Distributed generator coordination for initialization and anytime optimization in economic dispatch

Ashish Cherukuri Jorge Cortés

Abstract—**This paper considers the economic dispatch problem for a group of generator units communicating over an arbitrary weight-balanced digraph. The objective of the individual units is to collectively generate power to satisfy a certain load while minimizing the total generation cost, which corresponds to the sum of individual arbitrary convex functions. We propose a class of distributed Laplacian-gradient dynamics that are guaranteed to asymptotically find the solution to the economic dispatch problem with and without generator constraints. The proposed coordination algorithms are anytime, meaning that its trajectories are feasible solutions at any time before convergence, and they become better and better solutions as time elapses. Additionally, we design the provably correct, distributed DETERMINE FEASIBLE ALLOCATION strategy to handle generator initialization and scenarios with addition and deletion of units. Several simulations illustrate our results.**

I. Introduction

Environmental concerns and economic challenges are fueling technological advancements in renewable energy sources and their integration into electricity grids. In the near future, this trend will make power generation highly distributed, giving rise to large-scale grid optimization problems with an extremely dynamic nature. Since centralized approaches to these problems might become impractical, there is a need to develop distributed methods that find solutions for load management and distribution. Such distributed algorithms have the potential to meet dynamic demands and be robust against generation and transmission failures. With this motivation in mind, we study here the economic dispatch (ED) problem where a group of generators whose generation costs are described by smooth, convex functions seek to determine generation levels that respect individual constraints, meet a specified load, and minimize the total generation cost. Our aim is to design distributed algorithms that asymptotically converge to the solutions of the ED problem, are *anytime*, i.e., generate executions which are feasible at any time and have monotonically decreasing cost, and can handle unit addition and deletion.

Literature review: Given the expected high density of the future electricity grid [1], the nature of the solution methodologies to the ED problem has shifted in recent years from centralized [2] to distributed ones. Among these, several works introduce consensus-based algorithms for generators with quadratic cost functions and undirected [3], [4], [5] or directed [6] communication topologies. A limitation of consensus-based approaches is that, in general, the resulting algorithm is not anytime. Instead, center-free algorithms [7], [8] solve an optimal resource allocation problem that corresponds to the ED problem for general convex functions, are distributed, and anytime, but cannot handle individual generator constraints. The work [9] deals with general convex functions and unit constraints, but the proposed algorithm only finds suboptimal solutions by solving a regularized version of the ED problem. None of the approaches mentioned above study scenarios where the set of generator units varies over time, which normally results in violations of the load requirements. The iterative algorithms in [10] solve asymptotically the problem of finding a feasible (not necessarily optimal) power allocation for the ED problem, i.e., one that satisfies the individual constraints and meets the load requirements. The distributed algorithmic solution that we provide in this paper is able to find a feasible allocation in finite time, and can therefore handle unit addition and deletion. Our work is also related to the emerging body of research on distributed optimization, see e.g., [11], [12], [13] and references therein. In this class of problems, each agent in the network maintains, communicates, and updates an estimate of the complete solution vector. This is a major difference with respect to our setting, where each unit optimizes over and communicates its own local variable, and these variables are tied in together through a global constraint.

Statement of contributions: Our starting point is the formulation of the economic dispatch (ED) problem for a group of generator units that communicate over an arbitrary weight-balanced, strongly connected digraph. The first contribution pertains to the relaxed economic dispatch (rED) problem, which is the ED problem without bounds on the individual generators' capacity. We introduce the distributed Laplacian-gradient dynamics, establish its exponential convergence to the set of solutions of the rED problem, and characterize the associated rate. As a by-product of our analysis, we establish the anytime nature of this algorithm and its convergence under jointly strongly connected communication topologies. Our second contribution concerns the ED problem. We use a nonsmooth exact penalty function to transform the problem, which has generators' capacity bounds, into an equivalent optimization with no such constraints. The resulting formulation resembles the rED problem, and this leads us to the design of the distributed Laplacian-nonsmooth-gradient dynamics. This algorithm provably converges to the solutions of the ED problem, and is also anytime and robust to switching communication topologies that remain strongly connected. Our third contribution deals with the distributed allocation of the load to the network of generators while respecting their capacity bounds. We propose a three-phase strategy termed DETERMINE FEASIBLE ALLOCATION. The first phase maintains a spanning tree over the units present in the network, the second phase determines the capacity of each subtree to

An incomplete version of this manuscript was submitted to the 2014 ACC American Control Conference, Portland, OR

A. Cherukuri and J. Cortés are with the Department of Mechanical and Aerospace Engineering, University of California, San Diego, {acheruku,cortes}@ucsd.edu.

allocate additional power, and the third phase allocates power to each individual unit, respecting the constraints, to meet the overall load. Our algorithm terminates in finite time and can be used for the initialization of the Laplacian-nonsmooth-gradient dynamics and to handle scenarios with power imbalances caused by the addition or deletion of generators.

Organization: Section II contains notation and basic preliminaries. Section III defines the ED and rED problems. Sections IV and V introduce, respectively, the Laplacian-gradient and the Laplacian-nonsmooth-gradient dynamics. Section VI presents the DETERMINE FEASIBLE ALLOCATION routine and establishes its correctness. Section VII presents simulation results. Finally, Section VIII gathers our conclusions.

II. Preliminaries

We begin with some notational conventions. Let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\mathbb{R}_{>0}$, $\mathbb{Z}_{\geq 1}$ denote the real, nonnegative real, positive real, and positive integer numbers, resp. The 2- and ∞-norms on $\mathbb{R}^n$ are $\|\cdot\|_2$ and $\|\cdot\|_\infty$, resp. We let $B(x,\delta) = \{y \in \mathbb{R}^n \mid \|y-x\|_2 < \delta\}$. For $D \subset \mathbb{R}^n$, $\mathrm{bd}(D)$ and $|D|$ denote its boundary and cardinality, resp. We denote $\mathbf{0}_n = (0,\dots,0) \in \mathbb{R}^n$, $\mathbf{1}_n = (1,\dots,1) \in \mathbb{R}^n$, and $I_n \in \mathbb{R}^{n\times n}$ for the identity matrix. For $x, y \in \mathbb{R}^n$, $x \leq y$ iff $x_i \leq y_i$ for $i \in \{1,\dots,n\}$. A set-valued map $f : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ associates to each point in $\mathbb{R}^n$ a set in $\mathbb{R}^m$. Finally, we let $[u]^+ = \max\{0, u\}$ for $u \in \mathbb{R}$.

A. Graph theory

We present basic notions from algebraic graph theory following [14]. A *directed graph* (or *digraph*) is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V}$ the vertex set and $\mathcal{E} \subseteq \mathcal{V}\times\mathcal{V}$ the edge set. A path is a sequence of vertices connected by edges. A digraph is *strongly connected* if there is a path between any pair of vertices. The sets of out- and in-neighbors of v_i are, resp., $N_{\text{out}}(v_i) = \{v_j \in \mathcal{V} \mid (v_i, v_j) \in \mathcal{E}\}$ and $N_{\text{in}}(v_i) = \{v_j \in \mathcal{V} \mid (v_j, v_i) \in \mathcal{E}\}$. A *weighted digraph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathsf{A})$ is composed of a digraph $(\mathcal{V}, \mathcal{E})$ and an *adjacency matrix* $\mathsf{A} \in \mathbb{R}^{n\times n}_{\geq 0}$ with $a_{ij} > 0$ iff $(v_i, v_j) \in \mathcal{E}$. The weighted out- and in-degree of v_i are, resp., $d_{\text{out}}(v_i) = \sum_{j=1}^n a_{ij}$ and $d_{\text{in}}(v_i) = \sum_{j=1}^n a_{ji}$. The *Laplacian* matrix is $\mathsf{L} = \mathsf{D}_{\text{out}} - \mathsf{A}$, where D_{out} is the diagonal matrix with $(\mathsf{D}_{\text{out}})_{ii} = d_{\text{out}}(i)$, for all $i \in \{1,\dots,n\}$. Note that $\mathsf{L}\mathbf{1}_n = 0$. If $\mathcal{G}$ is strongly connected, then zero is a simple eigenvalue of L. $\mathcal{G}$ is undirected if $\mathsf{L} = \mathsf{L}^\top$ and *weight-balanced* if $d_{\text{out}}(v) = d_{\text{in}}(v)$, for all $v \in \mathcal{V}$. Equivalently, $\mathcal{G}$ is weight-balanced iff $\mathbf{1}_n^\top\mathsf{L} = 0$ iff $\mathsf{L} + \mathsf{L}^\top$ is positive semidefinite. Any undirected graph is weight-balanced. If $\mathcal{G}$ is weight-balanced and strongly connected, then zero is a simple eigenvalue of $\mathsf{L} + \mathsf{L}^\top$. In such case, one has for $x \in \mathbb{R}^n$,

$$x^\top(\mathsf{L}+\mathsf{L}^\top)x \geq \lambda_2(\mathsf{L}+\mathsf{L}^\top)\Big\|x - \frac{1}{n}(\mathbf{1}_n^\top x)\mathbf{1}_n\Big\|_2^2, \tag{1}$$

with $\lambda_2(\mathsf{L}+\mathsf{L}^\top)$ the smallest non-zero eigenvalue of $\mathsf{L}+\mathsf{L}^\top$.

B. Nonsmooth analysis

Here, we introduce notions from nonsmooth analysis following [15]. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *locally Lipschitz* at $x \in \mathbb{R}^n$ if there exist $L_x, \epsilon \in (0,\infty)$ such that $\|f(y) - f(y')\|_2 \leq L_x\|y - y'\|_2$, for all $y, y' \in B(x,\epsilon)$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is *regular* at $x \in \mathbb{R}^n$ if, for all $v \in \mathbb{R}^n$, the right and generalized directional derivatives of f at x in the direction of v coincide, see [15] for definitions of these notions. A function that is continuously differentiable at x is regular at x. Also, a convex function is regular. A set-valued map $\mathcal{H} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is *upper semicontinuous* at $x \in \mathbb{R}^n$ if, for all $\epsilon \in (0,\infty)$, there exists $\delta \in (0,\infty)$ such that $\mathcal{H}(y) \subset \mathcal{H}(x) + B(0,\epsilon)$ for all $y \in B(x,\delta)$. Also, $\mathcal{H}$ is *locally bounded* at $x \in \mathbb{R}^n$ if there exist $\epsilon, \delta \in (0,\infty)$ such that $\|z\|_2 \leq \epsilon$ for all $z \in \mathcal{H}(y)$ and $y \in B(x,\delta)$.

