Probability in the Engineering and Informational Sciences, **31**, 2017, 516–539.
doi:10.1017/S0269964817000067

OPTIMUM ENERGY FOR ENERGY PACKET NETWORKS

Yonghua Yin
Intelligent Systems and Networks Group, Department of Electrical and Electronic Engineering, Imperial College, London SW7 2BT, UK
E-mail: y.yin14@imperial.ac.uk

The concept of Energy Packet Network (EPN) proposed by Gelenbe, is a new framework for modeling power grids that takes distributed energy generation such as renewable energy sources into consideration, and which contributes to modeling the smart grid. Based on G-network theory, this paper presents a simplified model of EPN and formulates energy-distribution as an optimization problem. We analyze it theoretically, and detail its optimal solutions. In addition to using existing optimization algorithms, a heuristic algorithm is proposed to solve for EPN optimization. The optimal solutions and efficacy of the algorithm are illustrated with numerical experiments. Further, we present an EPN with disconnections and a similar optimization problem is investigated. Optimal solutions are presented, and numerical results using the analytic optimal solutions, random solutions, a cooperative particle swarm optimizer and a heuristic algorithm illustrate the power of different approaches for solving energy-distribution problems using the EPN formalism.

Keywords: queueing theory, stochastic modeling

1. INTRODUCTION

The Smart Grid is a network of computers and power infrastructure that monitors and manages energy usage [47], and is emerging to become the next generation electricity grid [12]. An important goal of the smart grid is to integrate all forms of energy [49], including gas, heat, hydro power, solar power, off-shore wind, biomass energy, wave energy and many other combined distributed power, according to the future smart grid vision shown in [4]. Energy sources, such as the solar power, off-shore wind and wave energy, can be classified as renewable energy sources (RES). In terms of climate goals, RES are environment friendly and recent studies show that a smart grid incorporating RES can decrease annual CO_2 emissions by 5–16% [43]. In addition, RES are becoming commercially profitable in the energy markets, forcing the smart grid to include them. RES are becoming more plentiful [21] and electricity users may also utilize the RES to generate electricity and give it back into the grid, making the electricity storage in the grid extremely distributed. This distributed generation (DG) makes the energy market more feasibly [10,21]; however, controlling them can be much more complicated [50].

One challenge is that RES are very fluctuating and cannot always match electricity demands, making it difficult to keep the demand and supply in balance [48]. Much work related to DG systems has also been published. The concept of the virtual power plant (VPP) was introduced [7,46], which is a cluster of distributed generators, controllable loads and storages systems. The heart of the VPP is an energy management system that coordinates the power flows. A new grid concept, called microgrid, has also been developed to handle the DG [4,45]. The main idea of microgrid is that a localized self-contained energy network (namely, microgrid) can be disconnected and reconnected from the main grid (namely macrogrid). This disconnection (or say, islanded) mode of the microgrid has the potential to provide a higher local reliability by avoiding the faults coming from the macrogrid. The work by Hikihara in [50] focused on a small-scale power grid, that is, the in-home power grid, that also utilizes the RES and faces the DG issues. A novelty in [50] is that Hikihara proposed a DC power dispatching system via power packets that makes the control of DG easier. It means that, in the system, supply can be easily regulated by controlling the number of power packets. Hikihara did further research on this dispatching system and confirmed its feasibility in physical layer via experiments at the in-home DC network [51]. The work by Fang in [11] that is related to the microgrid first investigated the approach to find the most efficient and reliable power supply among a large group of distributed RES for a user in a microgrid and accordingly proposed a discovery approach to discover all the available RES within a microgrid, based on the assumption that the power can be intentionally delivered from a distributed generator to a user. It is reported that this assumption can be done by exploiting the above-mentioned work of Hikihara about the DC power dispatching system via power packets [50,51]. Then, Fang investigated the issue of how to find the power supply among the available RES that can deliver the highest profit and accordingly proposed two distributed algorithms based on the machine-learning algorithms in [2,3]. The work in [40] focuses on the power grid in buildings and formulated the minimization of energy cost with the significant uncertainties in the demand profile and RES energy supply considered into a scheduling problem. In addition, one of the world's major consumers of energy is Information Technology itself [5,29] which is estimated to consume 7–8% of the world's electricity production.

1.1. Energy Packet Networks (EPNs), the Random Neural Network (RNN) and G-networks

The concept of an EPN introduced by Erol Gelenbe [21,22] is a new framework for modeling smart power grids. Specifically, the EPN takes both the distributed energy generations and conventional sources of energy into consideration. It is a framework for both the macroscopic smart grid and micro-scale energy harvesting networks, and it is developed based on the theory of "Gelenbe Networks" (G-networks) for analytical modeling of complex systems [16, 32,36]. In this approach, the service by one energy packet of multiple jobs, or alternatively the consumption of multiple units of energy, can also be modeled, and its counterpart that includes the addition of energy or work is discussed in [13,32]. Multiple classes of work or energy addition and removal can also be modeled [14]. In recent work, the problem of adapting the flow of energy to dynamically varying demands was posed in the EPN framework [23] and the optimization of smart grid configurations and optimal flows of EPNs has been solved for various special instances [8,30,31].

The RNN [15] is a special case of G-networks. Its gradient-based learning algorithm [17] has been extended [35] to cover "multiple classes" that can represent several colors in an image, multi-sensory perception, or multiple flows of energy and workload. Soma-to-soma interactions between neurons, and not just interactions via synapses and dendrites, have also been modeled [39]. The RNN was used in real-time applications such as Packet Network Routing and Cloud Task Assignment with Reinforcement Learning [6,20,54]. Furthermore, RNN's application to deep learning yields very high recognition (as well as training) accuracy includes [41,42]. Note that the G-network model is significantly more general than the RNN [16,19,32] because it has state transitions with some arbitrary up or down jumps and it can model transitions that span any number of distinct nodes in the network [18].

1.2. EPNs and the Smart Grid

In the mathematical model for EPNs, the nodes of the EPN include energy sources, energy storage centers and energy consumption centers, where the energy sources can be distributed RES (i.e., the DG). The energy in the EPN is stored, distributed and consumed in the form of energy packets. In addition to the work in [21], the work by Hikihara presents the similar concept of "power packets" and verified the feasibility of a power packet dispatching system at the physical layer [50,51]. Then, the EPN can be regarded as a queueing network, the energy storage of the nodes are the queues, the energy packets are the regular customers in the queues and there may be control/request packets in the EPN corresponding to the signals (negative customers or triggers) in the G-networks. Using G-network theory [16, 32,36] in an approximate manner and certain assumptions, for example, unlimited storage capacity and energy packet generation and consumption submitting to Poisson process, the steady-state probabilities related to the nodes can be expressed mathematically by a system of equations. The purpose of the EPN in [21] is to meet the surges in energy demand in a grid, where the grid has both steady energy sources and distributed RES. In most of the time, the energy demand in the grid is satisfied by steady energy sources, but, in some instants, the energy demand may exceed the maximum level of steady energy and these excess energy requests may be met by the RES, which are managed by the EPN. Most concept used in the EPN model of [22] is similar to that of [21], but the EPN model is more specific and designed for the energy management for the Cloud computing servers, where the energy consumption centers become the Cloud computing centers. In the cases that scarce sources of energy must be shared by multi-computational units, the EPN can manage the energy in the storage centers, which is stored from DG when energy demand is not high, to best match and smooth the intermittent energy supply. Similarly, using G-network theory, the steady-state probabilities related to the nodes in the EPN can be approximately expressed by a system of equations. By analyzing the solutions of the equations, the approximate behaviors of the EPNs (in both [21] and [22]) can be analyzed, so that parameters in the EPN can be adjusted accordingly in order to achieve different objectives, for example, satisfying energy needs of energy consumption centers, or say, the Cloud computing centers.

Based on the G-network theory [16,32,36] and previous work on the EPN [21,22], this paper presents a simplified model of EPN that can be mathematically expressed by a systems of equations, whose nodes include distributed energy generators, energy storages and energy consumers. Then, we investigate the maximization of the amount of work done by the consumers per unit time in the EPN. This is an energy-distribution problem and we mathematically formulate it into an optimization problem. We analyze this problem based on the EPN model and present the analytic optimal solutions. Then, based on the learning algorithm for the RNN in [55], an heuristic algorithm is proposed to solve the problem. Then, we conduct numerical experiments to verify the correctness of analytic optimal solutions and demonstrate the efficacy of the proposed algorithm, compared with the random solutions and gradient-descent algorithm. Further, we present another model of EPN with disconnections based on the simplified EPN. A similar optimization problem is investigated. We present the analytic optimal solutions for two of three cases, and, for the rest case, we apply optimization algorithms to seeking for solutions. Comparative numerical results using the analytic optimal solutions, random solutions, a cooperative particle swarm optimizer and the proposed heuristic algorithm further demonstrate the superiority of the proposed algorithm for energy distribution of the EPNs.

1.3. E-Networks

Another stream of recent work launched by Erol Gelenbe to model energy consumption in wired and wireless networks, which we will call E-networks ("Erol Networks") is motivated by the need to model communication systems that operate with renewable energy [33,34]. In such systems, communications require both processing of the incoming packets and their wireless transmission. This can only occur if a node has enough energy; interesting cases arise when one considers "Cognitive Radio" where energy is needed both to sense a channel, decide about a transmission and then use significant energy to transmit a packet whose transmission may actually fail due to collisions, interference or noise [38].

On the other hand, the time scales for energy harvesting are slow, while the packet transmission times are very fast. Thus, mixed continuous-discrete models based on diffusion approximations, where transmissions and packet arrivals are instantaneous, while energy harvesting is a continuous process [1] are very useful. However, models that consider jump processes and discrete Markov chains can also provide elegant closed-form results [24,26].

In such systems, it is important to consider what happens between multiple hops. Indeed packets can travel over several hops [37], while energy remains local unless it is harvested by wireless means. Similarly, there can be an imbalance between the number of energy units (or packets) needed for transmission, and the number of packets that can be transmitted with one or more energy packets [44]. Generally, this is still in its infancy, especially when one considers that the carriers of information may be nano-particles, which are subject to quantum effects [25,27,28].

2. ENERGY DISTRIBUTION FOR A SIMPLIFIED ENERGY PACKET NETWORK

First, the simplified mathematic model of energy packet network and a related energy-distribution problem are presented. We then solve this energy-distribution problem in three ways: (1) via theoretical analyses, we present the optimal solutions; (2) we deduce a gradient-descent algorithm; and (3) a heuristic algorithm is developed. Last, we demonstrate the correctness of the optimal solutions and the efficacy of the proposed heuristic algorithm by comparative numerical results of the optimal solutions, random solutions, gradient-descent algorithm and heuristic algorithm. Note that the reason why we present the heuristic algorithm in this section even when the analytic optimal solutions have found is to demonstrate the efficacy of the heuristic algorithm such that we have better confidence that the algorithm is capable of finding good solutions in the case where it is difficult to deduce analytic optimal solutions, as illustrated in Section 3.

2.1. Mathematical Model and Problem Description

The simplified EPN consists of G energy generators, S energy storage units and C energy consumers. For better illustration, the model structure of the simplified EPN is presented in Figure 1. The units of the EPN interact with each other in the following manners, where $g = 1, \ldots, G$, $s = 1, \ldots, S$ and $c = 1, \ldots, C$.

1. The gth energy generator generates energy in a Poisson stream of rate Λ_g, which models the stochastic energy arrival of RES.
2. When an energy packet is generated from the gth energy generator, it is sent to all energy storages and consumers. The sent energy packet may go to the sth storage with probability $\hat{p}_{g,s}$ or the cth consumer with probability $\tilde{p}_{g,c}$. Here $\sum_{c=1}^{C} \tilde{p}_{g,c} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$.
3. The sth energy storage sends energy to all storage consumers in a Poisson stream of rate δ_s when it has energy, where the sent energy packet may go to the cth consumer with probability $p_{s,c}$. Here $\sum_{c=1}^{C} p_{s,c} = 1$.
4. Energy also leaks from the sth storage in a Poisson stream of rate γ_s when the storage has energy.
5. The cth consumer consumes energy in a Poisson stream of rate μ_c when it has energy.

