A Decentralized Mechanism for Improving the Functional Robustness of Distribution Networks

Benyun Shi and Jiming Liu, *Fellow, IEEE*

***Abstract*—Most real-world distribution systems can be modeled as distribution networks, where a commodity can flow from source nodes to sink nodes through junction nodes. One of the fundamental characteristics of distribution networks is the functional robustness, which reflects the ability of maintaining its function in the face of internal or external disruptions. In view of the fact that most distribution networks do not have any centralized control mechanisms, we consider the problem of how to improve the functional robustness in a decentralized way. To achieve this goal, we study two important problems: 1) how to formally measure the functional robustness, and 2) how to improve the functional robustness of a network based on the local interaction of its nodes. First, we derive a utility function in terms of network entropy to characterize the functional robustness of a distribution network. Second, we propose a decentralized network pricing mechanism, where each node need only communicate with its distribution neighbors by sending a "price" signal to its upstream neighbors and receiving "price" signals from its downstream neighbors. By doing so, each node can determine its outflows by maximizing its own payoff function. Our mathematical analysis shows that the decentralized pricing mechanism can produce results equivalent to those of an ideal centralized maximization with complete information. Finally, to demonstrate the properties of our mechanism, we carry out a case study on the U.S. natural gas distribution network. The results validate the convergence and effectiveness of our mechanism when comparing it with an existing algorithm.**

***Index Terms*—Distribution networks, network entropy, pricing mechanism, robustness.**

I. Introduction

DISTRIBUTION networks have been extensively adopted to model the dynamics of commodity flows on various real-world distribution infrastructures [1]–[3]. Each node in a distribution network may represent a source, a junction, or a sink. The function of a distribution network is to control commodity flows from where it is produced (i.e., the source nodes) to where it is consumed (i.e., the sink nodes) in order to maintain a supply-demand balance on individual nodes. Historical data shows that distribution networks may be disrupted due to internal or external uncertainties [4], where certain supply disruptions on a node in a network may spread to other nodes and potentially break their supply-demand balance. In this paper, we aim to formally characterize the function of a distribution network so that commodity flows can be precontrolled before a disruption happens.

Robustness is one of the most fundamental characteristics for representing how a system maintains its function under uncertainties. Accordingly, the *functional robustness* of a distribution network refers to its ability of maintaining a supply-demand balance on individual nodes in the face of possible disruptions. Specifically, a distribution network with high functional robustness should be capable of 1) mitigating the impact of possible disruptions based on the predefined commodity flows on the network, and 2) satisfying the demand of each sink node as much as possible. Different from existing studies on how to reduce the disruption impact *afterward* [5], in this paper, we focus mainly on how to improve the functional robustness of a distribution network *beforehand* in order to mitigate the impact of possible disruptions.

A. Motivation

To improve the functional robustness of a distribution network by precontrolling its commodity flows, it would be necessary to address the following two issues.

The first is how to formally evaluate the functional robustness in terms of the dynamics of commodity flows on a distribution network. So far as we know, there is no standard answer to this question. In the literature of complex networks, many efforts have been made to characterize network robustness in terms of degree distribution [6], remaining degree distribution [7], or even community structure [8]. However, such definitions only reflect the *structural robustness* of a network. With respect to a distribution network, the functional robustness is determined by not only the network structure but also commodity flows on the network. That is, the extent to which disruptions on one node may affect other nodes depends on the distribution strategy of each individual node. Distribution strategy of a node describes how the node distributes the commodity to its downstream neighbors in the network. Therefore, the evaluation of the functional robustness should characterize both network structure and the dynamics of commodity flows at the same time.

The second is how to improve the functional robustness of a distribution network in a decentralized manner. For many large-scale distribution networks, centralized control becomes impossible due to either technical reasons (e.g., computational complexity and communication costs/delays) or management reasons (e.g., distribution infrastructures belong to different owners). Here, we focus on the flow control problem with the

Manuscript received June 15, 2011; revised January 4, 2012; accepted March 12, 2012. Date of publication April 25, 2012; date of current version September 12, 2012. The research reported in this paper has been supported in part by a Hong Kong Research Grants Council (RGC) Grant (210508) and in part by a Hong Kong Baptist University (HKBU) Faculty Research Grant (FRG2/10-11/110). Prof. Jiming Liu is the corresponding author of this paper. This paper was recommended by Associate Editor T. Vasilakos.

The authors are with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong (e-mail: byshi@comp.hkbu.edu.hk; jiming@comp.hkbu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2012.2191774

following two properties. On the one hand, the nodes can make their own decisions about their distribution strategies. By doing so, the quantity of commodity flows passing through a node depends on distribution strategies of its upstream nodes. On the other hand, constrained by the network structure, the nodes are restricted to communicate only with its distribution neighbors. As for the decentralized mechanism proposed in this paper, we further address the following issues:

1) How do the nodes communicate only with their distribution neighbors so that the functional robustness can be improved by maximizing their own payoffs?
2) How does a node adaptively distribute commodity flows to its downstream neighbors in order to maximize its payoff?
3) Can the decentralized mechanism produce optimization results equivalent to those of a theoretically ideal, centralized control mechanism?

B. Related Work

Two areas of study are related to this work. They are: 1) robustness analysis and improvement of complex networks, and 2) dynamics optimization on networks such as resource allocation, flow, or congestion control in various networks.

Existing studies on complex networks have identified two types of robustness, i.e., *static robustness* and *dynamic robustness* [9]. Static robustness represents the ability of a network (e.g., random network [10], scale-free network [11], correlated networks [12], and uncorrelated networks [13]) to maintain its structural connectivity after a series of random or malicious node deletions, while dynamic robustness focuses on modeling cascading failures due to the disturbance of the dynamics on the network [14]–[16]. On the one hand, to characterize network robustness, the notion of network entropy has been adopted. However, existing definitions of network entropy use only the structural information of complex networks such as degree distribution [6], remaining degree distribution [7], and community structure [8]. They cannot be used to evaluate the functional robustness of commodity flows on distribution networks. On the other hand, to improve the network robustness, many optimization approaches have been proposed, most of which concentrate on changing the connectivity of a network [17]–[22]. Therefore, they cannot be directly adopted to tackle the functional robustness of distribution networks. In this paper, by formally defining a notion of network entropy in terms of the dynamics of commodity flows, we show how such an evaluation provides us with new insights into a way of improving the functional robustness of a distribution network through controlling its commodity flows.

Dynamics optimization on networks is one of the four fundamental types of optimization problems related to networks [23]. In recent years, the problems of resource allocation on networks have been extensively studied, most of which aim to efficiently or economically utilize limited resources by using various approaches such as operational research [24], [25] and negotiation [26]–[28]. Different from previous studies, we provide a new perspective of distributing/allocating resources in a robust manner. In order to solve optimization problems on networks, many decentralized mechanisms have be proposed, particularly for flow/congestion control, and route selection in communication networks [29]–[33]. One common feature of these problems is that the source and destination of a flow (e.g., the packages transmitted between two Internet users) is determined in advance. Decentralized pricing mechanisms are then designed 1) to control the transmission rates of information flows to avoid network congestion, 2) to minimize information loss at bottleneck nodes, or 3) to select transmission path with certain desirable purposes (e.g., minimum delay). Nevertheless, for the flow control problem in a distribution network, a sink node can receive commodity from any source nodes to meet its demand through any distribution paths; that is, there are no predetermined source-destination pairs in the network. The decentralized mechanism should determine both the destination and the distribution path for each unit of commodity flows. Moreover, the "price" signals (e.g., nodal price) in existing studies, which have been used to reflect the status of certain restrained resources such as transportation capacity [24], are available to all "users." While the "price" signal adopted in this paper is used for a node to communicate with its neighbors. Details of our mechanism will be discussed in Section IV.

C. Our Contributions

The main contributions of this work are twofold. On the one hand, to formally evaluate the functional robustness of a distribution network, we adopt the notion of network entropy in the field of complex networks. By treating commodity flows as random walks on the network, we present a definition of network entropy in terms of the dynamics of commodity flows. Such a definition can serve as a metric for globally measuring the robustness of commodity flows dynamics based on the Ruelle-Bowens random walk theory [34]. Further, to evaluate the demand satisfaction of the sink nodes, we adopt a penalty function for each sink node, which reflects the tolerance of the node for not receiving enough supply. Accordingly, a utility function, which integrates both network entropy and penalty functions together, has been derived to characterize the functional robustness of a distribution network. By doing so, the functional robustness of a distribution network can be improved by solving a utility optimization problem under some distribution constraints.

On the other hand, to solve the constrained utility optimization problem in a decentralized manner, we propose a network pricing mechanism in this paper. According to our mechanism, each node can determine its own distribution strategy by communicating only with its distribution neighbors with "price" signals. Specifically, there are two levels of dynamics in our mechanism:

- **Communication at the network level**: Each node communicates with its neighbors by sending a "price" signal to its upstream neighbors or receiving "price" signals from its downstream neighbors, where the "price" indicates its marginal payoff for per unit of commodity flows;
- **Maximization at the local level**: Each node maximizes its own payoff based on signals received from its downstream neighbors to determine its distribution strategy.

We adopt a single message transmission (SMT) algorithm for each node to maximize its payoff function at the local level. According to the algorithm, a node can adaptively adjust its distribution strategy by recursively interacting with its outbranching edges. Then, we mathematically prove that our decentralized mechanism can produce optimization results equivalent to that of a centralized maximization. To demonstrate the performances of our mechanism, we carry out a case study on the U.S. natural gas distribution network. By comparing it with a couple message transmission (CMT) algorithm, we further answer the following questions: Can the decentralized mechanism converge? What factors affect the convergence speed? How effective is our mechanism when compared with a centralized control?

For the decentralized mechanism in this paper, we focus mainly on how to achieve global optimization based on the local interactions of network nodes. Therefore, we ignore the incentive issues and assume that the nodes voluntarily announce their true "prices." Moreover, we assume that all nodes are price-takers, i.e., they do not anticipate how their announced "prices" will affect other nodes' decisions. This is reasonable because for large-scale distribution networks, it is hard for a node to anticipate its effects on other nodes [35].

