



HAL
open science

Efficient distributed solutions for sharing energy resources at local level: a cooperative game approach

Diego Kiedanski, Ana Bušić, Daniel Kofman, Ariel Orda

► To cite this version:

Diego Kiedanski, Ana Bušić, Daniel Kofman, Ariel Orda. Efficient distributed solutions for sharing energy resources at local level: a cooperative game approach. IEEE Conference on Decision and Control, Dec 2020, Jeju Island, South Korea. hal-02520203v1

HAL Id: hal-02520203

<https://hal.science/hal-02520203v1>

Submitted on 26 Mar 2020 (v1), last revised 18 Sep 2020 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient distributed solutions for sharing energy resources at local level: a cooperative game approach.

Diego Kiedanski, Ana Bušić, Daniel Kofman and Ariel Orda

Abstract—Local energy generation as well as local energy storage represent key opportunities for energy transition. Nevertheless, their massive deployment is being delayed mainly due to cost reasons. Sharing resources at the local level enables not only reducing these costs significantly, but also to further optimize the cost of the energy exchanged with providers external to the local community. A key question that arises while sharing resources is how to distribute the obtained benefits among the various local players that cooperate. In this paper we propose a cooperative game model, where the players are the holders of energy resources (generation and storage); they cooperate in order to reduce their individual electricity costs. We prove that the core of the game is non-empty; i.e., the proposed cooperative game has a stable solution (distribution of the payoffs among the players) for the case where all players participate in a unique community, and no strict subset of players can obtain a better gain by leaving the community.

We propose a formulation of this game, based on the theory of linear production games, which lead us to the two main contributions of this paper. First, we propose an efficient (with linear complexity) centralized algorithm for finding a stable payoff. Second, we provide an efficient distributed algorithm that computes an allocation in the core of the game without any requirement for the players to share any private information. The distributed algorithm requires the exchange of intermediate solutions among players. The topology of the network that enables these exchanges is closely related to the performance of the distributed algorithm. We show, by way of simulations, which are the best topologies for these communication graphs.

I. INTRODUCTION

The deep decarbonization of the electricity grid is significantly facilitated by the massive local deployment of distributed renewable resources, most of which are intermittent by nature (e.g. solar and wind). With such technologies, one of the biggest challenges becomes matching the energy demand with the intermittent generation, which lead to a large set of proposals for leveraging grid flexibility opportunities [1], i.e., the ability to adapt the energy demand in real time, usually with the constrain of avoiding significantly impacting the quality of experience.

Recently, it has been shown that the ability to locally match surplus of energy generation with consumption (namely, same low voltage grid), could further reduce energy losses and increase the quality of power supplied [2], [3].

The VALADOE Chair of IMT Atlantique in partnership with Télécom ParisTech and Mines Saint Etienne and supported by ENEDIS, Région Pays de la Loire, Nantes Metropole

Diego Kiedanski and Daniel Kofman with the INFRES Department, Télécom Paris, France.

Ana Bušić is with Inria and DI ENS, École Normale Supérieure, CNRS, PSL Research University, Paris, France.

Ariel Orda is with Technion, Haifa, Israel.

With the above goal in mind, and following the decentralization of the wholesale electricity market, local energy markets and local trading platforms were proposed and implemented [2], [4], [5]. Such platforms usually consist of a series of sequential trading opportunities, which result in very complex interactions among households (each being a “game” in the game-theoretic sense), which might not provide the benefits for which they were designed [6] (the authors of the present paper have provided clear evidence of this fact in a separate paper currently under review). These drawbacks, together with a better understanding that consumers might be prone to cooperate [7], [8], [9], prompted research of cooperative frameworks [10], [11]. There are already some simple implementations of such schemes, such as the “Autoconsommation collective” in France¹.

In this paper, we study the existence of stable cooperation agreements among households with renewable energy generation resources and, optionally, storage capacity. A coalition is considered to be stable if the distribution of the gains is satisfactory for all participants; which means that no strict subset of participants will benefit from leaving the coalition. Indeed, we consider that the main objective of forming coalitions is to optimize the cost of the energy that has to be exchanged with the traditional electricity company (usually, at any given time, these companies sell energy at a higher price than they buy locally generated energy). Our contributions are the following:

- 1) We prove that the centralized solution to the cooperative game introduced in [12] can be found efficiently using techniques from the theory of Linear Programming Games [13], instead of being NP-Complete as previously believed.
- 2) We provide a distributed algorithm to compute the aforementioned solution, in which each participant of the game can retain its private information. Previous proposed approaches require players to submit private information to a centralized solver.
- 3) In our distributed algorithm, a communication network enables players to exchange their updated solutions. We evaluate numerically several families of such network topologies and conclude that faster convergence of our algorithms is obtained with topologies based on expander graphs.

The rest of the paper is organized as follows. Section II introduces some required concepts and notation. In Section III, we position our paper within the literature. Our math-

¹<https://www.enedis.fr/lautoconsommation>

emational model is presented in Section IV. The distributed algorithm used to solve the problem is introduced in Section V. Numerical results are presented and discussed in Section VI. Final remarks are presented in Section VII.

II. PRELIMINARIES

A. Cooperative game theory

A coalitional game $G = (\mathcal{N}, v)$ is defined by a set of players $\mathcal{N} = \{1, \dots, N\}$ and a characteristic value function $v: 2^{\mathcal{N}} \rightarrow \mathbb{R}$ that specifies the value obtained by each subset $S \subseteq \mathcal{N}$ of players (i.e., each *coalition*) if they cooperate (we denote $v_S = v(S)$). The set of all players is known as the *Grand Coalition*. The solution of a cooperative game consists of a payoff vector $y \in \mathbb{R}^N$ in which each coordinate y_i specifies the value allocated to player i .

