

Received April 16, 2020, accepted May 11, 2020, date of publication May 21, 2020, date of current version June 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2996309

# Prosumers Matching and Least-Cost Energy Path Optimisation for Peer-to-Peer Energy Trading

OLAMIDE JOGUNOLA<sup>1</sup>, (Student Member, IEEE), WEIZHUO WANG<sup>1</sup>,  
AND BAMIDELE ADEBISI<sup>1</sup>, (Senior Member, IEEE)

Department of Engineering, Manchester Metropolitan University, Manchester M1 5GD, U.K.

Corresponding author: Bamidele Adebisi (b.adebisi@mmu.ac.uk)

This work was supported by the ENERGY-IQ, a UK-Canada Power Forward Smart Grid Demonstrator project funded by The Department for Business, Energy and Industrial Strategy (BEIS) under Grant 7454460.

**ABSTRACT** Potential benefits of peer-to-peer energy trading and sharing (P2P-ETS) include the opportunity for prosumers to exchange flexible energy for additional income, whilst reducing the carbon footprint. Establishing an optimal energy routing path and matching energy demand to supply with capacity constraints are some of the challenges affecting the full realisation of P2P-ETS. In this paper, we proposed a slime-mould inspired optimisation method for addressing the path cost problem for energy routing and the capacity constraint of the distribution lines for congestion control. Numerical examples demonstrate the practicality and flexibility of the proposed method for a large number of peers (15 – 2000) over existing optimised path methods. The result shows up to 15% cost savings as compared to a non-optimised path. The proposed method can be used to control congestion on distribution links, provide alternate paths in cases of disruption on the optimal path, and match prosumers in the local energy market.

**INDEX TERMS** Peer-to-peer energy trading, peer-to-peer energy trading and sharing, matching algorithm, shortest path, slime mould, smart grid.

## NOMENCLATURE

$\alpha$	Discrete time step.
$\bar{\Phi}_{i,j}(x_{i,j})$	The cost of path $P$ traversed from $i$ to $j$ .
$\bar{M}$	A subset of link $E$ .
$\gamma$	A constant representing the decay rate of the tube.
$\mathcal{A}$	Set of link weights representing the link costs.
$A_{i,j}$	The weight along the link $(i, j)$ representing the link cost.
$c_{i,j}$	The capacity of the link $(i, j)$ .
$D_{i,j}$	The conductivity/total traffic of the tube/link.
$E$	Set of network links $(i, j)$ connecting the prosumers.
$G$	The strongly connected network graph.
$G^*$	A subgraph of $G$ .
$I_o$	A constant flux/energy demand flowing from the $n_i$ to $n_j$ .
$L$	The length of the tube.
$N$	Total number of actors.
$n_i$	Source prosumer.
$n_j$	Destination prosumer.

$P$	The path traversed by traffic flow $x_{i,j}$ from producer $n_i$ to consumer $n_j$ .
$p_i, p_j$	The pressure at $n_i$ and $n_j$ .
$Q_{i,j}$	Flux/energy demands on the tube/link connecting $n_i$ and $n_j$ .
$R$	Set of nodes of size $ \bar{M} $ .
$t$	Iteration time.
$V$	Interconnected nodes representing the set of actors.
$X, Y$	Bipartite sets of graph $G$ .
$x_{i,j}$	The traffic/demand flow on the link $(i, j)$ .

## I. INTRODUCTION

In recent years, smart grid (SG) has emerged with intelligent monitoring, control, and management of the traditional power grid, offering increased automation and bi-directional communication to improve the efficiency of the grid [1], [2]. Peer-to-peer energy trading and sharing (P2P-ETS) is an application of SG for transacting energy among a community of connected peers to realise better performance of the grid including scalability, robustness, and reduction in carbon emissions [3]–[5]. One challenge of P2P-ETS is the large influx of distributed energy resources (DER) actively

The associate editor coordinating the review of this manuscript and approving it for publication was Bin Zhou<sup>1</sup>.

being connected to the grid, making grid management, power flow, and control challenging. To alleviate these challenges, distributed algorithms have been proposed for energy coordination and control [4]–[10]. Authors in [4] proposed pair matching strategies for prosumer's energy trading market. In [7], an energy exchange problem among several microgrids is addressed to minimise global operation costs. Study [10] presented a framework for P2P energy trading. The study assumes that the underlying communication link is perfect. The impact of imperfect communication links is assessed in [5] while [6] addressed the optimal routing algorithm to facilitate communication among microgrids. The aforementioned literature paid far little attention to the underlying energy path/route connecting these prosumers.

Energy path optimisation has been a fundamental distribution grid network challenge; the primary aim of which is to minimise the weights/cost of the path connecting generators to loads. Energy loss due to long-distance transmission and other associated costs has a direct correlation to the distance it travels. For instance, 'Ofgem' reported that 66.48% of an electricity bill is a service charge, with approximately 24% related to network characteristics including distance charge [11]. One solution to minimising cost path is the creation of multiple and redundant links between the generators and the loads, leading to increased network cost and complexities. With the integration of these numerous DERs, energy path complexity increases, thus, an optimised path for the energy demand/supply of a prosumer (a producer and consumer of energy) to be routed to their energy trading target based on least-cost path and capacity constraints is actively needed. This will inadvertently reduce the energy routing cost, improves control schemes, and reduces energy congestion on the distribution lines.

While several techniques have been proposed for path optimisation problems in the cyber-physical networks, including the Dijkstra algorithm [12], Bellman-Ford algorithm [13], etc., the computation time for these algorithms is excessive when the network scale becomes large [14]. To address the computational complexity, nature-based algorithms including bioinspired techniques like genetic algorithm, particle swarm optimisation, and ant colony optimisation have emerged [14]. Recently, the capability of *Physarum Polycephalum*, also known as slime mould dynamics, has been shown to offer efficient techniques in solving many graph network problems [14]–[17].

Interestingly, slime mould optimisation is particularly suited for the SG network, as it can model both the shortest path cost problem for energy routing and the capacity constraint of the distribution lines for congestion control. This combination is a major strength of the proposed scheme compared to other traditional algorithms that only model shortest path problems. Besides, for P2P-ETS where prosumers are distributed and diverse, a shortest path finding algorithm to establish an optimised path among connected peers in parallel is of most importance. Parallel execution reduces network computation time and improves efficiency. This would not

only result in low cost but would also reduce the search time significantly, especially in a large network as P2P-ETS.

The main contributions of this work can be summarised as follows:

- we present a slime mould-inspired approach for SG to determine the least-cost optimal path between consumers and producers whilst implementing different scenarios for energy network representation, and routing energy demand between the prosumers;
- a specific case of maximum flow capacity in the distribution network is considered to realise optimal path for energy flow in a capacitated network for congestion control and to provide an alternate path in scenarios of disruption with the optimal energy distribution path;
- finally, we extend the optimised path algorithm for perfect matching of the energy prosumers, ensuring all consumers are matched with producers to collectively reduce the network costs.

The remaining sections are organised as follows: Relevant literature review is presented in Section II. Section III introduced the problem formulation, the slime mould based optimised path algorithm, and an extension to the algorithm for energy matching between producer and consumer. Evaluation of the developed solution is discussed in Section IV, including its application to the optimal path for energy flow as well as its numerical examples and the results. Section V summarises the paper and identified future works.

