



Flocking-based decentralised double auction for P2P energy trading within neighbourhoods

Kosala Yapa Bandara^{*}, Subhasis Thakur, John Breslin

Insight Centre for Data Analytics, National University of Ireland-Galway, Ireland

ARTICLE INFO

Keywords:

Decentralised double auction
Flocking
Peer-to-Peer
Trading

ABSTRACT

Renewable energy sources are highly unreliable and create unreliable energy surplus and energy demands. These energy surpluses and demands can be traded frequently within neighbourhoods. This paper presents a trading algorithm to trade unreliable renewable energy frequently within neighbourhoods. We considered the flocking behaviour of birds and distributed double auction, and developed a trading algorithm to perform peer-to-peer (P2P) energy trading within neighbourhoods. Peers are prosumers which can be energy co-operatives, micro-grids, electric vehicles, charging stations, etc. We implemented our flocking-based decentralised double auction algorithm in Java and tested using a residential PV data set from California state in the USA. Our mathematical analysis shows that the peers are converging to average positions of neighbourhoods while performing P2P trading. Our experimental results showed that the proposed algorithm has a higher success rate and faster convergence to the success rate than the centralised double auction and distributed double auction presented in the literature. Moreover, nodes are converging to neighbourhoods. This guarantees trading within neighbourhoods. Further, we have observed that successful trading reaches around 80%.

1. Introduction

Renewable energy generators are not reliable because they are highly affected by changing environmental factors such as the amount of sunshine, wind speed, etc. Moreover, they do not naturally have large on-site energy storages hence their output is typically referred to as non-dispatchable[1]. Prosumers are proactive consumers with distributed energy resources, actively managing their consumption, production and storage of energy [2,3]. P2P energy trading among local prosumers is emerging as a promising method to manage unreliable and distributed energy resources[4] while offering three distinct value streams: energy matching, uncertainty reduction, and preference satisfaction [2].

P2P Energy trading goes from prosumer to prosumer rather than large commercial enterprise to consumer, and prosumers can sell their excess power for profit[5]. Authors in [5,6] discuss distributed double auction for P2P energy trading, and prosumers delegate their energy requirements between randomly selected neighbours. This random selection improves the anonymity of trading. The unsuccessful energy requirements will be further delegated to randomly selected neighbours

of the current prosumer. Therefore, the energy requirement of a prosumer can be traded outside of its neighbourhood. That is trading within the neighbourhood is not guaranteed. A review of existing P2P energy trading projects and auction mechanisms are presented in [4,7]. A Bayesian Equilibrium solution for energy trading is presented in [8]. However, the current literature regarding the random request delegation for distributed trading does not theoretically describe and implement guaranteed trading within the neighbourhoods, and that is the main objective of this research. The Joule effect in the transmission cables accounts for energy losses of about 2.5% while the losses in transformers range between 1% and 2% [9], hence guaranteed P2P trading within a neighbourhood is important.

The flocking behaviour of birds and the convergence of birds flocking are explained using cohesion, separation and alignment properties in [10–12]. This paper uses properties of the flocking behaviour of birds and defines the behaviour of prosumers to perform anonymous, secure and guaranteed P2P energy trading within neighbourhoods. Our decentralised double auction uses McAfee's double auction algorithm [13]. The proposed P2P energy trading was implemented and tested for

Abbreviations: P2P, Peer-to-Peer; PV, Photovoltaic; FDDA, Flocking-based Decentralised Double Auction; DDA, Distributed Double Auction; CDA, Centralised Double Auction.

^{*} Corresponding author.

E-mail address: kosala.yapa@nuigalway.ie (K.Y. Bandara).

<https://doi.org/10.1016/j.ijepes.2021.106766>

Received 22 April 2020; Received in revised form 9 October 2020; Accepted 28 December 2020

Available online 2 February 2021

0142-0615/© 2021 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

success rates, the amount of traded energy and convergence at decentralised neighbourhoods. This paper presents following contributions:

- Guaranteed P2P trading between prosumers within neighbourhoods.
- The proposed flocking-based decentralised double auction preserves the anonymity of the buyer and seller hence provide improved security and trust for P2P trading.
- The proposed approach shows better trading outcomes compared to distributed double auction and centralised double auction presented in the literature.

In this research, prosumers generate their energy requirements as Buying or Selling requests. A Buying request finding a successful Selling request is called successful trading. That is, one successful trading has two traded requests.

This paper was organised as follows: We discussed the relevant literature in Section 2 and introduced the flocking-based decentralised double auction for P2P energy trading in Section 3. In Section 4, we discussed the distributed double auction. The implementation, experimental evaluation, and results were discussed in Section 5. Finally, in Section 6, we presented findings and future research.

2. Related literature

Distributed Energy Resources (DERs) are being integrated into the energy market by transforming the centralised and unidirectional market into a decentralised market while transitioning customers into prosumers [14]. The direct energy trading between consumers and prosumers without the use of an intermediary is called P2P energy trading which is usually implemented within a local electricity distribution system [15]. A P2P system is a self-organising decentralised system for shared usage of resources avoiding central services [16,17]. The P2P architecture is emerging as a potential and promising architecture for decentralised energy trading in local energy markets [14,15].

The P2P trading enables multi-directional trading in local communities [15]. Energy trading platforms such as vandebron¹, Piclo², etc. act as an energy supplier who link consumers and DERs, and balance the whole market [18]. Consumers can select and prioritize from which generators to buy electricity. Moreover, [18] suggests P2P energy trading within a micro-grid, within a CELL(multi-micro-grids), and among CELLS(multi-cells). A four-layer system architecture is proposed for P2P energy trading in [15]. They are two dimensional power grid layer, ICT layer, control layer and business layer. A P2P control architecture is detailed in [14] discussing a case study in Nepal.