A payoff vector y is said to be an *imputation* for the *Grand Coalition* if $\sum_{i \in \mathcal{N}} y_i = v_{\mathcal{N}}$ (efficiency) and $y_i \geq v_{\{i\}}$ (individual rationality). The set of imputations are precisely the ways of splitting the value obtained by forming the *Grand Coalition*, while still guaranteeing that each player does at least as well as being alone. Furthermore, if y is an imputation and it satisfies $\sum_{i \in S} y_i \geq v_S, \forall S \subset \mathcal{N}$, then it is said to belong to the *core*.

The core is a standard solution concept in cooperative game theory. The core can be empty, hinting the lack of a stable solution of the game. Deciding whether a game G has an empty core is, in general, an NP-Complete problem [14].

Definition 1: A function $\gamma: 2^{\mathcal{N}} \setminus \emptyset \rightarrow [0, 1]$ satisfying Equation (1) is said to be *balanced*.

$$\sum_{S \subset \mathcal{N}, i \in S} \gamma_S = 1, \forall i \in \mathcal{N} \quad (1)$$

Definition 2: A game $G = (\mathcal{N}, v)$ is said to be *balanced* if for any balanced function γ it holds that

$$\sum_{S \subset 2^{\mathcal{N}} \setminus \emptyset} v_S \gamma_S \leq v_{\mathcal{N}} \quad (2)$$

Furthermore, a game is said to be *totally balanced* if each of its sub-games $G_S = (S \subset \mathcal{N}, v|_S)$ is balanced.

A fundamental theorem in the characterization of the core is due to Bondareva [15], also known as Bondareva-Shapely (Theorem 1).

Theorem 1: The core of a game $G = (\mathcal{N}, v)$ is non-empty if and only if the game G is balanced.

A particularly interesting family of cooperative games that we use extensively in this paper is that of linear programming and linear production games [16], [13].

Definition 3: Let \mathcal{N} be the set of N players, $A \in M(\mathbb{R})_{M \times N}$ a constraint matrix, $\vec{b}_{\{i\}} \in \mathbb{R}^M$ a vector of resources controlled by player i , $B \in M(\mathbb{R})_{M \times N}$ a matrix whose j -th column is the vector $\vec{b}_{\{j\}}$, $c \in \mathbb{R}^N$ a cost vector common to all players and let t^S be a binary vector whose j -th coordinate is 1 if $j \in S$.

A *linear programming game (LPG)* is a cooperative game $G = (\mathcal{N}, v)$ for which

$$v(S) = \max\{cx : Ax \leq Bt^S\} \quad (3)$$

Two important results regarding LPGs are summarized by the following theorem, due to Owen.

Theorem 2: Linear Programming Games have non-empty cores [16]. Furthermore, the set defined as:

$$DS = \{y \in \mathbb{R}^N : y = uB, \text{ for some dual optimal vector } u\} \quad (4)$$

is a subset of the core.

It can be shown [17] that the family of totally balanced games coincides with the family of linear programming games.

B. Graph Theoretic Concepts

A graph $G = (V_G, E_G)$ is defined by a set of nodes V_G and a set of edges E_G . In this paper we will assume that graphs are simple, weighted, undirected and have no loops. The adjacency matrix of G is denoted A_G and its entries a_{vw} are positive if and only if there is an edge between v and w , $v, w \in V_G$. We define the degree of a node $v \in V_G$ as $d_v = \sum_{w \in V_G} a_{vw}$. The degree matrix D_G is a diagonal matrix with the degrees of the nodes in its diagonal. The neighbourhood of a node $v \in V_G$, is defined as the set of adjacent vertices to it: $N(v) = \{w : a_{vw} > 0\}$ and the Laplacian of G by $L_G = D_G - A_G$. We will be interested in the largest eigenvalue of L_G , which we shall denote by μ_G . We denote the complete graph in N nodes K_N , the cycle graph in N nodes by C_n , the line graph by P_N , the wheel graph (namely, a cycle with a node connected to all other nodes) as W_n and an arbitrary tree by T_n .

We are particularly interested in *expander graphs* as these types of graphs have provided fast convergence for distributed algorithms in previous studies [18]. We require to additional definitions before defining *expander graphs*. Let $\partial A = \{(v, w) : (v, w) \in E_G, v \in A \subseteq V_G, w \in V_G \setminus A\}$ be the edge boundary of a set A . We define the Cheeger constant as: $h(G) = \min\{\frac{|\partial A|}{|A|} : A \subset V_G, 0 < |A| \leq \frac{1}{2}|V_G|\}$ [19]. With those definitions in place, a graph is said to be *expander* if it is sparse and its Cheeger constant is high. Intuitively, they are graphs that achieve good connectivity properties with only a few edges.

Two types of expander graphs will be used in this paper. First, a *cycle chordal graph* (denoted E_n) as defined in [20, Theorem 4.4.2], which can be obtained by adding to a cycle, the edges that connect each node (numbered from 0 to p) to its inverso moduli p (p prime). Finally, we consider randomly sampled 4-regular graphs (which we will denote $R_{4,n}$), as they are good expander graphs with high probability [21].

III. RELATED WORK

In [12], Han et al. propose the use of energy cooperatives among households with batteries and renewable energy generation with the target of minimizing the total cost paid to their Traditional Electricity Company (TEC). The authors model the energy cooperatives using cooperative game theory and prove that the considered game has a non-empty core. Their solution is based on the concept of nucleolus and on the Shapely value. Due to the complexity of solving the

cooperative game, their simulations are run up to 14 players, which took some 500 minutes to solve.

In [22], the authors study the same cooperative game as in [12] and propose a technique to estimate the Shapely value using stratified random sampling. They show that, using a high sampling rate, a game with 30 players can be solved in some 300 minutes, while a game with 50 players can be solved in the same amount of time by sacrificing some accuracy.

