Game-Theoretic Feedback-Based Optimization

Anurag Agarwal  John W. Simpson-Porco  Lacra Pavel*

* Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: anurag.agarwal@mail.utoronto.ca, {jusimpson,pavel}@ece.utoronto.ca).

Abstract: This paper examines the intersection between feedback-based optimization problems and distributed Nash equilibrium seeking algorithms. We consider a modification of typical GNE-seeking problems with affine coupling constraints, wherein each agent’s objective additionally depends on the measurable output of a nonlinear input-output mapping. Operator-theoretic methods are leveraged to develop an online distributed algorithm for this class of problems, with convergence criteria provided. We illustrate the algorithm via an application to coordination of distributed energy resources in a power distribution feeder.

Keywords: Game-theory and network games, Distributed optimization, Multi-agent systems, Generalized Nash equilibrium, Power systems.

1. INTRODUCTION

Traditionally, offline optimization methods are not well-suited for real-time control of complex, uncertain physical systems. A typical example of such a system would be the bulk power grid, where power generation must be optimized subject to real-time engineering and safety constraints, despite significant model uncertainty and unmeasured disturbances. Offline optimization of such systems imposes stringent informational demands, requiring accurate models and disturbance forecasting. This gap has led to the development of centralized online feedback-based optimization methods (Colombino et al., 2019; Hauswirth et al., 2021; Colombino et al., 2020; Lawrence et al., 2021), with applications to engineering systems such as communication networks (Low and Lapsley, 1999), power systems (Tang et al., 2017), and transportation (Vaquero and Cortés, 2018).

In contrast with centralized cooperative optimization, game theory provides a framework for self-interested optimal decision making. A game is a multi-agent problem, wherein each rational agent attempts to minimize its cost, dependent on other agents’ actions. The study of games often involves the study of Nash equilibria (NE), an optimal state from which no player can unilaterally deviate for a better outcome (Başar and Olsder, 1999). Reaching an NE of an appropriately set up game corresponds to the optimal state from which no player can unilaterally deviate, naturally leading to decentralized and distributed feedback controllers. Criteria for convergence of our online approximate NE-seeking method are provided. Finally, we illustrate our algorithm via a realistic case study involving the control of voltage deviations in a power distribution system.

Notation Given a positive definite matrix $P > 0$, $\langle x, y \rangle_P = x^T P y$ denotes the associated inner product on $\mathbb{R}^n$ with corresponding induced norm $\|x\|_P = \sqrt{x^T P x}$. If $P$ is omitted, it is assumed that $P = I_n$. A block diagonal matrix $A$ with matrices $A_1, \ldots, A_N$ along its diagonal is denoted $A = \text{diag}(A_1, \ldots, A_N)$. Denote $\text{col}(x_1, \ldots, x_N)$ as the column vector obtained by stacking vectors $x_1, \ldots, x_N$. Given a matrix $A \in \mathbb{R}^{n \times m}$, its $ij$th element is denoted by $[A]_{ij}$. Given a closed, convex set $\Omega \subset \mathbb{R}^n$, let $N_{\Omega}(x) = \{y \in \mathbb{R}^n | y^T (x' - x) \leq 0 \ \forall x' \in \Omega\}$ denote the normal cone of $\Omega$ at $x \in \Omega$. The projection of a point $x \in \mathbb{R}^n$ to the set $\Omega$ is the vector $P_\Omega(x) = \arg \min_{x' \in \Omega} \|x - x'\|$. For a function $f : \mathbb{R}^n \to \mathbb{R}$, $\nabla_x f(x) \in \mathbb{R}^n$ denotes its gradient, and for a function $f : \mathbb{R}^n \to \mathbb{R}^m$ defined $f(x) = (f_1(x), \ldots, f_N(x))$, the matrix $\partial_x f(x) = (\nabla_{x_1} f_1(x)^T, \ldots, \nabla_{x_N} f_N(x)^T)$ denotes its Jacobian.

2. BACKGROUND AND PRELIMINARIES

2.1 Monotone Operators

For a thorough discussion of monotone operators, we refer the reader to Bauschke and Combettes (2011). The presented properties can be used to develop guarantees for
the existence and uniqueness of solutions for variational inequalities (Facchinei and Pang, 2007).