The P2P energy trading enables trading locally generated renewable energy among local customers [2,3,1,7,18]. A Bayesian Equilibrium solution for energy trading considering utility, power loss and Bayesian strategic bidding are described in [8]. However, many of the trails have focused on the business models and it is still lacking necessary mechanisms to ensure energy trading within local prosumers [4]. In [10], Reynolds introduced three heuristic rules to explain the flocking behaviour of birds however these rules are interpreted broadly because they are not theoretically defined. A formalization of Reynolds's boids is presented in [19] while modelling the behaviour of a living being. A theoretical framework for the design and analysis of distributed flocking algorithms is presented in [12]. The leaderless coordination is theoretically defined in [20] where birds update their velocities by averaging them out over their nearest neighbours. The repeated averaging of velocities enables each bird to eventually converge to a fixed speed and heading [11].

A double auction mechanism with dominant strategies for both buyers and sellers are presented in [13]. Authors in [5,6] have discussed

distributed double auction and the use of blockchain technology for P2P energy trading. Blockchain transactions of energy surplus and deficiency are sent to neighbours. However, guaranteed trading within the neighbourhood has not yet been addressed in the literature.

3. Flocking-based decentralisation

The flocking-based decentralisation enables prosumers to delegate their energy requirements only between neighbours during continuous auctions.

3.1. Flocking behaviour of prosumers

This section defines and explains the flocking behaviour of prosumers to assure trading within neighbourhoods, improve anonymity and improve the security for P2P energy trading. In [10], Reynolds introduced three heuristic rules to explain the flocking behaviour of birds. As in [12], Cohesion - attempt to stay close to nearby flockmates, Alignment - attempt to match velocity with nearby flockmates, and Separation - avoid collisions with nearby flockmates. We discuss the cohesion, alignment and separation properties of flocking behaviour and their applicability for prosumers to perform P2P energy trading. We define a prosumer as a node in a two-dimensional space. The position of a prosumer is determined by X and Y coordinates in the Cartesian Coordinate System. That is, the internal state of a node consists of at least its position $p \in \mathcal{P}$, where \mathcal{P} is a position in the Cartesian Coordinate System.

Definition 1. The internal state $q \in \mathcal{Q}$ of a prosumer $n \in \mathcal{N}$ is defined by (1), where $p \in \mathcal{P}$ is its position which is defined by X and Y coordinates in the Cartesian Coordinate System, l is its label, and r^* is a list of requests as $r \in \mathcal{R}$.

$$\mathcal{Q} = \{q_n | q_n = \langle p, l, r^* \rangle\} \quad (1)$$

The position of each node is updated based on the average position of the neighbourhood. The neighbours of node i at a given time t , are those nodes that are in a circle of pre-defined radius r centred around the current position of node i . The position of node i (n_i), written in p_i evolves in discrete-times in accordance with the model defined by (2),

$$p_i(t+1) = \frac{1}{2} (p_i(t) + \langle p_i(t) \rangle_{r'}) \quad (2)$$

where t is a discrete-time index taking values in the non-negative integers $0, 1, 2, 3, \dots$, and $\langle p_i(t) \rangle_{r'}$ is the average of the positions of n_i 's neighbours at time t in radius r' as defined in (3).

$$\langle p_i(t) \rangle_{r'} = \frac{1}{M_i(t)} \left(\sum_{j \in \mathcal{N}_i(t)} p_j(t) \right) \quad (3)$$

where $\mathcal{N}_i(t)$ is the set of neighbours and $M_i(t)$ is the number of neighbours of node i (n_i) at time t . The internal state of node i (n_i), that is q_{n_i} is changed at discrete-times.

Definition 2. The cohesion function f for a node n_i is a process which is defined by the following steps; step1: select a request $r \in \mathcal{R}$ from n_i , step2: find the neighbourhood of n_i , step3: delegate the request to a randomly selected neighbour, step4: move n_i towards the centre of the neighbourhood as defined in the Eq. (2), and step5: repeat the above steps for all the requests in n_i .

A new $\langle p_i(t) \rangle_{r'}$ is calculated for each request delegation which we defined as,

$$f'(r_k, t) = \bar{v}_k \quad (4)$$

where $\bar{v} = \langle p_i(t) \rangle_{r'}$ and $k > 0$; k is a positive integer. The average

¹ <https://vandebron.nl/>

² <https://piclo.energy/>

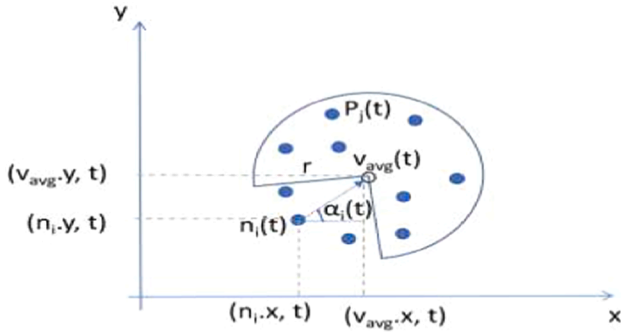


Fig. 1. Alignment behaviour.

position \bar{V} for k number of requests is,

$$\bar{V}_k = \frac{1}{k} \sum_{m=1}^k \bar{V}_m \quad (5)$$

where m is a positive integer. This further justifies, the node n_i converges to the average \bar{V}_k when delegating requests to neighbours as defined in 6.

$$\lim_{j \rightarrow k} f(n_{ij}) \rightarrow \bar{V}_k \quad (6)$$

where f is the cohesion function as defined in definition 2. Time to converge is proportional to the number of requests defined in k. The convergence is important because that guarantees requests are delegated within neighbouring nodes all the time and that is one of the main contributions of this paper. We use radius r to define the neighbourhood, and the distance between nodes is calculated to find neighbour nodes. The energy loss of electric power systems is more complex and that involves transformers, cables, etc. Transmitting electricity over distance and via networks involves energy loss. As the Joule effect described in the introduction section, transmission cables have a higher energy loss than transformers[9]. However, for simplicity, in this research, we assume the distance between nodes is proportional to energy loss. This convergence promises the minimum energy loss which is an interesting feature for P2P energy trading within local communities in low voltage networks.

