Expert Systems With Applications 245 (2024) 123132

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

On the estimation of the core for TU games

Alejandro Saavedra-Nieves *, Paula Saavedra-Nieves

CITMAga, Universidade de Santiago de Compostela. MODESTYA Research Group, Rúa Lope Gómez de Marzoa, s/n,, 15782 Santiago de Compostela, Spain

ARTICLE INFO

Keywords:
Game theory
Core estimator
Set estimation
Convex hull
Core-center estimator

ABSTRACT

The core of a transferable utility (TU) game, if it is not empty, is prescribed by the set of all stable allocations. The exact determination of the core reaches exponential time complexity. Therefore, its exact computation is often avoided as the number of players increases. In this work, we propose an estimator for the core of a TU game based on the statistical theory of set estimation. Concretely, we provide a core reconstruction that is obtained in polynomial time for general dimension. Additionally, convergence rates for the estimation error are derived. Finally, a consistent core-center estimator is established as a geometrical application of this methodology.

1. Introduction

In the analysis of expert and intelligent multi-agent decision making environments as an essential element of the society, the links between its components cannot be underestimated. These relationships guarantee that the decisions made by any group of agents. When adopting a collaborative approach, it is essential to take into account a game-theoretical framework used for modelling such cooperation. Thus, the transferable utility (TU) game approach realistically fits settings like social network analysis, economics, politics, management, or organisation (see, for example, Goyal & Kaushal, 2017; Mahdiraji, Razghandi, & Hatami-Marbini, 2021; Saavedra-Nieves & Fiestras-Janeiro, 2022, or Zhou, Lú, Yang, Wang, & Kong, 2015). Formally, the main goal of TU games consists of determining sets of allocations of the joint payoff that players can obtain under cooperation. For this purpose, criteria as the fair compromise for the players, given by the Shapley value (Shapley, 1953) or the nucleolus (Schmeidler, 1969) of a TU game (N, v) among others, and the stability, given by the elements of the core $C(N, v)$ of the TU game (Gillies, 1953) as its center of gravity, the core-center $cc(v)$ (González-Díaz & Sánchez-Rodríguez, 2007), can be considered when justifying the choice of each in which used.

In particular, the core of a TU game is prescribed by the set of all those allocations being efficient from which no coalition of players is incentivised to be deviated. The core elements are said to be stable and a TU game is known as balanced if it has a non-empty core. Formally, the core is a set-type solution for TU games and the task of obtaining core elements is not entirely straightforward as the number of players enlarges. Hence, research on this topic has been focused on selecting subsets (even with a single element) of stable allocations for particular situations. For instance, Guardiola, Meca, and Timmer (2007) analyse the core of TU games applied in supply chains under decentralised control. Drechsel and Kimms (2010) propose a general procedure for the computation of core elements by means of mathematical programming techniques. Algaba, Fragnelli, Llorca, and Sánchez-Soriano (2019), Hadas, Gnecco, and Sanguineti (2017) and Yang, Zhang, and Zhou (2023) use core allocations for transportation problems analysis. Alternatively, two core allocation rules are established in Schouten, Saavedra-Nieves, and Fiestras-Janeiro (2021) for sequencing problems. Luo, Zhou, and Lev (2022) analyse the recent developments and applications of the core of a TU game in operations management. Omrani, Shafaat, and Emrouznejad (2018) use a core element of Integrated Fuzzy Clustering TU games to rank the efficient Decision Making Units in a Data Envelopment Analysis setting.

From a purely geometrical viewpoint, the core $C(N, v)$ of a general TU game (if it exists) is a compact convex polyhedron. A wide collection of papers in the literature are focused on the study of its properties since the definition of the core of a TU game in Gillies (1953). Shapley (1971) can be regarded as the seminal paper dealing with the geometry of the core. Specifically, it proves that the core vertices are given by the set of marginal vectors when the TU game is convex. Table 1 aims to give an overview of some references in the game theoretic literature that study the structure of the core from a geometrical perspective, in terms of its vertices or facets for specific classes of TU games.

The existence of such large number of works focusing on certain features of the core $C(N, v)$ of a TU game is mainly due to the fact that its

* Corresponding author.
E-mail addresses: alejandro.saavedra.nieves@usc.es (A. Saavedra-Nieves), paula.saavedra@usc.es (P. Saavedra-Nieves).