Let $\hat{q}_s$ and q_c denote the stationary probabilities that the sth storage has energy in storage and cth consumer has energy to consume, respectively. According to the G-network theory [16,21,22,32,36], we can mathematically present the expressions of $\hat{q}_s$ and q_c as the following system of equations:

$$\hat{q}_s = \min\left(\frac{\sum_{g=1}^{G} (\Lambda_g \hat{p}_{g,s})}{\gamma_s + \sum_{c=1}^{C} (\delta_s p_{s,c})}, 1\right), \quad q_c = \min\left(\frac{\sum_{g=1}^{G} (\Lambda_g \tilde{p}_{g,c}) + \sum_{s=1}^{S} (\hat{q}_s \delta_s p_{s,c})}{\mu_c}, 1\right), \tag{1}$$

where $\forall s \in S$ and $\forall c \in C$. In addition, $\sum_{c=1}^{C} p_{s,c} = 1$ and $\sum_{c=1}^{C} \tilde{p}_{g,c} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$.

2.1.1. Energy-distribution problem. The energy-distribution problem of the EPN (1) we consider is described as follows. We want to maximize the amount of work done by the consumers per unit time in the EPN (1), or equivalently: given $\{\Lambda_g | g = 1, \ldots, G\}$, $\{\mu_c | c = 1, \ldots, C\}$, $\{\delta_s | s = 1, \ldots, S\}$ and $\{\gamma_s | s = 1, \ldots, S\}$,

$$\begin{aligned} &\text{maximize } K = \sum_{c=1}^{C} (\mu_c q_c) \text{ using } \{\hat{p}_{g,s}\}, \{\tilde{p}_{g,c}\}, \{p_{s,c}\}, \\ &\text{over} \quad 1 \le g \le G,\ 1 \le s \le S,\ 1 \le c \le C. \end{aligned} \tag{2}$$

2.2. Optimal Solutions for Energy Distribution

This subsection analyzes the energy-distribution problem (2) of the EPN (1) in theory and presents analytical optimal solutions. We consider the problem from two cases. One case is that the energy is limited. The other is that the energy is sufficient.

2.2.1. The energy is limited. In this case, the energy is limited such that $\sum_{g=1}^{G} \Lambda_g \le \sum_{c=1}^{C} \mu_c$. This means that the total energy-consumption rate denoted by $H = \sum_{c=1}^{C} \mu_c$ is larger than the total energy-generation rate $\sum_{g=1}^{G} \Lambda_g$. We want to make use of all energy and do not want the energy to go to the energy storages because the energy will leak without doing anything useful. So, it is reasonable to set $\hat{p}_{g,s} = 0, \forall g \in G, s \in S$. Since

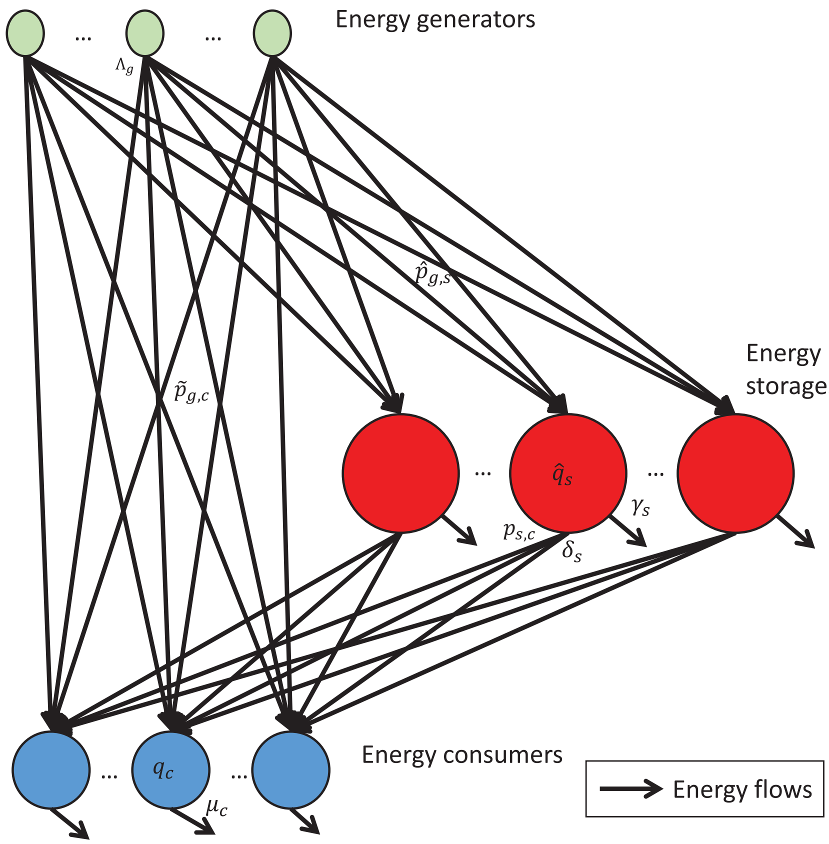

FIGURE 1. Brief model structure of the simplified EPN.

$\sum_{c=1}^{C} \tilde{p}_{g,c} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$, then $\sum_{c=1}^{C} \tilde{p}_{g,c} = 1$. From (1), we have $\hat{q}_s = 0$. Then,

$$\hat{q}_s = 0, q_c = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c})}{\mu_c}, 1\right). \tag{3}$$

Setting $\tilde{p}_{g,c} = \mu_c/H, \forall c \in C$. Then,

$$q_c = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \mu_c/H)}{\mu_c}, 1\right) = \min\left(\frac{\sum_{g=1}^{G}\Lambda_g}{H}, 1\right). \tag{4}$$

Then, we have $q_{c_1} = q_{c_2}, \forall c_1, c_2 \in C$. In addition, we also have $q_c \leq 1, \forall c \in C$ since $\sum_{g=1}^{G} \Lambda_g \leq H = \sum_{c=1}^{C} \mu_c$. Also,

$$K = \sum_{c=1}^{C}(\mu_c q_c) = \sum_{c=1}^{C}\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c}) = \sum_{g=1}^{G}\Lambda_g. \tag{5}$$

We can see that all energy has been fully used (no energy is wasted). Based on the law of conservation of energy, we could say that $K^* = \sum_{g=1}^{G} \Lambda_g$ is the optimal result for the maximization problem (2), where **the optimal solution is** $\hat{p}_{g,s} = 0$ **and** $\tilde{p}_{g,c} = \mu_c/H$, $\forall g \in G, s \in S, c \in C$. **In addition,** $p_{s,c}$ $\forall s \in S, c \in C$ **can be any non-negative value satisfying** $\sum_{c=1}^{C} p_{s,c} = 1$.

2.2.2. The available energy covers the needs. In this case, $\sum_{g=1}^{G}\Lambda_g > H = \sum_{c=1}^{C}\mu_c$. Let us set $\hat{p}_{g,s} = 0$ and $\tilde{p}_{g,c} = \mu_c/H$, $\forall g \in G, s \in S, c \in C$. From (3) and (4), we have

$$\hat{q}_s = 0,\ \ q_c = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c})}{\mu_c}, 1\right) = \min\left(\frac{\sum_{g=1}^{G}\Lambda_g}{H}, 1\right). \tag{6}$$

Since $\sum_{g=1}^{G}\Lambda_g > H$, in this case, the energy is sufficient such that $q_c = 1, \forall c \in C$, meaning the system is saturated. Then, $K = \sum_{c=1}^{C}(\mu_c q_c) = \sum_{c=1}^{C}\mu_c$. Although much energy is wasted, $K^* = \sum_{c=1}^{C}\mu_c$ is the optimal result for the maximization problem (2) in this case, where **the optimal solution is also $\hat{p}_{g,s} = 0$ and $\tilde{p}_{g,c} = \mu_c/H$, $\forall g \in G, s \in S, c \in C$. In addition, $p_{s,c}$ $\forall s \in S, c \in C$ can be any non-negative value satisfying $\sum_{c=1}^{C} p_{s,c} = 1$.**

2.3. Gradient-Descent Algorithm

We can also design a gradient-descent algorithm to solve the maximization problem (2).

First, let us define $P \in R^{(SC+GC+GS)\times 1}$ as a vector that consists of $\hat{p}_{g,s}$, $\tilde{p}_{g,c}$ and $p_{s,c}$, where

$$P(h_{s,c}) = p_{s,c}, \quad P(\tilde{h}_{g,c}) = \tilde{p}_{g,c}, \quad P(\hat{h}_{g,s}) = \hat{p}_{g,s}, \tag{7}$$

with $h_{s,c} = (s-1)C + c$, $\tilde{h}_{g,c} = SC + (g-1)C + c$ and $\hat{h}_{g,s} = SC + GC + (g-1)S + s$. Then, we need to derive the expression of $\partial K/\partial P$ such that a P-update formula can be derived.

To simplify the derivation, we first ignore the constraints of P, which are $P \geq 0$, $\sum_{c=1}^{C} p_{s,c} = 1$ and $\sum_{c=1}^{C}\tilde{p}_{g,c} + \sum_{s=1}^{S}\hat{p}_{g,s} = 1$. In addition, we also assume that $\hat{q}_s < 1$ and $q_c < 1$ such that the EPN (1) can be rewritten as

$$\hat{q}_s = \frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s + \sum_{c=1}^{C}(\delta_s p_{s,c})} = \frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s + \delta_s}, \quad q_c = \frac{\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c}) + \sum_{s=1}^{S}(\hat{q}_s \delta_s p_{s,c})}{\mu_c}. \tag{8}$$

Then,

$$\frac{\partial \hat{q}_{\hat{s}}}{\partial p_{s,c}} = 0, \quad \frac{\partial \hat{q}_{\hat{s}}}{\partial \tilde{p}_{g,c}} = 0, \quad \forall \hat{s} \in S; \tag{9}$$

$$\frac{\partial \hat{q}_{\hat{s}}}{\partial \hat{p}_{g,s}} = \frac{\Lambda_g}{\gamma_s + \delta_s}, \quad \text{if } \hat{s} = s; \ \frac{\partial \hat{q}_{\hat{s}}}{\partial \hat{p}_{g,s}} = 0, \quad \text{if } \hat{s} \neq s. \tag{10}$$

Then,

$$\frac{\partial(\mu_{\hat{c}} q_{\hat{c}})}{\partial p_{s,c}} = \frac{\partial\left(\sum_{\hat{s}=1}^{S}(\hat{q}_{\hat{s}}\delta_{\hat{s}} p_{\hat{s},\hat{c}})\right)}{\partial p_{s,c}} = \hat{q}_s\delta_s, \quad \text{if } \hat{c} = c; \ \frac{\partial(\mu_{\hat{c}} q_{\hat{c}})}{\partial p_{s,c}} = 0, \quad \text{if } \hat{c} \neq c; \tag{11}$$

$$\frac{\partial(\mu_{\hat{c}} q_{\hat{c}})}{\partial \tilde{p}_{g,c}} = \frac{\partial\left(\sum_{\hat{g}=1}^{G}(\Lambda_{\hat{g}} \tilde{p}_{\hat{g},\hat{c}})\right)}{\partial \tilde{p}_{g,c}} = \Lambda_g, \quad \text{if } \hat{c} = c; \ \frac{\partial(\mu_{\hat{c}} q_{\hat{c}})}{\partial \tilde{p}_{g,c}} = 0, \quad \text{if } \hat{c} \neq c; \tag{12}$$

$$\frac{\partial(\mu_{\hat{c}} q_{\hat{c}})}{\partial \hat{p}_{g,s}} = \frac{\partial\left(\sum_{\hat{s}=1}^{S}(\hat{q}_{\hat{s}}\delta_{\hat{s}} p_{\hat{s},\hat{c}})\right)}{\partial \hat{p}_{g,s}} = \sum_{\hat{s}=1}^{S}\left(\delta_{\hat{s}} p_{\hat{s},\hat{c}} \frac{\partial \hat{q}_{\hat{s}}}{\partial \hat{p}_{g,s}}\right) = \delta_s p_{s,\hat{c}} \frac{\partial \hat{q}_s}{\partial \hat{p}_{g,s}} = \frac{\delta_s p_{s,\hat{c}} \Lambda_g}{\gamma_s + \delta_s}, \quad \forall \hat{c} \in C. \tag{13}$$