## II. LITERATURE REVIEW

The theory of complex networks involving a graph modelling of real-world networks and evolutionary computation algorithms have increasingly been used in SG applications for path optimisation, resource discovery and power routing. SG has been modelled as a complex network to analyse and adapt the distribution of power flow [18]. The optimisation objectives in SG, including the least-cost flow and shortest path finding problems, can be solved by using the technique of graph theory. Shortest path finding algorithms are used to find the minimum weighted or most efficient path in the network. They are used to identify a path or route between two vertices such that the sum of the weights of its constituent edges is minimised. This is equivalent to finding the optimal and also the alternate paths for the flow of power from a generating station to consumer ends [19]. By integrating the theory of complex network and evolutionary algorithm concepts, a multi-objective minimisation problem can be formulated [20]. This objective function combines cost elements, related to the number of electric cables (graph links), and several metrics that quantify properties beneficial for SG including energy exchange at the local scale (considering high robustness and resilience). A method to manage the active power in the distribution systems using an application of the graph theory, specifically, the successive shortest path algorithm, is introduced in [21], for optimal power generation, dispatch, and power flows. The algorithm is implemented in a distributed way with simulations to prove

its efficiency in cases of optimal operation, congestion management, and power generation cost. Furthermore, in minimising time-out delays in power system networks during outages, Hemalatha, et al. [19] investigated the transmission of power through the optimal path for quick reconfiguration of power system components using Bellman-Ford algorithm. The solution is modelled for a given set of generation, load pair, through the optimal path considering the capacity of the transmission line, voltage stability, shortest path (minimum losses), priority of loads, and power balance between the generation and demand. The algorithm was applied to a practical 230kV network to demonstrate its effectiveness. To maintain the stability of a microgrid (MG) system through load shedding, study [17] proposed a Physarum-based hybrid optimisation algorithm with ant colony optimisation (PM-ACO). The model improved the selection probability of important items and emerged a positive feedback process to generate optimal solutions. Experimental results demonstrate that the proposed PM-ACO algorithms have stronger robustness and a higher convergence rate. The authors in [22] extended the Dijkstra algorithm to a multi-objective shortest path algorithm to design a spanning graph of a communication infrastructure connecting the Phasor Measurement Units to the control centre.

From the preceding discussion, Dijkstra, Bellman-Ford, and Kruskal’s algorithms are examples of single objective shortest path algorithms used in the SG applications discussed. These algorithms track the shortest paths from a single source to one determined node in the graph and find the shortest paths independently [22]. However, according to the structure of power networks, specifically in large P2P-ETS networks, it is sometimes necessary to find shortest paths from one centered bus and/or multiple points to multiple buses (i.e. phasor measurements units, households) simultaneously checking overlapped paths in the routing problem. This requires a multi-objective shortest path algorithm like slime mould optimisation, to find the best solution routes for connecting SG components to ensure the least-cost power flow.

This motivated the study to harness the potential of slime mould-inspired path optimisation to address the problem of matching prosumers in the microgrid to facilitate P2P-ETS utilising the least-cost optimal path between them. Secondly, to route energy demand between the producers and consumers and reduce the chances of overloading the distribution lines.

### III. OPTIMAL PATH PROBLEM FORMULATION

Consider an energy network model that focuses on least-cost path optimisation and energy demand routing, among prosumers at the tertiary level of control without explicitly touching the physicality of the underlying distribution network such as power flow analysis. Here, least-cost path optimisation refers to the cost of routing energy demands

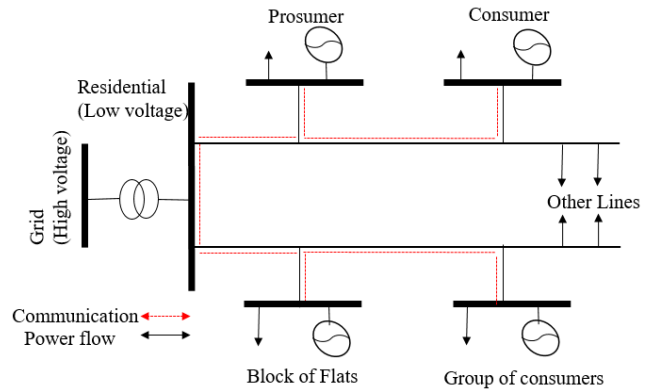


FIGURE 1. Benchmark of MG connection to the grid (adapted from [23]).

over a distribution link in relation to the weight or cost of the links used. Interactions among these prosumers are illustrated in Fig. 1, that shows the power and communication connection between prosumers, consumers, and the grid [23].

From Fig. 1, for power flow from the grid to the consumer, the optimal path would be through the connection to the prosumer, then to the consumer. The same analogy applies to other actors (prosumers, consumers, and producers) illustrated in Fig. 1. Thus, each actor ( $i = 1, \dots, N$ ) has its computed energy needs (demands or supplies) to be satisfied in the network. For instance, actor  $i$  is a producer (source) that desires to sell energy to consumer  $j$  (sink). The energy network is described as a connected graph  $G = (V, E)$ , where  $V = \{1, \dots, N\}$  denotes the set of actors.

$G$  is a strongly connected directed graph, where every node/actor in the network is reachable from every other node, i.e., there exists a directed link  $e_{ij} \in E$  denoted by  $(i, j)$  from node  $n_i$  to node  $n_j$  in the network.  $E(t) \subseteq V \times V$  is a set of links that changes over time according to the state of the link at the time,  $t$ . Each directed link is characterised by its capacity,  $c_{i,j}$ , (the maximum power that can flow through the link), and the power flow,  $x_{i,j}$  from the producer to the consumer. For each link,  $\mathbf{A}$  denotes the set of the link weights which represents the link costs. Representing  $|V|$  as  $n$ , then  $\mathbf{A}$  can be expressed as the symmetrical adjacency matrix of equation (1), where  $A_{i,j}$  denotes the weight on the link  $(i, j)$ ,

$$\mathbf{A} = \begin{bmatrix} 0 & A_{12} & A_{13} & \dots & A_{1n} \\ A_{21} & 0 & A_{23} & \dots & A_{2n} \\ A_{31} & A_{32} & 0 & \dots & A_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & A_{n3} & \dots & 0 \end{bmatrix} \quad (1)$$

Given a source node  $i \in V$ , a sink node  $j \in V$ , a function  $\bar{\Phi}_{i,j}(x_{i,j})$  represents the cost of path  $P$  that traversed by energy flow  $x_{i,j}$  from  $n_i$  to  $n_j$ . The optimal cost path problem can be

formulated as

$$\min \sum_{p \in P(i,j)} \bar{\Phi}_{i,j}(x_{i,j}) \quad (2a)$$

$$s.t. : \sum_{j:i,j \in E} x_{i,j} = \sum_{i:j,i \in E} x_{j,i}, \quad \forall i, j \in E \quad (2b)$$

$$\sum_{p \in P(i,j)} x_{i,j} \leq c_{i,j}, \quad \forall i, j \in E, p \in P \quad (2c)$$

$$\sum_{p \in P(i,j)} x_{i,j} \geq 0, \quad \forall i, j \in E, p \in P \quad (2d)$$

where  $P(i, j)$  represents the set of all paths from producer  $n_i$  to consumer  $n_j$ . Equation (2b) is the energy conservation constraints, where all energy entering a node (not the source nor destination) leaves the node. Equation (2c) is the capacity constraints that suggest that the energy flow  $x_{i,j}$  to be less than the capacity of the link  $c_{i,j}$ , while (2d) is the non-negativity constraints meaning that, there should be a positive flow from  $n_i$  to  $n_j$ .