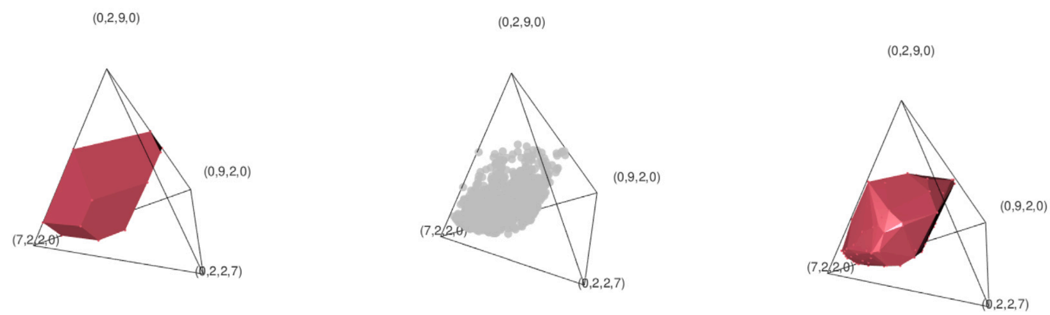
[1] More information available on the link: https://polymake.org/.

[2] More information available on the link: https://porta.zib.de/.

https://doi.org/10.1016/j.eswa.2023.123132
Received 9 June 2023; Received in revised form 15 December 2023; Accepted 31 December 2023
Available online 4 January 2024

Table 1
Summary of references on the theoretical analysis of the geometry of the core.

	Reference	Results
General TU games	Maschler, Peleg, and Shapley (1979)	Analysis the geometry of the core.
	Derks and Kuipers (2002)	Bound of the number of vertices of the core.
	Tijs (2005)	Determining the core vertices.
	Grabisch and Sudhölter (2018)	Upper bound on the number of core vertices.
	Mirás-Calvo, Quinteiro-Sandomingo, and Sánchez-Rodríguez (2020)	Core facets.
Specific classes of TU games	Shapley (1971)	Core vertices for convex TU games.
	Shapley and Shubik (1971)	Core analysis for assignment TU games.
	Aumann and Dreze (1974), Faigle (1989), Myerson (1977) and Owen (1977)	Core for TU games under restricted cooperation.
	Derks and Gilles (1995)	Core for supermodular games.
	Núñez and Rafels (1998)	Core vertices for quasi-convex TU games.
	Núñez and Rafels (2003)	Core vertices for assignment TU games.
	Grabisch (2013)	Core of TU games on ordered structures and graphs.
	González-Díaz, Mirás-Calvo, Quinteiro-Sandomingo, and Sánchez-Rodríguez (2016)	Core-center of airport games.
	Trudeau and Vidal-Puga (2017)	Core vertices for minimum cost spanning tree games.



(a) Core $C(N, v)$ of a TU game (N, v) with 4 players.

(b) A random sample of points supported on $C(N, v)$.

(c) The reconstruction of $C(N, v)$, $\hat{C}(N, v)$.

Fig. 1. Scheme for the reconstruction of $C(N, v)$ for a 4-player TU game from set estimation theory.

exact determination reaches exponential time complexity. Specifically, for a general TU game with n players, the method introduced in Avis and Fukuda (1992) computes the core vertices in time $O(h^2 n v)$ where h denotes the number of non-degenerate inequalities defining the polyhedron and v, the number of vertices. For a TU game with n players, h may be exactly equal to 2^n and, following Derks and Kuipers (2002), v may grow up to $n!$. The exact calculation is therefore a complex task as the number of players increases. Consequently, several software tools for determining and visualising the core are only available up to four players. See, for instance, the R package CoopGame (Staudacher & Anwander, 2019) and the Matlab toolbox *TUGlab* (Mirás-Calvo & Sánchez-Rodríguez, 2006). Of course, general software for research in polyhedral geometry could be also used in this setting. Some examples are the open source softwares *polymake*[1] or *PORTA*.[2]

The main goal of this work is to propose a core estimator $\hat{C}(N, v)$ for a TU game of polynomial time complexity. The properties of compactness and convexity satisfied by the set $C(N, v)$ allow to reconstruct it from set estimation techniques. This statistical theory (closely related to stochastic geometry) focuses on the estimation of a set and any of its features, as its vertices or its volume, from a sample of points randomly taken inside the set. Fig. 1(a) shows the exact core for a 4-players TU game. The scheme to follow to reconstruct the core from set estimation approach is summarised in Fig. 1(b) and (c). According to Fig. 1(b), a random sample supported on $C(N, v)$ need to be generated first. Fig. 1(c) contains the core reconstruction obtained from the sample points as the only input.