Here, the Alignment $\alpha_i(t)$ is defined as illustrated in Fig. 1. The node n_i finds its neighbourhood for each request delegation at time t as in step 2 of definition 2. Then the average position $V_{avg}(t)$ is calculated as in Eq. 3.

$$V_{avg}(t) = \frac{1}{M_1(t)} \left(\sum_{j \in \mathcal{N}_i(t)} P_j(t) \right) \quad (7)$$

As in Fig. 1, node $n_i(t)$ move towards $V_{avg}(t)$ as defined in the Eq. 2 while keeping the alignment $\alpha_i(t)$ where

- $n_i(t) = ((n_i.x, t), (n_i.y, t))$
- $V_{avg}(t) = ((V_{avg}.x, t), (V_{avg}.y, t))$

Blockchain-based smart contract implementation of 7 and $\alpha_i(t)$ alignment verification can strengthen the security of request delegation among neighbours and P2P trading. For example, a prosumer who is moving towards the average position of a neighbourhood to delegate a request is paying a less fee than other positions. However, secure request delegation is not part of objectives here and we will address it in our future work.

We find a collision neighbourhood $\mathcal{N}_{collision}$ where nodes in this area have a risk of collision. Each prosumer should see their own local prosumers in $\mathcal{N}_{collision}$ and steer away from the centre of $\mathcal{N}_{collision}$ as defined

in the Algorithm 1.

Algorithm 1. :Prosumer Separation from Collisions

Input: $n \in \mathcal{N}$
Output: $n \in \mathcal{N}$ free from collisions
1: $r \leftarrow \text{collisionRadius}()$
2: $\mathcal{N}_{collision} \leftarrow \text{collisionNeighbours}(\mathcal{N}, n, r)$ 3: while ($\mathcal{N}_{collision} \cdot \text{isEmpty}()$) do
4: $\text{centre} \leftarrow \text{findCentre}(\mathcal{N}_{collision})$
5: $n \leftarrow \text{separate}(n, \text{centre})$
6: $\mathcal{N}_{collision} \leftarrow \text{collisionNeighbours}(\mathcal{N}, n, r)$
7: end while

Algorithm 1 can be explained as follows:

- For node $n \in \mathcal{N}$, find neighbours in the collision radius r which we call $\mathcal{N}_{collision}$.
- Find the centre of all nodes in $\mathcal{N}_{collision}$.
- Node n steer away from the centre recalculating x and y coordinates with r distance.
- Repeat above steps for node n until $\mathcal{N}_{collision}$ is empty.

Separation is useful to differentiate prosumers while trading and simulate trading behaviours.

3.2. Decentralisation of prosumers for P2P trading

Reynolds' rules are not mathematically defined, and it is unclear when and how each rule is applied[12]. Here, we use the internal state of each node as in Eq. (1), average position as in Eq. (2), cohesion function as in Eq. (6), alignment as in Eq. (7), separation as in Algorithm 1, and defined an Algorithm 2 to run double auctions to trade local energy demands and surpluses frequently within dynamically created decentralised neighbourhoods. The unsuccessful trading requests regarding energy surplus and demand will be addressed using the Utility grid which we do not discuss in this paper.

Algorithm 2. : Flocking-based Decentralised Double Auction (FDDA)

Input: \mathcal{N} , Each node $n \in \mathcal{N}$ has a set of \mathcal{R} requests
Output: Traded transactions
for each t do
2: while not converged do
 $\mathcal{N} \leftarrow \text{nodesWithRequests}(\mathcal{N})$
4: for each $n \in \mathcal{N}$ do
 $\text{neighbours} \leftarrow \text{findNeighbours}(n)$
6: $\mathcal{R} \leftarrow \text{allRequests}(n)$
for each request $\in \mathcal{R}$ do
8: $\text{centre} \leftarrow \text{findCentre}(\text{neighbours})$
 $\text{randomNode} \leftarrow \text{select}(\text{neighbours})$
10: $\text{randomNode} \leftarrow \text{delegate}(\text{request})$
 $n \leftarrow \text{moveTowardsCentre}(n, \text{centre})$
12: $\text{neighbours} \leftarrow \text{findNeighbours}(n)$
end for
14: $n \leftarrow \text{Algorithm}_1(\mathcal{N}, n)$
end for
15: $\mathcal{N} \leftarrow \text{doubleAuction}(\mathcal{N})$
end while
end for

Here in Algorithm 2, we assume prosumers are defined by nodes, and prosumers are capable of performing double auction and trading. Nodes generate two types of requests which are Buying and Selling requests. These requests are delegating to randomly selected neighbouring nodes for double auction as defined in the Algorithm 2. At a discrete-time t , a node can have no requests, a single request or a set of requests. For example, a node defines a home that is connected to a wind farm, the utility grid and provides an electric vehicle charging facility.