Given a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$, let Ω_f be the set (of measure zero) of points where f is not differentiable. The *generalized gradient* $\partial f : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is

$$\partial f(x) = \mathrm{co}\{\lim_{i\to\infty} \nabla f(x_i) \mid x_i \to x, x_i \notin S \cup \Omega_f\},$$

where co denotes convex hull and $S \subset \mathbb{R}^n$ is any set of measure zero. The set-valued map ∂f is locally bounded, upper semicontinuous, and takes non-empty, compact, and convex values. A *critical point* $x \in \mathbb{R}^n$ of f satisfies $0 \in \partial f(x)$.

C. Stability of differential inclusions

We gather here some useful tools for the stability analysis of differential inclusions [15]. A differential inclusion on $\mathbb{R}^n$ is

$$\dot{x} \in \mathcal{H}(x), \tag{2}$$

where $\mathcal{H} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is a set-valued map. A solution of (2) on $[0,T] \subset \mathbb{R}$ is an absolutely continuous map $x : [0,T] \to \mathbb{R}^n$ that satisfies (2) for almost all $t \in [0,T]$. If $\mathcal{H}$ is locally bounded, upper semicontinuous, and takes non-empty, compact, and convex values, then existence of solutions is guaranteed. The set of equilibria of (2) is $\mathrm{Eq}(\mathcal{H}) = \{x \in \mathbb{R}^n \mid 0 \in \mathcal{H}(x)\}$. A set $S \subset \mathbb{R}^n$ is *weakly (resp., strongly) positively invariant* under (2) if, for each $x \in S$, at least a solution (resp., all solutions) starting from x is (resp., are) entirely contained in S. For dynamics with uniqueness of solution, both notions coincide and are referred as *positively invariant*. Given $f : \mathbb{R}^n \to \mathbb{R}$ locally Lipschitz, the *set-valued Lie derivative* $\mathcal{L}_{\mathcal{H}}f : \mathbb{R}^n \rightrightarrows \mathbb{R}$ of f with respect to (2) at x is

$$\mathcal{L}_{\mathcal{H}}f = \{a \in \mathbb{R} \mid \exists v \in \mathcal{H}(x) \text{ s.t. } \zeta^\top v = a \text{ for all } \zeta \in \partial f(x)\}.$$

The next result characterizes the asymptotic properties of (2).

Theorem 2.1: (LaSalle Invariance Principle for differential inclusions): Let $\mathcal{H} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be locally bounded, upper semicontinuous, with non-empty, compact, and convex values. Let $f : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz and regular. If $S \subset \mathbb{R}^n$ is compact and strongly invariant under (2) and $\max \mathcal{L}_{\mathcal{H}}f(x) \leq 0$ for all $x \in S$, then the solutions of (2) starting at S converge to the largest weakly invariant set M contained in $S \cap \{x \in \mathbb{R}^n \mid 0 \in \mathcal{L}_{\mathcal{H}}f(x)\}$. Moreover, if the set M is finite, then the limit of each solution exists and is an element of M.

D. Constrained optimization and exact penalty functions

Here, we introduce some notions on constrained optimization problems and exact penalty functions following [16], [17]. Consider the constrained optimization problem,

$$\text{minimize} \quad f(x), \tag{3a}$$

$$\text{subject to} \quad g(x) \leq \mathbf{0}_m, \quad h(x) = \mathbf{0}_p, \tag{3b}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$, and $h : \mathbb{R}^n \to \mathbb{R}^p$, with $p \leq n$, are continuously differentiable. The *refined Slater condition* is satisfied by (3) if there exists $x \in \mathbb{R}^n$ such that $h(x) = \mathbf{0}_p$, $g(x) \leq \mathbf{0}_m$, and $g_j(x) < 0$ for all nonaffine functions g_j. The optimization (3) is convex if f and g are convex and h affine. For convex optimization problems, the refined Slater condition implies that strong duality holds. A point $x \in \mathbb{R}^n$ is a Karush-Kuhn-Tucker (KKT) point of (3) if there exist Lagrange multipliers $\lambda \in \mathbb{R}^m_{\geq 0}$, $\nu \in \mathbb{R}^p$ such that

$$g(x) \leq \mathbf{0}_m, \quad h(x) = \mathbf{0}_p, \quad \lambda^\top g(x) = 0,$$
$$\nabla f(x) + \sum_{j=1}^m \lambda_j \nabla g_j(x) + \sum_{k=1}^p \nu_k \nabla h_k(x) = 0.$$

If the optimization (3) is convex and strong duality holds, then a point is a solution of (3) if and only if it is a KKT point.

In the presence of inequality constraints in (3), we are interested in using exact penalty function methods to eliminate them while keeping the equality constraints. Following [17], consider the nonsmooth exact penalty function $f^\epsilon : \mathbb{R}^n \to \mathbb{R}$,

$$f^\epsilon(x) = f(x) + \frac{1}{\epsilon} \sum_{j=1}^m [g_j(x)]^+$$

with $\epsilon > 0$, and define the minimization problem

$$\begin{aligned} &\text{minimize} \quad f^\epsilon(x), && \text{(4a)} \\ &\text{subject to} \quad h(x) = \mathbf{0}_p. && \text{(4b)} \end{aligned}$$

Note that, if f is convex, then f^ϵ is convex (given that $t \mapsto \frac{1}{\epsilon}[t]^+$ is convex). Therefore, if the problem (3) is convex, then the problem (4) is convex as well. The following result, see e.g. [17, Proposition 1], identifies conditions under which the solutions of the optimization problems (3) and (4) coincide.

Proposition 2.2: (Equivalence between (3) *and* (4)*):* Assume that the problem (3) is convex, has nonempty and compact solution set, and satisfies the refined Slater condition. Then, (3) and (4) have exactly the same solutions if

$$\frac{1}{\epsilon} > \|\lambda\|_\infty,$$

for some Lagrange multiplier $\lambda \in \mathbb{R}^m_{\geq 0}$ of the problem (3).

Note that a Lagrange multiplier for (3) exists because the refined Slater condition holds, and hence every solution is a KKT point. The next result characterizes the solutions of a class of optimization problems. The proof is straightforward.

Lemma 2.3: (Solution form for a class of constrained optimization problems): Consider the problem

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^n f_i(x_i), && \text{(5a)} \\ &\text{subject to} \quad \mathbf{1}_n^\top x = x_l, && \text{(5b)} \end{aligned}$$

where $f_i : \mathbb{R} \to \mathbb{R}$ is continuous, locally Lipschitz, and convex, for $i \in \{1, \dots, n\}$. Define $f : \mathbb{R}^n \to \mathbb{R}^n$ by $f(x) = (f_1(x_1), \dots, f_n(x_n))$. A point x^* is a solution of (5) if and only if there exists $\mu \in \mathbb{R}$ such that

$$\mu \mathbf{1}_n \in \partial f(x^*) \quad \text{and} \quad \mathbf{1}_n^\top x^* = x_l. \tag{6}$$

III. Problem statement

Consider a network of $n \in \mathbb{Z}_{\geq 1}$ power generator units whose communication topology is represented by a strongly connected and weight-balanced digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathsf{A})$. Each generator corresponds to a vertex and an edge (i, j) represents the capability of unit j to transmit information to unit i. The power generated by unit i is $P_i \in \mathbb{R}$. Each generator $i \in \{1, \dots, n\}$ has a cost generation function $f_i : \mathbb{R} \to \mathbb{R}_{\geq 0}$, assumed to be convex and continuously differentiable. The total cost incurred by the network with the power allocation $P = (P_1, \dots, P_n) \in \mathbb{R}^n$ is given by $f : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ as

$$f(P) = \sum_{i=1}^n f_i(P_i).$$

The function f is also convex and continuously differentiable. The generators must meet a total power load $P_l \in \mathbb{R}_{>0}$, i.e., $\sum_{i=1}^n P_i = P_l$, while at the same time minimizing the total cost $f(P)$. Each generator has upper and lower limits on the power it can produce, $P_i^m \leq P_i \leq P_i^M$ for $i \in \{1, \dots, n\}$. Formally, the economic dispatch (ED) problem is

$$\begin{aligned} &\text{minimize} \quad f(P), && \text{(7a)} \\ &\text{subject to} \quad \mathbf{1}_n^\top P = P_l, && \text{(7b)} \\ &\qquad\qquad P^m \leq P \leq P^M. && \text{(7c)} \end{aligned}$$

We refer to (7b) as the *load condition* and to (7c) as the *box constraints*. We let $\mathcal{F}_{\mathrm{ED}} = \{P \in \mathbb{R}^n \mid P^m \leq P \leq P^M \text{ and } \mathbf{1}_n^\top P = P_l\}$ denote the feasibility set of (7). Since $\mathcal{F}_{\mathrm{ED}}$ is compact, the set of solutions of (7) is compact. Moreover, since the constraints (7b) and (7c) are affine, feasibility of the ED problem implies that the refined Slater condition is satisfied and strong duality holds. Note that $P^M \in \mathcal{F}_{\mathrm{ED}}$ implies $\mathcal{F}_{\mathrm{ED}}$ is a singleton set, i.e., $\mathcal{F}_{\mathrm{ED}} = \{P^M\}$. Similarly $P^m \in \mathcal{F}_{\mathrm{ED}}$ implies $\mathcal{F}_{\mathrm{ED}} = \{P^m\}$. Without loss of generality, we assume that P^M and P^m are not feasible points.

A simpler version of this problem is the relaxed economic dispatch (rED) problem, where the total cost is optimized with the load condition but without the box constraints. Formally,

$$\begin{aligned} &\text{minimize} \quad f(P), && \text{(8a)} \\ &\text{subject to} \quad \mathbf{1}_n^\top P = P_l. && \text{(8b)} \end{aligned}$$

We let $\mathcal{F}_{\mathrm{rED}} = \{P \in \mathbb{R}^n \mid \mathbf{1}_n^\top P = P_l\}$ denote the feasibility set of (8). Our objective is to design distributed procedures that allow the network to solve the ED problem. In Section IV we present an algorithmic solution to the rED problem and then build on it in Section V to solve the ED problem.