**Definition 1.** Given $P > 0$ and $L > 0$, a map $F : S \subset \mathbb{R}^n \to \mathbb{R}^n$ is $L$-Lipschitz continuous on $S$ w.r.t. $(\cdot, \cdot)_P$ if $\|F(x) - F(y)\|_P \leq L\|x - y\|_P$ for all $x, y \in S$.

**Definition 2.** Given $P > 0$ and $\rho > 0$, a map $F : S \subset \mathbb{R}^n \to \mathbb{R}^n$ is $\rho$-strongly monotone on $S$ w.r.t. $(\cdot, \cdot)_P$ if $(x - y, F(x) - F(y)) \geq \rho\|x - y\|^2_P$, for all $x, y \in S$.

### 2.2 Graph Theory

We use graphs to describe information sharing between agents; our notation follows Bullo (2022). A weighted digraph is a triple $G = (\mathcal{V}, \mathcal{E}, \{w_{ij}\})$, where $\mathcal{V} = \{1, \ldots, N\}$ describes the set of agents, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges, and $w_{ij} > 0$ is the weight for edge $(i, j) \in \mathcal{E}$. If agent $i$ can obtain information from agent $j$, then $(i, j) \in \mathcal{E}$, and the neighbour set of agent $i$ is defined as $\mathcal{N}_G(i) = \{j \mid (j, i) \in \mathcal{E}\}$. The graph is said to be undirected if $(i, j) \in \mathcal{E} \iff (j, i) \in \mathcal{E}$ and $w_{ij} = w_{ji}$. The (out) degree of node $i$ is defined as $d_i = \sum_{j \in \mathcal{N}_G(i)} w_{ij}$. A path in an undirected graph $G$ is an ordered sequence of nodes such that any two consecutive nodes in the sequence form an edge of the graph. The graph is said to be connected if there exists a path between any two nodes.

### 3. GAME-THEORETIC FORMULATIONS WITH CONTROLLED OUTPUTS AND EXOGENOUS INPUTS

Most prior game theory literature focuses on games where each agent’s cost is solely impacted by all agents’ decisions (Yi and Pavel, 2019; Facchinei and Kanzow, 2010). This paper studies the case where, in addition, each agent is controlling a system whose output is affected by (potentially unknown) external disturbances. Such a setup would arise in applications to power systems, networks, smart cities (Ismagilova et al., 2020), etc.

Consider a set of players $\mathcal{V} = \{1, \ldots, \mathcal{N}\}$, each of which seeks to minimize a real-valued cost function in a noncooperative game. This cost function is denoted $f_i$ for player $i$, and is dependent on each other player’s decision. Each player chooses a local decision $u_i \in \Omega_i \subset \mathbb{R}^{n_i}$. Denote $\mathbf{u} = (u_1, \ldots, u_N) \in \mathbb{R}^N$ as the overall decision profile, where $N = \sum_{i=1}^N n_i$ and $\Omega = \prod_{i=1}^N \Omega_i$. Denote $\mathbf{u}_{-i}$ as the stacked decision profile of all players except $i$. We often denote $\mathbf{u}$ as $(u_i, u_{-i})$ to show the explicit dependence on other player’s decisions.

Suppose the decisions are subject to affine coupling constraints of the form $\sum_{i=1}^N A_{ii} u_i \geq \sum_{i=1}^N b_i$ and let

$$\mathcal{U} = \left\{ \mathbf{u} \in \mathbb{R}^N \mid \sum_{i=1}^N A_{ii} u_i \geq \sum_{i=1}^N b_i \right\}$$

 denote the constraint set for $\mathbf{u}$. As a result, $\mathcal{U}_i(\mathbf{u}_{-i}) := \{u_i \in \Omega_i \mid (u_i, \mathbf{u}_{-i}) \in \mathcal{U}\} \subset \mathbb{R}^{n_i}$ is the feasible decision set for agent $i$, given each other agents’ decisions.