### A. PHYSARUM POLYCEPHALUM ALGORITHM

The slime mould based path optimised algorithm is briefly introduced in this section. *Physarum polycephalum* forms a dynamic tubular network connecting peers, where the diameters of the tubes carrying large fluxes of flows grow to expand their capacities, and the tubes that are not used decline and disappear entirely. The segments of tubes of slime mould may be modelled as edges of a graph network, with an intersection point representing the nodes. The parameters  $Q_{i,j}$  represent the flux on the tubes connecting node  $n_i$  to node  $n_j$ . According to Kirchoff's law and the law of conservation of flow, the flux of input at a source is equal to the total flux of output at all node sets [24], [25]. Also, at any other node, the sum of flux flowing in is equal to the sum of flux flowing out [25]. The flow conservation may be expressed as:

$$\sum_j Q_{i,j} = \begin{cases} -I_o, & \text{for } i = \text{source} \\ I_o, & \text{for } j = \text{sink} \\ 0, & \text{otherwise.} \end{cases} \quad (3a)$$

$$\sum_j Q_{i,j} = \begin{cases} I_o, & \text{for } j = \text{sink} \\ 0, & \text{otherwise.} \end{cases} \quad (3b)$$

$$\sum_j Q_{i,j} = \begin{cases} 0, & \text{otherwise.} \end{cases} \quad (3c)$$

where  $I_o$  is constant and represents the flux flowing from the source node or into the sink node. Assuming the flow along the tubes is approximated by Poiseuille flow, the flux  $Q_{i,j}$  can be calculated by,

$$Q_{i,j} = \frac{D_{i,j}}{L_{i,j}}(p_i - p_j) \quad (4)$$

where  $D_{i,j}$  is the conductivity of the tube,  $L_{i,j}$  is the length of the link, and  $p_i$  is the pressure at node  $i$ . To calculate the pressure on each node, equation (4) is substituted in (5) as:

$$Q_{i,j} = \frac{D_{i,j}}{L_{i,j}}(p_i - p_j) = \begin{cases} -I_o, & \text{for } i = \text{source} \\ I_o, & \text{for } j = \text{sink} \\ 0, & \text{otherwise.} \end{cases} \quad (5a)$$

$$Q_{i,j} = \frac{D_{i,j}}{L_{i,j}}(p_i - p_j) = \begin{cases} I_o, & \text{for } j = \text{sink} \\ 0, & \text{otherwise.} \end{cases} \quad (5b)$$

$$Q_{i,j} = \frac{D_{i,j}}{L_{i,j}}(p_i - p_j) = \begin{cases} 0, & \text{otherwise.} \end{cases} \quad (5c)$$

By setting  $p_j$  to the basic pressure level of  $p_j = 0$ , all  $p_i$  and  $Q_{i,j}$  can be determined. To model the adaptive behaviour of the slime mould, the conductivity  $D_{i,j}$  is believed to change

over time as the flux increases or decreases, resulting in the evolution of  $D_{i,j}$  as

$$\frac{\delta D_{i,j}}{\delta t} = f(|Q_{i,j}|) - \gamma D_{i,j} \quad (6)$$

where  $\gamma$  is a constant that represents the decay rate of the tube that ensures the convergence of the algorithm when the value is set to 1. Equation (6) is called the adaptation equation, depicting the relationship between the conductivity and flux on the link. For instance, the conductivity diminishes when the flux on the links is zero and increases by the amount of flux on the link.  $f(\cdot)$  is assumed to be monotonically increasing continuous function satisfying  $f(0) = 0$ . Assume  $f(|Q|) = |Q|$ ,  $\gamma = 1$ , Physarum can always converge to the shortest path [25]. For an iterative process, and adopting the functional form  $f(Q) = |Q|$ , (7) is used instead of (6) to calculate the conductivity of the link as

$$\frac{D_{i,j}^{t+1} - D_{i,j}^t}{\delta t} = \alpha |Q_{i,j}^t| - \gamma D_{i,j}^{t+1} \quad (7)$$

where  $D_{i,j}^t$  is the value of  $D_{i,j}$  at the  $t^{th}$  iteration time. Thus, based on the feedback from the iteration, a critical link would be preserved and others would be deleted, thus forming a Physarum spanning tree. Other algorithms solve problems by progressing step-by-step in a single direction, the slime mould algorithm works by sampling a variety of directions in parallel. Based on the samples, it discards less optimal directions to progress toward the solution. After adaptation, the algorithm selects another set of sources and sink nodes and repeats the calculations.

*Example 1:* Consider the network shown in Fig. 2, the shortest path between nodes 1 and 5 needs to be determined.

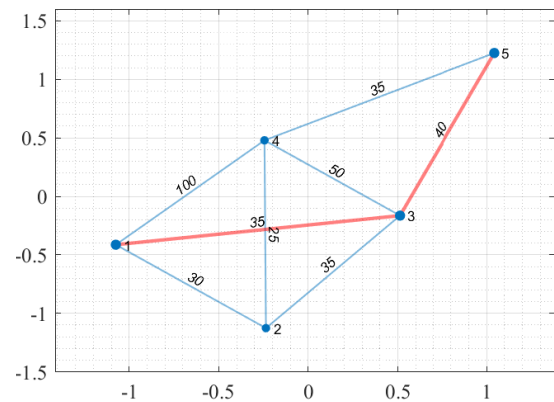


FIGURE 2. Example network showing the least-cost path from Prosumer 1 to Prosumer 5.

The number along the edge represents the weight of the edge. Each source node creates a unit flow of demands and the demands are consumed at destination nodes called sinks. To start the implementation, the conductivity is first initialised, as the flux moves along the edges, the conductivity is recorded, as shown in Fig. 3.

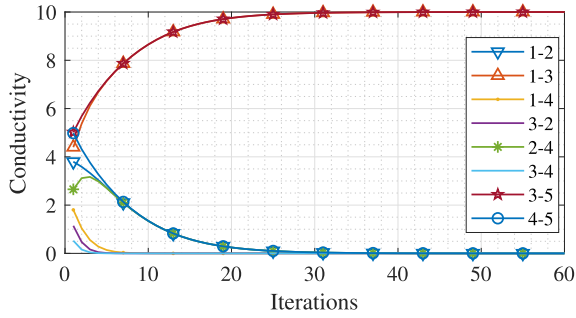


FIGURE 3. The changing trend of the conductivity on each edge.

It can be observed that the flux along edges (1, 3) and (3, 5) converge to 1 signifying the shortest path, with the same result evidence from other traditional algorithms including the Dijkstra algorithm, but with a longer convergence time. Here, convergence is achieved when the strongest flow moves from the source to the sink using the least-cost shortest path and remained constant (the flux of each arc does not change anymore) until the end of the simulation.

In Section IV-E there will be further comparison results of the convergence time with the traditional algorithms. Interestingly, if there is a unique shortest path from source to sink and the dynamics stabilise, the diameter of edges on the shortest path converges to  $> 0$ , or 0 otherwise, which means, the link constraint enforces unused links to zero thereby removing such links from the design. This suggests that other paths except the optimised one can be successfully removed/deleted from the network design, thus saving cost in terms of eliminating redundant links in the network.

### B. OPTIMAL PATH AMONG PROSUMERS

The physarum algorithm is modified for energy network optimisation as follows. Nodes are defined as energy producers, consumers, and prosumers; links are energy distribution links connecting the producers to the consumers. In a P2P-ETS, energy producers could assume the role of consumers or prosumers. To be more representative of the distribution network, an additional parameter of capacity constraint,  $c_{i,j}$ , is included to ensure the congestion control of power flow on the distribution lines. A single source, single sink of the original Physarum model of (5) is also extended to provide multiple sources, multiple sinks for distributed implementation expressed in (8)

$$\frac{D_{i,j}}{A_{i,j}} c_{i,j} (p_i - p_j) = \begin{cases} -\sum_{j \in N} I_o^{i,j}, & \text{for } i = \text{source} & (8a) \\ \sum_{j \in N} I_o^{i,j}, & \text{for } j = \text{sink} & (8b) \\ 0, & \text{otherwise.} & (8c) \end{cases}$$

where  $I_o^{i,j}$  represents the energy flow between producer  $i$  and consumer  $j$ ,  $A_{i,j}$  is the link cost on the distribution link, while  $c_{i,j}$  represents the capacity of the link. This is a practical

solution to ensure that the distribution line is not congested for control purposes. In the physarum model, the flow on each link is continuous during iterations; the costs on each link is updated with the flow based on peer activities at time  $t$ . The traditional shortest path algorithms including the Dijkstra algorithm only reflect the link length attributes (to calculate the shortest path), whereas, energy routing has two attributes: energy flow and the link cost (the edge weight as a function of the flow). This resulted in most algorithms utilising the Dijkstra as the path searching algorithm and then proposing another algorithm to optimise cost as found in [21], [26]. The developed algorithm is presented in Algorithm 1.