Then,

$$\frac{\partial K}{\partial p_{s,c}} = \frac{\partial\left(\sum_{\hat{c}=1}^{C}(\mu_{\hat{c}}q_{\hat{c}})\right)}{\partial p_{s,c}} = \frac{\partial\left(\mu_c q_c\right)}{\partial p_{s,c}} = \hat{q}_s\delta_s, \quad \frac{\partial K}{\partial \tilde{p}_{g,c}} = \frac{\partial\left(\sum_{\hat{c}=1}^{C}(\mu_{\hat{c}}q_{\hat{c}})\right)}{\partial \tilde{p}_{g,c}} = \frac{\partial\left(\mu_c q_c\right)}{\partial \tilde{p}_{g,c}} = \Lambda_g, \tag{14}$$

$$\frac{\partial K}{\partial \hat{p}_{g,s}} = \frac{\partial\left(\sum_{\hat{c}=1}^{C}(\mu_{\hat{c}}q_{\hat{c}})\right)}{\partial \hat{p}_{g,s}} = \sum_{\hat{c}=1}^{C}\left(\frac{\partial\left(\mu_{\hat{c}}q_{\hat{c}}\right)}{\partial \hat{p}_{g,s}}\right) = \sum_{\hat{c}=1}^{C}\left(\frac{\delta_s p_{s,\hat{c}}\Lambda_g}{\gamma_s+\delta_s}\right). \tag{15}$$

Then, the P-update formula can be

$$\begin{aligned} p_{s,c}^{(l+1)} &= p_{s,c}^{(l)} + \eta\frac{\partial K}{\partial p_{s,c}} = p_{s,c}^{(l)} + \eta\hat{q}_s\delta_s, \quad \tilde{p}_{g,c}^{(l+1)} = \tilde{p}_{g,c}^{(l)} + \eta\frac{\partial K}{\partial \tilde{p}_{g,c}} = \tilde{p}_{g,c}^{(l)} + \eta\Lambda_g, \\ \hat{p}_{g,s}^{(l+1)} &= \hat{p}_{g,s}^{(l)} + \eta\frac{\partial K}{\partial \hat{p}_{g,s}} = \hat{p}_{g,s}^{(l)} + \eta\sum_{c=1}^{C}\left(\frac{\delta_s p_{s,c}\Lambda_g}{\gamma_s+\delta_s}\right), \end{aligned} \tag{16}$$

where l denotes the current number of iterations and $\eta > 0$ is the step size. After each iteration, we make P satisfy constraints $P \geq 0$, $\sum_{c=1}^{C} p_{s,c} = 1$ and $\sum_{c=1}^{C}\tilde{p}_{g,c} + \sum_{s=1}^{S}\hat{p}_{g,s} = 1$ with minimal adjustments via linear normalization or other techniques.

Remark 1: Since we do not integrate the constraints of P into the derivation of the expression of $\partial K/\partial P$, the solutions found by the P-update formula (16) may not be optimal.

2.4. A Heuristic Algorithm

Hoping to find acceptable solutions to the maximization problem (2), we design a heuristic algorithm by adapting the learning algorithm for the RNN in [55] that is developed based on [52], where the idea of the algorithm is to solve a non-negative least-squares problem. First, let us find an approximate problem, which is easier to solve, to the maximization problem (2). In problem (2), we want to maximize $K = \sum_{c=1}^{C}(\mu_c q_c)$. Equivalently, we want to minimize $-K = -\sum_{c=1}^{C}(\mu_c q_c)$. Equivalently, we want to minimize $V - K = V - \sum_{c=1}^{C}(\mu_c q_c)$, where $V > 0$ is a constant. Let $V = \sum_{c=1}^{C}\mu_c$. Then, equivalently, we can minimize $\sum_{c=1}^{C}(\mu_c - \mu_c q_c)$. The approximate problem to problem (2) is that we want $\mu_c q_c$ to be as close to μ_c as possible (or q_c to be as close to 1 as possible) using $P \in R^{(SC+GC+GS)\times 1}$.

We show in the followings how to apply the idea in [55] to design a heuristic algorithm for solving the approximate problem. Let us make a reasonable assumption that $\hat{q}_s < 1$ and $q_c < 1$ such that the EPN (1) can be rewritten as (8). Let us rewrite (8) in the following form:

$$\sum_{c=1}^{C}(\hat{q}_s\delta_s p_{s,c}) - \sum_{g=1}^{G}(\Lambda_g\hat{p}_{g,s}) + \hat{q}_s\gamma_s = 0, \quad \sum_{s=1}^{S}(\hat{q}_s\delta_s p_{s,c}) + \sum_{g=1}^{G}(\Lambda_g\tilde{p}_{g,c}) - q_c\mu_c = 0. \tag{17}$$

If we substitute q_c with 1 and $\hat{q}_s$ with given parameters $\hat{y}_s$ into (17), the equality between the left and right sides of (17) no longer holds (in most cases). To approach equality as much as possible, we first define a cost function $f_{\hat{Y},P} = \|AP - b\|_2^2/2$ based on the left side of (17), where $\hat{Y} = [\hat{y}_1\ \hat{y}_2\ \dots\ \hat{y}_S]$, $A \in R^{(S+C)\times(SC+GC+GS)}$ and $b \in R^{(S+C)\times 1}$ are defined as

$$A(\hat{\underline{h}}_s, h_{s,c}) = \hat{y}_s\delta_s, \quad A(\hat{\underline{h}}_s, \hat{h}_{g,s}) = -\Lambda_g, \quad A(\hat{\underline{h}}_s, \text{otherwise}) = 0, \quad A(\underline{h}_c, h_{s,c}) = \hat{y}_s\delta_s,$$
$$A(\underline{h}_c, \tilde{h}_{g,c}) = \Lambda_g, \quad A(\underline{h}_c, \text{otherwise}) = 0, \quad b(\hat{\underline{h}}_s) = -\hat{y}_s\gamma_s, \quad b(\underline{h}_c) = \mu_c, \tag{18}$$

with $\hat{\underline{h}}_s = s$, $\underline{h}_c = S + c$, $h_{s,c} = (s-1)C + c$, $\tilde{h}_{g,c} = SC + (g-1)C + c$ and $\hat{h}_{g,s} = SC + GC + (g-1)S + s$. We solve the following problem, hoping that the solutions could also be good solutions to the approximate problem and problem (2):

$$\text{minimize}_{P,\hat{Y}}\ f_{\hat{Y},P} = \frac{1}{2}\|AP - b\|_2^2, \quad \text{subject to } 0 \le \hat{Y} \le 1,$$
$$P \ge 0, \quad \sum_{c=1}^{C} p_{s,c} = 1, \quad \sum_{c=1}^{C} \tilde{p}_{g,c} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1. \tag{19}$$

The heuristic learning algorithm for solving (19) is given in Algorithm 1, where $P^{(m)}$ denotes the values of P in the mth iteration, $\nabla f_{\hat{Y},P^{(m)}} = A^{\mathrm{T}}(AP^{(m)}) - A^{\mathrm{T}}b$, $\eta > 0$ is the step size and the operation Constraint$[P]$ makes P satisfy the corresponding constraints with minimal adjustments via linear normalization or other techniques.

Algorithm 1. A heuristic algorithm for solving problem (2)

Initialize $P^{(1)}$ and $\hat{Y}$; $l \leftarrow 1$; $\vartheta \leftarrow 2$;
while $l \le L$ **do**
  Construct A and b; $\eta \leftarrow 1.5$; $m \leftarrow 1$;
  $\eta_{\text{new}} \leftarrow \eta$, $P_{\text{new}} \leftarrow P^{(1)}$, $f_{\text{new}} \leftarrow f_{\hat{Y},P^{(1)}}$;
  while $m \le M$ **do**
    $P^{(m+1)} \leftarrow \text{Constraint}[P^{(m)} - \eta\nabla f_{\hat{Y},P^{(m)}}]$;
    while $f_{\hat{Y},P^{(m+1)}} < f_{\text{new}}$ **do**
      $\eta_{\text{new}} \leftarrow \eta$; $P_{\text{new}} \leftarrow P^{(m+1)}$; $f_{\text{new}} \leftarrow f_{\hat{Y},P^{(m+1)}}$;
      $\eta \leftarrow \eta\vartheta$;
      $P^{(m+1)} \leftarrow \text{Constraint}[P^{(m)} - \eta\nabla f_{\hat{Y},P^{(m)}}]$;
    end while
    while $f_{\hat{Y},P^{(m+1)}} > f_{\text{new}} \& \eta > 10^{-20}$ **do**
      $\eta \leftarrow \eta/\vartheta$;
      $P^{(m+1)} \leftarrow \text{Constraint}[P^{(m)} - \eta\nabla f_{\hat{Y},P^{(m)}}]$;
      if $f_{\hat{Y},P^{(m+1)}} < f_{\text{new}}$ **then**
        $\eta_{\text{new}} \leftarrow \eta$; $P_{\text{new}} \leftarrow P^{(m+1)}$; $f_{\text{new}} \leftarrow f_{\hat{Y},P^{(m+1)}}$;
      end if
    end while
    $\eta \leftarrow \eta_{\text{new}}$; $P^{(m+1)} \leftarrow P_{\text{new}}$; $f_{\hat{Y},P^{(m+1)}} \leftarrow f_{\text{new}}$;
    $m \leftarrow m + 1$;
  end while
  solve (1) using P_{new} for $\hat{q}_s, \forall s \in S$;
  $P^{(1)} \leftarrow P_{\text{new}}$; $\hat{y}_s \leftarrow \hat{q}_s, \forall s \in S$;
end while
get P_{new} as the final solution.

Table 1. K^* and the best K found by using analytical optimal solutions and different optimization algorithms for energy distribution of the EPN (1) with different numbers of generators G, storages S and consumers C.

(G,S,C)	K^*	Optimal	Random	Gradient	Heuristic
$(2, 2, 2)$	**0.1500**	**0.1500**	0.1496	0.1417	**0.1500**
$(10, 5, 16)$	**1.0931**	**1.0931**	1.0756	1.0684	**1.0931**
$(10, 10, 30)$	**1.0976**	**1.0976**	1.0702	1.0755	**1.0976**
$(20, 10, 60)$	**1.8948**	**1.8948**	1.8565	1.8634	**1.8948**
$(100, 100, 300)$	**9.0363**	**9.0363**	–	8.7008	**9.0363**

2.5. Numerical Results

The following numerical experiments are conducted to verify the analytic optimal solutions in Section 2.2 and test the performance of the gradient and heuristic algorithms in Sections 2.3 and 2.4. All results in this subsection are summarized into Table 1. In addition, the symbol "–" in the tables of this paper means that the result cannot be obtained due to hardware limitations or high complexity of analysis. Note that we consider only the case that the energy is limited in this subsection. In the whole paper, all numerical experiments are conducted in a MATLAB R2014a environment, which is operated on a personal computer (CPU: Intel i7-4770 3.40 GHz; memory: 8.00 GB).

