Efficient methods for approximating the Shapley value for asset sharing in energy communities

Sho Cremers [a,b,*], Valentin Robu [a,b,e], Peter Zhang [a,b], Merlinda Andoni [c], Sonam Norbu [c,d], David Flynn [c]

[a] *Intelligent and Autonomous Systems Group, Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands*
[b] *Algorithmics Group, Delft University of Technology (TU Delft), Delft, The Netherlands*
[c] *Systems Power & Energy Research Group, University of Glasgow, Glasgow, United Kingdom*
[d] *Smart System Research Group, Heriot-Watt University, Edinburgh, United Kingdom*
[e] *Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ, USA*

ARTICLE INFO

ABSTRACT

With the emergence of energy communities, where a number of prosumers invest in shared generation and storage, the issue of fair allocation of benefits is increasingly important. The Shapley value has attracted increasing interest for redistribution in energy settings — however, computing it exactly is intractable beyond a few dozen prosumers. In this paper, we first conduct a systematic review of the literature on the use of Shapley value in energy-related applications, as well as efforts to compute or approximate it. Next, we formalise the main methods for approximating the Shapley value in community energy settings, and propose a new one, which we call the *stratified expected value* approximation. To compare the performance of these methods, we design a novel method for *exact* Shapley value computation, which can be applied to communities of up to several hundred agents by clustering the prosumers into a smaller number of demand profiles. We perform a large-scale experimental comparison of the proposed methods, for communities of up to 200 prosumers, using large-scale, publicly available data from two large-scale energy trials in the UK (UKERC Energy Data Centre, 2017, UK Power Networks Innovation, 2021). Our analysis shows that, as the number of agents in the community increases, the relative difference to the exact Shapley value converges to under 1% for all the approximation methods considered. In particular, for most experimental scenarios, we show that there is no statistical difference between the newly proposed stratified expected value method and the existing state-of-the-art method that uses adaptive sampling (O'Brien et al., 2015), although the cost of computation for large communities is an order of magnitude lower.

1. Introduction

Recent years have seen a shift towards decentralised energy systems, in which communities of prosumers (consumers with their own local renewable generation capacity and storage) satisfy more of their own energy needs from renewable energy generated from local sources. A number of regions, such as the European Union [1] and the United Kingdom [2] are providing supportive regulations to encourage communities of consumers to shift away from the dependence on centralised energy generation, and towards more decentralised and local energy generation and storage systems.

One significant recent trend are transactive energy models that aim to achieve better coordination between production and consumption in local energy systems by use of market-based mechanisms that allow energy exchanges between energy end users and prosumers. In broad

lines, there are two main models of organisation for local transactive energy systems [3]. One is peer-to-peer (P2P) energy trading systems, in which prosumers invest in their own energy assets (such as solar PV panels, wind turbine, and or battery storage) and buy and sell energy with their neighbours directly, based on their individually-owned assets [4–7]. In this scenario, each prosumer is metered separately and pays the value of its *net metered* electricity demand (demand after using its generation and storage capacity). Another is the formation of *energy communities*, where prosumers group together and buy a shared generation resource (such as a community wind turbine) and or a shared community battery. Here, the whole community is *"behind the meter"*, i.e. pays for the net demand of the entire community over the billing period. The differences between the two models are illustrated in Fig. 1. Each prosumer in Fig. 1(a) owns energy sources

Nomenclature

Subscripts and sets

i	For agents (households)
k	For classes (clusters)
$\mathcal{N}$	Set of agents in the community
S	For subcoalitions formed by agents in community

Parameters

$SoC^{\max}$	Maximum battery SoC [%]
$SoC^{\min}$	Minimum battery SoC [%]
$p^{bat,\max}$	Maximum (dis)charging power of battery [kW]
$\tau^s(t)$	Export tariff at time t [pence/kWh]
$\tau^b(t)$	Import tariff at time t [pence/kWh]
η^c	Charging efficiency of battery
η^d	Discharging efficiency of battery
β and γ	Parameters for sigmoid function
$N_{cycles}^{DoD,\max}$	Maximum number of cycles allowed at specific DoD, provided from manufacturer specification

Variables

N	Number of agents
T	Number of time steps
t	For time steps
j	For strata
$d_i(t)$	Power demand of agent i at time t [kW]
$d_{\mathcal{N}}(t)$	Power demand of community $\mathcal{N}$ at time t [kW]
$g(t)$	Power generated by community RES (wind turbine) at time t [kW]
$p^{bat}(t)$	Power of community battery at time t [kW], negative when charging and positive when discharging
$p^{grid}(t)$	Power to/from utility grid at time t [kW], negative when selling and positive when buying
$SoC(t)$	Battery state of charge at time t [%]
$e^b(t)$	Imported energy at time t [kWh]
$e^s(t)$	Exported energy at time t [kWh]
$c_T^{grid}(\mathcal{N})$	Annual cost of community $\mathcal{N}$ importing energy from utility grid, with $T = 1$ year [£]
$c_T^{wind}(\mathcal{N})$	Annual cost of wind turbine of community $\mathcal{N}$, with $T = 1$ year [£]
$c_T^{bat}(\mathcal{N})$	Annual cost of battery of community $\mathcal{N}$, with $T = 1$ year [£]
DF	Depreciation factor of battery
$c_T(\mathcal{N})$	Total annual energy cost of community $\mathcal{N}$, with $T = 1$ year [£]
$c(\mathcal{N})$	Cost function, equivalent to $c_T(\mathcal{N})$
ϕ_i	Annual cost of agent i according to Shapley value [£]
MC_i	Annual cost of agent i according to last marginal contribution (unnormalised) [£]
$\overline{MC_i}$	Annual cost of agent i according to normalised last marginal contribution [£]

$d_{p_{-i}}(t)$	Average power demand of the community without agent i at time t [kW]
SEV_i	Annual cost of agent i according to stratified expected value (unnormalised) [£]
$\overline{SEV_i}$	Annual cost of agent i according to normalised stratified expected value [£]
M	Number of samples of marginal contributions per agent (for adaptive sampling)
RL_i	Annual cost of agent i according to adaptive sampling [£]
K	Number of classes of unique demands in the community
N_k	Number of agents that belongs to class k
$P()$	Multivariate hypergeometric distribution
$\hat{\phi}_k$	Cost redistributed to class k by certain redistribution method ($\overline{MC_k}$, $\overline{SEV_k}$, or RL_k) [£]
$RD_\phi(\hat{\phi}_k)$	Relative difference of a redistributed cost to the Shapley value for class k [%]
$RD_\phi(\hat{\phi})$	Average relative difference of a redistribution method to the Shapley value [%]
Δt	Duration of time period t [hour]
DoD	Depth of discharge of battery [%]
$DF^{regular}$	Depreciation factor by regular cycles
$DF^{irregular}$	Depreciation factor by irregular cycles
$n_{cycles}^{DoD,regular}$	Number of regular cycles at specific DoD
$SoC_l^{start/end}$	Starting/ending SoC of cycle l [%]
$\pi_{i,j}(m)$	Probability of sampling stratum j for agent i at sample m
$\epsilon(m)$	Sigmoid function
$\hat{\sigma}_{i,j}$	Estimated standard deviation expected marginal contribution of stratum j for agent i
$\hat{\mu}_{i,j}$	Estimated expected marginal contribution of agent i at stratum j in adaptive sampling
$\hat{\sigma}_{i,j}$	Estimated standard deviation of expected marginal contribution of agent i at stratum j in adaptive sampling
$h_{i,j}$	Count of agent i sampling from stratum j in adaptive sampling
$m2_{i,j}$	Sum of squared differences from mean of stratum j for agent i in adaptive sampling
mc	Sampled marginal contribution in adaptive sampling
Δ	Difference between the sampled value and the mean in adaptive sampling
$\tilde{d}_k(t)$	L2-normalised power demand of class k at time t

Abbreviations

DF	Depreciation Factor
DoD	Depth of Discharge
P2P	Peer-to-Peer
RES	Renewable Energy Source
SoC	State of Charge

and a battery, and individually interacts with the central power grid, in which the net demand is counted. It can be seen that the power flow between a prosumer and the utility grid is bidirectional, and any excess generation by the prosumer is sold to the grid. Furthermore, a P2P trading scheme makes buying and selling energy among peers possible, which is represented by dotted arrows in Fig. 1(a). On the contrary, the energy community in Fig. 1(b) presents a group of consumers sharing energy assets and interacting with the utility grid as a single entity. Net demand is computed for the whole community. These "behind the meter" models rely on a *community aggregator*, which controls the energy assets and distributes the generated/discharge power to the households in the community. The aggregator is also in charge of receiving energy from the utility grid whenever there is a deficit
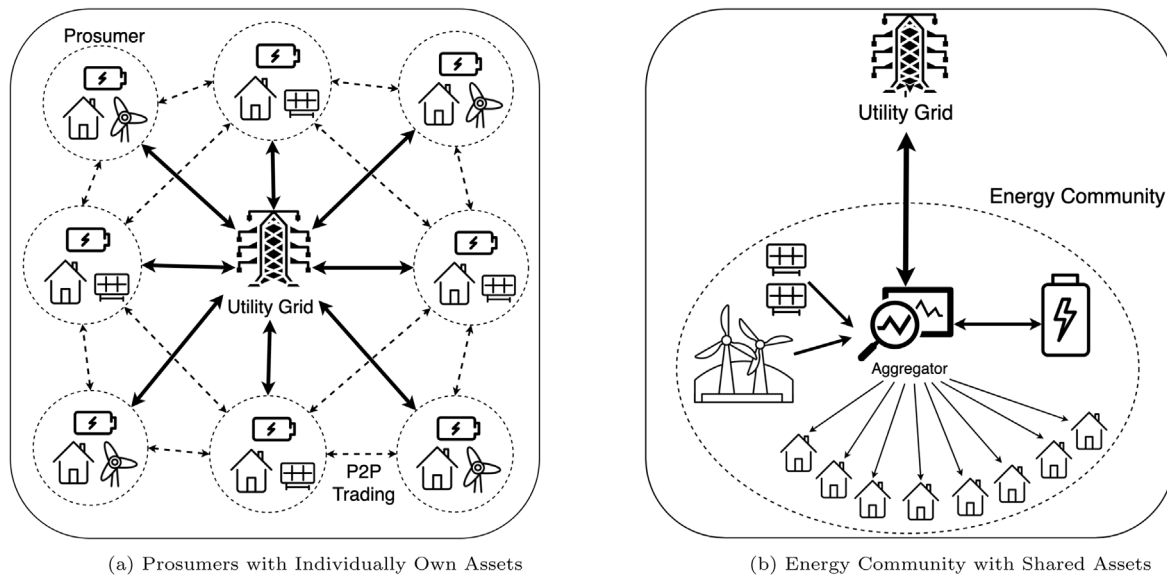
(a) Prosumers with Individually Own Assets
(b) Energy Community with Shared Assets

Fig. 1. Two different configurations of a community of prosumers, where (a) prosumers with their own energy assets are connected to the central grid individually, and (b) prosumers with jointly-owned assets interact with the grid as a whole through the aggregator.

and sending back energy to the grid when there is a surplus in the community.

This coalitional model around shared assets is increasingly popular, not just in academic research – for example, in Scotland, UK, Community Energy Scotland[1] identified 300+ energy communities that formed around a shared energy asset – typically a wind turbine, but similar examples exist all over the world. Such energy communities can consist of anywhere between several dozens to several hundred houses (e.g. a village or a city neighbourhood), often located on the LV network behind a local transformer. The prosumers share the outputs of jointly-owned energy assets, as well as the energy bill for the aggregate residual demand, i.e. the part of the demand not covered by the local generation and storage assets. Therefore, the community aggregator is not only responsible for the control and distribution of energy in the community but also for allocating any revenues from exporting energy and the bills of the residual demand. Clearly, one of the key challenges in this setting is the redistribution of such costs and benefits to the prosumers in a *fair* way.

Coalitional game theory has long studied such redistribution problems in a wide variety of systems [8]. A key concept is the Shapley value, first proposed by the Nobel prize-winning economist Lloyd Shapley [9]. The Shapley value has recently begun getting substantial attention in the energy applications — with a rapid increase in the number of papers using Shapley value in energy systems in recent years (see Section 2). However, a key challenge with the Shapley value is that computing it is exponential in the number of agents, making exact computation intractable beyond a small number of agents.

The prior papers dealt with this in several ways. Most consider experimental models with up to a maximum of ~10-20 agents, to keep computations tractable. Another approach is to use some simpler heuristics for cost redistribution (e.g. [10]), but it is not clear how close these are to the exact Shapley value.

Yet another approach is to use sampling. Sampling-based approaches do have merit, and in this paper, we implemented the most advanced sampling-based method we are aware of, that of O'Brien et al. [11], which uses reinforcement learning techniques to perform adaptive sampling to calculate the Shapley value. However, they also have disadvantages: for larger settings, a very large number of samples may be needed to get a reasonable approximation of the true Shapley value,

which increases the computation cost considerably. Also, in community energy applications, sampling-based methods have the disadvantage that they may not produce a consistent result if they need to be rerun for verification purposes. Community energy schemes rely on the distributed trust of the prosumers in the community, and hence on the ability to sometimes rerun the calculations of the coalition coordinator, if they wish. But, as the calculation at each run depends on which random samples are drawn, results will be slightly different, even on the same data. Hence, there is an important knowledge gap about approximating the Shapley value in larger settings, which our paper aims to address. Specifically, we study both sampling-based and deterministic methods for approximating Shapley, compare their performance w.r.t. the true, exact Shapley value, and derive their computation costs. As one of the contributions of this paper, we introduce a novel redistribution method that approximates the Shapley value well within polynomial time, and compare it to existing methods.

A key open challenge in this space remains determining the "ground truth", i.e. computing the exact, true Shapley value to compare other methods to, especially for larger realistically-sized communities (e.g. dozens to several hundred prosumers). Prior approaches, like O'Brien et al. [11] use a setting of only 20 agents as a "ground truth" to compute the exact Shapley, as they naturally find larger settings unfeasible to compute with unique agents. Yet, as we show in our experiments, an approximation method that does poorly for a small number of agents (e.g. 5–20) may actually do well for a realistically sized setting of 100–200 agents. Another important contribution of this paper is that we develop a method to compute the *exact* Shapley value for larger communities, up to 200 agents. Intuitively, the core idea behind the method (see Section 4.2) is to cluster the agents in a much smaller number of consumption profiles, and use the symmetry of the combinations of agents to greatly reduce the cost of exact Shapley calculation.

Finally, as part of our contributions, we implemented our method in realistic community case studies, both in terms of demand, generation and battery data used, and in terms of size (up to 200 households), granularity and duration (half-hourly data over a whole year). We used two different datasets, both containing household energy consumption data in the UK and the corresponding wind generation and battery data. One draws data from the Thames Valley vision trial [12] while the other draws data from the Low Carbon London project [13]. This provides a highly realistic case study to provide confidence in the robustness of our experimental comparison results.

---

[1] https://communityenergyscotland.org.uk/

The rest of the paper is organised as follows. First, a review of the literature is provided in Section 2. Section 3 presents the community energy model used. Section 4 introduces the Shapley value and its computation methods. Then, Section 5 presents the experimental comparison across a number of scenarios. Finally, Section 6 concludes the paper with a discussion.

2. Literature study

This Section presents our systematic study of the literature on Shapley value computation and energy systems. We note that the distribution of benefits and costs in smart energy systems is a broad one, and Shapley value is just one of the possible solution concepts. It is, however, the most widely used concept and broadly applicable to a variety of settings — with many of the alternatives only applicable in specific settings. Also, Shapley value has a very strong foundation in coalitional game theory, and has had a wide impact in many fields, ever since it was proposed by Nobel-prize winning economist Lloyd Shapley. However, a key problem with applying the Shapley value (especially in the case we study, i.e. energy communities settings with a sizeable number of prosumers) is that it is not computable exactly in settings beyond a few dozen agents, as the computational cost of *exact* Shapley computation is combinatorial. As highlighted by the introduction and our systematic review below, our work helps to close this important knowledge gap by providing and validating computationally-efficient tools to approximate Shapley, with validation in a highly realistic case study of a community energy setting.

The literature study section is divided into two subsections: first, in Section 2.1, we provided a systematic overview of previous works that use Shapley value concept in energy applications, while in Section 2.2 we discuss existing state-of-the-art techniques for Shapley approximation. Our review is enhanced by providing a systematic table that captures and summarises the prior literature related to the application of Shapley values in energy settings, along four key dimensions: the particular sub-area of energy where a referenced paper applied the Shapley value, the computational techniques employed, the type of approach to computing Shapley (whether exact, or approximation based under some assumptions, or both), and finally the number of agents (be it prosumers, households, participants, etc.) that the experimental part of the paper considers. We argue that providing such a table of prior works is important to highlight the current state-of-the-art in the field and make the contribution of this work clearer to the reader.