The reminder of this paper is organized as follows. In Section II, we introduce the concepts of distribution networks and two fundamental challenges for improving their functional robustness. In Section III, we present a definition of network entropy and derive a constrained utility maximization problem for improving the functional robustness. In Section IV, we present a decentralized network pricing mechanism, and prove that our mechanism can produce results equivalent to those of a centralized maximization. In Section V, we carry out a case study on the U.S. natural gas distribution network to demonstrate the properties and evaluate the effectiveness of our pricing mechanism. Finally, we conclude this paper and present potential future work in Section VI.

II. Problem Statements

In this section, we first present the definitions of distribution networks and distribution strategy. Then, we introduce the concept of the functional robustness. To improve the functional robustness of a distribution network, we further propose two challenging issues to be addressed in this paper.

A. Distribution Networks

Most real-world distribution systems can be modeled as distribution networks, where a commodity can flow from source nodes to sink nodes through junction nodes [2]. A junction node in a distribution network may consume some of the incoming flows, and forward the rest to other nodes. As shown in Fig. 1, each node in a distribution network may associate with a quantity of self-supply (e.g., v_1 and v_2 with "+" circles) or self-consumption (e.g., v_3 and v_5 with "−" circles). By adding some virtual nodes to represent the junction nodes' self-consumption, such a distribution network can be described by a network with pure suppliers, intermediaries, and consumers. As shown in Fig. 2, by adding direct edges from v_3 to a virtual

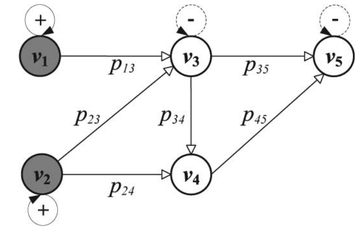


Fig. 1. Illustration of a distribution network with sources, conduit junctions, and sinks. Nodes v_1 and v_2 represent sources. v_3 and v_4 are conduit junctions with and without self-consumption, respectively. v_5 represents a sink.

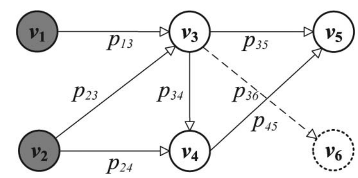


Fig. 2. Illustration of a distribution network with pure suppliers, intermediaries, and consumers. Nodes v_1 and v_2 represent resource suppliers. v_3 and v_4 are intermediaries. v_5 and v_6 represent consumers.

node v_6, the distribution network in Fig. 1 becomes a network with pure suppliers (v_1 and v_2), intermediaries (v_3 and v_4), and consumers (v_5 and v_6). In doing so, the inflows of an intermediary should be equal to the outflows. This property is called *flow conservation*.

Definition 1: A **distribution network** can be described by a directed acyclic network $G(V, E, W)$. Each node $v_i \in V$ represents either a supplier, or an intermediary, or a consumer. If there are flows from v_i to v_j, then $e_{ij} = 1$; otherwise, $e_{ij} = 0$. Moreover, each edge is associated with a weight $w_{ij} \in [0, 1]$, which represents the proportion of flows distributed from v_i to v_j.

According to the definition, the topology of $G(V, E, W)$ can be described by an $N \times N$ adjacency matrix $E = \{e_{ij}\}_{N\times N}$, which is directed and Boolean, i.e., $e_{ij} \in \{0, 1\}$. Furthermore, each node is associated with a weight set $\mathbf{w}_i = \{w_{ij} | e_{ij} = 1, \sum_j w_{ij} = 1\}$, which represents v_i's distribution strategy.

Definition 2: The **distribution strategy** of a node v_i can be represented by the weight set $\mathbf{w}_i$, which describes how the node distributes commodity flows to its downstream neighbors.

B. Functional Robustness

When a disruption happens, the nodes may not significantly change their distribution strategies in the short run. For example, as reported in [36], after the Tucson-to-Phoenix gasoline pipeline ruptured on July 30, 2003, Phoenix consumers did not reduce their gasoline purchases substantially. Therefore, the extent to which a node may be affected by a disruption depends on the distribution strategies of its upstream neighbors. In other words, the robustness of a distribution network relies on how the nodes mitigate the possible disruptions.

Definition 3: The **mitigation capability** of a node v_i is defined as $MC_i(\mathbf{w}_i) = -\sum_{w_{ij}\in\mathbf{w}_i} w_{ij} \log w_{ij}$, which is determined by its distribution strategy $\mathbf{w}_i$.

The value of MC_i measures the diversity of outflows from v_i. The maximum is obtained for $w_{ij} = 1/m_i$, where m_i is the number of outbranching edges from v_i. This means that its downstream neighbors will evenly undertake the possible disruptions. Therefore, the more evenly the commodity is distributed by v_i, the higher mitigation capacity v_i has. Consequently, the mitigation capacity of a distribution network depends on how the nodes collectively mitigate possible disruptions, which is related to both the network topology and the nodes' distribution strategies.

Definition 4: The **functional robustness** of a distribution network measures the capability of the network 1) to mitigate the possible disruption impact on each node, and 2) to meet the demand of all consumers in the face of the disruption.

Based on the definition, it should be noted that even if all nodes have the highest mitigation capabilities, the network may not have the highest functional robustness. For example, the distribution strategy $w_{23} = w_{24} = 1/2$ (respectively, $w_{34} = w_{35} = w_{36} = 1/3$) corresponds to the highest mitigation capability for v_2 (respectively, v_3) in Fig. 2. Suppose that a disruption on v_2 causes the reduction of its commodity supply by one unit. Consequently, v_5 will lose its supply by 5/6, and v_6 will lose its supply by 1/6. We cannot say the network achieves the highest functional robustness because there exists another set of distribution strategies that will make v_5 and v_6 evenly undertake the loss of supply. Therefore, to formally evaluate the functional robustness, it would be necessary to characterize network topology, distribution strategy of each node, as well as consumers' demand satisfaction.

C. Challenges

Due to the lack of centralized control in most distribution networks, the dynamics of commodity flows is often determined by the distribution strategies of interactive individuals [1]. In this paper, to improve the functional robustness of a distribution network, we aim to tackle the following two challenging problems.

Problem I: How can we formally evaluate the functional robustness of a distribution network?

Problem II: How can we design a decentralized mechanism so that each node can adaptively adjust its distribution strategy to improve the functional robustness of the whole network?

For the first problem, a mathematical evaluation, which integrates network topology, distribution strategies, and consumers' demand satisfaction, is necessary so that we can further improve the functional robustness by controlling the commodity flows on the network. For the second problem, the difficulties are twofold: 1) how to design the communication rules for the nodes to interact with their distribution neighbors, and 2) how the nodes adjust their distribution strategies individually to achieve the improvement of the functional robustness at the global level. More specifically, our research questions about the decentralized mechanism are as follows:

1) **Convergence**: Can the decentralized mechanism finally converge to a steady solution?
2) **Convergence speed**: What kind of factors will affect the convergence speed of the mechanism?
3) **Effectiveness**: Can our mechanism produce results equivalent to those of a centralized optimization?

III. Characterizing the Functional Robustness

In this section, we first define a notion of network entropy to evaluate the mitigation capability of a distribution network based on the Ruelle-Bowens random walk theory and related studies [37], [38]. Then, we derive a constrained utility optimization problem to improve the functional robustness, which takes into consideration consumers' demand satisfaction.

A. Definition of Network Entropy

We treat commodity flows in $G(V, E, W)$ as a Ruelle-Bowens random walk, in which w_{ij} represents the probability that a unit of flow which enters v_i is distributed along e_{ij} to v_j. We denote the stationary state of node v_i (i.e., π_i) as the quantity of the commodity that flows through v_i. Then, π_i can be calculated as follows:

$$\pi_i = \sum_{e_{ki}=1} \pi_k \cdot w_{ki}. \tag{1}$$

It is obvious that the stationary states of all nodes are interdependent, that is, each π_i depends on the stationary states of its upstream neighbors, i.e., $\{\pi_k | e_{ki} = 1\}$. Given the quantity of supply s_i of each supplier $v_i \in S$, $\pi_i \in \Pi^s$ can be calculated as $\pi_i = s_i / \sum_{v_k \in S} s_k$. Consequently, the stationary states of other nodes (i.e., Π^{-s}) can be calculated based on (1) and the weight matrix W. In other words, the stationary state distribution Π and the weight matrix W of the network are mutually dependent.

The network entropy is defined based on two mutually dependent distributions (i.e., Π and W) as follows:

$$H(\Pi * W) = H(\Pi^s) + H(W|\Pi^{-s}, \Pi^s) \tag{2}$$

where $H(\Pi^s)$ is the entropy of stationary distribution for all independent suppliers, $H(W|\Pi^{-s}, \Pi^s)$ is the entropy of weight matrix W condition upon the prior knowledge Π^s and Π^{-s}. The network entropy can be calculated as follows:

$$H(\Pi^s) = -\sum_{v_i \in S} \pi_i \log \pi_i \tag{3}$$

$$H(W|\Pi^{-s}, \Pi^s) = -\sum_{v_i \notin S} \pi_i \sum_{e_{ij}=1} w_{ij} \log w_{ij}. \tag{4}$$

Since the stationary states of suppliers are known in advance, the value of $H(\Pi^s)$ is a constant. Therefore, the value of network entropy is determined by $H(W|\Pi^{-s}, \Pi^s) = \sum_{v_i \notin S} H_i$, where $H_i = -\pi_i \sum_{e_{ij}=1} w_{ij} \log w_{ij}$ represents the local entropy of node v_i. We can find that the local entropy $H_i = \pi_i \cdot MC_i(\mathbf{w}_i)$ involves both mitigation capability MC_i and stationary state π_i of v_i, where π_i represents the extent to which v_i may be affected by other nodes.