In [23] the authors use K-means to cluster similar consumption profiles in order to reduce the number of players needed to solve the cooperative game introduced above. They found that the nucleolus obtained using the number of clusters instead of the number of players approximated reasonably well the nucleolus of the original game. With the proposed technique, they managed to solve a game with up to 200 players (albeit approximately).

In our paper, we study the same cooperative game as in [12], [22], [23]. By using techniques based on Linear Production Games (LPG) [16], we show that the core of the game is non-empty, and we derive a method for efficiently computing a payoff in the core that can scale up to several thousand players. Our second contribution consists in providing a distributed algorithm that computes such a payoff over any type of overlay communication network. In addition to the high efficiency, our solution has the key benefit that it does not require the disclosure by the households of local information.

In addition to the articles described above, there are several studies related to power systems from a cooperative approach. A review of different models and challenges associated with energy communities can be found in [24]. In [25], the authors model energy communities that interface with the wholesale energy market via a community manager, but they do not analyze their problem through cooperative game theory nor their consumers have batteries. The authors in [26] model a cooperative energy management system as a multi-stage stochastic optimization problem, but only focus on the optimization problem and not in the game theoretic aspects of cooperation. Kim et al. [27] model the trading of energy between microgrids using a Generalized Nash Bargaining solution. Their cooperative approach does not take into account the possibility of deviation of sub-coalitions.

Finally, cooperation for the shared investment in energy storage has also been extensively considered [9], [10], [28].

IV. MODEL

Consider a set of players \mathcal{N} that aim to minimize their electricity bill at a given day. Each player has a daily demand profile $x^i = (x_1^i, \dots, x_T^i)$ that is observed at discrete time-slots $t \in \mathcal{T} = \{1, \dots, T\}$. The demand of a player at time-slot t , denoted by x_t^i , will be positive if the energy consumed by her appliances (not counting the battery) is greater than the energy generated by her renewable resources (if any) and negative otherwise. The demand profiles are fixed and

known at the beginning of the day (a forecast might be known instead).

Each player owns a battery (possibly of size 0) that they can use to change their net load as seen from the grid. Let \mathbf{S}^i denote the size of player's i battery, \mathbf{S}_s^i the initial state of charge and $\bar{\delta}^i, \underline{\delta}^i$ the maximum energy that the battery can charge or discharge at any given time-slot, respectively. Furthermore, the battery is subject to charging and discharging efficiencies represented by the coefficients $\eta_c^i, \eta_d^i \in (0, 1]$, respectively.

The only decision that player i can take is whether to charge or discharge the battery. We denote by c_t^i how much player i charges the battery at time-slot t and how much she discharges it by d_t^i . The feasible set of charging and discharging actions is described in Equation (5).

$$\mathcal{F}_i = \{c^i, d^i : 0 \leq \mathbf{S}_s^i + \sum_{t=1}^j c_t^i - d_t^i \leq \mathbf{S}^i \forall j \in \mathcal{T}, \quad (5)$$

$$c_t^i \in [0, \bar{\delta}^i], d_t^i \in [0, \underline{\delta}^i]\}$$

The net load profile of player i (i.e., her demand profile plus the effect of the battery), as seen from the grid at time-slot t , is given by: $z_t^i = \frac{c_t^i}{\eta_c^i} - d_t^i \eta_d^i + x_t^i$. All players are charged by their net load according to the same buying and selling tariff. We will denote by p_t^b the price at which energy is bought during time-slot t and by p_t^s the price at which surplus of energy could be sold to the grid. In this paper we will assume that $p_t^b > p_t^s, \forall t \in \mathcal{T}$. Finally, the cost paid by player i , in the absence of collaboration, is given by Equation (6).

$$\sum_{t=1}^T [\max\{z_t^i, 0\}p_t^b - \max\{-z_t^i, 0\}p_t^s] \quad (6)$$

A. Cooperative game model

We introduce now the model for the case when players collaborate. We model the system by a cooperative game formed by their ensemble. The main idea behind the cooperative game is that players in a coalition $S \subseteq \mathcal{N}$ will only be charged by their aggregated net load. That is, if players manage to synchronize the consumption in the coalition with their surplus, they will be able to reduce their costs.

In order to define a cooperative game, it is required to define the value of each coalition S . As explained above, in our case the value v_S of a coalition S is (the minus of) the minimal electricity cost that players in S can achieve by cooperating.

Following [29], the value v_S of a coalition S is given by the optimal solution of the linear maximization problem (7).

$$v_S = \max_{c^i, d^i: i \in S} - \left[\sum_{t=1}^T \left(z_t^{S,+} p_t^b - z_t^{S,-} p_t^s \right) \right] \quad (7)$$

subject to:

$$c^i, d^i \in \mathcal{F}_i \quad \forall i \in S$$

$$z_t^{S,+} - z_t^{S,-} = \sum_{i \in S} \left(\frac{c_t^i}{\eta_c^i} - d_t^i \eta_d^i + x_t^i \right) \quad \forall t \in \mathcal{T}$$

$$c^i, d^i \geq 0 \quad \forall i \in S$$

where $z_t^{S,+} = \max\{z_t^S, 0\}$, $z_t^{S,-} = \max\{-z_t^S, 0\}$, and z_t^S represents the net energy consumption of the coalition S at time-slot t . Although $z^{S,+}$, $z^{S,-}$ are deterministic functions of the charging and discharging profiles, we define them as variables for our formal treatment of the problem. We will prove that the optimal solutions of (7) are *well behaved*, i.e., at every time-slot c_t^i and d_t^i cannot both be strictly positive. The same holds for $z^{S,+}$, $z^{S,-}$.

Proposition 1: In all of the optimal solutions of (7) it holds that $z_t^{S,+} z_t^{S,-} = 0$ and $c_t^i d_t^i = 0$ for all $t \in \mathcal{T}$ and $i \in \mathcal{N}$.