2.1. Use of shapley values in energy applications

Energy communities are an increasingly important topic of research in energy systems, and a notable number of recent papers consider using the Shapley value as an underlying redistribution method. Chiş and Koivunen [14] propose a coalitional cost-game optimisation of a portfolio of energy assets using Shapley value as the underlying redistribution method, modelling a realistic case study of 9 households. Safdarian et al. [15] use the Shapley value for coalition-based value sharing in energy communities, modelling an energy community in southern Finland with up to 24 apartments. Vespermann et al. [16] study the market design of a local energy community with shared storage and consider a number of solution concepts such as the nucleolus and Shapley values. Their numerical simulations study communities ranging in size from 4 up to 16 prosumers. Robu et al. [17] consider a cooperative coalitional game for energy group buying. While they discuss Shapley value as a solution concept, their focus is on other coalition properties.

There are also works that study variations of energy communities. Vinyals [18] explores a model in which the community consists of prosumers with assets and pure consumers, and the excess energy generated is shared among the community members. Although the work focuses on the energy distribution model that minimises the total

cost of the community while meeting regulatory restrictions, it also presents an individually rational cost redistribution scheme. Long et al. [19] propose a method for energy trading of excess generation by the prosumers and individual cost calculation based on coalitional game theory and the Shapley value, and tests on a community that consists of 5 prosumers with solar PV generation and energy storage, and 5 consumers with no assets. Similarly, Hupez et al. [20] compare the use of Nash versus Shapley value concepts in an LV energy community model in which the excess energy of the prosumers is shared among other consumers with a case study of 3 prosumer nodes. Singh et al. [21] present the use of Shapley value for energy trading among microgrids, using a case study of 3 microgrids. Zhang et al. [22] consider the use of Shapley value to divide gains in alliances among retailers in the Chinese energy settlement market, considering alliances up to a size of 9 agents.

In addition to the above, applications of Shapley value can be found in many domains within energy systems. The most relevant previous works identified (after a systematic search) on the use of Shapley value in energy application are summarised on Table 1. It reviews 40 selected papers that the authors found to be relevant both to the energy domain and Shapley value computation. It classifies them based on four criteria. The first is regarding the energy application domain in which the Shapley value is applied. The second is techniques used in the work, which could be for computing the Shapley value, but also for solving the underlying problem. The third criterion is how the Shapley value is computed. Most papers compute the exact Shapley value, but there are also many works that make use of approximation methods. Finally, the maximum number of agents used for computing Shapley value in their experimental analyses is given. Some works that apply approximation methods also compute the exact Shapley value as a benchmark. In such cases, the corresponding maximum number of agents for both methods is listed.

From this analysis, we observe that community energy/P2P trading was the most popular application of the Shapley value, but they were also common in other domains, such as the allocation of distribution loss [30,32,33] and congestion cost [39–41], as well as profit distribution in virtual power plants (VPPs) [42–44]. There were some less obvious applications, namely, cost allocation of net loss variability [50] and coordinated operation of existing facilities and the emerging power-to-gas technology [51,52]. The range of techniques used by the authors was very broad, ranging from machine learning techniques (e.g., reinforcement learning, K-means clustering), optimisation techniques (e.g., mixed-integer linear programming, particle swarming), to comparison to other allocation methods (e.g., nucleolus, Nash equilibrium).

We found it especially important to provide a classification of Shapley value computation methods and the maximum number of agents considered. Crucially, for exact Shapley computation methods, the number of agents is always kept low to keep the computation tractable, usually to less than 10 agents. There are few papers that take into account more agents, such as Alam et al. [28] and previous work by some of the co-authors of this paper (Norbu et al. [10,24]), but these studies do not attempt to compute the Shapley value exactly for a large number of agents and instead use approximation methods like the simple marginal contribution method also used in this study. The complexity of computing Shapley often restricts studies to experimental simulations with small numbers of agents — yet, in practice, larger settings appear frequently. Realistically sized energy communities have more members, e.g., there are usually 50–200 consumers behind a substation/LV transformer in Europe [54], or potentially even more sharing an asset such as a large community wind turbine. Hence Shapley approximation methods are needed — yet, the understanding of what is a good approximation for large settings is still lacking. Our work aims to fill this knowledge gap.

Table 1
Summary of published studies on the use of Shapley value in energy applications.

Reference	Energy application area	Techniques used	Type of Shapley computation	Max No. of agents
Norbu et al. [10,23,24]	Energy community	Systematic comparison, Data-driven approach	Marginal contribution	200 prosumers
Chiş & Koivunen [14]	Energy community	Cost minimisation under constraints	Exact	9 households
Safdarian et al. [15]	Energy community	K-means clustering	Random sampling	24 apartments
Han et al. [25]	Energy community	Sample allocation with estimated variance	Exact Stratified sampling	16 prosumers 50 prosumers
Kulmala et al. [26] & Baranauskas et al. [27]	Energy community	Comparison of cost redistribution methods	Exact & approximation	6 households
Long et al. [19]	Energy community/P2P trading	Fairness analysis	Exact	10 households
Alam et al. [28]	Energy community/P2P trading	Utility maximisation under constraints	Exact Random sampling	10 prosumers 100 prosumers
Hupez et al. [20]	Energy community/P2P trading	Shapley and Nash equilibrium analysis	Exact	3 nodes
Vespermann et al. [16]	Storage in energy communities	Shapley and nucleolus analysis	Exact	16 prosumers
Singh et al. [21]	Trading between microgrids	Decentralised coordinated scheduling	Exact	3 microgrids
Jia et al. [29]	Trading between microgrids	Cost minimisation under constraints	Exact	3 microgrids
Zhang et al. [22]	Coordination of electr. retailers	Transaction cost, Resource dependence theory	Bilateral Shapley value	9 retailers
Sharma & Abhyankar [30,31]	Loss allocation in distribution	Exploitation of network topology	Exact Sequential Shapley	24 participants 68 participants
Amaris et al. [32]	Loss allocation in distribution	Circuit laws, Systematic comparison	Aumann–Shapley value	35 units
Pourahmadi & Dehghanian [33]	Loss allocation in distribution	Benchmarking, Systematic comparison	Exact	15 units
Azad-Farsani et al. [34]	Allocation of loss reduction in distribution	Point estimation, stochastic iterative algorithm	Exact	15 units
Yu et al. [35]	Allocation of loss reduction in distribution	Approximation of Shapley value and nucleolus	Aumann–Shapley value	12 units
Vicente-Pastor et al. [36]	Network coordination	Mechanism design	Exact	3 stakeholders
Azuatalam et al. [37]	Network cost allocation	SD estimation	Stratified sampling	25 customers
O'Brien et al. [11]	Demand-side response	Reinforcement Learning	Adaptive sampling	20 participants
Maleki et al. [38]	Coordination of cooling loads	Bounded rational reasoning, Dynamic programming	Bounded rational Shapley value	15 apartments
Singh et al. [39]	Congestion cost allocation	Comparison to Shapley	Exact	3 nodes
Xiao and Li [40]	Congestion cost allocation	Pool-based model	Exact	6 lines
Voswinkel et al. [41]	Congestion cost allocation	Constraint optimisation	Exact & approximation	11 congestions
Cheng et al. [42]	Profit distribution in VPP	Coalition and core analysis	Marginal contribution	3 participants
Wang et al. [43]	Profit distribution in VPP	Real-world feasibility study	Exact	2 participants
Dabbagh & Sheikh-El-Eslami [44]	Profit distribution in VPP	Risk aversion degree	Exact	6 participants
Fang et al. [45]	Profit distribution in CHP-VPP	Particle swarm	Exact (modified)	4 stakeholders
Chattopadhyay [46]	Profit distribution in emission trading	Linear programming	Exact	3 participants
Liao et al. [47]	Allocation of emission allowance	Systematic comparison	Exact	3 power plants
Zhou et al. [48]	Carbon obligation allocation	Systematic comparison	Exact Aumann–Shapley value	3 DSOs 20 DSOs
Zhang et al. [49]	Allocation of emission allowance	Entropy, gravity model	Exact	8 regions
Mays [50]	Net load variability	Consumption behaviour profiling	Exact	9 profiles
Zhang et al. [51]	Coordinated bidding of wind farms and P2G facilities	Shapley and nucleolus analysis	Exact	3 participants
Li et al. [52]	Coordinated operation of NGG and P2G facilities	Risk aversion degree, MILP	Exact	2 parties
Churkin et al. [53]	Transmission expansion planning	Shapley and nucleolus analysis	Exact	5 countries

2.2. Approximation methods for Shapley values

Due to the large runtime of the Shapley value computation, it has received strong interest in efficient approximation methods since its introduction. Currently, many approximation methods compute the expected marginal contribution of an agent to the *sampled* coalitions, initially suggested by Mann and Shapley [55]. Furthermore, the seminal work of Castro et al. [56,57] proposes a polynomial calculation method which highlights the concept of *stratified sampling*, which has been refined in other works [25,28], and is a key concept in the method we develop as well. Many recent works also provide theoretical error bound of sampling-based approximation methods [58–62].

A major obstacle to approximating the Shapley value is that there does not exist a general deterministic approximation method that is a fully polynomial-time approximation scheme (FPTAS), and a fully polynomial-time randomised approximation scheme (FPRAS) is the best one can achieve when approximating the Shapley value [58]. Yet, deterministic methods have desirable characteristics for cost redistributions of consumers. Such methods produce the same results after every run given the same inputs, allowing consumers to verify the calculated cost themselves, in contrast to random sampling methods where the redistributed cost can differ depending on the samples drawn. Furthermore, deterministic methods would also guarantee the same cost redistributed to consumers with the exact same demand profile. Such properties can provide consumers with additional trust in the model. Bhagat et al. [63] provides a deterministic Shapley approximation method to their newly proposed budgeted games. Their method is theoretically proven to approximate the Shapley value with a constant additive error by replacing the value function with a relaxed function. However, theoretical analyses of deterministic methods are especially difficult in many real-world energy applications, in which the cost function results from a control procedure over the energy assets over a long time horizon rather than in closed form. Hence, many recent studies in the energy communities have performed empirical analyses to evaluate the performances of the approximation methods (e.g. [14–16]).

The publications closest to this work are O'Brien et al. [11] and Norbu et al. [10]. O'Brien et al. [11] propose an enhancement of the methods first outlined by Castro et al. that uses reinforcement learning to do the stratified sampling in an adaptive way. Their method is one of the methods used as a benchmark in this paper. However, a key limitation of [11] is that they still use a comparison benchmark of only 20 agents, while we develop a way to compute it exactly for much larger communities. Moreover, we wanted to develop and test some deterministic methods of Shapley value approximation that do not depend on the number of samples and can be reproduced to give the same result. Finally, the work of Norbu et al. [10] considers redistribution in realistic community energy settings, starting from marginal value principles, but they do not approximate Shapley value as such. However, with their support, we use the same demand/generation dataset of a community of 200 prosumers in the UK, as it provides a realistic experimental case study to test the methods we develop. Additionally, we also look at a different dataset with a larger number of households to further provide confidence within our methods. This paper is a considerably extended and revised version of preliminary work presented in a poster at the the ACM 2022 E-Energy conference [64].

3. Community energy model

Consider an energy community $\mathcal{N}$ consisting of a set of $|\mathcal{N}| = N$ prosumers, a shared battery and renewable energy source (RES). In this study, a lithium-ion battery and Enercon E-33 wind turbines [65] with a rated power of 330 kW were considered as the community's energy storage system and RES, respectively. Each prosumer in the community has a half-hourly power demand profile represented as $d_i(t)$ for the power demand of agent i at time step t. The final time step of the operation of the system is denoted as T. In this study, the data consists of half-hourly demands and generation during a 1 year period, and hence $T = 365 \times 48 = 17520$.

The demand of the community at time t, $d_{\mathcal{N}}(t)$, is simply the sum of the demands of the agents in the community at t, described as the following.

$$d_{\mathcal{N}}(t) = \sum_{i \in \mathcal{N}} d_i(t), \quad \forall t \in \{1, \dots, T\} \tag{1}$$

Furthermore, a community has a generation profile, $g(t)$, by the jointly owned local renewable energy generation, and the power of the battery, $p^{\text{bat}}(t)$. The battery is considered charging when $p^{\text{bat}}(t)$ is negative

and discharging when $p^{\text{bat}}(t)$ is positive. Finally, a community is required to buy power from the utility grid if the community assets do not provide enough power for the demand. If there is a surplus of power, on the other hand, a community can sell excess power to the grid. The power of the utility grid is denoted as $p^{\text{grid}}(t)$, where the value is positive when power is bought from the grid and negative when power is sold to the grid.

Given these variables, the following constraint needs to be satisfied at every time step.

$$d_{\mathcal{N}}(t) = p^{\text{grid}}(t) + p^{\text{bat}}(t) + g(t), \quad \forall t \in \{1, \dots, T\} \tag{2}$$

The constraint assures that the community power demand is met from the power sources. Additionally, when the generation is greater than the demand, all of the energy from the excess power is stored in the battery and or sold to the utility grid.

3.1. Battery control algorithm

The use of battery was regulated at each time point using the heuristic-based battery control algorithm from Norbu et al. [10]. The battery keep tracks of its state of charge (SoC), so that the battery capacity is not exceeded. The algorithm first looks at whether the community's demand, $d_{\mathcal{N}}(t)$, is smaller than the generation of the local RES, $g(t)$. If the generation is greater than the demand, the battery is charged as long as it has not reached the maximum battery capacity, SoC^{max}. If the battery has reached the maximum capacity or the surplus power is larger than the maximum (dis)charging power of the battery, $p^{\text{bat, max}}$, then the remaining energy is sold to the utility grid. The energy is sold with the price of $\tau^s(t)$ (pence/kWh), also known as the export tariff. If the community power demand is greater than generation, the battery is discharged if it has not reached the minimum battery capacity, SoC^{min}. If the battery has reached the minimum capacity or the power deficit is larger than $p^{\text{bat, max}}$, then energy is bought from the grid to meet demand. The import tariff, or the price of buying energy from the grid at t (in pence/kWh), is denoted as $\tau^b(t)$. More details about the battery control algorithm can be found in Appendix A.

Note that, in the heuristic-based battery control algorithm above, we considered *flat* import/export tariffs (in which the price remains the same throughout the time period of the operation), and moreover, importing or exporting energy to the grid is always worse price-wise than consuming/storing it locally, when possible. This is a realistic assumption in the current climate, when import prices are high, and so-called feed-in tariffs (i.e. tariffs paid to very small renewable generators) are being phased out. It is possible to have more advanced control heuristics in case of dynamic or time-of-use prices from the grid that include, e.g. a price prediction component. However, the Shapley computation methods proposed in this paper can also be combined with more complex control cases. This is because the methods we develop apply to the overall cost function, working to minimise the times of iterations needed to recompute it — but are independent of how the control is performed.

3.2. Community cost calculation

A cost function is a key attribute of a coalitional game. Here, energy cost calculation of the community (or any subset of prosumers) is explained. The community energy cost calculation can be seen as the cost function in this study, and it is required for redistribution methods described in Section 4.1.

The community energy cost is composed of three components. The first is the cost of energy bought from the grid, subtracted by the revenue of energy sold to the grid during the time period. The energy bought and sold at each time point, $e^b(t)$ and $e^s(t)$ respectively, are

determined by the battery control algorithm explained in Section 3.1. The cost $c_T^{\text{grid}}(\mathcal{N})$ is computed as the following.

$$c_T^{\text{grid}}(\mathcal{N}) = \sum_{t=1}^{T} e^b(t)\tau^b(t) - \sum_{t=1}^{T} e^s(t)\tau^s(t) \tag{3}$$

The second component of the cost for installing and operating the wind turbine, $c_T^{\text{wind}}(\mathcal{N})$. The annual cost is calculated as the following.

$$c_T^{\text{wind}}(\mathcal{N}) = \frac{\text{WT generation capacity} * \text{cost per kW}}{\text{Lifetime (in years)}} \tag{4}$$

The wind turbine generation capacity is calculated as the maximum receiving power from the wind turbine in one time step. The receiving power from the wind turbine was chosen to be $0.006 \times N$ times the power output of one wind turbine. The maximum capacity of the wind turbine increases linearly with the number of prosumers in the community, and hence the cost also increases linearly with the size. The cost of the wind turbine was set to 1072 £(GB pounds)/kW and lifetime to 20 years, which is realistic for current technologies in the UK market [10].