Our formulation has one key difference from traditional game-theoretic formulations. We associate to each agent $i \in \mathcal{V}$ a vector of measurable output variables $y_i \in \mathbb{R}^{r_i}$, which are assumed to be a function of all agents’ decisions $\mathbf{u} \in \Omega$ and of an unmeasured exogenous disturbance $w \in \mathcal{W} \subset \mathbb{R}^p$ via

$$y_i = \pi_i(u_i, \mathbf{u}_{-i}, w),$$

where $\pi_i : \Omega \times \mathcal{W} \to \mathbb{R}^{r_i}$ is continuously differentiable in $\mathbf{u}$. For compactness, we set $\pi(\mathbf{u}, w) := \pi_1(u_1, \mathbf{u}_{-1}, w), \ldots, \pi_N(u_N, \mathbf{u}_{-N}, w)$.

### 4. DEVELOPING THE DISTRIBUTED FEEDBACK-BASED GNE-SEEKING ALGORITHM

#### 4.1 Game-Theoretic Equilibrium Analysis

**Definition 3.** A GNE of (3) is a decision profile $\mathbf{u}^* = (u_1^*, \ldots, u_N^*)$ such that for all $i \in \mathcal{V}$

$$u_i^* \in \arg \min_{u_i} f_i(u_i, \mathbf{u}_{-i}^*) + g_i(y_i, \mathbf{y}_{-i}^*)$$

s.t. $y_i = \pi_i(u_i, \mathbf{u}_{-i}^*, w)$

$$u_i \in \Omega_i(u_i^*),$$

where $\mathbf{y}_{-i}^* = (y_1^*, \ldots, y_{i-1}^*, y_{i+1}^*, \ldots, y_N^*)$.

If $\mathbf{u}^*$ is a GNE of the game (3), then $u_i^*$ is an optimum of the local optimization problem

$$\min_{u_i \in \mathbb{R}^{n_i}} f_i(u_i, \mathbf{u}_{-i}^*) + g_i(y_i, \mathbf{y}_{-i}^*)$$

subject to $y_i = \pi_i(u_i, \mathbf{u}_{-i}^*, w)$

$$u_i \in \Omega_i,$$

$$A_i u_i \geq b - \sum_{j \in \mathcal{V}\setminus\{i\}} A_j u_j',$$
where $b = \sum_{i=1}^{N} b_i$. A Lagrangian $L_i$ for agent $i$ can be defined as

$$L_i(u_i, \lambda_i; u_{-i}) = f_i(u_i, u_{-i}) + g_i(\pi_i(u_i, u_{-i}, w), \pi_{-i}(u_i, u_{-i}, w)) + \lambda_i^T (b - A u_i),$$

where $A = [A_1 \cdots A_N]$ and $\lambda_i$ is a Lagrange multiplier. We seek to enforce the condition $\lambda_i = \cdots = \lambda_N$, corresponding to a variational GNE (vGNE) as in Facchinei and Kanzow (2010). This ensures a more “stably social” equilibrium, where each player is equally penalized for violating the constraints.

The KKT conditions of the problem (5) are, as per Facchinei and Kanzow (2010),

$$0 \in \nabla_u f_i(u_i^*, u_{-i}^*) + \sum_{j=1}^{N} \partial_{u_j} \pi_j(u_j^*, u_{-i}^*, w)^T \nabla_{y_j} g_i(y_i^*, y_{-i}^*),$$

$$-A_i^T \lambda^* + N_{\Omega_i}(u_i^*) + N_{\mathbb{R}_{\geq 0}}(\lambda^*),$$

(6)

which $\lambda^* \in \mathbb{R}^m$ is the common multiplier, $y^* = \pi(u^*, w)$, and $\partial_{u_i} \pi_i$ denotes the Jacobian of the $j$th agent’s output map with respect to the $i$th decision variable. Let $H_w$ denote the *pseudogradient operator of the players’ cost functions*, given by

$$H_w(u) = \text{col}(\nabla_{u_1} f_1(u_1, u_{-1}) + g_1(y_1, y_{-1}), \ldots, \nabla_{u_N} f_N(u_N, u_{-N}) + g_N(y_N, y_{-N})), $$