Proof: By way of contradiction. We begin by looking at $z^{S,+}$, $z^{S,-}$. Suppose that there is a time-slot in which $z_t^{S,+} - z_t^{S,-} = K$ and $z_t^{S,+} z_t^{S,-} \neq 0$. Then, if $K \geq 0$, The corresponding term in the objective function is $c_t = -p_t^b(K + z_t^{S,-}) + p_t^s z_t^{S,-} = -p_t^b K + z_t^{S,-}(p_t^s - p_t^b)$ and it is maximized when $z_t^{S,-} = 0$. This is analogous for $K < 0$. We proceed to look at the charging and discharging variables.

Let us assume that, in an optimal solution, there is one player i such that $c_t^i d_t^i \neq 0$ at time-slot t and $J = c_t^i - d_t^i$, where J is the desired usage of the battery. Fix the variables of all other players in that time-slot together with the load and denote the sum by L . We will denote $X(d_t^i) = L + J \frac{1}{\eta_c^i} + d_t^i \left(\frac{1}{\eta_c^i} - \eta_d^i \right)$. Note that $X(d_t^i)$ is an increasing function in d_t^i because $\frac{1}{\eta_c^i} - \eta_d^i$ is always positive.

The cost at time-slot t is given by $c_t = -p_t^b X(d_t^i)$ if $X(d_t^i) \geq 0$ or $c_t = |p_t^s X(d_t^i)|$ if $X(d_t^i) < 0$. In both cases, c_t is maximized by setting d_t^i as small as possible, hence $d_t^i = 0$. The result is analogous for c_t^i . This concludes the proof. \blacksquare

Observation 1: The coalitional game whose characteristic value function is given by the value of (7) is not an LPG. This can be seen from the fact that the cost function is piecewise linear and the fact that the variables $z^{S,+}$, $z^{S,-}$ are not associated with a single player, as required by Definition 3.

In spite of Observation 1, we shall be able to maintain the results obtained for LPGs (namely, Theorem 2) using the same techniques as in [16]. First, we need to formulate the optimization problem defined in (7) as in Equation (3).

We shall use the notation $\vec{V} = (\alpha|a)(\beta|b) \dots (\zeta|z)$ to represent the vector $\vec{V} = (\underbrace{\alpha, \dots, \alpha}_a, \underbrace{\beta, \dots, \beta}_b, \dots, \underbrace{\zeta, \dots, \zeta}_z)$.

With each player i , we will associate the vector \vec{b}_i defined as follows:

$$\vec{b}_i = (0|i4T - 4T) \overbrace{(\mathbf{S}^i - \mathbf{S}_s^i|T)}^I \overbrace{(\mathbf{S}_s^i|T)}^{II} \overbrace{(\bar{\delta}^i|T)}^{III} \overbrace{(\underline{\delta}^i|T)}^{IV} \dots \dots (0|(N-i)4T) \underbrace{(x_1^i|1) \dots (x_T^i|1)}_V \underbrace{(-x_1^i|1) \dots (-x_T^i|1)}_{VI} \dots \quad (8)$$

The vectors \vec{b}_i shall be used as the right-hand side constraints of linear programs. Therefore, with each entry, there will be an associated dual variable. The roman numbers indicated in Equation (8) will be used to cross-reference those dual variables with the corresponding entry.

Let B be the matrix whose columns are the vectors \vec{b}_i , $\vec{b}_S = Bt^S$, \vec{x} a vector of variables ordered as in Equation (9) and \vec{c} the cost vector defined in Equation (10).

$$\vec{x} = (z_1^{S,+}, \dots, z_T^{S,+}, z_1^{S,-}, \dots, z_T^{S,-}, c_1^1, \dots, c_T^1, d_1^1, \dots, d_T^1, \dots, c_1^N, \dots, d_T^N) \quad (9)$$

$$\vec{c} = (-p_1^b, \dots, -p_T^b, p_1^s, \dots, p_T^s, 0, \dots, 0, \dots, 0) \quad (10)$$

Finally, let $A \in M(\mathbb{R})_{4TN+2T \times 2TN+2T}$ be the matrix obtained by writing all the constraints associated with the optimization problem (7) in the order associated with vectors \vec{b} and \vec{x} .

Proposition 2: The optimization problem defined in (7) is equivalent to $v_S = \min\{\vec{c}\vec{x}: A\vec{x} \leq Bt^S, \vec{x} \geq 0\}$.

The main advantage of the new formulation is that the matrix A and the cost \vec{c} do not depend on the coalition. Furthermore, the right hand sides of the formulation are additive.

With the formulation introduced above, the dual D_S associated with the linear programming formulation of the optimization problem (7) is given by

$$D_S) \min\{\vec{\lambda}_S Bt^S: A^T \vec{\lambda}_S \geq \vec{c}, \vec{\lambda}_S \geq 0\} \quad (11)$$

We are ready to prove the main results of the paper.

Theorem 3: The cooperative game $G = (\mathcal{N}, v)$ where v_S is obtained by solving optimization problem (7) has a non-empty core.

Proof: See Appendix . \blacksquare

Theorem 4: The payoff vector $y = (u_1, \dots, u_N)$ is in the core, where u_i is defined by Equation (12).

$$u_i = \sum_{t=1}^T [\vec{\lambda}_N^{I,i,t} \mathbf{S}^i + \vec{\lambda}_N^{III,i,t} \bar{\delta}^i + \vec{\lambda}_N^{IV,i,t} \underline{\delta}^i + \vec{\lambda}_N^{V,i,t} x_t^i - \vec{\lambda}_N^{VI,i,t} x_t^i] = \langle \vec{\lambda}_N, \vec{b}_{\{i\}} \rangle \quad (12)$$

where $\vec{\lambda}_S^{X,i,t}$ is the optimal dual variable of the problem of coalition S associated with the constraint of player i at time-slot t , labelled with the roman number X in the Equation (8).