The last component of the cost is the battery. The cost of the battery, $c_T^{\text{bat}}(\mathcal{N})$ is computed as the following.

$$c_T^{\text{bat}}(\mathcal{N}) = \frac{\text{battery capacity} * \text{cost per kWh}}{\max\left(\text{Lifetime (in years)}, \frac{1}{\text{DF}}\right)} \tag{5}$$

In this study, the community battery capacity is set to be $5 \times N$ kWh. Similarly to the wind turbine, the battery capacity increases linearly with the community size, and therefore the community battery cost also increases linearly with the community size. The cost of battery per kWh was set to 150 £/kWh and the lifetime of the battery to 20 years. The variable DF is the depreciation factor of the battery determined by the battery degradation model from Norbu et al. [10]. Although the battery is given a lifetime, the lifetime can be shortened or additional maintenance costs may be required depending on the number of charge cycles and depth of discharge (DoD). Hence, using a battery degradation model can give a better assessment of the annual battery cost. The details of the battery degradation model are presented in Appendix B.

The total cost of the community, $c_T(\mathcal{N})$ is the sum of the three components, which is the following.

$$c_T(\mathcal{N}) = c_T^{\text{grid}}(\mathcal{N}) + c_T^{\text{wind}}(\mathcal{N}) + c_T^{\text{bat}}(\mathcal{N}) \tag{6}$$

The community cost can be computed for any subset of agents, and thus the cost contribution of an agent to a group can be determined by comparing the cost of the group with and without the agent. Specifically, every agent in the group contributes equally to the cost of the wind turbine and the battery (from Eq. (4) and (5)), but it does not mean the usage of the assets are equal among agents. For example, agents with demand profiles that are well-aligned to the energy generation of the wind turbine will make better use of the community generation assets, resulting in requiring less imported energy from the utility grid to match their demand. On the other hand, agents with demand profiles that are poorly aligned with the generation will put greater pressure on the community battery capacity and equivalently cause more energy to be imported. Therefore, the *marginal value* with which each prosumer causes the total cost to rise is a key factor to consider.

The community energy cost calculation can be seen as a cost function for a set of prosumers with demands. The notation of the community cost is simplified to $c(\mathcal{N})$ w.l.o.g., because time horizon $T = 1$ year is used to compute costs in the rest of the paper.

4. Shapley value computation

The redistribution of costs or benefits in a game using the Shapley values is considered to be fair in the literature [9,66]. The cost of prosumer i according to the Shapley value, ϕ_i, is computed as the following.

$$\phi_i = \sum_{S \subseteq \mathcal{N} \setminus \{i\}} \frac{|S|!(N - |S| - 1)!}{N!}(c(S \cup \{i\}) - c(S)) \tag{7}$$

The marginal contribution of prosumer i to the subcoalition of prosumers S, denoted as $c(S \cup \{i\}) - c(S)$, is how much the prosumer adds to the cost by joining the subcoalition. Then, the Shapley value of agent i can be seen as the mean marginal contribution of i for all possible subcoalitions in the community and all possible permutations of these subcoalitions.

An alternative way to write the Shapley equation that is particularly useful for our approach is through using the concept of *stratum*, given as the following.

$$\phi_i = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{\substack{S \subseteq \mathcal{N} \setminus \{i\}, \\ |S| = j}} \frac{j!(N - 1 - j)!}{(N - 1)!}(c(S \cup \{i\}) - c(S)) \tag{8}$$

It can be seen that the marginal contribution of agent i to a subcoalition S is computed as in Eq. (7). Then, the marginal contribution is multiplied by the relative frequency of S in the stratum. A stratum j is a set of all possible subcoalition S with $|S| = j$. From this, the expected marginal contribution of agent i to a stratum 0 (empty subcoalition) up to stratum $N - 1$ (subcoalition of the whole community except i) can be computed. Then, the Shapley value of agent i is equivalent to averaged expected marginal contributions over the strata.

Yet, computing Shapley exactly from these equations has a very large time complexity (exponential to the number of agents in the community, as the marginal contributions to every subcoalition of the community is needed), which makes it intractable very quickly as the community size increases.

4.1. Methods for determining approximate Shapley values

In this subsection, we present 3 key methods for computing the Shapley values, starting from the simplest one (last marginal contribution), to increasingly more complex ones such as stratified expected value and adaptive sampling. In Section 4.2 we will present a method for exact Shapley computation in the case of a restricted number of types, while in Section 4.3 we discuss the computational properties of these methods.

4.1.1. Last marginal contribution

While computing Shapley value directly requires exponential number of steps, it is possible to use the marginal contribution principle to design a much simpler scheme that considers the marginal contribution of each agent w.r.t. the other $N - 1$ [10,26]. Formally, let the cost of an agent i in the community $\mathcal{N}$ be simply the marginal contribution of agent i to the rest of the community, defined as the following.

$$MC_i = c(\mathcal{N}) - c(\mathcal{N} \setminus \{i\}) \tag{9}$$

The annual energy cost MC_i of agent i uses the same intuition as Eq. (7) in Shapley value calculation. But, whereas the Shapley value takes the mean marginal contribution of agent i for *every* possible subcoalition in the community, this method computes the cost by only looking at the *last* marginal contribution, making it a much more time-efficient method. However, costs based on the last marginal contributions do not hold the same property of the Shapley values in which the sum of individual cost is equivalent to the total community cost [9]. Hence, the last marginal cost needs to be normalised. The final redistributed cost $\overline{MC}_i$ of agent i according to the normalised last

marginal contribution (simply the marginal contribution method from now on) is given as:

$$\overline{MC}_i = c(\mathcal{N}) \frac{MC_i}{\sum_{q \in \mathcal{N}} MC_q} \tag{10}$$

The time complexity of the marginal contribution method is $\mathcal{O}(N)$, so, while simple, it is a very computationally efficient method.

4.1.2. Stratified expected value

The last marginal contribution method only takes into account the marginal contribution of the last stratum. Starting from this observation, we propose a novel Shapley redistribution scheme that goes a step further and considers the expected marginal contribution for *every stratum*, while still avoiding the huge combinatorial cost of the exact Shapley method. We call this the *stratified expected values* method.

Formally, for agent i, an agent profile p_{-i} that has average energy demands from the rest of the agents in the community is created. The demand of the agent profile p_{-i} at time t is calculated as:

$$d_{p_{-i}}(t) = \frac{\sum_{q \in \mathcal{N} \setminus \{i\}} d_q(t)}{N-1}, \quad \forall t \in \{1, \dots, T\} \tag{11}$$

The main idea of the method is that since p_{-i} has the average demand of the rest of the community for every time step, computing the marginal contribution from a set of agents with such a demand profile can approximate the *expected* marginal contribution of that stratum. Since the Shapley value can also be seen as the mean of expected marginal contribution of every stratum, taking the mean of approximated marginal contribution of every stratum should give an average "in expectation" value that approximates the Shapley value. Hence, the cost of agent i based on the stratified expected values method SEV_i is calculated as the following.

$$SEV_i = \frac{1}{N} \sum_{j=0}^{N-1} c(\{1, \dots, j\} \cup \{i\}) - c(\{1, \dots, j\}),$$

$$\text{such that } d_1 = \dots = d_j = d_{p_{-i}} \tag{12}$$

Similarly to the marginal contribution method, the sum of individual energy costs does not equal the community's total cost since this method uses fictitious agents with demand profiles $d_{p_{-i}}$. Hence, a normalisation step is required, given as follows.

$$\overline{SEV}_i = c(\mathcal{N}) \frac{SEV_i}{\sum_{q \in \mathcal{N}} SEV_q} \tag{13}$$

The time complexity of computing the individual costs with this method is $\mathcal{O}(N^2)$ since for each agent, it requires to calculate the average marginal contribution once for every stratum, ranging from 0 to $N-1$. While this is obviously more than the $\mathcal{O}(N)$ computation of the last marginal value method, it is still much less than the exponential cost of computing the Shapley values, and still very tractable for medium and relatively large community sizes . The stratified expected value method uses the same intuition as the last marginal contribution method: it considers the last marginal contribution with respect to the expected demand value of the other agents (thus ignoring the combinatorial explosion of computing all orders) — but it does so *for every stratum*, taking an average among them. Thus, it is intuitive to formulate a hypothesis that the stratified expected value method should give a better estimation of the Shapley value than the simpler marginal contribution method, which ignores the strata structure. We explore this hypothesis in Section 5.

4.1.3. Adaptive sampling Shapley approximation

The previous redistribution methods were deterministic, providing the same numerical results every time the redistributed costs are calculated, given the demands of the agents remain the same. We also compared the performance of a state-of-the-art, random sampling Shapley approximation method. Specifically, we implemented the adaptive sampling method using reinforcement learning introduced by O'Brien

et al. [11]. For each agent i, this method samples a subcoalition randomly from a stratum and computes the marginal contribution of agent i to the subcoalition, repeating this step for M samples predetermined by the user. After every sample, the expected marginal contribution and its estimated standard deviation (SD) of the stratum are updated. The selection of the stratum at the next sample is dependent on the estimated SDs of the strata, where strata with larger spread are more likely chosen. Such a procedure allows to sample more from strata with larger uncertainty, hence sampling more efficiently. Finally, the mean of all expected marginal contributions of the strata is computed as the cost. The details of the algorithm are given in Appendix C.

The time complexity of this redistribution scheme is $\mathcal{O}(N \cdot M)$. Note that, in principle, the number of samples required to approximate the Shapley value well increases faster as the community size increases, hence M is set to a value that is $M \gg N$. For this study, M was set to 1000 when running this method to assure multiple samples are taken from each stratum.

4.2. Exact computation of Shapley values with k classes

Given the above redistribution methods, what is really needed is the "ground truth" consisting of the exact Shapley values, to compare the performance of these approximation methods for a realistic size community (e.g. $N = 200$ prosumers behind a transformer). Prior works that do this, like O'Brien et al. [11], reduce the number of prosumers to $N = 20$ to compute the exact Shapley, but we argue this method is not really a satisfactory way to proceed. This is because, crucially, the quality of an approximation for a larger community (e.g. $N = 50$, 100 or 200 agents) can be very different than for a very small number of agents, up to 20 (we clearly show this effect in our experiments as well).

The key intuition is that, while computing the Shapley values of N unique agents requires a time complexity that is exponential to N, the computation time can be significantly reduced if the community consists of a limited number of classes of agents, where agents in the same class have the same demand profile.

Let the new model be defined as the following. A community $\mathcal{N}$ still consists of N agents, with now K classes of demand profiles in the community such that every agent belongs to one class, and all the agents in the same class have equal half-hourly demands. We assume w.l.o.g. that classes are ordered by size, i.e.

$$N \geq N_1 \geq \dots \geq N_K \geq 1 \tag{14}$$

where N_k is the size of the class k. Then, the number of all possible energy costs of subcoalition in the community is $(N_1 + 1) \times \dots \times (N_K + 1)$, since from each class k you can have 0 to N_k agents being part of the subcoalition. This is important to note because computing the annual costs of subcoalitions (the cost function) is the most computationally expensive part of computing the redistributed costs, since it has to run through one year of half-hourly demands datapoints every time the cost function is called. The energy costs of every subcoalition may be used multiple times to compute the marginal contribution in Shapley calculation, hence storing the values in a table of the dimension $(N_1 + 1) \times \dots \times (N_K + 1)$ can be time-saving. Algorithm 1 shows the creation of the table storing the costs of all possible subcoalitions in a community.

The table containing costs of every subcoalition can also be represented as a hyperrectangle of K dimensions and the size $N_1 \times \dots \times N_K$. Each axis represents the number of agents in the class. A stratum can be represented in such a hyperrectangle as a hyperplane cutting through in which the sum of axes equals the size of the stratum. Hence, strata correspond to planes parallel to each other. Fig. 2 shows an example case where $K = 3$ with $N_1 = 7$ (x-axis), $N_2 = 4$ (y-axis), and $N_3 = 2$ (z-axis). Stratum 5 is represented by the plane, where $x + y + z = 5$.

Once the energy costs of all possible subcoalitions have been computed, the Shapley values of the agents, which is the energy cost the

---

Algorithm 1 Create a table containing energy costs of every possible combinations of number of classes

---

Input: Number of prosumers in community, N. Number of classes, K. Number of prosumers in each class, $N_1, N_2, ..., N_K$ with $N_1 + N_2 + ... + N_K = N$ and $N_1 \geq N_2 \geq ... \geq, N_K \geq 1$. Demands of the classes, $d_1, d_2, ..., d_K$, where each d contains half hourly demands during time period of T (1 year).
Output: Table containing costs of all possible subcoalition combinations, CS.

1: **function** CREATETABLE_ALLSUBCOALTIONCOSTS($N, N_1, ..., N_K, d_1, ..., d_K$)
2:     **for** all $(n_1, n_2, ..., n_K) \in \prod_{k=1}^{K}\{0, 1, ..., N_k\}$ **do**   ▷ Cartesian product
3:         $S = \bigcup_{k=1}^{K}\{1, ..., n_k\}$ with demands d_k ▷ Union of sets with $1, ..., n_k$ having the same demand d_k
4:         $CS[n_1, n_2, ..., n_K] = c(S)$ ▷ Cost of subcoalition, Eq. (6)
5:     **end for**
6:     **return** CS
7: **end function**

Algorithm 2 Compute Shapley Exact values of K classes

Input: $N, N_1, ..., N_K$. Table of energy costs of subcoalitions, CS
Output: Array of Shapley values (redistributed cost) for agents in each class, $Shap$

1: **function** COMPUTEEXACTSHAPLEY($N, N_1, ..., N_K, CS$)
2:     **for** $k \leftarrow 2$ to K **do**   ▷ Iterate through every class except the first one
3:         $Shap_k \leftarrow 0$
4:         **for** $j \leftarrow 0$ to $N - 1$ **do**   ▷ Iterate through every stratum
5:             **for** all $(n_1, n_2, ..., n_K) \in \{0, 1, ..., \max(j, N_k - 1)\} \times \prod_{i=1, i \neq k}^{K}\{0, 1, ..., \max(j, N_i)\}$ **do**
6:                 **if** $j = \sum_{i=1}^{K}(n_i)$ **then**
7:                     $rel\_f \leftarrow P(\{n_1, ..., n_K\}, \{N_1, ..., N_k - 1, ..., N_K\}, N, j)$ ▷ Hypergeometric function
8:                     $mc \leftarrow CS[n_1, n_2, ..., n_k + 1, ..., n_K] - CS[n_1, n_2, ..., n_k, ..., n_K]$ ▷ Marginal contribution
9:                     $Shap_k \leftarrow Shap_k + rel\_f * mc$
10:                 **end if**
11:             **end for**
12:         **end for**
13:         $Shap_k \leftarrow \frac{1}{N} Shap_k$
14:     **end for**
15:     $Shap_1 \leftarrow \frac{1}{N_1}(CS[N_1, N_2, ..., N_K] - \sum_{i=2}^{K} N_i * Shap_i)$
16:     **return** $Shap$
17: **end function**
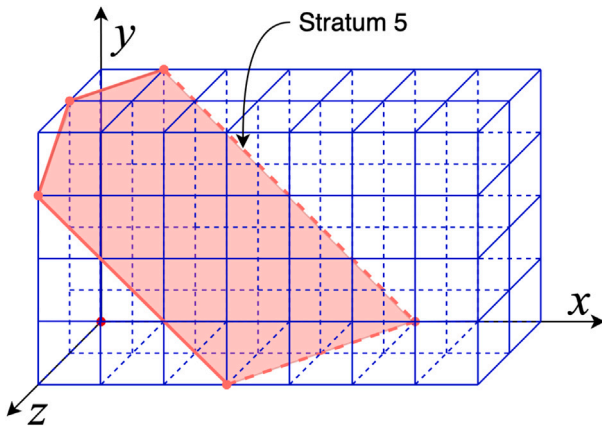
Fig. 2. A representation of all possible subcoalitions in a community of $K = 3$ classes with $N_1 = 7$, $N_2 = 4$, and $N_3 = 2$.

agent owes to the community, can be found. Because of the symmetry axiom [9], the Shapley values of agents in the same class (same energy demands) are equal, and hence it is only required to calculate the Shapley values once for each class. Algorithm 2 is used to determine the Shapley values when the community consists of K classes of agents. The algorithm first loops over the number of classes starting from $k = 2$ (Line 2, Algorithm 2). Class 1, the largest of the classes, is skipped for efficiency since it can be computed after the Shapley values of all

other classes are determined. Then, for each class, it will iterate through the strata from 0 to $N - 1$ (Line 4, Algorithm 2). Between Line 5 and 12, the Shapley value of the iterating class is updated by adding the marginal contribution. Line 5 of Algorithm 2 shows that it will iterate through every possible subcoalition of the size of the stratum. As shown in Eq. (7), the marginal contribution of an agent in a community is described as $c(S \cup \{i\}) - c(S)$. Hence, it can be seen in Line 5 that the maximum number of agents from the iterating class k is one less than N_k, so that the marginal contribution can be computed. Line 6 assures that the formed subcoalition is from stratum j.