where $y_i = \pi_i(u_i, u_{-i}, w)$ for all $i \in \mathcal{V}$. The subscript $w$ indicates dependence on the disturbance $w$. Compacty, $H_w$ may be expressed as

$$H_w(u) = F(u) + \mathcal{E}(\partial_u \pi(u, w) \partial_y g(y)^T),$$

(7)

where $F(u) = \text{col}(\nabla_{u_1} f_1, \ldots, \nabla_{u_N} f_N)$ is the pseudogradient of the agents’ decision costs. The measured output is $y = \pi(u, w)$, and $g(y) = \text{col}(g_1(y_1, y_{-1}), \ldots, g_N(y_N, y_{-N}))$ is the stacked vector of each agent’s output cost functions. The linear operator $\mathcal{E} : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^m$ is of the form $\mathcal{E}(M) = \sum_{k=1}^{N} \sum_{e_k \in \mathcal{E}_k} e_k^T M e_k \varpi_k$, where $e_k \in \mathbb{R}^n$ and $e_k \in \mathbb{R}^n$ are the $k$th and $k$th unit vectors in their respective spaces, and $\mathcal{E}_k = \{\sum_{i=1}^{k-1} n_i, \ldots, \sum_{i=1}^{k+1} n_i - 1\}$. With this notation, the KKT conditions (6) can be rewritten as

$$0 \in H_w(u^*) - A^T \lambda^* + N_{\Omega_i}(u^*)$$

$$0 \in (Au - b) + N_{\mathbb{R}_{\geq 0}}(\lambda^*),$$

(8)

4.2 Approximating the Jacobian in the vKKT Conditions

The expression for $H_w$ in (7) contains both the mapping $\pi$ and its Jacobian $\partial_u \pi$. In our subsequent algorithmic development, knowledge of $\pi$ is replaced using direct measurement of the output variables $y$. However, the Jacobian may not be known exactly and may depend on the unmeasured disturbance $w$. To obtain an implementable algorithm, this exact Jacobian can be replaced by an approximation at some nominal operating condition. In Colombino et al. (2019); Dall’Anese and Simonetto (2018); Hauswirth et al. (2016) algorithms for various online optimal power flow problems are similarly formulated and shown to robustly converge to near-optimal solutions with an approximated Jacobian. We thus consider the approximate Jacobian

$$\partial_u \pi(u, w) \approx \Pi, \forall u \in \mathcal{U}, \forall w \in \mathcal{W},$$

(9)

for some matrix $\Pi$.

**Definition 4.** Given $\Pi \in \mathbb{R}^{l \times n}$ and a disturbance $w \in \mathbb{R}^p$, a vector $\bar{u} = (u_1, \ldots, u_n) \in \mathbb{R}^n$ is an approximate variational GNE (vGNE) of the game (3) if

$$\bar{y} = \pi(\bar{u}, w)$$

$$\bar{u} \in \mathcal{U}$$

$$\bar{\lambda} \in \mathbb{R}^m$$

$$A^T \bar{\lambda} - F(\bar{u}) - \mathcal{E}(\Pi^T \partial_y g(y)^T) \in N_{\mathcal{U}}(\bar{u})$$

$$b - A \bar{u} \in N_{\mathbb{R}_{\geq 0}}(\bar{\lambda}).$$

(10)

Intuitively, an OA vGNE is a point which would be a vKKT point if the model of the Jacobian were locally accurate, and would further be a global vGNE if $g_i \circ \pi$ were convex in $u_i$. Further, the approximation resembles a vGNE in that if each agent believes the nominal model of the input-output sensitivity, no agent has any incentive to deviate. Equivalently, an OA vGNE $(\bar{u}, \bar{\lambda})$ must satisfy the approximated KKT conditions

$$0 \in H_w(\bar{u}) - A^T \bar{\lambda} + N_{\Omega}(\bar{u})$$

$$0 \in (A \bar{u} - b) + N_{\mathbb{R}_{\geq 0}}(\bar{\lambda}),$$

(11)

where the operator $H_w(\bar{u})$ is the approximation of the operator (7) and is defined as

$$H_w(\bar{u}) = F(\bar{u}) + \mathcal{E}(\Pi^T \partial_y g(y)^T).$$

(12)

4.3 Distributed OA vGNE-seeking Algorithm

**Assumption 2.** $H_w(\bar{u})$ defined in (12) is $\eta$-stronoune monotone over $\Omega$ and $\theta$-Lipschitz continuous over $\Omega$, uniformly in $w \in \mathcal{W}$ w.r.t. to the Euclidean norm.