Proof:

The main idea behind the proof is to observe that the feasible set of the dual of (7) does not depend on the coalition, hence the optimal solution of the problem associated with the *Grand Coalition* is also feasible in the problem associated with all the other coalitions S .

Let $\vec{\lambda}_S$ be the optimal solution of the dual D_S as defined in Equation (11). Because of duality in LPs we know that $\vec{\lambda}_S \vec{b}_S = v_S$, for all coalitions S .

Furthermore, we know that $\vec{\lambda}_S$ is feasible in the dual associated with all coalitions.

We proceed to consider the sum of the pay-offs defined in Equation (12) for a given coalition S .

$$\sum_{i \in S} u_i = \sum_{i \in S} \langle \vec{\lambda}_N, \vec{b}_{\{i\}} \rangle = \langle \vec{\lambda}_N, \sum_{i \in S} \vec{b}_{\{i\}} \rangle = \langle \vec{\lambda}_N, \vec{b}_S \rangle = \vec{\lambda}_N \vec{b}_S \quad (13)$$

From Equation (13), we can conclude the proof, as follows. First, when $S = \mathcal{N}$, $\sum_{i \in S} u_i = \vec{\lambda}_N \vec{b}_N = v_N$, which implies that the payoff is efficient. Finally, we know that $\vec{\lambda}_N \vec{b}_S$ is the objective value obtained by substituting $\vec{\lambda}_N$ in the dual associated with coalition S . Because the dual is a minimization problem, $\vec{\lambda}_N \vec{b}_S$ is an upper-bound on v_S . We can thus derive that $\sum_{i \in S} u_i \geq v_S$, thus concluding the proof. ■

V. DISTRIBUTED ALGORITHM FOR THE DUAL

In the previous section we showed that the cooperative game considered in this paper has a non-empty core. Furthermore, we showed that a pay-off in the core could be built from the dual of the optimization problem associated with the value of the grand coalition v_N , D_N . In this section we will establish a distributed algorithm to compute a vector in the core by solving the dual D_N in which players do not need to transmit their private information.

We assume that the players can exchange information using a communication network represented by a graph \mathcal{B} with adjacency matrix B . Each player holds an estimate of the solution of the dual D_N and in each iteration of the algorithm, they send their estimate to all their neighbours (as defined by the network \mathcal{B}). The algorithm finishes when the players reach a consensus, i.e., their estimates of the solution coincides.

We begin by rewriting D_N in a way that exposes the information available to each player (the private information of player i is the vector \vec{b}_i). Let $A_i \in \mathbb{M}_{4TN+2T, 2T}$ denote the block matrix in A obtained by restricting A to only the columns associated with the variables c^i and d^i . Then, the dual D_N can be written as:

$$D_N) \quad \min_{\vec{\lambda}} \sum_{i \in \mathcal{N}} \vec{b}_{\{i\}} \vec{\lambda}$$

subject to:

$$\bigcap_{i \in \mathcal{N}} \Omega^i$$

$$\vec{\lambda} \geq 0$$

where

$$\Omega^i = \{ \vec{\lambda} : A_i^T \vec{\lambda} \geq 0, \vec{\lambda}^{V,t} - \vec{\lambda}^{VI,t} \in [-p_t^b, p_t^s], \quad \forall t \in T \} \quad (14)$$

Furthermore, denote by $\Pi^i : \mathbb{R}^{4TN+2T} \rightarrow \Omega^i$ the projection into the set Ω^i .

Our implementation of a distributed algorithm is based on the consensus protocol introduced in Liu et al. [30]. The following result can be derived from [30].

Theorem 5: Under the following assumptions:

- 1) the sets Ω^i are closed and convex,
- 2) the functions f^i are convex on Ω^i , differentiable and its gradient is Lipschitz continuous on Ω^i ,
- 3) the communication between players occurs through an undirected and connected graph,

the iteration:

$$\begin{cases} x_{k+1}^i = \Pi^i(x_k^i - q_{k+1}^i) \\ q_{k+1}^i = \alpha \left[\nabla f^i(x_k^i) + w_k^i + \sum_{j \in N(i)} b_{ij}(x_k^i - x_k^j) \right] \\ w_{k+1}^i = w_k^i + \sum_{j \in N(i)} b_{ij}(x_{k+1}^i - x_{k+1}^j) \end{cases} \quad (15)$$

with $\alpha < \frac{1}{\mu_B}$ converges to the optimum.

We used the notation x_k^i to represent the estimate of the solution held by player i at iteration k and b_{ij} (the (i, j) entry of the adjacency matrix B) to represent the weight of the edge between i and j in the graph \mathcal{B} .

In our setting, the sets Ω^i are closed and convex as they are defined as the finite intersection of half-spaces. The functions f^i are linear, so they are convex in Ω^i , differentiable and their gradient is Lipschitz.

Proposition 3: In each iteration, each player needs to solve a projection operation Π^i . Because the feasible set is linear, this accounts to solving a quadratic programming problem. Furthermore, observe that it is possible to implement the projection step for each player in a way that does not depend on the number of players.

Proof:

Let r_i^j be the j -th row of matrix A_i . The row $r_i^j \neq \vec{0}$ if and only if the variables c^i or d^i have positive coefficients in it (because A_i is restricted to only those variables). In the original matrix A , there are only $6T$ rows in which c^i, d^i or both appear (those associated with entries denoted with the roman numbers from I to VI in Equation (8)). Therefore, there are only $6T$ non-zero rows in A_i , which proves that the size of the problem Π^i does not depend on the number of players. ■

In Proposition 3, it was shown that the projection step in each iteration does not depend on the number of players, but only on the number of time-slots used. We conclude that the time required to reach consensus in the distributed algorithm depends on the number of players only through the required number of iterations before convergence and the size of the neighbourhood of each player in the graph \mathcal{B} , but not on the size of the problem that needs to be solved at each iteration. In spite of this, the memory requirements do scale linearly with the number of players as each vector $x_{k+1}^i \in \mathbb{R}^{4TN+2T}$.

