games

MDPI

Article

Mechanism Design for Demand Management in Energy Communities

Xupeng Wei and Achilleas Anastasopoulos *

Electrical Engineering and Computer Sciences Department, University of Michigan, Ann Arbor, MI 48109-2122, USA; xupwei@umich.edu
* Correspondence: anastas@umich.edu

Abstract: We consider a demand management problem in an energy community, in which several users obtain energy from an external organization such as an energy company and pay for the energy according to pre-specified prices that consist of a time-dependent price per unit of energy as well as a separate price for peak demand. Since users' utilities are their private information, which they may not be willing to share, a mediator, known as the planner, is introduced to help optimize the overall satisfaction of the community (total utility minus total payments) by mechanism design. A mechanism consists of a message space, a tax/subsidy, and an allocation function for each user. Each user reports a message chosen from her own message space, then receives some amount of energy determined by the allocation function, and pays the tax specified by the tax function. A desirable mechanism induces a game, the Nash equilibria (NE), of which results in an allocation that coincides with the optimal allocation for the community. As a starting point, we design a mechanism for the energy community with desirable properties such as full implementation, strong budget balance and individual rationality for both users and the planner. We then modify this baseline mechanism for communities where message exchanges are allowed only within neighborhoods, and consequently, the tax/subsidy and allocation functions of each user are only determined by the messages from their neighbors. All of the desirable properties of the baseline mechanism are preserved in the distributed mechanism. Finally, we present a learning algorithm for the baseline mechanism, based on projected gradient descent, that is guaranteed to converge to the NE of the induced game.

Keywords: mechanism design; Nash equilibrium; demand management; energy networks; learning in games

Citation: Wei, X.; Anastasopoulos, A. Mechanism Design for Demand Management in Energy Communities. *Games* **2021**, *12*, 61. https://doi.org/10.3390/g12030061

Academic Editors: Alison Watts and Randall A Berry

Received: 14 June 2021
Accepted: 30 July 2021
Published: 31 July 2021

1. Introduction

Resource allocation is an essential task in networked systems such as communication networks, energy/power networks, etc. [1–3]. In such systems, there is usually one or multiple kinds of limited and divisible resources allocated among several agents. When full information regarding agents' interests is available, solving the optimal resource allocation problem reduces to a standard optimization problem. However, in many interesting scenarios, strategic agents may choose to conceal or misreport their interests in order to obtain more resources. In such cases, it is possible that appropriate incentives are designed so that selfish agents are incentivized to truly report their private information, thus enabling optimal resource allocation [4].

In existing works related to resource allocation problems, *mechanism design* [5,6] is frequently used to address the agents' strategic behavior mentioned above. In the framework of mechanism design, the participants reach an agreement regarding how they exchange messages, how they share the resources, and how much they pay (or get paid). Such agreements are designed to incentivize the agents to provide the information needed to solve the optimization problem.

Contemporary energy systems have witnessed an explosion of emerging techniques, such as *smart meters* and *advanced metering infrastructure (AMI)* in the last decade. These smart devices and systems facilitate the transmission of users' data via communication networks and enable dynamic pricing and demand response programs [7], making it possible to implement mechanism design techniques on energy communities for improved efficiency. In this paper, we develop mechanisms to solve a demand management problem in energy communities. In an energy community, users obtain energy from an energy company and pay for it. The pre-specified prices dictated by the energy company consist of a time-dependent price per unit of energy as well as a separate price for peak demand. Users' demand is subject to constraints related to equipment capacity and minimum comfort level. Each user possesses a utility as a function of their own demand. Utilities are private information for users. The welfare of the community is the sum of utilities minus energy cost. If users were willing to truthfully report their utilities, one could easily optimize energy allocations to maximize social welfare. However, since users are strategic and might not be willing to report utilities directly, to maximize the welfare, we need to find an appropriate mechanism that incentivizes them to reveal some information about their utilities, so that optimal allocation is reached even in the presence of strategic behaviors. These mechanisms are usually required to possess several interesting properties, among which are full implementation in Nash equilibria (NE), individual rationality, and budget balance [6,8,9]. Moreover, in environments with communication constraints, it is desirable to have "distributed" mechanisms, whereby energy allocation and tax/subsidies for each user can be evaluated using only local messages in the user's neighborhood. Finally, for actual deployment of practical mechanisms, we hope that the designed mechanism has convergence properties that guarantee that NE is reached by the agents by means of a provably convergent learning algorithm.

1.1. Contributions

This paper proposes a method of designing a mechanism for implementing the optimal allocation of the demand management problem in an energy community, where there are strategic users communicating over a pre-specified message exchange network. The main contributions of our work are as follows:

(a) We design a baseline, "centralized" mechanism for an environment with concave utilities and convex constraints. A "centralized" mechanism allows for messages from all users to be communicated to the planner [6,8,9]. To avoid excessive communication cost brought about by direct mechanisms (due to messages being entire utility functions), the mechanisms proposed in this paper are indirect, non-VCG type [10–12], with messages being real vectors with finite (and small) dimensionality. Unlike related previous works [13–16], a simple form of allocation function is adopted, namely, allocation equals demand. The mechanism possesses the properties of full implementation, budget balance, and individual rationality [6,8,9]. Although we develop the mechanism for demand management in energy communities, the underlying ideas can be easily adapted to other problems and more general environments. Specifically, environments with non-monotonic utilities, external fixed unit prices, and the requirement of peak shaving are tractable with the proposed mechanism.

(b) Inspired by the vast literature on distributed non-strategic optimization [17–21], as well as our recent work on distributed mechanism design (DMD) [22,23], we modify the baseline mechanism and design a "distributed" version of it. A distributed mechanism can be deployed in environments with communication constraints, where users' messages cannot be communicated to the central planner; consequently the allocation and tax/subsidy functions for each user should only depend on messages from direct neighbors. The focus of our methodology is to show how a centralized mechanism can be modified into a decentralized one in a systematic way by means of introducing extra message components that act as proxies of the messages not available to a user due to communication constraints. An added benefit of this

systematic design is that the new mechanism preserves all of the desirable properties of the centralized mechanism.

(c) Since mechanism design (centralized or distributed) deals with equilibrium properties, one relevant question is how equilibrium is reached when agents enter the mechanism. Our final contribution in this paper is to provide "learning" algorithms [24–28] that address this question for both cases of the proposed centralized and decentralized mechanisms. The algorithm is based on the projected gradient descent method in optimization theory ([29] Chapter 7). Learning proceeds through price adjustments and demand announcements according to the prices. During this process, users do not need to reveal their entire utility functions. Convergence of the message profile toward one NE is conclusively proven, and since the mechanism is designed to fully implement the optimal allocation in NE, this implies that the allocation corresponding to the limiting message profile is the social welfare maximizing solution.

1.2. Related Literature

The model for demand management in energy communities investigated in this paper originates from network utility maximization (NUM) problems, which is one typical category of resource allocation problems in networks (see [30], Chapter 2, for a detailed approach to models and algorithms for solving NUM problems). There are two distinct research directions that have emanated from the standard centralized formulations of optimization problems.

The first direction addresses the problem of communication constraints when solving an optimization problem in a centralized fashion. Taking into account these communication constraints, several researchers have proposed *distributed optimization* methods [17–21] whereby an optimization problem is solved by means of message-passing algorithms between neighbors in a communication network. The works have been further refined to account for possible users' privacy concerns during the optimization processes [31–34]. Nevertheless, the users are assumed to be non-strategic in this line of works.

The second research direction, namely *mechanism design*, addresses the presence of strategic agents in optimization problems in a direct way. The past several decades have witnessed applications of this approach in various areas of interest, such as market allocations [14,35,36], spectrum sharing [37–39], data security [40–42], smart grid [43–45], etc. The well-known VCG mechanism [10–12] has been utilized extensively in this line of research. In VCG, users have to communicate utilities (i.e., entire functions), which leads to a high cost of information transmission. To ease the burden of communication, Kelly's mechanism [13] (and extensions to multiple divisible resources [46]) has been proposed as a solution, which uses logarithmic functions as surrogates of utilities. The users need to only report a real number and thus the communication cost reduces dramatically, at the expense of efficiency loss [47] and/or the assumption of price-taking agents. A number of works extend Kelly's idea to reduce message dimensionality in strategic settings [48–50]. Other indirect mechanisms guaranteeing full implementation in environments with allocative constraints have been proposed in [51,52] using penalty functions to incentivize feasibility and in [13–15] using *proportional allocation* or its generalization, *radial allocation* [16,53]. All aforementioned works on mechanism design can be categorized as "centralized" mechanisms, which means that agents' messages are broadcasted to a central planner who evaluates allocation and taxation for all users.

The first contribution of this paper (contribution (a)) relates to the existing research detailed above in the following way. Instead of adopting the classic VCG framework as in [10–12], the proposed centralized mechanism is indirect with finite message dimensionality. In line with previous works [14–16,53], our centralized mechanism guarantees full implementation in contrast to some other indirect mechanisms (e.g., Kelly's mechanism [13,46,47]). Additionally, compared to the works in [13–16,53], the form of allocation functions in our centralized mechanism is simpler, while at the same time, it enables the

use of the mechanism in more general environments (e.g., environments allowing negative demands, non-monotonic utilities, extra prices, etc.).

The first attempts in designing decentralized mechanisms were reported in [22,23], where mechanisms are designed with the additional property that the allocation and tax functions for each agent depend only on the messages emitted by neighboring agents. As such, allocation and taxation can be evaluated locally.

Similar to the works in [22,23], the centralized mechanism in our work can be modified into a decentralized version (contribution (b)). Unlike the previous two works, however, the distributed mechanism in this paper has a straightforward form for the allocation function, making the proofs much less involved. In addition and partially due to this simplification, these decentralized mechanisms can be applied to a broader range of environments.

Recently, there has also been a line of research designing mechanisms over networks. The works of [54,55] present mechanisms for applications such as prize allocation and peer ranking, where agents are aware of their neighbors' private characteristic (e.g., suitability, ability, or need). We caution the reader not to confuse the networks in the context of [54,55] with the message exchange networks introduced in our work in the context of contribution (b). Indeed, the networks in the two works listed above reveal how much of the private information of other agents is available to each agent, but it does not put restrictions on allocation/tax functions. On the other hand, the message exchange networks in our work treat utilities as each agent's strictly private information and puts restrictions on the form of allocation/tax functions.

Finally, learning in games is motivated by the fact that NE is, theoretically, a complete information solution concept. However, since users do not know each-others' utilities, they cannot evaluate the designed NE offline. Instead, there is a need for a process (learning), during which the NE is learnt by the community. The classic works [24–26] adopted *fictitious play*, while in [27], a connection between supermodularity and convergence of learning dynamics within an *adaptive dynamics* class was made and was further specialized in [56] to the Lindahl allocation problem. A general class of learning dynamics named *adaptive best response* was discussed in [57] in connection with *contractive games*. Learning in *monotone games* [28,58] was investigated in [59–64], with further applications in network optimization [65,66]. Recently, learning NE by utilizing *reinforcement learning* has been reported in [67–71].

In the last contribution of this paper (contribution (c)), we develop a learning algorithm for the proposed mechanisms. Most of the learning algorithms mentioned above require a strong property for induced games. For example, the learning algorithm of [27] requires supermodularity and the adaptive best response class of dynamics in [57] only applies to contractive games. Unfortunately, guaranteeing such strong properties limits the applicability of the mechanism. We overcome this difficulty with the proposed learning algorithm based on a PGD of the dual problem, thus guaranteeing convergence to NE with more general settings than the ones described in [27,57].

2. Model and Preliminaries

2.1. Demand Management in Energy Communities

Consider an energy community consisting of N users and a given time horizon T, where T can be viewed as the number of days during one billing period. Each user i in the user set $\mathcal{N}$ has their own prediction of their usage across one billing period denoted by $x^i = (x_1^i, \ldots, x_T^i)$, where x_t^i is the predicted usage of user i on the tth time slot of the billing period. Regarding notation, throughout the paper we use superscripts to denote users and constraints, and subscripts to denote time slots. Note that x_t^i can be a negative number due to the potential possibility that users in the electrical grid can generate power

through renewable technologies (e.g., photovoltaic) and can return the surplus back to the grid. The users are characterized by their utility functions as

$$v^i(x^i) = \sum_{t=1}^{T} v_t^i(x_t^i), \forall i \in \mathcal{N}.$$

The energy community, as a whole, pays for the energy. The unit prices are given separately for every time slot t, denoted by p_t. These prices are considered given and fixed (e.g., by the local utility company). In addition, the local utility company imposes a unit peak price p_0 in order to incentivize load balancing and to lessen the burden of peaks in demand. To conclude, the cost of energy to the community is as follows:

$$J(x) = \sum_{t=1}^{T} p_t \left(\sum_{i=1}^{N} x_t^i \right) + p_0 \cdot \max_{1 \le t \le T} \sum_{i=1}^{N} x_t^i, \tag{1}$$

where x is a concatenation of demand vectors $x^1, \ldots, x^N$.

The centralized demand management problem for the energy community can be formulated as

$$\underset{x \in \mathcal{X}}{\text{maximize}} \quad \sum_{i=1}^{N} v^i(x^i) - J(x). \tag{2}$$

The meaning of the feasible set $\mathcal{X}$ is to incorporate possible lower bounds on each user's demand (e.g., minimal indoor heating or AC) and/or upper bounds due to the capacities of the facilities as well as transmission line capacities.

In order to solve the optimization problem (2) using convex optimization methods, the following assumptions are made.

Assumption 1. *All of the utility functions $v_t^i(\cdot)$s are twice differentiable and strictly concave.*

Assumption 2. *The feasible set $\mathcal{X}$ is a polytope formed by several linear inequality constraints, and $\mathcal{X}$ is coordinate convex, i.e., if $x \in \mathcal{X}$, then setting any of the components of x to 0 does not let it fall outside of set $\mathcal{X}$.*

By Assumption 2, $\mathcal{X}$ can be written as $\{x | Ax \le b\}$ for some $A \in \mathbb{R}^{L \times NT}$ and $b \in \mathbb{R}_+^L$, where L is the number of linear constraints in $\mathcal{X}$, and

$$\begin{aligned} A &= \left[a^1 \ \ldots \ a^L \right]^\mathsf{T}, \\ a^l &= \left[a_1^{1,l} \ \ldots \ a_T^{1,l} \ \ldots \ a_1^{N,l} \ \ldots \ a_T^{N,l} \right]^\mathsf{T}, \ l = 1, \ldots, L, \\ b &= \left[b^1, \ldots, b^L \right]^\mathsf{T}. \end{aligned}$$

The coordinate convexity in Assumption 2 is mainly used for the outside option required by the individual rationality. Under this assumption, for a feasible allocation x, if any user i changes their mind and chooses not to participate in the mechanism, the mechanism yields a feasible allocation with $x^i = 0$ fixed.

With Assumptions 1 and 2, the energy community faces an optimization problem with a strictly concave and continuous objective function over a nonempty compact convex feasible set. Therefore, from convex optimization theory, the optimal solution for this problem always exists and should be unique [29].

Substituting the max function in (1) with a new variable w, the optimization problem in (2) can be equivalently restated as

$$\underset{x,w}{\text{maximize}} \quad \sum_{i=1}^{N} v^i(x^i) - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} x_t^i - p_0 w \tag{3a}$$

$$\text{subject to} \quad Ax \leq b, \tag{3b}$$

$$\sum_{i=1}^{N} x_t^i \leq w, \ \forall t \in \{1, \ldots, T\}. \tag{3c}$$

The proof of this equivalency can be found in Appendix A. The new optimization problem has a differentiable concave objective function with a convex feasible set, which means that it is still a convex optimization, and therefore, KKT conditions are sufficient and necessary conditions for a solution (x, λ, μ) to be the optimal solution, where $\lambda = [\lambda^1, \ldots, \lambda^L]^\mathsf{T}$ are the Lagrange multipliers for each linear constraint $a^{l\mathsf{T}} x \leq b^l$ in constraint $x \in \mathcal{X}$ and $\mu = [\mu_1, \ldots, \mu_T]^\mathsf{T}$ are the Lagrange multipliers for (3c). The KKT conditions are listed as follows:

1. Primal Feasibility:

$$x \in \mathcal{X}, \tag{4a}$$

$$\sum_{i=1}^{N} x_t^i \leq w. \tag{4b}$$

2. Dual Feasibility:

$$\lambda^l \geq 0, \ l = 1, \ldots, L; \mu_t \geq 0, \ t = 1, \ldots, T. \tag{4c}$$

3. Complementary Slackness:

$$\lambda^l (a^{l\mathsf{T}} x - b^l) = 0, \ l = 1, \ldots, L, \tag{4d}$$

$$\mu_t (\sum_{i=1}^{N} x_t^i - w) = 0, \ t = 1, \ldots, T. \tag{4e}$$

4. Stationarity:

$$p_0 = \sum_t \mu_t, \tag{4f}$$

$$\dot{v}_t^i(x_t^i) = p_t + \sum_l \lambda^l a_t^{i,l} + \mu_t, \ t = 1, \ldots, T, \ i \in \mathcal{N}. \tag{4g}$$

where $\dot{v}_t^i(\cdot)$ is the first order derivative of $v_t^i(\cdot)$.