The relationship between the Ruelle-Bowens random walk and the robustness of a complex network has been explored by Demetrius and Manke [38], and extended by Gómez-Gardeñes and Latora [39] and Delvenne and Libert [34]. With respect to a distribution network, such a definition of network entropy reflects the long-term behavior of the flow dynamics in a distribution network by measuring the possibility of a unit of energy flow on all possible distribution paths. The larger the network entropy, the less predictable the random walk; that is, the more equally the flow is carried on all possible distribution paths. Such a macroscopic characterization provides a new insight into how to control energy flows to improve the functional robustness of a distribution network.

B. Maximizing Network Entropy

Given the weight matrix of a distribution network, the mitigation capability of the network can be evaluated by the network entropy. In doing so, we can control the commodity flows (i.e., by adjusting the nodes' distribution strategies) so as to improve the mitigation capability of the network. Moreover, based on the definition of functional robustness, the commodity flows should be one of the perfect matches[1] among supplier and consumers, i.e., each consumer's demand should be satisfied. Therefore, a natural way to improve the functional robustness is to maximize network entropy under a set of distribution constraints, i.e., to solve the following constrained network entropy optimization problem:

$$\text{maximize} \quad H(W|\Pi^{-s}, \Pi^{s}) = \sum_{v_i \notin S} H_i \tag{5}$$

$$\text{subject to} \quad \sum_{e_{ji}=1} \pi_j w_{ji} = \sum_{e_{ik}=1} \pi_i w_{ik}, \quad \forall v_i \notin S \tag{6}$$

$$\sum_{e_{ij}=1} w_{ij} = 1, \quad \forall v_i \notin C \tag{7}$$

$$\pi_i = D_i, \quad \forall v_i \in C \tag{8}$$

$$w_{ij} \geq 0. \tag{9}$$

Here, (6) represents the inflows of an intermediary node should be equal to its outflows; that is, it should have the property of *flow conservation*. Equation (8) describes a crucial constraint of a distribution network, i.e., the demand D_i of each consumer $v_i \in C$ should be satisfied through the distribution. Equation (9) avoids negative flows on each edge. Because the objective function is continuous and the feasible region is compact, an optimal solution exists.

With respect to the objective function, the value of the local entropy H_i is determined by not only the stationary state π_i but also the distribution strategy $\mathbf{w}_i$ of v_i. Since the stationary states are interdependent, an increase in one node's local entropy may cause a decrease in other nodes. With respect to the constraints, the most important constraint is to satisfy the demand of all consumers [i.e., (8)]. However, the constraint of (8) seems to be too strict. Practically, we may allow a slight mismatch between π_i and D_i for achieving a high mitigation capability. Therefore, in the next section, we relax this constraint in order to form a more general formulation to improve the functional robustness.

C. General Formulation for the Functional Robustness

In this section, we attempt to relax the demand satisfaction constraint [i.e., (8)] by associating each consumer node with a "penalty" function $F_i(\cdot)$, which penalizes the mismatch between the stationary state π_i and the demand D_i of consumer $v_i \in C$. By doing so, the objective function becomes to be a utility function representing the difference between network entropy and total penalties received by all consumers. In other words, we need to maximize

$$U = \sum_{v_i \notin S} H_i - \sum_{v_i \in C} F_i(\pi_i - D_i). \tag{10}$$

Note that by appropriately choosing the penalty function $F_i(\cdot)$, the general formulation can be transformed to the original problem proposed in the previous section. One way is to set $F_i(\cdot)$ to be zero if $\pi_i = D_i$, and $+\infty$ if $\pi_i \neq D_i$.

To be consistent with the original network entropy optimization problem, one important setting for the penalty function $F_i(\cdot)$ is that $F_i(\pi_i - D_i)$ should be equal to zero if $\pi_i = D_i$. Further, we can set $F_i(\pi_i - D_i)$ to be a continuously increasing (respectively, decreasing) and differentiable function if $\pi_i > D_i$ (respectively, $\pi_i < D_i$), which means that it satisfies:

$$F_i(\pi_i - D_i) = \int_0^{|\pi_i - D_i|} f_i(x)dx \tag{11}$$

where $f_i(\cdot)$ is a continuously increasing function. We call $f_i(\cdot)$ the penalty price function associated with consumer $v_i \in C$ for the mismatch between its stationary state and demand. In practice, the penalty price can be determined in advance to reflect the consumers' tolerance for the lack of supply. By doing so, we can improve the functional robustness by maximizing the utility function [i.e., (10)] under the distribution constraints of (6), (7), and (9).

To centrally solve the constrained utility maximization problem, we would need complete information and full control of a distribution network. However, for most real-world distribution networks, the commodity flows are determined by specific nodes rather than a central authority, i.e., there is no central authority that has such complete information to control whole distribution networks. Furthermore, a centralized solution does not scale well when the size of the distribution networks becomes large. Therefore, in the next section, we propose a network pricing mechanism that is aimed to solve the generalized network entropy maximization problem in a decentralized way.

IV. Decentralized Network Pricing Mechanism

In this section, we first present a network pricing mechanism and introduce in detail the implementation of the mechanism. Then, we mathematically prove that the competitive equilibrium of our mechanism is also an optimal solution to the generalized entropy maximization problem.

A. Pricing Mechanism

The pricing mechanism consists of two levels of dynamics. On the one hand, each node on a distribution network communicates only with its neighbors by sending a "price" signal to its upstream neighbors and receiving "price" signals from its downstream neighbors. On the other hand, each node adaptively

[1]Since we focus mainly on how to improve the functional robustness of an existing distribution network, it is reasonable to assume that there exists at least one perfect match among suppliers and consumers in terms of both the network structure and their supply and demand

adjusts its distribution strategy by maximizing its own payoff functions at the local level.

1) Communication Rules: Given a distribution network $G(V, E, W)$, we assume that there are N node players $\{I_1, \ldots, I_N\}$, each of which plays as a manager of a node to determine its outflows. For each node v_i with m_i out-edges, we further assume that there are m_i edge players (i.e., $\{L_1^i, \ldots, L_{m_i}^i\}$) who compete for the outflows from the node player I_i. We propose the following communication rules for the node players so that they need to communicate only with their neighbors to achieve the global maximization.

At the network level, each node player I_i announces a "price" to all its upstream neighbors $\{I_j | e_{ji} = 1\}$, which reflects the marginal benefit or loss with respect to the value of stationary state π_i. The price is calculated as follows:

$$\xi_i = \frac{\partial U}{\partial \pi_i}. \tag{12}$$

Based on (10) and (12), it is easy to calculate that 1) for a supplier or intermediary, the price equals to $-\sum_{e_{ij}=1} w_{ij} \log w_{ij}$, and 2) for a consumer, the price equals to $-f(\pi_i - D_i)$. In addition to announcing a "price" to its upstream neighbors, each node player also receives "price" signals from its downstream neighbors. By doing so, a node player I_i can adjust its distribution strategy $\mathbf{w}_i = \{w_{ij} | e_{ij} = 1\}$ by maximizing the following payoff function:

$$Q_i(\mathbf{w}_i) = \sum_{e_{ij}=1} U_{ij}(w_{ij}) = \pi_i \cdot \sum_{e_{ij}=1} (-w_{ij} \log w_{ij} + w_{ij}\xi_j) \tag{13}$$

subject to

$$\sum_j w_{ij} \leq 1. \tag{14}$$

Up till now, we have successfully transformed the generalized utility maximization problem into a set of resource allocation problems at the local level, where each node only needs to control (or allocate) its outflows to maximize its own payoff.

2) Node Payoff Maximization: At the local level, we focus on how a node player I_i maximizes its payoff function [(13)] by adaptively allocating commodity flows to its downstream neighbors. To solve this problem, we introduce an auction game as follows. Each edge player L_j^i bids for outflows from I_i with a payment (or bid) $\kappa_j^i (\geq 0)$. After receiving all edge players' bids, the node player then announces a common payment μ_i, which is determined by the bids from all edge players. Thus, the edge weight w_{ij} is determined by $w_{ij} = \kappa_j^i / \mu_i$. Since a bid is the private information of an edge player, i.e., an edge player cannot know other edge players' bids, once the common payment μ_i is given, an edge player sets its bid κ_j^i by maximizing the following payoff function:

$$\begin{aligned} P_{ij}\left(\kappa_j^i, \mu_i\right) &= U_{ij}\left(\frac{\kappa_j^i}{\mu_i}\right) - \kappa_j^i \\ &= \pi_i \cdot \left(-\frac{\kappa_j^i}{\mu_i} \log \frac{\kappa_j^i}{\mu_i} + \frac{\kappa_j^i}{\mu_i} \cdot \xi_j\right) - \kappa_j^i. \end{aligned} \tag{15}$$

The optimal κ_j^i should satisfy the following equation:

$$\frac{\partial P_{ij}\left(\kappa_j^i, \mu_i\right)}{\partial \kappa_j^i} = 0 \Leftrightarrow \mu_i = \frac{\partial U_{ij}(w_{ij})}{\partial w_{ij}} \tag{16}$$

which means

$$\mu_i = \pi_i \cdot (\xi_j - \log w_{ij} - 1). \tag{17}$$

In summary, a node player I_i has two kinds of action: 1) to announce a common payment μ_i based on edge players' bids to determine its outflows w_{ij}, and 2) to announce a price ξ_i to its upstream neighbors in order to affect their distribution strategies. Based on the local maximization and the global communication rules, the functional robustness of a distribution network can be improved by adaptively adjusting the nodes' distribution strategies.

B. Implementation

Based on the network pricing mechanism proposed in Section IV-A, we introduce how to implement the pricing mechanism in this section. First, we introduce a SMT algorithm for each node to maximize its payoff. Then, we propose three network updating algorithms according to the order of the nodes maximizing their payoffs.

1) Single Message Transmission Algorithm: According to the auction game for node payoff maximization, each edge player in the SMT algorithm only needs to transmit a single message (i.e., its bid κ_j^i) to its corresponding node player I_i, and receive the common payment (i.e., the comment payment μ_i) from the node player. Initially, the common payment is set to be 0, i.e., $\mu_i(0) = 0$.