2.5.1. $G = 2$, $S = 2$ and $C = 2$. Let us consider an example of the EPN (1) with 2 generators ($G = 2$), 2 storages ($S = 2$) and 2 consumers ($C = 2$), which is shown in Figure 2, where $\Lambda_1 = 0.07$, $\Lambda_2 = 0.08$, $\mu_1 = 1$, $\mu_2 = 10$, $\delta_1 = 0.19$, $\delta_2 = 0.2$, $\gamma_1 = 0.01$ and $\gamma_2 = 0.05$.

First, we use the analytic optimal solution in Section 2.2.1. The numerical result $K = 0.15$, which is the same as the analytic optimal result $K^* = 0.15$. Since the dimension of P is small, we can randomly select the values of P for many trials (e.g., 20,000 trials) to find a nearly optimal solution. Figure 3 presents the values of K in these 20,000 trials. The best result is $K = 0.1496$. Figure 4 presents the results by using the gradient algorithm, where the number of iterations is 100. The best result is $K = 0.1417$, which is close to the optimal result $K^* = 0.15$. Figure 5 presents the results by using the heuristic algorithm, where $M = 5$ and $L = 20$. The best result is $K = 0.15$, which is the same as the optimal result $K^* = 0.15$.

2.5.2. $G = 10$, $S = 5$ and $C = 16$. Let us consider an EPN (1) with 10 generators ($G = 10$), 5 storages ($S = 5$) and 16 consumers ($C = 16$), where Λ_g, δ_s and γ_s with $\forall g \in G, s \in S$ are randomly generated in the ranges of $[0, 0.2]$, $[0, 1]$ and $[0, 0.1]$, respectively. In addition, we assume that the consumers differ from each other significantly. So, we set $\mu_c = 1 + 9(c-1)/(C-1)$ with $c = 1, \ldots, C$. First, we use the analytic optimal solution in Section 2.2.1. The numerical result $K = 1.0931$, which is the same as the analytic optimal result $K^* = \sum_{g=1}^{10} \Lambda_g$. We then randomly select the values of P for 20,000 trials. However, since the dimension of P is not small, a nearly-optimal solution may not be found. In these 20,000 trials, the best result is $K = 1.0756$. Figure 6 presents the results by using the gradient-descent algorithm, where the number of iterations is 100. The best result is $K = 1.0684$. Figure 7 presents the results by using the heuristic algorithm, where $M = 5$ and $L = 20$. The best result is $K = 1.0931$, which is the same as K^*.

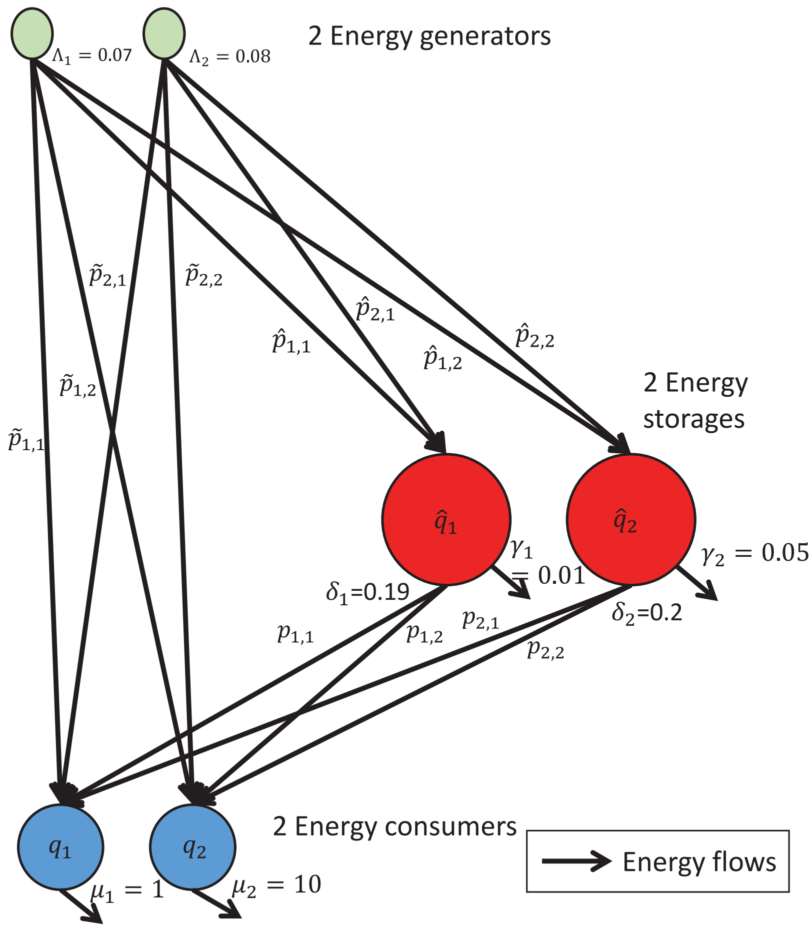

FIGURE 2. An example of EPN (1) with two generators ($G = 2$), two storages ($S = 2$) and two consumers ($C = 2$).

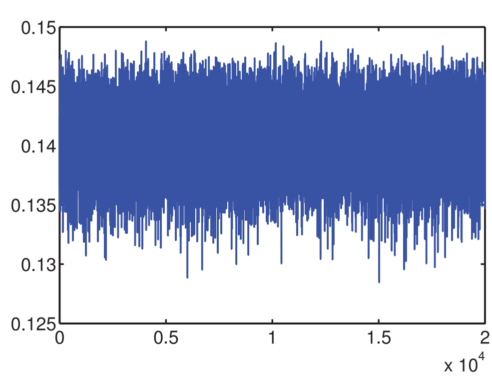

FIGURE 3. Values of K of the EPN (1) with $G = 2$, $S = 2$ and $C = 2$ using randomly generated P for energy distribution.

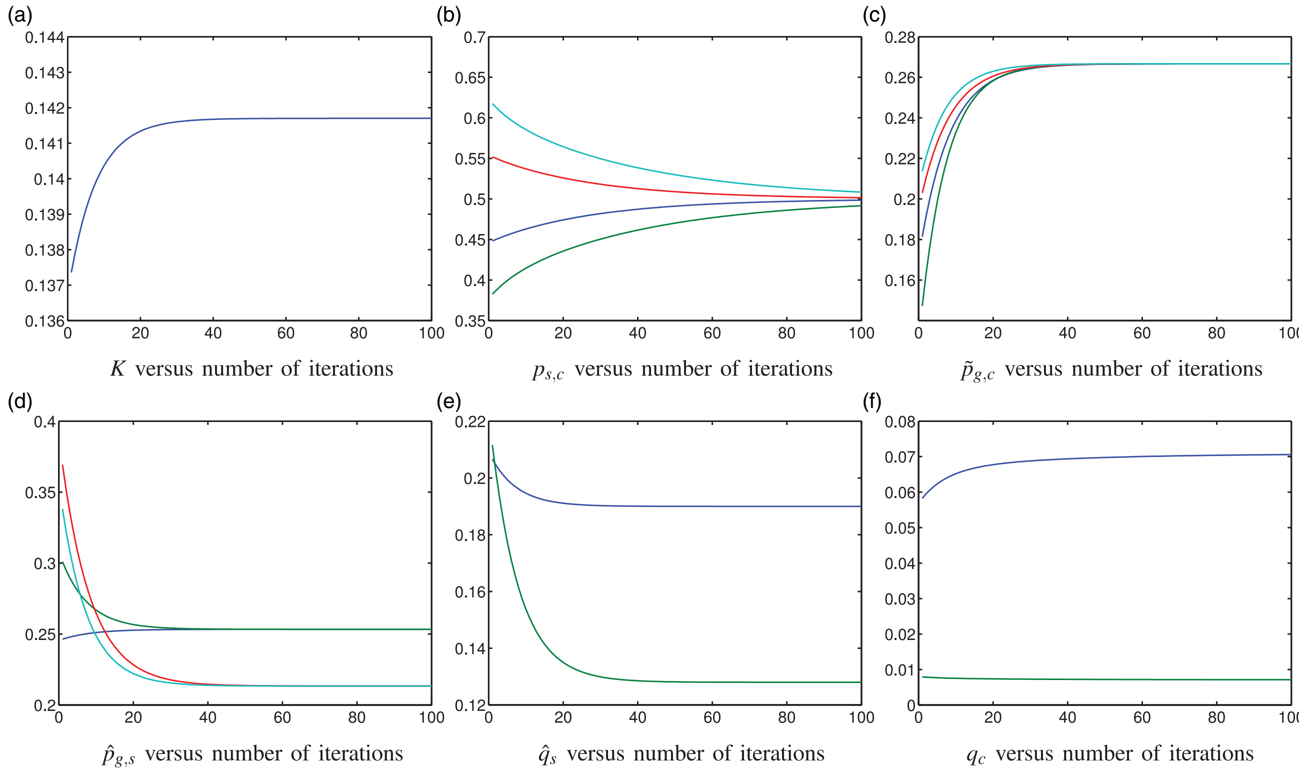

FIGURE 4. Performance of the gradient-descent algorithm for energy distribution of the EPN (1) with $G = 2$, $S = 2$ and $C = 2$. (a) K versus number of iterations, (b) $p_{s,c}$ versus number of iterations, (c) $\tilde{p}_{g,c}$ versus number of iterations, (d) $\hat{p}_{g,s}$ versus number of iterations, (e) $\hat{q}_s$ versus number of iterations, (f) q_c versus number of iterations.

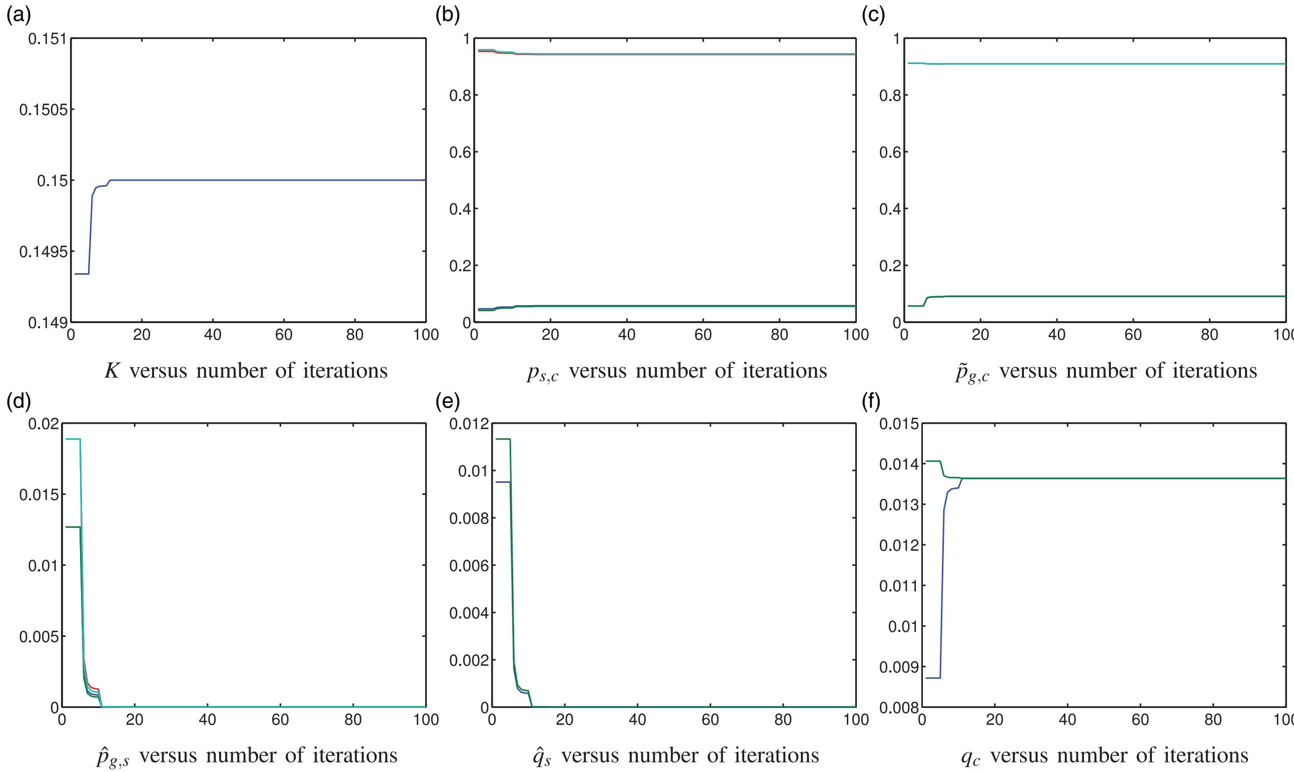

FIGURE 5. Performance of the heuristic algorithm 1 for energy distribution of the EPN (1) with $G = 2$, $S = 2$ and $C = 2$. (a) K versus number of iterations, (b) $p_{s,c}$ versus number of iterations, (c) $\tilde{p}_{g,c}$ versus number of iterations, (d) $\hat{p}_{g,s}$ versus number of iterations, (e) $\hat{q}_s$ versus number of iterations, (f) q_c versus number of iterations.