Under Assumptions 1-2, the existence and uniqueness of the OA vGNE (Definition 4) is guaranteed (Facchinei and Pang, 2007, Theorem 2.3.3).

**Remark 2.** Since the disturbance and the mapping $\pi$ are both unknown, the operator $H_w(\bar{u}$ forms a set of operators, rather than one precisely known operator. Techniques to test Lipschitz continuity and strong monotonicity of a set of operators have been presented in Colombino et al. (2019); Simpson-Porco (2021) based on linear matrix inequalities, and can be adapted to the current setting.}

We now outline the notations used in the algorithm. Each player has access to its local cost function data $f_i$ and $g_i$ and can compute the gradients $\nabla_{u_i} f_i$ and $\nabla_{y_i} g_i$. Each
player also knows \( \Pi_1, \ldots, \Pi_N \), which are submatrices of the nominal sensitivity matrix \( \Pi \). The global affine constraint \( Au \geq b \) is not known in full by any agent; each agent only knows \( A_i, b_i \), and \( \Omega_i \), which characterizes its own involvement in the global constraint.

Agent \( i \) controls its local decision \( u_i \in \mathbb{R}^{m_i} \) and a local copy of the multiplier \( \lambda_i \in \mathbb{R}^{m_i} \). It has a local auxiliary variable \( z_i \in \mathbb{R}^{m_i} \) to coordinate with its neighbours in order to achieve consensus on \( \lambda_i \). The interference graph \( G_f = (V, \mathcal{E}_f) \) describes communication between agents with cost-decision dependencies, as in Yi and Pavel (2019); Yin et al. (2011). The weighted communication graph \( G_\lambda = (V, \mathcal{E}_\lambda, \{w_{ij}\}) \) describes the sharing of \( \lambda_i \) and \( z_i \) values between agents.

**Assumption 3.** The weighted communication graph \( G_\lambda \) is undirected and connected.

Algorithm 1 is our OA vGNE-seeking algorithm, derived from forward-backward operator splitting techniques. The idea of the algorithm is that it is seeking zeros of appropriately defined monotone operators. For a detailed treatment on the operator splitting method, we refer the reader to Yi and Pavel (2019). We now outline the algorithm. The scalars \( \tau_i, \nu_i \), and \( \sigma_i \) are step sizes. Figure 1 depicts the decentralized nature of the algorithm.

**Algorithm 1 Distributed online vGNE-seeking algorithm**

| Initialization | \( u_{i,0} \in \Pi_i, \lambda_i \in \mathbb{R}^{m_i}_+, \) and \( z_{i,0} \in \mathbb{R}^{m_i}_+ \) |
| Iteration: \( \text{Player } i \) | |
| Step 1: | Receives \( u_{j,k}, y_{j,k}, j \in N_{G_f}(i), \lambda_{j,k} \), \( j \in N_{G_\lambda}(i), y_{i,k} \) and updates: |
| & \( u_{i,k+1} \leftarrow P_\Pi(u_{i,k} - \tau_i(V_{u_{i,k}}(u_{i,k}) + \sum_{j \in N_{G_f}(i)} \Pi_j\nabla g_i(y_{j,k}; y_{i,k} - A_i^\top \lambda_{i,k})) \) |
| & \( z_{i,k+1} \leftarrow z_{i,k} + \nu \sum_{j \in N_{G_\lambda}(i)} w_{ij}(\lambda_{i,k} - \lambda_{j,k}) \) |
| Step 2: | Receives \( z_{j,k+1}, j \in N_{G_\lambda}(i) \) and updates: |
| & \( \lambda_{i,k+1} \leftarrow P_{R_+}^{\sigma_i} \lambda_{i,k} - \sigma_i(A_i(2u_{i,k+1} - u_{i,k}) - b_i) + \sum_{j \in N_{G_\lambda}(i)} w_{ij}(z_{i,k+1} - z_{j,k}) - z_{i,k} \) |
| & \( + \sum_{j \in N_{G_f}(i)} w_{ij}(\lambda_{i,k} - \lambda_{j,k}) \) |