Algorithm 1: UpdateEdgeWeight(I_i)

Initialize $\mu_i(0) = 0$ for the node player I_i;
Calculate π_i using inflows of I_i;
Find $\{I_j | e_{ij} = 1\}$ of players I_i;
$relativeError = 1$;
while $relativeError > threshold$ **do**
  Calculate $\mu_i(t)$ based on Equation (22);
  for $I_j \in \{I_j\}$ **do**
    Receive ξ_j from I_j;
    Update w_{ij} based on Equation (18);
  end for
  $relativeError = |\sum_j w_{ij} - 1|$;
end while
Normalize w_{ij} such that $\sum_j w_{ij} = 1$;
Update ξ_i using Equation (30);
for $I_j \in \{I_j\}$ **do**
  if I_j is a consumer **then**
    Update ξ_j using Equation (30);
  end if
end for

Given $\mu_i(t)$ at each iteration t, based on (17), $w_{ij}(t)$ is updated by

$$w_{ij}(t) = \exp\left\{\xi_j - \frac{\mu_i(t)}{\pi_i(t)} - 1\right\}. \tag{18}$$

To update the common payment $\mu_i(t)$ at each iteration t, we consider the dual problem of the node payoff maximization (i.e., (13)). The objective function of the dual problem is to find μ_i that minimize

$$D(\mu_i) = \sum_{e_{ij}=1} U_{ij}(w_{ij}) - \mu_i \left(\sum_j w_{ij} - 1\right). \tag{19}$$

We solve the dual problem using a gradient projection method, where μ_i is adjusted in an opposite direction to the gradient $\nabla D(\mu_i)$:

$$\mu_i(t+1) = \mu_i(t) - \gamma \cdot \frac{\partial D}{\partial \mu_i}(\mu_i(t)) \tag{20}$$

where γ is a step size. Based on (19), we have

$$\frac{\partial D}{\partial \mu_i} = -\left(\sum_j w_{ij} - 1\right). \tag{21}$$

Substituting (21) into (20), we obtain the following updating rule for the common payment of I_i at each iteration:

$$\mu_i(t+1) = \mu_i(t) + \gamma \cdot \left(\sum_j w_{ij}(t) - 1\right). \tag{22}$$

At each iteration t, $\mu_i(t)$ is first updated based on (22). Then, the weight $w_{ij}(t)$ is calculated based on (18). We use the difference between $\sum_j w_{ij}(t)$ and 1 as a signal to characterize the convergence of the algorithm. If the difference is smaller than a threshold, i.e., $|\sum_j w_{ij} - 1| \le threshold$, the iteration stops. That means, the algorithm converges to a steady state. By doing so, the node payoff maximization problem can be solved iteratively. When the iteration stops, the values of $\mathbf{w}_i$ are then normalized such that $\sum_j w_{ij} = 1$. Then, the "price" signal ξ_i is updated based on newly updated edge weights and (30). If there are consumers in I_i's downstream neighbors, i.e., $\{I_j\}$, the value of ξ_j is also updated. The detailed algorithm is shown in Algorithm 1.

2) Network Updating Algorithms: Each node player in the network pricing mechanism only needs to communicate with its upstream/downstream neighbors. According to the iterative node payoff maximization algorithm (i.e., the SMT algorithm), each node player I_i receives signals ξ_j from its downstream node players to update its distribution strategy $\mathbf{w}_i$. However, the order of node players doing payoff maximization may also be crucial. In this section, we propose three updating algorithms corresponding to three different node maximization orders: the depth-first updating algorithm, the breadth-first updating algorithm, and the random updating algorithms.

For the depth-first and breadth-first updating algorithms, at each round, consumer players are first allowed to announce the "price" signals (i.e., the values of ξ) to their upstream neighbors. Then, other players are allowed to do maximization in depth-first and breadth-first orders, respectively. For the random updating algorithm, the node players do payoff maximization based on Algorithm 1 in a random order. Two important processes for the three algorithms are as follows:

- Each player I_i receives the "price" signals (e.g., ξ_j) from its downstream neighbors and updates its distribution strategy by Algorithm 1 to maximize its payoff function;
- After I_i updating its distribution strategy, it will send feedback information to its downstream neighbors so that they can recalculate their values of ξ.

Details of the three algorithms are shown in Algorithms 2, 3, and 4.

Algorithm 2: Depth-FirstUpdate(G)
1: Starting from consumers, generate a queue Q in a depth-first order;
2: **for** Round $r = 1 : R$**do**
3: **for**$\forall I_i \in Q$**do**
4: UpdateEdgeWeight(I_i);
5: **end for**
6: **end for**

Algorithm 3: Breadth-FirstUpdate(G)
1: Starting from consumers, generate a queue Q in a breadth-first order;
2: **for** Round $r = 1 : R$**do**
3: **for**$\forall I_i \in Q$**do**
4: UpdateEdgeWeight(I_i);
5: **end for**
6: **end for**

Algorithm 4: RandomUpdate(G)
1: **for** Round $r = 1 : R$**do**
2: Generate a queue Q in a random order;
3: **for**$\forall I_i \in Q$**do**
4: UpdateEdgeWeight(I_i);
5: **end for**
6: **end for**

There are two kinds of dynamics in our network pricing mechanism. First, each node player allocates its outflows by maximizing its payoff function using Algorithm 1. Second, node players communicate with each other by sending or receiving "price" signals. With respect to the network updating algorithms, we assume that each node player will not send signals to its upstream neighbors until it finishes the maximization. This is reasonable because in practice it is easy for a node to individually determine its distribution strategy.

C. Equivalence Analysis

In this section, we show that an equilibrium of the network pricing mechanism proposed in Section IV-A is an optimal solution to the generalized entropy optimization problem in Section III-C. There are two steps to prove the equivalence. First, we focus on the auction game at the local level. We show that a *competitive equilibrium* of the game is also an optimal solution of (13). Second, we prove that if each node maximizes its payoff function as defined in the pricing mechanism [i.e., (13)], it is equivalent to maximize the generalized objective function [i.e., (10)].

We analyze the competitive equilibrium of the game at the local level following the development of Kelly [29]. We say a pair (κ_j^i, μ_i) is a competitive equilibrium if each edge player maximizes its payoff as defined in (15), i.e.,

$$P_{ij}\left(\kappa_j^i, \mu_i\right) \geq P_{ij}\left(\bar{\kappa}_j^i, \mu_i\right), \quad for \quad \forall \bar{\kappa}_j^i \geq 0 \tag{23}$$

and the price is set as follows:

$$\mu_i = \sum_j \kappa_j^i. \tag{24}$$

The following theorem shows that when edge players are price-takers, there exists a competitive equilibrium, which is also a solution of (13).

Theorem 1: There exists vectors $\kappa^i = (\kappa_{j_1}^i, \ldots, \kappa_{j_{m_i}}^i)$, $\mathbf{w}_i = (w_{ij_1}, \ldots, w_{ij_{m_i}})$, and μ_i such that

- κ^i maximizes (15) for all j;
- $w_{ij} = \kappa_j^i/\mu_i$, where $\mu_i = \sum_j \kappa_j^i$.

The vector $\mathbf{w}_i$ then also solves (13).

Proof: If κ^i solves (15), the derivative of (15) should satisfy

$$\frac{\partial P_{ij}\left(\kappa_j^i, \mu_i\right)}{\partial \kappa_j^i} = 0 \Leftrightarrow U'_{ij}\left(\frac{\kappa_j^i}{\mu_i}\right) = \mu_i. \tag{25}$$

The Lagrangian for the optimization problem of (13) is as follows:

$$L(\mathbf{w}_i, \lambda) = \sum_{e_{ij}=1} U_{ij}(w_{ij}) - \lambda\left(\sum_j w_{ij} - 1\right). \tag{26}$$

Then, the Karush-Kuhn-Tucker (KKT) necessary conditions is that

$$\frac{\partial L}{\partial w_{ij}} = \frac{\partial U_{ij}(w_{ij})}{\partial w_{ij}} - \lambda = 0 \tag{27}$$

$$\frac{\partial L}{\partial \lambda} = \sum_j w_{ij} - 1 = 0. \tag{28}$$

Let $\lambda = \mu_i$, we can say that (20) is equivalent to (25). Furthermore, by setting $w_{ij} = \kappa_j^i/\mu_i$ and $\mu_i = \sum_j \kappa_j^i$, we have (21) satisfied. By doing so, we complete the proof of the theorem. ■

Note that the out-edge e_{ij} of v_i in the distribution network $G(V, E, W)$ is also an in-edge of v_j at the same time, which means that the value of w_{ij} may directly affect both H_i and H_j. At the network level, node v_j may adjust its local entropy H_j by sending a price ξ_j to v_i so that w_{ij} can be adjusted by v_i. In the following theorem, we show that maximizing the general objective function [i.e., (10)] is equivalent to the pricing mechanism, where each node player maximizes its payoff function defined in (13).

Theorem 2: If there exists a weight matrix W and ξ_i such that

- for each $1 \leq i \leq N$, $\mathbf{w}_i$ maximizes (13);
- $\xi_i = -\sum_{e_{ij}=1} w_{ij} \log w_{ij}$, if v_i is a supplier or intermediary;
- $\xi_i = -f(\pi_i - D_i)$, if v_i is a consumer.

The matrix W also solves (10).