What makes it possible to compute the Shapley values efficiently for limited number of classes is the *(multivariate) hypergeometric distribution* [67]. The probability mass function of the multivariate hypergeometric distribution $P(\{n_1, ..., n_K\}, \{N_1, ..., N_K\}, N, n)$ computes the relative frequency of selecting n_1 agents from class 1 with the size of N_1, n_2 agents from class 2 of size of N_2, and repeating until class K, in a community of N agents. The function is formulated as the following.

$$P(\{n_1, ..., n_K\}, \{N_1, ..., N_K\}, N, n) = \frac{\prod_1^K \binom{N_k}{n_k}}{\binom{N}{n}} \tag{15}$$

where $\sum_1^K N_k = N$ and $\sum_1^K n_k = n$. The hypergeometric distribution allows to compute the probability of certain set of agents to be selected ahead over the chosen agent at the specific stratum. Line 7 in Algorithm 2 shows the step where the probability of the set of agents being ahead at stratum s is computed. (The Shapley value of an agent can then be computed by replacing the relative frequency of a subcoalition in Eq. (8), with the hypergeometric function.)

Table 2
Time complexity per algorithm.

Algorithm	Time Complexity	
	Unique	K Classes
Exact Shapley	$\mathcal{O}(2^N \cdot (N-1))$	$\mathcal{O}(N^K \cdot (K-1))^{\text{a}}$
Marginal contribution	$\mathcal{O}(N)$	$\mathcal{O}(K)$
Stratified expected values	$\mathcal{O}(N^2)$	$\mathcal{O}(K \cdot N)$
Approx. Shapley RL	$\mathcal{O}(N \cdot M)$	$\mathcal{O}(K \cdot M)$

[a]Upper bound time complexity.

Line 8 in Algorithm 2 computes the marginal contribution of the agent from class k using the table containing costs of subcoalitions from Algorithm 1. The marginal contribution in Line 9 is added with the factor of the relative frequency of the subcoalition from stratum j. After iterating through every strata and subcoalitions, the value is divided by the total number of strata, which is N (Line 13). It can be seen that computation steps of Lines 8, 9, and 13 are equivalent to Eq. (8), with the only difference being the relative frequency is computed using the hypergeometric function.

Line 15 computes the Shapley value of agents in class 1. Since the values of agents of all the other classes are known and the sum of Shapley values of all agents must equal to the community cost, the Shapley value of class 1 is equal to the remaining cost after subtracting the cost distributed to agents in class 2 to K from the community cost, then equally divide it by the agents from class 1, by the efficiency property [9].

4.3. Complexity of shapley value computation

Table 2 shows the time complexities of exact Shapley values and the three approximation methods used in this study for two scenarios; when the community of size N consists of unique demand profiles and when the number of demand profiles is limited to K classes.

In the case of N unique demands, it was explained previously that it requires 2^N steps to compute the Shapley value of an agent. To compute the Shapley values of the whole community, it is required for $N-1$ agents since the value of the last agent can be determined by simply subtracting the sum of the rest of the agents' values from the total cost. This is due to the efficiency property of the Shapley value, in which the sum of the redistributed values equals the total value [9]. Hence, the time complexity of a community of unique agents is $\mathcal{O}(2^N \cdot (N-1))$.

When the community is restricted to K classes, the number of times the cost function needs to be computed by Algorithm 1 is equal to the number of all possible combinations of agents which is $(N_1 + 1) \times \cdots \times (N_K + 1)$ (illustrated in Fig. 2). Considering w.l.o.g that the classes are ordered by the size, i.e., $N_1 \geq N_2 \geq \cdots \geq N_K$, and assuming there are at least two non-empty classes, i.e. $K \geq 2$, then it holds that $N_i + 1 \leq N, \forall i = 1, \ldots, K$. The number of cost function calculations is hence upper bounded by N^K. Due to the symmetry property of the Shapley value [9], it is only required to be computed once per class. Furthermore, it is required to compute $K-1$ times with the same reasoning as in the unique demand profiles scenario. Therefore, it gives the time complexity of $\mathcal{O}(N^K \cdot (K-1))$. While it seems that $\mathcal{O}(N^K \cdot (K-1))$ (for K classes) is large, in fact, for a large N and a small number of classes K this is much smaller than 2^N, hence in practice, exact Shapley computation with K classes has a much lower computation cost than unique agents.

For the marginal contribution method (Section 4.1.1), the complexity was determined to be $\mathcal{O}(N)$ for a community of N agents, as it requires to compute the marginal contribution once for every agent. With K classes, however, this is reduced to $\mathcal{O}(K)$. Since agents with the same energy demands are assigned the same cost, Eq. (9) and (10) are only required to be computed once for each class.

The time complexity of stratified expected values method (Section 4.1.2) is $\mathcal{O}(N^2)$ for N unique agents, but it can be reduced to $\mathcal{O}(K \cdot N)$ for K classes. Similarly to the marginal contribution method, it is only required to compute Eq. (12) once per class.

Finally, RL-based Shapley approximation method (Section 4.1.3) has the time complexity reduced from $\mathcal{O}(N \cdot M)$ to $\mathcal{O}(K \cdot M)$, where M is the number of samples per agent chosen by the user, by the same reasoning as the marginal contribution and stratified expected values methods.

5. Experimental comparison

For the experimental comparison, the energy demands of 200 households in a realistically-sized energy community in the UK sharing a community wind turbine and battery were used. In this study, experiments are carried out on two scenarios. Section 5.1 presents the experimental setup of the first scenario. Here, agents of the community are grouped into two classes based on their annual consumption size (large vs. small energy consumers). The performance of the redistribution methods is tracked with increasing community size, keeping the ratio of large to small consumers constant. Section 5.2 presents the experimental setup of the second scenario. Here, agents are grouped into four classes based on their consumption profile throughout a typical day. Again, the performances of the redistribution methods are compared with increasing community size. Finally, Section 5.3 provides discussions on the results from the two scenarios. All of the experimental code was written in and run with Python 3 (version 3.8.5).

5.1. Scenario 1: Large and small consumers

Dataset and parameters. For the first experimental comparison, the energy demands of 200 households in a realistically-sized energy community in the UK sharing a community wind turbine and battery were used, using the case study from Norbu et al. [10] (with the kind permission of the authors). The energy demands of the households are provided for every 30 min during one calendar year, which was largely collected in a well-known smart energy demonstrator project in the UK, the Thames Valley Vision project [12]. The half-hourly power generated by the wind turbine was calculated based on the power curve of the Enercon E-33 wind turbines [65] and real wind data of the Kirkwall airport weather station in Orkney, Scotland from the UK Met Office Integrated Data Archive System (MIDAS) [68] provided by the British Atmospheric Data Centre (BADC). Furthermore, an import tariff of 16 UK pence/kWh and an export tariff of 0 pence/kWh were used.

Demand profiles. Two-hundred prosumers are grouped into two classes of small consumers or large consumers according to the annual energy consumption. Small consumer and large consumer profiles are made from the average half-hourly demands of each group. In this study, two cases are tested; groups split into the 90% smallest and 10% largest consumers by total annual demand, respectively, and 80% smallest and 20% largest in the second test. The community consists of agents that had small and large consumer profiles, with the corresponding ratios (9:1 or 8:2).

Setup and performance measure. Communities with small and large consumer profiles are used to compare how well the redistribution methods approximate the Shapley values. In the first setting, the ratio of the community is kept to 9:1, and the approximation performances are measured for varying community size of $N = 10$ up to $N = 200$. Similarly, the ratio is kept constant to 8:2 in the second setting, and performances of varying community size is tested.

The redistribution methods are compared to the exact Shapley values (the ground truth) of small and large agent profiles. The *relative difference* to the exact Shapley values was used for the comparison. The percentage relative difference of a cost to the Shapley value is defined as the following.

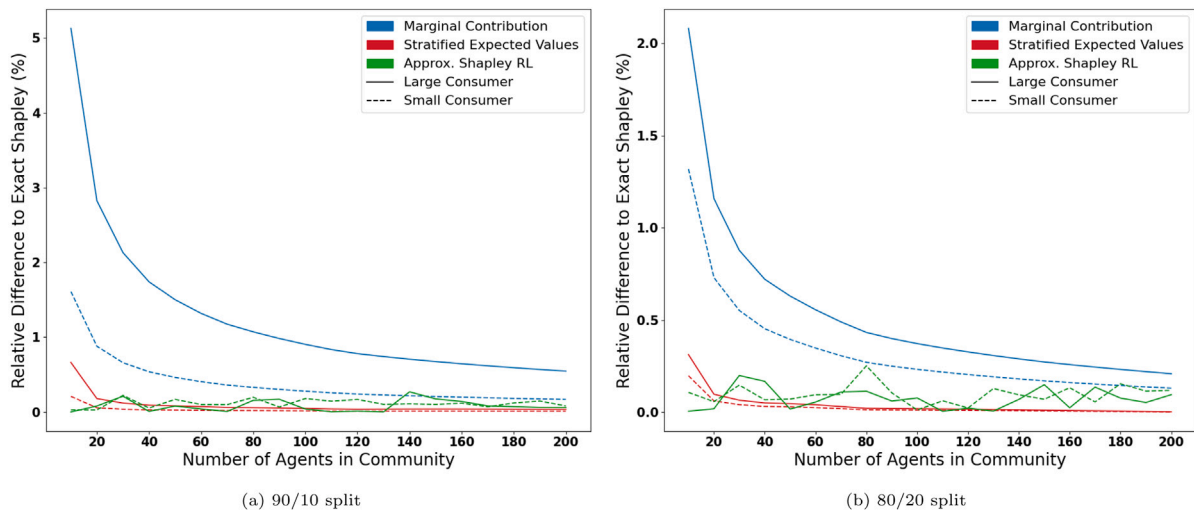$$RD_\phi(\hat{\phi}_k) = \frac{|\hat{\phi}_k - \phi_k|}{\phi_k} \times 100 \tag{16}$$

Fig. 3. Relative differences of the small and large consumer agent profiles to the exact Shapley values for the redistribution methods with increasing size of the community.

where $\hat{\phi}_k$ is the energy cost of agent of class k (in this simulation, either small or large consumer profile) from a particular redistribution method, which are $\overline{MC}_k$, $\overline{SEV}_k$, and RL_k. The variable ϕ_k is the cost redistributed to class k according to the Shapley value. The relative difference does not only take the magnitude of difference between the approximation method and the exact value, but also considers how large the exact value is. This provides a fairer evaluation between different demand profiles, as demand profiles with naturally large energy cost could have significant approximation error in terms of magnitude only from slight deviation.

Results. We investigated whether the size of the community influences how well the redistribution methods approximate the Shapley values. Fig. 3 shows the change in relative difference to the exact Shapley values of the redistribution methods with increasing community size up to 200 households while keeping the same ratio of small and large agent profiles. In Fig. 3(a), the ratio of small consumer agents and large consumer agents were kept to 9:1, while Fig. 3(b) used the ratio of 8:2.

5.2. Scenario 2: Different consumption profiles

Dataset and parameters. In the second case study, the demands and the wind data were taken from a dataset in Kaggle,[2] a ML platform. The half-hourly demands of 5567 households in the London area, UK, between November 2011 and February 2014 were recorded by the UK Power Networks during the Low Carbon London project [13]. The corresponding London weather data was provided by Dark Sky [69], and the generated power by the wind turbine was calculated using the same method as Norbu et al. [10].

Both the demands and wind power data were aggregated to generate an averaged half-hourly data points for one year, aligned by the calendar weeks. From the demand data, households were removed if less than 95% of the data points from the year were missing, resulting in 5251 households left. The remaining miss data points were filled using linear interpolation.

Similarly to the first case study, an import tariff of 16 UK pence/kWh and an export tariff of 0 pence/kWh were used.

Demand profiles. In the second case, the agents were grouped not by their total demands but rather by their consumption profiles over a day (24 h). Clustering energy consumers into a number of classes, according to their daily consumption profile, is a well-established practice in energy demand modelling [70–72], used both in research

and practice, by energy suppliers. Identifying consumption patterns of customers can help the energy provider to provide customers with recommendation as well as managing energy loads.

The 5251 agents are clustered using K-means clustering from the energy consumption of the winter months. From the resulting clusters, four groups showing distinct behaviours were chosen as consumption classes, and is presented in Fig. 4.

Fig. 4 shows the daily consumption behaviours of the four selected clusters. The first cluster on the left shows a increase in demand in the morning, then a large peak in the evening, thus named "evening peaker". The second cluster from the left has energy demands increased in the morning and stay high during the day. There is a small evening peak, but has relatively even consumption throughout the day. This group is called "stay at home", as it requires certain energy consumption during the day such as heating, computers, and kitchen appliances. The third cluster shows a large peak in the morning followed by a decreased consumption during the day, and a final large peak in the evening. It can be seen that the morning and evening peaks are roughly the same size, and therefore it is named "M-shaped" consumers. The fourth cluster had almost no energy demand during the day, but had high demand overnight. Such behaviour is also observed in [70], a study on clustering consumers on demand profiles. This group was named "night owl", and it is more of a rare case, having less than 1% of the households classified in this study.

In 2020, due to COVID-19, working from home became the norm, thus it would be of interest to look at a change of behaviour from working at the office to home. Hence looking at consumption behaviours of classes like "evening peak" and "stay at home" were chosen for this study. Furthermore, to add more variety and create a realistic community, we have included classes with distinct behaviours such as "M-shape" and "night owl" classes. From grouped agents, the half-hourly energy demands are created for four consumer profiles. The details of clustering and the production of demand profiles are given in Appendix D.

Setup and performance measure. Communities with four consumer profiles are used to perform experiments. In this case study, we perform two experiments. In the first experiment, the ratio of the community was kept constant, and the performances of the redistribution methods were tested for community sizes of $N = 10$ to $N = 200$. We tested two scenarios; one in which the community is concentrated to one class, and one in which the community is more evenly spread out between the classes. In the second setting, the community size is kept constant, but the composition (ratios) of the consumer classes in the community changes.
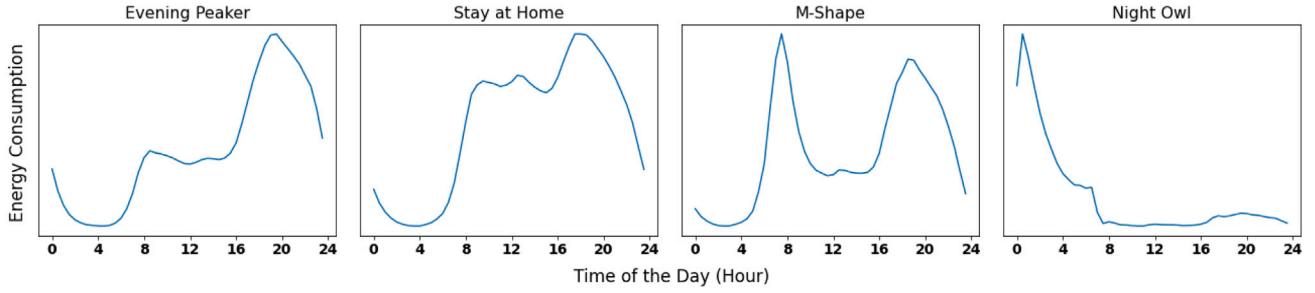
Fig. 4. Average daily energy demands of consumer profiles, evening peaker, stay at home, M-shape, and night owl.
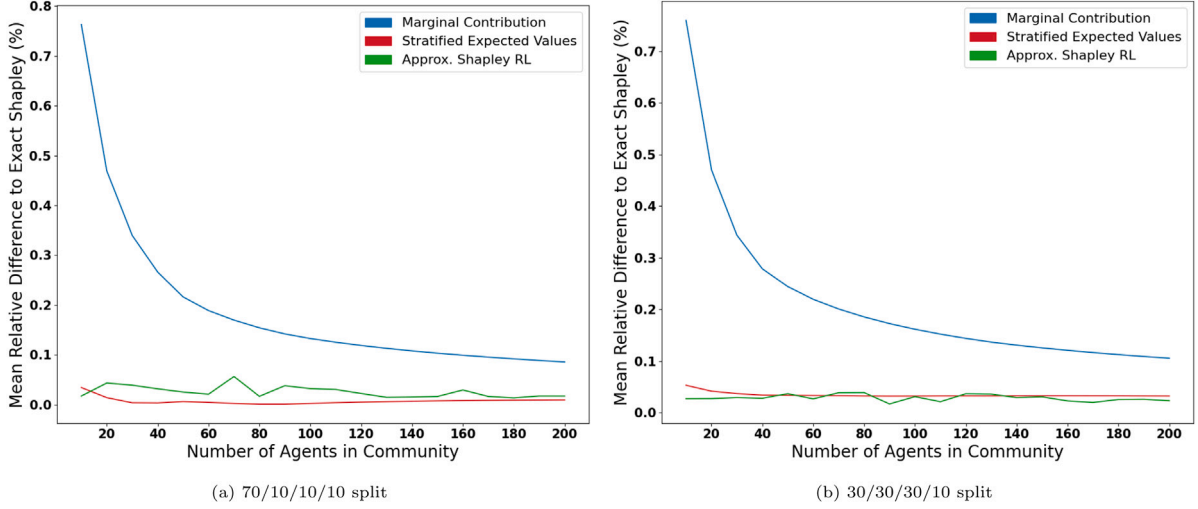

(a) 70/10/10/10 split

(b) 30/30/30/10 split

Fig. 5. Average relative differences to the exact Shapley values for the redistribution methods with increasing community size in a community with four consumption profiles.