**Theorem 1.** Suppose Assumptions 1-3 hold. Take \( 0 < \beta \leq \min \{ \frac{1}{2d^*}, \frac{\eta}{\beta} \} \), where \( d^* \) is the maximal weighted degree of \( G_\lambda \). Take some \( \delta > \frac{1}{d^*} \) and choose step sizes to satisfy the following for each player \( i \)

\[
0 < \tau_i \leq \left( \max_{j=1, \ldots, n_i} \left\{ \sum_{k=1}^{m_i} |[A_i^T]_{jk}| + \delta \right\} \right)^{-1}
\]

\[
0 < \sigma_i \leq \left( \max_{j=1, \ldots, m_i} \left\{ \sum_{k=1}^{n_i} |[A_i]_{jk}| + 2d_i + \delta \right\} \right)^{-1}
\]

\[
0 < \nu_i \leq (2d_i + \delta)^{-1}
\]

Then with Algorithm 1, each player’s local decision \( u_{i,k} \) converges to its corresponding component of the OA vGNE of the game (3), and each agent’s local multiplier \( \lambda_{i,k} \) converges to the common multiplier of the corresponding KKT condition (11).

**Proof.** Define the operators

\[
\mathfrak{A} : \left[ \begin{array}{c} u \\ \lambda \end{array} \right] \mapsto \left[ \begin{array}{c} H_{w,\Pi}(u) \\ -b \end{array} \right], \quad \mathfrak{B} : \left[ \begin{array}{c} u \\ \lambda \end{array} \right] \mapsto \left[ -A^\top \lambda + \lambda \Pi \right].
\]

The KKT conditions (11) can be rewritten as \( \text{col}(\mathfrak{u}, \lambda) \in \text{zer}(\mathfrak{A} + \mathfrak{B}) \). By Assumption 2, the operator \( H_{w,\Pi}(u) \) satisfies Assumption 2 in Yi and Pavel (2019). Assumption 1 ensures that the Assumption 1 from Yi and Pavel (2019) is satisfied. Thus the rest of the proof follows from Lemmas 5-6 and Theorem 3 in Yi and Pavel (2019). 

**Remark 3.** The algorithm presented here and the one in Yi and Pavel (2019) make the assumption that \( G_f \) has an edge for each cost dependence. This assumption is adopted in Yin et al. (2011); Yu et al. (2017); Zhu and Frazzoli (2016) as well. However, if there is too much dependence between agents’ cost functions, this graph becomes very large, and could even be a complete graph as in Grammatico (2017). Distributed GNE-seeking with only partial or incomplete information obtained from the interference graph is an area of future research (Pavel, 2020). 

Intuitively, we expect the distance between the OA vGNE and a real vKKT point of the game (3) to be bounded by the difference between the approximated Jacobian \( \Pi \) and the real Jacobian \( \partial_{\mathfrak{u}} \pi(u, w) \). The following proposition reinforces that notion. The basis of the proof is the concept of variational inequalities; we denote a variational inequality \((F(x), x' - x) \geq 0, \forall x' \in \mathcal{X} \) as VI\((F, \mathcal{X})\). For a detailed reading on variational inequalities and their relation to optima of games, we refer the reader to Facchinei and Kanzow (2010); Yi and Pavel (2019).

**Proposition 1.** If \( H_{w,\Pi} \) is \( \eta \)-strongly monotone, then

\[
\| \mathfrak{u} - \mathfrak{u}^* \| \leq \frac{1}{\eta} \| \mathcal{E} \left[ (\Pi^\top - \partial_{\mathfrak{u}} \pi(\mathfrak{u}, w)^\top) \partial_{\mathfrak{u}} g(\pi(\mathfrak{u}, w)^\top) \right] \|
\]
where \( \mathbf{u} \) is the unique OA vGNE as defined in Definition 4, and \( \mathbf{u}^* \) is a point satisfying the vKKT conditions (8).