---

#### Algorithm 1 Proposed Optimal Path Algorithm for Prosumers

---

- 1 **Input:** The graph  $G = (V, E)$ , the link cost  $A_{i,j}$ , total flow traffic  $D_{i,j}$ ,  $p_j = 0$ , the demand flow  $Q_{i,j}$ , the capacity  $c_{i,j}$ ,  $\gamma$ , and the step-size,  $\alpha$ , for all  $i \in N$
  - 2 **for**  $t > 0$ , **do**
  - 3     Calculate  $p_i$ , according to (8)
  - 4     Update the demand flow  $Q_{i,j} = \frac{D_{i,j}}{A_{i,j}} c_{i,j} (p_i - p_j)$
  - 5     Update  $D_{i,j}$  using (7),  $\forall i, j \in N$
  - 6     Go to next time slot until maximum time-step is reached
  - 7 **end**
- 

### C. ENERGY DEMAND PERFECT MATCHING

This section presents an extension to the algorithm to realise a perfect demand matching among the prosumers in the network. Consider a network of 5 consumers and 5 producers; each consumer requires a demand for 10kWh of energy. Energy demands of Consumer A can be satisfied by Producers B, C, and D. However, there is a requirement to ensure that the producer utilising the minimum cost would be appropriate to satisfy the demand among the Producers B, C, and D. Thus using the proposed slime mould algorithm and a Hungarian matching algorithm (also called the Kuhn-Munkres algorithm), the proposed optimised path algorithm is extended for prosumers matching in the network for a perfect match. This is to ensure that all the consumers are matched to the least-cost producers whilst meeting their demand requirements. The Hungarian algorithm solves the maximum-weight matching (perfect matching) in a complete bipartite graph represented with an adjacency matrix described in (1). The proposed algorithm is discussed in the next section.

### D. PROSUMERS MATCHING ALGORITHM

Let  $\bar{M}$  be a subset of  $E$  of the graph  $G$ . If any of two links of  $\bar{M}$  are disjoint in  $G$ ,  $\bar{M}$  is a matching of  $G$ , the two connected nodes of a link of  $\bar{M}$  are matched in  $G$ . Here,  $n$  consumers' energy demands are to be satisfied by  $m$

producers in the network, considering that the consumers' demands can be met by one or more producers. The aim is to ensure that at least one consumer is matched with one producer with the least path costs. Consider the bipartite subgraph  $G_{N_i, N_j}^*$ , where  $N_i = \{n_{i,1}, n_{i,2}, \dots, n_{i,n}\}$ , and  $N_j = \{n_{j,1}, n_{j,2}, \dots, n_{j,m}\}$  denotes set of consumers and producers in the network respectively. Note that a bipartite graph is a graph that its nodes can be divided into two independent and non-empty sets.

To accommodate the matching, Algorithm 1 now becomes Algorithm 2. After the initial path optimisation, Step 6 of Algorithm 2 starts by searching for a maximum matching in the subgraph  $G_{N_i, N_j}^*$ , with  $G = (X \setminus Y, E)$  of bipartite sets X, Y, if one is found, the algorithm stops, else, it proceeds to the next step, Step 9. The algorithm proceeds by initialising an empty matching (Steps 11 and 12), then search for augmenting matches in the subgraph, by flipping the matched and unmatched links along the search path (Steps 14 and 15).

**Algorithm 2** Proposed Optimal Path and Prosumers Matching Algorithm

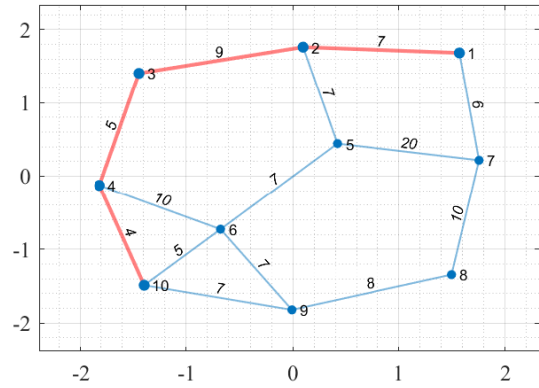
```

1 Input: The graph  $G = (V, E)$ , the link cost  $A_{i,j}$ , total
   flow traffic  $D_{i,j}$ ,  $p_j = 0$ , the demand flow  $Q_{i,j}$ ,
   the capacity  $c_{i,j}$ ,  $\gamma$ , and the step-size,  $\alpha$ , for all  $i \in N$ 
2 for  $t > 0$ , do
3   Calculate  $p_i$ , according to (8)
4   Update the demand flow  $Q_{i,j} = \frac{D_{i,j}}{A_{i,j}} c_{i,j} (p_i - p_j)$ 
5   Update  $D_{i,j}$  using (7),  $\forall i, j \in N$ 
6   if maximum matching  $\bar{M}$  in  $G_{N_i, N_j}^*$  then
7     Output  $\bar{M}$ 
8   else
9     Let  $n_i = \max\{A_{i,j} : j = 1, \dots, n\}$ ,  $n_j = 0$ 
10    Let  $R$  be a set of nodes of size  $|\bar{M}|$  in  $G_{N_i, N_j}^*$ 
11     $S := X \cap R$ ;  $\bar{T} := Y \cap R$ 
12     $\epsilon := \min\{n_i + n_j - A_{i,j} : x_i \in X \setminus S, y_j \in Y \setminus \bar{T}\}$ 
13    Update  $n_i$  and  $n_j$  using steps 14 &
14    15 respectively
15     $n_i := n_i - \epsilon$  if  $x_i \in X \setminus S$ 
16     $n_j := n_j + \epsilon$  if  $y_j \in Y \setminus \bar{T}$ 
17  end
18 end

```

**IV. NUMERICAL SIMULATION AND RESULT ANALYSIS**

To test the effectiveness of the proposed optimised path solution, the result of a network consisting of 10 prosumers is presented in Fig. 4. Simulations are performed in MATLAB using random graphs which are presumed to represent real-world distributed systems [26]. Assigning arbitrary costs to the connecting links, and setting  $\alpha = 1$  ( $\alpha$  serves as a multiplier to calculate the value of  $D_{i,j}$  from the demand flow  $Q_{i,j}$ )



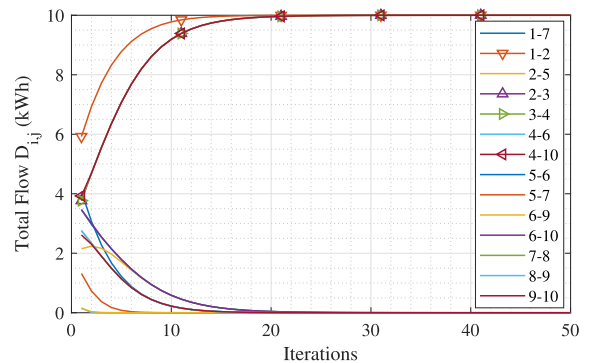
**FIGURE 4.** P2P-ETS network showing the least cost path from producer 1 to consumer 10.

and  $\gamma = 1$  ( $\gamma$  ensures faster convergence of the algorithm, the sensitivity analysis of  $\alpha$  and  $\gamma$  will be discussed further in Section IV-G). In the following, simulation results to demonstrate the performances of the system under different network scenarios are presented.

**A. SINGLE PRODUCER AND SINGLE CONSUMER**

The convergence of the system is achieved when the flow runs from the source to the destination on the shortest path and remains until the simulation ends. Since this is the single producer-consumer pair problem, the calculations of step 4 of Algorithm 1 is according to (5). The result of the total demand convergence is illustrated in Fig. 5, while Fig. 4 shows the energy network with the weights on the links.

This shows that the proposed solution solves the network problem by first locating the consumer using the least cost path, and then transmitting a demand value of 10kWh from Producer 1 to Consumer 10, as evident in Fig. 5, the total flow on edges 1 → 2 → 3 → 4 → 10 converged to 10kWh indicating the shortest path from Producer 1 to Consumer 10, while the conductivity on other edges converged to 0.