To compare the performance of the redistribution methods, we used the *average relative difference* to the exact Shapley values. The relative difference to the exact Shapley values is as defined in Eq. (16). The average relative difference to the Shapley value of a redistribution method is the mean relative difference of the community:

$$RD_\phi(\hat{\phi}) = \frac{1}{N} \sum_k^K N_k \cdot RD_\phi(\hat{\phi}_k) \qquad (17)$$

where $\hat{\phi}$ is a redistribution method with costs $\hat{\phi}_1, \ldots, \hat{\phi}_K$ assigned to K classes.

Results. Fig. 5 shows the change in average relative differences with increasing community size, starting from $N = 10$ up to $N = 200$. Fig. 5(a) presents the result of the community with compositions of 70% "evening peak", 10% "stay at home", 10% "M-shape", and 10% "night owl", and Fig. 5(b) with compositions of 30% "evening peak", 30% "stay at home", 30% "M-shape", and 10% "night owl". Fig. 6 shows the change in average relative differences of redistribution methods with change in the composition of the community. The community size was set to 200, and the ratios of "M-shape" and "night owl" agents were also kept constant to 20% and 10% respectively. Initially, the "evening peak" class is set to be 65% of the community and "stay at home" class to 5%. After every run, the ratio of "evening peak" is reduced by 5% and "stay at home" increased by 5%, until the "stay at home" makes up 65% of the community and "evening peak" with only 5%. Detailed results showing the breakdown of the performance across the different consumer profiles are presented in Appendix E.

5.3. Discussion

Figs. 3, 5 and 6 showed that all the tested redistribution methods approximated Shapley values well for a large community. Although the

marginal contribution method yielded high errors for small community size (for example, 5% difference with exact Shapley for large consumer profile in 90/10 split scenario, Fig. 3(a)), for community size of over 100 prosumers, all methods were below 1% difference with the exact Shapley in all scenarios. We attribute the smaller difference to the exact Shapley in larger communities to a smoothing effect that a large number of agents have to individual variations. In a large community, the Shapley calculation is dominated by marginal contributions of the agent to already large-sized subcoalitions. Variations in marginal contributions to large subcoalitions are often small, making it possible for less complex methods to approximate well for large communities.

The stratified expected values method outperforms the simpler marginal contribution method in all cases and all scenarios, hence the intuitive hypothesis we formulated in Section 4.1.2 clearly holds. Furthermore, there is a minimal difference in the performances between the stratified expected values and the adaptive sampling methods in most cases. The number of samples per agent was set to 1000 for the adaptive sampling method, meaning that for a case of 100 prosumers community, the adaptive sampling method had a time complexity ten times higher than the stratified expected values method (from Table 2). Yet the figures show that the stratified expected values method outperform the adaptive sampling method in many scenarios and perform comparatively overall. In fact, paired two-sample t-tests on 2 class experiments with 0.05 significance level showed that the stratified expected values method (90/10 split: $M = 0.0339$, $SD = 0.0520$. 80/20 split: $M = 0.0285$, $SD = 0.0471$) had a smaller average relative difference to true Shapley values compared to RL-based adaptive sampling method (90/10 split: $M = 0.1110$, $SD = 0.0496$. 80/20 split: $M = 0.0934$, $SD = 0.0444$) for both 90/10 split($t(19) = -3.919$, $p < 0.001$) and 80/20 split ($t(19) = -4.236$, $p < 0.001$). Furthermore, in the 4 class case with community concentrated to 1 class (Fig. 5(a)), the
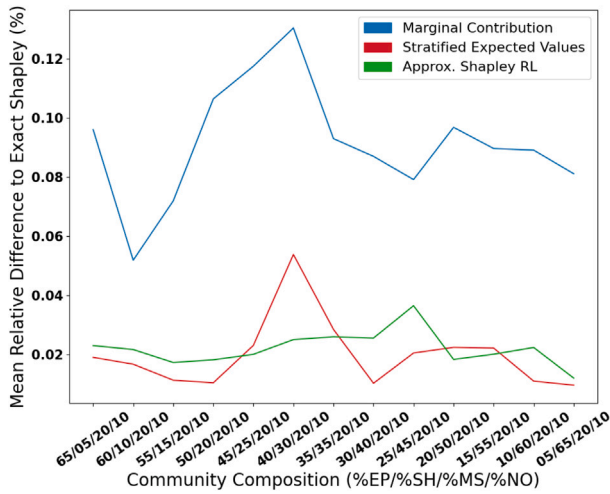
Fig. 6. Average relative differences to the exact Shapley values for the redistribution methods of different community compositions with $N = 200$.

stratified expected values ($M = 0.0071$, $SD = 0.0070$) outperformed the adaptive sampling method ($M = 0.0254$, $SD = 0.0114$) ($t(19) = -5.376$, $p < 0.001$). Only in the case of evenly spread 4 class community (Fig. 5(b)), the adaptive sampling method ($M = 0.0283$, $SD = 0.0062$) outperformed the stratified expected values ($M = 0.0341$, $SD = 0.0048$) ($t(19) = 3.173$, $p = 0.005$). Hence from these results, it seems that the stratified expected values method does well approximating the Shapley values when the community is concentrated to one class, and outperforms the state-of-the-art sampling method.

When looking at how the composition of the community affects the performances of the redistribution methods in Fig. 6, all three methods approximate Shapley values well (all methods in every scenario less than 0.15% difference), as the community size is large already. Still, the stratified expected values and the RL-based adaptive sampling methods outperform the marginal contribution method. A paired t-test with 0.05 significance level showed that there is no significant difference between the stratified expected values ($M = 0.0199$, $SD = 0.0114$) and the adaptive sampling ($M = 0.0220$, $SD = 0.0056$) methods on their average relative differences to the Shapley values ($t(12) = -0.660$, $p = 0.52$). Yet it can be seen from Fig. 6 that the stratified expected values method has smaller difference to true Shapley values when the community is concentrated on one class, and shows larger errors when the community is more even. This in line with the findings from Fig. 5. It can be seen in Appendix D that most consumers had a "evening peak" and made up 60% of the households studied. Hence it is common to have a community that contains majority of the same consumption behaviour class, making the stratified expected values method desirable in real-world scenarios.

Although it approximates the exact Shapley values very well, a potential disadvantage of the RL-based method (and any method using random sampling) is that the redistributed values can vary every time the algorithm is run. The fluctuating performance of the RL-based method can be seen in Figs. 3 and 5. In practice, the random output of the method can have an undesirable effect on the *perceived fairness* of the redistribution, as prosumers with the same demand profile can result in being assigned slightly different costs.

6. Conclusions & further work

While the use of the Shapley value is increasingly popular in energy systems, previous works often sidestep the issue of how it can be efficiently computed in large, realistically-sized settings. The issue is made more pressing by the increasing popularity of community energy

projects, where prosumers share joint renewable generation and storage assets and costs.

This paper aims to close this gap by proposing a new method to efficiently approximate the Shapley value, and characterising both their computational complexity and performance (in terms of distance to the exact Shapley value), using large-scale, realistic case studies of energy communities in the UK. We compare the performance of the new method with an already-existing deterministic method and a non-deterministic, state-of-the-art sampling method. Moreover, in order to develop a "ground truth" benchmark to compare these approximations, we propose a novel method to compute the Shapley value *exactly* even for large population sizes by clustering agents into a smaller number of consumption profiles or classes.

Our experimental analysis shows that the relative difference to the true Shapley (while large for a few agents) converges to under 1% for larger scenarios, basically for all methods considered. In particular, in almost all scenarios studied, the newly proposed stratified expected value method and the state-of-the-art adaptive sampling method perform extremely close to true Shapley values. Interesting to observe that the stratified expected value method performs similarly to the adaptive sampling method [11] for large populations, although its computational cost is often much lower. In fact, the stratified expected values method outperformed the adaptive sampling method when the community was concentrated to one class, showing a high potential for application in real-world energy communities.

There are a number of directions we find promising to explore in future work. An interesting question to explore is the case when the local distribution network, where the energy community is based is subject to physical capacity constraints (voltage, power) [24]. Such constraints could potentially restrict all prosumers to participate in the scheme equally at certain times, and would lead to changes in the coalitional game, as well as in the computation of fair redistribution payments based on the Shapley value. Another possible improvement on our current model can be made by providing a more detailed cost calculation of the assets, such as in [73]. Although our model takes into account battery degradation for a more accurate annual cost of the battery, a better overall cost estimation can be achieved by considering a longer period of time and taking into account the investment and maintenance costs of these assets, as well as economic factors such as the inflation rate.

We are also considering extending this work, by implementing our redistribution strategies in a blockchain-enabled smart contract (such as in [74,75]), which would commit the members of the energy community to a protocol to share the benefits and costs. Based on systematic reviews on blockchains in energy systems [75,76], smart contracts should allow a more decentralised energy system, while preserving the privacy of individual prosumer data, such as demand data. The marginal contribution and stratified expected values methods used in this study already do have favourable characteristics for preserving privacy, as they do not require other prosumers' individual consumption information, only the aggregate consumption of the community. The use of smart contracts could further strengthen the protection of sensitive information and a more secure asset monitoring.

Finally, while this paper focuses on the key topic of Shapley value computation, there are many other fairness concepts that could be explored in energy applications, and it would be relevant to compare their outcome to Shapley values. Conversely, there are many promising concepts proposed in coalitional game theory literature [8] that – to our knowledge, have been explored much less in energy applications – such as the least-core [77] or the nucleolus [16]. The application and adaptation of such fairness concepts in energy could be a fruitful area, for both research and practice, providing energy communities with the computational tools to make best use of shared energy assets.

CRediT authorship contribution statement

Sho Cremers: Methodology, Formal analysis, Investigation, Software, Validation, Visualisation, Writing – original draft, Writing – review & editing, Project administration. **Valentin Robu:** Conceptualization, Formal analysis, Investigation, Writing – original draft, Supervision, Funding acquisition. **Peter Zhang:** Investigation, Visualisation, Software, Writing – review & editing. **Merlinda Andoni:** Validation, Writing – review & editing, Resources, Supervision. **Sonam Norbu:** Software, Resources, Validation, Writing – review & editing. **David Flynn:** Writing – review & editing, Supervision, Project administration.

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.

Data availability

Data will be made available on request.

Acknowledgements

Appendix A. Battery control algorithm equations

The battery keeps track of its SoC level and the power of the battery, $p^{\text{bat}}(t)$. The power of the battery is negative when charging and positive when discharging. While the battery is charging, the SoC level needs to remain below or equal to the maximum battery capacity, SoC^{max}. In addition, the magnitude of $p^{\text{bat}}(t)$ cannot exceed the maximum (dis)charging power of the battery, $p^{\text{bat, max}}$. These constraints are expressed as the followings.

$$SoC(t) \leq SoC^{\text{max}} \tag{A.1}$$

$$\left| p^{\text{bat}}(t) \right| \leq p^{\text{bat, max}} \tag{A.2}$$

When discharging the battery, similar constraints apply. First, the SoC level cannot go below the minimum battery capacity, SoC^{min}. Second, the magnitude of the battery power may not exceed the maximum discharging power. These constraints are represented as the following.

$$SoC(t) \geq SoC^{\text{min}} \tag{A.3}$$

$$\left| p^{\text{bat}}(t) \right| \leq p^{\text{bat, max}} \tag{A.4}$$

The heuristic-based battery control algorithm is described as the following. When the generated power from the RES is greater than the demand ($g(t) > d(t)$), the excess power can be used to charge the battery. However, if the battery is already full or the power exceeds the maximum charging power $p^{\text{bat, max}}$, not all the energy can be stored in the battery, and the surplus power will be sold to the utility grid. The updated $p^{\text{bat}}(t)$, the SoC level, and the exported energy to the utility grid ($e^s(t)$) are determined as the following:

$$p^{\text{bat}}(t) = -\min\left(\min((g(t) - d(t)), p^{\text{bat, max}}), \frac{SoC^{\text{max}} - SoC(t-1)}{\eta^c \Delta t}\right) \tag{A.5}$$

$$SoC(t) = SoC(t-1) - \eta^c p^{\text{bat}}(t)\Delta t \tag{A.6}$$

$$e^s(t) = (g(t) - d(t) + p^{\text{bat}}(t))\Delta t \tag{A.7}$$

where η^c is the charging efficiency and Δt is the duration of time step t in hours. The profit from exporting the energy to the grid can be expressed as the product of the energy exported, $e^s(t)$, and the export tariff, $\tau^s(t)$.

Similarly, if the demand is greater than the generated power from RES ($g(t) < d(t)$), the battery is discharged to meet the demand. If the power supplied from discharging the battery is still not enough to meet the demand, energy needs to be imported from the utility grid, denoted as $e^b(t)$. The followings are the updated $p^{\text{bat}}(t)$, SoC levels, and $e^b(t)$ in case of shortage of power:

$$p^{\text{bat}}(t) = \min\left(\min((d(t) - g(t)), p^{\text{bat, max}}), \frac{\eta^d}{\Delta t}(SoC(t-1) - SoC^{\text{min}})\right) \tag{A.8}$$

$$SoC(t) = SoC(t-1) - \frac{p^{\text{bat}}(t)}{\eta^d}\Delta t \tag{A.9}$$

$$e^b(t) = (d(t) - g(t) - p^{\text{bat}}(t))\Delta t \tag{A.10}$$

where η^d is the discharging efficiency. Again, the cost of importing energy from the grid at time t is the product of imported energy, $e^b(t)$, and the import tariff, $\tau^b(t)$.

Appendix B. Battery degradation model

In this section, the battery degradation model used in this study and developed by Norbu et al. [10] is described.

Acceleration of battery degradation caused by frequent charging and discharging operations as well as deep discharging may shorten the battery lifetime to be less than the one specified by the manufacturer. With shortened lifetime, the community needs to replace the battery earlier, incurring additional costs to the households. Hence, the battery degradation model takes this factor into account when calculating the annual cost of the battery for a more accurate representation of the real-world simulation.

The number of cycles and depth of discharge (DoD) influences battery degradation. A full cycle is defined as SoC returning to the starting value after a discharging and charging phase. On the other hand, a half cycle is defined to be simply the charging or discharging phase. Furthermore, a cycle can be classified as regular or irregular. A regular cycle starts the cycle with the SoC of 100%, whereas an irregular cycle has the starting SoC to be other than 100%. Although regular and irregular can have the same DoD (for example, a regular cycle of SoC 100% discharged to 50% and charged to 100%, and an irregular cycle of SoC starting at 80%, discharged to 30% and charged to 80% both have a DoD of 50%), the battery is depreciated differently. In this study, the rainflow cycle counting algorithm [78] is used to count the number of full and half cycles as well as identify whether the cycle is regular or irregular.

