who talks to whom) may differ significantly from the physical power system electrical coupling among the agents and is, in fact, typically much sparser than the physical coupling topology. In particular, we only assume that there exists a path comprising, perhaps, of multiple communication hops between any two pair of agents, i.e., the inter-agent communication network is *connected*. Moreover, note that, in the update rule (22) each agent needs to be aware of its local model parameters (a_n's and b_n's) only; this is unlike a centralized optimization approach in which each agent needs to communicate and coordinate with a fusion center, the latter having access to the model parameters of all the agents. Interestingly enough, we show that, even under such restrictions on agent communication and lack of global model information, the proposed update rule converges to the optimal schedules at each agent (see also [9]).

C. Multi-Step

In the multi-step application as described in Sect. II-B, a horizon of multiple time steps in the future is considered concurrently. This results in an overall decrease in cost of supply as preventive actions such as charging the storage device in anticipation of needed energy in the future can be taken. In this case, the consensus + innovations application becomes multi-dimensional, i.e., $\boldsymbol{\lambda}_j$ and $\boldsymbol{P}_n,\ n \in \Omega_j$ are vectors including the prices/marginal costs and the power output/consumption for the time steps within the horizon, respectively. The update given in (22) still applies but now as a vector update resulting in the following update for each time step k

$$\lambda_{j,k}(i+1) = \lambda_{j,k}(i) - \beta_i \sum_{l \in \omega_j} (\lambda_{j,k}(i) - \lambda_{l,k}(i)) - \alpha_i \sum_{n \in \Omega_j} P_{n,k}(i) \quad (24)$$

with $k = 0, \ldots, K-1$.

The projection into the solution space can be achieved by the following constrained least square minimization for each component $n \in \Omega_j$[3]

$$\min_{\boldsymbol{P}_n} \quad \sum_{k=0}^{K-1} \left(P_{n,k} - \frac{\lambda_{j,k} - b_n}{a_n} \right)^2 \quad (25)$$

$$s.t.\ \ \underline{P}_n \le P_{n,k} \le \overline{P}_n, \quad k = 0, \ldots, K-1 \quad (26)$$

$$\Delta\underline{P}_n \le P_{n,k} - P_{n,k-1} \le \Delta\overline{P}_n, k = 0, \ldots, K-1 \quad (27)$$

$$\underline{E}_n \le E_{n,0} - \sum_{l=1}^{k} P_{n,l} \le \overline{E}_n, \quad k = 0, \ldots, K-1 \quad (28)$$

[3]For simplicity, the iteration counter $(i+1)$ is omitted in $P_{n,k}(i+1)$ and $\lambda_{j,k}(i+1)$.

$$E_{n,N} = E_{n,0} - \sum_{l=0}^{K-1} P_{n,l} = E_n^f \quad (29)$$

Constraints (28) and (29) are only applicable if component n is a storage device. If the component is a load and ramping of the load is unlimited, then constraint (27) can either be neglected or limits are set to infinity.

The same statements concerning communication topology and connectivity hold as in the single step application.

D. Overview Application Flow

The application of the multi-step consensus approach is visualized in Fig. 2. At time $t = t_1$ the goal is to determine the optimal settings $P_{n,k}$ for the generators, loads and storage devices for the next $k = 0, \ldots, K-1$ steps fulfilling all constraints on ramp rates, maximum generation output, energy level, etc. Hence, the optimization horizon corresponds to $t = t_1, \ldots, t_1 + K$. The consensus algorithm as described in Sect. III-C is used to determine these settings with i corresponding to the iteration counter. Once the agents have agreed on $\lambda_{j,k}, k = 0, \ldots, K-1$ and optimal settings $P_{n,k}, k = 0, \ldots, K-1$ have been found, the first step $P_{n,1}$ is applied, the optimization horizon is moved by one time step and the consensus algorithm is restarted for t_2.

The choice of the initial starting point in the iterative process of the consensus algorithm has great influence on the convergence of the algorithm. In the considered application for the coordination in a microgrid, it can be assumed that most of the time the external inputs such as the parameters a_n, b_n and any upper and lower limits do not change drastically from one time step to the next. Hence, a reasonable approach is to use the solution determined at the previous time step $t-1$ as a starting point for the time step t.