We conclude this section by pointing out once more that our objective is not to solve (3a)–(3c) or (4a)–(4g) in a centralized or decentralized fashion. Such a methodology is well established and falls under the research area of centralized or decentralized (non-strategic) optimization. Furthermore, such a task can be accomplished only under the assumption that users report their utilities (or related quantities, such as derivatives of utilities at specific points) truthfully, i.e., they do not act strategically. Instead, our objective is to design a mechanism (i.e., messages and incentives) so that strategic users are presented with a game, the NE of which is designed so that it corresponds to the optimal solution of (3a)–(3c) or (4a)–(4g).

2.2. Mechanism Design Preliminaries

In an energy community, utilities are users' private information. Due to privacy and strategic concerns, users might not be willing to report their utilities. As a result, (3a)–(3c) or (4a)–(4g) cannot be solved directly. In order to solve (3a)–(3c) and (4a)–(4g)

under the settings stated above, we introduce a planner as an intermediary between the community and the energy company. To incentivize users to provide necessary information for optimization, the planner signs a contract with users, which prespecifies the messages needed from users and rules for determining the allocation and taxes/subsidies from/to the users. The planner commits to the contract. Informally speaking, the design of such contract is referred to as *mechanism design*.

More formally, a *mechanism* is a collection of message sets and an outcome function [8]. Specifically, in resource allocation problems, a mechanism can be defined as a tuple $(\mathcal{M}, \hat{x}(\cdot), \hat{t}(\cdot))$, where $\mathcal{M} = \mathcal{M}^1 \times \ldots \times \mathcal{M}^N$ is the space of message profile; $\hat{x} : \mathcal{M} \mapsto \mathcal{X}$ is an allocation function determining the allocation x according to the received message profile $m \in \mathcal{M}$; and $\hat{t} : \mathcal{M} \mapsto \mathbb{R}^N$ is a tax function that defines the payments (or subsidies) of users based on m (specifically, $\hat{t} = \{\hat{t}^i\}_{i\in\mathcal{N}}$ with $\hat{t}^i : \mathcal{M} \mapsto \mathbb{R}$ defining the tax/subsidy function for user i). Once defined, the mechanism induces a game $(\mathcal{N}, \mathcal{M}, \{u^i\}_{i\in\mathcal{N}})$. In this game, each user i chooses her message m^i from the message space $\mathcal{M}^i$, with the objective to maximize her payoff $u^i(m) = v^i(\hat{x}^i(m)) - \hat{t}^i(m)$. The planner charges taxes and pays for the energy cost to the company, so the planner's payoff turns out to be $\sum_i \hat{t}^i(m) - J(\hat{x}(m))$ (the net income of the planner).

For the mechanism-induced game $\mathcal{G}$, NE is an appropriate solution concept. At the equilibrium point m^*, if $\hat{x}(m^*)$ coincides with the optimal allocation (i.e., the solution of (3a)–(3c)), we say that the mechanism *implements* the optimal allocation at m^*. A mechanism has the property of *full implementation* if all of the NE m^*s implement the optimal allocation.

There are other desirable properties in a mechanism. *Individual rationality* is the property that everyone volunteers to participate in the mechanism-induced game instead of quitting. For the planner, this means that the sum of taxes $\sum_i \hat{t}^i(m^*)$ collected at NE is larger than the cost paid to the energy company $J(\hat{x}(m^*))$. In the context of this paper, *strong budget balance* is the property that the sum of taxes is exactly the same as the cost paid to the energy company, so no additional funds are required by the planner or the community to run the mechanism other than the true energy cost paid to the energy company. In addition, if we use the solution concept of NE, one significant problem is how the users know the NE without full information. Therefore, some learning algorithm is needed to help users learn the NE. If under a specific class of learning algorithm, the message profile m converges to NE m^*, then we say that the mechanism has learning guarantees with this certain class.

3. The Baseline "Centralized" Mechanism

In this section, we temporarily assume that there are no communication constraints, i.e., all of the message components are accessible for the calculations of the allocation and taxation. The mechanism designed under this assumption is called a "centralized" mechanism. In the next section, we extend this mechanism to an environment with communication constraints.

In the proposed centralized mechanism, we define user i's message m^i as

$$m^i = \left(\left\{ y_t^i \right\}_{t=1}^T, \left\{ q^{i,l} \right\}_{l\in\mathcal{L}}, \left\{ s_t^i \right\}_{t=1}^T, \left\{ \beta_t^i \right\}_{t=1}^T \right).$$

Each message component above has an intuitive meaning. Message $y_t^i \in \mathbb{R}$ can be regarded as the demand for time slot t announced by user i. Message $q^{i,l} \in \mathbb{R}_+$ is the additional price that user i expects to pay for constraint l, which corresponds to the Lagrange multiplier λ^l. Message $s_t^i \in \mathbb{R}_+$ is proportional to the peak price that user i expects to pay at time t. Intuitively, setting one s_t^i greater than $s_{t'}^i$ means that user i thinks day t is more likely to be the day with the peak demand rather than t'. This component corresponds to the Lagrange multiplier μ_t. Message $\beta_t^i \in \mathbb{R}$ is the prediction of user $(i+1)$'s usage at time t by user i. This message is included for technical reasons that will

become clear in the following (for a user index $i \in \mathcal{N}$, let $i-1$ and $i+1$ denote modulo N operations).

Denote the message space of user i by $\mathcal{M}^i$, and the space of the message profile is represented as $\mathcal{M} = \mathcal{M}^1 \times \ldots \times \mathcal{M}^N$. The allocation functions and the tax functions are functions defined on $\mathcal{M}$. The allocation functions follow the simple definition:

$$\hat{x}_t^i(m) = y_t^i,\ t = 1, \ldots, T,\ \forall i \in \mathcal{N}. \tag{5}$$

i.e., users obtains exactly what they request.

Prior to the definition of the tax functions, we want to find some variable that acts similar to μ_t at NE. Although s_t^i is designed to be proportional to μ_t, it does not guarantee $\sum_t s_t^i = p_0$, which is the KKT condition (4f). To solve this problem, we utilize a technique similar to the proportional/radial allocation in [13–16,53] to shape the suggested peak price vector s into a form that satisfies (4f). For a generic T-dimensional peak price vector $\tilde{s} = (\tilde{s}_1, \ldots, \tilde{s}_T)$ and a generic T-dimensional total demand vector $\tilde{y} = (\tilde{y}_1, \ldots, \tilde{y}_T)$, define a radial pricing operator $\mathcal{RP}^i : \mathbb{R}_+^T \times \mathbb{R}^T \mapsto \mathbb{R}_+^T$ as

$$\mathcal{RP}^i(\tilde{s}, \tilde{y}) = \Big(\mathcal{RP}_1^i(\tilde{s}, \tilde{y}), \ldots, \mathcal{RP}_T^i(\tilde{s}, \tilde{y})\Big), \tag{6a}$$

where

$$\mathcal{RP}_t^i(\tilde{s}, \tilde{y}) = \begin{cases} \frac{\tilde{s}_t}{\sum_{t'} \tilde{s}_{t'}} p_0, & \text{if } \tilde{s} \neq 0, \\ \frac{p_0 \cdot 1\{t \in \arg\max_{t'} \tilde{y}_{t'}\}}{\#(\arg\max_{t'} \tilde{y}_{t'})}, & \text{if } \tilde{s} = 0, \end{cases} \tag{6b}$$

and $\#(\arg\max_{t'} \tilde{y}_{t'})$ represents the number of elements in $\tilde{y}$ that are equal to the maximum value.

The output of the radial pricing $\mathcal{RP}(\cdot, \cdot)$ is taken as the peak price in the subsequent tax functions. When the given suggested price vector $\tilde{s}$ is a nonzero vector, the unit peak price is allocated to each day proportional to $\tilde{s}_t$. If the suggested price vector $\tilde{s} = 0$, then divide p_0 to the days with peak demand with equal proportion.

The tax functions are defined as

$$\hat{t}^i(m) = \text{cost}^i(m) + \sum_{t=1}^{T} \text{pr}\beta_t^i(m) + \sum_{l \in \mathcal{L}} \text{con}^{i,l}(m) + \sum_{t=1}^{T} \text{con}_t^i(m), \tag{7}$$

where

$$\text{cost}^i(m) = \sum_{t=1}^{T} (p_t + \mathcal{RP}_t^i(s^{-i}, \zeta^{-i}))\hat{x}_t^i(m) + \sum_{l \in \mathcal{L}_i} q^{-i,l} a^{i,l} \hat{x}^i(m), \tag{8a}$$

$$\text{pr}\beta_t^i(m) = (\beta_t^i - y_t^{i+1})^2, \tag{8b}$$

$$\text{con}^{i,l}(m) = (q^{i,l} - q^{-i,l})^2 + q^{i,l}(b^l - \sum_{j \neq i} a^{j,l} y^j - a^{i,l} \beta^{i-1}), \tag{8c}$$

$$\text{con}_t^i(m) = (s_t^i - s_t^{-i})^2 + s_t^i (z^{-i} - \zeta_t^{-i}), \tag{8d}$$

and

$$s_t^{-i} = \frac{1}{N-1} \sum_{j \neq i} s_t^j \quad \forall i\ \forall t, \tag{9a}$$

$$q^{-i,l} = \frac{1}{N-1} \sum_{j \neq i} q^{j,l} \quad \forall i\ \forall l, \tag{9b}$$

$$\zeta_t^{-i} = \sum_{j \neq i} y_t^j + \beta_t^{i-1}, \quad \forall i\ \forall t, \tag{9c}$$

$$z^{-i} = \max_t \left\{ \zeta_t^{-i} \right\} \quad \forall i, \tag{9d}$$

and $a^{i,l}$ is defined as $a^{i,l} = [a_1^{i,l}, \ldots, a_T^{i,l}]$.

The tax function for user i consists of three parts. The first part $\text{cost}^i(m)$ is the cost for the demand. According to this part, user i pays a fixed price and the peak price for their demand. Note that the peak price at time t, $\mathcal{RP}_t^i(s^{-i}, \zeta^{-i})$, is generated by the vector of peak prices from all other agents, s^{-i}, and the total demand from all other agents (agent i's demand at time t is approximated by β_t^{i-1}). As a result, the peak price is not controlled by user i at all. The second part $\text{pr}\beta_t^i(m)$ ($\text{pr}\beta$ stands for "proxy-β") is a penalty term for the imprecision of prediction β^i that incentivizes β^i to align with y^{i+1} at NE. The third part consists of two penalty terms $\text{con}^{i,l}(m)$ and $\text{con}_t^i(m)$ for each constraint $l \in \mathcal{L}$ and each peak demand inequality $t \in \{1, \ldots, T\}$, respectively. Both of them have a quadratic term that incentivizes consensus of the messages $q^{i,l}$ and s_t^i among agents, respectively. In addition, they possess a form that looks similar to the complementary slackness conditions (4d) and (4e). This special design facilitates the suggested price to come to an agreement and ensures that the primal feasibility and complementary slackness hold at NE, which are shown in Lemma 2.

The main property we want from this mechanism is full implementation. We expect the allocation scheme under the NE of the mechanism-induced game to coincide with that of the original optimization problem. Full implementation can be shown in two steps. First, we show that, if there is a (pure strategy) NE, it must induce the optimal allocation. Then, we prove the existence of such (pure strategy) NE.

From the form of the tax functions, we can immediately obtain the following lemma.

Lemma 1. *At any NE, for each user i, the demand proxy β_t^i is equal to the demand of their neighbor, i.e., $\beta_t^i = y_t^{i+1}$ for all t.*

Proof. Suppose that m is an NE, where there exists at least one user i whose message β^i does not agree with the next user's demand, i.e., $\beta^i \neq y^{i+1}$. Say, $\beta_t^i \neq y_t^{i+1}$ for some t. Then, we can find a profitable deviation $\tilde{m}$ that keeps everything other than β^i the same as m but modifies β_t^i with $\tilde{\beta}_t^i = y_t^{i+1}$. Compare the payoff value u_i before and after the deviation:

$$\begin{aligned} u_i(\tilde{m}) - u_i(m) =& -(\tilde{\beta}_t^i - y_t^{i+1})^2 + (\beta_t^i - y_t^{i+1})^2 \\ =& (\beta_t^i - y_t^{i+1})^2 > 0. \end{aligned}$$

Thus, if there is some $\beta^i \neq y^{i+1}$, user i can always construct another announcement $\tilde{m}^i$, such that user i gets a better payoff. □

It can be seen from Lemma 1 that the messages β play an important role in the mechanism. They appear in two places in the tax functions: first, in the expression of $\zeta_t^{-i} = \sum_{j \neq i} y_t^j + \beta_t^{i-1}$ which is the total demand at time t used in user i's tax function, and second, in the expression for excess demand $b^l - \sum_{j \neq i} a^{j,l} y^j - a^{i,l} \beta^{i-1}$ for the lth constraint. Note that we do not want user i to control these terms with their messages (specifically y_t^i) because they already control their allocation directly and this creates technical difficulties. Indeed, quoting the self-announced demand in the tax function raises the possibility of unexpected strategic moves for user i to obtain extra profit. Instead, using the proxy β^{i-1} instead of y^i eliminates user i's control on their own slackness factor, while Lemma 1 guarantees that, at NE, these quantities become equal.

With the introduction of these proxies, we show in the following lemmas that, at NE, all KKT conditions required for the optimal solution are satisfied. First, we prove that primal feasibility (KKT 1) and complementary slackness (KKT 3) are ensured by the design of the penalty terms "pr"s and constraint-related terms "con"s, if we treat q and $\mathcal{RP}(s, \zeta)$ as the Lagrange multipliers.

Lemma 2. *At any NE, users' suggested prices are equal:*

$$q^{i,l} = q^l,\ \forall l \in \mathcal{L}_i\ \forall i \in \mathcal{N},$$
$$s_t^i = s_t,\ t = 1, \ldots, T,\ \forall i \in \mathcal{N}.$$

Furthermore, users' announced demand profile satisfies $y \in \mathcal{X}$, and equal prices, together with the demand profile, satisfy complementary slackness:

$$q(Ax - b) = 0,$$
$$s_t\left(z - \sum_i y_t^i\right) = 0,\ \forall t = 1, \ldots, T,$$

which implies

$$\mathcal{RP}_t^i(s, \zeta^{-i})\left(z - \sum_i y_t^i\right) = 0,\ \forall t = 1, \ldots, T, \tag{10}$$

where z is the peak demand during the billing period.

Proof. The proof can be found in Appendix B. □

Dual feasibility (KKT 2) holds trivially by definition. We now show that stationarity condition (KKT 4) holds at NE by imposing a first-order condition on the partial derivatives of user i's utility w.r.t. their message component y_t^i.

Lemma 3. *At NE, stationarity holds, i.e.,*

$$\dot{v}_t^i(\hat{x}_t^i(m)) = p_t + \mathcal{RP}_t^i(s, \zeta^{-i}) + \sum_{l \in \mathcal{L}_i} q^l a_t^{i,l}, \tag{11}$$

$$p_0 = \sum_{t=1}^{T} \mathcal{RP}_t^i(s, \zeta^{-i}). \tag{12}$$

Proof. The proof is in Appendix C. □

With Lemmas 1–3, it is straightforward to derive the first part of our result, i.e., efficiency of the allocation at any NE.

Theorem 1. *For the mechanism-induced game $\mathcal{G}$, if NE exists, then the NE results in the same allocation as the optimal solution to the centralized problem (3a)–(3c).*

Proof. If m^* is an NE, from Lemmas 1 and 2, we know that, at NE, $\beta^{i*} = y^{i+1}$ and that all of the prices q^{i*}, s^{i*}, and all of the ζ^{-i*} are the same among all users $i \in \mathcal{N}$. We denote these equal quantities by y^*, q^*, s^* and ζ^*.

Consider the solution $sol = (x, w, \lambda, \mu) = (y^*, \max_t\{\zeta_t^*\}, q^*, \mathcal{RP}(s^*, \zeta^*))$. From Lemma 2, the solution sol satisfies (4a)–(4e) (primal feasibility and complementary slackness). From Lemma 3, sol has (4g) and (4f) (stationarity). The dual feasibility (4c) holds because of the nonnegativity of q and s.