Input data is a set of \mathcal{N} nodes. Each node $n \in \mathcal{N}$ has a set of \mathcal{R} requests. These requests can be either Buying requests or Selling requests which will be processed at double auctions. The output is records of successfully traded transactions. Algorithm 2 can be explained as follows:

- In a given discrete-time t , nodes which have requests are in a set of \mathcal{N} .
- Select a node $n \in \mathcal{N}$ and find the neighbourhood. If there are no neighbours exist, then go to the next node. If there are neighbours, then go to the next step.
- Find all the requests of node n and for each request $r \in \mathcal{R}$, calculate the average position of the neighbourhood, select a neighbour node, delegate r to the selected neighbour node, move n towards the calculated average position of the neighbourhood, and find the new neighbourhood for n after its move. Repeat this process for all the requests in \mathcal{N} .
- Avoid collision using Algorithm 1 and repeat this process for all the nodes $n \in \mathcal{N}$.
- Apply double auction for all the nodes. Successful trading transactions are executed and recorded in a blockchain. Nodes having unsuccessful trading transactions are returned. Repeat above steps for number of auctions until find the convergence before going to next discrete-time t .
- Repeat the process for next discrete-time t .

This algorithm generates converged decentralised nodes. This convergence is important because that guarantees the requests are delegated only within neighbour nodes and that enables the minimum energy loss at energy transmission. The energy losses are mainly due to transformers and cables [9]. In a general P2P scheme, households interact directly with other households [21]. In this work, we assume prosumers interact directly with other prosumers, the energy loss is propositional to the distance between prosumers and energy requests are generated by prosumers considering the energy losses within neighbourhoods. Our future work will address prosumers in proper infrastructure networks considering network constraints[21] and calculate energy losses between prosumers to decide potential neighbourhoods.

In the above algorithm, requests delegation among randomly selected neighbours increases the anonymity of trading nodes for requests. This is important to improve the security of trading nodes from potential planned attacks. If there are \mathcal{M} nodes in the neighbourhood, the probability of finding the trading node for a particular request is $\frac{1}{\mathcal{M}}$. That is if \mathcal{M} increases then probability goes down and difficulty of finding the trading node goes high. Moreover, \mathcal{M} is changing dynamically for each request delegation and that further increases the anonymity of trading nodes.

4. Double auction and trading

This section discusses double auction for energy trading at each node which has both Buying and Selling requests. In this research, we used the double auction defined in [13] with a trade-reduction mechanism for P2P energy trading.

$\delta^a(N_i) < 0$ indicates that N_i needs δ^a amount of energy at the next time instance. Similarly $\delta^a(N_i) > 0$ indicates that N_i has a surplus of δ^a amount of energy to sell at the next time instance. Both Buying and Selling requests are generated accordingly.

$N = (N_1, \dots, N_k)$	a set of k nodes where $k \in \mathbb{Z}$ and $k > 0$
$G = (N, E)$	An undirected graph with a set of nodes and a set of edges. G represents the energy distribution network.
$t = (t_1, t_2, \dots)$	Discrete-time instances
$\delta^a(N_i)$	Energy requirement for the node N_i at time t_a

In our double auction market, each node N_i will have a set of Buying requests from buyers (B_1, B_2, \dots, B_p) and a set of Selling requests from sellers (S_1, S_2, \dots, S_q) at a given time t where p and q are positive integers which are greater than or equal to 0. Each node solves the double auc-

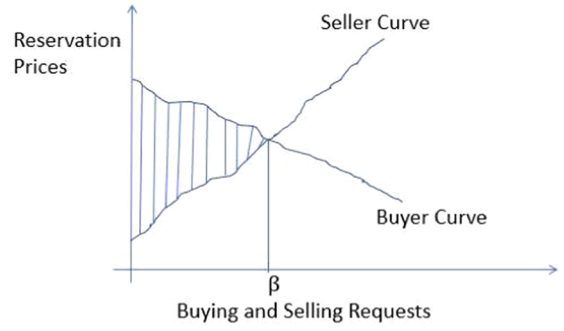


Fig. 2. Process of double auction: Curves of Buyer and Seller intersect at β .

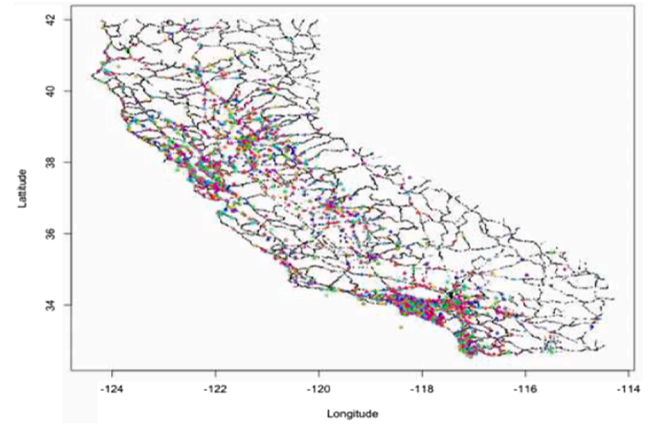


Fig. 3. Road network and houses with PVs in the US state California.

tion as follows:

- Sort the Selling requests in the increasing order as $SR_1 \leq SR_2 \leq \dots \leq SR_q$. The list of sorted Selling requests contributes to the seller curve.
- Sort the Buying requests in the decreasing order as $BR_1 \geq BR_2 \geq \dots \geq BR_p$. The list of sorted Buying requests contributes to the buyer curve.
- Let curves of Buying requests and Selling requests to cross at index β so that $BR_k \geq SR_k$ and $k \leq \beta$. The break-even index is k . Calculate p so that, $p = (SR_{k+1} + BR_{k+1})/2$.
- If $BR_k \geq p \geq SR_k$, then all the k requests of both types trade for price p .
- Otherwise, the Selling requests $SR_1 \leq SR_2 \leq \dots \leq SR_{k-1}$ trade for SR_k and the Buying requests $BR_1 \leq BR_2 \leq \dots \leq BR_{k-1}$ trade for BR_k as in the trade reduction mechanism.