Figure 3 gives an overview over the entire application flow where the solution for time step t is denoted by $\boldsymbol{\lambda}_j^t, \boldsymbol{P}_n^t$. In the very first time step $t = 0$, no solution $\boldsymbol{\lambda}_j^{t-1}, \boldsymbol{P}_n^{t-1}$ for the previous time step is available to use as starting point in the consensus algorithm. Hence, an initialization is required,

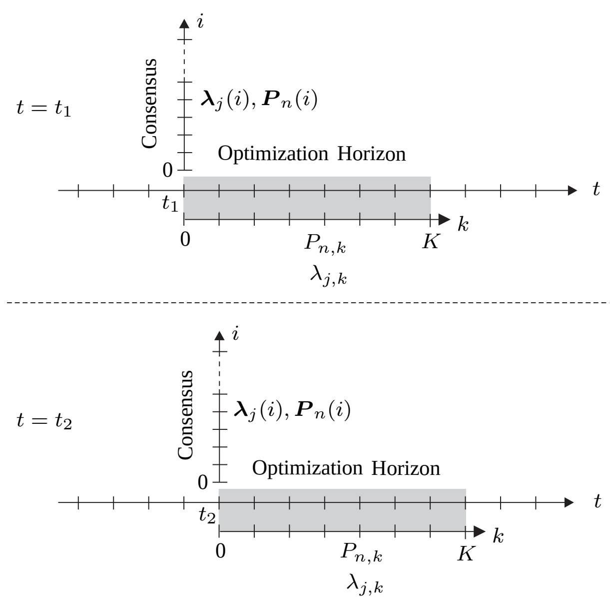

Fig. 2. Visualization of multi-step consensus algorithm.

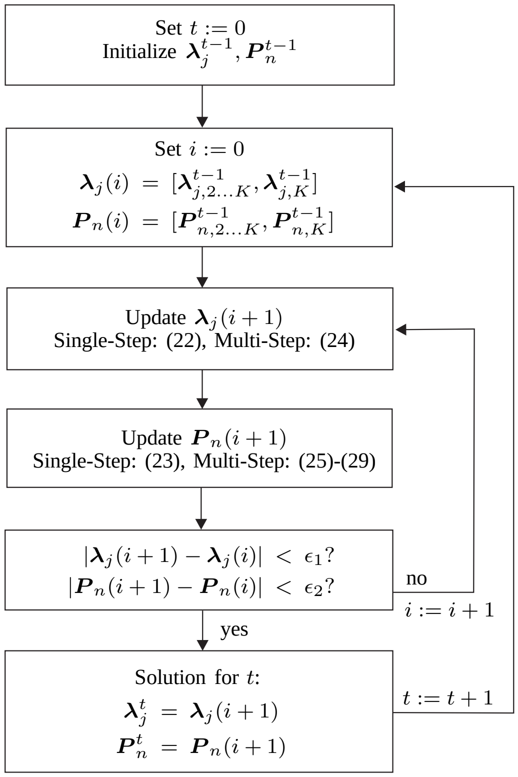

Fig. 3. Flow chart consensus update.

e.g. by using a flat start or by using historic solution data. In the following time steps t, $\boldsymbol{\lambda}_j^{t-1}, \boldsymbol{P}_n^{t-1}$ corresponds to the solution at the previous step $t-1$. An advantage of the multi-step application is that good initial settings for $k = 0, \ldots, K-1$ are available from the solution at the previous time step. Only the initial settings for the newly included time step $k = K$ need to be found. We make the assumption that the settings for $k = K$ will be close to the settings at $k = K-1$ resulting in initial settings

$$\boldsymbol{\lambda}_j(0) = [\boldsymbol{\lambda}_{j,2\ldots N}^{t-1}, \boldsymbol{\lambda}_{j,N}^{t-1}] \tag{30}$$

$$\boldsymbol{P}_n(0) = [\boldsymbol{P}_{n,2\ldots N}^{t-1}, \boldsymbol{P}_{n,N}^{t-1}] \tag{31}$$

for the consensus algorithm. Then, the iterative process of the consensus approach is carried out, i.e. $\boldsymbol{\lambda}_j, \boldsymbol{P}_n$ are updated and after each iteration the stoping criteria is checked. If changes in $\boldsymbol{\lambda}_j$ and $\boldsymbol{P}_n$ are lower than ϵ_1 and ϵ_2, respectively, the consensus iterations are stopped and restarted for a shifted optimization horizon with initial values given by (30) and (31).

For the single step approach, the procedure is very similar to setting the horizon equal to $K = 1$. In that case, λ_j, P_n are scalars and the initial values for the consensus algorithm correspond the values obtained at the previous time step.