Therefore, sol satisfies all four KKT conditions, which means that the allocation $\hat{x}(m^*)$ is the optimal allocation. □

The following theorem shows the existence of NE.

Theorem 2. *For the mechanism-induced game $\mathcal{G}$, there exists at least one NE.*

Proof. From the theory of convex optimization, we know that the optimal solution of (3a)–(3c) exists. Based on this solution, one can construct a message profile that satisfies all

of the properties we present in Lemmas 1–3 and prove that there is no unilateral deviation for all users. The details are presented in Appendix D. □

Full implementation indicates that, if all users are willing to participate in the mechanism, the equilibrium outcome is the optimal allocation. For each user i, the payoff at NE is

$$u_i(m^*) = v^i(\hat{x}^i(m^*)) - \sum_{t=1}^{T} \underbrace{\left(p_t + \mathcal{RP}_t^i(s, \zeta^{-i}) + \sum_{l \in \mathcal{L}_i} q^{-i,l} a_t^{i,l} \right)}_{\text{Aggregated unit price for } \hat{x}_t^i} \hat{x}_t^i(m^*). \tag{13}$$

In other words, the users pay for their own demands via the aggregated unit prices given by the consensus at NE. By counting the planner as a participant of the mechanism with utility $\sum_{i \in \mathcal{N}} \hat{t}^i(m^*) - J(x^*)$, a strong budget balance is automatically achieved. However, there are still two questions remaining. Are the users willing to follow this mechanism or would they rather not participate? Will the planner have to pay extra money for implementing such mechanism? The two theorems below answer these questions.

Theorem 3 (Individual Rationality for Users)**.** *Assume that agent i obtains $x^i = 0$ and pays nothing if they choose not to participate in the mechanism. Then, at NE, participating in the mechanism is weakly better than not participating, i.e.,*

$$u_i(m^*) \geq v^i(0).$$

Proof. The main idea for the proof of Theorem 3 is to find a message profile with m^{-i*}, in which user i's payoff is $v^i(0)$, and then, we can argue that following NE is not worse since m^* is a best response to m^{-i*}. The details of the proof can be found in Appendix E. □

Theorem 4 (Individual Rationality for the Planner)**.** *At NE, the planner does not need to pay extra money for the mechanism:*

$$\sum_{i \in \mathcal{N}} \hat{t}^i(m^*) - J(\hat{x}(m^*)) \geq 0. \tag{14}$$

Moreover, by slight modification of the tax functions defined in (7), the total payment of users and the energy cost achieve a balance at NE:

$$\sum_{i \in \mathcal{N}} \tilde{t}^i(m^*) - J(\hat{x}(m^*)) = 0. \tag{15}$$

Proof. The individual rationality of the planner can be verified by substituting m^* in (14) directly. By redistributing the income of the planner back to the users in a certain way, the total payment of users is exactly $J(\hat{x}(m^*))$, and consequently, no money is left after paying the energy company. The details are left to Appendix F. □

4. Distributed Mechanism

In the previous mechanism, allocation functions and tax functions of users depend on the global message profile m. If one wants to compute the tax $\hat{t}^i$ for a certain user i, all messages m^j for all $j \in \mathcal{N}$ are needed. Such mechanisms are not desirable for environments with communication constraints, where such a global message exchange is restricted. To tackle this problem, we provide a distributed mechanism, in which the calculation of the allocation and tax of a certain user depends only on the messages from the "available" users and therefore satisfies the communication constraints. In this section, we first introduce communication constraints using a message exchange network model.

We then develop a distributed mechanism, which accommodates the communication constrains and preserves the desirable properties of the baseline centralized mechanism.

4.1. Message Exchange Network

In an environment with communication constraints, all of the users are organized in a undirected graph $\mathcal{GR} = (\mathcal{N}, \mathcal{E})$, where the set of nodes $\mathcal{N}$ is the set of users and the set of edges $\mathcal{E}$ indicates the accessibility to the message for each user. If $(i, j) \in \mathcal{E}$, user i can access the message of user j, i.e., the message of j is available for user i when computing the allocation and tax of user i, and vice versa. Here, we state a mild requirement for the message exchange network:

Assumption 3. *The graph $\mathcal{GR}$ is a connected and undirected graph.*

In fact, the mechanism we show works for the cases where $\mathcal{GR}$ is an undirected tree. Although an undirected connected graph is not necessarily a tree, since we can always find a spanning tree from such graph, it is safe to consider the mechanism under the assumption that the given network has a tree structure. If that is not the case, the mechanism designer can claim a spanning tree from the original message exchange network and design the mechanism only based on the tree instead of the whole graph (essentially some of the connections of the original graph are never used for message exchanges).

The basic idea behind the decentralized modification of the baseline mechanism is intuitively straightforward. Looking at the tax function for user i in the centralized mechanism, we observe that several messages required do not come from i's immediate neighbors. For this reason, we define new "summary" messages that are quoted by i's neighbors and represent the missing messages. At the same time, for this to work, we add additional penalty terms that guarantee that the summary messages indeed represent the needed terms at NE.

Notice that, in the previous mechanism, user i is expected to announce a β_t^i equal to the demand of the next user $(i+1)$. However, here, we might have $(i, i+1) \notin \mathcal{E}$, and owing to the communication constraint, we are not able to compare β_t^i with y_t^{i+1}. Instead, β_t^i should be a proxy of the demand of user i's direct neighbor. This motivates us to define the function $\phi(i)$, where $\phi(i) \in \mathcal{N}(i)$; $\mathcal{N}(i)$ is the set of user i's neighbors (excluding i); and $\phi(i) = j$ denotes that, in user i's tax function, the proxy variable β used for user i's $\text{con}_i^l(m)$ terms in their tax function is provided by user j. In other words, $\phi(i)$ is a "helper" for user i who quotes a proxy of their demand whenever needed.

In the next part, we use the summaries of the demands to deal with the distributed issue. For the sake of convenience, we define $n(i, k)$ as the nearest user to user k among the neighbors of user i and user i itself. $n(i, k)$ is well-defined because of the tree structure. The proof is omitted here. The details can be found in ([72] Chapter 4, Section 7.1).

4.2. The Message Space

In the distributed mechanism, the message m^i in user i's message space $\mathcal{M}^i$ is defined as

$$m^i = \Big(\left\{ y_t^i \right\}_{t=1}^T, \left\{ q^{i,l} \right\}_{l \in \mathcal{L}}, \left\{ s_t^i \right\}_{t=1}^T, \left\{ \beta_t^{i,j} : \phi(j) = i \right\}_{t=1}^T, \left\{ n^{i,j,l} : j \in \mathcal{N}(i) \right\}_{l \in \mathcal{L}}, \left\{ \nu_t^{i,j} : j \in \mathcal{N}(i) \right\}_{t=1}^T \Big). \tag{16}$$

Here, $n^{i,j,l}$ is a summary for demands of users related to constraint l and connected to user i via j, as depicted in Figure 1. Message $\nu_t^{i,j}$ serves a similar role for the peak demand.

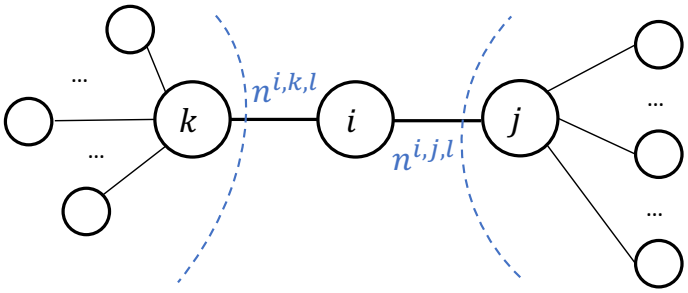

Figure 1. Proxies in the Message Exchange Network: for constraint l, user i announces $n^{i,k,l}$ as a summary of demands for the tree on the left of i (starting from k) and $n^{i,j,l}$ as a summary of demands for the tree on the right of i (starting from j).

4.3. The Allocation and Tax Functions

The allocation functions $\hat{x}_t^i(m) = y_t^i$ are still straightforward. There are some modifications on tax functions, including adjustments on prices, consensus of new variables, and terms for complementary slackness.

$$\begin{aligned}\hat{t}^i(m) = \mathrm{cost}^i(m) + \textstyle\sum_l (\mathrm{pr}n^{i,l}(m) + \mathrm{con}^{i,l}(m)) \\ + \textstyle\sum_t (\mathrm{pr}\beta_t^i(m) + \mathrm{pr}v_t^i(m) + \mathrm{con}_t^i(m)),\end{aligned} \tag{17}$$

where

$$\mathrm{cost}^i(m) = \sum_{t=1}^{T} (p_t + \mathcal{RP}_t^{-i}(s^{-i}, \zeta^{-i}))\hat{x}_t^i(m) + \sum_{l \in \mathcal{L}_i} q^{-i,l} a^{i,l} \hat{x}^i(m), \tag{18a}$$

$$\mathrm{con}^{i,l}(m) = (q^{i,l} - q^{-i,l})^2 + q^{i,l}(b^l - \sum_{j \in \mathcal{N}(i)} f^{i,j,l} - a^{i,l}\beta^{\phi(i),i}), \tag{18b}$$

$$\mathrm{con}_t^i(m) = (s_t^i - s_t^{-i})^2 + s_t^i(z^{-i} - \zeta_t^{-i}), \tag{18c}$$

$$\mathrm{pr}n^{i,l}(m) = \sum_{j \in \mathcal{N}(i)} \left(n^{i,j,l} - f^{i,j,l}\right)^2, \tag{18d}$$

$$\mathrm{pr}\beta_t^i(m) = \sum_{j:\phi(j)=i} (\beta_t^{i,j} - y_t^j)^2, \tag{18e}$$

$$\mathrm{pr}v_t^i(m) = \sum_{j \in \mathcal{N}(i)} \left(v_t^{i,j} - f_t^{i,j}\right)^2 \tag{18f}$$

$$f^{i,j,l} = a^{j,l} y^j + \sum_{h \in \mathcal{N}(j) \setminus \{i\}} n^{j,h,l}, \tag{18g}$$

$$f_t^{i,j} = y_t^j + \sum_{h \in \mathcal{N}(j) \setminus \{i\}} v_t^{j,h}. \tag{18h}$$

and

$$s_t^{-i} = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} s_t^j \quad \forall i \, \forall t, \tag{19a}$$

$$q^{-i,l} = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} q^{j,l} \quad \forall i \, \forall l, \tag{19b}$$

$$\zeta_t^{-i} = \sum_{j \in \mathcal{N}(i)} f_t^{i,j} + \beta_t^{\phi(i),i}, \quad \forall i \, \forall t, \tag{19c}$$

$$z^{-i} = \max_t \left\{ \zeta_t^{-i} \right\} \quad \forall i. \tag{19d}$$

In order to intuitively see how the decentralized mechanism works, take as an example the term $\text{con}^{i,l}(m)$ in (18b), which is a modified version of (8c), repeated here for convenience $\text{con}^{i,l}(m) = (q^{i,l} - q^{-i,l})^2 + q^{i,l}(b^l - \sum_{j\neq i} a^{j,l}y^j - a^{i,l}\beta^{i-1})$ related to the l-th constraint. Other than the quadratic term, which is identical in both expressions, the difference between the centralized and decentralized versions is in the expressions $\sum_{j\neq i} a^{j,l}y^j + a^{i,l}\beta^{i-1}$ and $\sum_{j\in\mathcal{N}(i)} f^{i,j,l} + a^{i,l}\beta^{\phi(i),i}$, respectively. The second term in each of these expressions relates to the proxy β^{i-1}, which in the decentralized version is substituted by the proxy $\beta^{\phi(i),i}$ due to the fact that the proxy for y^i is not provided by user $i-1$ anymore but is provided by user i's helper $\phi(i)$. The first term, $\sum_{j\neq i} a^{j,l}y^j = \sum_{j\in\mathcal{N}(i)} a^{j,l}y^j + \sum_{j\notin\mathcal{N}(i)\cup\{i\}} a^{j,l}y^j$, which cannot be directly evaluated in the decentralized version (since it depends on messages outside the neighborhood of i) is now evaluated as $\sum_{j\in\mathcal{N}(i)} f^{i,j,l} = \sum_{j\in\mathcal{N}(i)} a^{j,l}y^j + \sum_{j\in\mathcal{N}(i)}\sum_{h\in\mathcal{N}(j)\setminus\{i\}} n^{j,h,l}$. It should now be clear that the role of the new messages $n^{j,h,l}$ quoted by the neighbors $j\in\mathcal{N}(i)$ of i is to summarize the total demands of other users. Furthermore, the additional quadratic penalty terms has to effectuate this equality. This idea is made precise in the next section.

4.4. Properties

It is clear that this mechanism is distributed, since all of the messages needed for the allocation and tax functions of user i come from neighborhood $\mathcal{N}(i)$ and the user themselves. Due to way the messages and taxes are designed, the proposed mechanism satisfies properties similar to those in Lemmas 2 and 3, and, consequently, Theorems 1 and 2. The reason is that the components n and ν behave in the same manner as the absent y^h, $h\notin\mathcal{N}(i)$ in user i's functions at NE, which makes the proofs of the properties in previous mechanism still work here. We elaborate on these properties in the following.

Lemma 4. *At any NE, we have the following results regarding the proxy messages:*

$$\beta_t^{i,j} = y_t^j, \qquad \forall j:\phi(j)=i, \tag{20}$$

$$n^{i,j,l} = a^{j,l}y^j + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} n^{j,h,l}, \qquad \forall i,\forall j\in\mathcal{N}(i),\forall l\in\mathcal{L}, \tag{21}$$

$$\nu_t^{i,j} = y_t^j + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} \nu_t^{j,h}, \qquad \forall t,\forall i,\forall j\in\mathcal{N}(i). \tag{22}$$

Proof. $\beta^{i,j}, n^{i,j,l}$, and $\nu_t^{i,j}$ only appear in the quadratic penalty terms of user i's tax function. Therefore, for any user i, the only choice to minimize the tax is to bid $\beta^{i,j}, n^{i,j,l}$, and $\nu_t^{i,j}$ by (20)–(22). □

Now, based on the structure of the message exchange network, we have

Lemma 5. *At any NE, $n^{i,j,l}$ and $\nu_t^{i,j}$ satisfy*

$$n^{i,j,l} = \sum_{t=1}^{T}\sum_{h:n(i,h)=j} a_t^{h,l}y_t^h, \qquad \forall i\in\mathcal{N},\forall j\in\mathcal{N}(i),\forall l\in\mathcal{L}, \tag{23}$$

$$\nu_t^{i,j} = \sum_{h:n(i,h)=j} y_t^h, \forall i\in\mathcal{N}, \qquad \forall j\in\mathcal{N}(i),\forall t\in\mathcal{T}. \tag{24}$$

Proof. The proof is presented in Appendix G. □

With Lemma 5, we immediately obtain the following results.

Lemma 6. *At any NE, for all user i, we have*

$$\sum_{j\in\mathcal{N}(i)} f^{i,j,l} = \sum_{j\in\mathcal{N}\setminus\{i\}} a^{j,l}y^j, \quad \forall l \in \mathcal{L}_i, \tag{25}$$

$$\sum_{j\in\mathcal{N}(i)} f_t^{i,j} = \sum_{j\in\mathcal{N}\setminus\{i\}} y_t^j, \quad \forall t \in \mathcal{T}. \tag{26}$$

Proof. At NE, by directly substitution,

$$\begin{aligned}\sum_{j\in\mathcal{N}(i)} f^{i,j,l} &= \sum_{j\in\mathcal{N}(i)} \left(a^{j,l}y^j + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} n^{j,h,l} \right) \\ &= \sum_{j\in\mathcal{N}(i)} \left(a^{j,l}y^j + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} \sum_{k:n(j,k)=h} a^{k,l}y^k \right) \\ &= \sum_{j\in\mathcal{N}(i)} \left(\sum_{h:n(i,h)=j} a^{h,l}y^h \right) = \sum_{j\in\mathcal{N}\setminus\{i\}} a^{j,l}y^j, \quad \forall l \in \mathcal{L}_i,\end{aligned}$$

The third equality holds by the fact that the users in set $\{k|h \in \mathcal{N}(j)\setminus\{i\}, n(j,k) = h\}$ are the ones that are not in the subtree starting from a single branch (j,i) with root j, which is exactly $\{k|n(i,k) = j\}\setminus\{j\}$.

Equation (26) holds for a similar reason. □

Lemma 6 plays a similar role to Lemma 1. With Lemma 6, the properties in Lemmas 2 and 3 can be reproduced in the distributed mechanism. We then obtain the following theorem.