By counting the number of cycles during the time period, the depreciation factor (DF) of the battery is computed to estimate the battery useful lifetime. As mentioned before, regular and irregular cycles influence the depreciation factor differently. Hence, DF can be defined as the following.

$$\text{DF} = \text{DF}^{\text{regular}} + \text{DF}^{\text{irregular}} \tag{B.1}$$

where $\text{DF}^{\text{regular}}$ and $\text{DF}^{\text{irregular}}$ are the depreciation factors of regular and irregular cycles respectively. The depreciation factor of regular cycles is determined as follows,

$$\text{DF}^{\text{regular}} = \sum_{DoD=0\%}^{100\%} \frac{n_{cycles}^{\text{DoD,regular}}}{N_{cycles}^{\text{DoD,max}}} \tag{B.2}$$

where $n_{cycles}^{\text{DoD,regular}}$ is the number of regular cycles at a certain DoD value during the time period, and $N_{cycles}^{\text{DoD,max}}$ is the lifetime of the battery in number of cycles for that DoD value given by the manufacturer. This study used the battery cycle life data of a lithium battery from the work of Xu et al. [79]. The depreciation factor of irregular cycles is expressed as the following.

$$\text{DF}^{\text{irregular}} = \sum_{l \in L} n_l \times \left| \frac{1}{N_{cycles}^{\text{DoD}^{eq}(SoC_l^{Start}),max}} - \frac{1}{N_{cycles}^{\text{DoD}^{eq}(SoC_l^{End}),max}} \right| \tag{B.3}$$

where L is the set of all irregular cycles, and cycle l's starting and ending SoC levels, SoC_l^{Start} and SoC_l^{End} respectively. The value of n_l is determined by whether the cycle l is full or half, defined as the following.

$$n_l = \begin{cases} \frac{1}{2}, & \text{if } l \text{ is a half cycle} \\ 1, & \text{if } l \text{ is a full cycle} \end{cases} \tag{B.4}$$

Finally, $N_{cycles}^{\text{DoD}^{eq}(SoC_l^{Start}),max}$ is the lifetime in number of cycles for $\text{DoD}^{eq}(SoC_l^{Start})$, a DoD of a cycle equivalent to starting at 100% SoC and ending at the value of SoC_l^{Start}. $\text{DoD}^{eq}(SoC_l^{Start})$ is computed as the following.

$$\text{DoD}^{eq}(SoC_l^{Start}) = 100 - \left(\frac{SoC_l^{Start}}{SoC^{\max}} \times 100 \right) \tag{B.5}$$

$\text{DoD}^{eq}(SoC_l^{End})$ and $N_{cycles}^{\text{DoD}^{eq}(SoC_l^{End}),max}$ are similarly computed.

DF resulting from the former calculation is used in Eq. (5) for computing the cost of battery during the time period.

Appendix C. Full algorithm of adaptive sampling shapley approximationx

In this section, the details of the RL-based Shapley approximation algorithm by O'Brien et al. [11] from Section 4.1.3 are described.

Prior to running the algorithm, the number of samples per agent, M, is predefined by the user. Then, for each agent i, the estimated expected marginal contributions of stratum j, $\hat{\mu}_{i,j}$ is initialised to 0. Similarly, the count of how many times stratum j was visited, $h_{i,j}$, and the sum of squared differences from the current mean of stratum j, $m2_{i,j}$, are initialised to 0. Finally, the estimated standard deviation of the marginal contributions of stratum j, $\hat{\sigma}_{i,j}$ is initialised to a very large value (here, 10,000).

For every sample of agent i, the stratum j is selected according to the probabilities of each stratum at the certain sample. The probability of stratum j being selected for agent i's mth sample $\pi_{i,j}(m)$ is given as the following.

$$\pi_{i,j}(m) = \frac{\epsilon(m)}{N} + (1 - \epsilon(m)) \frac{\hat{\sigma}_{i,j}}{\sum_{s=0}^{N-1} \hat{\sigma}_{i,s}} \tag{C.1}$$

where $\epsilon(m)$ is a double sigmoid function which helps exploration at the beginning (small m value) and exploitation near the end (large m), defined as the following.

$$\epsilon(m) = 1 + \frac{1}{1 + e^{\frac{\gamma}{\beta}}} - \frac{1}{1 + e^{-\frac{m - \gamma M}{\beta M}}} \tag{C.2}$$

The parameters β and γ were set to 0.075 and 0.2 respectively during the experiment since it was found that the sigmoid function with these parameter setting to approximate ideal sampling well [11]. To approximate the Shapley value well, more samples may be required from certain strata in which the marginal contribution values can vary highly. On the other hand, if the marginal contributions are similar within a stratum, such strata would not need large samples to be approximated. Eq. (C.1) helps to distribute the samples in such a way.

Once a stratum is chosen, a subcoalition S is chosen randomly from the selected stratum (i.e., $|S| = j$), and the marginal contribution of agent i to the subcoalition is computed as $mc = c(S \cup \{i\}) - c(S)$. The difference between the sampled marginal contribution and the estimated expected marginal contribution, $\Delta = mc - \hat{\mu}_{i,j}$ is also calculated. Then, the variables are updated after each sample as followings.

$$h_{i,j} \leftarrow h_{i,j} + 1 \tag{C.3}$$

$$\hat{\mu}_{i,j} \leftarrow \hat{\mu}_{i,j} + \frac{\Delta}{h_{i,j}} \tag{C.4}$$

$$m2_{i,j} \leftarrow m2_{i,j} + \Delta(mc - \hat{\mu}_{i,j}) \tag{C.5}$$

Furthermore, $\hat{\sigma}_{i,j}$ is also updated if the stratum has been visited more than once by agent i, as the following.

$$\hat{\sigma}_{i,j} \leftarrow \sqrt{\frac{m2_{i,j}}{h_{i,j} - 1}} \tag{C.6}$$

Once all the variables are updated, it moves on to the next sample. After M samples are taken for agent i, the redistributed energy cost according to RL-based Shapley Approximation method, RL_i is calculated by taking the mean of expected marginal contributions over the strata, i.e.,

$$RL_i = \frac{1}{N} \sum_{j=0}^{N-1} \hat{\mu}_{i,j} \tag{C.7}$$

A difference from the original work is that strata 0 and $N - 1$ are only chosen once each since no matter how many times these strata are sampled, they will always have the same marginal contributions (since there is only 1 possible subcoalition). By doing so, more samples can be used in different strata, allowing the algorithm to make use of samples slightly more efficiently.

Appendix D. Clustering & consumer profiles

For the experiments from Section 5.2, the 5251 consumers from the Kaggle dataset are clustered with the following steps. The half-hourly energy demands were normalised using L2 normalisation for each agent. From the normalised dataset, only the winter months of the UK were kept, which were January, February, November, and December. This is because the energy consumption is larger during winter, and hence clearer consumption patterns should be observable. From the remaining data, the days on and near Christmas and new year were also removed, which were January 1st to 6th and December 22nd to 31st. This is because many households would likely have abnormal consumption behaviour, such as being absent from home during these days or not working. Finally, only weekdays (from Monday to Thursday) were kept so that working days are what is being clustered. The remaining data consisted of 60 days×48 normalised demands per day = 2880 datapoints per agent. The new dataset is aggregated so that it contains averaged half-hourly normalised demands of each agent (48 datapoints per agent). Then, K-means clustering is used to group the agents on winter days' energy consumption. The elbow method was used to determine the number of clusters to be 9.

The average daily demands and the sizes of the 9 resulting clusters are presented in Fig. D.7. It can be seen from the figure that although the detailed consumption behaviours are unique, many classes shows similarity in terms of having a small morning peak in the morning and a evening peak. The cluster classes 1, 2, 3, 7, and 9 shows variations of such behaviour, and it makes up more than 60% of the 5251 consumers. Note that the data is before the onset of COVID-19 pandemic, and
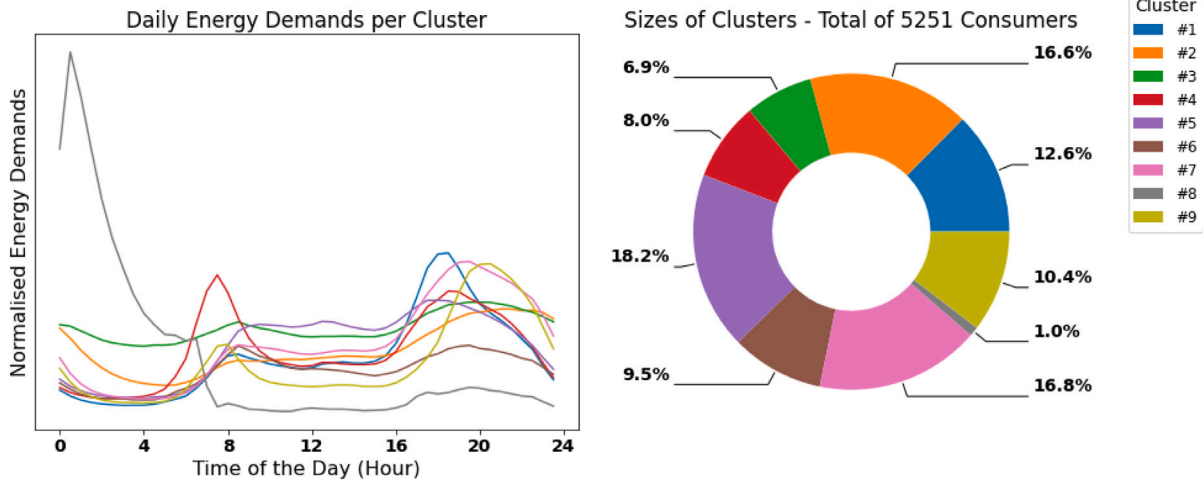
Fig. D.7. Daily energy demands and the relative sizes of the 9 consumption clusters.

(a) Evening Peak

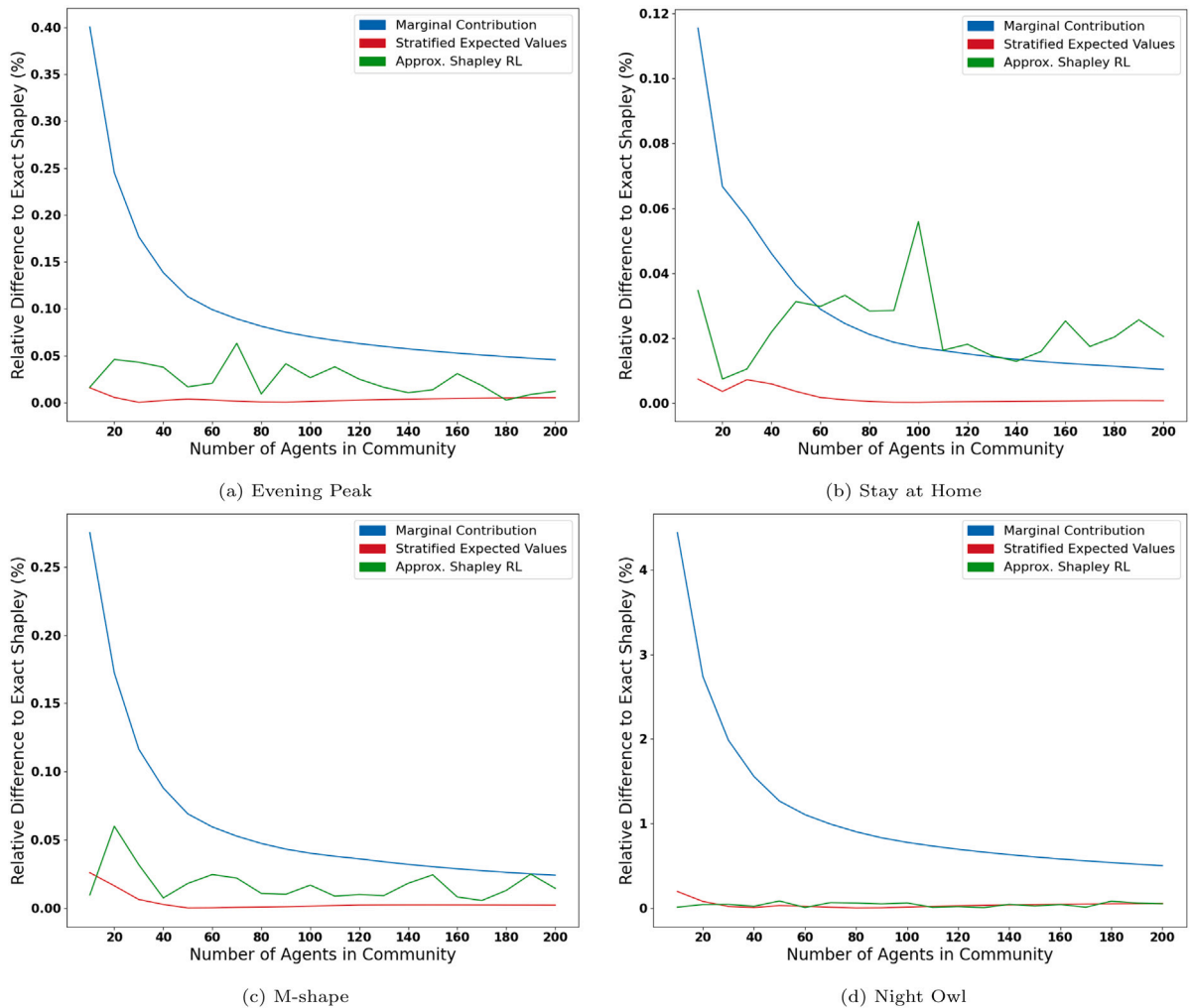(b) Stay at Home

(c) M-shape

(d) Night Owl

Fig. E.8. Individual relative differences to the exact Shapley values for the redistribution methods of four consumer profiles (70% "evening peak", 10% "stay at home", 10% "M-shape", and 10% "night owl") with increasing size of the community.

therefore working from the office during the day was common, making the consumption during the day low. We have chosen the "evening peak" class and used the demand profile of class 7 as the representative as it is the largest class out of the evening peaking classes. Next, class 5 shows a high and constant consumption during the day. This behaviour can be thought of as the household working at home as staying at home requires certain energy consumption during the day such as for heating, computers, and kitchen appliances, resulting in an overall high
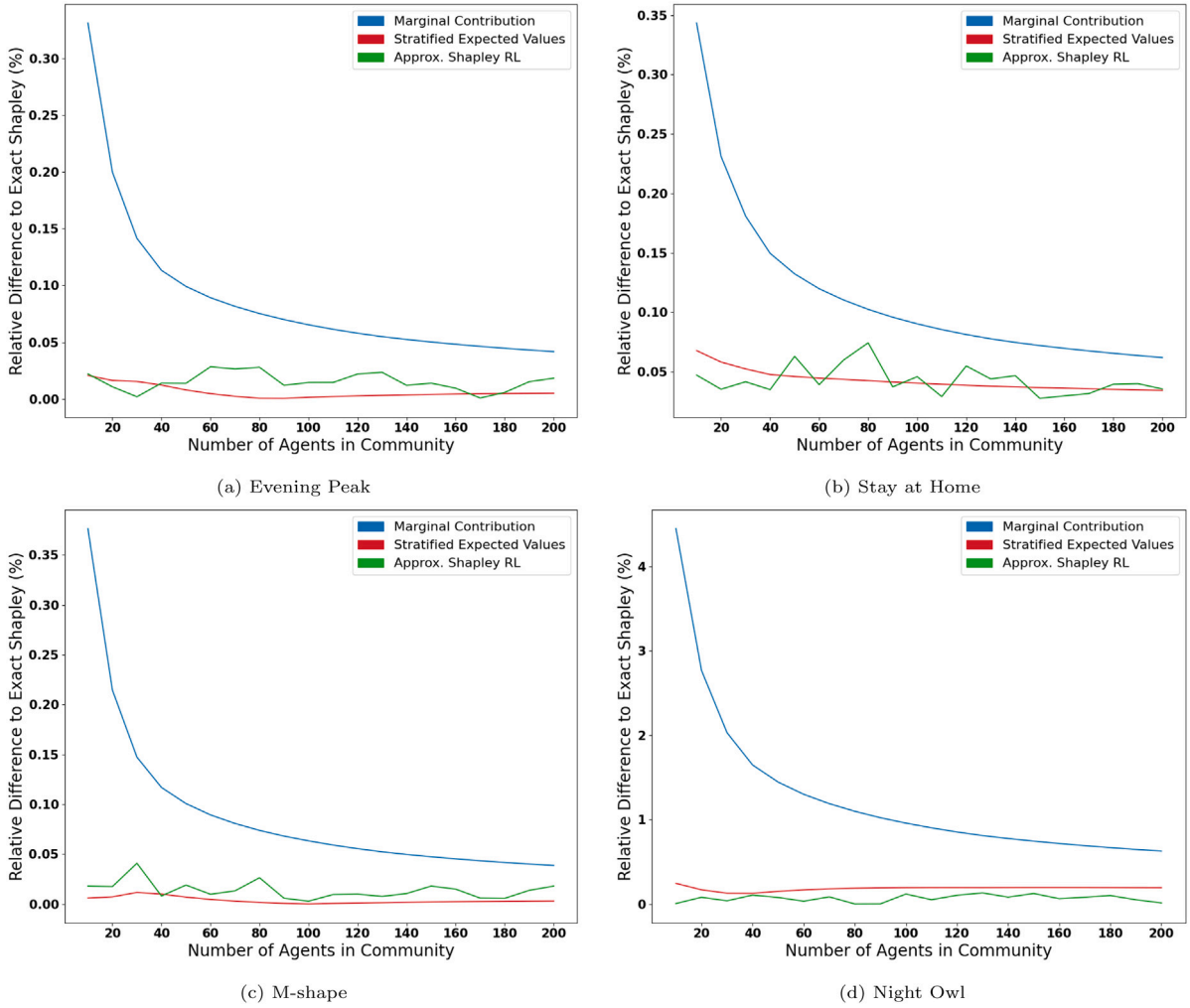
(a) Evening Peak