IV. Robustness

The proposed distributed scheduling algorithms of the consensus + innovations type are robust to a wide class of perturbations resulting from intermittent inter-agent communication failures, noisy or quantized data exchange, and other forms of randomness in the communication infrastructure. For instance, due to data-packet scheduling and interference issues in multi-agent scenarios involving wireless agent-to-agent communication or infrastructure failures in wired communication environments, the designated communication links may not be active at all instants; moreover, even in the event of an active communication link between a pair of agents, the transmitted data may be noisy or distorted due to quantization and other channel effects. The proposed consensus + innovations scheduling structure stays valid under a wide range of such communication imperfections; for instance, by adding the square summability requirement $\sum_{i\geq 0} \alpha_i^2 < \infty$ on the innovations weight sequence $\{\alpha_i\}$, the effect of independent and identically distributed additive communication noise may be mitigated. For random communication link failures, as long as the network is *connected in the mean*, the convergence of the proposed iterative procedure to the optimal will be retained, see, for example, [8].

V. Simulations

In this section, simulation results are provided as a proof of concept. Due to limited space, we focus on simulations for the multi-step application. For single step simulations, the reader is referred to [9].

A. Simulation Setup

The considered microgrid consists of 14 nodes as shown in Fig. 4. Dispatchable generators are located at buses 1, 2, 3, 6 and 8, a wind generator is placed at bus 9 and a storage device is placed at bus 7. The connections indicate communication lines between components in the microgrid. The test system has been derived from the IEEE 14 bus system using the physical connections as indications for communication lines. However, physical and communication connections do not necessarily have to coincide.

The parameters for generators, loads and storage are given in Tables II - IV. It is assumed that the minimum generation level for all dispatchable generators is 0pu and that the ramp up and down limits are equal, i.e. $\Delta \underline{P}_n = -\Delta \overline{P}_n$. The simulations are carried out for an entire day in 5 minute intervals and the prediction horizon K is chosen to be equal to 20 time steps.

It can be assumed that the parameters for the generators and storage stay the same over a longer time period whereas

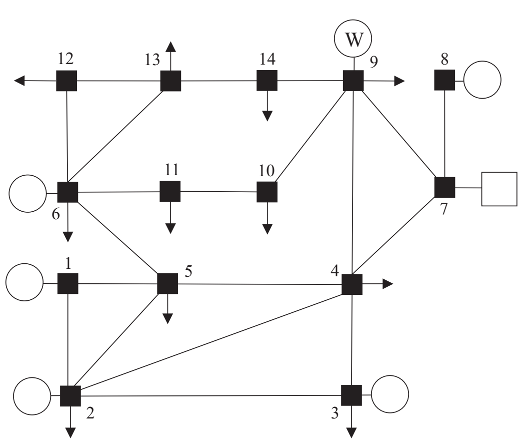

Fig. 4. Test System.

TABLE II
GENERATOR PARAMETERS (MU = MONETARY UNITS)

Bus	a_n [mu/pu^2]	b_n [mu/pu]	$\Delta\overline{P}_n$ [pu/5min]	$\overline{P}_n$ [pu]
1	0.084	2.0	20	100
2	0.056	3.0	5	105
3	0.070	4.0	10	100
6	0.060	4.0	50	90
8	0.080	2.5	15	80

TABLE III
LOAD PARAMETERS (MU = MONETARY UNITS)

Bus	a_n [mu/pu^2]	b_n [mu/pu]	$\underline{P}_n$ [pu]	$\overline{P}_n$ [pu]
2	0.080	7.5	-50	-5
3	0.060	6.0	-40	-3
4	0.070	8.0	-80	-6
5	0.064	6.5	-30	-2
6	0.040	7.5	-70	-8
9	0.060	8.0	-80	-10
10	0.076	7.0	-40	-5
11	0.070	7.5	-25	-2
12	0.080	8.0	-90	-8
13	0.070	7.0	-30	-2
14	0.084	8.0	-80	-10

TABLE IV
STORAGE PARAMETERS (MU = MONETARY UNITS)

Bus	a_n [mu/pu^2]	$\Delta\overline{P}_n$ [pu]	$\overline{P}_n$ [pu/5min]	$\overline{E}_n$ [puh]
7	0.02	50	50	20

the parameters for the load change to reflect varying needs over the course of the day. Each load is an aggregation of multiple loads, including flexible and inflexible loads. Varying needs for such an aggregated load are modeled by adjusting the parameter b_n in the demand function and the minimum and maximum demand limits $\underline{P}_n$ and $\overline{P}_n$, respectively. It is assumed that the agent of each load predicts the parameters required to cover its consumption and then schedules the load according to the outcome of the distributed energy balancing management. The upper limits $\overline{P}_n$, i.e. lowest absolute values for the demand, correspond to the portion of the fixed loads.