Proof: If $\mathbf{w}_i$ solves (13), for each $j(e_{ij} = 1)$, the derivative of (13) should satisfy

$$\frac{\partial Q_i(\mathbf{w}_i)}{\partial w_{ij}} = \pi_i \cdot \left[\frac{\partial(-w_{ij} \log w_{ij})}{\partial w_{ij}} + \xi_j\right] = 0 \tag{29}$$

where

$$\xi_j = \begin{cases} -\sum_{e_{jk}=1} w_{jk} \log w_{jk} & \text{if } v_j \notin C, \\ -f(\pi_j - D_j) & \text{if } v_j \in C. \end{cases} \tag{30}$$

If matrix W solves the optimization problem of (10), then the KKT necessary conditions is as follows:

$$\frac{\partial U(W)}{\partial w_{ij}} = \frac{\partial H_i}{\partial w_{ij}} + \frac{\partial H_j}{\partial w_{ij}} = 0, \qquad \text{if } v_j \notin C \tag{31}$$

$$\frac{\partial U(W)}{\partial w_{ij}} = \frac{\partial H_i}{\partial w_{ij}} - \frac{\partial F_j}{\partial w_{ij}} = 0, \qquad \text{if } v_j \in C. \tag{32}$$

It is easy to verify that

$$\frac{\partial H_i}{\partial w_{ij}} = \pi_i \cdot \frac{\partial(-w_{ij} \log w_{ij})}{\partial w_{ij}} \tag{33}$$

$$\frac{\partial H_j}{\partial w_{ij}} = -\pi \cdot \sum_{e_{jk}=1} w_{jk} \log w_{jk} \tag{34}$$

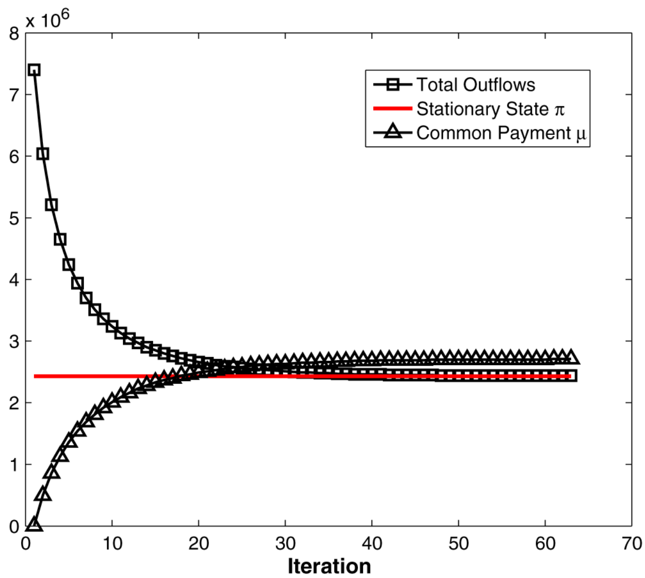
$$\frac{\partial F_j}{\partial w_{ij}} = \pi_i \cdot f(\pi_j - D_j). \tag{35}$$

By setting ξ_j as (30), we conclude that the matrix $W = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ solves (10). ■

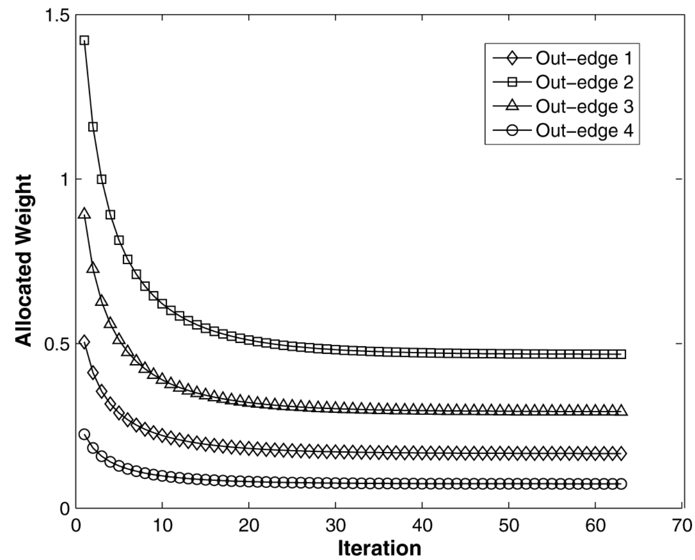
V. Case Study on the U.S. Natural Gas Distribution Network

In this section, we aim to evaluate the performances of the decentralized network pricing mechanism by conducting a case study on the U.S. natural gas distribution network. Specifically, we have designed and implemented experiments toward the following objectives:

1) To evaluate the convergence of the SMT algorithm adopted by each node;
2) To test the convergence speed of the SMT algorithm with different step sizes;
3) To show the dynamics of the functional robustness with different step sizes and network updating algorithms;



(a)



(b)

Fig. 3. Convergence and dynamic process of the SMT algorithm for node player I_1 (i.e., the Alabama state) with step size $\gamma = 0.1$. Fig. 3(a) demonstrates that as the common payment increases, the total outflows finally converge to the stationary state of I_1. Fig. 3(b) shows the convergence of allocated weights on its four out-edges.

4) To validate the effectiveness of our decentralized mechanism by comparing it with the CMT algorithm (see Appendix A for detail).

A. Data

The network represents the natural gas flows among the U.S. states in the year of 2007. Each node in the network corresponds to a state, and each directed edge corresponds to natural gas flows from one node to another. The data was compiled by the U.S. Energy Information Administration.[2]There are totally 52 nodes on the original distribution network, which can be converted to a network of size 92 with 12 pure suppliers, 52 intermediaries, and 28 consumers. We initialize the weight matrix of the distribution network according to the actual U.S. gas flows in the year of 2007.

B. Settings

There are three parameters that should be determined for the network pricing mechanism: the penalty price function $f_i(\cdot)$ for consumers, the step sizes γ and r for the SMT and CMT, respectively, and the stop criteria for node payoff optimization algorithms (i.e., the threshold).

- The penalty price function: the price function is defined for relaxing the demand satisfaction of consumers. Once it is defined, we can use network entropy to represent the functional robustness. For different applications, we can define different price functions. In this paper, we focus mainly on the following price function:

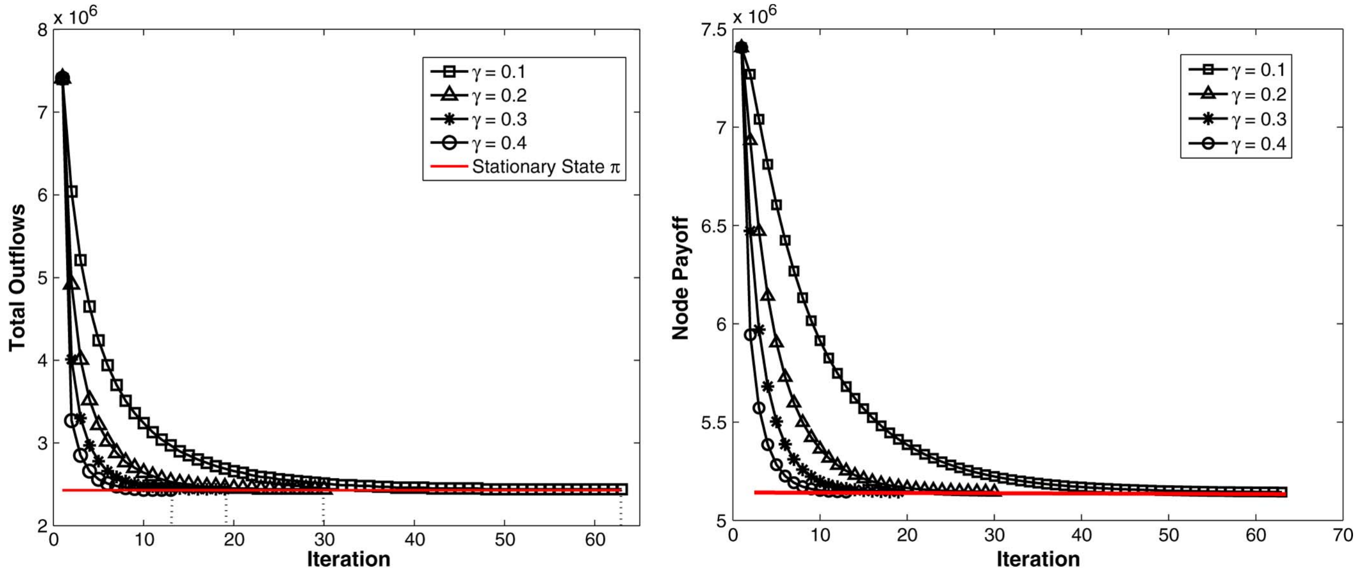
$$f_i\left(\pi_i(t)\right) = \frac{\pi_i(t) - D_i(t)}{\max\left\{\pi_i(t), D_i(t)\right\}}.$$

- The step size: the step size γ of the SMT algorithm determines how node players update the common payment μ_i in an opposite direction to the gradient $\nabla D(\mu_i)$ at each iteration, while the step size r of the CMT algorithm determines how edge players update their prices when the excess demand is nonzero (see Appendix A for detail). In this paper, we evaluate the performances of the SMT and CMT algorithms for different step sizes (i.e., $\gamma = 0.1$, 0.2, 0.3, 0.4 and $r = 5$, 10, 20, respectively).
- The threshold: the threshold of the node maximization algorithms plays as a stop criteria for the iterative process. In this paper, we set $threshold = 0.001$.