**FIGURE 5.** The demand flow on each link for a single producer to a single consumer.

**B. A SINGLE PRODUCER AND n CONSUMERS**

For the case of connecting a single producer to multiple consumers, the calculations of step 4 of Algorithm 1 is

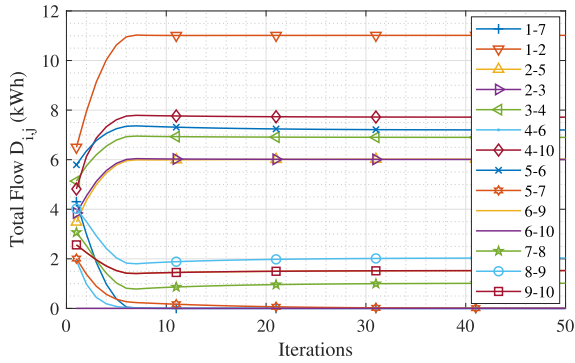


FIGURE 6. The demand flow on each link for a single producer to multiple consumers.

given as

$$\frac{D_{i,j}}{A_{i,j}} c_{i,j}(p_i - p_j) = \begin{cases} -I_o, & \text{for } i = \text{source} \quad (9a) \\ \sum_{j \in N} I_o^{i,j}, & \text{for } j = \text{sink} \quad (9b) \\ 0, & \text{otherwise.} \quad (9c) \end{cases}$$

The resulting flow plot of transmitting a demand value of 10kWh from Producer 1 to Consumer 6 and 10 of the network layout of Fig. 4 is illustrated in Fig. 6.

The plot shows the total flow on each link, with higher flows on the optimal path to the destination. For instance, the flow on links 1 → 2 → 3 → 4 → 10 and 1 → 2 → 5 → 6 are the highest. While 7 → 8 → 9 → 10 shows alternate paths (less optimal) to the destination consumers. Besides, the other links converged to 0, since they are not involved in the optimised path to the consumer.

### C. n PRODUCERS AND A SINGLE CONSUMER

In this setting, the demands are set to flow from multiple producers in the network directed to a single consumer.

With the configurations, the calculations of step 4 of Algorithm 1 is according to:

$$\frac{D_{i,j}}{A_{i,j}} c_{i,j}(p_i - p_j) = \begin{cases} -\sum_{j \in N} I_o^{i,j}, & \text{for } i = \text{source} \quad (10a) \\ I_o, & \text{for } j = \text{sink} \quad (10b) \\ 0, & \text{otherwise.} \quad (10c) \end{cases}$$

Similarly, from the total flow convergence of Fig. 7, it can be observed that the routes 1 → 2 → 3 → 4 → 10 and 7 → 8 → 9 → 10 converges to higher flow value than the other links 12 → 5 → 6. While other unused links converged to 0.

### D. n PRODUCERS AND n CONSUMERS

Further, setting some prosumers as producers and some as consumers, and transmitting a demand value of 10kWh among them, in this case, two consumers. Similarly, the calculations of step 4 of Algorithm 1 is according to (8). The convergence result of Fig. 8 demonstrates that the proposed optimised path algorithm solves the network problem by

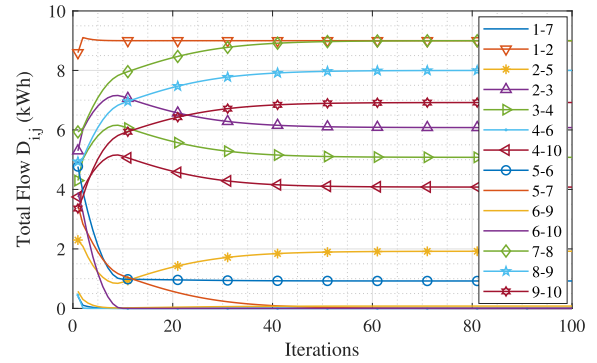


FIGURE 7. The demand flow on each link for multiple producers to a single consumer.

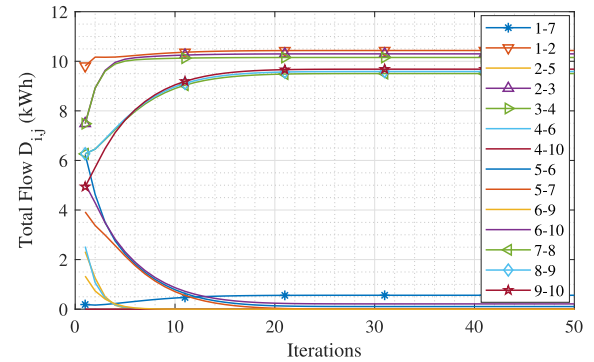


FIGURE 8. The demand flow on each link for multiple producers to multiple consumers.

transmitting demand value of 10kWh each from Producers 1 and 7 to Consumers 4 and 10 using the least cost path. i.e., using edges 1 → 2 → 3 → 4 and 7 → 8 → 9 → 10, while other links converged to 0.

### E. COMPARISON WITH OTHER SHORTEST-PATH ALGORITHMS

Whilst this work is envisioned for energy networks, the solution therein can be proven for other cyber-physical networks requiring path finding algorithms. In this section, the proposed shortest path algorithm is compared to the traditional path-finding algorithms including ACO, Dijkstra [12], and IPPA (improved physarum polycephalum algorithm) [14] using different datasets shown in Table 1. The efficiency of the algorithm is tested over a network with random and varying topologies, with network sizes ranging from 15 to 2000 nodes [14], and link costs ranging from 1 to 100 and analysed based on the algorithm execution time. The reported probability is the probability of establishing a connection between the nodes.

#### 1) COMPARISON BASED ON EXECUTION TIME

The performance of an algorithm is mostly determined by its execution time and its accuracy in solving the proposed problem. The accuracy of the algorithm has been confirmed from the previously presented cases, which located the destination prosumer utilising the least-cost path and routed the

TABLE 1. The test data sets for the comparison study.

No. of Nodes	No. of Links	Probability
15	51	0.25
30	57	0.06
50	116	0.1
80	229	0.03
100	218	0.02
200	660	0.02
250	879	0.01
400	1592	0.01
500	1661	0.006
800	3446	0.005
1000	3324	0.005
1200	4791	0.005
1500	5614	0.003
1800	5629	0.002
2000	6393	0.002

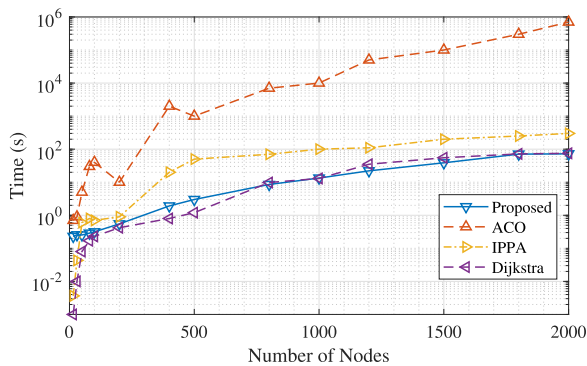


FIGURE 9. The comparison result of other algorithms using the execution time.

demand successfully. Thus the execution time of its shortest path property is compared against other algorithms illustrated in Fig. 9, including the Dijkstra, ACO, and IPPA. In IPPA, the authors combined the original slime mould algorithm with a parameter called Energy, which quantifies the energy provided and consumed by the tube. However, in this paper, we combined the capacity constraints to the original slime mould algorithm to model the distribution network problem, by setting a limit to the amount of power flow, thereby controlling congestion and reducing the algorithm execution time. Algorithm response time would affect network application performances, especially in a large network as P2P-ETS. It can be observed that with a fewer number of nodes, the Dijkstra is faster than the developed algorithm. However, as the number of nodes increased from 100, they both have similar execution time, which is faster than both the IPPA and the ACO algorithms.

Moreover, in the Dijkstra algorithm, each link is only associated with one criterion; length, and there is no equivalent attribute like ‘flow’ evident in the Physarum model reacting to the change of the link cost. As a result, many classical algorithms for the cyber-physical network problem

must have two separate processes/algorithms: path finding and flow optimisation. In contrast, the presented Physarum-based algorithm solves both problems simultaneously; once the link cost  $A_{i,j}$  is updated, with the help of (4), the flow is redistributed and reallocated dynamically in the next iteration. Physarum algorithm is suitable for solving the network optimisation problems in a dynamic environment because it can utilise the computational (or intermediate) results in the previous iterations and respond to the changes by adjusting the flow. The scalability result of the algorithm by utilising 2000 prosumers is equally deduced from Fig. 9.