As predictive optimization is employed, the agents need to predict the consumption for the entire prediction horizon K but these predictions may be inaccurate. Hence, such uncertainty is simulated by introducing prediction errors. To simulate the varying demand over the day, we multiply $b_n, \underline{P}_n, \overline{P}_n$ with the values given in Fig. 5. To distinguish between prediction and actually occurring load, the bold line in Fig. 5 gives the multiplication factor which actually occurs and the dotted line what has been predicted. Hence, $b_n, \underline{P}_n, \overline{P}_n$ are multiplied with the values of the dotted line for the predictions and multiplied with the values of the bold line for the actual realizations of the load. It is assumed that predictions improve the closer the predicted time step to the current time step, i.e. for the first five time steps $k = 0, \ldots, 4$ in the prediction horizon the errors are reduced from what is shown in Fig. 5 inversely proportionally to k.

Similarly, also the output from the non-dispatchable generator at bus 9 needs to be predicted. The actual and the predicted power output over the entire day is given in Fig. 6. Again, it is assumed that predictions for the near term future are more accurate than the predictions for the rest of the optimization horizon. Hence, again, for the first five time steps $k = 0, \ldots, 4$ in the prediction horizon the prediction errors are reduced inversely proportionally to k.

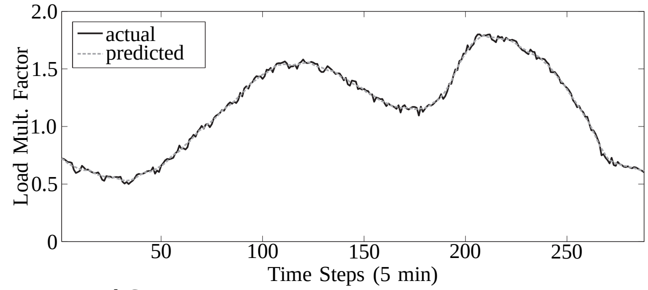

Fig. 5. Load Curve.

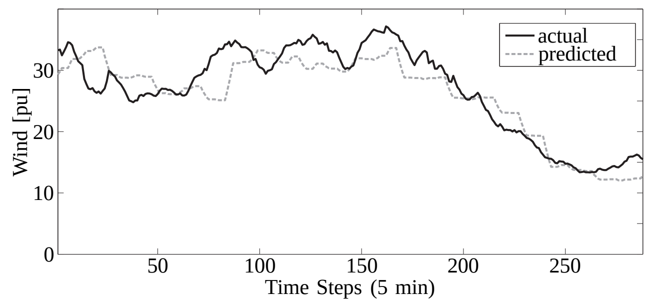

Fig. 6. Wind Curve.

The tuning parameters α and β are set to

$$\alpha_i = \frac{0.055}{i^{0.98}}, \qquad \beta_i = \frac{0.2}{i^{0.001}} \tag{32}$$

where i is the iteration counter. Stopping tolerances as defined in Fig. 3 are set to

$$\epsilon_1 = 0.0001, \qquad \epsilon_2 = 0.001. \tag{33}$$

B. Simulation Results: Normal Operation

The number of iterations required to converge given the convergence criteria and simulation parameters defined in the previous section are given in Fig. 7. Depending on the time step, the algorithm converges within 50 to 250 iterations. The computational effort required at each of these iteration steps is limited to evaluating the algebraic equations (24) and projecting the values of the variables into the feasible solution space. Hence, computationally each iteration step is very inexpensive. Compared to approaches based on decomposition theory, the number of iterations of the proposed approach is most likely higher, however the computational complexity at each individual iteration is significantly lower.

High accuracy has been chosen as convergence criterion. If accuracy is to be improved even further, more iteration steps will be necessary. On the other hand, if accuracy can be reduced, also the number of required iterations reduces. It should be noted that the same settings for α_i and β_i as defined in (32) are used for each time step t. The values decay over the iterations but the initial values and the speed of decay is the same for every simulated time step. Hence, making these settings adaptive for different levels of loading based e.g. on learning would improve convergence.