C. Results and Discussions

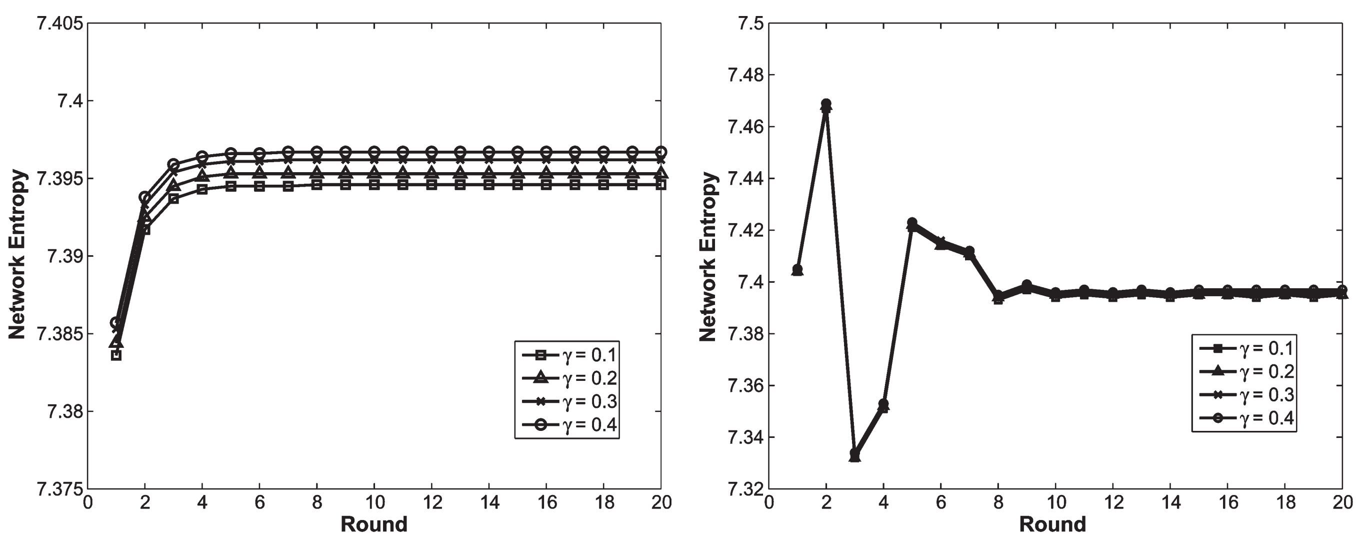
1) Convergence of the SMT Algorithm: We evaluate the convergence of the SMT algorithm for node payoff maximization on the network with step size $\gamma = 0.1$. The results show that the SMT algorithm for all node players can reach the stop criteria and finally converge to a steady state. Fig. 3 demonstrates the convergence and dynamic process of the SMT algorithm for node player I_1 (i.e., the state of Alabama) that has four out-edges on the distribution network. Simulation results show that dynamic processes of other node players are similar. It can be observed from Fig. 3(a) that as the common payment μ increases, the total outflows (i.e., $\pi_1 \cdot \sum_j w_{1j}$) finally converge to the stationary state π_1 of node player I_1, which means that all its inflows are allocated to its four out-edges. Fig. 3(b) shows the convergence of allocated weight (i.e., $w_{1j}(t)$) for each out-edge. It can be observed that the allocated weight on each edge will finally converge to a steady state, and the sum of all weights is close to 1. The results for all nodes on the network show that by appropriate step size, the SMT algorithm is convergent. However, in this paper, to find the set of step sizes that guarantee convergence for the SMT algorithm is not our major focus.

[2]Date source: http://www.eia.doe.gov/pub/oil_gas/natural_gas/analysis_publications/ngpipeline/usage.html.



(a) (b)

Fig. 4. Total outflows and payoffs of the SMT algorithm (i.e., Algorithm 1) for node player I_1 with respect to different step sizes, i.e., $\gamma = 0.1$, 0.2, 0.3, and 0.4, respectively. Fig. 4(a) and (b) show that the larger the step size is, the faster the steady state is reached. The reason for the decreasing of node payoffs in Fig. 4(b) is that the value of $\sum_j w_{1j}$ is greater than 1 and is converging to 1 before the steady state is reached.



(a) (b)

Fig. 5. Performance of depth-first updating algorithm [Fig. 5(a)] and breadth-first updating algorithm [Fig. 5(b)] with respect to different step sizes, i.e., $\gamma = 0.1$, 0.2, 0.3, and 0.4, respectively. It can be observed that the differences among final network entropy values are less than 0.01, which may be caused by the stop criteria at each node. In terms of the network entropy maximization, the step size may only slightly affect the final results.

2) Convergence Speed of the SMT Algorithm: We demonstrate the performances (i.e., total outflows and payoffs) of the SMT algorithm with respect to different step sizes, i.e., $\gamma = 0.1$, 0.2, 0.3, and 0.4, respectively. The results in Fig. 4 show that the step size can significantly affect the convergence speed of the SMT algorithm. It can be observed from Fig. 4(a) and (b) that the larger the step size is, the faster the steady state is reached. It can also be observed from Fig. 4(b) that the node payoffs decrease during the iterative process. This is because the value of $\sum_j w_{ij}$ is greater than 1 and is converging to 1 before the steady state is reached. Fig. 4 also shows that even with different step sizes, the final node payoffs are almost the same. This means that although the step size can significantly affect the convergence speed of the SMT algorithm, it can only slightly affect the final network entropy of the distribution network.

3) Functional Robustness With Respect to Different Step Sizes: We evaluate the performances of the SMT algorithm in terms of the functional robustness of the network. Because the penalty price function is given in advance, the functional robustness here can be represented with network entropy. Fig. 5 shows the performances of depth-first and breadth-first updating algorithms at the network level, where the SMT algorithm with different step sizes (i.e., $\gamma = 0.1$, 0.2, 0.3 and 0.4, respectively) is adopted for node payoff maximization at the local level. It can be observed that even with different step sizes for the node payoff maximization, the network updating

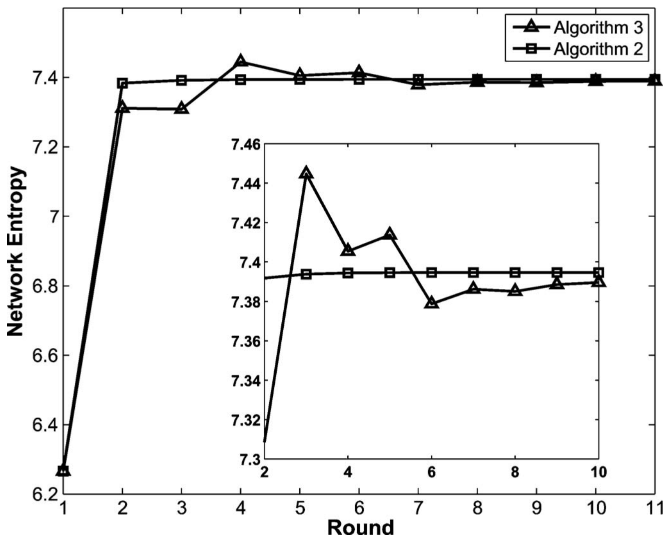


Fig. 6. Performance of Algorithms 2, 3, and 4 at the network level, where the SMT algorithm with $\gamma = 0.1$ is adopted at the local level.

algorithms (i.e., Algorithms 2 and 3) achieve almost the same network entropy at each round. As shown in Fig. 5(a) and (b), the differences of network entropy are less than 0.01. This is because at each round, each node has enough time to reach its stop criteria, that is, only the values of steady state are used at the network level.

4) Functional Robustness With Respect to Different Network Updating Algorithms: We demonstrate the comparison of the three network updating algorithms in terms of the convergence and the value of network entropy, where the SMT algorithm with step size 0.1 is adopted for node payoff maximization. We have similar observations for the SMT algorithm with other step sizes because the step size has negligible effects on the final network entropy as shown in Fig. 5. The dynamics of network entropy is shown in Fig. 6. We can find that: 1) the depth-first and breadth-first updating algorithms reach a stable state faster (i.e., 8 rounds and 10 rounds, respectively) than the random-order updating algorithm (i.e., 26 rounds); and 2) the breadth-first and random-order updating algorithms are very unstable at the earlier stages than the depth-first updating algorithm. We can also find that the network entropy may get larger than the final steady state for the breadth-first and random updating algorithms. This is because the consumers' dissatisfaction can well propagate to suppliers by the depth-first updating algorithm, however, for the other two algorithms the penalty function $F(\cdot)$ may disturb network entropy during the updating process. However, all of them can finally approach to the same value of network entropy. This means that our network pricing mechanism has the potential to be extended to asynchronous updating algorithms, which may be more feasible in solving real-world distribution problems.

5) Effectiveness of the Decentralized Mechanism: We finally evaluate the effectiveness of the SMT algorithm together with our mechanism. Based on the analysis of Section IV-C, to verify the effectiveness of our pricing mechanism, we only need to verify that the SMT algorithm can perform as well as a centralized optimization algorithm at the local level. Fu *et al.* have shown that their pricing mechanism generates the same resource allocation as the centralized optimization algorithm [33]. Therefore, in this paper, to validate the effectiveness of the SMT algorithm, we compare the results of the SMT algorithm with that of the CMT algorithm, where the CMT algorithm is introduced in Appendix A based on Fu *et al.*'s work [33]. Table I shows the performance comparison between the SMT algorithm with step size $\gamma = 0.1$ and the CMT algorithm with step size $r = 20$ in terms of node player I_1 with four out-edges (i.e., Alabama-Florida, Alabama-Georgia, Alabama-Tennessee, and Self-consumption). Similar to the SMT algorithm, the CMT algorithm is also insensitive to step size (see Appendix B for detail), therefore, it is sufficient to carry out a comparison between the SMT algorithm with step size $\gamma = 0.1$ and the CMT algorithm with step size $r = 20$. The results show that the difference of allocated w_{ij} on each out-edge between the two algorithms is very small, i.e., less than 0.001 (the $threshold$). This indirectly verifies that the SMT algorithm together with the pricing mechanism can perform as well as a centralized optimization algorithm.

6) Discussion on the Differences Between the SMT and CMT Algorithms: The CMT algorithm (see Appendix A for detail) is designed to meet the budget balance criteria, i.e., all bids the edge players pay to the node player are allocated back to them. Based on the CMT algorithm, edge players buy/sell their resources (i.e., the commodity flows in this paper) among themselves through a third authority (i.e., the node player in this paper) to maximize the sum of their utilities, where the third authority is prevented from behaving as a profit maker. This means that the third authority has no resources to allocate to each edge player. However, for distribution networks (e.g., supply networks), each node has the power to distribute its resources. Therefore, although the CMT algorithm can solve the node payoff maximization problem, it is inappropriate to be adopted for our network pricing mechanism in distribution networks.

In addition, there are three other differences between the SMT and CMT algorithms: 1) Message exchange overhead: for the SMT algorithm, only one message (i.e., bid) is necessary to transmit from edge players to the node player, while for the CMT algorithm, two messages (i.e., w_{ij} and p_j) are needed to transmit; 2) Flow distribution ways: for the SMT algorithm, the node player determines the flows on each edge according to $w_{ij} = \kappa_j^i/\mu_i$, while for the CMT algorithm, the edge players themselves determine the flows by considering other players' demand d_{-j} and average price p_{-j}; and 3) The role of step size: for the SMT algorithm, the larger the step size γ is, the faster the algorithm converges to steady state, while for the CMT algorithm, the smaller the step size r is, the faster the algorithm converges to steady state.