**Proof.** We note that \( \mathbf{u} \) solves the variational inequality \( VI(\mathcal{H}_{\mathcal{V}^+} \cup \mathcal{U}) \) and \( \mathbf{u}^* \) solves the variational inequality \( VI(\mathcal{H}_{\mathcal{V}^-} \cup \mathcal{U}) \). Thus, the above follows from Theorem 1.14 in Nagurne (1993).

5. APPLICATION: DISTRIBUTION FEEDER

We now consider a practical application arising in control of renewable sources in a distribution feeder. We use the same setup as Colombino et al. (2019), with a power distribution feeder (Figure 2) whose details can be found in Dall’Anese and Simonetto (2018). In this grid, large quantities of solar generation cause bus voltages to rise above acceptable levels; the goal is to limit voltages, while minimizing the system-wide power curtailment in the photovoltaic (PV) systems.

Let \( \mathcal{V} = \{4, 7, \ldots, 36\} \) be the set of all 18 nodes equipped with controllable PV systems (grey nodes in Figure 2). Each PV system \( i \in \mathcal{V} \) is provided with active and reactive power injection set-points, denoted by \( u_i = (p_i, q_i) \), which are subject to the constraint \( u_i \in \Omega_i \), where
\[
\Omega_i := \{u_i = [p_i, q_i]’ | 0 \leq p_i \leq p_i^{\text{max}}, q_i^2 + p_i^2 \leq s_i^{\text{rated}}\}.
\]
Here \( p_i^{\text{max}} \) is the available active power for each PV unit, and \( s_i^{\text{rated}} \) is the rated apparent power of each PV inverter. The stacked decision vector is \( \mathbf{u} = \text{col}(\{u_i\}_{i \in \mathcal{V}}) \in \Omega = \prod_{i \in \mathcal{V}} \Omega_i \subset \mathbb{R}^{18} \). Similarly, let \( \mathbf{y} = \text{col}(\{y_i\}_{i \in \mathcal{V}}) \) be the vector of measured voltage magnitudes at every PV-equipped node. Let \( w \in \mathbb{R}^{70} \) be the collection of all uncontrollable loads and power injections (active and reactive) at all 35 nodes excluding the PCC. Each node’s cost function is \( f_i(u_i) = \|u_i - u_i^\text{ref}\|^2 \), where \( u_i^\text{ref} = [p_i^{\text{max}} 0]^’ \); this penalizes curtailment of the unit from its maximum real power production. The output \( \mathbf{y} = \pi(\mathbf{u}, \mathbf{w}) \) is dictated by the solution of the power flow equations for the distribution feeder (Dall’Anese and Simonetto, 2018). We assume that the mapping is nonlinear and not known in full, and we only have access to the measurement \( \mathbf{y} \) and a nominal Jacobian \( \Pi \in \mathbb{R}^{18\times36} \) of the mapping \( \pi \).

We define the safety constraints on each output to be the set \( \mathcal{Y} = \bigcap_{i \in \mathcal{V}} \mathcal{Y}_i, \) where \( \mathcal{Y}_i = \{y_i | y_i \leq y_i \leq \bar{y}_i\} \), with \( y = 0.95 \) p.u. and \( \bar{y} = 1.05 \) p.u. We define the cost function \( g_i(u_i) = \|\max(0, y_i - y_i, y_i - \bar{y}_i)^2 \| \) for each node to penalize outputs outside the constraints.