It is also possible to choose fixed, non-decaying values for α and β which will speed up convergence, in which case the algorithm in fact converges exponentially fast, however, this will result in a deviation from the optimal solution. The magnitude of the error is a function of the chosen values for α and β [9]. Furthermore, it is possible to trade-off convergence

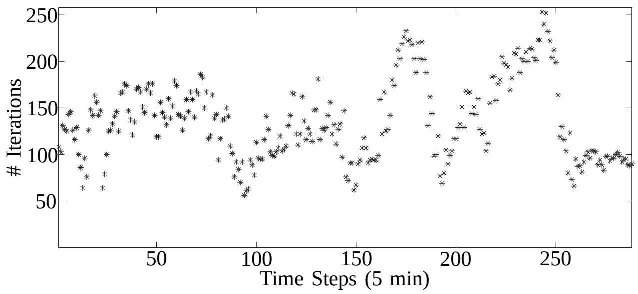

Fig. 7. Iteration steps required to converge.

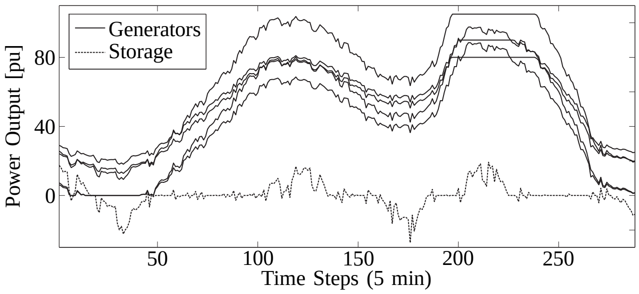

Fig. 8. Power output from generation and storage.

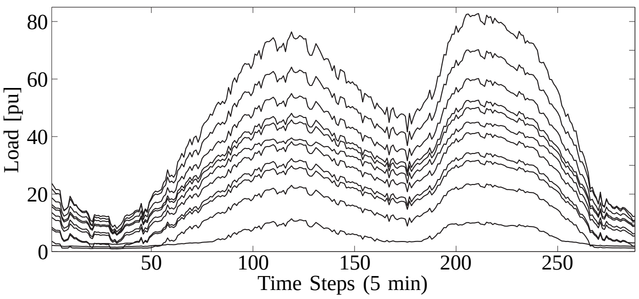

Fig. 9. Consumption by the loads.

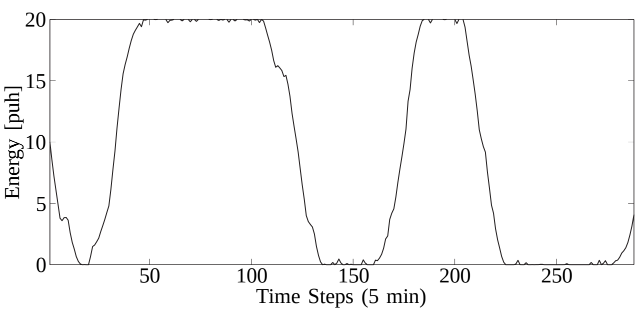

Fig. 10. Energy level in the storage.

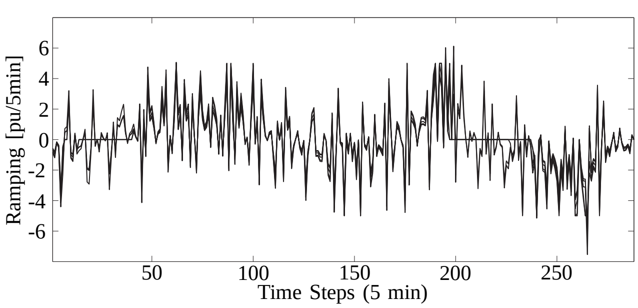

Fig. 11. Ramping of generators.

rate with accuracy, i.e., by properly selecting the fixed values of α and β, the deviation from the optimal solution can be made arbitrary small at the cost of slower convergence rate.

Figure 8 shows the generation and the storage power output settings over the course of the simulated time range. It can be seen that the proposed algorithm is capable of successfully taking into account upper and lower limits on generation capacity. Figure 9 provides the consumption by the individual loads. At the beginning and the end of the simulation there are multiple time steps during which loads reach their lower limits indicating that only inflexible/fixed loads are supplied at these nodes. In Fig. 10, the resulting level of energy in the storage device is given. The storage is mostly used for arbitrage, i.e. charging in low load situations and discharging in high load situations allowing for a reduced required generation capacity. Furthermore, it provides support in balancing out some of the short term variability. Figure 11 provides the ramping of the generators, indicating that constraints on ramp rates are met.