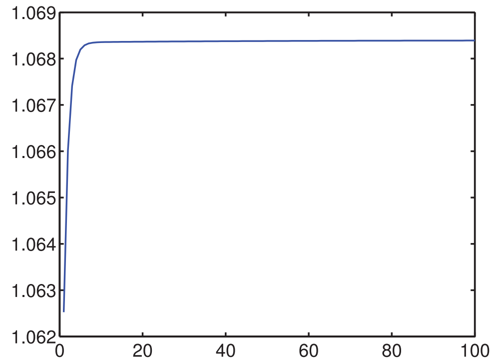

FIGURE 6. Values of K of the EPN (1) with $G = 10$, $S = 5$ and $C = 16$ using the gradient-descent algorithm for energy distribution.

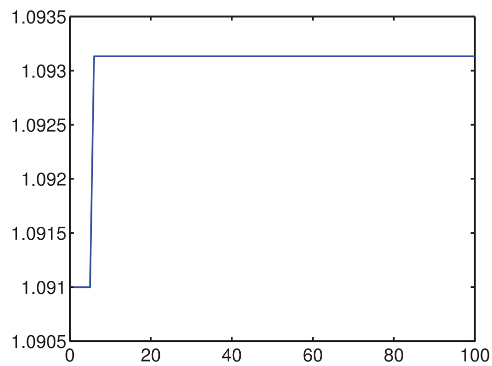

FIGURE 7. Values of K of the EPN (1) with $G = 10$, $S = 5$ and $C = 16$ using the heuristic algorithm 1 for energy distribution.

More results are given in Table 1. In all cases except the case of $G = 2$, $S = 2$ and $C = 2$, Λ_g, δ_s and γ_s with $\forall g \in G, s \in S$ are randomly generated in the ranges of $[0, 0.2]$, $[0, 1]$ and $[0, 0.1]$, respectively, while $\mu_c = 1 + 9(c - 1)/(C - 1), \forall c \in C$. We can see that, among the algorithms, the the heuristic algorithm (Algorithm 1) performs the best even when the network size becomes as large as 500. In summary, to manage energy distribution of the EPN (1) in order to maximize K, the analytic optimal solutions presented in Section 2.2 and the heuristic algorithm in Section 2.4 are the best choices.

3. ENERGY DISTRIBUTION FOR AN EPN WITH DISCONNECTIONS

Based on the simplified EPN (1), another EPN model with disconnections taken into consideration is presented and a related optimization problem that is similar to problem (2) is then described. We provide analytic optimal solutions to the problem for two of three cases. Then, we develop optimization algorithms to handle all three cases. First, we exploit a simple yet effective algorithm known as the cooperative particle swarm optimizer (CPSO-S) from [53] that was first introduced in [9]. Then, we adapt the effective heuristic algorithm 1 developed in Section 2.4. Numerical results verify the analytic optimal solutions for the two cases. Moreover, the results show that the heuristic algorithm 1 is capable of finding nearly-optimal solutions and performs better than the CPSO-S, especially when the dimensions of the problem become high.

3.1. Mathematical Model and Problem Description

Suppose that, in the EPN (1), there are C_1 consumers located too far away from the energy generators such that these consumers cannot get energy directly from the generators (or say, these consumers are disconnected from the generators), where $0 < C_1 < C$. Let $C_2 = C - C_1$. This EPN can be described in Figure 8, where there are G energy generators, S energy storages, C_1 disconnected energy consumers and C_2 connected energy consumers. Let $\hat{q}_s$, $\bar{q}_{c_1}$ and q_{c_2} denote the stationary probabilities that the sth storage has energy in storage and c_1th disconnected consumer and c_2th connected consumer have energy to consume, respectively. According to the G-network theory [16,21,22,32,36], these probabilities can also be calculated directly as:

$$\begin{aligned}
\hat{q}_s &= \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s + \sum_{c_1=1}^{C_1}(\delta_s \bar{p}_{s,c_1}) + \sum_{c_2=1}^{C_2}(\delta_s p_{s,c_2})}, 1\right), \\
\bar{q}_{c_1} &= \min\left(\frac{\sum_{s=1}^{S}(\hat{q}_s \delta_s \bar{p}_{s,c_1})}{\bar{\mu}_{c_1}}, 1\right), \\
q_{c_2} &= \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c_2}) + \sum_{s=1}^{S}(\hat{q}_s \delta_s p_{s,c_2})}{\mu_{c_2}}, 1\right),
\end{aligned} \tag{20}$$

where $\forall s \in S$, $\forall c_1 \in C_1$ and $\forall c_2 \in C_2$. In addition, $\sum_{c_1=1}^{C_1} \bar{p}_{s,c_1} + \sum_{c_2=1}^{C_2} p_{s,c_2} = 1$ and $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$.

We try to manage the energy distribution in this EPN (20) in order to maximize the amount of work done by the consumers per unit time, or equivalently: given $\{\Lambda_g | g = 1, \ldots, G\}$, $\{\bar{\mu}_{c_1} | c_1 = 1, \ldots, C_1\}$, $\{\mu_{c_2} | c_2 = 1, \ldots, C_2\}$, $\{\delta_s | s = 1, \ldots, S\}$ and $\{\gamma_s | s = 1, \ldots, S\}$,

$$\begin{aligned}
&\text{maximize } K = \sum_{c_1=1}^{C_1}(\bar{\mu}_{c_1} q_{c_1}) + \sum_{c_2=1}^{C_2}(\mu_{c_2} q_{c_2}) \text{ using } \{\hat{p}_{g,s}\}, \{\tilde{p}_{g,c_2}\}, \{\bar{p}_{s,c_1}\}\{p_{s,c_2}\}, \\
&\text{over } 1 \le g \le G, \quad 1 \le s \le S, \quad 1 \le c_1 \le C_1, \quad 1 \le c_2 \le C_2.
\end{aligned} \tag{21}$$

3.2. Optimal Solutions for Energy Distribution

We consider the maximization problem (21) in three cases. We present the analytical optimal solutions for first and second cases. For the third case, we search for acceptable solutions using optimization algorithms that will be detailed in Sections 3.3 and 3.4.

3.2.1. The energy is limited. In this case, the energy is so limited that $\sum_{g=1}^{G} \Lambda_g \le \sum_{c_2=1}^{C_2} \mu_{c_2}$. This means that the total energy-generation rate is not larger than the total consumption rate of C_2 connected consumers.

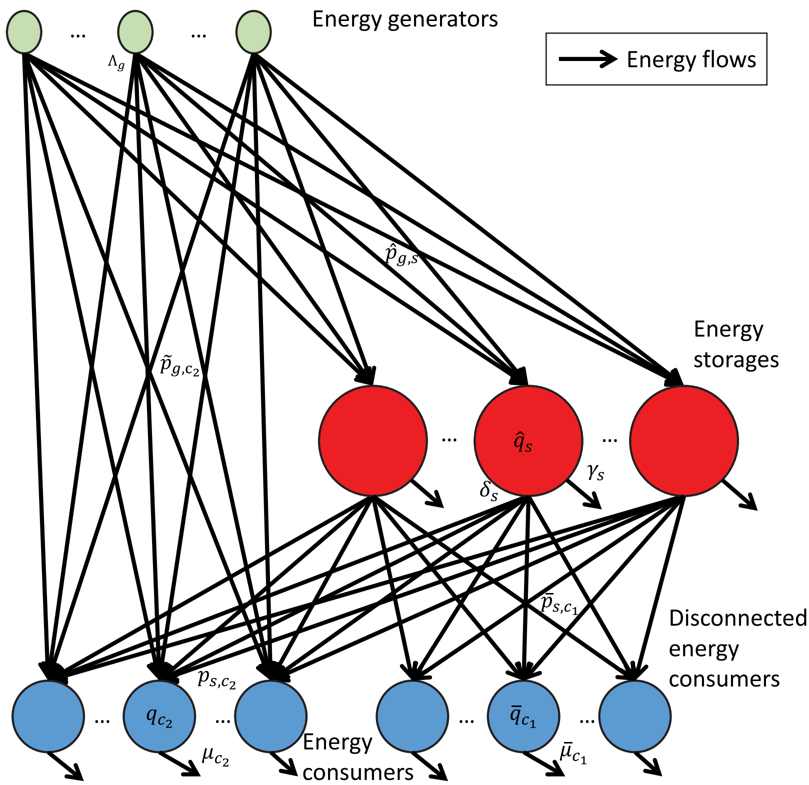

FIGURE 8. Brief model structure of the EPN with disconnections.

Let us set $\hat{p}_{g,s} = 0, \forall g \in G, s \in S$. Since $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$, then $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} = 1$. Then, the system (20) becomes

$$\hat{q}_s = 0, \quad \bar{q}_{c_1} = 0, \quad q_{c_2} = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c_2})}{\mu_{c_2}}, 1\right). \tag{22}$$

Let $H_2 = \sum_{c_2=1}^{C_2} \mu_{c_2}$. In addition, let us set

$$\tilde{p}_{g,c_2} = \frac{\mu_{c_2}}{H_2}, \quad \forall c_2 \in C_2. \tag{23}$$

Then, we have $q_{c_2} = (\sum_{g=1}^{G} \Lambda_g)/H_2 \leq 1, \forall c_2 \in C_2$. Then, $q_{c_2} = (\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c_2}))/\mu_{c_2}$ and $\mu_{c_2} q_{c_2} = \sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c_2})$.

Then,

$$K = \sum_{c_2=1}^{C_2}(\mu_{c_2} q_{c_2}) = \sum_{c_2=1}^{C_2}\sum_{g=1}^{G}(\Lambda_g \tilde{p}_{g,c_2}) = \sum_{g=1}^{G}\Lambda_g. \tag{24}$$

We could say that $K^* = \sum_{g=1}^{G} \Lambda_g$ is the optimal result for the problem (21), where **the optimal solution is $\hat{p}_{g,s} = 0$ and $\tilde{p}_{g,c_2} = \mu_{c_2}/H_2$, $\forall g \in G, s \in S, c_2 \in C_2$. In addition, $\bar{p}_{s,c_1}, p_{s,c_2}$ $\forall s \in S, c_1 \in C_1, c_2 \in C_2$ can be any non-negative value satisfying $\sum_{c_1=1}^{C_1} \bar{p}_{s,c_1} + \sum_{c_2=1}^{C_2} p_{s,c_2} = 1$.**

3.2.2. The energy is sufficient. In this case, the energy is sufficient such that $\sum_{c_2=1}^{C_2} \mu_{c_2} < \sum_{g=1}^{G} \Lambda_g \leq \sum_{c_2=1}^{C_2} \mu_{c_2} + \Upsilon_{s_1}$, where $s_1 = \arg\min_s \Upsilon_s = \arg\min_s(\gamma_s + \delta_s)$ with $s \in S$. Here, the s_1th storage is the one with the lowest energy consumption rate (the summation of energy transfer rate δ_s and energy leakage rate γ_s) among all energy storages.