Finally, we define a set of coupling constraints differentiating this setup from the one in Colombino et al. (2019). Consider the case where, perhaps due to contractual agreements, there is a hard upper limit on the total curtailed PV power within the feeder. To model this, we define a global coupling constraint of the form \( \sum_{i \in \mathcal{V}} (p_i^{\text{max}} - p_i) \leq l \) where \( l \in \mathbb{R} \) is the upper bound on the total curtailed real power. Note that \( p_i = \mathcal{A}_i u_i \) where \( \mathcal{A}_i = [1 0] \), and each \( p_i^{\text{max}} \) is a known quantity at any given time. Thus the global constraint can be defined as \([A_1 \ldots A_N]u \geq -l + \sum_{i \in \mathcal{V}} p_i^{\text{max}}\). We can define \( b_i = p_i^{\text{max}} - \frac{l}{\mathcal{N}} \). Thus we have the sets \( \mathcal{U} \) and \( \mathcal{U}_i(u_{-i}) \) defined as (1), (3).

With this setup, we are now ready to formulate the gametheoretic problem. At each time-step, node \( i \in \mathcal{V} \) aims to optimize the following constrained game:
\[
\min_{u_i} \|\mathbf{u}_{i^\text{ref}} - u_i\|^2 + n\text{max}(0, y_i - y_i, y_i - \bar{y}_i)^2 \\
\text{s.t. } y_i = \pi_i(u, w) \\
u_i \in \mathcal{U}_i(u_{-i}).
\]

We aim to apply our distributed controller from Section 4.3 to this game. Due to the lack of explicit coupling between players’ cost functions, this application does not require an interference graph \( G_j \). For the purpose of this simulation, the communication graph \( G_i \) is defined as connecting any nodes in \( \mathcal{V} \) that are adjacent to one another. Figure 2 shows the communication graph for the problem. Thus, at each time step, each node receives multipliers \( \lambda_i \) and auxiliary variables \( z_j \) from neighbouring nodes \( j \in \mathcal{N}(i) \), along with the local voltage measurement \( y_i \), and performs the update
\[
u_{i, k+1} = P_{\Omega_i} \left[ u_{i, k} - \tau_i \left( 2(u_i - \mathbf{u}_{i^\text{ref}}\lambda_i) \\
+ 2\Pi_i s_{y_i\bar{y}_i}(y_i) - \begin{bmatrix} 1 \\ 0 \end{bmatrix} \lambda_i k \right) \right],
\]
where \( s_{y_i\bar{y}_i} \) is the soft-thresholding function, defined as
\[
s_{y_i\bar{y}_i}(y_i) = \begin{cases} y_i - \bar{y}_i & y_i < \bar{y}_i \\ 0 & \bar{y}_i \leq y_i \leq \bar{y}_i \\ y_i - y_i & y_i > \bar{y}_i \end{cases}
\]
The matrix \( \Pi_i \in \mathbb{R}^{18\times2} \) approximately captures the sensitivity of local voltage changes with respect to local active/reactive power changes. The multipliers \( \lambda_i \) and the auxiliary variables \( z_i \) are updated as in the distributed algorithm in Section 4.3.

We simulate our distributed controller using ten hours of solar irradiance and load consumption data collected from Anatolia, CA, USA, with a granularity of one second. The maximum permissible curtailment was set to \( l = 0.006 \) p.u. As can be seen in Figure 3, voltages are maintained within safety limits and the total curtailed power is constrained. The constrained power curtailment shows the algorithm’s ability to enforce coupling constraints between nodes despite lacking central knowledge of all nodes’ constraints.

![Fig. 2. IEEE 37-node feeder (Colombino et al., 2019).](image-url)
and their multipliers. Further note that, unlike in Colombino et al. (2019), each node only needs knowledge of its own input-output sensitivity $\Pi_i$, and has no dependence on the rest of the nodes' sensitivities.

![Game-Theoretic Optimization](image)

**Fig. 3.** Comparison of the distributed algorithm vs. no control.

6. CONCLUSION AND FUTURE WORK

In this paper we developed a framework for merging decentralized, game-theoretic, globally constrained optimization with real-time measurement feedback from an uncertain system. Theoretical convergence results have been given, along with a practical application of the method. A direction for future work would be to extend the above formulation to the case with hard output constraints, or to the case where each agent cannot communicate with every agent whose decision or output can explicitly affect its cost. Analyzing that class of problems would have a major impact in tractability and portability of solutions to similar problems in networks and power systems.

REFERENCES