IV. Distributed algorithmic solution to the relaxed economic dispatch problem

Here we introduce a distributed algorithm to solve the rED problem (8). Consider the Laplacian-gradient dynamics

$$\dot{P} = -\mathsf{L} \nabla f(P), \tag{9}$$

where L is the Laplacian of the communication digraph $\mathcal{G}$. This dynamics is distributed in the sense that, to implement it, each generator only requires information from its out-neighbors. Specifically, if each generator knows the cost function of its neighbors, then they interchange messages that contain their respective power levels. Otherwise, if such knowledge is not available, (9) can be implemented in a distributed manner with neighboring generators interchanging messages with their respective gradient information. The next result states that (9) asymptotically converges to the set of solutions of (8).

Theorem 4.1: (Convergence of the Laplacian-gradient dynamics to the solutions of rED *problem):* Consider the rED problem (8) with $f : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ radially unbounded. Then, the feasible set $\mathcal{F}_{\mathrm{rED}}$ is positively invariant under the dynamics (9) and all trajectories starting from $\mathcal{F}_{\mathrm{rED}}$ converge to the set of solutions of (8).

Proof: For convenience, we use the shorthand notation $X_{\text{L-g}} : \mathbb{R}^n \to \mathbb{R}^n$ to refer to (9). We first establish that the total power generated by the network is conserved,

$$\mathcal{L}_{X_{\text{L-g}}}(\mathbf{1}_n^\top P) = \mathbf{1}_n^\top X_{\text{L-g}}(P) = -(\mathbf{1}_n^\top \mathsf{L})\nabla f(P) = 0, \quad (10)$$

where in the last equality we have used the fact that $\mathcal{G}$ is weight-balanced. As a consequence, $\mathcal{F}_{\mathrm{rED}}$ is positively invariant under (9). Next, we show that f is monotonically nonincreasing. Its Lie derivative along (9) is

$$\begin{aligned}\mathcal{L}_{X_{\text{L-g}}} f(P) &= -\nabla f(P)^\top \mathsf{L} \nabla f(P) \\ &= -\frac{1}{2}\nabla f(P)^\top (\mathsf{L}+\mathsf{L}^\top)\nabla f(P) \leq 0, \end{aligned} \quad (11)$$

where again we have used in the last equality that $\mathcal{G}$ is weight-balanced. Given $P_0 \in \mathbb{R}^n$, let

$$f^{-1}(\leq f(P_0)) = \{P \in \mathbb{R}^n \mid f(P) \leq f(P_0)\}.$$

Note that this sublevel set is closed, and since f is radially unbounded, bounded. Then, the set $\mathcal{W}_{P_0} = f^{-1}(\leq f(P_0)) \cap \mathcal{F}_{\mathrm{rED}}$ is closed, bounded, and from (10) and (11), positively invariant. The application of the LaSalle Invariance Principle, cf. Theorem 2.1, implies that the trajectories starting in $\mathcal{W}_{P_0}$ converge to the largest invariant set M contained in $\{P \in \mathcal{W}_{P_0} \mid \mathcal{L}_{X_{\text{L-g}}} f(P) = 0\}$. From (11) and the fact that $\mathcal{G}$ is weight-balanced and strongly connected, we deduce that $\mathcal{L}_{X_{\text{L-g}}} f(P) = 0$ implies $\nabla f(P) \in \operatorname{span}\{\mathbf{1}_n\}$, and hence $P \in \operatorname{Eq}(X_{\text{L-g}})$. Since $\mathbf{1}_n^\top P_0 = P_l$ by hypothesis, we conclude that $M = \operatorname{Eq}(X_{\text{L-g}}) \cap \mathcal{F}_{\mathrm{rED}}$, which precisely corresponds to the set of solutions of (8), cf. Lemma 2.3. ■

Remark 4.2: (Initialization of the Laplacian-gradient dynamics): To guarantee convergence to the solutions of the rED problem, the Laplacian-gradient dynamics (9) requires an initial condition satisfying the load constraints. Such initialization can be performed in a number of ways. For instance, if each unit knows P_l and n, then the algorithm can start from $\frac{P_l}{n}\mathbf{1}_n$. If only one unit knows P_l, then that unit can set its initial generation level set to P_l while the others start from 0. •

If in addition to the hypothesis of Theorem 4.1, f is strictly convex, then (8) has a unique solution, and the trajectories of (9) converge to it. The proof of Theorem 4.1 reveals two important properties of the dynamics: the load condition is satisfied at all times and the total cost is monotonically decreasing until convergence. These properties imply that (9) is *anytime*, i.e., its trajectories are feasible solutions at any time before convergence, and they become better and better solutions as time elapses. We next characterize its convergence rate.

Proposition 4.3: (Convergence rate of the Laplacian-gradient dynamics): Under the hypotheses of Theorem 4.1, further assume that there exist $k, K \in \mathbb{R}_{>0}$ such that $kI_n \preceq \nabla^2 f(P) \preceq KI_n$ for $P \in \mathbb{R}^n$. Then, the dynamics (9) converges to the unique solution of (8) exponentially fast with rate greater than or equal to $\frac{k\lambda_2(\mathsf{L}+\mathsf{L}^\top)}{2}$.

Proof: Uniqueness of the solution to (8) follows from noting that strong convexity implies strict convexity. Let $P^{\mathrm{opt}} \in \mathbb{R}^n$ denote the unique optimizer and define $V : \mathcal{F}_{\mathrm{rED}} \subset \mathbb{R}^n \to \mathbb{R}$ by $V(P) = f(P) - f(P^{\mathrm{opt}})$. Note that $V(P) \geq 0$, and $V(P) = 0$ iff $P = P^{\mathrm{opt}}$. From (11),

$$\mathcal{L}_{X_{\text{L-g}}} V(P) \leq -\frac{1}{2}\lambda_2(\mathsf{L}+\mathsf{L}^\top)\|\nabla f(P) - \frac{1}{n}(\mathbf{1}_n^\top \nabla f(P))\mathbf{1}_n\|_2^2,$$

where we have used (1). For convenience, let $e(P) = \nabla f(P) - \frac{1}{n}(\mathbf{1}_n^\top \nabla f(P))\mathbf{1}_n$. Using the fact that f is strongly convex, for $P, P' \in \mathcal{F}_{\mathrm{rED}}$, we have

$$f(P') \geq f(P) + e(P)^\top (P' - P) + \frac{k}{2}\|P' - P\|_2^2. \quad (12)$$

For fixed P, the minimum of the right-hand side is $f(P) - \frac{1}{2k}\|e(P)\|_2^2$, and hence $f(P') \geq f(P) - \frac{1}{2k}\|e(P)\|_2^2$. In particular, for $P' = P^{\mathrm{opt}}$, this yields $V(P) \leq \frac{1}{2k}\|e(P)\|_2^2$. Combining this with the bound on $\mathcal{L}_{X_{\text{L-g}}} V$ above, we get

$$\mathcal{L}_{X_{\text{L-g}}} V(P) \leq -k\lambda_2(\mathsf{L}+\mathsf{L}^\top)V(P),$$

which implies that, along any trajectory $t \mapsto P(t)$ of (9), one has $V(P(t)) \leq V(P(0))e^{-k\lambda_2(\mathsf{L}+\mathsf{L}^\top)t}$. Our next objective is to relate the magnitude of V at P with $\|P - P^{\mathrm{opt}}\|$. From $\nabla^2 f(P) \preceq KI_n$, one has $f(P') \leq f(P) + \nabla f(P)^\top (P' - P) + \frac{K}{2}\|P' - P\|_2^2$. Minimizing both sides over $P' \in \mathcal{F}_{\mathrm{rED}}$,

$$V(P) \geq \frac{1}{2K}\|e(P)\|_2^2. \quad (13)$$

Now, using (12) for $P' = P^{\mathrm{opt}}$, one has

$$\begin{aligned} f(P^{\mathrm{opt}}) &\geq f(P) + e(P)^\top (P^{\mathrm{opt}} - P) + \frac{k}{2}\|P^{\mathrm{opt}} - P\|_2^2 \\ &\geq f(P) - \|e(P)\|_2\|P^{\mathrm{opt}} - P\|_2 + \frac{k}{2}\|P^{\mathrm{opt}} - P\|_2^2. \end{aligned}$$

Since $f(P^{\mathrm{opt}}) \leq f(P)$ for any $P \in \mathcal{F}_{\mathrm{rED}}$, we deduce $\|P - P^{\mathrm{opt}}\|_2 \leq \frac{2}{k}\|e(P)\|_2$. Combining this with (13), we get

$$\|P - P^{\mathrm{opt}}\|_2^2 \leq \frac{8}{k^2}KV(P). \quad (14)$$

To obtain an upper bound, we use the fact that f is convex, and hence $f(P^{\mathrm{opt}}) \geq f(P) + \nabla f(P)^\top (P^{\mathrm{opt}} - P)$. Rearranging,

$$\begin{aligned} V(P) &\leq \nabla f(P)^\top (P - P^{\mathrm{opt}}) \\ &= e(P)^\top (P - P^{\mathrm{opt}}) \leq \|e(P)\|_2\|P - P^{\mathrm{opt}}\|_2, \end{aligned}$$

implying $V(P)^2 \leq \|e(P)\|_2^2\|P - P^{\mathrm{opt}}\|_2^2$. Using (13), we get

$$V(P) \leq 2K\|P - P^{\mathrm{opt}}\|_2^2. \quad (15)$$

Finally, along any trajectory $t \mapsto P(t)$,

$$\|P(t) - P^{\mathrm{opt}}\|_2^2 \leq \frac{16K^2}{k^2}\|P(0) - P^{\mathrm{opt}}\|_2^2 e^{-k\lambda_2(\mathsf{L}+\mathsf{L}^\top)t},$$

using (14) and (15) with $P = P(0)$, as claimed. ■

Remark 4.4: (Comparison with the center-free algorithm): The work [7] proposes the center-free algorithm to solve the rED problem (termed there optimal resource allocation problem). This algorithm essentially corresponds to a discrete-time implementation of the Laplacian-gradient dynamics (9). The convergence analysis of the center-free algorithm relies on two assumptions. First, $\nabla^2 f$ needs to be globally upper and lower bounded (in particular, this implies that f is strongly convex). Second, the Laplacian must satisfy a linear matrix inequality that constrains the choice of weights. In contrast, no such conditions are required here to establish the convergence of (9). In addition, the guaranteed rate of convergence of the center-free algorithm vanishes as the upper bound on $\nabla^2 f$ increases for a fixed weight assignment unlike the one obtained in Proposition 4.3 for (9). •

We next establish the convergence of the Laplacian-gradient dynamics in scenarios where the communication topology is switching under a weaker form of connectivity.