Theorem 5. *For the mechanism-induced game $\mathcal{G}$, NE exists. Furthermore, any NE of game $\mathcal{G}$ induces the optimal allocation.*

Proof. By substituting (25) and (26) in (17), we obtain exactly the same form of tax function in a centralized mechanism on equilibrium, which yields desirable results, as shown in Lemmas 2 and 3. We conclude that any NE induces the optimal allocation. The existence of NE can be proved by a construction similar to that of Theorem 2. □

As was true in the baseline centralized mechanism, in the distributed case, the planner may also have concerns about whether the users have an incentive to participate and whether the mechanism requires external sources of funds to maintain the balance. As it turns out, Theorems 3 and 4 still hold here. As a result, the users are better off joining the mechanism, and the market has a balanced budget. The proofs and the construction of the subsidies can be performed in a manner similar to the centralized case and therefore are omitted.

5. Learning Algorithm

5.1. A Learning Algorithm for the Centralized Mechanism

The property of full implementation ensures that social welfare maximization can be reached if all participants reach NE in the mechanism-induced game and no one obtains a unilateral profitable deviation at NE. Nevertheless, it is troublesome for participants to anticipate NE as being the outcome if none of them knows (or can calculate) NE without knowledge of other users' utilities. To settle this issue, one can design a learning algorithm to help participants learn the NE in an online fashion. In this section, we present such a learning algorithm for the centralized mechanism discussed in Section 3. Instead of using Assumption 1, here, we make a stronger assumption in order to obtain a convergent algorithm.

Assumption 4. *All of the utility functions $v_t^i(\cdot)$s are proper, twice differentiable concave functions with δ-strong concavity.*

Here, δ-strong concavity of a function $g(\cdot)$ is defined by the δ-strong convexity of $-g(\cdot)$. A function $f(\cdot)$ is strongly convex with parameter δ if

$$f(y) \geq f(x) + \nabla f(x)^{\mathsf{T}}(y-x) + \frac{\delta}{2}||y-x||^2.$$

The design of the learning algorithm involves three steps. First, we find the relation between NE and the optimal solution of the original optimization problem. This step was performed in the proof of Theorem 1: we see in NE that y^* coincides with x^* in the optimal allocation, that q^{i*} equals λ^*, and that the components of s^{i*} are proportional to the components of μ^*. Then, by Slater's condition, strong duality holds here, so we connect the Lagrange multipliers λ^*, μ^* with the optimal solution of the dual problem. Due to the strong concavity of the utilities and stationarity, given λ^* and μ^*, the optimal allocation x^* can be uniquely determined. Finally, if we can find an algorithm to solve the dual problem, the design is completed.

The first two steps are straightforward. For the third one, we can see that the dual problem is also a convex optimization problem, so projected gradient descent (PGD) is one of the choices for the learning algorithm. The proof of convergence of PGD is not trivial. In the proof developed in [29], the convergence of PGD holds when (a) the objective function is β-smooth and (b) the feasible set is closed and convex. In Appendix H, we show that (a) is satisfied by Assumption 4. To check (b), we need to find a feasible set for the dual variables. Since in PGD of the dual problem, the gradient of the dual function turns out to be a combination of functions of the form $(\dot{v}_t^i)^{-1}(\cdot)$, the feasible set should satisfy two requirements: first, all of the elements are in the domain of the dual function's gradient in order to make every iteration valid; second, (λ^*, μ^*) is in the feasible set so that we do not miss it. With these requirements in mind, we make Assumption 5 and construct a feasible set for the dual problem based on that.

Assumption 5. *For each utility $v_t^i(\cdot)$, there exist $\underline{r}_t^i, \bar{r}_t^i \in \mathbb{R}$ satisfying*

1. $\forall x \in \mathcal{X}$, $\dot{v}_t^i(x_t^i) \in [\underline{r}_t^i, \bar{r}_t^i]$,
2. $\forall p \in [\underline{r}_t^i, \bar{r}_t^i]$, $\exists x_t^i \in \mathbb{R}, s.t.\ \dot{v}_t^i(x_t^i) = p$.

Before we explain this assumption, we define
$\underline{r} = [\underline{r}_1^1, \ldots, \underline{r}_T^1, \ldots, \underline{r}_T^N]^{\mathsf{T}}$, $\bar{r} = [\bar{r}_1^1, \ldots, \bar{r}_T^1, \ldots, \bar{r}_T^N]^{\mathsf{T}}$,
$p = [p_1 \ldots p_T]^{\mathsf{T}}$, $\tilde{p} = 1_N \otimes p$ and

$$\tilde{A} = \begin{pmatrix} A \\ 1_N^{\mathsf{T}} \otimes I_T \end{pmatrix}, \ \tilde{\lambda} = \begin{pmatrix} \lambda \\ \mu \end{pmatrix}, \ \tilde{p} = 1_N \otimes p. \tag{27}$$

where $\otimes$ represents the Kronecker product of matrices. Then, we define a set of proper prices $\mathcal{P}$ as the feasible set for the dual problem:

$$\mathcal{P} = \{\tilde{\lambda} = (\lambda, \mu) \geq 0 : \underline{r} \leq \tilde{A}^{\mathsf{T}}\tilde{\lambda} + \tilde{p} \leq \bar{r}, \ 1_T^{\mathsf{T}}\mu = p_0\}.$$

Observe that, by stationarity, the $((i-1)T+t)$th entry of $\tilde{A}^{\mathsf{T}}\tilde{\lambda} + \tilde{p}$ equals $\dot{v}_t^i(x_t^{i*})$ in the optimal solution. Consequently, Assumption 5 implies two things: first, $\tilde{\lambda}^* \in \mathcal{P}$; second, all $\tilde{A}^{\mathsf{T}}\tilde{\lambda} + \tilde{p}$ can be a vector of $\dot{v}_t^i$s on some $x \in \mathbb{R}^{NT}$ if $\tilde{\lambda} \in \mathcal{P}$. Hence, with Assumption 5, it is safe to narrow down the feasible set of the dual problem to $\mathcal{P}$ without changing the optimal solution. Furthermore, for all of the price vectors in $\mathcal{P}$, $(\dot{v}_t^i)^{-1}(\cdot)$ in PGD can be evaluated. Back to condition (b) stated above, since $\mathcal{P}$ is closed and convex, PGD is convergent in this case.

Based on all of the assumptions and the PGD method, we propose Algorithm 1 as a learning algorithm for the NE of the centralized mechanism.

The convergence of PGD yields the convergence of proposed learning algorithm:

Theorem 6. *Choose a step size $\alpha \leq \delta'/\|A\|$, where $\|A\|$ is A's spectral norm and δ' is the parameter of strong concavity of the centralized objective function. As the number of iterations K grows, the distance between the computed price vector $(q(K), s(K))$ and the optimal price vector (q^*, s^*) is non-increasing. Furthermore, $\lim_{K\to\infty} m(K) = m^*$, where m^* is the NE.*

Proof. See Appendix H. □

Algorithm 1: Learning algorithm for the centralized mechanism.

Data: Time index k, a set of proper prices $\mathcal{P}$, a vector of initial prices $(q^0, s^0) \in \mathcal{P}$, message profiles $m(k)$, iteration step size α, number of iterations K.
Result: Message profile $m(K) = (y(K), \beta(K), q(K), s(K))$.

1 $k = 0$, $q^{i,l}(0) = q^{0,l}$, $s^i_t(0) = s^0_t$, $\forall i \in \mathcal{N}, l \in \mathcal{L}, t \in \mathcal{T}$;
2 $y^i_t(0) = (v^i_t)^{-1}(p_t + \sum_{l\in\mathcal{L}} a^{i,l} q^{i,l}(0) + s^i_t(0))$, $\forall i \in \mathcal{N}, t \in \mathcal{T}$;
3 **while** $k < K$ **do**
4 $\quad \tilde{q}^{i,l}(k+1) = q^{i,l}(k) - \alpha(b^l - a^l y(k))$, $\forall i \in \mathcal{N}, l \in \mathcal{L}$;
5 $\quad \tilde{s}^i_t(k+1) = s^i_t(k) + \alpha \sum_j y^j_t(k)$, $\forall i \in \mathcal{N}, t \in \mathcal{T}$;
6 $\quad (q^i(k+1), s^i(k+1)) = \mathbf{Proj}_{\mathcal{P}}(\tilde{q}^i(k+1), \tilde{s}^i(k+1))$, $\forall i \in \mathcal{N}$;
7 $\quad y^i_t(k+1) = (v^i_t)^{-1}(p_t + \sum_{l\in\mathcal{L}} a^{i,l}_t q^{i,l}(k+1) + s^i_t(k+1))$, $\forall i \in \mathcal{N}, t \in \mathcal{T}$;
8 $\quad k \leftarrow k+1$;
9 **end**
10 $\beta^i_t(K) = y^{i+1}_t(K)$ $\forall i \in \mathcal{N}, t \in \mathcal{T}$;

5.2. A Learning Algorithm for the Distributed Mechanism

Algorithm 1 cannot be applied in a distributed environment because lines 4 and 5 for the price adjustments require global user messages. Luckily, the price adjustment is the only part requiring messages from other users during an iteration. Based on this fact, the main modification of the algorithm is to figure out a way for users to obtain the aggregated demand of each constraint, which was indeed the same problem we needed to address when we modified the centralized mechanism to a distributed one. Naturally, we can utilize the proxies ν, n to help exchange the necessary messages, as we did in the design of the distributed mechanism. If the proxies ν and n behave the same way as described in Lemma 5, then we can modify line 4 and 5 in Algorithm 1 as follows and we can expect the same output as that of the original algorithm:

$$\tilde{q}^{i,l}(k+1) = q^{i,l}(k) - \alpha\left(b^l - \sum_{j\in\mathcal{N}(i)}\left(a^{j,l}y^j(k) + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} n^{j,h,l}(k)\right) - a^{i,l}y^i(k)\right), \quad \forall i \in \mathcal{N}, l \in \mathcal{L}, \tag{28}$$

$$\tilde{s}^i_t(k+1) = s^i_t(k) + \alpha\left(\sum_{j\in\mathcal{N}(i)}\left(y^j_t(k) + \sum_{h\in\mathcal{N}(j)\setminus\{i\}} v^{j,h}_t(k)\right) + y^i_t(k)\right), \quad \forall i \in \mathcal{N}, t \in \mathcal{T}. \tag{29}$$

To make sure the proxies behave the same way as described in Lemma 5, some additional message exchanges are required in the beginning of each iteration before price adjustments. We design the maintenance algorithm for n, ν in Algorithms 2 and 3.

It is worth noting that a deadlock does not occur during the proxy maintenance and that the proxies after maintenance behave the way we expect in Lemma 5. The reason is that the distributed mechanism uses an acyclic subgraph of the network, so there are no loops. Notice that the proof of Lemma 5 (see Appendix G) uses an iterative process to justify the statement. Indeed, what happens during the maintenance process described

here is exactly the same as the process depicted in the proof of Lemma 5. The termination signal is used to notify users when to proceed to the price adjustment step.

With the assistance of proxy maintenance Algorithms 2 and 3, we arrive at an extended version of the original learning algorithm for the distributed mechanism stated as Algorithm 4.

Algorithm 2: Maintenance algorithm for proxy $n^{i,j,l}$.

Data: Time index k, messages $m^h(k)$ for $h \in \mathcal{N}(i)$.
Result: The proxy value $n^{i,j,l}(k)$, a termination signal.
while $y^j(k)$ *or* $n^{j,h,l}(k)$ *for some* $h \in \mathcal{N}(j)\backslash\{i\}$ *is not calculated* **do**
  wait;
end
$n^{i,j,l}(k) = a^{j,l}y^j + \sum_{h\in\mathcal{N}(j)\backslash\{i\}} n^{j,h,l}(k)$;
Broadcast the termination signal of $n^{i,j,l}$ to all the neighbor $l \in \mathcal{N}(i)$;

Algorithm 3: Maintenance algorithm for proxy $\nu_t^{i,j}$.

Data: Time index k, messages $m^h(k)$ for $h \in \mathcal{N}(i)$.
Result: The proxy value $\nu_t^{i,j}(k)$, a termination signal.
while $y_t^j(k)$ *or* $\nu_t^{j,h}(k)$ *for some* $h \in \mathcal{N}(j)\backslash\{i\}$ *is not calculated* **do**
  wait;
end
$\nu_t^{i,j}(k) = y_t^j(k) + \sum_{h\in\mathcal{N}(j)\backslash\{i\}} \nu_t^{j,h}(k)$;
Broadcast the termination signal of $\nu_t^{i,j}$ to all the neighbor $l \in \mathcal{N}(i)$;

Algorithm 4: Learning algorithm for user i in the distributed mechanism.

Data: Time index k, a set of proper prices $\mathcal{P}$, a vector of initial prices $(q^0, s^0) \in \mathcal{P}$, message profiles $m(k)$, iteration step size α, number of iterations K.
Result: Message $m^i(K)$ with form (16).
$k = 0$, $q^{i,l}(0) = q^{0,l}$, $s_t^i(0) = s_t^0$, $\quad \forall i \in \mathcal{N}, l \in \mathcal{L}, t \in \mathcal{T}$;
$y_t^i(0) = (\ddot{v}_t^i)^{-1}(p_t + \sum_{l\in\mathcal{L}} a^{i,l}q^{i,l}(0) + s_t^i(0)), \quad \forall i \in \mathcal{N}, t \in \mathcal{T}$;
while $k < K$ **do**
  Run Algorithms 2 and 3 for $n^{i,j,l}(k)$ and $\nu_t^{i,j}(k)$ $\forall j \in \mathcal{N}(i), l \in \mathcal{L}, t \in \mathcal{T}$ simultaneously;
  Wait for the termination signals with timestamp k from all the neighbors $j \in \mathcal{N}(i)$ for all of their proxy maintenances;
  Price adjustments (28) and (29), $\forall i \in \mathcal{N}, l \in \mathcal{L}, t \in \mathcal{T}$;
  $(q^i(k+1), s^i(k+1)) = \mathbf{Proj}_{\mathcal{P}}(\tilde{q}^i(k+1), \tilde{s}^i(k+1)), \quad \forall i \in \mathcal{N}$;
  $y_t^i(k+1) = (\ddot{v}_t^i)^{-1}(p_t + \sum_{l\in\mathcal{L}} a_t^{i,l}q^{i,l}(k+1) + s_t^i(k+1)), \quad \forall i \in \mathcal{N}, t \in \mathcal{T}$;
  $k \leftarrow k+1$;
end
Run Algorithms 2 and 3 to obtain $n^{i,j,l}(K), \nu_t^{i,j}(K)$ $\forall j \in \mathcal{N}(i), l \in \mathcal{L}, t \in \mathcal{T}$;
$\beta_t^{i,j}(K) = y_t^j(K) \quad \forall j : \phi(j) = i, t \in \mathcal{T}$;

With the support of Lemma 5 and Theorem 6, this modified learning algorithm is also guaranteed to converge to NE in the distributed mechanism. Nevertheless, this algorithm is not as fast as the original one. We can observe that one user can do nothing but wait if any of the dependencies for their calculations is absent. As a result, this scheme has a mixed sequential and concurrent operation. To be more precise, the extra time for the modified algorithm is proportional to the length of the longest path of the chosen spanning tree in the network (one can derive this result following the procedures detailed in Lemma 5 by

focusing on the steps for the transmission of the message from one side of the network to another side). Thus, the algorithm time complexity is impacted by the network structure.

6. A Concrete Example

To give a sense of how the two mechanisms and the learning algorithm work, we provide a simple non-trivial example here. We first present the original centralized problem for the example, and then identify the NE of the centralized mechanism based on the properties we found. For the distributed mechanism, we illustrate how the proxy variables at NE are determined with a simple example of a message exchange network. Lastly, we implement the learning algorithm for the centralized mechanism.

6.1. The Demand Management Optimization Problem

In the energy community, assume that there are three users in the user set $\mathcal{N} = \{1, 2, 3\}$ and $T = 2$ days in a billing period. Suppose user i on day t has the following utility function:

$$v_t^i(x_t^i) = i \cdot t \cdot \ln(2 + x_t^i).$$

Set $p_1 = 0.1$, $p_2 = 0.2$, and the peak price $p_0 = 0.05$. We adopt the following centralized problem as a concrete example:

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \sum_{t=1}^{2} \sum_{i=1}^{3} i \cdot t \cdot \ln(2 + x_t^i) - J(x) \\ \text{subject to} \quad & x_t^i \geq -1, \ i = 1, 2, 3, \ t = 1, 2, \\ & \sum_{t=1}^{2} (x_t^1 + x_t^2 + x_t^3) \leq 2, \end{aligned}$$

where $J(x) = 0.1 \cdot \sum_{t=1}^{2} t \cdot \left(\sum_{i=1}^{3} x_t^i \right) + 0.05 \cdot \max_t \{ \sum_{i=1}^{3} x_t^i \}$.