F. CONGESTION CONTROL IN DISTRIBUTION NETWORK

While the previous sections are motivated by the desire to address the problem of the optimal least-cost path between producers and consumers, the solution proffer can be extended to realise optimal path for power flow in a capacitated network. Thus, the power flow optimisation is defined as a minimum cost flow problem relating to both the shortest optimised path and maximum flow capacitated problem [21]. This section’s focus is on an optimal path for energy transfer in a capacitated network for congestion control. As defined in Section III that each link is characterised by two non-negative attributes  $c_{i,j}$ , capacity, and  $A_{i,j}$ , link cost. Given a source node  $i \in V$ , a sink node  $j \in V$ . Function  $\Phi_{i,j}(x_{i,j})$  represents the optimisation problem defined in (2a).  $c_{i,j}$  is the capacity of the link  $i,j$  that ensures the distribution line is not congested, as well as to maintain a maximum flow of power for control purposes. The traditional solution to solving the maximum cost flow problem is the successive shortest path algorithm, which has been discussed to have higher computational time. Thus, using the proposed slime mould solution, by setting the capacity of the distribution line, the maximum flow problem is remodelled to cope with congestion on the lines. For instance, Fig. 10 illustrates the same test graph but comprising of the link cost as well as the capacity of each link.

In Fig. 11, the convergence result of the optimised path from Producer 1 to Consumer 10 is shown, which is consistent with Fig. 5. However, it can be observed that, while all other links not on the optimal path converged to zero in Fig. 5,

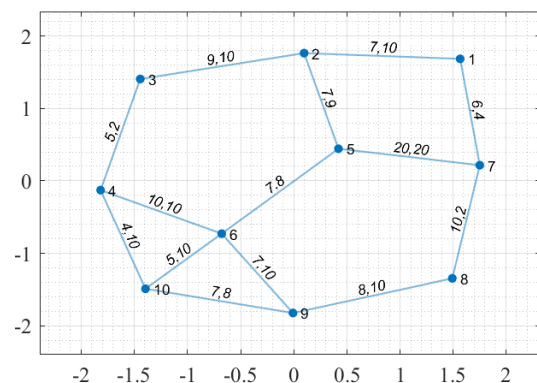


FIGURE 10. P2P-ETS network with capacity constraints and link costs.



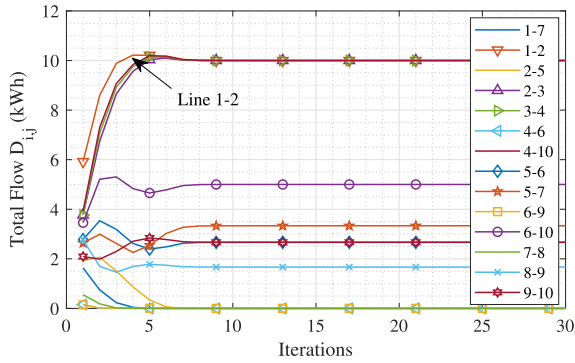


FIGURE 11. The demand flow on each link for the optimal path for energy transfer.

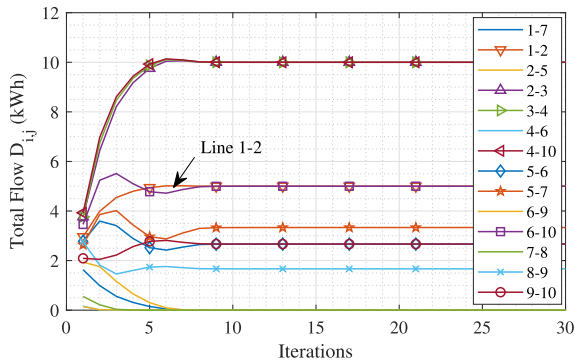


FIGURE 12. The demand flow on each link for the optimal path for energy transfer with capacity constraints.

Fig. 11 provided alternate paths in cases of fault with the main optimised path. In addition, limiting the capacity installed on link 1  $\rightarrow$  2 from 10kWh to 5kWh, it can be observed that the power flow on the link has been suppressed to 5kWh which is different from Fig. 11 due to the restriction on the lines as shown in Fig. 12. This property is useful in coping with congestion on the distribution lines for control purposes.

G. SENSITIVITY ANALYSIS

To further analyse the performance of the algorithm, a sensitivity analysis is done by varying the variables  $\alpha$  and  $\gamma$ . Fig. 13 shows the corresponding result. It can be observed that the time to convergence decreases when  $\gamma$  is 1, compared to when  $\gamma$  is 0.01 and 0.1 irrespective of the corresponding  $\alpha$  value. The selection of  $\alpha$  and  $\gamma$  for the test cases is based on this property.

H. ROUTING POWER COST COMPARISON USING IEEE 39 BUS

For further analysis of the proposed solution, this subsection shows a comparison result with study [26] using an IEEE 39 bus. The IEEE 39 bus system represents an approximation of an electrical power system with 39 buses, comprising of 46 lines, 10 power sources, and 19 loads. Representing the power sources as producers and the loads as consumers, Fig. 14 shows the IEEE 39 bus test including the line

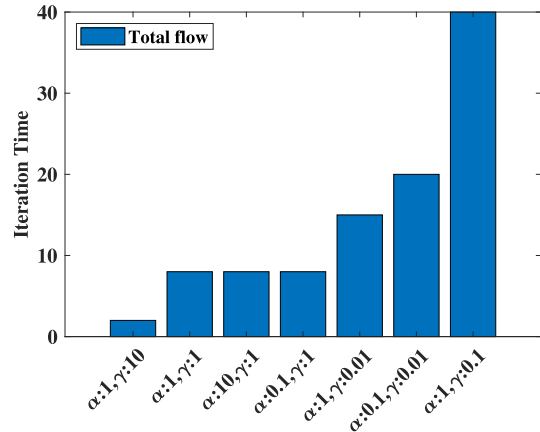


FIGURE 13. Sensitivity analysis for varying values of  $\alpha$  and  $\gamma$ .

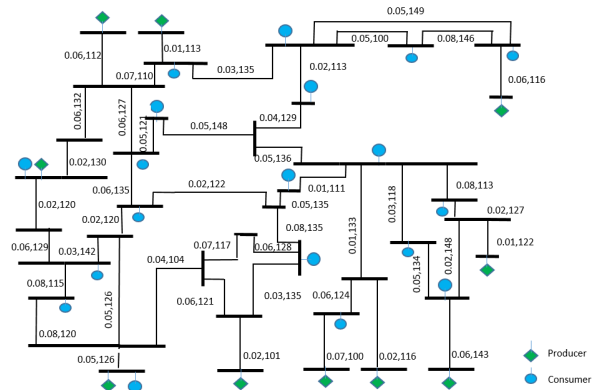


FIGURE 14. IEEE 39-bus test case for energy routing.

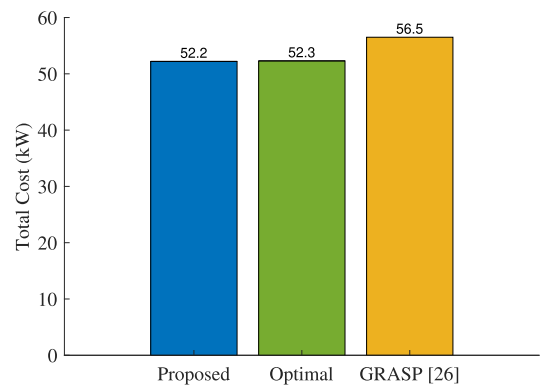


FIGURE 15. Comparison of the power costs for the Proposed, GRASP, and Optimal for IEEE 39-bus test network.

capacities and costs of the links. A total power generated by all producers in the networks equal 570kW, while the 19 consumers demanded a total of 570kW of power. To compare the effectiveness of the proposed slime mould solution, the line cost here represents the total power loss as modelled in greedy smallest-cost-rate path first (GRASP) [26]. Fig. 15 shows the result of GRASP, optimal, and the proposed solution. It can be observed that the proposed slime mould approach yielded the least total cost in routing the power from the producers to the consumers in the network.