Proposition 4.5: (Convergence of the Laplacian-gradient dynamics under switching topology): Let Ξ_n be the set of weight-balanced digraphs over n vertices. Denote the communication digraph of the group of units at time t by $\mathcal{G}(t)$. Let $t \mapsto \mathcal{G}(t) \in \Xi_n$ be piecewise constant and assume there exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals over which the union of communication graphs is strongly connected. Then, the dynamics

$$\dot{P} = -\mathsf{L}(\mathcal{G}(t))\nabla f(P), \tag{16}$$

starting from an initial power allocation P_0 satisfying $\mathbf{1}_n^\top P_0 = P_l$ converges to the set of solutions of (8).

The proof is similar to that of Theorem 4.1 using that (i) the load condition is preserved along the switched dynamics, (ii) f is a common Lyapunov function, and (iii) infinite switching implies convergence to the invariant set characterized by $\nabla f \in \operatorname{span}\{\mathbf{1}_n\}$, the set of solutions of the rED problem.

V. Distributed algorithmic solution to the economic dispatch problem

Here we propose a distributed algorithm to solve the ED problem. We first develop an alternative formulation of this problem without inequality constraints using an exact penalty function approach. This allows us to synthesize the distributed Laplacian-nonsmooth-gradient dynamics mimicking the algorithm design of Section IV.

A. Exact penalty function formulation

We first show that, unlike the rED problem, there might be no network-wide agreement on the gradients of the local objective functions at the solutions of the ED problem.

Lemma 5.1: (Solution form for the ED *problem):* For any solution P^{opt} of the ED problem (7), there exist $\nu \in \mathbb{R}$, $\lambda^m, \lambda^M \in \mathbb{R}^n_{\geq 0}$ with $\|\lambda^m\|_\infty, \|\lambda^M\|_\infty, 2|\nu| \leq 2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty$ such that

$$\nabla f_i(P_i^{\text{opt}}) = \begin{cases} -\nu + \lambda_i^m & \text{if } P_i^{\text{opt}} = P_i^m, \\ -\nu & \text{if } P_i^m < P_i^{\text{opt}} < P_i^M, \\ -\nu - \lambda_i^M & \text{if } P_i^{\text{opt}} = P_i^M. \end{cases}$$

Proof: The Lagrangian for the ED problem (7) is $L(P,\lambda^m,\lambda^M,\nu) = f(P) + (\lambda^m)^\top(P^m - P) + (\lambda^M)^\top(P - P^M) + \nu(\mathbf{1}_n^\top P - P_l)$. A point P^{opt} is a solution of (7) iff there exist $\nu \in \mathbb{R}$, $\lambda^m, \lambda^M \in \mathbb{R}^n_{\geq 0}$ satisfying the KKT conditions

$$P^m - P^{\text{opt}} \leq \mathbf{0}_n, \qquad (\lambda^m)^\top(P^m - P^{\text{opt}}) = 0, \tag{17a}$$
$$P^{\text{opt}} - P^M \leq \mathbf{0}_n, \qquad (\lambda^M)^\top(P^{\text{opt}} - P^M) = 0, \tag{17b}$$
$$\mathbf{1}_n^\top P^{\text{opt}} = P_l, \quad \nabla f(P^{\text{opt}}) - \lambda^m + \lambda^M = -\nu\mathbf{1}_n. \tag{17c}$$

Now, consider the partition of $\{1,\dots,n\}$ associated to P^{opt},

$$I_0(P^{\text{opt}}) = \{i \in \{1,\dots,n\} \mid P_i^m < P_i^{\text{opt}} < P_i^M\},$$
$$I_+(P^{\text{opt}}) = \{i \in \{1,\dots,n\} \mid P_i^{\text{opt}} = P_i^M\},$$
$$I_-(P^{\text{opt}}) = \{i \in \{1,\dots,n\} \mid P_i^{\text{opt}} = P_i^m\}.$$

If $i \in I_0(P^{\text{opt}})$, then (17a)-(17b) imply $\lambda_i^m = \lambda_i^M = 0$, and hence $\nabla f_i(P_i^{\text{opt}}) = -\nu$ by (17c). If $i \in I_+(P^{\text{opt}})$, then (17a)-(17b) imply $\lambda_i^m = 0$, $\lambda_i^M > 0$, and hence $\nabla f_i(P_i^{\text{opt}}) = -\nu - \lambda_i^M$ by (17c). Finally, if $i \in I_-(P^{\text{opt}})$, then (17a)-(17b) imply $\lambda_i^m > 0$, $\lambda_i^M = 0$, and hence $\nabla f_i(P_i^{\text{opt}}) = -\nu + \lambda_i^m$ by (17c). To establish the bounds on the multipliers, we distinguish between whether (a) $I_0(P^{\text{opt}})$ is non-empty or (b) $I_0(P^{\text{opt}})$ is empty. In case (a), from (17), $\nu = -\nabla f_i(P_i^{\text{opt}})$ for all $i \in I_0(P^{\text{opt}})$, and therefore $|\nu| \leq \|\nabla f(P^{\text{opt}})\|_\infty$. In case (b), from (17), we get $\nu \leq -\nabla f_j(P_j^{\text{opt}})$ for all $j \in I_+(P^{\text{opt}})$. Similarly, we obtain $\nu \geq -\nabla f_k(P_k^{\text{opt}})$ for all $k \in I_-(P^{\text{opt}})$. Therefore, $-\nabla f_k(P_k^{\text{opt}}) \leq \nu \leq -\nabla f_j(P_j^{\text{opt}})$ for all $j \in I_+(P^{\text{opt}})$ and $k \in I_-(P^{\text{opt}})$. Since $I_0(P^{\text{opt}})$ is empty and by assumption $P^m, P^M \notin \mathcal{F}_{\text{ED}}$, both $I_-(P^{\text{opt}})$ and $I_+(P^{\text{opt}})$ are non-empty. Therefore, we obtain $|\nu| \leq \|\nabla f(P^{\text{opt}})\|_\infty$. This inequality, together with (17c) and the fact that either λ_i^m or λ_i^M is zero for each $i \in \{1,\dots,n\}$, implies $\|\lambda^m\|_\infty, \|\lambda^M\|_\infty \leq 2\|\nabla f(P^{\text{opt}})\|_\infty \leq 2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty$. ■

Our next step is to provide an alternative formulation of the ED problem that is similar in structure to that of the rED problem. We do this by using an exact penalty function method to remove the box constraints. Specifically, let

$$f^\epsilon(P) = \sum_{i=1}^n f_i(P_i) + \frac{1}{\epsilon}\Big(\sum_{i=1}^n([P_i - P_i^M]^+ + [P_i^m - P_i]^+)\Big).$$

Note that this corresponds to a scenario where generator $i \in \{1,\dots,n\}$ has local cost given by

$$f_i^\epsilon(P_i) = f_i(P_i) + \frac{1}{\epsilon}\Big([P_i - P_i^M]^+ + [P_i^m - P_i]^+\Big). \tag{18}$$

This function is convex, locally Lipschitz, and continuously differentiable in $\mathbb{R}$ except at $P_i = P_i^m$ and $P_i = P_i^M$. Its generalized gradient $\partial f_i^\epsilon : \mathbb{R} \rightrightarrows \mathbb{R}$ is given by

$$\partial f_i^\epsilon(P_i) = \begin{cases} \{\nabla f_i(P_i) - \frac{1}{\epsilon}\} & \text{if } P_i < P_i^m, \\ [\nabla f_i(P_i) - \frac{1}{\epsilon}, \nabla f_i(P_i)] & \text{if } P_i = P_i^m, \\ \{\nabla f_i(P_i)\} & \text{if } P_i^m < P_i < P_i^M, \\ [\nabla f_i(P_i), \nabla f_i(P_i) + \frac{1}{\epsilon}] & \text{if } P_i = P_i^M, \\ \{\nabla f_i(P_i) + \frac{1}{\epsilon}\} & \text{if } P_i > P_i^M. \end{cases}$$

As a result, the total cost f^ϵ is convex, locally Lipschitz, and regular. Its generalized gradient at $P \in \mathbb{R}^n$ is $\partial f^\epsilon(P) = \partial f_1^\epsilon(P_1) \times \cdots \times \partial f_n^\epsilon(P_n)$. Consider the optimization

$$\text{minimize} \quad f^\epsilon(P), \tag{19a}$$
$$\text{subject to} \quad \mathbf{1}_n^\top P = P_l. \tag{19b}$$

We next establish the equivalence of (19) with the ED problem.

Proposition 5.2: (Equivalence between (7) *and* (19)*):* The solutions of (7) and (19) coincide for $\epsilon \in \mathbb{R}_{>0}$ such that

$$\epsilon < \frac{1}{2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty}. \tag{20}$$

Proof: Observe the parallelism between (7) and (3) on one side and (19) and (4) on the other. Recall that, for the ED problem (7), the set of solutions is nonempty and compact, and the refined Slater condition is satisfied. Thus, from Proposition 2.2, the solutions of (19) and (7) coincide if $\frac{1}{\epsilon} > \|\lambda^m\|_\infty, \|\lambda^M\|_\infty$ for some Lagrange multipliers λ^m and λ^M. From Lemma 5.1, there exists λ^m and λ^M satisfying $\|\lambda^m\|_\infty, \|\lambda^M\|_\infty \leq 2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty$. Thus, if $\epsilon < \frac{1}{2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty}$, then $\frac{1}{\epsilon} > 2\max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty \geq \|\lambda^m\|_\infty, \|\lambda^M\|_\infty$ and the claim follows. ■

B. Laplacian-nonsmooth-gradient dynamics

Here, we propose a distributed algorithm to solve the ED problem. Our design builds on the alternative formulation (19). Consider the Laplacian-nonsmooth-gradient dynamics

$$\dot{P} \in -\mathsf{L}\partial f^\epsilon(P). \tag{21}$$

The set-valued map $-\mathsf{L}\partial f^\epsilon$ is non-empty, takes compact, convex values, and is locally bounded and upper semicontinuous. Therefore, existence of solutions is guaranteed (cf. Section II-C). Moreover, this dynamics is distributed in the sense that, to implement it, each generator only requires information from its out-neighbors. When convenient, we denote the dynamics (21) by $X_{\text{L-n-g}} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$. The next result establishes the strongly positively invariance of $\mathcal{F}_{\text{ED}}$.