First, we use part of the energy to satisfy the energy needs of all connected energy consumers such that $q_{c_2} = 1, \forall c_2 \in C_2$. Let $B = \sum_{g=1}^{G} \Lambda_g$. Let us set $\tilde{p}_{g,c_2} = \mu_{c_2}/B, p_{s,c_2} = 0, \forall g \in G, s \in S, c_2 \in C_2$. Then, from (20),

$$q_{c_2} = \min\left(\frac{\sum_{g=1}^{G}\left(\Lambda_g \frac{\mu_{c_2}}{B}\right)}{\mu_{c_2}}, 1\right) = 1, \quad \forall c_2 \in C_2. \tag{25}$$

And, $\sum_{c_2=1}^{C_2}(\mu_{c_2} q_{c_2}) = \sum_{c_2=1}^{C_2} \mu_{c_2} = H_2$.

Let $H = \sum_{c_1=1}^{C_1} \bar{\mu}_{c_1}$. We then set $\bar{p}_{s,c_1} = \bar{\mu}_{c_1}/H, \forall s \in S, c_1 \in C_1$. Then, $\bar{q}_{c_1} = \min(\sum_{s=1}^{S}(\hat{q}_s \delta_s)/H, 1)$, which means that each $\bar{q}_{c_1}, \forall c_1 \in C_1$ is equal to each other. Since $\sum_{c_1=1}^{C_1} \bar{p}_{s,c_1} = 1$ because of the setting of $p_{s,c_2} = 0, \forall c_2 \in C_2$, then from (20), we have

$$\hat{q}_s = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s + \sum_{c_1=1}^{C_1}(\delta_s \bar{p}_{s,c_1}) + \sum_{c_2=1}^{C_2}(\delta_s p_{s,c_2})}, 1\right) = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s + \delta_s}, 1\right). \tag{26}$$

Since, in this case, $\sum_{g=1}^{G} \Lambda_g \leq \sum_{c_2=1}^{C_2} \mu_{c_2} + \Upsilon_{s_1}$, we have $\hat{q}_s = \sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})/(\gamma_s + \delta_s) \leq 1$. Until here, we have two possible situations for $\bar{q}_{c_1}$ that is either $\bar{q}_{c_1} = 1$ or $\bar{q}_{c_1} < 1$.

If $\bar{q}_{c_1} = 1$, it is easy to have

$$K = \sum_{c_1=1}^{C_1} \bar{\mu}_{c_1} + \sum_{c_2=1}^{C_2} \mu_{c_2}, \quad \text{if } \bar{q}_{c_1} = 1, \quad \forall c_1 \in C_1. \tag{27}$$

If $\bar{q}_{c_1} < 1$, we need to analyze it in a different way. First, we make an assumption that $\bar{q}_{c_1} < 1$. Then,

$$\begin{aligned}\sum_{c_1=1}^{C_1}(\bar{\mu}_{c_1}\bar{q}_{c_1}) &= \sum_{c_1=1}^{C_1}\sum_{s=1}^{S}(\hat{q}_s \delta_s \bar{p}_{s,c_1}) = \sum_{c_1=1}^{C_1}\sum_{s=1}^{S}\left(\frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s/\delta_s + 1}\bar{p}_{s,c_1}\right) \\ &= \sum_{s=1}^{S}\left(\frac{\sum_{g=1}^{G}(\Lambda_g \hat{p}_{g,s})}{\gamma_s/\delta_s + 1}\right).\end{aligned} \tag{28}$$

To find a maximal value of $\sum_{c_1=1}^{C_1}(\bar{\mu}_{c_1}\bar{q}_{c_1})$, it is reasonable to make the denominators as large as possible while the numerators as small as possible. In addition, since we have set $\tilde{p}_{g,c_2} = \mu_{c_2}/B$, we have $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} = \sum_{c_2=1}^{C_2} \mu_{c_2}/B = H_2/B$. Since $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$, then $\sum_{s=1}^{S} \hat{p}_{g,s} = 1 - H_2/B = (B - H_2)/B$.

We then set $\hat{p}_{g,s_{\min}} = (B - H_2)/B$, where $s_{\min} = \arg\min_s(\gamma_s/\delta_s)$; and we set $\hat{p}_{g,s} = 0$ if $s \neq s_{\min}$. This setting means that we send all the remaining energy to the storage that has the smallest γ_s/δ_s, so that the denominator in (28) is made as large as possible and its numerator is made as small as possible at the same time. Then,

$$\sum_{c_1=1}^{C_1}(\bar{\mu}_{c_1}\bar{q}_{c_1}) = \frac{\sum_{g=1}^{G}(\Lambda_g(B-H_2)/B)}{\gamma_{s_{\min}}/\delta_{s_{\min}}+1} = \frac{B-H_2}{\gamma_{s_{\min}}/\delta_{s_{\min}}+1}. \quad (29)$$

Finally, we have that

$$K^* = \sum_{c_1=1}^{C_1}(\bar{\mu}_{c_1}\bar{q}_{c_1}) + \sum_{c_2=1}^{C_2}(\mu_{c_2}q_{c_2}) = \frac{B-H_2}{\gamma_{s_{\min}}/\delta_{s_{\min}}+1} + \sum_{c_2=1}^{C_2}\mu_{c_2}, \quad \text{if } \bar{q}_{c_1} < 1, \quad \forall c_1 \in C_1, \quad (30)$$

or

$$K^* = \sum_{c_1=1}^{C_1}\bar{\mu}_{c_1} + \sum_{c_2=1}^{C_2}\mu_{c_2}, \quad \text{if } \bar{q}_{c_1} = 1, \quad \forall c_1 \in C_1, \quad (31)$$

may be the optimal result for the maximization problem (21), where **the optimal solution is** $\tilde{p}_{g,c_2} = \mu_{c_2}/B, p_{s,c_2} = 0\ \forall g \in G, s \in S, c_2 \in C_2$**,** $\hat{p}_{g,s_{\min}} = (B-H_2)/B$ **for the storage** $s_{\min} = \arg\min_s(\gamma_s/\delta_s)$ **among all storages and** $\hat{p}_{g,s} = 0$ **for the rest storages** $s \neq s_{\min}$**. In addition,** $\bar{p}_{s,c_1} = \bar{\mu}_{c_1}/H, \forall s \in S, c_1 \in C_1$**.**

Remark 2: If $C_1 = C$ that means all consumers are disconnected from energy generators, the system becomes

$$\hat{q}_s = \min\left(\frac{\sum_{g=1}^{G}(\Lambda_g\hat{p}_{g,s})}{\gamma_s+\delta_s}, 1\right), \quad \bar{q}_{c_1} = \min\left(\frac{\sum_{s=1}^{S}(\hat{q}_s\delta_s\bar{p}_{s,c_1})}{\bar{\mu}_{c_1}}, 1\right). \quad (32)$$

Then, the optimal result may be $K^* = B\delta_{s_{\min}}/(\gamma_{s_{\min}}+\delta_{s_{\min}})$ if $\bar{q}_{c_1} < 1, \forall c_1 \in C_1$ or $K^* = \sum_{c_1=1}^{C_1}\bar{\mu}_{c_1}$, if $\bar{q}_{c_1} = 1, \forall c_1 \in C_1$, where **the optimal solution is** $\hat{p}_{g,s_{\min}} = 1$ **for** $s_{\min} = \arg\min_s(\gamma_s/\delta_s)$ **and** $\hat{p}_{g,s} = 0$ **for** $s \neq s_{\min}$**. In addition,** $\bar{p}_{s,c_1} = \bar{\mu}_{c_1}/H\ \forall s \in S, c_1 \in C_1$**.**

Remark 3: It can be complicate to analyze the optimal result and optimal solution for the maximization problem (21) if the energy is more sufficient such that $\sum_{g=1}^{G}\Lambda_g > \sum_{c_2=1}^{C_2}\mu_{c_2} + \Upsilon_{s_1}$ with $s_1 = \arg\min_s(\gamma_s+\delta_s), \forall s \in S$. In this case, designing an optimization algorithm to find approximate solutions may be a more practical choice.

3.3. Cooperative Particle Swarm Optimizer

In this subsection, we apply the simple yet effective algorithm, the CPSO-S algorithm [53], to solving the maximization problem (21).

First, let us define $\bar{P} \in R^{(SC+GC_2+GS)\times 1}$ as a vector that consists of $\hat{p}_{g,s}$, $\tilde{p}_{g,c_1}$, $\bar{p}_{s,c_1}$ and p_{s,c_2}, where

$$\bar{P}(h_{1s,c_1}) = \bar{p}_{s,c_1}, \quad \bar{P}(h_{2s,c_2}) = p_{s,c_2}, \quad \bar{P}(\tilde{h}_{g,c_2}) = \tilde{p}_{g,c_2}, \quad \bar{P}(\hat{h}_{g,s}) = \hat{p}_{g,s}, \quad (33)$$

with $h_{1s,c_1} = (s-1)C_1 + c_1$, $h_{2s,c_2} = SC_1 + (s-1)C_2 + c_2$, $\tilde{h}_{g,c_2} = SC + (g-1)C_2 + c_2$ and $\hat{h}_{g,s} = SC + GC_2 + (g-1)S + s$.

Let $M = SC + GC_2 + GS$ and N respectively denote the number of swarms and the number of particles in each swarm, where N is selected by the algorithm user. Let $\bar{P}_n(m)$ and $\bar{\bar{P}}_n(m)$ with $\forall m \in M, n \in N$ respectively denote the current state and best state of the

nth particle in the mth swarm. Let $\bar{P}^{(\text{best})}$ denote the best $\bar{P}$ found so far. Assume that $\kappa_n \sim U(0,1)$ and $\hat{\kappa}_n \sim U(0,1)$ with $\forall n \in N$ are uniform random sequences in the range $(0,1)$, which will be updated after each iteration, then, the state-update formula can be

$$\bar{P}_n^{(l+1)}(m) = \bar{P}_n^{(l+1)}(m) + v_{n,m}^{(l+1)}, \tag{34}$$

where

$$v_{n,m}^{(l+1)} = \omega v_{n,m}^{(l)} + 2\kappa_n^{(l)}(\tilde{\bar{P}}_n^{(l)}(m) - \bar{P}_n^{(l)}(m)) + 2\bar{\kappa}_n^{(l)}(\bar{P}^{(\text{best})}(m) - \bar{P}_n^{(l)}(m)), \tag{35}$$

ω is called the inertia weight and setup to vary from 1 to near 0 during the search process, l denotes the current number of iterations. The CPSO-S algorithm for solving problem (21) is similar to the one in [53] and thus omitted here. Note that, after each update, we make $\bar{P}$ satisfy constraints $\bar{P} \geq 0$, $\sum_{c_1=1}^{C_1} \bar{p}_{s,c_1} + \sum_{c_2=1}^{C_2} p_{s,c_2} = 1$ and $\sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1$ with minimal adjustments via linear normalization or other techniques.

Remark 4: The CPSO-S algorithm performs well in solving the problem (21). But, it requires solving the system (20) for $3MN$ times in a single iteration, which can be quite time-consuming when dealing with a system with high dimensions.

3.4. A Heuristic Algorithm

The followings present briefly how to adapt the heuristic algorithm 1 in Section 2.4 to find approximate solutions for the maximization problem (21).