VI. Conclusion and Future Work

In this paper, we have studied the problem of how to improve the functional robustness of a distribution network in a decentralized way. The functional robustness involves two issues: 1) to mitigate the possible disruptions, and 2) to meet all consumers' demand. To achieve this goal, we have addressed two important issues. First, we have derived a utility function, which integrates both the mitigation capability of the network

TABLE I
PERFORMANCE COMPARISON OF SMT ALGORITHM WITH $\gamma = 0.1$ AND CMT ALGORITHM WITH $r = 20$ IN TERMS OF NODE PLAYER I_1

	The SMT Algorithm			The CMT Algorithm		
	Allocated weight w_{ij}	Payoff ($\times 10^6$)	Step size γ	Allocated weight w_{ij}	Payoff ($\times 10^6$)	Step size r
Out-edge 1	0.1660829	0.8535251	0.1	0.1660856	0.8536107	20
Out-edge 2	0.4676097	2.4031170	0.1	0.4676392	2.4034440	20
Out-edge 3	0.2934432	1.5080490	0.1	0.2934670	1.5082650	20
Out-edge 4	0.0737818	0.3791760	0.1	0.0737879	0.3792268	20

and the demand satisfaction of all consumers together, to characterize the functional robustness of a distribution network. A notion of network entropy is defined in terms of the dynamics of commodity flows based on the Ruelle-Bowens random walk theory. By doing so, the problem of improving the functional robustness can be represented as a constrained utility optimization problem. To centrally solve the problem, there exist two obstacles: 1) complete information and full control of the distribution network may not be available, and 2) it may not be scalable for large-scale distribution networks. Therefore, we have further proposed a network pricing mechanism to solve the problem in a decentralized way. According to the mechanism, each node can adaptively adjust its distribution strategy by communicating only with its neighbors so that the functional robustness of the overall distribution network can be improved.

There are two levels of dynamics in our pricing mechanism. At the network level, each node communicates with its neighbors by sending a "price" signal to its upstream neighbors and receiving "price" signals from its downstream neighbors. Then at the local level, each node maximizes its payoff to determine its distribution strategy. We have implemented the mechanism with the following algorithms: 1) a SMT algorithm to iteratively maximize node payoff at the local level, and 2) three network updating algorithms, i.e., depth-first, breadth-first, and random updating algorithms, to determine the maximization order. Furthermore, we have mathematically proved that our decentralized mechanism can achieve results equivalent to those of a centralized optimization approach. Finally, we have also carried out a case study on the U.S. natural gas distribution network to evaluate the performance of our network pricing mechanism. The results show that: 1) by setting appropriate step sizes, the SMT algorithm can finally converge to a steady solution; 2) although the step size can significantly affect the convergence speed of the SMT algorithm, it has negligible effects on the final results; 3) the order (i.e., the depth-first, breadth-first, and random order) of node maximization does not affect the final results of network entropy; and 4) our mechanism can perform as well as a centralized optimization by comparing it with the CMT algorithm.

Although we have analyzed the convergence and effectiveness of the decentralized mechanism, there still remain some issues to be considered in the future:

- We assume that the nodes are price-taker, i.e., they do not anticipate how their prices will affect other nodes' decision during payoff maximization. What about the efficiency loss if without this assumption [35], [40]?
- We assume in this paper that all nodes have incentives to announce their true "prices." What kinds of incentive mechanism can motivate the nodes to reveal their true "prices"?
- We have only validated our mechanism in the case of the U.S. natural gas distribution network with respect to a specific penalty price function $f_i(\cdot)$. How will different penalty price functions affect the distribution robustness of a distribution network?

All these issues will be pursued in our future studies.

APPENDIX A
THE COUPLE MESSAGE TRANSMISSION ALGORITHM

Based on the pricing mechanism proposed by Fu *et al.*[33], we introduce the CMT algorithm in this section. At each iteration, each edge player L_j^i transmits to its corresponding node player two messages: a flow demand w_{ij} and a corresponding price p_j. After receiving messages from all edge players, the node player I_i conveys a couple of messages to each player, i.e., (p_{-j}, d_{-j}), where

$$p_{-j} = \frac{1}{m_i - 1} \sum_{k=1, k\neq j}^{m_i} p_k \tag{36}$$

$$d_{-j} = \frac{1}{m_i - 1} \sum_{k=1, k\neq j}^{m_i} w_{ik}. \tag{37}$$

Given $(p_{-j}(t), d_{-j}(t))$ of iteration t, each edge player determines its $p_j(t+1)$ and $w_{ij}(t+1)$ independently:

$$\begin{aligned} p_j(t+1) =& p_{-j}(t)\left(1 + \frac{d_{-j}(t) + w_{ij}(t)}{r}\right) \\ &+ \max\left\{0, \frac{d_{-j}(t) + w_{ij}(t)}{r}\right\} \end{aligned} \tag{38}$$

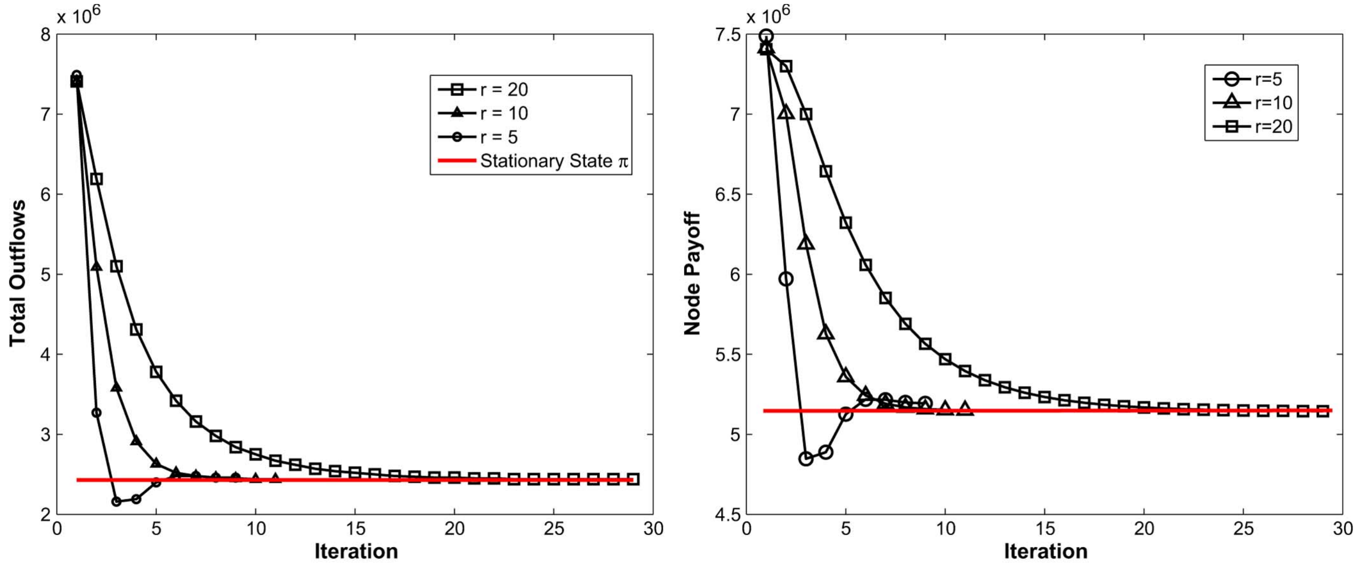
$$\begin{aligned} w_{ij}(t+1) =& \arg\max_{w_{ij}} \{U_{ij}(w_{ij}(t+1)) \\ &- \pi_i p_{-j}(t+1)(w_{ij}(t+1) - w_{ij}(t))\} \\ =& \exp\{\xi_j - p_{-j}(t+1) - 1\} \end{aligned} \tag{39}$$

where r is the step size of this algorithm.

By doing so, the node payoff maximization problem can be solved iteratively based on (38) and (39). The algorithm stops when the sum of $\mathbf{w}_i$ is close to 1.

APPENDIX B
PERFORMANCES OF THE CMT ALGORITHM

We evaluate the performances of the CMT algorithm in the case of the U.S. natural gas distribution network in this paper. The results show that with an appropriate step size r, the iterative process for all nodes are convergent. Fig. 7 demonstrates the convergence of total outflows and payoffs of node player I_1 with respect to different step sizes, i.e., $r = 5$, 10, and 20, respectively. We can find that the step size can significantly



(a) (b)

Fig. 7. Total outflows and payoffs of the CMT algorithm for node player I_1 with respect to different step sizes, i.e., $r = 5$, 10, and 20, respectively. Fig. 4(a) and (b) show that the smaller the step size is, the faster the steady state is reached. We can also find that a small step size (e.g., $r = 5$) leads to oscillation before reaching a steady state, which is consistent with the analysis in [33].

affect the speed of convergence, i.e., the smaller the step size is, the faster the algorithm converges. However, even with different step size, the algorithm can finally reach almost the same results (as shown in Fig. 7). We can also find that a small step size (e.g., $r = 5$) in the CMT algorithm may lead to oscillation before reaching a steady state. All these observations are consistent with the analysis in [33].

ACKNOWLEDGMENT

The authors would like to thank the editors and the anonymous reviewers for their constructive comments and suggestions.

REFERENCES

[1] J. Beck, R. Kempener, B. Cohen, and J. Petrie, “A complex systems approach to planning, optimization and decision making for energy networks,” *Energy Policy*, vol. 36, no. 8, pp. 2795–2805, Aug. 2008.

[2] A. Kumar, B. Faltings, and A. Petcu, “Distributed constraint optimization with structured resource constraints,” in *Proc. 8th Int. Conf. Autonomous Agents Multiagent Syst.*, Richland, SC, 2009, pp. 923–930.