Lemma 5.3: (Invariance of the feasibility set): The feasibility set $\mathcal{F}_{\text{ED}}$ is strongly positively invariant under the Laplacian-nonsmooth-gradient dynamics (21) provided that $\epsilon \in \mathbb{R}_{>0}$ satisfies (with $d_{\text{out,max}} = \max_{i\in\mathcal{V}} d_{\text{out}}(i)$)

$$\epsilon < \frac{\min_{(i,j)\in\mathcal{E}} a_{ij}}{2d_{\text{out,max}} \max_{P\in\mathcal{F}_{\text{ED}}} \|\nabla f(P)\|_\infty}. \tag{22}$$

Proof: We begin by noting that, if ϵ satisfies (22), then there exists $\alpha > 0$ such that

$$\epsilon < \frac{\min_{(i,j)\in\mathcal{E}} a_{ij}}{2d_{\text{out,max}} \max_{P\in\mathcal{F}^\alpha_{\text{ED}}} \|\nabla f(P)\|_\infty}, \tag{23}$$

where $\mathcal{F}^\alpha_{\text{ED}} = \{P \in \mathbb{R}^n \mid \mathbf{1}_n^\top P = P_l \text{ and } P^m - \alpha\mathbf{1}_n \leq P \leq P^M + \alpha\mathbf{1}_n\}$. Now, we reason by contradiction. Assume that $\mathcal{F}_{\text{ED}}$ is not strongly positively invariant under the Laplacian-nonsmooth-gradient dynamics $X_{\text{L-n-g}}$. This implies that there exists a boundary point $\bar{P} \in \text{bd}(\mathcal{F}_{\text{ED}})$, a real number $\delta > 0$, and a trajectory $t \mapsto P(t)$ obeying (21) such that $P(0) = \bar{P}$ and $P(t) \notin \mathcal{F}_{\text{ED}}$ for all $t \in (0,\delta)$. Without loss of generality, assume that $P(t) \in \mathcal{F}^\alpha_{\text{ED}}$ for all $t \in (0,\delta)$. Now, using the same reasoning as in the proof of Theorem 4.1, it is not difficult to see that the load condition is preserved along $X_{\text{L-n-g}}$. Therefore, trajectories can only leave $\mathcal{F}_{\text{ED}}$ by violating the box constraints. Thus, without loss of generality, there must exist a unit i such that $P_i(0) = P_i^M$ and $P_i(t) > P_i^M$ for all $t \in (0,\delta)$. This means that there must exist $t \to \zeta(t) \in -\mathsf{L}\partial f^\epsilon(P(t))$ and $\delta_1 \in (0,\delta)$ such that $\zeta_i(t) \geq 0$ a.e. in $(0,\delta_1)$. Next we show that this can only happen if $P_j(t) \geq P_j^M$ for all $j \in N_{\text{out}}(i)$. Since $P_i(t) > P_i^M$ for $t \in (0,\delta_1)$, then $\partial f_i(P_i(t)) = \{\nabla f_i(P_i(t)) + \frac{1}{\epsilon}\}$. Therefore,

$$\zeta_i(t) = -\sum_{j\in N_{\text{out}}(i)} a_{ij}\Big(\nabla f_i(P_i(t)) + \frac{1}{\epsilon} - \eta_j(t)\Big),$$

where $\eta_j(t) \in \partial f_j(P_j(t))$. Note that if $P_j(t) \geq P_j^M$, then $\eta_j(t) \leq \nabla f_j(P_j(t)) + \frac{1}{\epsilon}$, whereas if $P_j(t) < P_j^M$, then $\eta_j(t) \leq \nabla f_j(P_j(t))$. For convenience, denote this latter set of units by $N^<_{\text{out}}(i)$. Now, we can upper bound $\zeta_i(t)$ by

$$\begin{aligned}\zeta_i(t) &\leq -\sum_{j\in N_{\text{out}}(i)} a_{ij}\Big(\nabla f_i(P_i(t)) - \nabla f_j(P_j(t))\Big) - \frac{1}{\epsilon}\sum_{j\in N^<_{\text{out}}(i)} a_{ij} \\ &\leq 2\max_{P\in\mathcal{F}^\alpha_{\text{ED}}} \|\nabla f(P)\|_\infty d_{\text{out,max}} - \frac{1}{\epsilon}\sum_{j\in N^<_{\text{out}}(i)} a_{ij} < 0,\end{aligned}$$

where the last inequality follows from (23). Hence, $\zeta_i(t) \geq 0$ only if $P_j(t) \geq P_j^M$ for all $j \in N_{\text{out}}(i)$ and so the latter is true on $(0,\delta_1)$ by continuity of the trajectories. Extending the argument to the neighbors of each $j \in N_{\text{out}}(i)$, we obtain an interval $(0,\delta_2) \subset (0,\delta_1)$ over which all one- and two-hop neighbors of i have generation levels greater than or equal to their respective maximum limits. Recursively, and since the graph is strongly connected and the number of units finite, we get an interval $(0,\bar{\delta})$ over which $P(t) \geq P^M$, which implies $P(0) = P^M$, contradicting the fact that $P^M \notin \mathcal{F}_{\text{ED}}$. ■

We next build on this result to show that the dynamics (21) asymptotically converges to the set of solutions of (7).

Theorem 5.4: (Convergence of the Laplacian-nonsmooth-gradient dynamics to the solutions of ED *problem):* For ϵ satisfying (22), all trajectories of the dynamics (21) starting from $\mathcal{F}_{\text{ED}}$ converge to the set of solutions of the ED problem (7).

Proof: Our proof strategy relies on the LaSalle Invariance principle for differential inclusions (cf. Theorem 2.1). Recall that the function f^ϵ is locally Lipschitz and regular. Furthermore, the set-valued map $P \mapsto X_{\text{L-n-g}}(P) = -\mathsf{L}\partial f^\epsilon(P)$ is locally bounded, upper semicontinuous, and takes non-empty, compact, and convex values. The set-valued Lie derivative $\mathcal{L}_{X_{\text{L-n-g}}} f^\epsilon : \mathbb{R}^n \rightrightarrows \mathbb{R}$ of f^ϵ along (21) is

$$\mathcal{L}_{X_{\text{L-n-g}}} f^\epsilon(P) = \{-\zeta^\top \mathsf{L}\zeta \mid \zeta \in \partial f^\epsilon(P)\}. \tag{24}$$

Since $\mathcal{G}$ is weight-balanced $-\zeta^\top\mathsf{L}\zeta = -\frac{1}{2}\zeta^\top(\mathsf{L}+\mathsf{L}^\top)\zeta \leq 0$, which implies $\max \mathcal{L}_{X_{\text{L-n-g}}} f^\epsilon(P) \leq 0$ for all $P \in \mathbb{R}^n$. From Lemma 5.3, the compact set $\mathcal{F}_{\text{ED}}$ is strongly positively invariant under $X_{\text{L-n-g}}$. Therefore, the application of Theorem 2.1 yields that all evolutions of (21) starting in $\mathcal{F}_{\text{ED}}$ converge to the largest weakly invariant set M contained in $\mathcal{F}_{\text{ED}} \cap \{P \in \mathbb{R}^n | 0 \in \mathcal{L}_{X_{\text{L-n-g}}} f^\epsilon(P)\}$. From (24) and the fact that $\mathcal{G}$ is weight-balanced, we deduce that $0 \in \mathcal{L}_{X_{\text{L-n-g}}} f^\epsilon(P)$ if and only if there exists $\mu \in \mathbb{R}$ such that $\mu\mathbf{1}_n \in \partial f^\epsilon(P)$. Using Lemma 2.3, this is equivalent to $P \in \mathcal{F}_{\text{ED}}$ being a solution of (19). This implies that M corresponds to the set of solutions of (19). Finally, since (22) implies (20), Proposition 5.2 guarantees that the solutions of (7) and (19) coincide. ■

Since, $\mathcal{F}_{\text{ED}}$ is strongly positively invariant under $X_{\text{L-n-g}}$, f^ϵ is nonincreasing along $X_{\text{L-n-g}}$ (cf. proof of Theorem 5.4), and f^ϵ and f coincide on $\mathcal{F}_{\text{ED}}$, the Laplacian-nonsmooth-gradient dynamics is an anytime algorithm for the ED problem (7).

Because these properties do not depend on the specific graph, the convergence properties of (21) are the same if the communication topology is time-varying as long as it remains weight-balanced and strongly connected. Finally, the initialization procedures of Remark 4.2 do not work for (21) because of the box constraints. The iterative algorithms in [10] provide initialization procedures that only converge asymptotically to a feasible point in $\mathcal{F}_{\mathrm{ED}}$. We address this issue next.

VI. Algorithm initialization and robustness against generator addition and deletion

The distributed dynamics proposed in Sections IV and V rely on a proper initialization of the power levels of the units to satisfy the load condition, which remains constant throughout the execution. However, the latter is no longer the case if some generators leave the network or new generators join it. For the rED problem, this issue can easily be resolved by prescribing that the power of each unit leaving the network is compensated with a corresponding increase in the power of one of its neighbors, and that new generators join the network with zero power. However, for the ED problem, the presence of the box constraints makes the design of a distributed solution more challenging. This is the problem we address here. Interestingly, our strategy, termed DETERMINE FEASIBLE ALLOCATION, can also be used to initialize the dynamics (21).

We assume here that the communication topology among the generators is undirected and connected at all times. A unit deletion event corresponds to removing the corresponding vertex, and all edges associated with it, in the communication graph. A unit addition event corresponds to adding a vertex, and some additional edges associated with it, to the communication graph. At any given time, the communication topology is represented by $\mathcal{G}_{\text{events}} = (\mathcal{V}_{\text{events}}, \mathcal{E}_{\text{events}})$.

A. Algorithm rationale and informal description

We begin by providing an informal description of the DETERMINE FEASIBLE ALLOCATION strategy which allows individual generators to collectively adjust their powers in finite time to meet the total load while satisfying the box constraints. The strategy has three components. The first maintains a spanning tree and ensures that information about added or deleted generators is incorporated. The second computes the aggregated capacity of the network to accommodate power loads. The third determines a feasible power allocation for the generators. We next describe these components in more detail.