Similar to what is done in Section 2.4, we first define a cost function $\bar{f}_{\hat{Y},\bar{P}} = \|\bar{A}\bar{P} - \bar{b}\|_2^2/2$ based on (20) with $\bar{q}_{c_1}, \bar{q}_{c_2}$ replaced by 1 and $\hat{q}_s$ replaced by $\hat{y}_s$, where $\hat{Y} = [\hat{y}_1\ \hat{y}_2\ \cdots\ \hat{y}_S]$, $\bar{A} \in R^{(S+C)\times(SC+GC_2+GS)}$ and $\bar{b} \in R^{(S+C)\times 1}$ are defined as

$$\begin{aligned}
&\bar{A}(\underline{\hat{h}}_s, h_{1s,c_1}) = \bar{A}(\underline{\hat{h}}_s, h_{2s,c_1}) = \hat{y}_s\delta_s, \quad \bar{A}(\underline{\hat{h}}_s, \hat{h}_{g,s}) = -\Lambda_g, \quad \bar{A}(\underline{\hat{h}}_s, \text{otherwise}) = 0,\\
&\bar{A}(\underline{h}_{1c_1}, h_{1s,c_1}) = \bar{A}(\underline{h}_{2c_2}, h_{2s,c_2}) = \hat{y}_s\delta_s, \quad \bar{A}(\underline{h}_{1c_1}, \text{otherwise}) = 0, \quad \bar{A}(\underline{h}_{2c_2}, \tilde{h}_{g,c_2}) = \Lambda_g,\\
&\bar{A}(\underline{h}_{2c_2}, \text{otherwise}) = 0, \quad \bar{b}(\underline{\hat{h}}_s) = -\hat{y}_s\gamma_s, \ \bar{b}(\underline{h}_{1c_1}) = \bar{\mu}_{c_1}, \quad \bar{b}(\underline{h}_{2c_2}) = \mu_{c_2},
\end{aligned} \tag{36}$$

with $\underline{\hat{h}}_s = s$, $\underline{h}_{1c_1} = S + c_1$ and $\underline{h}_{2c_2} = S + C_1 + c_2$, $\forall c_1 \in C_1, c_2 \in C_2$. We solve the following problem:

$$\begin{aligned}
&\text{minimize}_{\bar{P},\hat{Y}}\ \bar{f}_{\hat{Y},\bar{P}} = \frac{1}{2}\|\bar{A}\bar{P} - \bar{b}\|_2^2,\\
&\text{subject to } 0 \leq \hat{Y} \leq 1,\ \bar{P} \geq 0, \quad \sum_{c_1=1}^{C_1} \bar{p}_{s,c_1} + \sum_{c_2=1}^{C_2} p_{s,c_2} = 1, \ \sum_{c_2=1}^{C_2} \tilde{p}_{g,c_2} + \sum_{s=1}^{S} \hat{p}_{g,s} = 1.
\end{aligned} \tag{37}$$

The heuristic algorithm for solving problem (21) is quite similar to Algorithm 1. To illustrate the algorithm clearly, the detailed procedure is also given in this paper, shown in Algorithm 2, where $\bar{P}^{(m)}$ denotes the values of $\bar{P}$ in the mth iteration, $\nabla\bar{f}_{\hat{Y},\bar{P}^{(m)}} = \bar{A}^{\mathrm{T}}(\bar{A}\bar{P}^{(m)}) - \bar{A}^{\mathrm{T}}\bar{b}$, $\eta > 0$ is the step size and the operation Constraint$[\bar{P}]$ makes $\bar{P}$ satisfy the corresponding constraints with minimal adjustments.

Remark 5: Initial values of the parameters $\hat{Y}$ in Algorithm 2 affect its performance for solving problem (21). Multi-trials with different initial $\hat{Y}$ can be conducted to find better results.

Algorithm 2. A heuristic algorithm for solving problem (21)

Initialize $\bar{P}^{(1)}$ and $\hat{Y}$; $l \leftarrow 1$; $\vartheta \leftarrow 2$;
while $l \leq L$ **do**
  Construct $\bar{A}$ and $\bar{b}$; $\eta \leftarrow 1.5$; $m \leftarrow 1$;
  $\eta_{\text{new}} \leftarrow \eta$, $\bar{P}_{\text{new}} \leftarrow \bar{P}^{(1)}$, $\bar{f}_{\text{new}} \leftarrow \bar{f}_{\hat{Y},\bar{P}^{(1)}}$;
  while $m \leq M$ **do**
    $\bar{P}^{(m+1)} \leftarrow \text{Constraint}[\bar{P}^{(m)} - \eta\nabla\bar{f}_{\hat{Y},\bar{P}^{(m)}}]$;
    while $\bar{f}_{\hat{Y},\bar{P}^{(m+1)}} < \bar{f}_{\text{new}}$ **do**
      $\eta_{\text{new}} \leftarrow \eta$; $\bar{P}_{\text{new}} \leftarrow \bar{P}^{(m+1)}$; $\bar{f}_{\text{new}} \leftarrow \bar{f}_{\hat{Y},\bar{P}^{(m+1)}}$;
      $\eta \leftarrow \eta\vartheta$;
      $\bar{P}^{(m+1)} \leftarrow \text{Constraint}[\bar{P}^{(m)} - \eta\nabla\bar{f}_{\hat{Y},\bar{P}^{(m)}}]$;
    end while
    while $\bar{f}_{\hat{Y},\bar{P}^{(m+1)}} > \bar{f}_{\text{new}} \& \eta > 10^{-20}$ **do**
      $\eta \leftarrow \eta/\vartheta$;
      $\bar{P}^{(m+1)} \leftarrow \text{Constraint}[\bar{P}^{(m)} - \eta\nabla\bar{f}_{\hat{Y},\bar{P}^{(m)}}]$;
      if $\bar{f}_{\hat{Y},\bar{P}^{(m+1)}} < \bar{f}_{\text{new}}$ **then**
        $\eta_{\text{new}} \leftarrow \eta$; $\bar{P}_{\text{new}} \leftarrow \bar{P}^{(m+1)}$; $\bar{f}_{\text{new}} \leftarrow \bar{f}_{\hat{Y},\bar{P}^{(m+1)}}$;
      end if
    end while
    $\eta \leftarrow \eta_{\text{new}}$; $\bar{P}^{(m+1)} \leftarrow \bar{P}_{\text{new}}$; $\bar{f}_{\hat{Y},\bar{P}^{(m+1)}} \leftarrow \bar{f}_{\text{new}}$;
    $m \leftarrow m+1$;
  end while
  solve (20) using $\bar{P}_{\text{new}}$ for $\hat{q}_s, \forall s \in S$;
  $\bar{P}^{(1)} \leftarrow \bar{P}_{\text{new}}$; $\hat{y}_s \leftarrow \hat{q}_s, \forall s \in S$;
end while
get $\bar{P}_{\text{new}}$ as the final solution.

3.5. Numerical Results

In numerical experiments of this subsection, we also assume that the consumers differ from each other significantly, and there are at least two connected consumers ($C_2 > 1$) and two disconnected consumers ($C_1 > 1$). Therefore, we set $\bar{\mu}_{c_1} = 1 + 9(c_1 - 1)/(C_1 - 1)$ with $c_1 = 1, \ldots, C_1$, $\mu_{c_2} = 1 + 9(c_2 - 1)/(C_2 - 1)$ with $c_2 = 1, \ldots, C_2$. In addition, δ_s and γ_s with $\forall s \in S$ are randomly generated in the ranges of [7,12,47,50], respectively.

3.5.1. The energy is limited. In this case, $\sum_{g=1}^{G} \Lambda_g \leq \sum_{c_2=1}^{C_2} \mu_{c_2} = H_2$ and $\Lambda_g, \forall g \in G$ is randomly generated in the range of $[0, H_2/G]$.

Table 2 summarizes the results by using the optimal solutions in Section 3.2.1, the randomly-generated solutions (20,000 trials), the CPSO-S algorithm (50 iterations, $N = 3$) and the heuristic algorithm 2 ($M = 1, L = 100$ and 100 trials) to solve problem (21). It can be seen that Algorithm 2 is able to find solutions that are as good as the optimal solutions for the EPN with many generators and consumers, while the CPSO-S is so time-consuming that we cannot obtain results for it when there are 100 generators and 300 consumers.

3.5.2. When the energy is sufficient. In this case, $H_2 < \sum_{g=1}^{G} \Lambda_g \leq H_2 + \Upsilon_{s_1}$ with $s_1 = \arg\min_s(\gamma_s + \delta_s), \forall s \in S$ and $\Lambda_g, \forall g \in G$ is randomly generated in the range of $[H_2/G, (H_2 + \Upsilon_{s_1})/G]$.

TABLE 2. K^* and the best K found by using different algorithms for energy distribution of the EPN (20) with different G, S, C_1 and C_2 when the energy is limited.

(G, S, C_1, C_2)	K^*	Optimal	Random	CPSO-S	Heuristic
$(2, 2, 2, 2)$	**7.3110**	**7.3110**	7.0086	**7.3110**	**7.3110**
$(10, 5, 8, 8)$	**14.6502**	**14.6502**	14.0626	14.6498	**14.6502**
$(10, 10, 15, 15)$	**38.3693**	**38.3693**	36.1363	38.3254	**38.3693**
$(20, 10, 30, 30)$	**68.3396**	**68.3396**	65.2869	68.3377	**68.3396**
$(100, 100, 150, 150)$	**416.1667**	**416.1667**	–	–	**416.1667**

TABLE 3. K^* and the best K found by using different algorithms for energy distribution of the EPN (20) with different G, S, C_1 and C_2 when the energy is sufficient.

(G, S, C_1, C_2)	K^*	Optimal	Random	CPSO-S	Heuristic
$(2, 2, 2, 2)$	**15.4128**	**15.4128**	14.4421	15.2019	15.3463
$(10, 5, 8, 8)$	**48.6020**	**48.6020**	42.7710	48.3117	48.4840
$(10, 10, 15, 15)$	**86.2535**	**86.2535**	74.8169	85.7801	86.1077
$(20, 10, 30, 30)$	**168.4479**	**168.4479**	142.1887	168.0898	168.3106
$(100, 100, 150, 150)$	**828.9338**	**828.9338**	–	–	828.6729

TABLE 4. K^* and the best K found by using different algorithms for energy distribution of the EPN (20) with different G, S, C_1 and C_2 when the energy is more sufficient.

(G, S, C_1, C_2)	K^*	Optimal	Random	CPSO-S	Heuristic
$(2, 2, 2, 2)$	–	–	21.6368	**22.0000**	**22.0000**
$(10, 5, 8, 8)$	–	–	56.0484	68.6422	**69.7767**
$(10, 10, 15, 15)$	–	–	101.2606	123.6435	**125.8052**
$(20, 10, 30, 30)$	–	–	176.4236	220.0482	**222.4845**
$(100, 100, 150, 150)$	–	–	–	–	**1285.6231**

Table 3 summarizes the results by using the optimal solutions in Section 3.2.2, the randomly-generated solutions (20,000 trials), the CPSO-S algorithm (50 iterations, $N = 3$) and Algorithm 2 ($M = 1, L = 100$ and 10 trials) to solve problem (21). The results show that Algorithm 2 performs better than the CPSO-S and obtains results close to those of the optimal solutions.

3.5.3. The energy is more sufficient. In this case, $\sum_{g=1}^{G} \Lambda_g > H_2 + \Upsilon_{s_1}$ with $s_1 = \arg\min_s(\gamma_s + \delta_s), \forall s \in S$ and $\Lambda_g, \forall g \in G$ is randomly generated in the range of $[(H_2 + \Upsilon_{s_1})/G, (H_2 + \sum_{s=1}^{S}(\gamma_s + \delta_s))/G]$. In this case, analytic optimal solutions are not available. Table 4 summarizes the results by using the randomly-generated solutions (20,000 trials), the CPSO-S algorithm (50 iterations, $N = 3$) and Algorithm 2 ($M = 1, L = 100$ and 10 trials) to solve problem (21). We can see that Algorithm 2 performs the best.