The exact solution to this problem is $x_1^{1*} = -1$, $\lambda^{7*} = (249 + \sqrt{106201})/520$, $\mu_2 = 0.05$, and $x_t^{i*} = 2/(\lambda^{7*} + p_t + \mu_t^*) - 2$ for $(i, t) \neq (1, 1)$, $\lambda^{1*} = \lambda^{7*} + p_1 - 1/(x_1^{1*} + 2)$. $\lambda^{l*} = 0$ for $l = 2, \ldots, 6$, $\mu_1^* = 0$. The interested readers can verify it by using KKT conditions. For the sake of convenience, we adopt the following approximate results:

$$(x_1^{1*}, x_2^{1*}, x_1^{2*}, x_2^{2*}, x_1^{3*}, x_2^{3*}) = (-1.0000, -0.5246, -0.3410, 0.9508, 0.4885, 2.4263).$$

The lower bound constraint for x_1^1 and the upper bound constraint for the sum are active. Thus, according to KKT conditions, $\lambda^{l*} = 0$ for $l = 2, \ldots, 6$, $\lambda^{1*} = 0.2056$, and $\lambda^{7*} = 1.1056$ by stationarity. The total demands of Day 1 and Day 2 are -0.8525 and 2.8525, respectively, so Day 2 has the peak demand $w^* = 2.8525$, Day 1 charges no peak price ($\mu_1^* = 0$), and Day 2 has an extra unit peak price $\mu_2^* = 0.05$.

6.2. The Centralized Mechanism

For this example, in the centralized mechanism, user i needs to choose their message m^i with the following components:

$$m^i = (y_1^i, y_2^i, \{q^{i,l}\}_{l=1}^7, s_1^i, s_2^i, \beta_1^i, \beta_2^i).$$

For the sake of brevity, let us take user 1 for example. In this problem setting, user 1 needs to report their demands for two days (y_1^1, y_2^1); suggest a set of prices for constraints 1–7 ($q^{1,1}, \ldots, q^{1,7}$), suggest unit peak prices for two days (quantities s_1^1 and s_2^1 do not necessarily sum up to $p_0 = 0.05$); and lastly, provide proxies β_1^1 and β_2^1 for user 2's demands.

User 1's tax function is

$$\begin{aligned}\hat{t}^1(m) = &\sum_{t=1}^{2}\Big(p_t + \mathcal{RP}_t^1(s^{-1},\zeta^{-1})\Big)y_t^1 - q^{-1,1}y_1^1 - q^{-1,2}y_2^1 + q^{-1,7}(y_1^1+y_2^1)\\ &+\sum_{t=1}^{2}(\beta_t^1 - y_t^2)^2 + (q^{1,1}-q^{-1,1})^2 + q^{1,1}(1+\beta_1^3) + (q^{1,2}-q^{-1,2})^2 + q^{1,2}(1+\beta_2^3)\\ &+\sum_{l=3}^{6}(q^{1,l}-q^{-1,l})^2 + q^{1,3}(1+y_1^2) + q^{1,4}(1+y_2^2) + q^{1,5}(1+y_1^3) + q^{1,6}(1+y_2^3)\\ &+(q^{1,7}-q^{-1,7})^2 + q^{1,7}(2-y_1^2-y_2^2-y_1^3-y_2^3-\beta_1^3-\beta_2^3)\\ &+(s_1^1-s_1^{-1})^2 + s_1^1(z^{-1}-\zeta_1^{-1}) + (s_2^1-s_2^{-1})^2 + s_2^1(z^{-1}-\zeta_2^{-1})\end{aligned}$$

where

$$\begin{aligned}q^{-1,l} &= (q^{2,l}+q^{3,l})/2,\\ s_t^{-1} &= (s_t^2+s_t^3)/2,\\ \zeta_t^{-1} &= \beta_t^3 + y_t^2 + y_t^3,\\ z^{-1} &= \max\Big\{\zeta_1^{-1},\zeta_2^{-1}\Big\},\end{aligned}$$

and according to the definition (6a) and (6b) of the radial pricing operator, we have

$$\mathcal{RP}_t^1(s^{-1},\zeta^{-1}) = \begin{cases}\frac{s_t^{-1}}{s_1^{-1}+s_2^{-1}}p_0, & \text{if } s_1^{-1}+s_2^{-1}>0,\\ p_0, & \text{if } s_1^{-1}=s_2^{-1}=0 \text{ and } \zeta_t^{-1}>\zeta_{t'}^{-1}(t'\neq t),\\ p_0/2, & \text{if } s_1^{-1}=s_2^{-1}=0 \text{ and } \zeta_t^{-1}=\zeta_{t'}^{-1}(t'\neq t),\\ 0, & \text{if } s_1^{-1}=s_2^{-1}=0 \text{ and } \zeta_t^{-1}<\zeta_{t'}^{-1}(t'\neq t).\end{cases}$$

From Theorem 1, we know that, at NE, user 1's message m^{*1} is such that y^1 corresponds to the optimal solution x^1, q^1 equals the optimal Lagrange multiplier λ, β^1 equals y^2, and finally s^1 is proportional to the Lagrange multiplier μ.

6.3. The Distributed Mechanism

In this subsection, we first demonstrate the modifications on message spaces compared with the centralized mechanism and then show how the newly introduced components n and ν work. The specific NE can be determined in a similar way to that of the centralized mechanism and is therefore omitted.

Assume that the energy community has communication constraints with the message exchange network depicted in Figure 2. Apart from the network topology, the ϕ-relation that indicates the responsibility of proxy β is also an important part of distributed mechanism. Here, we set $\phi(1)=2, \phi(2)=1, \phi(3)=2$. Then, for proxy variables $\beta^{\phi(i),i}, i=1,2,3$, $\beta^{2,1}$ in user 1's tax is provided by user 2, $\beta^{1,2}$ in user 2's tax is provided by user 1, and $\beta^{2,3}$ in user 3's tax is provided by user 2.

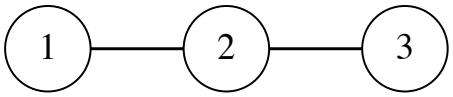

Figure 2. Message exchange network: Users 1 and 2, and users 2 and 3 are neighbors. User 1's message is invisible to user 3, and vice versa.

For this message exchange network, the message components for each user are

$$\begin{aligned}
m^1 =&\Big(y_1^1, y_2^1, \{q^{1,l}\}_{l=1}^7, \{s_t^1\}_{t=1}^2, \{\beta_t^{1,2}\}_{t=1}^2, \{n^{1,2,l}\}_{l=1}^7, \{\nu^{1,2}\}_{t=1}^2\Big),\\
m^2 =&\Big(y_1^2, y_2^2, \{q^{2,l}\}_{l=1}^7, \{s_t^2\}_{t=1}^2, \{\beta_t^{2,1}\}_{t=1}^2, \{\beta_t^{2,3}\}_{t=1}^2,\\
&\qquad \{n^{2,1,l}\}_{l=1}^7, \{n^{2,3,l}\}_{l=1}^7, \{\nu^{2,1}\}_{t=1}^2, \{\nu^{2,3}\}_{t=1}^2\Big),\\
m^3 =&\Big(y_1^3, y_2^3, \{q^{3,l}\}_{l=1}^7, \{s_t^3\}_{t=1}^2, \{n^{3,2,l}\}_{l=1}^7, \{\nu^{3,2}\}_{t=1}^2\Big).
\end{aligned}$$

Therefore, in the distributed mechanism, users are still required to provide their demands y, suggested unit prices q, and suggested unit peak prices s. Different from the centralized mechanism, there are no β among user 3's message components, while user 2 needs to provide two βs, namely $\beta^{2,1}, \beta^{2,3}$. In addition, for each constraint l, every user needs to announce variable n to each of their neighbors; for each day t, every user also needs to provide variable ν to each of their neighbors.

For the rest of this subsection, we focus on user 3 and consider how the n variables play their roles in the tax evaluation. With this message exchange network, we can write down user 3's tax function explicitly:

$$\begin{aligned}
\hat{t}^3(m) =& \sum_{t=1}^{2}\Big(p_t + \mathcal{RP}_t^3(s^{-3}, \zeta^{-3})\Big)y_t^3 - q^{-3,5}y_1^3 - q^{-3,6}y_2^3 + q^{-3,7}(y_1^3 + y_2^3)\\
&+ \sum_{l=1}^{7}\mathrm{pr}n^{3,l}(m) + \sum_{t=1}^{2}\mathrm{pr}\nu_t^3(m) + \sum_{l=1}^{7}(q^{3,l} - q^{-3,l})^2\\
&+ q^{3,1}(1 - n^{2,1,1}) + q^{3,2}(1 - n^{2,1,2}) + q^{3,3}(1 + y_1^2 - n^{2,1,3}) + q^{3,4}(1 + y_2^2 - n^{2,1,4})\\
&+ q^{3,5}(1 + \beta_1^2 - n^{2,1,5}) + q^{3,6}(1 + \beta_2^2 - n^{2,1,6}) + q^{3,6}(1 + \beta_2^2 - n^{2,1,6})\\
&+ q^{3,7}(2 - \beta_1^2 - \beta_2^2 - y_1^2 - y_2^2 - n^{2,1,7}) + \sum_{t=1}^{2}\Big((s_t^3 - s_t^{-3})^2 + s_t^3(z^{-i} - \zeta_t^{-3})\Big),
\end{aligned}$$

where

$$\begin{aligned}
\mathrm{pr}n^{3,l}(m) &= (n^{3,2,l} - n^{2,1,l})^2, \text{ for } l = 1,2,5,6,\\
\mathrm{pr}n^{3,3}(m) &= (n^{3,2,3} + y_1^2 - n^{2,1,3})^2,\\
\mathrm{pr}n^{3,4}(m) &= (n^{3,2,4} + y_2^2 - n^{2,1,4})^2,\\
\mathrm{pr}n^{3,7}(m) &= (n^{3,2,7} - y_1^2 - y_2^2 - n^{2,1,7})^2,\\
\mathrm{pr}\nu_t^3(m) &= (\nu_t^{3,2} - y_t^2 - \nu_t^{2,1})^2,
\end{aligned}$$

and

$$\begin{aligned}
q^{-3,l} &= q^{2,l},\ l = 1, \ldots, 7,\\
s_t^{-3} &= s_t^2,\ t = 1,2,\\
\zeta_t^{-3} &= \beta_t^{2,3} + y_t^2 + \nu_t^{2,1},\ t = 1,2,\\
z^{-3} &= \max\Big\{\zeta_1^{-3}, \zeta_2^{-3}\Big\}.
\end{aligned}$$

In user 3's tax function, there are no $\mathrm{pr}\beta_t^3(m)$ terms because user 3 is not assigned to any other users for providing β proxies.

To figure out how the proxies n work, here, we focus on the seventh constraint, and see how the corresponding constraint term is evaluated in user 3's tax function. The reason why other ns and νs work is similar. For user 3, the constraint term is

$$\text{con}^{3,7}(m) = (q^{3,7} - q^{-3,7})^2 + q^{3,7} \underbrace{(2 - y_1^2 - y_2^2 - \beta_1^{2,3} - \beta_2^{2,3} - n^{2,1,7})}_{\text{Slackness part}}.$$

In the centralized mechanism, the slackness part turns out to be $1 - \sum_{t=1}^{2}\sum_{i=1}^{3} x_t^{i*}$ at NE. What we want to show is that, with the distributed mechanism, the same outcome can be realized at NE. Similar to the centralized mechanism, $y^i = x^{i*}$ at NE, so $y_1^2 = x_1^{2*}, y_2^2 = x_2^{2*}$. By Lemma 4, $\beta_1^{2,3} + \beta_2^{2,3} = y_1^3 + y_2^3 = x_1^{3*} + x_2^{3*}$, so it remains to show that $n^{2,1,7} = x_1^{1*} + x_2^{1*}$.

Let us trace how the $n^{2,1,7}$ is generated at NE. From (21),

$$n^{2,1,7} = a_1^{1,7} y_1^1 + a_2^{1,7} y_2^1 + \sum_{h \in \mathcal{N}(1)\setminus\{2\}} n^{1,h,7}.$$

Notice that $\mathcal{N}(1) = \{2\}$, so $\mathcal{N}(1)\setminus\{2\}$ is empty. As a result, at NE, $n^{2,1,7} = y_1^1 + y_2^1 = x_1^{1*} + x_2^{1*}$.

6.4. The Learning Algorithm

Before the algorithm is implemented, one might want to check whether the problem setting satisfies Assumptions 4 and 5.

First, we can check Assumption 5. Suppose for the specific environment that we have $\underline{r}_t^i = i \cdot t/9$ and $\bar{r}_t^i = i \cdot t$. Then, for the first condition in Assumption 5, since each x_t^i has a lower bound -1, we have

$$\dot{v}_t^i(x_t^i) = \frac{i \cdot t}{x_t^i + 2} \leq \frac{i \cdot t}{-1 + 2} = \bar{r}_t^i.$$

Additioanlly, every x_t^i is upper bounded by 7 because, from the seventh constraint, we have

$$x_t^i \leq 2 - \sum_{(i',t') \neq (i,t)} x_{t'}^{i'} \leq 2 - 5 \cdot (-1) = 7,$$

and thus

$$\dot{v}_t^i(x_t^i) = \frac{i \cdot t}{x_t^i + 2} \geq \frac{i \cdot t}{7 + 2} = i \cdot t/9.$$

For the second condition in Assumption 5, for all $p \in [i \cdot t/9, i \cdot t]$, we have

$$\dot{v}_t^i(x_t^i) = p \Leftrightarrow x_t^i = \frac{i \cdot t}{p} - 2,$$

so Assumption 5 is verified.

With the $\underline{r}_t^i$s and $\bar{r}_t^i$s chosen above, a dual feasible set $\mathcal{P}$ is constructed. Within this price set $\mathcal{P}$, Algorithm 1 evaluates the function $(\dot{v}_t^i)^{-1}(\cdot)$ only in the interval $[i \cdot t/9, i \cdot t]$. Consequently, in running Algorithm 1, we only need to define $v_t^i(\cdot)$ on the interval $[-1, 7]$.

Regarding Assumption 4, we need to show that $v_t^i(\cdot)$ is strongly concave on $[-1, 7]$. Since one can verify that the function $-it\ln(2 + x) - ax^2/2$ is convex on $[-1, 7]$ for $0 \leq a \leq it/81$, every $v_t^i(\cdot)$ is strong concave, and thus, Assumption 4 holds (this is based on the fact that f is strongly convex with parameter δ iff $g(x) = f(x) - \frac{\delta}{2}\|x\|^2$ is convex.).

To choose an appropriate step size α for the algorithm, we need to investigate further the parameter δ. In our environment, the sum of utility functions $f(x) = -\sum_{t=1}^{2}\sum_{i=1}^{3} v_t^i(x_t^i)$ is strongly concave on $[-1, 7]^6$ with parameter $\delta = 18/81$ because each component v_t^i of f is a strongly concave function with parameter $i \cdot t/81$ and the parameter δ is additive: $\sum_{t=1}^{2}\sum_{i=1}^{3} it/81 = 18/81$. By calculation, $\|\tilde{A}\| \approx 3.1623$, so one possible step size can

be $\alpha = 0.1 < 2 \times \delta / \|\tilde{A}\|$. According to Algorithm 1, the updates required are (define $\eta(i,t) = 2(i-1)+t$ for convenience):

$$\tilde{q}^{i,\eta(j,t)}(k+1) = q^{i,\eta(j,t)}(k) - \alpha(1 + y_t^j(k)),\ j = 1,2,3,\ t = 1,2,$$
$$\tilde{q}^{i,7} = q^{i,7} - \alpha(2 - \sum_{j=1}^{3}\sum_{t=1}^{2} y_t^j(k)),$$
$$\tilde{s}_t^i(k+1) = s_t^i(k) + \alpha \sum_{j=1}^{3} y_t^j,\ t = 1,2,$$
$$(q^i(k+1), s^i(k+1)) = \mathbf{Proj}_{\mathcal{P}}(\tilde{q}^i(k+1), \tilde{s}^i(k+1)),$$
$$y_t^i(k+1) = \frac{i \cdot t}{p_t - q^{i,\eta(i,t)}(k+1) + q^7(k+1) + s_t^i(k+1)} - 2.$$

To verify the convergence of the learning algorithm, we run it with the initial price set to $(q(0), s(0)) = \mathbf{Proj}_{\mathcal{P}}(0_{9\times 1})$. After $K = 100$ iterations, we observe the convergence for both the suggested prices q, s and the corresponding announced demands y. Figure 3 shows the process of convergence and verifies that the convergence rate is exponential, as expected.