(i) *Phase 1 (tree maintenance):* This phase maintains a spanning rooted tree T_{root} whose vertices are, at any instant of time, the generators present in the network. When a unit enters the network, it sets its power to zero (all units fall into this case when this procedure is run to initialize (21)) and is assigned a token of the same value. A unit that leaves the network transfers a token with its power level to one of its neighbors. Every unit i, except the root, resets its current generation to $P_i + P_i^{\text{tkn}}$, where P_i^{tkn} is the summation of the tokens of i (with default value zero if no token is received). The root adds P_l to its token if the algorithm is executed for the initialization of (21). With these levels, the network allocation might be unfeasible and sums $P_l - P_{\text{root}}^{\text{tkn}}$.

(ii) *Phase 2 (capacity computation):* Each unit i aggregates the difference between the current generation and the lower and upper limits, respectively, for all the units in the subtree T_i of T_{root} that has i as its root. Mathematically, $C_i^{\text{m}} = \sum_{j \in T_i}(P_j - P_j^m)$ and $C_i^{\text{M}} = \sum_{j \in T_i}(P_j^M - P_j)$. These values represent the collective capacity of T_i to decrease or increase, respectively, the total power of the network while satisfying the box constraints. If $-C_{\text{root}}^{\text{m}} \leq P_{\text{root}}^{\text{tkn}} \leq C_{\text{root}}^{\text{M}}$ does not hold, then the root declares that the load cannot be met.

(iii) *Phase 3 (feasible power allocation):* The root initiates the distribution of $P_{\text{root}}^{\text{tkn}}$, starting with itself and going down the tree until the leaves. Each unit gets a power value from its parent, which it distributes among itself (respecting its box constraints) and its children, making sure that the ulterior assignments down the tree are feasible.

We next provide a formal description and analysis of phases 2 and 3. Regarding the tree maintenance in phase 1, we do not enter into details given the ample number of solutions to achieve this task available in the literature on parallel algorithms, see e.g. [18], [19]. We only mention that the root can be arbitrarily selected, the tree can be built via any tree construction algorithm, and addition and deletion events can be handled via tree repairing algorithms [20], [21], [22].

B. The GET CAPACITY strategy

Here, we describe the GET CAPACITY strategy that implements the capacity computation in phase 2. The procedure assumes that each unit i knows the identity of its parent parent_i and children children_i in the tree T_{root}, and is therefore distributed. An informal description is as follows.

> *[Informal description]:* The leaves of the tree start by sending their capacities $P_i - P_i^m$ and $P_i^M - P_i$ to their parents. Each unit, i, upon receiving the capacities of all its children, adds them along with its own to get C_i^{m} and C_i^{M}, and sends the value to its parent. The routine ends upon reaching the root.

Algorithm 1: GET CAPACITY

Executed by: generators $i \in \mathcal{V}_{\text{events}}$
Data : P_i, P_i^m, P_i^M, parent_i, children_i
Initialize : $\vec{C}_i^{\text{m}} = \vec{C}_i^{\text{M}} := -\infty \mathbf{1}_{|\text{children}_i|}$
  if children_i *is empty* **then**
    $C_i^{\text{m}} = P_i - P_i^m, C_i^{\text{M}} := P_i^M - P_i$
  else
    $C_i^{\text{m}} = C_i^{\text{M}} := -\infty$

if children_i *is empty* **then** send $(C_i^{\text{m}}, C_i^{\text{M}})$ to parent_i
while $(C_i^{\text{m}}, C_i^{\text{M}}) = (-\infty, -\infty)$ **do**
  if *message* $(C_j^{\text{m}}, C_j^{\text{M}})$ *received from child* j **then**
    update $\vec{C}_i^{\text{m}}(j) = C_j^{\text{m}}$ and $\vec{C}_i^{\text{M}}(j) = C_j^{\text{M}}$
    if $(\vec{C}_i^{\text{m}}(k), \vec{C}_i^{\text{M}}(k)) \neq (-\infty, -\infty)$ *for all* $k \in \text{children}_i$ **then**
      set $(C_i^{\text{m}}, C_i^{\text{M}}) = (P_i - P_i^m + \text{Sum}(\vec{C}_i^{\text{m}}), P_i^M - P_i + \text{Sum}(\vec{C}_i^{\text{M}}))$
      if i *is not* root **then**
        send $(C_i^{\text{m}}, C_i^{\text{M}})$ to parent_i

Algorithm 1 gives a formal description of GET CAPACITY. The following result summarizes its properties. The proof is straightforward and we omit it for brevity.

Lemma 6.1: (Correctness of GET CAPACITY*):* Starting from the spanning tree T_{root} over $\mathcal{G}_{\text{events}}$ and $P \in \mathbb{R}^{|\mathcal{V}_{\text{events}}|}$, the algorithm GET CAPACITY terminates in finite time, with each unit $i \in \mathcal{V}_{\text{events}}$ having the following information:

(i) the capacities $C_i^{\mathrm{m}} = \sum_{k \in T_i} P_k - P_k^m$ and $C_i^{\mathrm{M}} = \sum_{k \in T_i} P_k^M - P_k$ of the subtree T_i, and

(ii) the capacities $C_j^{\mathrm{m}}, C_j^{\mathrm{M}}$ of the subtrees $\{T_j\}_{j \in \text{children}_i}$ stored in $\vec{C}_i^{\mathrm{m}}, \vec{C}_i^{\mathrm{M}} \in \mathbb{R}^{|\text{children}_i|}$.

Note that the capacities C_i^{m} and C_i^{M} are non-negative if all units in the subtree T_i satisfy the box constraints. However, this might not be the case due to the resetting of generation levels in phase 1 to account for unit addition and deletion. We next establish some important properties of these capacities.

Lemma 6.2: (Bounds on feasible power allocations to subtree): Given $P \in \mathbb{R}^{|\mathcal{V}_{\text{events}}|}$, the following holds

(i) $C^{\mathrm{m}} + C^{\mathrm{M}} \geq 0$ if $P^M \geq P^m$ (and the same result holds with strict inequalities)

(ii) for each $i \in |\mathcal{V}_{\text{events}}|$, the additional power $P_i^{\text{gv}} \in \mathbb{R}$ can be further allocated to the units in T_i respecting their box constraints if and only if $-C_i^{\mathrm{m}} \leq P_i^{\text{gv}} \leq C_i^{\mathrm{M}}$.

Proof: Fact (i) follows from noting that

$$C_i^{\mathrm{m}} = \sum_{k \in T_i} (P_k - P_k^m) = \sum_{k \in T_i} (P_k^M - P_k^m) - C_i^{\mathrm{M}}.$$

Regarding fact (ii), P_i^{gv} can be allocated among the units in T_i while satisfying the box constraints for each of them iff

$$\sum_{k \in T_i} P_k^m \leq \sum_{k \in T_i} P_k + P_i^{\text{gv}} \leq \sum_{k \in T_i} P_k^M,$$

that is, adding P_i^{gv} to the current generation of T_i gives a value that falls between the collective lower and upper limits of T_i. Rearranging the terms yields the desired result. ■

C. Algorithm: FEASIBLY ALLOCATE

Here, we describe the FEASIBLY ALLOCATE strategy that implements the feasible allocation computation of phase 3. Before this strategy is executed, the generation levels computed in phase 1 are unfeasible because their sum is $P_l - P_{\text{root}}^{\text{tkn}}$ and does not satisfy the load condition. Additionally, because of unit addition and deletion, some might not be satisfying their box constraints. The FEASIBLY ALLOCATE strategy addresses both issues. The procedure assumes that each unit i knows parent$_i$, children$_i$, and the capacities C_i^{m}, C_i^{M}, $\vec{C}_i^{\mathrm{m}}$, and $\vec{C}_i^{\mathrm{M}}$ obtained in GET CAPACITY, and is therefore distributed. Informally,

> *[Informal description]:* The root initiates the algorithm by setting $P_{\text{root}}^{\text{gv}} = P_{\text{root}}^{\text{tkn}}$. Each unit i, upon initializing P_i^{gv}, computes its change in power generation ($P_i^{\text{chg}} \in \mathbb{R}$) and the power to be allocated among its children ($\vec{P}_i^{\text{chg}} \in \mathbb{R}^{|\text{children}_i|}$). The unit sets its generation to $P_i + P_i^{\text{chg}}$ and sends $\vec{P}_i^{\text{chg}}(j)$ to child $j \in \text{children}_i$. The strategy ends at the leaves.

Algorithm 2 gives a formal description of FEASIBLY ALLOCATE. The next result establishes its correctness.

Algorithm 2: FEASIBLY ALLOCATE

Executed by: generators $i \in \mathcal{V}_{\text{events}}$
Data : P_i, P_i^m, P_i^M, parent$_i$, children$_i$, $\vec{C}_i^{\mathrm{m}}$, $\vec{C}_i^{\mathrm{M}}$
Initialize : $P_i^{\text{chg}} := -\infty$, $\vec{P}_i^{\text{chg}} := -\infty \mathbf{1}_{|\text{children}_i|}$, $myP_i^{\text{dm}} := P_i - P_i^m$, $myP_i^{\text{dM}} := P_i^M - P_i$

while $P_i^{\text{chg}} = -\infty$ **do**
  if i root *or message* $\vec{P}_{\text{parent}_i}^{\text{chg}}(i)$ *from* parent$_i$ **then**
    if i root **then** $P_i^{\text{gv}} = P_{\text{root}}^{\text{tkn}}$ **else** $P_i^{\text{gv}} = \vec{P}_{\text{parent}_i}^{\text{chg}}(i)$
    set $P_i^{\text{chg}} = \text{argmin}_{x \in [-myP_i^{\text{dm}}, myP_i^{\text{dM}}]} |x|$
    for $j \in \text{children}_i$ **do**
      set $\vec{P}_i^{\text{chg}}(j) = \text{argmin}_{x \in [-\vec{C}_i^{\mathrm{m}}(j), \vec{C}_i^{\mathrm{M}}(j)]} |x|$
    set $P_i^{\text{gv}} = P_i^{\text{gv}} - P_i^{\text{chg}} - \text{Sum}(\vec{P}_i^{\text{chg}})$
    if $P_i^{\text{gv}} \geq 0$ **then**
      set $X = \min\{P_i^{\text{gv}}, myP_i^{\text{dM}} - P_i^{\text{chg}}\}$
      set $(P_i^{\text{chg}}, P_i^{\text{gv}}) = (P_i^{\text{chg}} + X, P_i^{\text{gv}} - X)$
      for $j \in \text{children}_i$ **do**
        set $X = \min\{P_i^{\text{gv}}, \vec{C}_i^{\mathrm{M}}(j) - \vec{P}_i^{\text{chg}}(j)\}$
        set $(\vec{P}_i^{\text{chg}}(j), P_i^{\text{gv}}) = (\vec{P}_i^{\text{chg}}(j) + X, P_i^{\text{gv}} - X)$
    else
      set $X = \max\{P_i^{\text{gv}}, -myP_i^{\text{dm}} - P_i^{\text{chg}}\}$
      set $(P_i^{\text{chg}}, P_i^{\text{gv}}) = (P_i^{\text{chg}} + X, P_i^{\text{gv}} - X)$
      for $j \in \text{children}_i$ **do**
        set $X = \max\{P_i^{\text{gv}}, -\vec{C}_i^{\mathrm{m}}(j) - \vec{P}_i^{\text{chg}}(j)\}$
        set $(\vec{P}_i^{\text{chg}}(j), P_i^{\text{gv}}) = (\vec{P}_i^{\text{chg}}(j) + X, P_i^{\text{gv}} - X)$
    set $P_i = P_i + P_i^{\text{chg}}$
    send $\vec{P}_i^{\text{chg}}(j)$ to each $j \in \text{children}_i$