In summary, the numerical results in this subsection well verify the correctness of the analytic optimal solutions and the efficacy of Algorithm 2 for managing energy flows inside the EPN with disconnections. The analytic optimal solutions are the best choices for the first and second cases when the energy is not very sufficient, while Algorithm 2 is the best choice when a widely applicable solution is needed.

4. CONCLUSIONS AND FUTURE WORK

We have presented two EPN models consisting of energy generators, energy storages and energy consumers, which could be used to model (or say, describe) certain types of power grids with distributed and intermittent energy generators such as RES. Based on the EPN model, we have formulated the problem of maximizing the effective work done by the energy consumers per unit time. We provide analytic optimal solutions to the maximization problem for most cases, and we have also presented an effective heuristic to handle all the cases. For comparison, we have also developed a gradient-descent algorithm and adapted the CPSO-S for solving the problem. Numerous numerical experiments have been conducted to verify the analytic solutions and to demonstrate the efficacy of the optimization algorithms. Furthermore, the proposed heuristics also show better performance as compared with other optimization algorithms. Thus, this paper shows that by using EPN models with G-network theory for analytical modeling, various operating objectives of power grids can be mathematically formulated into optimization problems. As a result, the control of the power grid can be simplified and the grid itself can become smarter. In future work, we plan to develop EPN models for the smart grid that take more real-world parameters into consideration, so as to investigate different relevant optimization objectives and techniques.

References

1. Abdelrahman, O.H. & Gelenbe, E. (2016). A diffusion model for energy harvesting sensor nodes. in *IEEE 24th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'16)*. IEEEXplore, September 2016, pp. 154–158.
2. Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R.E. (1995). Gambling in a rigged casino: The adversarial multi-armed bandit problem. in *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. IEEE, pp. 322–331.
3. Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R.E. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing* 32(1): 48–77.
4. Bamberger, Y., Baptista, J., Belmans, R., Buchholz, B.M., Chebbo, M., Doblado, J.L.D.V., Efthymiou, V., Gallo, L., Handschin, E., & Hatziargyriou, N. *et al.* (2006). Vision and strategy for Europe's electricity networks of the future.
5. Berl, A., Gelenbe, E., Di Girolamo, M., Giuliani, G., De Meer, H., Dang, M.Q., & Pentikousis, K. (2010). Energy-efficient cloud computing. *The Computer Journal* 53(7): 1045–1051.
6. Brun, O., Wang, L., & Gelenbe, E. (2016). Big data for autonomic intercontinental overlays. *IEEE Journal on Selected Areas in Communications* 34(3): 575–583.
7. Caldon, R., Patria, A.R., & Turri, R. (2004). Optimal control of a distribution system with a virtual power plant. *Bulk power system dynamics and control, Cortina. d'Ampezzo, Italy*.
8. Ceran, E.T. & Gelenbe, E. (2016). Energy packet model optimisation with approximate matrix inversion. In *Proceedings of the 2nd International Workshop on Energy – Aware Simulation*. ACM, p. 4.
9. Den Bergh, V. & Engelbrecht, A.P. (2000). Cooperative learning in neural networks using particle swarm optimizers. *South African Computer Journal* 26: 84–90.
10. Fang, X., Misra, S., Xue, G., & Yang, D. (2012). Smart grid – the new and improved power grid: a survey. *IEEE Communications Surveys & Tutorials*, 14(4): 944–980.
11. Fang, X., Yang, D., & Xue, G. (2011). Online strategizing distributed renewable energy resource access in Islanded microgrids. In *2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)*, IEEE, pp. 1–6.
12. Farhangi, H. (2010). The path of the smart grid. *IEEE Power and Energy Magazine*, 8(1): 18–28.
13. Fourneau, J.-M. & Gelenbe, E. (2004). Flow equivalence and stochastic equivalence in G-networks. *Computational Management Science* 1(2): 179–192.
14. Fourneau, J.-M., Gelenbe, E., & Suros, R. (1996). G-networks with multiple classes of negative and positive customers. *Theoretical Computer Science* 155(1): 141–156.
15. Gelenbe, E. (1990). Stability of the random neural network model. *Neural Computation* 2(2): 239–247.
16. Gelenbe, E. (1993). G-networks with signals and batch removal. *Probability in the Engineering and Informational Sciences* 7(3): 335–342.

17. Gelenbe, E. (1993). Learning in the recurrent random neural network. *Neural Computation* 5(1) 154–164.
18. Gelenbe, E. (1993). G-networks by triggered customer movement. *Journal of Applied Probability* 30(03): 742–748.
19. Gelenbe, E. (1994). G-networks: a unifying model for neural and queueing networks. *Annals of Operations Research* 48(5): 433–461.
20. Gelenbe, E. (2009). Steps toward self-aware networks. *Communications of the ACM* 52(7): 66–75.
21. Gelenbe, E. (2012). Energy packet networks: smart electricity storage to meet surges in demand. In *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 1–7.
22. Gelenbe, E. (2012). Energy packet networks: adaptive energy management for the cloud. In *Proceedings of the 2nd International Workshop on Cloud Computing Platforms*. ACM, p. 1.
23. Gelenbe, E. (2014). Adaptive management of energy packets. In *2014 IEEE 38th International Computer Software and Applications Conference Workshops (COMPSACW)*. IEEE, pp. 1–6.
24. Gelenbe, E. (2014). A sensor node with energy harvesting. *ACM SIGMETRICS Performance Evaluation Review* 42(2): 37–39.
25. Gelenbe, E. (2014). Error and energy when communicating with spins. In 2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP). IEEE, pp. 784–787.
26. Gelenbe, E. (2015). Synchronising energy harvesting and data packets in a wireless sensor. *Energies* 8(1): 356–369. [Online]. Available: http://www.mdpi.com/1996-1073/8/1/356.
27. Gelenbe, E. (2015). Errors and power when communicating with spins. *IEEE Transactions on Emerging Topics in Computing* 3(4): 483–488.
28. Gelenbe, E. (2016). Agreement in spins and social networks. *ACM SIGMETRICS Performance Evaluation Review* 44(2): 15–17.
29. Gelenbe, E. & Caseau, Y. (2015). The impact of information technology on energy consumption and carbon emissions. *Ubiquity*, vol. 2015, no. June, p. 1.
30. Gelenbe, E. & Ceran, E.T. (2015). Central or distributed energy storage for processors with energy harvesting. Sustainable Internet and ICT for Sustainability (SustainIT), 2015. IEEE, pp. 1–3.
31. Gelenbe, E. & Ceran, E.T. (2016). Energy packet networks with energy harvesting. *IEEE Access* 4: 1321–1331.
32. Gelenbe, E. & Fourneau, J.-M. (2002). G-networks with resets. *Performance Evaluation* 49(1): 179–191.
33. Gelenbe, E., Gesbert, D., Gunduz, D., Kulah, H., & Uysal-Biyikoglu, E. (2013). Energy harvesting communication networks: Optimization and demonstration (the e-crops project). In *24th Tyrrhenian International Workshop on Digital Communications-Green ICT (TIWDC), 2013*. IEEE, pp. 1–6.
34. Gelenbe, E. & Gunduz, D. (2013). Optimum power level for communications with interference. Digital Communications-Green ICT (TIWDC), 2013 24th Tyrrhenian International Workshop on. IEEE, pp. 1–6.
35. Gelenbe, E. & Hussain, K. (2002). Learning in the multiple class random neural network. *IEEE Transactions on Neural Networks* 13(6): 1257–1267.
36. Gelenbe, E. & Labed, A. (1998). G-networks with multiple classes of signals and positive customers. *European Journal of Operational Research* 108(2): 293–305.
37. Gelenbe, E. & Marin, A. (2015). Interconnected wireless sensors with energy harvesting. International Conference on Analytical and Stochastic Modeling Techniques and Applications. Springer International Publishing, May 2015, pp. 87–99.
38. Gelenbe, E. & Oklander, B. (2014). Energy and QoS in a cognitive channel. *European Wireless 2014; 20th European Wireless Conference*. In *Proceedings of VDE*, pp. 1–6.
39. Gelenbe, E. & Timotheou, S. (2008). Random neural networks with synchronized interactions. *Neural Computation* 20(9): 2308–2324.
40. Guan, X., Xu, Z., & Jia, Q.-S. (2010). Energy-efficient buildings facilitated by microgrid. *IEEE Transactions on Smart Grid,* 1(3): 243–252.
41. Gelenbe, E. & Yin, Y. (2016). Deep learning with random neural networks. In *International Joint Conference on Neural Networks (IJCNN'16)*. IEEEXpress, July 2016.
42. Gelenbe, E. & Yin, Y. (2016). Deep learning with random neural networks. In *SAI Intelligent Systems Conference, 2016*. IEEEXpress, September 2016.
43. Hledik, R. (2009). How green is the smart grid? *The Electricity Journal* 22(3): 29–41.
44. Kadioglu, Y.M. & Gelenbe, E. (2016). Packet transmission with k energy packets in an energy harvesting sensor. In *Proceedings of the 2nd International Workshop on Energy-Aware Simulation*. ACM, p. 1.
45. Lasseter, R.H. & Paigi, P. (2004). Microgrid: a conceptual solution. In *2004 IEEE 35th Annual Power Electronics Specialists Conference, 2004 (PESC 04)*, vol. 6. IEEE, pp. 4285–4290.

46. Lombardi, P., Powalko, M., & Rudion, K. (2009). Optimal operation of a virtual power plant. In *Power & Energy Society General Meeting, 2009 (PES'09).* IEEE, pp. 1–6.
47. McDaniel, P. & McLaughlin, S. (2009). Security and privacy challenges in the smart grid. *IEEE Security & Privacy* 7(3): 75–77.
48. Molderink, A., Bakker, V., Bosman, M.G., Hurink, J.L., & Smit, G.J. (2010). Management and control of domestic smart grid technology. *IEEE Transactions on Smart Grid,* 1(2): 109–119.
49. Palensky, P. & Kupzog, F. (2013). Smart grids. *Annual Review of Environment and Resources* 38: 201–226.
50. Takuno, T., Koyama, M., & Hikihara, T. (2010). In-home power distribution systems by circuit switching and power packet dispatching. In *2010 First IEEE International Conference on Smart Grid Communications, 2010 (SmartGridComm, 2010),* IEEE, pp. 427–430.
51. Tashiro, K., Takahashi, R., & Hikihara, T. (2012). Feasibility of power packet dispatching at in-home dc distribution network. In *2012 IEEE Third International Conference on Smart Grid Communications, 2012 (SmartGridComm),* IEEE, pp. 401–405.
52. Timotheou, S. (2008). Nonnegative least squares learning for the random neural network. In V. Kůrková, R. Neruda & J. Koutník (eds), *Artificial Neural Networks – ICANN 2008: 18th International Conference, Prague, Czech Republic, September 3–6, 2008, Proceedings, Part I.* Springer, Berlin, Heidelberg, pp. 195–204.
53. Van den Bergh, F. & Engelbrecht, A.P. (2004). A cooperative approach to particle swarm optimization. *IEEE Transactions on Evolutionary Computation,* 8(3): 225–239.
54. Wang L. & Gelenbe, E. (2015). Adaptive dispatching of tasks in the cloud. *IEEE Transactions on Cloud Computing* PP(99): 1–1.
55. Yin, Y. (2016). Line-search aided non-negative least-square learning for random neural network. In O.H. Abdelrahman, E. Gelenbe, G. Gorbil & R. Lent (eds), *Information Sciences and Systems 2015: 30th International Symposium on Computer and Information Sciences (ISCIS 2015).* Springer International Publishing, Cham, pp. 181–189.