As far as we know, the problem of estimating this set-valued solution $C(N, v)$ (and, therefore, its main characteristics such as facets or vertices) has not been considered in the literature. The use of alternative methods for obtaining some of its points (even for its approximation), however, has been addressed. Although Chalkiadakis, Elkind, and Wooldridge (2011) gives an overview on some computational issues related to TU games, Table 2 lists some references for dealing with this task. Sampling methodologies were also applied for estimating other kind of sets of allocations, those composed by a single element.

This paper is organised as follows. Section 2 introduces some background in Cooperative Game Theory. The estimator for the core $C(N, v)$ of a general TU game is presented in Section 3. First, the algorithm to estimate the core of TU game is described and its complexity is discussed. Theoretical bounds for the estimation error are also derived. Since the estimator of $C(N, v)$ is a set, discrepancies between $C(N, v)$ and the proposed reconstruction are measured, commonly in the literature, by computing distances between them. Besides, a specific core estimator for convex TU games is also introduced. Section 4 comprises the simulation results that analyse the consistency of the novel core reconstruction. As an application of the core estimation, we also propose an estimator of the core-center of a TU game in Section 5. Its behaviour is checked from simulations. Section 6 contains some concluding remarks. Finally, Appendix shows the R code needed to obtain the centroid of a polyhedron, which is used throughout the paper.

2. Some background on TU games

This section formally introduces some Game Theory notions to make the rest of the manuscript easier to understand. Our notation is mainly that of González-Díaz, García-Jurado, and Fiestras-Janeiro (2010).

A transferable utility cooperative game (hereafter, TU game) is a pair (N, v), where $N = \{1, \ldots, n\}$ is the set of players and $v : 2^N \longrightarrow \mathbb{R}$

Table 2
Summary of references on alternative methods for the exact/approximate computation of point-valued solution concepts.

Methodology	Reference	Solution concept
Heuristics	Perea and Puerto (2019)	Nucleolus for general TU games
Sampling approximation	Castro, Gómez, and Tejada (2009) and Fernández-García and Puerto-Albandoz (2006)	Shapley value for general TU games.
	Saavedra-Nieves (2023) and Saavedra-Nieves, García-Jurado, and Fiestras-Janeiro (2018)	Owen value for general TU games.
	Bachrach et al. (2010)	Banzhaf value for general TU games.
	Saavedra-Nieves (2023) and Saavedra-Nieves and Fiestras-Janeiro (2021)	Banzhaf-Owen value for general TU games.
	Espinoza Burgos (2020)	Core-center for general TU games.
Stochastic approximation	Benati, López-Blázquez, and Puerto (2019)	Solutions for general TU games.

is a map that assigns to each coalition $T \subseteq N$ a real number $v(T)$ and is $v(\emptyset) = 0$. It represents the worth of the cooperation of the members of T.[3] G^N denotes the class of TU games with player set N.

Two natural questions arise: (1) determine which coalitions will be formed; (2) how to divide the profits of the cooperation of the involved players. Due to the cooperative approach inherent in the situations modelled by TU games, the answer to the first question addresses in the formation of coalition N in such a way that each player obtains a portion of $v(N)$.

An important task is the definition of admissible allocations for players in N to address question (2). Formally, an *allocation* is a vector $x = (x_1, \dots, x_n) \in \mathbb{R}^{|N|}$, with x_i being the total amount assigned to player i, for each $i \in N$. Two requirements for an allocation to be admissible are individual rationality and efficiency. An allocation $x \in \mathbb{R}^{|N|}$ is said to be *individually rational* if $x_i \geq v(\{i\})$ for each player $i \in N$. This means that each player gets at least what he can get for himself. An allocation $x \in \mathbb{R}^{|N|}$ is *efficient* if it distributes the worth of the cooperation of N, i.e. $\sum_{i\in N} x_i = v(N)$. The *set of imputations* of a TU game, $I(N, v)$, consists of all the efficient and individually rational allocations. That is,

$$I(N,v) = \left\{x \in \mathbb{R}^{|N|} \ : \ \sum_{i\in N} x_i = v(N) \text{ and } x_i \geq v(\{i\}), \text{ for all } i \in N\right\}. \tag{1}$$

Generally, *solution concepts* are introduced as mechanisms providing each TU game with sets of allocations. Formally, a solution concept is a map assigning each TU game an allocation set. With respect to its cardinality, we distinguish two cases.