Proposition 4: The structure of the projection step Π^i is the same for all players and time-slots.

Proof: The matrix A_i by definition is obtained by linearizing the constraints in \mathcal{F}_i , and so all the projection steps have the same constraints and objective. The difference between them is that the projection step affect different coordinates of the dual $\vec{\lambda}$. Nevertheless, this can be surmounted by selecting the appropriate coordinates to feed into the projection step and updating only those. ■

VI. NUMERICAL EXPERIMENTS

In this section we provide numerical evidence on the (efficient) performance of our proposed algorithms ²

The numerical experiments shall focus on the running time of the algorithms and not on the economic benefits of the cooperation. Such benefits have already been studied in [23], [11], [22], [12], [31].

In our experiments, we assumed that all players owned energy storage and we modeled the characteristics of those devices based on a Tesla's Powerwall 2. That is, we considered that $\mathbf{S}^i = 13.5 \text{ kWh}$, $\mathbf{S}_s^i = 0$, $\bar{\delta}^i = 5 = \underline{\delta}$ kW for all $i \in \mathcal{N}$.

In Proposition 4 it was shown that the structure of the projection steps is the same for all players and time-slots. This fact can be exploited by the projection algorithm, which can reuse the steps of previous executions to increase its speed. In particular, we use the OSQP solver [32], [33], [34] to improve the running time of the projection step of each player.

A. On the impact of the network structure

We begin our numerical study by comparing the performance of the distributed algorithm using different graph families. For the experiments in this section, we assumed $T = 10$ time-slots and we sampled the demand profiles of players from a uniform distribution $x_t^i \sim \mathcal{U}[-3, 3]$ i.i.d ³. We compare seven families of graphs: the complete graph in n nodes, the cycle in n nodes, a wheel graph, a path graph, a randomly sampled 4-regular graph ($R_{4,n}$), a random tree and a cycle chordal graph.

In Figure 1, the running times of the distributed algorithm for the different families of graphs are shown. It can be

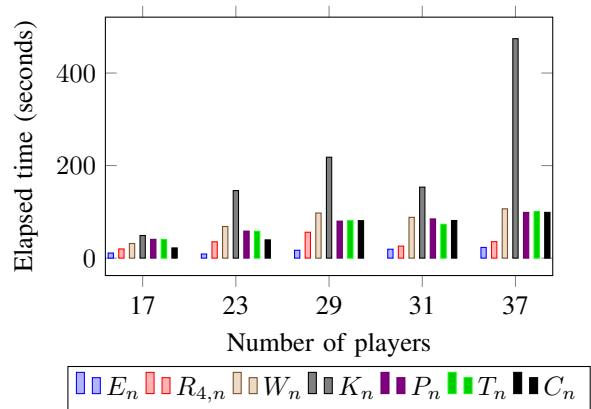


Fig. 1. Comparison of the average running time of the distributed algorithm using different topologies for an increasing number of players.

seen that the regular and chordal graphs provide the better performance, followed by the W_n , T_n , P_n , C_n . The worst performance in running time is achieved by the complete graph.

Our results regarding the performance of the different families of graphs mirror those in Duchi et al. [18]. In [18], the authors prove (for a similar but different algorithm, introduced to solve a different problem) that the fastest convergence of their distributed algorithm is achieved by expander graphs.

To study more in detail the reason behind the poor performance of complete graphs, we measured the number of iterations before convergence. Figure 2 provides the result of such an experiment. All simulations were stopped after 10000 iterations, a threshold that was consistently reached by P_n , T_n and often by C_n . On the other hand, simulations using K_n as the underlying communication graph managed to converge before the cut-off, in the neighbourhood of the 6000th iteration.

The observed performance can be explained as follows. The running time of each iteration of the algorithm is dominated by two factors: the time required to project into the sets Ω^i and the average size of the neighbourhood of each node. Observe that there are two vector additions for each player for each one of her neighbours. In path graphs or cycles, the last value does not depend on N , whereas in K_n it does.

In summary, expander graphs require the least number of iterations to converge and provide the best running times. Furthermore, complete graphs converge in less iterations than cycles and trees, but, depending on the specific implementation, might take longer overall.

B. On the impact of players on performance

In this subsection we benchmark three different approaches to solve the cooperative game discussed in this paper. The first approach consists of using the traditional definition of the core, i.e., computing the value of $2^N - 1$ coalitions and then finding a vector inside the set. The second approach consists of solving the dual of the problem

²The code used for our numerical experiments can be found in https://github.com/gus0k/cdc20_code.

³Since the load distribution does not impact the execution time, which is the metric we want to evaluate, we chose a simple one, although not realistic

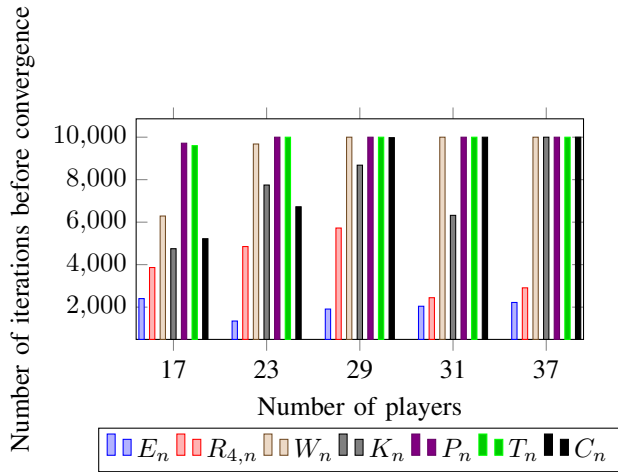


Fig. 2. Comparison of the number of iterations before convergence of the distributed algorithm using different topologies for an increasing number of players. The simulations had a 10000 cut-off.

associated with the grand coalition and building a vector in the core (in a centralized fashion), as described in Theorem (4). Finally, the third benchmark consists of running the distributed version of the algorithm, as described in the previous section. From the results in the previous subsection, we know that the performance of the distributed algorithm is better when the underlying topology in an expander graph, so that is the topology used in our experiments.