The buyers and sellers curves are illustrated in Fig. 2. In our trading platform, double auction is processing as defined in Algorithm 2. Nodes to perform double auctions should have both buying and selling requests.

5. Experimental evaluation

5.1. Data processing

We used road networks data from the California state from [22] and processed locations, demand profiles (energy needs for houses) and PV profiles (energy generation information) of houses fitted with PVs using data sources from [23,24,5]. We created clusters of houses in close proximity. In each cluster, the maximum difference in house locations (difference in latitude or longitude) is 1 which corresponds to 11 km. We created 5000 clusters of houses where each cluster has at most 50 houses. Fig. 3 shows the locations of houses with fitted PVs and houses in

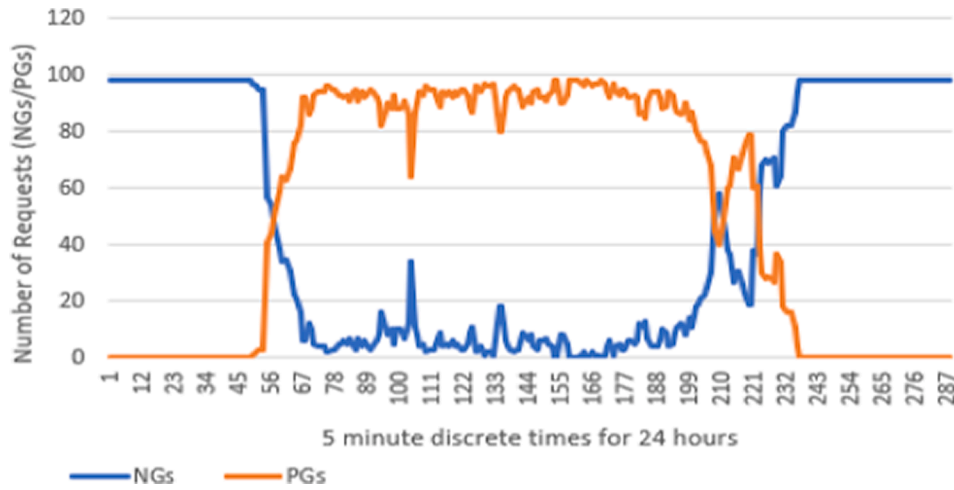


Fig. 4. Compare buying and selling requests for 24 h.

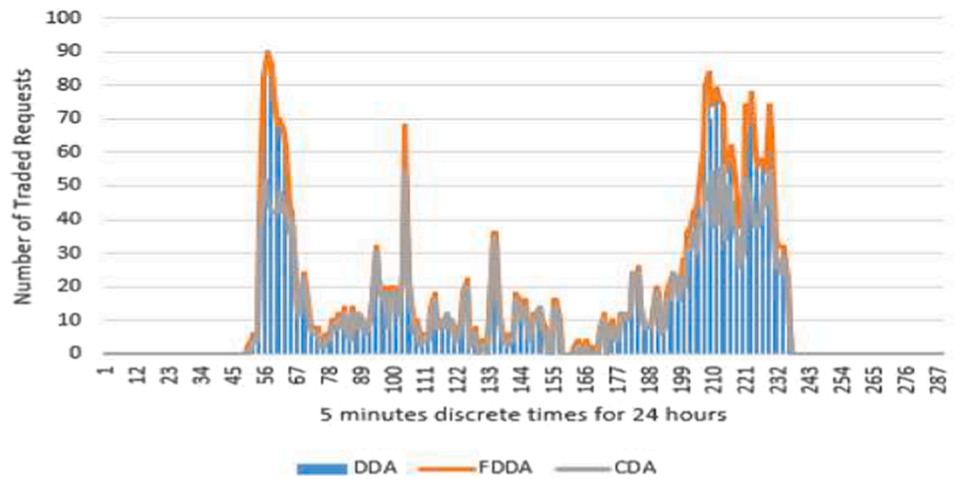


Fig. 5. Trading Comparisons: Flocking-based decentralised double auction (FDDA), distributed double (DDA) and centralised double auction(CDA) for 24 h.

each cluster is coloured with the same colour. This presents a high-level view regarding the density of the number of PVs and their locations. From this PV locations, we have selected 100 houses in the range of latitudes between 33 and 34, and longitudes between -119 and -118 . Then we created demand profiles and PV production profiles for each house every 5 min for 24 h. Using this information, we created an excess energy profile for each house in every 5 min for 24 h. We created two types of energy requests (Buying request, Selling request) for each house based on the excess energy profile.