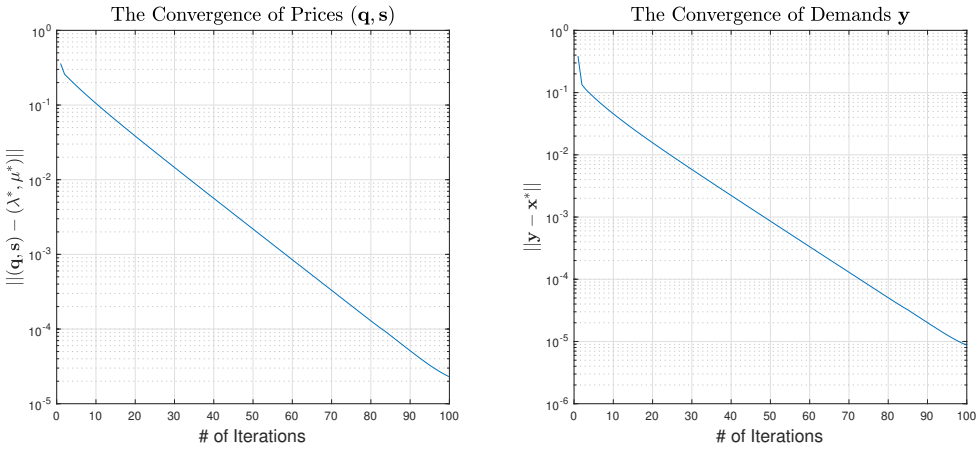

Figure 3. The convergence of the learning algorithm.

7. Conclusions

Motivated by the work of mechanism design for NUM problems, we proposed a new class of (indirect) mechanisms, with application in demand management in energy communities. The proposed mechanisms possess desirable properties including full implementation, individual rationality, and budget balance and can be easily generalized to different environments with peak shaving and convex constraints. We showed how the original "centralized" mechanism can be modified in a systematic way to account for environments with communication constraints. This modification leads to a new type of mechanisms that we call "decentralized" mechanisms and can be thought of as the analog to decentralized optimization (developed for optimization problems with non-strategic agents) for environments with strategic users. Finally, motivated by the need for practical deployment of these mechanisms, we introduced a PGD-based learning algorithm for users to learn the NE of the mechanism-induced game.

Possible future research directions include seeking more efficient learning algorithms for the distributed mechanism as well as co-design of a (distributed) mechanism and characterization of the class of convergent algorithms for this design.

Author Contributions: X.W.—conceptualization, formal analysis, investigation, methodology, software, visualization, original draft, review and editing; A.A.—conceptualization, formal analysis,

funding acquisition, investigation, methodology, project administration, resources, supervision, review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from National Science Foundation.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Equivalence of Centralized Optimization Problem (2) and Original Problem (3a)–(3c)

We first prove this sufficiency by showing that we can always derive the optimal solution of (2) from the optimal solution of newly constructed (3a)–(3c). Suppose that the optimal solution of (3a)–(3c) is (x^*, w^*). We claim that x^* is the optimal solution of the original problem (2). First, the feasibility of x^* in (2) is ensured by (3b) in the newly constructed problem.

Now, we check the optimality. Suppose that x^* is not the optimal for (2), and instead, x' is the optimal. In the new problem, construct

$$\tilde{x} = x', \quad \tilde{w} = \max_{1\le t\le T} \sum_{i=1}^{N} x_t'^i,$$

then it is easy to verify that $(\tilde{x}, \tilde{w})$ is feasible for the new optimization. Notice that

$$\begin{aligned}
&\sum_{i=1}^{N} v^i(\tilde{x}^i) - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} \tilde{x}_t^i - p_0 \tilde{w} \\
=&\sum_{i=1}^{N} v^i(x'^i) - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} x_t'^i - p_0 \max_{1\le t\le T} \sum_{i=1}^{N} x_t'^i \\
>&\sum_{i=1}^{N} v^i(x^{*i}) - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} x_t^{*i} - p_0 \max_{1\le t\le T} \sum_{i=1}^{N} x_t^{*i} \\
\ge&\sum_{i=1}^{N} v^i(x^{*i}) - \sum_{t=1}^{T} p_t \sum_{i=1}^{N} x_t^{*i} - p_0 w^*.
\end{aligned} \tag{A1}$$

The first inequality follows the optimality of x' in the original optimization (2); the second inequality comes from the constraint (3c) in the new optimization. By this inequality chain, we find a $(\tilde{x}, \tilde{w})$ with a better objective function value in (3a)–(3c) than (x^*, w^*), which contradicts the assumption that (x^*, w^*) is an optimal solution of (3a)–(3c).

Therefore, by contradiction, we shows that, if (x^*, w^*) is optimal solution of (3a)–(3c), x^* must be the optimal solution for the original optimization (2).

For the other direction, we need to show if x' is optimal solution of (2), then we are able to construct an optimal solution of (3a)–(3c) based on x'. We construct $\tilde{x} = x'$ and $\tilde{w} = \max_{1\le t\le T} \sum_{i=1}^{N} x_t'^i$ and argue that this $(\tilde{x}, \tilde{w})$ is the optimal for (3a)–(3c). Assume that (x^*, w^*) is the optimal for (3a)–(3c); then, we still obtain the same inequality chain as (A1) (except that, for the second line, there should be a "greater than or equal" sign instead), and the equality and inequalities hold for the same reasons, as stated above. This shows that $(\tilde{x}, \tilde{w})$ has the same objective value as the optimal solution of (3a)–(3c), and thereforem $(\tilde{x}, \tilde{w})$ constructed from x' of the original problem is also the optimal for the new problem.

Appendix B. Proof of Lemma 2

Proof. At NE m^*, for the constraint l in $\mathcal{L}$, consider the message components $q^{i,l}$ for each user i. In user i's tax function, denote the part relative to $q^{i,l}$ by $\hat{t}_q^{i,l}$. We have

$$\begin{aligned}\hat{t}_q^{i,l}(m^i, m^{-i*}) &= (q^{i,l} - q^{-i,l*})^2 + q^{i,l}\left(b^l - \sum_{j\neq i} a^{j,l} y^{j*} - a^{i,l}\beta^{i-1}\right)\\ &= (q^{i,l} - q^{-i,l*})^2 + q^{i,l}\underbrace{\left(b^l - \sum_{j} a^{j,l} y^{j*}\right)}_{\text{denoted by } e^l(y^*)}\\ &\quad (\beta^{i-1} = y^i)\end{aligned}$$

by Lemma 1.

For any user i, there is no unilateral profitable deviation on m^{i*}. Hence, if we fix m^{-i*} and all of the message components of m^{i*} except $q^{i,l}$, it is a necessary condition that user i cannot find a better response than $q^{i,l*}$.

Consider the best response of $q^{i,l}$ in different cases of $e^l(y^*)$.

Case 1. $e^l(y^*) > 0$, i.e., the constraint l is inactive at NE. Note that $\hat{t}_q^{i,l}$ is a quadratic function of $q^{i,l}$ of the following form

$$\hat{t}_q^{i,l} = (q^{i,l})^2 - (2q^{-i,l*} - e^l(y^*))q^{i,l} + (q^{-i,l*})^2.$$

Without considering the nonnegative restriction, the best choice should be $q^{-i,l*} - e^l(y^*)/2$. Since $q^{i,l} \geq 0$, the best choice for $q^{i,l}$ would be $(q^{-i,l*} - e^l(y^*)/2)^+$ (here, $(\cdot)^+ = \max\{\cdot, 0\}$), which is unique with fixed m^*_{-i}.

Therefore,

$$q^{i,l*} = (q^{-i,l*} - e^l(y^*)/2)^+.$$

Observe that $(q^{-i,l*} - e^l(y^*)/2)^+ \leq (q^{-i,l*})^+ = q^{-i,l*}$. Equality holds only if $q^{-i,l*} \leq e^l(y^*)/2$ and $q^{-i,l*} = 0$. Thus, for all i, $q^{i,l*} \leq q^{-i,l*}$, equality holds only if $q^{i,l*} = 0$ and $q^{-i,l*} = 0$. In other words, if for one user i we have $q^{i,l*} = q^{-i,l*}$, then all of the $q^{i,l*} = 0$.

Notice that $q^{i,l*} < q^{-i,l*}$ implies that $q^{i,l}$ is smaller than one of the $q^{j,l}$ among user $j \neq i$, which means that $q^{i,l}$ is not the largest. Assume that $q^{i,l*} < q^{-i,l*}$ for all i; then, no $q^{i,l}$ can be the largest among $\{q^{i,l}\}_{i\in\mathcal{N}}$, but we also know that $\{q^{i,l}\}_{i\in\mathcal{N}}$ is a finite set, and therefore, it must have a maximum. Here comes the contradiction. As a result, there must exist at least one i such that $q^{i,l*} = q^{-i,l*}$, which implies that all of the $q^{i,l*} = 0$.

Case 2. $e^l(y^*) = 0$, i.e., the constraint l is active at NE. In this case, $\hat{t}_q^{i,l} = (q^{i,l} - q^{-i,l*})^2$. It is clear that every user's best response is to make their own price align with the average of others.

Notice that, if $q^{i,l*} = q^{-i,l*}$, then $q^{-i,l}$ is equal to the average of all q^l. Consequently, $q^{i,l*} = q^{j,l*}$ for all $i, j \in \mathcal{N}$.

Case 3. $e^l(y^*) < 0$, i.e., the constraint l is violated at NE. In this case,

$$\hat{t}_q^{i,l} = (q^{i,l})^2 - \underbrace{(2q^{-i,l*} - e^l(y^*))}_{>0} q^{i,l} + (q^{-i,l*})^2,$$

which leads to a condition for all user i as

$$q^{i,l*} = q^{-i,l*} + \underbrace{(-e^l(y^*)/2)}_{>0} > q^{-i,l*}.$$

In a finite set, if one number is strictly larger than the average of the others, it means that it is not the smallest number in the set. If this condition is true for all user i, it means

there is no smallest number among the set, which is impossible. Therefore, Case 3 does not occur at NE.

In summary, at NE, we always have $e^l(y^*) \geq 0$, and $q^{i,l}$s are equal. Moreover, $q^{i,l*}e^l(y^*) = 0$. These prove the primal feasibility, equal prices, and complementary slackness on prices q in the Lemma 2.

Now, for time t, consider the message component s_t^i for each user i. In user i's tax function, denote the part relative to s_t^i by $\hat{t}_s^{i,l}$. We have

$$\begin{aligned}\hat{t}_s^{i,t}(m^i, m^{-i*}) &= (s_t^i - s_t^{-i*})^2 + s_t^i\left(z^{-i} - \sum_{j\neq i} y_t^{j*} - \beta_t^{i-1*}\right)\\ &= (s_t^i - s_t^{-i*})^2 + s_t^i \underbrace{\left(z^* - \sum_j y_t^{j*}\right)}_{\text{denoted by } g_t(y^*)},\end{aligned}$$

where $z^* = \max_t \sum_j y_t^{j*}$.

Different from the proof of previous part, here, we only need to consider two cases of $g_t(y^*)$ because, by definition of z, $g_t(y^*)$ is always nonnegative. Another thing we can observe is that there exists at least one t such that $g_t(y^*) = 0$, i.e., at least one time t is the time for the peak demand. Define the set $\tilde{\mathcal{T}}$ as the time set containing all time t with peak demand.

Check the best response of s_t separately. For all $t' \notin \tilde{\mathcal{T}}$, $g_{t'}(y^*) > 0$, then following the similar steps shown above, we know $s_{t'}^{i*} = 0$ for all i. For those $t \in \tilde{\mathcal{T}}$, we have already had $g_t(y^*) = 0$. For those t, the best response is $s_t^{i*} = s_t^{-i*}$, which is true for every user i. As a result, we also have the equal prices and complementary slackness for s_t^i.

Check (10). For $t \in \tilde{\mathcal{T}}$, $g_t(y^*) = 0$, so (10) holds.

For $t \notin \tilde{\mathcal{T}}$, $g_t(y^*) > 0$, so

$$t \notin \arg\max_{\tilde{t}} \sum_{j\neq i} y_{\tilde{t}}^j + \beta_{\tilde{t}}^{i-1}.$$

Adiitionally, we know that such a $t \notin \tilde{\mathcal{T}}$ has $s_t = 0$. Therefore, for $t \notin \tilde{\mathcal{T}}$, for either branch in the definition of radial pricing $\mathcal{RP}$, $\mathcal{RP}_t^i(s, \zeta^{-i}) = 0$. Hence, (10) holds for $t = 1, \ldots, T$. □

Appendix C. Proof of Lemma 3

Proof. At NE, for user i, the $u^i(m^i, m^{-i})$ can be treated as a function of m^i, with m^{-i} fixed. By the assumption of the existence of NE, $u^i(m^i, m^{-i})$ must have a global maximizer with respect to m^i. Given m^{-i}, all of the auxiliary variables and functions only determined by m^{-i} are constants here, and one can check that the other terms in (5) and (7) are differentiable. Necessary conditions for the global maximizer are

$$\frac{\partial u^i}{\partial y_t^i} = \dot{v}_t^i(\hat{x}_t^i(m)) - \left(p_t + \mathcal{RP}_t^i(s, \zeta^{-i}) + \sum_{l\in\mathcal{L}_i} a_t^{i,l} q^l\right) = 0.$$

Therefore,

$$\dot{v}_t^i(\hat{x}_t^i(m)) = p_t + \mathcal{RP}_t^i(s, \zeta^{-i}) + \sum_{l\in\mathcal{L}_i} a_t^{i,l} q^l,$$

which is exactly the equation (11).

Equation (12) follows directly from the definition of $\mathcal{RP}$ operator. □

Appendix D. Proof of Theorem 2

Proof. By assumption, the centralized problem is a convex optimization problem with a non-empty feasible set, so there must exist an optimal solution $\{x^*, w^*\}$ and corresponding Lagrange multipliers λ^{l*}, μ_t^* which satisfy KKT conditions (4a)–(4g).

Consider the message profile m^* consisting of

$$\begin{aligned} y_t^i &= x_t^{i*},\ t = 1, \ldots, T,\ \forall i \in \mathcal{N}, \\ q^{i,l} &= \lambda^{l*},\ l \in \mathcal{L}_i,\ \forall i \in \mathcal{N}, \\ s_t^i &= \mu_t^*,\ t = 1, \ldots, T,\ \forall i \in \mathcal{N} \\ \beta_t^i &= x_t^{i+1*},\ t = 1, \ldots, T,\ \forall i \in \mathcal{N}. \end{aligned}$$

If for arbitrary user i, no profitable unilateral deviations exist, i.e., there does not exist an $\tilde{m} = (\tilde{m}_i, m_{-i}^*)$ such that $u_i(\tilde{m}) > u_i(m^*)$, then m^* is a NE of the game $\mathcal{G}$.

We can focus on $u_i(m)$ of user i to see whether they have a profitable deviation given m_{-i}^*. For user i, we have

$$\begin{aligned} q^{-i,l} &= \lambda^{l*},\ \forall l \in \mathcal{L} \\ s_t^{-i} &= \mu_t^*,\ t = 1, \ldots, T, \\ \mathcal{RP}_t^i(s^{-i}, y^{-i}, \beta^{i-1}) &= \mu_t^*,\ t = 1, \ldots, T, \\ z^{-i} &= w^* = \max_t \left(\textstyle\sum_{j \in \mathcal{N}} x_t^{j*} \right). \end{aligned}$$

Therefore, in the interest of user i, they want to maximize the following

$$\begin{aligned} u^i(m^i, m^{-i*}) = &\sum_{t=1}^{T} \underbrace{\left(v_t^i(y_t^i) - (p_t + \mu_t^*) y_t^i - \sum_{l \in \mathcal{L}_i} \lambda^{l*} a_t^{i,l} y_t^i \right)}_{\text{Function of } y_t^i} \\ &- \sum_{l \in \mathcal{L}} \underbrace{\left((q^{i,l} - \lambda^{l*})^2 + q^{i,l} \left(b^l - \sum_j \sum_{t=1}^{T} a_t^{j,l} x_t^{j*} \right) \right)}_{\text{Function of } q^{i,l}} \\ &- \sum_{t=1}^{T} \underbrace{\left((s_t^i - \mu_t^*)^2 + s_t^i \left(w^* - \sum_j x_t^{j*} \right) \right)}_{\text{Function of } s_t^i} \\ &- \sum_{t=1}^{T} \underbrace{(\beta_t^i - x_t^{i+1*})^2}_{\text{Function of } \beta_t^i}. \end{aligned} \tag{A2}$$

The last term of (A2) is the only term related to β^i, which is a quadratic terms. As a strategic agent, it is clear that user i does not deviate from $\beta^i = x^{i+1*}$; otherwise, they pay for the penalty.