[3] R. Shaw, M. Attree, and T. Jackson, “Developing electricity distribution networks and their regulation to support sustainable energy,” *Energy Policy*, vol. 38, no. 10, pp. 5927–5937, Oct. 2010.

[4] P. Burgherr and S. Hirschberg, “Severe accident risks in fossil energy chains: A compavative analysis,” *Energy*, vol. 33, no. 4, pp. 538–553, Apr. 2008.

[5] A. E. Motter, “Cascade control and defense in complex networks,” *Phys. Rev. Lett.*, vol. 93, no. 9, pp. 098701-1–098701-4, Aug. 2004.

[6] B. Wang, H. Tang, C. Guo, and Z. Xiu, “Entropy optimization of scale-free networks robustness to random failures,” *Phys. A*, vol. 363, no. 2, pp. 591–596, May 2006.

[7] R. V. Solé and S. Valverde, “Information theory of complex networks: On evolution and architectural constraints,” *Lecture Notes in Physics*, vol. 650, pp. 189–207, Sep. 2004.

[8] K. Anand and G. Bianconi, “Entropy measures for complex networks: Toward an information theory of complex topologies,” *Phys. Rev. E*, vol. 80, no. 4, pp. 045102-1–045102-4, Oct. 2009.

[9] S. Boccaletti, V. Latora, Y. Moreno, M. Chavezf, and D. Hwang, “Complex networks: Structure and dynamics,” *Phys. Rep.*, vol. 424, no. 4/5, pp. 175–308, 2006.

[10] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, “Network robustness and fragility: Percolation on random graphs,” *Phys. Rev. Lett.*, vol. 85, no. 25, pp. 5468–5471, Dec. 2000.

[11] R. Albert, H. Jeong, and A. Barabási, “Error and attack tolerance of complex networks,” *Nature*, vol. 406, no. 6794, pp. 378–382, Jul. 2000.

[12] A. Vázquez1 and Y. Moreno, “Resilience to damage of graphs with degree correlations,” *Phys. Rev. E*, vol. 67, no. 1, pp. 015101-1–015101-4, Jan. 2003.

[13] R. Cohen, K. Erez, D. ben Avraham, and S. Havlin, “Breakdown of the internet under intentional attack,” *Phys. Rev. Lett.*, vol. 86, no. 16, pp. 3682–3685, Apr. 2001.

[14] R. Guimerà, A. Arenas, A. Díaz-Guilera, and F. Giralt, “Dynamical properties of model communication networks,” *Phys. Rev. E*, vol. 66, no. 2, pp. 026704-1–026704-8, Aug. 2002.

[15] Y. Moreno, J. B. Gómez, and A. F. Pacheco, “Instability of scale-free networks under node-breaking avalanches,” *Europhys. Lett.*, vol. 58, no. 4, pp. 630–636, May 2002.

[16] D. P. Nedic, I. Dobson, D. S. Kirschen, B. A. Carreras, and V. E. Lynch, “Criticality in a cascading failure blackout model,” *Int. J. Elect. Power Energy Syst.*, vol. 28, no. 9, pp. 627–633, Nov. 2006.

[17] K. Zhao, A. Kumar, and J. Yen, “Achieving high robustness in supply distribution networks by rewiring,” *IEEE Trans. Eng. Manag.*, vol. 58, no. 2, pp. 347–362, May 2011.

[18] B. Shargel, H. Sayama, I. R. Epstein, and Y. Bar-Yam, “Optimization of robustness and connectivity in complex networks,” *Phys. Rev. Lett.*, vol. 90, no. 6, p. 068701, Feb. 2003.

[19] A. Beygelzimer, G. E. Grinstein, R. Linsker, and I. Rish, “Improving network robustness by edge modification,” *Phys. A, Stat. Mech. Appl.*, vol. 357, no. 3/4, pp. 593–612, Nov. 2005.

[20] J. Zhao and K. Xu, “Enhancing the robustness of scale-free networks,” *J. Phys. A, Math. Theor.*, vol. 42, no. 19, p. 195 003, May 2009.

[21] D. Liu, H. Wang, and P. V. Mieghem, “Spectral perturbation and reconstructability of complex networks,” *Phys. Rev. E*, vol. 81, no. 1, pp. 016101-1–016101-9, Jan. 2010.

[22] A. Tizghadam and A. Leon-Garcia, “Autonomic traffic engineering for network robustness,” *IEEE J. Sel. Areas Commun.*, vol. 28, no. 1, pp. 39–50, Jan. 2010.

[23] A. E. Motter and Z. Toroczkai, “Introduction: Optimization in networks,” *Chaos*, vol. 17, no. 2, pp. 026101-1–026101-3, Jun. 2007.

[24] A. Quelhas, E. Gil, J. D. McCalley, and S. M. Ryan, “A multiperiod generalized network flow model of the U.S. integrated energy system: Part I—Model description,” *IEEE Trans. Power Syst.*, vol. 22, no. 2, pp. 829–836, May 2007.

[25] A. Quelhas and J. D. McCalley, “A multiperiod generalized network flow model of the U.S. integrated energy system: Part II—Simulation results,” *IEEE Trans. Power Syst.*, vol. 22, no. 2, pp. 837–844, May 2007.

[26] B. An and V. Lesser, "Characterizing contract-based multiagent resource allocation in networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 575–586, Jun. 2010.

[27] K. M. Sim and B. Shi, "Concurrent negotiation and coordination for grid resource coallocation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 753–766, Jun. 2010.

[28] K.-S. Hwang, S.-W. Tan, M.-C. Hsiao, and C.-S. Wu, "Cooperative multiagent congestion control for high-speed networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 2, pp. 255–268, Apr. 2005.

[29] F. P. Kelly, "Charging and rate control for elastic traffic," *Eur. Trans. Telecommun.*, vol. 8, no. 1, pp. 33–37, Jan./Feb. 1997.

[30] S. Shakkottai and R. Srikant, "Network optimization and control," *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, Jan. 2007.

[31] Y. Masuda and S. Whang, "Capacity management in decentralized networks," *Manag. Sci.*, vol. 48, no. 12, pp. 1628–1634, Dec. 2002.

[32] E. Altman, T. Basar, and R. Srikant, "Nash equilibria for combined flow control and routing in networks: Asymptotic behavior for a large number of users," in *Proc. 38th IEEE Conf. Decision Control*, 1999, pp. 4002–4007.

[33] F. Fu, T. Stoenescu, and M. van der Schaar, "A pricing mechanism for resource allocation in wireless multimedia applications," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 2, pp. 264–279, Aug. 2007.

[34] J.-C. Delvenne and A.-S. Libert, "Centrality measures and thermodynamic formalism for complex networks," *Phys. Rev. E*, vol. 83, no. 4, pp. 046117-1–046117-7, Apr. 2011.

[35] R. Johari and J. N. Tsitsiklis, "Efficiency loss in a network resource allocation game," *Math. Oper. Res.*, vol. 29, no. 3, pp. 407–435, 2004.

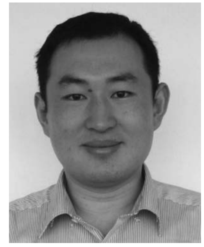
[36] D. P. Majoras, O. Swindle, T. B. Leary, P. J. Harbour, and J. Leibowitz, "Gasoline price changes: The dynamic of supply, demand, and competition," Federal Trade Commission, Washington, DC, Tech. Rep., 2005.

[37] L. Demetrius, V. M. Gundlach, and G. Ochs, "Complexity and demographic stability in population models," *Theor. Population Biol.*, vol. 65, no. 3, pp. 211–225, May 2004.

[38] L. Demetrius and T. Manke, "Robustness and network evolution: An entropic principle," *Phys. A, Stat. Mech. Appl.*, vol. 346, no. 3/4, pp. 682–696, Feb. 2005.

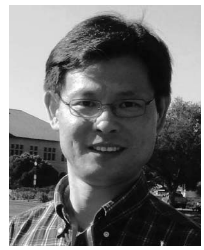
[39] J. Gómez-Gardeñes and V. Latora, "Entropy rate of diffusion processes on complex networks," *Phys. Rev. E*, vol. 78, no. 6, pp. 065102-1–065102-4, Dec. 2008.

[40] R. Johari and J. N. Tsitsiklis, "Efficiency of scalar-parameterized mechanisms," *Oper. Res.*, vol. 57, no. 4, pp. 823–839, Jul. 2009.



Benyun Shi received the M.Phil degree in computer science from Hong Kong Baptist University, Kowloon Tong, Hong Kong, in 2008. He is currently a Ph.D. student in the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong.

His main research interests include multi-agent autonomy-oriented computing, real-world complex systems/networks modeling and analysis particularly for energy distribution systems, and energy sustainability management.



Jiming Liu (F'11) received the B.Sc. degree from East China Normal University, Shanghai, China, M.A. degree from Concordia University, Montreal, QC, Canada, and M.Eng. and Ph.D. degrees in electrical engineering from McGill University, Montreal.

He is Associate Dean of Science (Research) and Chair Professor in Computer Science at Hong Kong Baptist University (HKBU). He was Professor and Director of School of Computer Science at University of Windsor, Canada. Before 1994, he held full-time R&D positions at Computer Research Institute of Montreal, Virtual Prototypes Inc., and Knowledge Engineering Tech. Inc. in Canada.

Dr. Liu's current research focuses on complex systems modeling, complex networks, web intelligence, and multi-agent autonomy-oriented computing. He has contributed to the scientific literature in those areas. Dr. Liu received the President's Award for Outstanding Performance in Scholarly Work at HKBU in 2007 and was named 2011 IEEE Fellow for contributions to Web Intelligence and Multi-Agent Autonomy Oriented Computing. Dr. Liu has served as an Editor-in-Chief of Web Intelligence and Agent Systems, an Associate Editor of IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART B, AND COMPUTATIONAL INTELLIGENCE, among others, and an Editorial Board Member of several international journals. He is the Chair of IEEE Computer Society Technical Committee on Intelligent Informatics and Codirector of Web Intelligence Consortium.