5.2. Experimental evaluation

We created 100 nodes for 100 houses having buying/selling requests for each house based on the excess energy profiles in every 5 min for 24 h. Our flocking algorithm considers houses within a radius of 2 km as a neighbourhood. That is, the difference between the two latitudes or two longitudes is 0.02 for 2 km. We compare the number of buying requests (NGs) and selling requests (PGs) throughout the day for 24 h as in Fig. 4. We implemented centralised double auction [13], distributed double auction [5] and the proposed flocking-based decentralised double auction using java, tested with the above data set, and compared their results as in Fig. 5. Here, one successful trading has one buying request and one selling request that is two traded requests. The results showed that both the FDDA and DDA in [5] have a higher success rate than CDA in [13]. However, the proposed FDDA showed the highest success rate

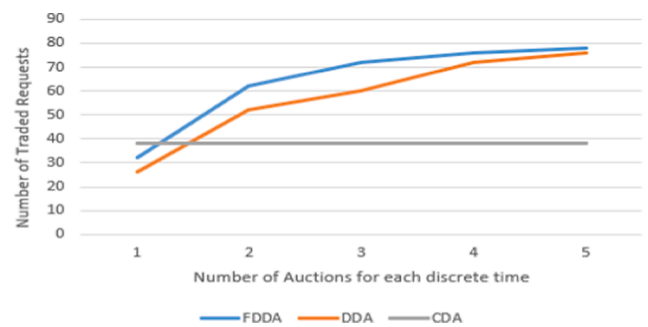


Fig. 6. Trading Comparisons when $t = 210$: FDDA, DDA and CDA.

throughout the day as shown in Fig. 5. When comparing Fig. 5 and Fig. 4, we can observe that a higher number of requests show higher success rates in all three algorithms. Moreover, we compared success rates in various auctions at various discrete-times. Here we present a comparison of successful trading percentages when $t = 210$ (t is an integer in the range 0 to 287 for 24 h) in Fig. 6. As in Fig. 6, at the first auction, we have observed the number of successful tradings given by the CDA is higher than that of FDDA and DDA. However, both FDDA and DDA enable several distributed auctions at a discrete-time and contribute to overall higher success rates than CDA. In a centralised

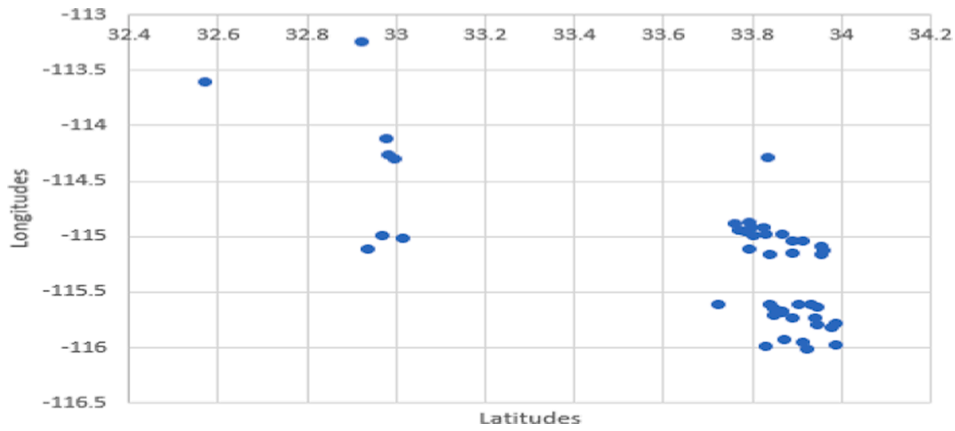


Fig. 7. Behaviour of Nodes: flocking-based decentralised double auction.

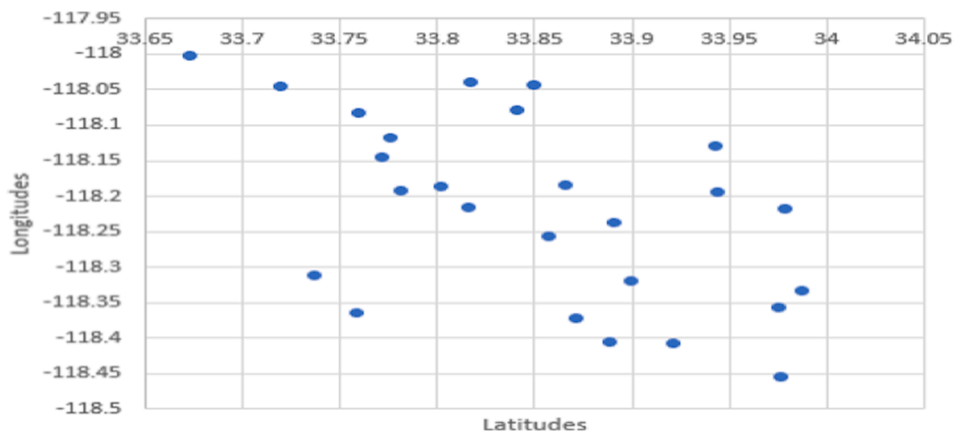


Fig. 8. Behaviour of Nodes: distributed double auction.

auction, bids from all the peers are used to determine the price for energy trade, but in distributed auctions, each local auction uses bids from a subset of peers. Hence the price for energy trade can differ for these local auctions. Moreover, if a prosumer fails to find a winner for an energy requirement, it will forward that energy requirement to a neighbouring prosumer. The energy requirements are processed in several distributed auctions with various trading prices at a discrete-time. As in Fig. 2 and experimental results in Fig. 6, the CDA shows

around 38% of success. However, at a discrete-time, both DDA and FDDA enable distributed auctions on prosumers, various trading prices, and several iterations showing a higher success rate than CDA. We have observed around 80% of energy requirements were traded in DDA and FDDA using 5 distributed auctions at a discrete-time. The results showed that the FDDA converges to 80% successful trading quicker than DDA in [5] having a higher success rate.