Firstly, we mention the *point-valued solution concepts*. Among others, the *Shapley value* (Shapley, 1953) of (N, v), denoted by $Sh(N, v)$, is formally defined by

$$Sh_i(N,v) = \frac{1}{|\Pi(N)|} \sum_{\sigma\in\Pi(N)} m_i^\sigma(v), \text{ for each } i \in N\ , \tag{2}$$

being $\Pi(N)$ the set of orders of N, $P_i^\sigma = \{k \in N \ : \sigma(k) < \sigma(i)\}$ the set of the *predecessors* of i in any order $\sigma \in \Pi(N)$ and $m_i^\sigma(v) = v(P_i^\sigma \cup i) - v(P_i^\sigma)$, for each $\sigma \in \Pi(N)$, the marginal contribution of player i to its predecessors in σ.

Secondly, we focus on *set-valued solution concepts*. Although the individual rationality ensures that no player can block an imputation, a coalition of players may have incentives to block some of the allocations in $I(N, v)$. Gillies (1953) introduces the *core* as the subset of $I(N, v)$ given by those coalitionally rational allocations. Fixed $(N, v) \in G^N$, an allocation $x \in \mathbb{R}^{|N|}$ is *coalitionally rational* if, for each $T \subseteq N$, $\sum_{i\in T} x_i \geq v(T)$. Formally, the *core* of (N, v), denoted by $C(N, v)$, is specified by

$$C(N,v) = \left\{x \in \mathbb{R}^{|N|} : \ \sum_{i\in N} x_i = v(N) \text{ and } \sum_{i\in T} x_i \geq v(T), \text{ for each } T \subset N\right\}. \tag{3}$$

[3] The TU game is called a cost game when the costs of cooperation are associated. It is denoted by (N, c).

The core elements are said to be stable since that there is no coalition $T \subseteq N$ that can complain about the allocations of $v(N)$ belonging to this set. In some TU games the core can prescribe the empty set. Bondareva (1963) and Shapley (1967) characterise the non-emptiness character of the core of a TU game as follows. Given $(N, v) \in G^N$, $C(N, v) \neq \emptyset$ if and only if for each family of coalitions $\mathcal{F} \subset 2^N \setminus \emptyset$ and a collection of positive real numbers $\{\alpha_T \ : \ T \in \mathcal{F}\}$ with $\sum_{T\in\mathcal{F}, i\in T} \alpha_T = 1$ for each $i \in N$, $\sum_{T\in\mathcal{F}} \alpha_T v(T) \leq v(N)$. However, the non-empty character of the core can be alternatively checked from a solution concept considered as its "relaxed" version. The ε-*core* of (N, v) is given, for any $\varepsilon \in \mathbb{R}$, by

$$C_\varepsilon(N,v) = \left\{x \in \mathbb{R}^{|N|} \ : \ \sum_{i\in N} x_i = v(N) \text{ and } \sum_{i\in T} x_i \geq v(T) - \varepsilon, \text{ for each } T \subset N\right\},$$

and the *least core* of (N, v) (cf. Maschler et al., 1979) is specified by $C_{\varepsilon^*}(N, v)$, where $\varepsilon^* = \min\{\varepsilon \ : \ C_\varepsilon(N, v) \neq \emptyset\}$. Note that $C(N, v)$ is not empty if and only if $\varepsilon^* \leq 0$.

The exhaustive study of the core becomes a complex issue in cooperative game theory given that the number of the associated constraints increases considerably with the number of players. However, core elements can be easily obtained for certain classes of TU games. We highlight the case of convex TU games. A TU game (N, v) is *convex* if

$$v(T \cup \{i\}) - v(T) \leq v(T' \cup \{i\}) - v(T'), \tag{4}$$

for all $i \in N$ and for all $T \subset T' \subseteq N \setminus \{i\}$. This property ensures that players prefer to join large coalitions. Although for specific classes of TU games this property is always satisfied, its checking for general TU games requires a very high computational complexity. This is mainly due to the fact that, if the number of players enlarges, the number of conditions as the ones in (4) also grows exponentially. Regarding the core, it is non-empty since that the Shapley value of convex games always belongs to it. The reason is that, for convex TU games, $m_i^\sigma(v) \in C(N, v)$ for each $\sigma \in \Pi(N)$ and that, in addition, the core is determined by the convex hull of the marginal contributions (see Weber, 1988, for more details). That is,

$$C(N,v) = H(\{m^\sigma(v) : \ \sigma \in \Pi(N)\}), \tag{5}$$

where $H(\cdot)$ denotes the operator convex hull of a given set of points.