(b) Stay at Home

(c) M-shape

(d) Night Owl

Fig. E.9. Individual relative differences to the exact Shapley values for the redistribution methods of four consumer profiles (30% "evening peak", 30% "stay at home", 30% "M-shape", and 10% "night owl") with increasing size of the community.

consumption that is not too concentrated in the evening. We suspect that consumers with such consumption behaviour has increased since the outbreak of COVID-19, and hence class 5 was chosen as the "stay at home" class to see the impact of such a behaviour in the community. Classes 4 and 6 are unique from the rest of the classes as they both have two almost equal peaks in the morning and in the evening, though the morning peak is slightly larger. This behaviour is a minority in the community, yet 17.5% of the consumers belong in these groups. Hence, these consumers needs to be represented in the community as well, and the demand profile of class 4 is used for the "M-shaped" class in this paper as it shows the behaviour more clearly. Finally, one clear outlier cluster is class 8. Although only 1% of the consumers belongs to this class, it has the most distinct behaviour from the rest of the classes. This class has a very high demand during the night, yet very little demand during the day. This "night owl" class was chosen as the fourth class for this study as the outliers of the community.

Once the clustering is done and consumption behaviours are identified, half-hourly energy demands for 1 year for the four consumer profiles need to be identified. For each class, the L2 normalised half-hourly demands of the year (no days removed and not aggregated) were averaged over the agents belonging in that cluster. The yearly demand of the community of these agents is kept equal to N times the average yearly consumption of the 5251 agents from the original dataset. Given the sizes of the four classes N_{ep}, N_{sh}, N_{ms}, and N_{no} and $N = N_{ep} + N_{sh} + N_{ms} + N_{no}$, the restriction is represented as the following.

$$\frac{N}{200} \sum_{i=1}^{200} \sum_{t=1}^{T} d_i(t) = \sum_{k \in \{ep,sh,ms,no\}} \sum_{t=1}^{T} N_k d_k(t) \tag{D.1}$$

From the restriction, the demand profiles of each class can be computed as the following, given $\tilde{d}_k(t)$, the normalised demand of class k at time step t.

$$d_k(t) = \tilde{d}_k(t) \cdot \frac{\frac{N}{200} \sum_{i=1}^{200} \sum_{t=1}^{T} d_i(t)}{\sum_{q \in \{ep,sh,ms,no\}} \sum_{t=1}^{T} N_q \tilde{d}_q(t)}, \tag{D.2}$$
$$\forall k \in \{ep, sh, ms, no\}, \forall t \in \{1, \dots, T\}$$

Appendix E. Additional experimental results

Additional experimental results are presented here, in particular in terms of the performance of the Shapley approximations for the different types of consumer classes in Section 5.2.

First, Table E.3 shows the comparison of redistributed yearly energy costs of a small and large consumer agent for both 90/10 and 80/20 splits of 200 households community from Section 5.1. It can be seen that, for both splits, the energy costs of the small consumer agent are very close across the four redistribution methods, although the redistributed costs based on the marginal contribution are slightly higher than the other methods. On the other hand, the marginal contribution method is lower than the other methods for the large consumer
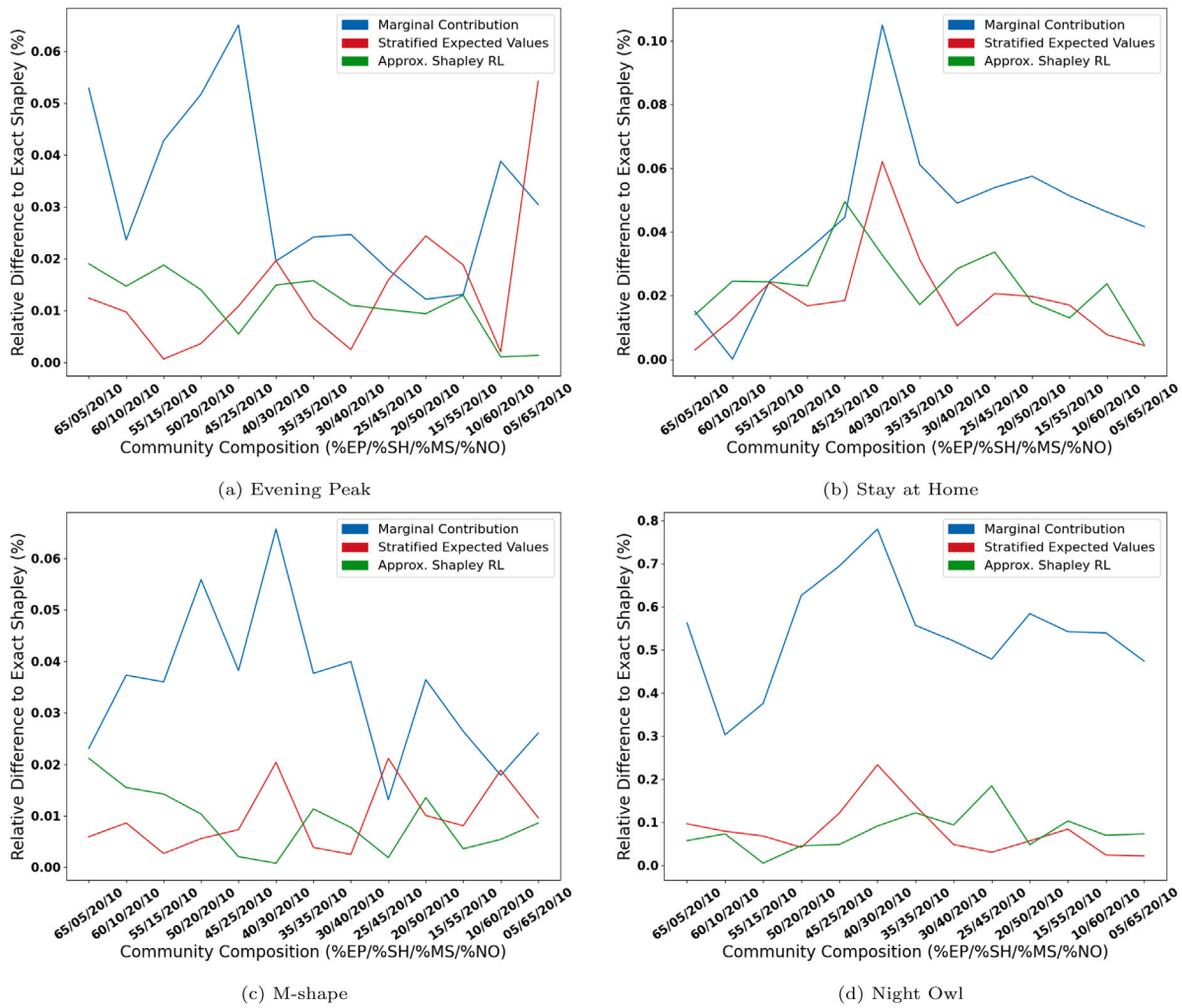
(a) Evening Peak

(b) Stay at Home

(c) M-shape

(d) Night Owl

Fig. E.10. Individual relative differences to the exact Shapley values for the redistribution methods of four consumer profiles of different community compositions with $N = 200$.

Table E.3

Redistributed costs of small and large consumer agents in a community of 200 agents.

Split	90/10		80/20	
Agent class	Small	Large	Small	Large
Shapley	245.30	674.88	221.78	554.18
Marginal contribution	245.71	671.19	222.07	553.02
Stratified expected values	245.33	674.64	221.78	554.16
Approx. Shapley RL	245.50	674.47	222.04	553.65

agent. The values for the large consumer agent are still similar across the methods, but the difference between the marginal contribution methods and the Shapley value is more significant compared to the small consumer agent.

Similarly, Table E.4 shows the comparison of redistributed costs of four consumption profiles in a community of 200 agents from Section 5.2. All methods approximate the Shapley value well for every consumption profile. Especially, the differences between the Shapley values for "evening peaker", "stay at home", and "M-shape" agents are very small. For the "night owl" agent, the difference is slightly larger for marginal contribution method, but the difference is still only about 0.6% for the 70/10/10/10 composition and 0.5% for the 30/30/30/10 composition.

Fig. E.8 contains the relative differences to the exact Shapley values of the redistribution methods for each consumer profile ("evening peak", "stay at home", "M-shaped", and "night owl") with increasing community size and the composition of 70/10/10/10 (concentrated community), from Section 5.2. It can be seen that for all classes, the marginal contribution method has similar performance curve, where the error is very large for small community size but decreases quickly as the community size increases. Still, the error value is significantly larger for "night owl" class (Fig. E.8(d)) than the other three. However, it is also noticeable for "stay at home" agent (Fig. E.8(b)) that it outperforms the state-of-the-art adaptive sampling method from medium-sized communities ($N \geq 70$). Stratified expected values method, on the other hand, shows high similarity to the exact Shapley values for any community size, and outperforms the simpler marginal contribution method in every scenario. Furthermore, it also outperforms the computationally larger adaptive sampling method in most scenarios, especially for "evening peak" (Fig. E.8(a)), "stay at home" (Fig. E.8(b)), and "M-shape" (Fig. E.8(c)) agents. The RL-based adaptive sampling method also showed high performance regardless of the community size. Yet, it can be seen from all classes that the performance can

Table E.4

Redistributed costs of four consumer profiles ("evening peak":EP, "stay at home":SH, "M-shaped":MS, and "night owl":NO) in a community of 200 agents with two different compositions.

Split	70/10/ 10/10				30/30/30/10			
Agent class	EP	SH	MS	NO	EP	SH	MS	NO
Shapley	341.55	369.41	325.76	239.86	334.90	364.49	321.34	234.65
Marginal contribution	341.70	369.45	325.84	238.65	335.04	364.71	321.47	233.18
Stratified expected values	341.56	369.41	325.77	239.72	334.92	364.61	321.35	234.19
Approx. Shapley RL	341.59	369.48	325.81	239.73	334.96	364.62	321.40	234.62

significantly vary between runs or scenarios due to the random nature of the method.

Similarly, Fig. E.9 shows the relative differences of each class with increasing community size and community composition of 30/30/30/10 (even community) from Section 5.2. Again, the marginal contribution method shows fast improvement in performances as the community size grow for every class. Still, it is outperformed by the stratified expected values and the adaptive sampling methods. The stratified expected values does not perform as well as in Fig. E.8, only outperforming the adaptive sampling method on "M-shape" class (Fig. E.9(c)). Although there seems little difference in the overall performances between the two methods for "evening peak" (Fig. E.9(a)) and "stay at home" (Fig. E.9(b)) classes, the adaptive sampling outperforms the stratified expected values for "night owl" class (Fig. E.9(d)) with relatively larger values. In fact, the "night owl" class is the main contributor of the overall error of the stratified expected values method in Fig. 5(b).

Finally, Fig. E.10 shows the relative differences of each class with changing community composition from Section 5.2. Again, the marginal contribution method shows higher error than the other two redistribution methods. It is also noticeable that the trends of the lines of the marginal contribution and the stratified expected values methods are almost identical between Fig. 6 and Fig. E.10(d), displaying that the error caused by the "night owl" class is the main contribution of the overall error of these two methods. Overall, this shows that the precision of the estimation methods is sensitive to more unusual, rarer demand profiles.

References

[1] Commission E. Communication from the commission to the European parliament, the council, the European economic and social committee and the committee of the regions 'fit for 55': delivering the EU's 2030 climate target on the way to climate neutrality. 2021, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52021DC0550.

[2] GOVUK. Biggest ever renewable energy support scheme opens. 2021, [Online] URL: https://www.gov.uk/government/news/biggest-ever-renewable-energy-support-scheme-opens. (Accessed 21 January 2022).

[3] Capper T, Gorbatcheva A, Mustafa MA, Bahloul M, Schwidtal JM, Chitchyan R, et al. Peer-to-peer, community self-consumption, and transactive energy: A systematic literature review of local energy market models. Renew Sustain Energy Rev 2022;162:112403. http://dx.doi.org/10.1016/j.rser.2022.112403, URL: https://www.sciencedirect.com/science/article/pii/S1364032122003112.

[4] Tushar W, Yuen C, Saha TK, Morstyn T, Chapman AC, Alam MJE, et al. Peer-to-peer energy systems for connected communities: A review of recent advances and emerging challenges. Appl Energy 2021;282:116131. http://dx.doi.org/10.1016/j.apenergy.2020.116131.

[5] Mengelkamp E, Gärttner J, Rock K, Kessler S, Orsini L, Weinhardt C. Designing microgrid energy markets: A case study: The brooklyn microgrid. Appl Energy 2018;210:870–80. http://dx.doi.org/10.1016/j.apenergy.2017.06.054.

[6] Ramchurn SD, Vytelingum P, Rogers A, Jennings N. Agent-based control for decentralised demand side management in the smart grid. In: The 10th international conference on autonomous agents and multiagent systems. 2011, p. 5–12.

[7] Zhang C, Wu J, Zhou Y, Cheng M, Long C. Peer-to-peer energy trading in a microgrid. Appl Energy 2018;220:1–12. http://dx.doi.org/10.1016/j.apenergy.2018.03.010, URL: https://www.sciencedirect.com/science/article/pii/S0306261918303398.

[8] Chalkiadakis G, Elkind E, Wooldridge M. Computational aspects of cooperative game theory. Morgan Claypool; 2011.

[9] Shapley LS. A value for n-person games. In: Contributions to the theory of games. Vol. 2. (28):1953, p. 307–17.

[10] Norbu S, Couraud B, Robu V, Andoni M, Flynn D. Modelling the redistribution of benefits from joint investments in community energy projects. Appl Energy 2021;287:116575. http://dx.doi.org/10.1016/j.apenergy.2021.116575.

[11] O'Brien G, El Gamal A, Rajagopal R. Shapley value estimation for compensation of participants in demand response programs. IEEE Trans Smart Grid 2015;6(6):2837–44. http://dx.doi.org/10.1109/TSG.2015.2402194.

[12] UKERC Energy Data Centre. New thames valley vision - end point monitors. In: UKERC EDC: data catalogue. 2017, [Online], https://ukerc.rl.ac.uk/DC/cgi-bin/edc_search.pl?WantComp=147. (Accessed 24 October 2022).

[13] UK Power Networks Innovation. Low carbon London. 2021, [Online] https://innovation.ukpowernetworks.co.uk/projects/low-carbon-london/. (Accessed 24 October 2022).

[14] Chiş A, Koivunen V. Coalitional game-based cost optimization of energy portfolio in smart grid communities. IEEE Trans Smart Grid 2019;10(2):1960–70. http://dx.doi.org/10.1109/TSG.2017.2784902.

[15] Safdarian A, Divshali PH, Baranauskas M, Keski-Koukkari A, Kulmala A. Coalitional game theory based value sharing in energy communities. IEEE Access 2021;9:78266–75. http://dx.doi.org/10.1109/ACCESS.2021.3081871.

[16] Vespermann N, Hamacher T, Kazempour J. Access economy for storage in energy communities. IEEE Trans Power Syst 2021;36(3):2234–50. http://dx.doi.org/10.1109/TPWRS.2020.3033999.