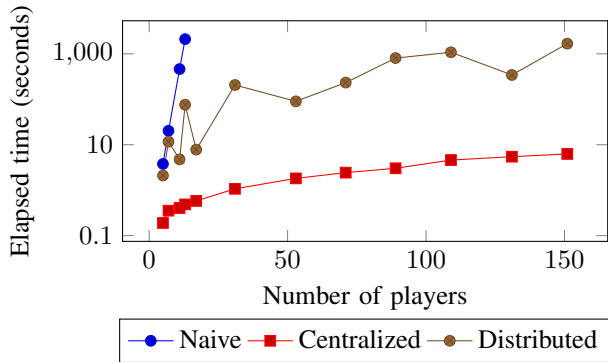


Fig. 3. Running time of the different algorithms with a varying number of players. The y-axis is in log scale.

Figure 3 presents the results obtained by running the algorithms on an increasing number of players. The *Naive* implementation is known to be exponential in the number of players as the results show, so we computed its value only up to 14 players. The centralized algorithm proposed in this paper has excellent scaling properties, and its much faster than the other alternatives. Finally, the distributed algorithm performs considerably better than the naive implementation, although it is slightly slower than its centralized counterpart. This is due to the fact that, as the number of players increases, so does the required time to reach consensus.

It should be observed that our implementation of the distributed algorithm has not been optimized, so there is

room for further improvement.

VII. CONCLUSION

In this paper we study the formation of energy communities in which households with energy resources cooperate to minimize their electricity costs. We significantly enhance the results of previous works in this domain, making the solution scalable and providing a distributed algorithm for its implementation. These results are obtained by introducing an appropriate reformulation of the game used to model the system, which enabled us to harness techniques from the theory of Linear Programming Games. On these bases, we derive an efficient algorithm - with linear complexity in terms of the number of players - to find a vector in the core, that is, a distribution of the benefits among the players that guarantees that no subset of players could do better by leaving the cooperative. Furthermore, using consensus algorithms' results, we provide a distributed implementation of the algorithm in which players do not need to reveal their private information to any third party. Finally, through numerical simulations, we show that using expander graphs as the underlying communication network provides the best running time for the distributed algorithm among several well-known families of graphs. We consider that, with these results, the proposed approach has now the required characteristics to be deployed in the field and may represent a driver for accelerating the deployment of local energy resources. In order to evaluate the real potential gains, the authors are currently working on two extensions of the proposed game. One direction aims to include the investment cost in solar panels and batteries, while the other one aims to consider additional recurring costs players must cover, including taxes on the usage of the grid.

REFERENCES

- [1] P. Hawken, *Drawdown: The most comprehensive plan ever proposed to reverse global warming*. Penguin, 2017.
- [2] J. Horta, D. Kofman, D. Menga, and A. Silva, "Novel market approach for locally balancing renewable energy production and flexible demand," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct 2017, pp. 533–539.
- [3] J. Horta, E. Altman, M. Caujolle, D. Kofman, and D. Menga, "Real-time enforcement of local energy market transactions respecting distribution grid constraints," in *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Oct 2018, pp. 1–7.
- [4] I. Lopez-Rodriguez, M. Hernandez-Tejera, and A. L. Lopez, "Methods for the management of distributed electricity networks using software agents and market mechanisms: A survey," *Electric Power Systems Research*, vol. 136, pp. 362–369, 2016.
- [5] C. Weinhardt, E. Mengelkamp, W. Cramer, S. Hambridge, A. Hobert, E. Kremers, W. Otter, P. Pinson, V. Tiefenbeck, and M. Zade, "How far along are local energy markets in the dach+ region? a comparative market engineering approach," in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, 2019, pp. 544–549.
- [6] J. K. Kok, C. J. Warmer, and I. Kamphuis, "Powermatcher: multiagent control in the electricity infrastructure," in *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, 2005, pp. 75–82.
- [7] J. Hufen and J. Koppenjan, "Local renewable energy cooperatives: revolution in disguise?" *Energy, Sustainability and Society*, vol. 5, no. 1, pp. 1–14, 2015.