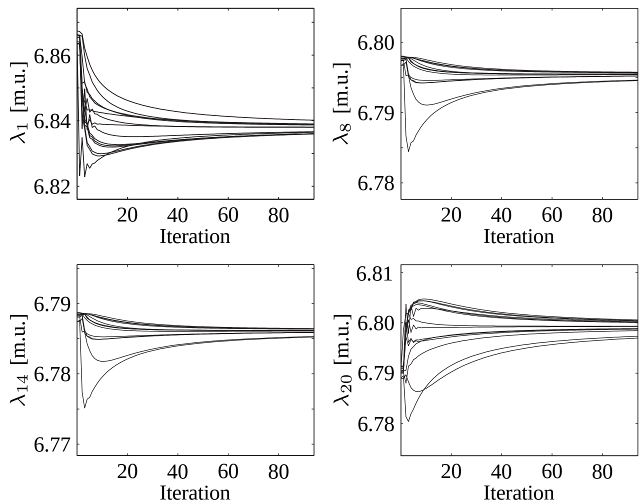

Fig. 12. Evolution of λ's for all nodes for prediction steps $k = \{1, 8, 14, 20\}$ at time step $t = 160$.

To provide insight into the convergence behavior, Fig. 12 shows the evolution of $\boldsymbol{\lambda}$ for the consensus iterations at time step $t = 160$. The dimension of $\boldsymbol{\lambda}$ for each node j is equal to the prediction horizon. Here, we show the λ's at all nodes for steps $k = 1, 8, 14, 20$ in the prediction horizon.

C. Simulation Results: Generator Disconnect

An advantage of the fully distributed approach is that it allows for a plug-and-play mechanism. Hence, in the following simulation, the performance of the proposed algorithm is tested for the case in which the generator at node 3 is disconnected at time step $t = 80$ and reconnected at time $t = 90$. Fig. 13 gives the number of iterations to converge, the generator

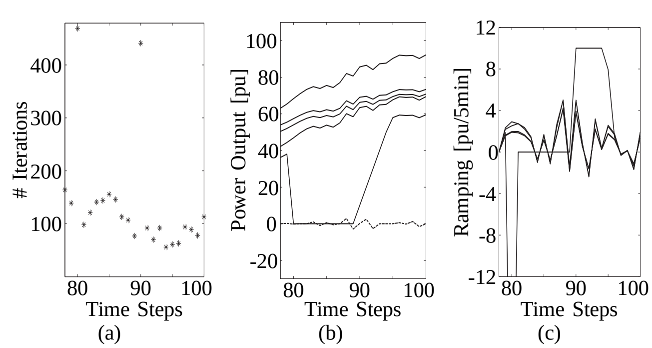

Fig. 13. (a) Number of iterations, (b) generator and storage outputs, (c) generator ramping for generator disconnect at $t = 80$ and reconnect at $t = 90$.

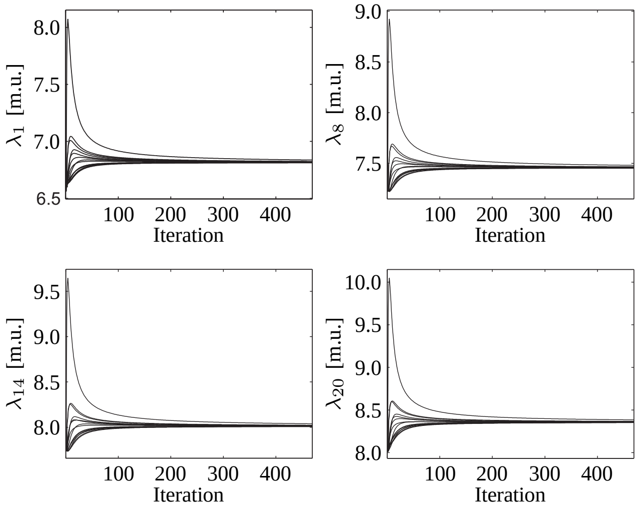

Fig. 14. Evolution of λ's for all nodes for prediction steps $k = \{1, 8, 14, 20\}$ at time step $t = 80$.