Proposition 6.3: (Correctness of FEASIBLY ALLOCATE*):* Let $P_{\text{root}}^{\text{tkn}} \in \mathbb{R}$ such that $-C_{\text{root}}^{\mathrm{m}} \leq P_{\text{root}}^{\text{tkn}} \leq C_{\text{root}}^{\mathrm{M}}$. Then, the FEASIBLY ALLOCATE strategy terminates in finite time and the resulting power allocation $P^+ \in \mathbb{R}^{|\mathcal{V}_{\text{events}}|}$ satisfies the box constraints, $P_i^m \leq P_i^+ \leq P_i^M$ for all $i \in \mathcal{V}_{\text{events}}$, and the load condition, $P_l = \sum_{i \in \mathcal{V}_{\text{events}}} P_i^+$.

Proof: Note that, by Lemma 6.2(ii), $-C_{\text{root}}^{\mathrm{m}} \leq P_{\text{root}}^{\text{tkn}} \leq C_{\text{root}}^{\mathrm{M}}$ implies that $P_{\text{root}}^{\text{tkn}}$ can be allocated to the generators in T. In turn, by the same result, for an arbitrary unit i, $-C_i^{\mathrm{m}} \leq P_i^{\text{gv}} \leq C_i^{\mathrm{M}}$ is equivalent to the existence of a decomposition $P_i^{\text{chg}} \in \mathbb{R}$ and $\vec{P}_i^{\text{chg}} \in \mathbb{R}^{|\text{children}_i|}$ such that

$$P_i^{\text{chg}} + \text{Sum}(\vec{P}_i^{\text{chg}}) = P_i^{\text{gv}}, \tag{25a}$$

$$-myP_i^{\text{dm}} \leq P_i^{\text{chg}} \leq myP_i^{\text{dM}}, \tag{25b}$$

$$-\vec{C}_i^{\mathrm{m}} \leq \vec{P}_i^{\text{chg}} \leq \vec{C}_i^{\mathrm{M}}, \tag{25c}$$

where we use the short-hand notation $myP_i^{\text{dm}} = P_i - P_i^m$ and $myP_i^{\text{dM}} = P_i^M - P_i$. Equation (25b) corresponds to the box constraints being satisfied for unit i if assigned the additional power P_i^{chg} to generate. Equation (25c) ensures that a feasible power allocation exists for the subtree of each of its children. We break down the computation of P_i^{chg} and $\vec{P}_i^{\text{chg}}$ in two steps. First, we find the portion of power that ensures feasibility for i and its children. This is done via

$$a_i = \text{argmin}_{x \in [-myP_i^{\text{dm}}, myP_i^{\text{dM}}]} |x|,$$

$$\vec{b}_i(j) = \text{argmin}_{x \in [-\vec{C}_i^{\mathrm{m}}(j), \vec{C}_i^{\mathrm{M}}(j)]} |x|, \text{ for } j \in \text{children}_i.$$

Observe that $P_i^{\text{chg}} = a_i$ and $\vec{P}_i^{\text{chg}} = \vec{b}_i$ satisfy (25b) and (25c) but not necessarily (25a). The second step takes care of this shortcoming by defining $X_i \in \mathbb{R}$ and $\vec{Y}_i \in \mathbb{R}^{|\text{children}_i|}$ as

$$P_i^{\text{chg}} = a_i + X_i, \quad \vec{P}_i^{\text{chg}} = \vec{b}_i + \vec{Y}_i.$$

In these new variables, (25) reads as

$$X_i + \text{Sum}(\vec{Y}_i) = P_i^{\text{gv}} - a_i - \text{Sum}(\vec{b}_i), \tag{26a}$$

$$-myP_i^{\text{dm}} - a_i \le X_i \le myP_i^{\text{dM}} - a_i, \tag{26b}$$

$$-\vec{C}_i^{\text{m}} - \vec{b}_i \le \vec{Y}_i \le \vec{C}_i^{\text{M}} - \vec{b}_i. \tag{26c}$$

Adding the lower limits of (26b) and (26c) yields $-C_i^{\text{m}} - a_i - \text{Sum}(\vec{b}_i)$, where we use $C_i^{\text{m}} = myP_i^{\text{dm}} + \text{Sum}(\vec{C}_i^{\text{m}})$. Similarly, the upper limits sum $C_i^{\text{M}} - a_i - \text{Sum}(\vec{b}_i)$. Therefore, with $-C_i^{\text{m}} \le P_i^{\text{gv}} \le C_i^{\text{M}}$, (26) is solvable by unit i with knowledge of P_i^{gv}, myP_i^{dm}, myP_i^{dM}, $\vec{C}_i^{\text{m}}$, and $\vec{C}_i^{\text{M}}$. Note that the lower limits of (26b) and (26c) are nonpositive and the upper ones are nonnegative. Therefore, if $P_i^{\text{gv}+} \ge 0$, FEASIBLY ALLOCATE considers first unit i and then its children sequentially and assigns the maximum power each can take (bounded by the upper limit of (26b) and (26c)) as X_i and $\vec{Y}_i$ until there is no more to allocate. Similarly if $P_i^{\text{gv}+} < 0$ negative values are assigned (lower bounded by lower limits of (26b) and (26c)). For unit i, this corresponds to steps 9-10 (if $P_i^{\text{gv}+} \ge 0$) or 15-16 (if $P_i^{\text{gv}+} < 0$) of Algorithm 2. For the children, this corresponds to steps 11-13 (if $P_i^{\text{gv}+} \ge 0$) or steps 17-19 (if $P_i^{\text{gv}+} < 0$) of Algorithm 2. Consequently, the resulting power allocation $P^+ = P + P^{\text{chg}}$ satisfies $P^m \le P^+ \le P^M$ because (25b) holds for each unit $i \in \mathcal{V}_{\text{events}}$. Additionally,

$$\sum_{i \in \mathcal{V}_{\text{events}}} P_i^{\text{chg}} = P_{\text{root}}^{\text{chg}} + \sum_{i \in \mathcal{V}_{\text{events}} \setminus \text{root}} P_i^{\text{chg}} = P_{\text{root}}^{\text{chg}} + \sum_{i \in \text{children}_{\text{root}}} \vec{P}_{\text{root}}^{\text{chg}} = P_{\text{root}}^{\text{gv}},$$

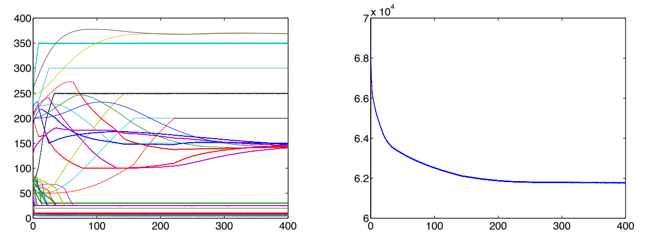
where we use that (25a) holds for each $i \in \mathcal{V}_{\text{events}}$ in the second and third inequalities. Since $P_{\text{root}}^{\text{gv}} = P_{\text{root}}^{\text{tkn}}$ and $\sum_{i \in \mathcal{V}_{\text{events}}} P_i = P_l - P_{\text{root}}^{\text{tkn}}$, we get $\sum_{i \in \mathcal{V}_{\text{events}}} P_i^+ = P_l$. ■

VII. Simulations

In this section, we illustrate the application of the Laplacian-nonsmooth-gradient dynamics to solve the ED problem (7) and the performance of the DETERMINE FEASIBLE ALLOCATION strategy to handle unit addition and deletion.

1) IEEE 118 bus: Consider the ED problem for the IEEE 118 bus test case [23]. This test case has 54 generators, with quadratic cost functions for each unit i, $f_i(P_i) = a_i + b_i P_i + c_i P_i^2$, whose coefficients belong to the ranges $a_i \in [6.78, 74.33]$, $b_i \in [8.3391, 37.6968]$, and $c_i \in [0.0024, 0.0697]$. The total load is $P_l = 4200$ and the capacity bounds vary as $P_i^m \in [5, 150]$ and $P_i^M \in [150, 400]$. The communication topology is a directed cycle with the additional bi-directional edges $\{1, 11\}, \{11, 21\}, \{21, 31\}, \{31, 41\}, \{41, 51\}$, with all weights equal to 1. Fig. 1 depicts the execution of (21). Note that as the network converges to the optimizer while satisfying the constraints, the total cost is monotonically decreasing.

2) Unit addition and deletion: Consider six power generators initially communicating over the graph in Fig. 2(a).