We have observed that the flocking-based decentralised double

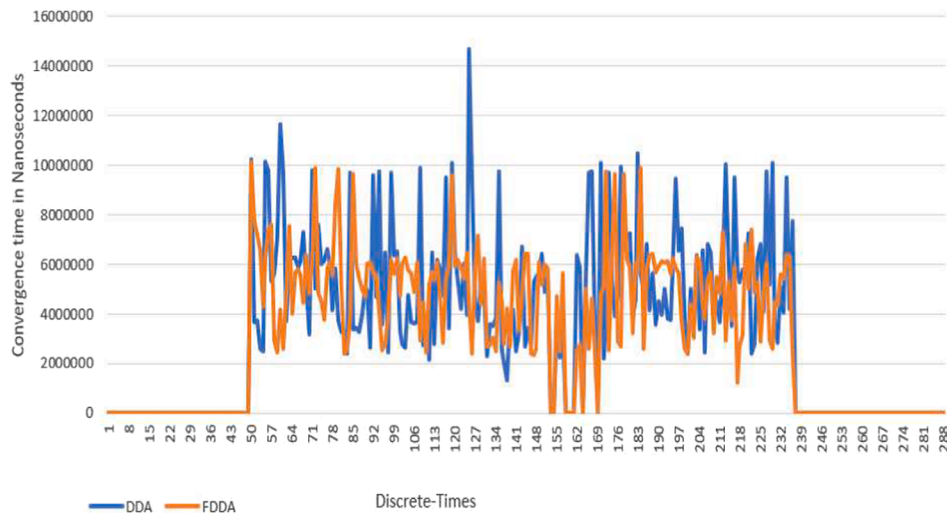


Fig. 9. Compare convergence times of DDA and FDDA.

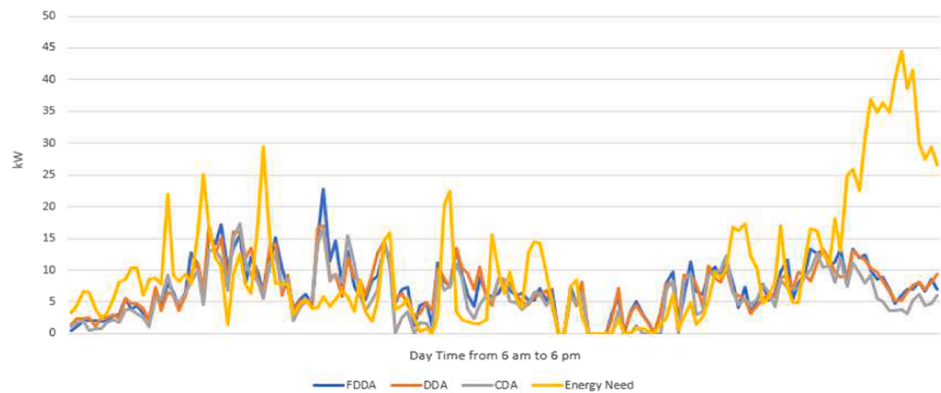


Fig. 10. The amount of energy traded and energy requirement.

auction guarantees trading within neighbourhoods because of the flocking behaviour of neighbour nodes. In [5], when a request does not achieve successful trading, it is forwarded to a neighbour. However, it is not guaranteed that the 3rd neighbour is a neighbour of the request generator. That is, distributed double auction in [5] does not guarantee to trade within the neighbourhoods. Here we present the behaviour of nodes in the 2nd auction at a discrete-time, $t = 210$. The behaviour of nodes using the proposed flocking-based algorithm is illustrated in Fig. 7 and distributed double auction in [5] is illustrated in Fig. 8. The flocking-based algorithm creates decentralised flocks considering energy demands and distance between PV locations while nodes in distributed double auction in [5] are not flocking around the neighbourhoods.

As defined in lines 7 to 13 in Algorithm 2, prosumers are moving towards the centre of the neighbourhood while delegating requests until all the requests are delegated. This process will be repeated for all the prosumers which have energy requirements. The double auction will be applied at prosumers after delegating requests. The unsuccessful requests will be processed again following the above procedures as defined in lines 2 to 17 of Algorithm 2. Our experimental results showed that both FDDA and DDA converged in 5 distributed auctions hence we repeated the above processes for 5 times. At convergence, most of the trading possible requests have been traded as illustrated in Fig. 5. Moreover, we have compared the convergence times of DDA and FDDA throughout the day and illustrated in Fig. 9. The results show that FDDA has a better performance than DDA at most of the discrete-times.

DERs can trade the excess energy. We have observed that the energy trading is successful mostly during the day time when the DERs generate energy. We have analysed the amount of energy traded using FDDA, DDA and CDA within the day time from 6 am to 6 pm using a summertime dataset. We have observed that FDDA has the highest amount of energy traded and that is 70.60461% of the total energy requirement. DDA and CDA can satisfy 69.64921% and 58.58982% of the total energy requirement. Fig. 10 illustrates the amount of energy traded using FDDA, DDA and CDA compared to the amount of energy needed from 6 am to 6 pm.

6. Discussion

Renewable energy sources are highly unreliable hence unreliable energy surplus and demands should be addressed frequently within neighbourhoods. In this research, we used the flocking behaviour of birds to define the dynamic behaviour of prosumers for P2P trading. We defined, implemented and tested a flocking-based decentralised double auction algorithm(FDDA) to perform P2P trading within neighbourhoods. We compared our algorithm with the centralised double auction (CDA) and the distributed double auction algorithm(DDA) defined in the literature. We used a PV data set from California state and tested algorithms for successful tradings, trading percentages and trading within

neighbourhoods at every 5 min for 24 h. We have observed that anonymous trading nodes are delegating requests to neighbours and converging to average positions of neighbourhoods at discrete-times. This guarantee that both Buying and Selling requests are delegated and traded within the neighbourhoods. The random request delegation preserves the anonymity of trading nodes. Further, the test results showed that FDDA has the highest success rate, faster convergence than DDA, and around 80% successful trading is achieved in five distributed auctions. In this work, we assumed the distance between two nodes was proportional to energy loss. Our future work will enrich a flocking-based distributed double auction to consider energy transmission loss based on the energy infrastructure network.