- [8] A. Schreuer and D. Weismeier-Sammer, “Energy cooperatives and local ownership in the field of renewable energy technologies: A literature review,” 2010.
- [9] D. Kalathil, C. Wu, K. Poolla, and P. Varaiya, “The sharing economy for the electricity storage,” *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 556–567, Jan 2019.
- [10] D. Kiedanski, A. Orda, and D. Kofman, “The effect of ramp constraints on coalitional storage games,” in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, ser. e-Energy '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 226–238. [Online]. Available: <https://doi.org/10.1145/3307772.3328300>
- [11] L. Han, T. Morstyn, and M. McCulloch, “Constructing prosumer coalitions for energy cost savings using cooperative game theory,” in *2018 Power Systems Computation Conference (PSCC)*, June 2018, pp. 1–7.
- [12] —, “Incentivizing prosumer coalitions with energy management using cooperative game theory,” *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 303–313, Jan 2019.
- [13] D. Samet and E. Zemel, “On the core and dual set of linear programming games,” *Mathematics of Operations Research*, vol. 9, no. 2, pp. 309–316, 1984. [Online]. Available: <http://www.jstor.org/stable/3689256>
- [14] X. Deng and C. H. Papadimitriou, “On the complexity of cooperative solution concepts,” *Mathematics of Operations Research*, vol. 19, no. 2, pp. 257–266, 1994.
- [15] L. S. Shapley, “On balanced sets and cores,” *Naval Research Logistics Quarterly*, vol. 14, no. 4, pp. 453–460, 1967. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800140404>
- [16] G. Owen, “On the core of linear production games,” *Mathematical Programming*, vol. 9, no. 1, pp. 358–370, Dec 1975. [Online]. Available: <https://doi.org/10.1007/BF01681356>
- [17] E. Kalai and E. Zemel, “Totally balanced games and games of flow,” *Mathematics of Operations Research*, vol. 7, no. 3, pp. 476–478, 1982. [Online]. Available: <http://www.jstor.org/stable/3689611>
- [18] J. C. Duchi, A. Agarwal, and M. J. Wainwright, “Dual averaging for distributed optimization: Convergence analysis and network scaling,” *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, March 2012.
- [19] B. Mohar, “Isoperimetric numbers of graphs,” *Journal of Combinatorial Theory, Series B*, vol. 47, no. 3, pp. 274 – 291, 1989. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0095895689900294>
- [20] A. Lubotzky, *Discrete groups, expanding graphs and invariant measures*. Springer Science & Business Media, 2010.
- [21] J. Friedman, J. Kahn, and E. Szemerédi, “On the second eigenvalue of random regular graphs,” in *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, ser. STOC '89. New York, NY, USA: Association for Computing Machinery, 1989, p. 587–598. [Online]. Available: <https://doi.org/10.1145/73007.73063>
- [22] L. Han, T. Morstyn, and M. McCulloch, “Estimation of the shapley value of a peer-to-peer energy sharing game using coalitional stratified random sampling,” 2019.
- [23] L. Han, T. Morstyn, C. Crozier, and M. McCulloch, “Improving the scalability of a prosumer cooperative game with k-means clustering,” in *2019 IEEE Milan PowerTech*, June 2019, pp. 1–6.
- [24] B. P. Koirala, E. Koliou, J. Friege, R. A. Hakvoort, and P. M. Herder, “Energetic communities for community energy: A review of key issues and trends shaping integrated community energy systems,” *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 722 – 744, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1364032115013477>
- [25] F. Moret and P. Pinson, “Energy collectives: A community and fairness based approach to future electricity markets,” *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3994–4004, Sep. 2019.
- [26] G. Darivianakis, A. Georghiou, R. S. Smith, and J. Lygeros, “A stochastic optimization approach to cooperative building energy management via an energy hub,” in *2015 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 7814–7819.
- [27] H. Kim, J. Lee, S. Bahrami, and V. W. S. Wong, “Direct energy trading of microgrids in distribution energy market,” *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 639–651, Jan 2020.
- [28] P. Chakraborty, E. Baeyens, K. Poolla, P. P. Khargonekar, and P. Varaiya, “Sharing storage in a smart grid: A coalitional game approach,” *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4379–4390, July 2019.
- [29] M. U. Hashmi, A. Mukhopadhyay, A. Bušić, J. Elias, and D. Kiedanski, “Optimal storage arbitrage under net metering using linear programming,” in *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Oct 2019, pp. 1–7.
- [30] Q. Liu, S. Yang, and Y. Hong, “Constrained consensus algorithms with fixed step size for distributed convex optimization over multiagent networks,” *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 4259–4265, Aug 2017.
- [31] C. Feng, F. Wen, S. You, Z. Li, F. Shahnia, and M. Shahidehpour, “Coalitional game based transactive energy management in local energy communities,” *IEEE Transactions on Power Systems*, pp. 1–1, 2019.
- [32] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, “OSQP: An operator splitting solver for quadratic programs,” *ArXiv e-prints*, Nov. 2017.
- [33] G. Banjac, P. Goulart, B. Stellato, and S. Boyd, “Infeasibility detection in the alternating direction method of multipliers for convex optimization,” *Journal of Optimization Theory and Applications*, vol. 183, no. 2, pp. 490–519, 2019. [Online]. Available: <https://doi.org/10.1007/s10957-019-01575-y>
- [34] G. Banjac, B. Stellato, N. Moehle, P. Goulart, A. Bemporad, and S. Boyd, “Embedded code generation using the OSQP solver,” in *IEEE Conference on Decision and Control (CDC)*, 2017. [Online]. Available: <https://doi.org/10.1109/CDC.2017.8263928>

APPENDIX

Proof: We shall prove the theorem by a direct application of Theorem 1. The key idea behind the proof is to show that the linear combination of the solutions for each coalition S using balanced coefficients is feasible in the problem associated with the grand coalition. Because the later is a maximization problem, it holds that the cost of the linear combination is a lower bound of v_N , which proves the theorem. To do this, let γ_S be an arbitrary balanced function satisfying Equation (1).

By linearity of the objective function we have that:

$$\sum_S \gamma_S v_S = \sum_S \gamma_S (\vec{c} \vec{x}_{S^*}) = \vec{c} \left(\sum_S \gamma_S \vec{x}_{S^*} \right)$$

where \vec{x}_{S^*} represents an optimal solution of the optimization problem (7) associated with coalition S .

It remains to show that $\sum_{S \subset N \setminus \emptyset} \gamma_S (A \vec{x}_{S^*}) = \sum_{S \subset N \setminus \emptyset} \gamma_S \vec{b}_{S^*} \stackrel{?}{=} \vec{b}_N$

Indeed, by doing so, we will have shown that $\sum_S \gamma_S v_S \leq v_N$, because the balanced sum $\sum_S \gamma_S \vec{x}_{S^*}$ is feasible in the maximization problem, hence it is a lower bound.

By construction, $\vec{b}_S = \sum_{i \in S} \vec{b}_{\{i\}}$. Consequently,

$$\sum_{S \subset N \setminus \emptyset} \gamma_S \vec{b}_{S^*} = \sum_{i \in S} \vec{b}_{\{i\}} \sum_{S \subset N \setminus \emptyset, i \in S} \gamma_S = \vec{b}_N$$

and this concludes the proof. ■