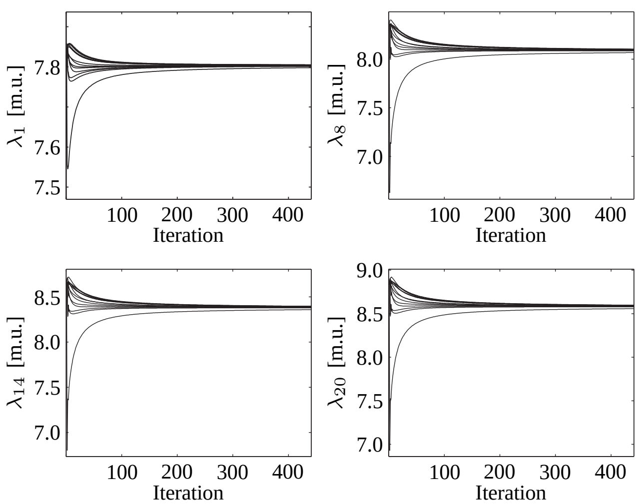

Fig. 15. Evolution of λ's for all nodes for prediction steps $k = \{1, 8, 14, 20\}$ at time step $t = 90$.

outputs and the ramp rates. It can be seen that convergence is achieved within ~ 450 iterations in the time steps of the disconnection and reconnection. The iterations for the other time steps stay in the range as in Fig. 7. The reason for the increased number of iterations is the fact that we assume that the disconnection/reconnection has not been predicted, hence, the initial point for the consensus algorithm is not as accurate as with no disconnection. Figs. 13(b) and 13(c) show the disconnect of the generator and the increase in power injection at the maximum ramp rate as soon as it is reconnected.

Figures 14 and 15 show the evolution of the λ's over the iteration of the consensus algorithm. It is obvious that a greater correction from the initial point is required which leads to a higher number of iterations to fulfill the stopping criterion.

D. Simulation Results: Additional Communication

The convergence rate, measured in number of iterations until a pre-defined convergence criterion is fulfilled, of any distributed algorithm is dependent on the choice of the communication graph. It is not the focus of this paper to derive the optimal communication graph, however, in this section, simulations are provided which indicate how the convergence rate can be improved by adding a few additional communication links.

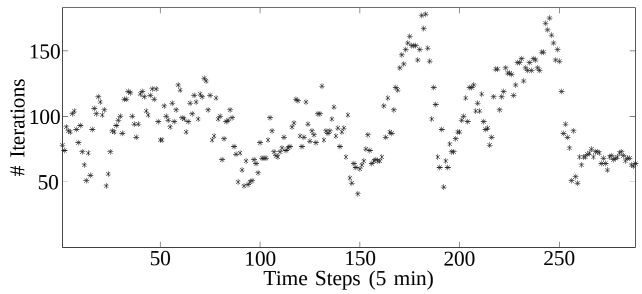

Fig. 16. Iteration steps required to converge with additional communication links.

In addition to the communication network shown in Fig. 4, we add three communication links, namely between nodes 3 and 12, between nodes 1 and 8 and between nodes 6 and 7. The additional communication links have been chosen such as to connect buses which are at opposite ends of the communication network with the result that the diameter of the network defined as the maximum number of nodes over which information needs be communicated to travel from one node to another in the network is reduced. The effect is that information spreads faster in the network leading to improved convergence.

The resulting numbers of iterations for the normal operation case, i.e. the same simulation setup as in Sect. V-B but with additional communication links, are provided in Fig. 16. It can be seen that the required number of iterations has decreased significantly.

VI. Conclusion

In this paper, an approach for the coordination of agents in a microgrid is presented. It is assumed that each generator, load and storage device is connected to a specific node which is shadowed by an agent. The respective agent defines cost and demand functions for the producers and consumers at its node and communicates with the agents of neighboring nodes. The goal of the communication process is to find an agreement on an incremental "price" for power provision while ensuring that overall generation is equal to load. The proposed approach uses the consensus + innovations method allowing for a robust and fully distributed coordination of the components in the microgrid. Optimal usage of available storage and incorporation of ramp rate limitations is achieved by applying the method to a multi-step economic dispatch. In the multi-step case, agents need to agree on prices for each time interval in the prediction horizon. Operational constraints are observed by projecting the obtained temporary solution into the feasible solution space. Adding a receding horizon to the multi-step optimization enables a distributed Model Predictive Control approach and allows for good initial starting points for the consensus algorithm.