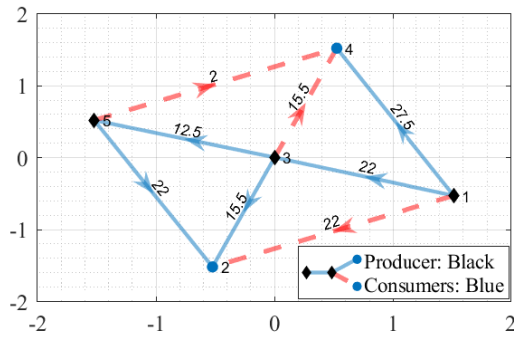


FIGURE 16. P2P-ETS network showing the paired two producers to three consumers.

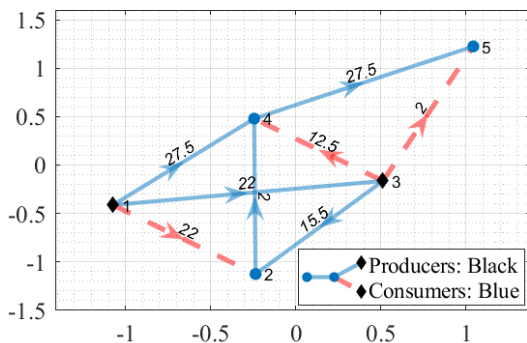


FIGURE 17. P2P-ETS network showing the paired three producers to two consumers.

**I. NUMERICAL SIMULATION AND RESULT OF THE PERFECT MATCHING ALGORITHM**

The effectiveness of the proposed perfect matching algorithm, i.e. Algorithm 2, is presented as follows. A situation involving an odd number of actors is first considered, with a focus on one-to-many or many-to-one matching, i.e. a producer to many consumers, as the case arises. This reflects a real-world scenario where the demands of a consumer are satisfied by one or two producers and vice versa. Utilising the average separation distances between houses in the UK [27], ranging from a minimum of 1m of a bed dwelling to 27.5m of a three/four storey building, the distances between the actors are determined accordingly and illustrated in Figs 16 and 17.

Fig. 16 reflect the case of three producers to two consumers, the producers are  $N_j = \{1, 3, \&5\}$ , while the consumers are  $N_i = \{2, \&4\}$ . It can be observed that the algorithm successfully matched peers with the least-cost to minimise the overall network cost. For instance, Producers 3 and 5 to Consumer 4, while Consumer’s 2 demands are met by Producer 1.

Similarly, Fig. 17 shows the result of two producers  $N_j = \{1, \&3\}$ , to three consumers  $N_i = \{2, 4, \&5\}$ . Demands of Consumer 2 is satisfied by Producer 1, while demands of Consumers 4 and 5 are both supplied by Producer 3 to save costs as well as to ensure all demands are met in the network.

To further quantify the effectiveness of the proposed optimised path algorithm, Fig. 18 reflect the saved costs from the two cases previously considered. The non-optimal path is the path the peers would have taken without the algorithm, while

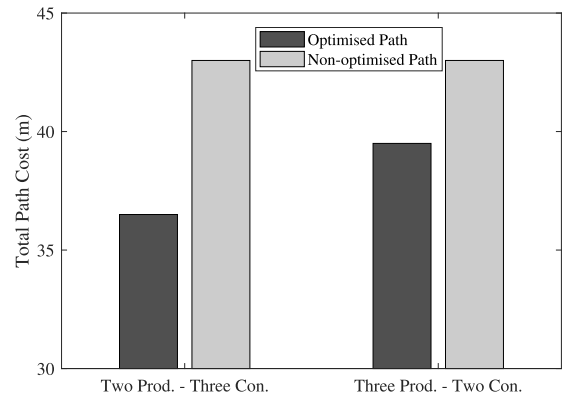


FIGURE 18. Cost savings for the optimised path and non-optimised path.

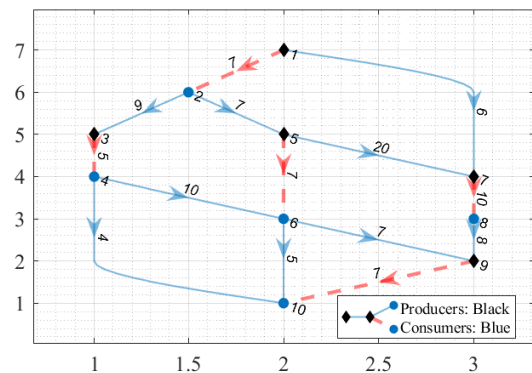


FIGURE 19. P2P-ETS network showing the paired producers and consumers.

the optimal path is the path taken as a result of the proposed algorithm. A total of 15% cost was saved for the case of establishing an optimised path between two producers and three consumers, while 8% was saved for the case of three producers to two consumers. This cost savings reflect the variety of roles of participants in the network. For instance, the cost savings increases when there are more consumers to buy energy than when there are more producers than consumers.

**J. ONE-TO-ONE MATCHING**

The numerical example presented in Section IV-I considers a case of an odd number of actors. Here, we focus on an even number of producers and consumers in the network to produce a one-to-one matching. Using the same set numbers of prosumers from Section IV, but assigning 5 producers and 5 consumers, Fig. 19 is a resulting plot of the matching algorithm. The producers are  $N_j = \{1, 3, 5, 7, \&9\}$ , while the consumers are  $N_i = \{2, 4, 6, 8, \&10\}$ . It can be observed that the algorithm successfully matched the peers with the least-network cost, for instance, Producer 1 to Consumer 2, Producer 3 to Consumer 4, Producer 5 to Consumer 6, Producer 7 to Consumer 8, Producer 9 to Consumer 10. Although, when observed closely, and based solely on the link costs, Consumer 8 could have been paired with Producer 9, however, if this were to be true, it would result in no pairing (or higher cost) for Consumer 10.

As previously established that energy loss is mostly due to long distance-transmission, thus producing power locally and matching supply and demand can minimise the losses from transportation for economic and environmental benefits. Invariably, matching local energy demand, would lower the control effort of the overall power system. Furthermore, distribution networks are vulnerable to a variety of faults, thus by providing an alternate route for power distribution, disruption to the consumers would be minimised making the electric grid more resilient.

## V. CONCLUSION

This paper addressed the problem of matching prosumers in MG to facilitate P2P-ETS. Two main issues were addressed; Firstly, as the cost of energy has a direct correlation with the distance between energy producer and consumer, a path-optimised system was developed for energy routing. This was shown to have up to 15% cost savings as compared to a non-optimised path. Also, the execution time for the developed algorithm, as the number of peers increases (15 – 2000), is reduced as compared to other traditional algorithms, which is highly desirable in a large network as P2P-ETS. Secondly, the proposed solution was applied to a maximum flow capacity problem in the energy distribution network to reduce congestion on the power distribution lines. This result shows the secure operation and control of the grid. Future work will look at including additional constraints such as the cost of renewable energy generation, different generation capacities from prosumers, energy storage systems, and electric vehicles in the problem formulations.