CRediT authorship contribution statement

Kosala Yapa Bandara: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Validation, Visualization, Writing - original draft, Software. **Subhasis Thakur:** Funding acquisition, Project administration, Resources, Writing - review & editing. **John Breslin:** Funding acquisition, Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This publication has emanated from research supported in part by a research grant from the Department of Enterprise, Trade and Employment's Disruptive Technologies Innovation Fund (DTIF), managed by Enterprise Ireland on behalf of the Government of Ireland under Grant Number DT20180040C, and by a research grant from SFI under Grant Number SFI 12/RC/2289_P2 (Insight), with the latter two grants co-funded by the European Regional Development Fund.

References

- [1] Benjamin K. Integrating high levels of variable renewable energy into electric power systems. *J Mod Power Syst Clean Energy* 2017;831–7.
- [2] Thomas M, Niall F, Sarah JD, Malcolm DM. Using peer-to-peer energy-trading platforms to incentivize prosumers to form federated power plants. *Nature Energy* 2018;94–101.
- [3] Dimeas A, Drenkard S, Hatzigiorgiou N, Karnouskos S, Kok K, Ringelstein J, Weidlich A. Smart houses in the smart grid: Developing an interactive network. *IEEE Electrification Mag* 2014;2(1):81–93.
- [4] Chenghua Z, Jianzhong W, Chao L, Meng C. Review of existing peer-to-peer energy trading projects. *Energy Procedia* 2017;105:2563–8.
- [5] Thakur S, Hayes BP, Breslin JG. Distributed double auction for peer to peer energy trade using blockchain. In: 5th International symposium on environment-friendly energies and applications; 2018. p. 1–8.

- [6] Thakur S, Breslin JG. Peer to peer energy trade among microgrids using blockchain based distributed coalition formation method. *Technol Econ Smart Grids Sustain Energy* 2018.
- [7] Wörner A, Meeuw A, Ableitner L, Wortmann F, Schopfer S, Tiefenbeck V. Trading solar energy within the neighborhood: field implementation of a blockchain-based electricity market. *Energy Informatics* 2019;2(1).
- [8] Leong CH, Gu C, Li F. Auction mechanism for p2p local energy trading considering physical constraints. *Energy Procedia* 2019;158:6613–8. <https://doi.org/10.1016/j.egypro.2019.01.045>.
- [9] IEC, Efficient electrical energy transmission and distribution, <https://www.iec.ch/about/brochures/pdf/technology/transmission.pdf>, Accessed: 2020-04-22.
- [10] Reynolds CW. Flocks, herds and schools: A distributed behavioral model, in: *14th Annual conference on computer graphics and interactive techniques*. ACM; 1987. p. 25–34.
- [11] Chazelle B. The convergence of bird flocking. *J ACM* 2014;61(4). <https://doi.org/10.1145/2629613>.
- [12] Olfati-Saber R. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Trans Autom Control* 2006;51(3):401–20. <https://doi.org/10.1109/TAC.2005.864190>.
- [13] McAfee R. A dominant strategy double auction. *J Econ Theory* 1992;56(2). [https://doi.org/10.1016/0022-0531\(92\)90091U](https://doi.org/10.1016/0022-0531(92)90091U).
- [14] Shrestha A, Bishwokarma R, Chapagain A, Banjara S, Aryal S, Mali B, Thapa R, Bista D, Hayes BP, Papadakis A, Korba P. Peer-to-peer energy trading in micro/mini-grids for local energy communities: A review and case study of nepal. *IEEE Access* 2019;7.
- [15] Zhang C, Wu J, Zhou Y, Cheng M, Long C. Peer-to-peer energy trading in a microgrid. *Appl Energy* 2018;220:1–12.
- [16] Steinmetz R, Wehrle K. Peer-to-peer systems and applications. Berlin, Heidelberg: Springer; 2005. p. 9–16. https://doi.org/10.1007/11530657_2. Ch. 2. What Is This “Peer-to-Peer” About?
- [17] Balakrishnan H, Kaashoek MF, Karger D, Morris R, Stoica I. Looking up data in p2p systems. *Commun ACM* 2003;46(2):43–8. <https://doi.org/10.1145/606272.606299>.
- [18] Zhang C, Wu J, Long C, Cheng M. Review of existing peer-to-peer energy trading projects. *Energy Procedia* 2017;105:2563–8. <https://doi.org/10.1016/j.egypro.2017.03.737>.
- [19] Bajec IL, Zimic N, Mraz M. The computational beauty of flocking: boids revisited. *Math Comput Modell Dyn Syst* 2007;13(4). <https://doi.org/10.1080/13873950600883485>.
- [20] Jadbabaie A, Lin J, Morse AS. Coordination of groups of mobile autonomous agents using nearest neighbor rules. In: *Proceedings of the 41st IEEE conference on decision and control*, vol. 3; 2002. p. 2953–8.
- [21] Guerrero J, Chapman AC, Verbic G. Decentralized p2p energy trading under network constraints in a low-voltage network. *IEEE Trans Smart Grid* 2019;10: 5163–73.
- [22] Dataset, Real datasets for spatial databases: Road networks and points of interest, <https://www.cs.utah.edu/lifeifei/SpatialDataset.htm>, Accessed: 2020-04-22.
- [23] NREL. Solar resource data, tools, and maps, <https://www.nrel.gov/gis/solar.html>, Accessed: 2020-04-22.
- [24] GoSolarCalifornia. California solar initiative data, https://www.californiasolarstatistics.ca.gov/data_downloads/, Accessed: 2020-04-22.