References

[1] R.H. Lasseter, "MicroGrids", in *IEEE Power Engineering Society Winter Meeting*, 2002, vol. 1, pp. 305 – 308.
[2] R.H. Lasseter and P. Paigi, "Microgrid: a conceptual solution", in *IEEE 35th Annual Power Electronics Specialists Conference*, June 2004, vol. 6, pp. 4285 – 4290.

[3] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management", *IEEE Power and Energy Magazine*, vol. 6, no. 3, pp. 54 –65, June 2008.

[4] A.G. Tsikalakis and N.D. Hatziargyriou, "Centralized control for optimizing microgrids operation", *IEEE Transactions on Energy Conversion*, vol. 23, no. 1, pp. 241–248, 2008.

[5] S. Conti, R. Nicolosi, S.A. Rizzo, and H.H. Zeineldin, "Optimal dispatching of distributed generators and storage systems for MV islanded microgrids", *IEEE Transactions on Power Delivery*, vol. 27, no. 3, pp. 1243–1251, 2012.

[6] C.M. Colson and M.H. Nehrir, "Algorithms for distributed decision-making for multi-agent microgrid power management", in *IEEE Power and Energy Society General Meeting*, 2011.

[7] A.L. Dimeas and N.D. Hatziargyriou, "A MAS architecture for microgrids control", in *13th International Conference on Intellgient Systems Application to Power Systems, Arlington, USA*, 2005.

[8] S. Kar, J.M.F. Moura, and K. Ramanan, "Distributed parameter estimation in sensor networks: nonlinear observation models and imperfect communication", *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3575 – 3605, June 2012.

[9] S. Kar and G. Hug, "Distributed robust economic dispatch in power systems: A consensus + innovations approach", in *IEEE Power and Energy Society General Meeting*, July 2012.

[10] A.J. Conejo, F.J. Nogales, and F.J. Prieto, *Decomposition Techniques in Mathematical Programming*, Springer, 2006.

[11] A.N. Venkat, I.A. Hiskens, J.B. Rawlings, and S.J. Wright, "Distributed MPC strategies with application to power system automatic generation contro", *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1192–1206, 2008.

[12] S. Talukdar, J. Dong, P. Hines, and B. Krogh, "Distributed model predictive control for the mitigation of cascading failures", 2005, pp. 4440–4445.

[13] M. Kraning, E. Chu, J. Lavaei, and S. Boyd, "Dynamic network energy management via proximal message passing", *Foundations and Trends in Optimization*, vol. 1, no. 2, pp. 1–54, 2013.

[14] Z. Zhang, X. Ying, and M.-Y. Chow, "Decentralizing the economic dispatch problem using a two-level incremental cost consensus algorithm in a smart grid environment", in *North American Power Symposium (NAPS)*, Aug. 2011.

[15] Z. Zhang and M.-Y. Chow, "Incremental cost consensus algorithm in a smart grid environment", in *IEEE Power and Energy Society General Meeting*, July 2011.

[16] Z. Zhang and M.-Y. Chow, "The leader election criterion for decentralized economic dispatch using incremental cost consensus algorithm", in *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*, Nov. 2011, pp. 2730 –2735.

[17] Z. Zhang and M.-Y. Chow, "Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid", *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1761 –1768, Nov. 2012.

[18] A.D. Dominguez-Garcia, C.N. Hadjicostis, and N.H. Vaidya, "Resilient networked control of distributed energy resources", *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1137 –1148, July 2012.

[19] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems", *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, January 2007.

[20] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing", *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

Gabriela Hug (S'05, M'08) was born in Baden, Switzerland. She received the M.Sc. degree in electrical engineering from the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, in 2004 and the Ph.D. degree from the same institution in 2008. After her PhD, she worked in the Special Studies Group of Hydro One in Toronto, Canada and since 2009 she is an Assistant Professor at Carnegie Mellon University in Pittsburgh, USA. Her research is dedicated to control and optimization of electric power systems.

Soummya Kar (S'05, M'10) received the B.Tech. degree in Electronics and Electrical Communication Engineering from the Indian Institute of Technology, Kharagpur, India, in May 2005 and the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, in 2010. From June 2010 to May 2011 he was with the EE Department at Princeton University as a Postdoctoral Research Associate. He is currently an Assistant Research Professor of ECE at Carnegie Mellon University. His research interests include performance analysis and inference in large-scale networked systems, adaptive stochastic systems, stochastic approximation, and large deviations.