[17] Robu V, Vinyals M, Rogers A, Jennings NR. Efficient buyer groups with prediction-of-use electricity tariffs. IEEE Trans Smart Grid 2018;9(5):4468–79. http://dx.doi.org/10.1109/TSG.2017.2660580.

[18] Vinyals M. Scalable multi-agent local energy trading — Meeting regulatory compliance and validation in the Cardiff grid. Appl Energy 2021;298:117231. http://dx.doi.org/10.1016/j.apenergy.2021.117231, URL: https://www.sciencedirect.com/science/article/pii/S0306261921006528.

[19] Long C, Zhou Y, Wu J. A game theoretic approach for peer to peer energy trading. Energy Procedia 2019;159:454–9. http://dx.doi.org/10.1016/j.egypro.2018.12.075, Renewable Energy Integration with Mini/Microgrid. URL: https://www.sciencedirect.com/science/article/pii/S1876610218313705.

[20] Hupez M, Toubeau J-F, De Grève Z, Vallée F. A new cooperative framework for a fair and cost-optimal allocation of resources within a low voltage electricity community. IEEE Trans Smart Grid 2021;12(3):2201–11. http://dx.doi.org/10.1109/TSG.2020.3040086.

[21] Singh A, Sethi BK, Singh D, Misra RK. Shapley value method and stochastic Dantzig–Wolfe decomposition for decentralized scheduling of multimicrogrid. IEEE Syst J 2021;1–13. http://dx.doi.org/10.1109/JSYST.2021.3119426.

[22] Zhang Z, Jiang Y, Lin Z, Wen F, Ding Y, Yang L, et al. Optimal alliance strategies among retailers under energy deviation settlement mechanism in China's forward electricity market. IEEE Trans Power Syst 2020;35(3):2059–71. http://dx.doi.org/10.1109/TPWRS.2019.2955479.

[23] Norbu S, Couraud B, Andoni M, Flynn D, Robu V. Enabling smart control and fair sharing of renewable resources in energy communities. ERCIM News 2021;2021(126):2.

[24] Norbu S, Couraud B, Robu V, Andoni M, Flynn D. Modeling economic sharing of joint assets in community energy projects under LV network constraints. IEEE Access 2021;9:112019–42. http://dx.doi.org/10.1109/ACCESS.2021.3103480.

[25] Han L, Morstyn T, McCulloch M. Estimation of the shapley value of a peer-to-peer energy sharing game using coalitional stratified random sampling. 2019, arXiv preprint arXiv:1903.11047.

[26] Kulmala A, Baranauskas M, Safdarian A, Valta J, Järventausta P, Björkqvist T. Comparing value sharing methods for different types of energy communities. In: 2021 IEEE PES innovative smart grid technologies Europe. 2021, p. 1–6. http://dx.doi.org/10.1109/ISGTEurope52324.2021.9640205.

[27] Baranauskas M, Keski-Koukkari A, Divshali PH, Safdarian A, Kulmala A. Value creation and sharing methods in household energy communities. In: 2022 IEEE international conference on power electronics, smart grid, and renewable energy. 2022, p. 1–8. http://dx.doi.org/10.1109/PESGRE52268.2022.9715888.

[28] Alam M, Ramchurn SD, Rogers A. Cooperative energy exchange for the efficient use of energy and resources in remote communities. In: Proceedings of the 2013 international conference on autonomous agents and multi-agent systems. 2013, p. 731–8.

[29] Jia Y, Wen P, Yan Y, Huo L. Joint operation and transaction mode of rural multi microgrid and distribution network. IEEE Access 2021;9:14409–21. http://dx.doi.org/10.1109/ACCESS.2021.3050793.

[30] Sharma S, Abhyankar A. Loss allocation of radial distribution system using Shapley value: A sequential approach. Int J Electr Power Energy Syst 2017;88:33–41. http://dx.doi.org/10.1016/j.ijepes.2016.11.015, URL: https://www.sciencedirect.com/science/article/pii/S0142061516311310.

[31] Sharma S, Abhyankar AR. Loss allocation for weakly meshed distribution system using analytical formulation of Shapley value. IEEE Trans Power Syst 2017;32(2):1369–77. http://dx.doi.org/10.1109/TPWRS.2016.2571980.

[32] Amaris H, Molina YP, Alonso M, Luyo JE. Loss allocation in distribution networks based on Aumann–Shapley. IEEE Trans Power Syst 2018;33(6):6655–66. http://dx.doi.org/10.1109/TPWRS.2018.2844740.

[33] Pourahmadi F, Dehghanian P. A game-theoretic loss allocation approach in power distribution systems with high penetration of distributed generations. Mathematics 2018;6(9). http://dx.doi.org/10.3390/math6090158, URL: https://www.mdpi.com/2227-7390/6/9/158.

[34] Azad-Farsani E, Agah S, Askarian-Abyaneh H, Abedi M, Hosseinian S. Stochastic LMP (locational marginal price) calculation method in distribution systems to minimize loss and emission based on Shapley value and two-point estimate method. Energy 2016;107:396–408. http://dx.doi.org/10.1016/j.energy.2016.04.036, URL: https://www.sciencedirect.com/science/article/pii/S036054421630442X.

[35] Yu Q, Xie J, Chen X, Yu K, Gan L. Loss and emission reduction allocation in distribution networks using MCRS method and Aumann-Shapley value method. IET Gener Transm Distrib 2018;12:5975–81.

[36] Vicente-Pastor A, Nieto-Martin J, Bunn DW, Laur A. Evaluation of flexibility markets for retailer–DSO–TSO coordination. IEEE Trans Power Syst 2019;34(3):2003–12. http://dx.doi.org/10.1109/TPWRS.2018.2880123.

[37] Azuatalam D, Chapman AC, Verbič G. Shapley value analysis of distribution network cost-causality pricing. In: 2019 IEEE Milan powertech. 2019, p. 1–6. http://dx.doi.org/10.1109/PTC.2019.8810968.

[38] Maleki S, Rahwan T, Ghosh S, Malibari A, Alghazzawi D, Rogers A, et al. The Shapley value for a fair division of group discounts for coordinating cooling loads. PLoS One 2020;15(1):e0227049.

[39] Singh H, Hao S, Papalexopoulos A. Transmission congestion management in competitive electricity markets. IEEE Trans Power Syst 1998;13(2):672–80. http://dx.doi.org/10.1109/59.667399.

[40] Xiao H, Li W. Allocation of congestion cost in a pool based model using Shapley value. In: 2009 Asia-pacific power and energy engineering conference. 2009, p. 1–5. http://dx.doi.org/10.1109/APPEEC.2009.4918127.

[41] Voswinkel S, Höckner J, Khalid A, Weber C. Sharing congestion management costs among system operators using the Shapley value. Appl Energy 2022;317:119039. http://dx.doi.org/10.1016/j.apenergy.2022.119039, URL: https://www.sciencedirect.com/science/article/pii/S030626192200441X.

[42] Cheng Y, Fan S, Ni J, Sun S, Wang R, Li G, et al. An innovative profit allocation to distributed energy resources integrated into virtual power plant. In: International conference on renewable power generation. 2015, p. 1–6. http://dx.doi.org/10.1049/cp.2015.0547.

[43] Wang Y, Gao W, Qian F, Li Y. Evaluation of economic benefits of virtual power plant between demand and plant sides based on cooperative game theory. Energy Convers Manage 2021;238:114180. http://dx.doi.org/10.1016/j.enconman.2021.114180, URL: https://www.sciencedirect.com/science/article/pii/S0196890421003563.

[44] Dabbagh SR, Sheikh-El-Eslami MK. Risk-based profit allocation to DERs integrated with a virtual power plant using cooperative game theory. Electr Power Syst Res 2015;121:368–78. http://dx.doi.org/10.1016/j.epsr.2014.11.025, URL: https://www.sciencedirect.com/science/article/pii/S0378779614004325.

[45] Fang F, Yu S, Liu M. An improved Shapley value-based profit allocation method for CHP-VPP. Energy 2020;213:118805. http://dx.doi.org/10.1016/j.energy.2020.118805, URL: https://www.sciencedirect.com/science/article/pii/S0360544220319125.

[46] Chattopadhyay D. An energy brokerage system with emission trading and allocation of cost savings. IEEE Trans Power Syst 1995;10(4):1939–45. http://dx.doi.org/10.1109/59.476061.

[47] Liao Z, Zhu X, Shi J. Case study on initial allocation of Shanghai carbon emission trading based on Shapley value. J Clean Prod 2015;103:338–44. http://dx.doi.org/10.1016/j.jclepro.2014.06.045, Carbon Emissions Reduction: Policies, Technologies, Monitoring, Assessment and Modeling. URL: https://www.sciencedirect.com/science/article/pii/S0959652614006337.

[48] Zhou Q, Shahidehpour M, Sun T, Feng D, Yan M. Cooperative game for carbon obligation allocation among distribution system operators to incentivize the proliferation of renewable energy. IEEE Trans Smart Grid 2019;10(6):6355–65. http://dx.doi.org/10.1109/TSG.2019.2903686.

[49] Zhang Y-J, Wang A-D, Da Y-B. Regional allocation of carbon emission quotas in China: Evidence from the Shapley value method. Energy Policy 2014;74:454–64. http://dx.doi.org/10.1016/j.enpol.2014.08.006, URL: https://www.sciencedirect.com/science/article/pii/S0301421514004595.

[50] Mays J. Cost allocation and net load variability. IEEE Trans Power Syst 2018;33(2):2030–9. http://dx.doi.org/10.1109/TPWRS.2017.2732921.

[51] Zhang R, Jiang T, Li F, Li G, Chen H, Li X. Coordinated bidding strategy of wind farms and power-to-gas facilities using a cooperative game approach. IEEE Trans Sustain Energy 2020;11(4):2545–55. http://dx.doi.org/10.1109/TSTE.2020.2965521.

[52] Li Y, Liu W, Shahidehpour M, Wen F, Wang K, Huang Y. Optimal operation strategy for integrated natural gas generating unit and power-to-gas conversion facilities. IEEE Trans Sustain Energy 2018;9(4):1870–9. http://dx.doi.org/10.1109/TSTE.2018.2818133.

[53] Churkin A, Sauma E, Pozo D, Bialek J, Korgin N. Enhancing the stability of coalitions in cross-border transmission expansion planning. IEEE Trans Power Syst 2022;37(4):2744–57. http://dx.doi.org/10.1109/TPWRS.2021.3124988.

[54] Prettico G, Gangale F, Mengolini A, Lucas A, Fulli G. Distribution system operators observatory - European electricity distribution systems to representative distribution networks. 2016, http://dx.doi.org/10.2790/701791.

[55] Mann I, Shapley LS. Values of large games, IV: evaluating the electoral college by Montecarlo techniques. Santa Monica, CA: RAND Corporation; 1960.

[56] Castro J, Gómez D, Tejada J. Polynomial calculation of the Shapley value based on sampling. Comput Oper Res 2009;36(5):1726–30.

[57] Castro J, Gómez D, Molina E, Tejada J. Improving polynomial estimation of the Shapley value by stratified random sampling with optimum allocation. Comput Oper Res 2017;82:180–8.

[58] Liben-Nowell D, Sharp A, Wexler T, Woods K. Computing Shapley value in supermodular coalitional games. In: Gudmundsson J, Mestre J, Viglas T, editors. Computing and combinatorics. Berlin, Heidelberg: Springer Berlin Heidelberg; 2012, p. 568–79.

[59] Maleki S, Tran-Thanh L, Hines G, Rahwan T, Rogers A. Bounding the estimation error of sampling-based Shapley value approximation. 2013, arXiv preprint arXiv:1306.4265.

[60] Aziz H, de Keijzer B. Shapley meets shapley. 2013, arXiv preprint arXiv:1307.0332.

[61] Bachrach Y, Markakis E, Resnick E, Procaccia A, Rosenschein J, Saberi A. Approximating power indices: Theoretical and empirical analysis. Auton Agents Multi-Agent Syst 2010;20:105–22. http://dx.doi.org/10.1007/s10458-009-9078-9.

[62] Burgess M, Chapman A. Approximating the Shapley value using stratified empirical Bernstein sampling. In: Proceedings of the 30th international joint conference on artificial intelligence. 2021, p. 73–81. http://dx.doi.org/10.24963/ijcai.2021/11.

[63] Bhagat S, Kim A, Muthukrishnan S, Weinsberg U. The shapley value in knapsack budgeted games. In: International conference on web and internet economics. Springer; 2014, p. 106–19.

[64] Cremers S, Robu V, Hofman D, Naber T, Zheng K, Norbu S. Efficient methods for approximating the Shapley value for asset sharing in energy communities. In: Proceedings of the thirteenth ACM international conference on future energy systems. e-energy '22, New York, NY, USA: Association for Computing Machinery; 2022, p. 320–4. http://dx.doi.org/10.1145/3538637.3538861.

[65] Enercon. ENERCON wind energy converters product overview. 2010, Enercon. Enercon GmbH. URL: https://docs.wind-watch.org/Enercon.pdf.

[66] van den Brink R. An axiomatization of the Shapley value using a fairness property. Internat J Game Theory 2002;30(3):309–19.

[67] Janardan KG, Patil GP. A unified approach for a class of multivariate hypergeometric models. Sankhyā: Indian J Stat Ser A (1961-2002) 1972;34(4):363–76, URL: http://www.jstor.org/stable/25049821.

[68] Met Office. Met office MIDAS open: UK land surface stations data (1853-current). Centre for Environmental Data Analysis; 2019, [Online] https://catalogue.ceda.ac.uk/uuid/dbd451271eb04662beade68da43546e1. (Accessed 31 May 2022).

[69] Inc. A. Dark sky API. 2022, URL: https://darksky.net/dev.

[70] Jeong HC, Jang M, Kim T, Joo S-K. Clustering of load profiles of residential customers using extreme points and demographic characteristics. Electronics 2021;10(3). http://dx.doi.org/10.3390/electronics10030290, URL: https://www.mdpi.com/2079-9292/10/3/290.

[71] Kwac J, Flora J, Rajagopal R. Household energy consumption segmentation using hourly data. IEEE Trans Smart Grid 2014;5(1):420–30. http://dx.doi.org/10.1109/TSG.2013.2278477.

[72] Zhou K, Yang S, Shen C. A review of electric load classification in smart grid environment. Renew Sustain Energy Rev 2013;24:103–10. http://dx.doi.org/10.1016/j.rser.2013.03.023, URL: https://www.sciencedirect.com/science/article/pii/S1364032113001822.

[73] Villa-Arrieta M, Sumper A. Economic evaluation of nearly zero energy cities. Appl Energy 2019;237:404–16. http://dx.doi.org/10.1016/j.apenergy.2018.12.082, URL: https://www.sciencedirect.com/science/article/pii/S0306261918319007.

[74] Hua W, Chen Y, Qadrdan M, Jiang J, Sun H, Wu J. Applications of blockchain and artificial intelligence technologies for enabling prosumers in smart grids: A review. Renew Sustain Energy Rev 2022;161:112308. http://dx.doi.org/10.1016/j.rser.2022.112308, URL: https://www.sciencedirect.com/science/article/pii/S1364032122002222.

[75] Kirli D, Couraud B, Robu V, Salgado-Bravo M, Norbu S, Andoni M, et al. Smart contracts in energy systems: A systematic review of fundamental approaches and implementations. Renew Sustain Energy Rev 2022;158:112013. http://dx.doi.org/10.1016/j.rser.2021.112013, URL: https://www.sciencedirect.com/science/article/pii/S1364032121012764.

[76] Andoni M, Robu V, Flynn D, Abram S, Geach D, Jenkins D, et al. Blockchain technology in the energy sector: A systematic review of challenges and opportunities. Renew Sustain Energy Rev 2019;100:143–74. http://dx.doi.org/10.1016/j.rser.2018.10.014, URL: https://www.sciencedirect.com/science/article/pii/S1364032118307184.

[77] Schulz AS, Uhan NA. Approximating the least core value and least core of cooperative games with supermodular costs. Discrete Optim 2013;10(2):163–80. http://dx.doi.org/10.1016/j.disopt.2013.02.002.

[78] Downing S, Socie D. Simple rainflow counting algorithms. Int J Fatigue 1982;4(1):31–40. http://dx.doi.org/10.1016/0142-1123(82)90018-4, URL: https://www.sciencedirect.com/science/article/pii/0142112382900184.

[79] Xu B, Oudalov A, Ulbig A, Andersson G, Kirschen DS. Modeling of lithium-ion battery degradation for cell life assessment. IEEE Trans Smart Grid 2018;9(2):1131–40. http://dx.doi.org/10.1109/TSG.2016.2578950.