The second and third terms of (A2) are quite similar: they both consist of a quadratic term and a term for complementary slackness. For the second term, let us consider constraint l. If l is active in the optimal solution, the complementary slackness term goes to 0. To avoid extra payment, user i does not deviate $q^{i,l}$ from the price suggested by optimal

solution λ^{l*}. If l is inactive, the price λ^{l*} suggested by optimal solution is 0. Then, the penalty of constraint l for user i is

$$(q^{i,l})^2 + q^{i,l} \underbrace{\left(b^l - \sum_j \sum_{t=1}^{T} a_t^{j,l} x_t^{j*} \right)}_{>0},$$

where user i can only select a nonnegative price $q^{i,l}$. There are no better choices better than choosing $q^{i,l} = 0 = \lambda^{l*}$. A similar analysis works for the third term of (A2). As a result, there are no unilateral profitable deviations on $q^{i,l}$ for all l and s_t^i for all t.

Now, we denote the terms in the parentheses in the first part of (A2) by $f_t^i(y_t^i)$. Since these four terms are disjoint in the aspect of inputted variables, $u^i(m^i, m^{-i*})$ achieves its maximum if and only if every $f_t^i(y_t^i)$ achieves its maximum, and the other three terms equal their minimum. For the first part, due to the strict concavity of $v_t^i(\cdot)$, the second-order derivative of $f_t^i(y_t^i)$ for each t is negative, which indicates that $f_t^i(y_t^i)$ is strictly concave as well. We can find the maxima of $f_t^i(y_t^i)$ by the first-order condition:

$$\frac{\mathrm{d}f_t^i}{\mathrm{d}y_t^i}(y_t^{i*}) = \dot{v}_t^i(y_t^i) - (p_t + \mu_t^*) - \sum_{l \in \mathcal{L}_i} \lambda^{l*} a_t^{i,l} = 0. \tag{A3}$$

By (4g) in the KKT conditions, we know that the only y_t^i that makes (A3) hold is $y_t^i = x_t^{i*}$ for all t. The reason is that, by the strict concavity assumption of utility function $v_t^i(\cdot)$, the first-order derivative of v_t^i strictly decreases, and therefore, for one aggregated price, there is at most one demand value x that makes $\dot{v}_t^i(x)$ equal that price.

Therefore, for any agent i, if others send messages m^{-i*}, the only best response of agent i is to announce m^{i*}. Under this circumstance, sending messages other than m^{i*} does not increase agent i's payoff $u^i(m)$. Consequently, m^* is an NE of the induced game $\mathcal{G}$. □

Appendix E. Proof of Theorem 3

Proof. For any user i, if they choose to participate with other users, when everyone anticipates the NE, user i's payoff is of the form (13) if they only consider modifying y^i and keeps other components unchanged. Thus, user i faces the following optimization problem:

$$y^i = \arg\max_{y^i \in \mathbb{R}^T} \quad v^i(y^i) - \sum_{t=1}^{T} \Big(p_t + \mathcal{RP}_t^i(s, \zeta^{-i}) \Big) y_t^i - \sum_{l \in \mathcal{L}_i} q^{-i,l} \sum_{t=1}^{T} a_t^{i,l} y_t^i.$$

By the definition of NE, y^{i*} is one of the best solutions, which yields a payoff $u^i(m^*)$. User i can also choose $\tilde{y}^i = 0$. Denote the corresponding message by $\tilde{m}_i$. Then, the payoff value becomes $u^i(\tilde{m}^i, m^{-i*}) = v^i(0)$, which coincides with the payoff for not participating. Since m^{i*} is the best response to m^{-i*}, we have $u^i(m^*) \geq u^i(\tilde{m}^i, m^{-i*}) = v^i(0)$. In other words, if every one anticipates the NE as the outcome, to participate is at least not worse than not participating. □

Appendix F. Proof of Theorem 4

Proof. Suppose that the optimal solution for the original problem given by NE is (x^*, λ^*, μ^*); then, the tax for user i is

$$\hat{t}^i(m^*) - J(\hat{x}_t^i(m^*)) = \sum_{t=1}^{T} (p_t + \mu_t^*) x_t^{i*} + \sum_{l \in \mathcal{L}_i} \lambda^{l*} \sum_{t=1}^{T} a_t^{i,l} x_t^{i*} - J(x^{i*}).$$

The total amount of tax is

$$\begin{aligned}
&\sum_{i\in\mathcal{N}} \hat{t}^i(m^*) - J(x^{i*}) \\
&= \sum_{i\in\mathcal{N}} \sum_{t=1}^{T} (p_t + \mu_t^*) x_t^{i*} + \sum_{i\in\mathcal{N}} \sum_{l\in\mathcal{L}_i} \lambda^{l*} \sum_{t=1}^{T} a_t^{i,l} x_t^{i*} - J(x^{i*}) \\
&= \sum_{t=1}^{T} \left(p_t \sum_{i\in\mathcal{N}} x_t^{i*} + \mu_t^* \sum_{i\in\mathcal{N}} x_t^{i*} \right) + \sum_{l\in\mathcal{L}} \lambda^{l*} \sum_{t=1}^{T} \sum_{i\in\mathcal{N}} a_t^{i,l} x_t^{i*} - J(x^{i*}) \\
&= \sum_{l\in\mathcal{L}} \lambda^{l*} \sum_{t=1}^{T} \sum_{i\in\mathcal{N}} a_t^{i,l} x_t^{i*}.
\end{aligned}$$

For each constraint l, by the complementary slackness, we have

$$\lambda^{l*} \left(b^l - \sum_{t=1}^{T} \sum_{i\in\mathcal{N}} a_t^{i,l} x_t^{i*} \right) = 0.$$

Therefore,

$$\sum_{i\in\mathcal{N}} \hat{t}^i(m^*) - J(x^{i*}) = \sum_{l\in\mathcal{L}} \lambda^{l*} b^l \geq 0,$$

which shows that, at NE, the planner's payoff is nonnegative.

Furthermore, in order to save unnecessary expenses on the planner, the energy community can adopt the mechanism with the following tax function $\tilde{t}^i(m)$ instead

$$\tilde{t}^i(m) = \hat{t}^i(m) - \sum_{l\in\mathcal{L}} q^{-i,l} b^l / N.$$

Note that user i has no control on the additional term because no components of m^i are in that term, and thus, the additional term does not change NE. Since the prices are equal at NE, the planner actually gives $\sum_{l\in\mathcal{L}} \lambda^{l*} b^l$ back to the users. Hence,

$$\sum_{i\in\mathcal{N}} \tilde{t}^i(m^*) - J(x^{i*}) = 0,$$

As a side comment, the choice of $\tilde{t}^i(m)$ is not unique. Any adjustment works here as long as it does not depend on m^i for each $t^i(\cdot)$ and sums up to $\sum_{l\in\mathcal{L}} \lambda^{l*} b^l$ at NE. □

Appendix G. Proof of Lemma 5

Proof. Here, we provide a non-rigorous proof of (24). The proof of (23) is quite similar. For a detailed version of the proof, we refer the interested readers to 7.1, Chapter 4, of [72].

Before we show the proof of this part, for the sake of convenience, we define $n(i,k)$ as the nearest user among the neighbors of user i and user i itself to user k. $n(i,k)$ is well-defined because one can show that $n(i,k) = j$ provides a partition for all the users.

Equation (24) can be shown by applying (22) iteratively. Recall that the message exchange network is assumed to be a undirected acyclic graph (i.e., a tree). First, consider the user j on the leaves (the nodes with only one degree). Suppose the neighbor of user j is i, then $\mathcal{N}(j) = \{i\}$. By (22), we have $\nu_t^{i,j} = y_t^j$. Since no k satisfies $n(i,k) = j$ other than j themselves, (24) holds for $\nu_t^{i,j}$, where j is a leaf node.

For more general cases, to compute $\nu_t^{i,j}$, it is safe to only consider the subgraph $\mathcal{GR}_i$ that contains only node i and node k such that $n(i,k) = j$. When applying (22), it is impossible to have node $l \in \mathcal{GR}_i^C$ involved because, if it occurs when expanding "ν" term for some j', l is a neighbor of j'. We know that there is a route from i to j', say route iLj'. Since $l \in \mathcal{GR}_i^C$, $n(i,l) \neq j$, there exists a route L' that does not involve any node in the branch starting from node j, such that $lL'i$, which results in a loop $lL'iLj'l$.

Then, by using (22) iteratively, we can see that (1) every node in $\mathcal{GR}_i$ is visited at least once and gives a corresponding demand "y"; (2). each y_t^j is given only once (except root i, which does not give y_t^i in this procedure); and (3) when it proceeds to the leaf nodes, the iteration terminates because there are no more "v" terms to expand. Hence, $v_t^{i,j} = \sum_{h \in \mathcal{GR}_i} y_t^h$, and we can easily verify that $\mathcal{GR}_i \backslash \{i\}$ is nothing but $\{h : n(i,h) = j\}$. □

Appendix H. Convergence of the Learning Algorithm for Centralized Mechanism

The convergence of the proposed learning algorithm can be shown in the three steps mentioned in Section 5. THe first step shows the connection between m^* and x^*, λ^*, μ^* of the optimal solution for the original optimization, which has already been clarified in Section 5. As a result, learning NE is equivalent to learning the optimal solution of the original optimization problem. For the second step, as a convex optimization problem with a non-empty feasible set defined by linear inequalities, Slater's condition is easy to check. Therefore, we have a strong duality in this problem, which means we can obtain the optimal solution of the original problem as long as we solve the dual problem. The last step is to identify the dual problem and to find a convergent algorithm for it. This part of appendix explains how to pin down the dual function and the dual feasible set, and shows the convergence of PGD algorithm on this dual problem.

Before we identify the dual function of the original problem, for the sake of convenience, in constraint (3c) of the original problem, move w to the left-hand side and rewrite (3b) and (3c) into one matrix form

$$\tilde{A}x + \tilde{1}w \leq \tilde{b},$$

where $\tilde{A}$ is defined in (27), and

$$\tilde{1} = \begin{pmatrix} 0_L \\ -1_T \end{pmatrix}, \ \tilde{b} = \begin{pmatrix} b \\ 0_T \end{pmatrix}.$$

Suppose that $f(x) = \sum_i \sum_t (v_t^i(x_t^i) - p_t x_t^i)$; then, the objective function can be written as $f(x) - p_0 w$. Observe that, by Assumption 4, $(v_t^i(x_t^i) - p_t x_t^i)$s are also strongly concave without cross terms. Sequently, by the definition of strong concavity one can show directly that, as the sum of these strongly concave functions, $f(x)$ is strongly concave as well. Let $h(x) = -f(x)$; then, $h(x)$ is strongly convex with parameter δ'. Denote by $h^*(\cdot)$ the conjugate function of $h(x)$.

With these notations in mind, the dual function of the original problem is

$$\begin{aligned} D(\tilde{\lambda}) =& \sup_{x,w} \left\{ f(x) - p_0 w - \tilde{\lambda}^\mathsf{T}(\tilde{A}x + \tilde{1}w - \tilde{b}) \right\} \\ =& b^\mathsf{T}\lambda + \sup_x \left\{ (-\tilde{A}^\mathsf{T}\tilde{\lambda})^\mathsf{T} x - h(x) \right\} + \sup_w \left\{ 1_T^\mathsf{T}\mu w - p_0 w \right\} \\ =& b^\mathsf{T}\lambda + h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}), \end{aligned}$$

Here, we should be cautious about the domain of $D(\tilde{\lambda})$. In the second line, $\sup_w \{1_T^\mathsf{T}\mu w - p_0 w\}$ is only defined when the coefficient $1_T^\mathsf{T}\mu - p_0 = 0$, i.e., $\sum_t \mu_t = p_0$. Therefore, we get the following dual problem:

$$\underset{\lambda,\mu}{\text{minimize}} \quad b^\mathsf{T}\lambda + h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}) \tag{A4}$$

$$\text{subject to} \quad \sum_t \mu_t = p_0, \tag{A5}$$

$$\lambda \geq 0, \ \mu \geq 0. \tag{A6}$$

Now, we derive the dual problem for the original optimization. To find the optimal solution, one direct thought is to use a projected gradient descent. Luckily, we have the following theorem, which ensures the convergence of PGD algorithm.

Theorem A1. *For a minimization problem on a closed and convex feasible set $\mathcal{X}$ with objective function $f(x)$, suppose that $\mathcal{X}^*$ is the set of optimal solutions. If f is convex and β-smooth on $\mathcal{X}$, by using PGD with step size $\alpha < 2/\beta$, there exists $x^* \in \mathcal{X}^*$, such that*

$$\lim_{k\to\infty} x(k) = x^*.$$

Proof. The proof can be found in ([29] Theorem 1, Section 7.2). □

Theorem A1 indicates that, if the dual problem satisfies certain conditions, the solution converges to the set of optimal solutions. Although it is not clear whether the dual problem has a unique solution, by strong duality and the uniqueness of the solution to the primal problem, no matter which dual optimal solution is achieved, the corresponding primal solution can only be the unique optimal one and results in the same outcome.

Now, for the dual problem, we need to check the conditions required by Theorem A1. First, check the objective function. It is clear that any conjugate functions are convex, so $h^*(\cdot)$ is convex and, consequently, $h^*(-\tilde{A}\tilde{\lambda})$ is convex in $\tilde{\lambda}$ as a composition of convex function and affine function. Thus, the objective function is convex. Since $h(x)$ is strongly convex with parameter δ', by the result mentioned in [73], the δ'-strong convexity of $h(\cdot)$ implies that its conjugate $h^*(\cdot)$ is $1/\delta'$-smooth. Then, we have

$$||\nabla h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}^1) - \nabla h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}^2)|| \le 1/\delta' \cdot || - \tilde{A}^T(\tilde{\lambda}^1 - \tilde{\lambda}^2)|| \le (\|\tilde{A}\|/\delta') \cdot ||\tilde{\lambda}^1 - \tilde{\lambda}^2||,$$

which indicates that the objective function is β-smooth with $\beta = \|\tilde{A}\|/\delta'$. However, we are not sure whether the objective function is well-defined on the whole feasible set. Fortunately, by Assumption 5, we know that the optimal price vector $\tilde{\lambda}^*$ lies in $\mathcal{P}$ and we can verify that $\mathcal{P}$ is a subset of the feasible set generated by (A5) and (A6). Therefore, by solving the following optimization problem, we can obtain the same optimal solution, and Theorem A1 is applicable here.

$$\underset{\tilde{\lambda}\in\mathcal{P}}{\text{minimize}} \quad b^\mathsf{T}\lambda + h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}). \tag{A7}$$

By applying PGD to (A7) with step size $\alpha \le 2/\beta = 2\delta'/\|\tilde{A}\|$, the update rules are as follows:

$$\hat{\lambda}^l(k+1) = \lambda^l(k) - \alpha\Big(b^l + [-\tilde{A}\nabla h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}(k))]_l\Big), \tag{A8}$$

$$\hat{\mu}_t(k+1) = \mu(k) - \alpha[-\tilde{A}\nabla h^*(-\tilde{A}^\mathsf{T}\tilde{\lambda}(k))]_{L+t}, \tag{A9}$$

$$(\lambda(k+1), \mu(k+1)) = \mathbf{Proj}_{\mathcal{P}}\big(\hat{\lambda}(k+1), \hat{\mu}(k+1)\big). \tag{A10}$$

where $[\cdot]_j$ represents the jth entry of the inputted vector. To modify these rules into a learning algorithm for a centralized mechanism, by the relation between m^* and the optimal solution of the original problem and the dual, one might want to substitute λ, μ with q^i, s^i for each user i. However, ∇h^* is not tractable for users as they do not know the utilities of the others. Thankfully, users can obtain the values of ∇h^*-related terms by cooperation without revealing their entire utility functions. This method is realized by inquiries for the demands under given prices from each user. A key point for this implementation is to build a connection between ∇h^* and the marginal value function $\dot{v}_t^i$ for each demand x_t^i.

A useful result of subgradient of function f and its conjugate f^* can be used here, which is quoted as Theorem A2.