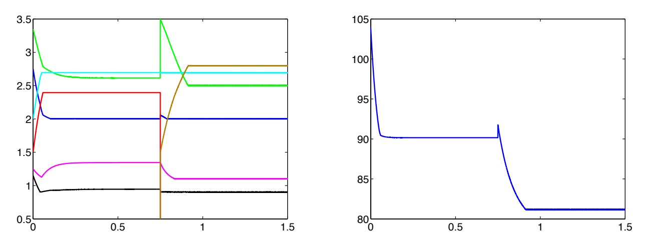
(a) Power allocation (b) Total cost

Fig. 1. Evolution of the power allocation (a) and the network cost (b) under the Laplacian-nonsmooth-gradient dynamics in the IEEE 118 bus test case.

Unit	a_i	b_i	c_i	P_i^m	P_i^M
1	1	4	5	0.9	1.5
2	1	2	3	2	3.6
3	4	4	1	1	2.4
4	2	3	2	2.5	3.5
5	1	0	5	1.1	1.6
6	1	1	1	1	2.7
7	2	2	1	1.5	3

TABLE I
COEFFICIENTS OF THE QUADRATIC COST FUNCTION $f_i(P_i) = a_i + b_i P_i + c_i P_i^2$ AND LOWER P_i^m AND UPPER P_i^M GENERATION LIMITS FOR EACH UNIT i.

The units implement (21) starting from the allocation $P_0 = (1.15, 2.75, 1.5, 3.35, 1.25, 2)$ that meets the power load $P_l = 12$ and quickly achieve a close neighborhood of the optimizer $(0.94, 2, 2.4, 2.61, 1.35, 2.7)$. After 0.75 seconds, unit 7 joins the network and unit 3 leaves it, with the resulting communication topology depicted in Fig. 2(b). The network then employs the DETERMINE FEASIBLE ALLOCATION strategy, whose execution is illustrated in Fig. 2(b)-2(d), to handle these events and determine the new feasible allocation $(0.9, 2.05, 3.5, 1.35, 2.7, 1.5)$ from which (21) is re-initialized. Table I details the cost function and the box constraints for each generator. Fig. 3 shows the evolution of the power allocations and the network cost. The network asymptotically



(a) Power allocation (b) Total cost

Fig. 3. Time evolutions of the power allocation and the network cost under the Laplacian-nonsmooth-gradient dynamics. The network of 6 generators with topology depicted in Fig. 2(a) converges towards the optimizer $(0.94, 2, 2.4, 2.61, 1.35, 2.7)$ when, at $t = 0.75s$, unit 3 (red line) leaves and unit 7 (brown line) gets added. After executing the DETERMINE FEASIBLE ALLOCATION strategy to find a feasible power allocation, the network with topology depicted in Fig. 2(b) evolves along the Laplacian-nonsmooth-gradient dynamics to arrive at the optimizer $(0.9, 2, 2.5, 1.1, 2.7, 2.8)$.

converges to the optimizer $(0.9, 2, 2.5, 1.1, 2.7, 2.8)$. The discontinuity in the allocation observed at $t = 0.75s$ corresponds to the DETERMINE FEASIBLE ALLOCATION strategy taking care of the addition of unit 7 and deletion of unit 3. Note also the corresponding jump in the network cost. In this case,

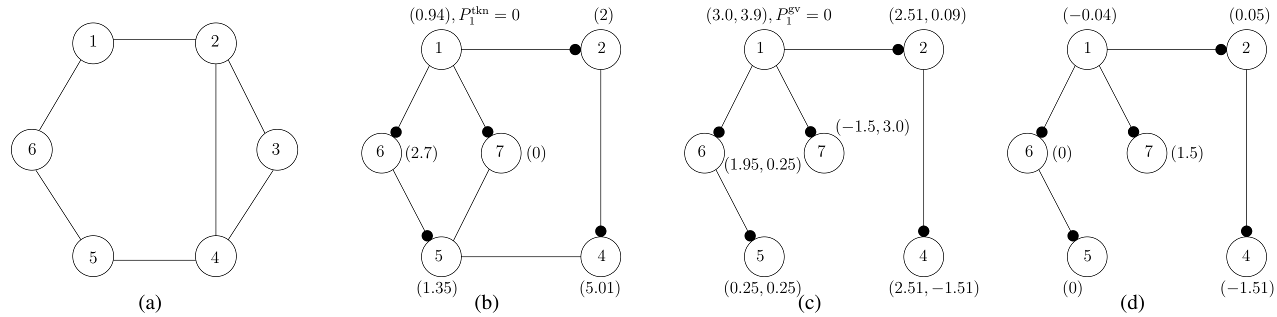


Fig. 2. (a) Initial communication topology with all edge weights equal to 1. (b) Communication topology after the addition of unit 7 and deletion of unit 3. Generation levels at the end of Phase 1 of the DETERMINE FEASIBLE ALLOCATION strategy are in parentheses. The tree is depicted via edges with dots. When leaving, unit 3 transfers its power as a token to unit 4 and hence, after token addition, 4's generation becomes 5.01 (higher than its maximum capacity). Unit 7 enters with zero power. Thus, all units except 4 have zero token value. Unit 1, being the root of the tree, sets $P_1^{\rm tkn} = 0$. (c) State after the execution of GET CAPACITY. For each unit i, $(C_i^{\rm m}, C_i^{\rm M})$ are indicated in parentheses. Unit 1 initiates FEASIBLY ALLOCATE to distribute $P_1^{\rm gv} = 0$. (d) State at the end of FEASIBLY ALLOCATE, with values of the power distributed to the units in parentheses. These values sum up to 0, and when added to their respective generation levels in (b) result into the allocation $P_0^+ = (0.9, 2.05, 3.5, 1.35, 2.7, 1.5)$ that satisfies the load condition and the box constraints.

the jump is to a higher value, although in general it could go either way depending on the network topology, the cost functions, and the box constraints of the group of generators after the events. The algorithm eventually achieves a lower network cost than the one obtained before the events because the added generator 7 incurs a lower cost when producing the same amount of power as the deleted generator 3.

VIII. CONCLUSIONS

We have proposed a class of anytime, distributed dynamics to solve the economic dispatch problem over a group of generators with convex cost functions. When generators communicate over a weight-balanced, strongly connected digraph, the Laplacian-gradient and the Laplacian-nonsmoooth-gradient dynamics provably converge to the solutions of the economic dispatch problem without and with generator constraints, respectively. We have also designed the DETERMINE FEASIBLE ALLOCATION strategy to allow a group of generators with box constraints communicating over an undirected graph to find a feasible power allocation in finite time. This method can be used to initialize the Laplacian dynamics and to handle scenarios where the load condition might be violated by the addition and/or deletion of generators. Future work will focus on the characterization of the rate of convergence of the Laplacian-nonsmooth-gradient dynamics, the extension of the algorithms to make them oblivious to initialization errors, the consideration of transmission losses and nonsmooth cost functions, and the study of more general generator dynamics and communication topologies.

REFERENCES

[1] H. Farhangi, "The path of the smart grid," *IEEE Power and Energy Magazine*, vol. 8, no. 1, pp. 18–28, 2010.

[2] B. H. Chowdhury and S. Rahman, "A review of recent advances in economic dispatch," *IEEE Transactions on Power Systems*, vol. 5, pp. 1248–1259, Nov. 1990.

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

[4] S. Kar and G. Hug, "Distributed robust economic dispatch in power systems: A consensus + innovations approach," in *IEEE Power and Energy Society General Meeting*, (San Diego, CA), July 2012. Electronic proceedings.

[5] V. Loia and A. Vaccaro, "Decentralized economic dispatch in smart grids by self-organizing dynamic agents," *IEEE Transactions on Systems, Man & Cybernetics: Systems*, 2013. To appear.

[6] A. D. Dominguez-Garcia, S. T. Cady, and C. N. Hadjicostis, "Decentralized optimal dispatch of distributed energy resources," in *IEEE Conf. on Decision and Control*, (Hawaii, USA), pp. 3688–3693, Dec. 2012.

[7] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *Journal of Optimization Theory & Applications*, vol. 129, no. 3, pp. 469–488, 2006.

[8] B. Johansson and M. Johansson, "Distributed non-smooth resource allocation over a network," in *IEEE Conf. on Decision and Control*, (Shanghai, China), pp. 1678–1683, Dec. 2009.

[9] A. Simonetto, T. Keviczky, and M. Johansson, "A regularized saddle-point algorithm for networked optimization with resource allocation constraints," in *IEEE Conf. on Decision and Control*, (Hawaii, USA), pp. 7476–7481, Dec. 2012.

[10] A. D. Dominguez-Garcia and C. N. Hadjicostis, "Distributed algorithms for control of demand response and distributed energy resources," in *IEEE Conf. on Decision and Control*, (Orlando, Florida), pp. 27–32, Dec. 2011.

[11] M. Zhu and S. Martínez, "On distributed convex optimization under inequality and equality constraints," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 151–164, 2012.

[12] A. Nedic, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922–938, 2010.

[13] B. Johansson, M. Rabi, and M. Johansson, "A randomized incremental subgradient method for distributed optimization in networked systems," *SIAM Journal on Control and Optimization*, vol. 20, no. 3, pp. 1157–1170, 2009.

[14] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*. Applied Mathematics Series, Princeton University Press, 2009. Electronically available at http://coordinationbook.info.

[15] J. Cortés, "Discontinuous dynamical systems - a tutorial on solutions, nonsmooth analysis, and stability," *IEEE Control Systems Magazine*, vol. 28, no. 3, pp. 36–73, 2008.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2009.

[17] D. P. Bertsekas, "Necessary and sufficient conditions for a penalty method to be exact," *Mathematical Programming*, vol. 9, no. 1, pp. 87–99, 1975.

[18] N. A. Lynch, *Distributed Algorithms*. Morgan Kaufmann, 1997.

[19] D. Peleg, *Distributed Computing. A Locality-Sensitive Approach*. Monographs on Discrete Mathematics and Applications, SIAM, 2000.

[20] F. C. Gaertner, "A survey of self-stabilizing spanning-tree construction algorithms," tech. rep., Ecole Polytechnique Fdrale de Lausanne, 2003.

[21] B. Awerbuch, I. Cidon, and S. Kutten, "Optimal maintenance of a spanning tree," *Journal of the Association for Computing Machinery*, vol. 55, no. 4, pp. 1–45, 2008.

[22] M. D. Schuresko and J. Cortés, "Distributed tree rearrangements for reachability and robust connectivity," *SIAM Journal on Control and Optimization*, vol. 50, no. 5, pp. 2588–2620, 2012.

[23] "IEEE 118 bus test case." http://motor.ece.iit.edu/data/JEA_IEEE118.doc.