## REFERENCES

- [1] O. Jogunola, A. Ikpehai, K. Anoh, B. Adebisi, M. Hammoudeh, S.-Y. Son, and G. Harris, "State-of-the-art and prospects for peer-to-peer transaction-based energy system," *Energies*, vol. 10, no. 12, p. 2106, Dec. 2017.
- [2] N. T. Mbungu, R. M. Naidoo, R. C. Bansal, and V. Vahidinasab, "Overview of the optimal smart energy coordination for microgrid applications," *IEEE Access*, vol. 7, pp. 163063–163084, 2019.
- [3] O. Jogunola, M. Hammoudeh, B. Adebisi, and K. Anoh, "Demonstrating blockchain-enabled Peer-to-Peer energy trading and sharing," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2019, pp. 1–4.
- [4] E. Oh and S.-Y. Son, "Pair matching strategies for prosumer market under guaranteed minimum trading," *IEEE Access*, vol. 6, pp. 40325–40333, 2018.
- [5] O. Jogunola, B. Adebisi, K. Anoh, A. Ikpehai, M. Hammoudeh, G. Harris, and H. Gacanin, "Distributed adaptive primal algorithm for P2P-ETS over unreliable communication links," *Energies*, vol. 11, no. 9, p. 2331, Sep. 2018.
- [6] O. Jogunola, B. Adebisi, K. Anoh, A. Ikpehai, and M. Hammoudeh, "Adaptive routing algorithm for information management in distributed microgrids in smart grid," in *IFAC Workshop Control Smart Grid Renew. Energy Syst.* Geneva, Switzerland: IFAC, Jun. 2019.
- [7] D. Gregoratti and J. Matamoros, "Distributed energy trading: The multiple-microgrid case," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2551–2559, Apr. 2015.
- [8] N. Liu, J. Wang, and L. Wang, "Distributed energy management for interconnected operation of combined heat and power-based microgrids with demand response," *J. Modern Power Syst. Clean Energy*, vol. 5, no. 3, pp. 478–488, May 2017.
- [9] T. Yang, J. Lu, D. Wu, J. Wu, G. Shi, Z. Meng, and K. H. Johansson, "A distributed algorithm for economic dispatch over time-varying directed networks with delays," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5095–5106, Jun. 2017.
- [10] Y. Zhou, S. Ci, H. Li, and Y. Yang, "A new framework for peer-to-peer energy sharing and coordination in the energy Internet," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.
- [11] Ofgem. (2018). *Breakdown of an Electricity Bill*. Accessed: Jun. 22, 2019. [Online]. Available: <https://www.ofgem.gov.uk/data-portal/breakdown-electricity-bill>
- [12] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.
- [13] R. Bellman, "On a routing problem," *Quart. Appl. Math.*, vol. 16, no. 1, pp. 87–90, 1958.
- [14] X. Zhang, Q. Wang, A. Adamatzky, F. T. Chan, S. Mahadevan, and Y. Deng, "An improved *Physarum polycephalum* algorithm for the shortest path problem," *Sci. World J.*, vol. 2014, Mar. 2014, Art. no. 487069.
- [15] C. Gao, C. Yan, Z. Zhang, Y. Hu, S. Mahadevan, and Y. Deng, "An amoeboid algorithm for solving linear transportation problem," *Phys. A, Stat. Mech. Appl.*, vol. 398, pp. 179–186, Mar. 2014.
- [16] X. Zhang, Q. Wang, F. T. Chan, S. Mahadevan, and Y. Deng, "A *physarum polycephalum* optimization algorithm for the bi-objective shortest path problem," *Intl. J. Unconventional Comput.*, vol. 10, pp. 143–162, Feb. 2014.
- [17] C. Gao, S. Chen, X. Li, J. Huang, and Z. Zhang, "A *physarum*-inspired optimization algorithm for load-shedding problem," *Appl. Soft Comput.*, vol. 61, pp. 239–255, Dec. 2017.
- [18] G. A. Pagani and M. Aiello, "From the grid to the smart grid, topologically," *Phys. A, Stat. Mech. Appl.*, vol. 449, pp. 160–175, May 2016.
- [19] S. Hemalatha and P. Valsalal, "Identification of optimal path in power system network using bellman ford algorithm," *Model. Simul. Eng.*, vol. 2012, p. 28, Sep. 2012.
- [20] L. Cuadra, M. Pino, J. Nieto-Borge, and S. Salcedo-Sanz, "Optimizing the structure of distribution smart grids with renewable generation against abnormal conditions: A complex networks approach with evolutionary algorithms," *Energies*, vol. 10, no. 8, p. 1097, Jul. 2017.
- [21] P. H. Nguyen, W. L. Kling, and J. M. A. Myrzik, "Power flow management in active networks," in *Proc. IEEE Bucharest PowerTech*, Jun. 2009, pp. 1–6.
- [22] A. Ghasemkhani, A. Anvari-Moghaddam, J. M. Guerrero, and B. Bak-Jensen, "An efficient multi-objective approach for designing of communication interfaces in smart grids," in *Proc. IEEE PES Innov. Smart Grid Technol. Conf. Eur. (ISGT-Eur.)*, Oct. 2016, pp. 1–6.
- [23] C. Zhang, J. Wu, Y. Zhou, M. Cheng, and C. Long, "Peer-to-Peer energy trading in a microgrid," *Appl. Energy*, vol. 220, pp. 1–12, Jun. 2018.
- [24] D. Sohler, G. Georgiadis, S. Clavière, M. Papatriantafylou, and A. Bui, "Physarum-inspired self-biased walkers for distributed clustering," in *Proc. Int. Conf. Princ. Distrib. Syst.* Springer, 2012, pp. 315–329.
- [25] A. Tero, R. Kobayashi, and T. Nakagaki, "A mathematical model for adaptive transport network in path finding by true slime mold," *J. Theor. Biol.*, vol. 244, no. 4, pp. 553–564, Feb. 2007.
- [26] Z. Jiang, V. Sahasrabudhe, A. Mohamed, H. Grebel, and R. Rojas-Cessa, "Greedy algorithm for minimizing the cost of routing power on a digital microgrid," *Energies*, vol. 12, no. 16, p. 3076, Aug. 2019.
- [27] Worcester.co.uk. (2004). *Residential Design Guide*. Accessed: Nov. 11, 2019. [Online]. Available: [https://www.worcester.gov.uk/documents/10499/377508/siting\\_of\\_buildings\\_to\\_buildings.pdf/70d8e22d-f15a-41c4-b552-407f09eb279b](https://www.worcester.gov.uk/documents/10499/377508/siting_of_buildings_to_buildings.pdf/70d8e22d-f15a-41c4-b552-407f09eb279b)



**OLAMIDE JOGUNOLA** (Student Member, IEEE) is currently a Research Associate with the Department of Engineering, Manchester Metropolitan University, working on Energy-IQ, a UK-Canada power forward smart grid demonstrator project. Her research interests include SG, P2P communication technology, the IoT, peer-to-peer energy trading, network optimization, and artificial intelligence for energy market. Dr. Jogunola was a recipient of the School of Engineering, Manchester Metropolitan University Ph.D. Student ship on an EPSRC UK-Korea-funded project: P2P-ETS system. She was previously involved in an Horizon 2020 Smart Cities and Communities programme; Triangulum, funded by the European Commission.



**WEIZHUO WANG** received the B.Eng., M.Sc. (Eng.), and Ph.D. degrees. He is currently a Senior Lecturer in mechanical engineering with Manchester Metropolitan University, U.K. He is specialising in modeling and simulation in engineering mechanics. He has been working in various multi-disciplinary projects, e.g., work-package coordinator in a EU project Advanced Dynamic Validations using Integrated Simulation and Experimentation (ADVISE). He has been also

working with industries on Knowledge Exchange Partnership (KTP) projects in the area of applying data analytics, digital twin and optimisation. Dr. Wang is a member of the Institute of Physics (MInstP) and a Fellow of the Higher Education Academy (FHEA).



**BAMIDELE ADEBISI** (Senior Member, IEEE) received the master's degree in advanced mobile communication engineering and the Ph.D. degree in communication systems from Lancaster University, U.K., in 2003 and 2009, respectively. He is currently a Full Professor of intelligent infrastructure systems with the Metropolitan University, Manchester, UK. He has worked on several commercial and government projects focusing on various aspects of smart infrastructure systems.

He is particularly interested in collaborative research and development of technologies for electrical energy monitoring/management, transport, water, home automation, the IoTs, cyber physical systems, and critical infrastructures protection. He is a Chartered Engineer and a member of IET.

• • •