Theorem A2. *Suppose $f^*(s)$ is the conjugate of $f(x)$, then*

$$x \in \partial f^*(s) \Leftrightarrow s \in \partial f(x).$$

Proof. The proof can be found in [73]. □

Since h is closed (because h is proper convex and continuous) and strictly convex by assumption, h^* is differentiable (see [73]), and therefore, the subgradient of h^* on a fixed p is a singleton. As a result,

$$x = \nabla h^*(-\rho) \Leftrightarrow \rho = -\nabla h(x) = \nabla f(x) \Leftrightarrow x_t^i = (\dot{v}_t^i)^{-1}(p_t + \rho_t^i).$$

The last equivalent sign comes from the fact that

$$[\nabla f(x)]_{(i-1)T+t} = \frac{\mathrm{d}}{\mathrm{d}x_t^i}(v_t^i(x_t^i) - p_t x_t^i) = \dot{v}_t^i(x_t^i) - p_t.$$

Thus, in every iteration of PGD, before using (A8) and (A9), one can first evaluate

$$x_t^i(k) = (\dot{v}_t^i)^{-1}(p_t + [\tilde{A}^\mathsf{T}\lambda(\tilde{k})]_{(i-1)T+t}), \tag{A11}$$

and then, (A8) and (A9) become

$$\hat{\lambda}^l(k+1) = \lambda^l(k) - \alpha\Big(b^l - [\tilde{A}x(k)]_l\Big) = \lambda^l(k) - \alpha\Big(b^l - a^l x(k)\Big), \tag{A12}$$

$$\hat{\mu}_t(k+1) = \mu(k) + \alpha[\tilde{A}x(k)]_{L+t} = \mu(k) + \alpha\sum_j x_t^j(k). \tag{A13}$$

Arranging (A10)–(A13) in an appropriate order, we obtain an algorithm with the same convergent property of the original PGD, and significantly, no ∇h^* is in the algorithm. By substituting λ and μ with q^i and s^i (by making duplications of (A12) and (A13) for each user i) and by substituting x with y, we obtain Algorithm 1. Consequently, Theorem 6 follows directly from the convergence of PGD indicated in Theorem A1.

References

1. Schmidt, D.A.; Shi, C.; Berry, R.A.; Honig, M.L.; Utschick, W. Distributed resource allocation schemes. *IEEE Signal Process. Mag.* **2009**, *26*, 53–63. [CrossRef]
2. Nair, A.S.; Hossen, T.; Campion, M.; Selvaraj, D.F.; Goveas, N.; Kaabouch, N.; Ranganathan, P. Multi-agent systems for resource allocation and scheduling in a smart grid. *Technol. Econ. Smart Grids Sustain. Energy* **2018**, *3*, 1–15. [CrossRef]
3. HamaAli, K.W.; Zeebaree, S.R. Resources allocation for distributed systems: A review. *Int. J. Sci. Bus.* **2021**, *5*, 76–88.
4. Menache, I.; Ozdaglar, A. Network games: Theory, models, and dynamics. *Synth. Lect. Commun. Netw.* **2011**, *4*, 1–159. [CrossRef]
5. Hurwicz, L.; Reiter, S. *Designing Economic Mechanisms*; Cambridge University Press: Cambridge, UK, 2006.
6. Börgers, T.; Krahmer, D. *An Introduction to the Theory of Mechanism Design*; Oxford University Press: Oxford, UK, 2015.
7. United States Department of Energy. *2018 Smart Grid System Report*; United States Department of Energy: Washington, DC, USA, 2018.
8. Garg, D.; Narahari, Y.; Gujar, S. Foundations of mechanism design: A tutorial part 1-key concepts and classical results. *Sadhana* **2008**, *33*, 83. [CrossRef]
9. Garg, D.; Narahari, Y.; Gujar, S. Foundations of mechanism design: A tutorial Part 2-Advanced concepts and results. *Sadhana* **2008**, *33*, 131. [CrossRef]
10. Vickrey, W. Counterspeculation, auctions, and competitive sealed tenders. *J. Financ.* **1961**, *16*, 8–37. [CrossRef]
11. Clarke, E.H. Multipart pricing of public goods. *Public Choice* **1971**, *11*, 17–33. [CrossRef]
12. Groves, T. Incentives in teams. *Econom. J. Econom. Soc.* **1973**, *41*, 617–631. [CrossRef]
13. Kelly, F.P.; Maulloo, A.K.; Tan, D.K. Rate control for communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.* **1998**, *49*, 237–252. [CrossRef]
14. Yang, S.; Hajek, B. Revenue and stability of a mechanism for efficient allocation of a divisible good. *Preprint* **2005**. Available online: https://www.researchgate.net/publication/238308421_Revenue_and_Stability_of_a_Mechanism_for_Ecient_Allocation_of_a_Divisible_Good (accessed on 14 May 2021).
15. Maheswaran, R.; Başar, T. Efficient signal proportional allocation (ESPA) mechanisms: Decentralized social welfare maximization for divisible resources. *IEEE J. Sel. Areas Commun.* **2006**, *24*, 1000–1009. [CrossRef]

16. Sinha, A.; Anastasopoulos, A. Mechanism design for resource allocation in networks with intergroup competition and intragroup sharing. *IEEE Trans. Control Netw. Syst.* **2018**, *5*, 1098–1109. [CrossRef]
17. Rabbat, M.; Nowak, R. Distributed optimization in sensor networks. In Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks, Berkeley, CA, USA, 26–27 April 2004; pp. 20–27.
18. Boyd, S.; Parikh, N.; Chu, E. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*; Now Publishers Inc.: Hanover, MA, USA, 2011.
19. Wei, E.; Ozdaglar, A.; Jadbabaie, A. A distributed Newton method for network utility maximization–I: Algorithm. *IEEE Trans. Autom. Control* **2013**, *58*, 2162–2175. [CrossRef]
20. Alvarado, A.; Scutari, G.; Pang, J.S. A new decomposition method for multiuser DC-programming and its applications. *IEEE Trans. Signal Process.* **2014**, *62*, 2984–2998. [CrossRef]
21. Di Lorenzo, P.; Scutari, G. Distributed nonconvex optimization over time-varying networks. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 4124–4128.
22. Sinha, A.; Anastasopoulos, A. Distributed mechanism design with learning guarantees for private and public goods problems. *IEEE Trans. Autom. Control* **2020**, *65*, 4106–4121. [CrossRef]
23. Heydaribeni, N.; Anastasopoulos, A. Distributed Mechanism Design for Network Resource Allocation Problems. *IEEE Trans. Netw. Sci. Eng.* **2020**, *7*, 621–636. [CrossRef]
24. Brown, G.W. Iterative solution of games by fictitious play. *Act. Anal. Prod. Alloc.* **1951**, *13*, 374–376.
25. Monderer, D.; Shapley, L.S. Fictitious play property for games with identical interests. *J. Econ. Theory* **1996**, *68*, 258–265. [CrossRef]
26. Hofbauer, J.; Sandholm, W.H. On the global convergence of stochastic fictitious play. *Econometrica* **2002**, *70*, 2265–2294. [CrossRef]
27. Milgrom, P.; Roberts, J. Rationalizability, learning, and equilibrium in games with strategic complementarities. *Econom. J. Econom. Soc.* **1990**, *58*, 1255–1277. [CrossRef]
28. Scutari, G.; Palomar, D.P.; Facchinei, F.; Pang, J.S. Monotone games for cognitive radio systems. In *Distributed Decision Making and Control*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 83–112.
29. Polyak, B. *Introduction to Optimization*; Optimization Software Inc.: New York, NY, USA, 1987.
30. Srikant, R.; Ying, L. *Communication Networks: An Optimization, Control, and Stochastic Networks Perspective*; Cambridge University Press: Cambridge, UK, 2013.
31. Huang, Z.; Mitra, S.; Vaidya, N. Differentially private distributed optimization. In Proceedings of the 2015 International Conference on Distributed Computing and Networking, Goa, India, 4–7 January 2015; pp. 1–10.
32. Cortés, J.; Dullerud, G.E.; Han, S.; Le Ny, J.; Mitra, S.; Pappas, G.J. Differential privacy in control and network systems. In Proceedings of the 2016 IEEE 55th Conference on Decision and Control (CDC), Las Vegas, NV, USA, 12–14 December 2016; pp. 4252–4272.
33. Nozari, E.; Tallapragada, P.; Cortés, J. Differentially private distributed convex optimization via objective perturbation. In Proceedings of the American Control Conference (ACC), Boston, MA, USA, 6–8 July 2016; pp. 2061–2066.
34. Han, S.; Topcu, U.; Pappas, G.J. Differentially private distributed constrained optimization. *IEEE Trans. Autom. Control* **2017**, *62*, 50–64. [CrossRef]
35. Groves, T.; Ledyard, J. Optimal allocation of public goods: A solution to the "free rider" problem. *Econom. J. Econom. Soc.* **1977**, *45*, 783–809. [CrossRef]
36. Hurwicz, L. Outcome functions yielding Walrasian and Lindahl allocations at Nash equilibrium points. *Rev. Econ. Stud.* **1979**, *46*, 217–225. [CrossRef]
37. Huang, J.; Berry, R.A.; Honig, M.L. Auction-based spectrum sharing. *Mob. Netw. Appl.* **2006**, *11*, 405–418. [CrossRef]
38. Wang, B.; Wu, Y.; Ji, Z.; Liu, K.R.; Clancy, T.C. Game theoretical mechanism design methods. *IEEE Signal Process. Mag.* **2008**, *25*, 74–84. [CrossRef]
39. Wang, S.; Xu, P.; Xu, X.; Tang, S.; Li, X.; Liu, X. TODA: Truthful online double auction for spectrum allocation in wireless networks. In Proceedings of the 2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN), Singapore, 6–9 April 2010; pp. 1–10.
40. Ghosh, A.; Roth, A. Selling privacy at auction. In Proceedings of the 12th ACM Conference on Electronic Commerce, San Jose, CA, USA, 5–9 June 2011; pp. 199–208.
41. Khalili, M.M.; Naghizadeh, P.; Liu, M. Designing cyber insurance policies: Mitigating moral hazard through security pre-screening. In Proceedings of the International Conference on Game Theory for Networks, Knoxville, TN, USA, 9 May 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 63–73.
42. Pal, R.; Wang, Y.; Li, J.; Liu, M.; Crowcroft, J.; Li, Y.; Tarkoma, S. Data Trading with Competitive Social Platforms: Outcomes are Mostly Privacy Welfare Damaging. *IEEE Trans. Netw. Serv. Manag.* **2020**. [CrossRef]
43. Caron, S.; Kesidis, G. Incentive-based energy consumption scheduling algorithms for the smart grid. In Proceedings of the 2010 First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, USA, 4–6 October 2010; pp. 391–396.
44. Samadi, P.; Mohsenian-Rad, H.; Schober, R.; Wong, V.W. Advanced demand side management for the future smart grid using mechanism design. *IEEE Trans. Smart Grid* **2012**, *3*, 1170–1180. [CrossRef]
45. Muthirayan, D.; Kalathil, D.; Poolla, K.; Varaiya, P. Mechanism design for demand response programs. *IEEE Trans. Smart Grid* **2019**, *11*, 61–73. [CrossRef]

46. Iosifidis, G.; Gao, L.; Huang, J.; Tassiulas, L. An iterative double auction for mobile data offloading. In Proceedings of the 2013 11th International Symposium and Workshops on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), Tsukuba, Japan, 13–17 May 2013; pp. 154–161.
47. Johari, R.; Tsitsiklis, J.N. Efficiency loss in a network resource allocation game. *Math. Oper. Res.* **2004**, *29*, 407–435. [CrossRef]
48. Yang, S.; Hajek, B. VCG-Kelly mechanisms for allocation of divisible goods: Adapting VCG mechanisms to one-dimensional signals. *IEEE J. Sel. Areas Commun.* **2007**, *25*, 1237–1243. [CrossRef]
49. Johari, R.; Tsitsiklis, J.N. Efficiency of scalar-parameterized mechanisms. *Oper. Res.* **2009**, *57*, 823–839. [CrossRef]
50. Farhadi, F.; Golestani, S.J.; Teneketzis, D. A surrogate optimization-based mechanism for resource allocation and routing in networks with strategic agents. *IEEE Trans. Autom. Control* **2018**, *64*, 464–479. [CrossRef]
51. Kakhbod, A.; Teneketzis, D. An Efficient Game Form for Unicast Service Provisioning. *IEEE Trans. Autom. Control* **2012**, *57*, 392–404. [CrossRef]
52. Kakhbod, A.; Teneketzis, D. Correction to "An Efficient Game Form for Unicast Service Provisioning" [Feb 12 392-404]. *IEEE Trans. Autom. Control* **2015**, *60*, 584–585. [CrossRef]
53. Sinha, A.; Anastasopoulos, A. A General Mechanism Design Methodology for Social Utility Maximization with Linear Constraints. *ACM Sigmetrics Perform. Eval. Rev.* **2014**, *42*, 12–15. [CrossRef]
54. aumann, L. Self-Ratings and Peer Review. 2018. Available online: https://www.amse-aixmarseille.fr/sites/default/files/events/JMPLeonieBaumann_0.pdf (accessed on 30 July 2021).
55. Bloch, F.; Olckers, M. Friend-Based Ranking in Practice. Available online: https://arxiv.org/abs/2101.02857 (accessed on 14 May 2021).
56. Chen, Y. A family of supermodular NAvailable online: ash mechanisms implementing Lindahl allocations. *Econ. Theory* **2002**, *19*, 773–790. [CrossRef]
57. Healy, P.J.; Mathevet, L. Designing stable mechanisms for economic environments. *Theor. Econ.* **2012**, *7*, 609–661. [CrossRef]
58. Scutari, G.; Facchinei, F.; Pang, J.S.; Palomar, D.P. Real and complex monotone communication games. *IEEE Trans. Inf. Theory* **2014**, *60*, 4197–4231. [CrossRef]
59. Gharesifard, B.; Cortés, J. Distributed convergence to Nash equilibria in two-network zero-sum games. *Automatica* **2013**, *49*, 1683–1692. [CrossRef]
60. Ye, M.; Hu, G. Game design and analysis for price-based demand response: An aggregate game approach. *IEEE Trans. Cybern.* **2016**, *47*, 720–730. [CrossRef]
61. Grammatico, S. Dynamic control of agents playing aggregative games with coupling constraints. *IEEE Trans. Autom. Control* **2017**, *62*, 4537–4548. [CrossRef]
62. Yi, P.; Pavel, L. Distributed generalized Nash equilibria computation of monotone games via double-layer preconditioned proximal-point algorithms. *IEEE Trans. Control Netw. Syst.* **2018**, *6*, 299–311. [CrossRef]
63. Paccagnan, D.; Gentile, B.; Parise, F.; Kamgarpour, M.; Lygeros, J. Nash and Wardrop equilibria in aggregative games with coupling constraints. *IEEE Trans. Autom. Control* **2018**, *64*, 1373–1388. [CrossRef]
64. Parise, F.; Ozdaglar, A. A variational inequality framework for network games: Existence, uniqueness, convergence and sensitivity analysis. *Games Econ. Behav.* **2019**, *114*, 47–82. [CrossRef]
65. Xiao, Y.; Hou, X.; Hu, J. Distributed solutions of convex-concave games on networks. In Proceedings of the 2019 American Control Conference (ACC), Philadelphia, PA, USA, 10–12 July 2019; pp. 1189–1194.
66. Bimpikis, K.; Ehsani, S.; Ilkılıç, R. Cournot competition in networked markets. *Manag. Sci.* **2019**, *65*, 2467–2481. [CrossRef]
67. Zhang, K.; Yang, Z.; Başar, T. Policy optimization provably converges to Nash equilibria in zero-sum linear quadratic games. *arXiv* **2019**, arXiv:1906.00729.
68. uz Zaman, M.A.; Zhang, K.; Miehling, E.; Başar, T. Reinforcement learning in non-stationary discrete-time linear-quadratic mean-field games. In Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC), Jeju, Korea, 14–18 December 2020; pp. 2278–2284.
69. Roudneshin, M.; Arabneydi, J.; Aghdam, A.G. Reinforcement learning in nonzero-sum Linear Quadratic deep structured games: Global convergence of policy optimization. In Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC), Jeju, Korea, 14–18 December 2020; pp. 512–517.
70. Shi, Y.; Zhang, B. Multi-agent reinforcement learning in Cournot games. In Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC), Jeju, Korea, 14–18 December 2020; pp. 3561–3566.
71. Sohet, B.; Hayel, Y.; Beaude, O.; Jeandin, A. Learning Pure Nash Equilibrium in Smart Charging Games. In Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC), Jeju, Korea, 14–18 December 2020; pp. 3549–3554.
72. Sinha, A. Mechanism Design with Allocative, Informational and Learning Constraints. Ph.D. Thesis, 2017. Available online: http://web.eecs.umich.edu/~anastas/anastas/docs/northwestern_2017.pdf (accessed on 14 May 2021).
73. Zhou, X. On the Fenchel duality between strong convexity and Lipschitz continuous gradient. *arXiv* **2018**, arXiv:1803.06573.