In general, if the core of (N, v) is non-empty, then it is a convex and compact polyhedron of dimension at most $|N| - 1$, since that it is contained in the hyperplane $\sum_{i\in N} x_i = v(N)$. Barycentric coordinates are used to parametrise the interior of an $(|N| - 1)$-simplex by $|N|$ real numbers in the interval $[0, 1]$. A k-simplex is a polytope of dimension k that is prescribed by the convex hull of its $k+1$ vertices. More precisely, suppose that the $k+1$ points $\{u_0, \dots, u_k\} \in \mathbb{R}^k$ are affinely independent. That is, $u_1 - u_0, \dots, u_k - u_0$ are linearly independent. Formally, the simplex that they prescribe corresponds to the set of points

$$S(\{u_0, u_1, \dots, u_k\}) = \left\{\gamma_0 u_0 + \cdots + \gamma_k u_k \ : \ \sum_{j=0}^{k} \gamma_j = 1 \text{ and } \gamma_j \geq 0 \text{ for } j = 0, \dots, k\right\}.$$

Directly, fixed a $(k + 1)$-simplex, it is possible to compute the Cartesian coordinates of one or more normalised points in $\mathbb{R}^k$, given their barycentric coordinates in $\mathbb{R}^{k+1}$, through the corresponding affine transformation (and vice versa). For instance, it helps to represent the core of a TU game with 3 and 4 players in $\mathbb{R}^2$ and $\mathbb{R}^3$, respectively. Fig. 2(a) is an example of representation for the core of TU games with 4 players by using the $(|N|-1)$-simplex induced by the canonical basis of $\mathbb{R}^{|N|-1}$ jointly with the null vector in $\mathbb{R}^{|N|-1}$.

Hereafter, we suppose that the characteristic function v can be obtained in polynomial time for every coalition $T \subseteq N$. The algorithm for core estimator proposed in this work, also covers the rest of TU games but polynomial time complexity may not be guaranteed. Furthermore, the case of an empty core is excluded. However, even if the core of a TU game is empty, the estimation methodology proposed here is directly applicable to the reconstruction of the least core because, following results in Maschler et al. (1979), it is also a convex and a non-empty polyhedron.

3. Polynomial estimation of the core for a TU game

An estimator for the core $C(N,v)$ of a TU game will be proposed from set estimation techniques in this section. A classical topic of this theory is concerned, as in this work, with the estimation of a distribution support. Therefore, we briefly review the support estimation problem in order to propose a general estimation algorithm for $C(N,v)$. Polynomial time complexity is checked for the presented methodology. Moreover, theoretical convergence rates for the estimation error (in Hausdorff distance) are derived. Finally, a specific procedure for reconstructing $C(N,v)$ for the family of convex TU games is introduced.

3.1. Algorithm for core estimation of general TU games

Mathematically, support estimation theory solves the task of reconstructing the compact and non-empty support $S \subset \mathbb{R}^D$ of an absolutely continuous random vector X from a random sample $\mathcal{X}_m = \{X_1, \dots, X_m\}$ of X. Cuevas and Fraiman (2010) provide a extensive review on this topic. In particular, when the support S is assumed to be convex, then the most natural estimator of S is the convex hull of the sample points, $H(\mathcal{X}_m)$, defined as the intersection of all convex sets in $\mathbb{R}^D$ containing $\mathcal{X}_m$. It can be easily checked that $H(\mathcal{X}_m)$ is the smallest convex set that contains $\mathcal{X}_m$. See, for a thorough theoretical analysis of this estimator, Dümbgen and Walther (1996), Reitzner et al. (2003) or Schneider (1988, 2014).

If the core of a TU game is non-empty, the definition of $C(N,v)$ in Section 2 ensures that it is a convex and compact polyhedron in $\mathbb{R}^{|N|-1}$, by the property of efficiency of its elements. If $C(N,v)$ is non-empty, support estimation approach allows to establish an estimator of $C(N,v)$. The convex hull of a set of random observations supported on $C(N,v)$, would be a natural reconstruction of $C(N,v)$. Therefore, it is necessary to address the problem of random generation of observations $\mathcal{X}_m$ supported on $C(N,v)$. Furthermore, if $\mathcal{X}_m$ has uniform distribution, consistency of $H(\mathcal{X}_m)$ as an estimator of $C(N,v)$, is guaranteed and convergence rates can be derived.

Conventional rejection techniques could be used for uniform sampling generation on $C(N,v)$. However, they are computationally inefficient comparing to polynomial time complexity algorithms. Dyer, Frieze, and Kannan (1991) introduce the first polynomial time algorithm for sampling convex sets. However, its practical implementation was not developed because the degree of the polynomial (over 20). Smith (1984) proposes the hit-and-run sampling method for Monte Carlo generation of points uniformly distributed within an arbitrary bounded (measurable) region in $\mathbb{R}^D$. Concretely, interactive solution presented in Smith (1984) produces points with asymptotic uniform distribution within the region of interest through a class of Markovian methods. Roughly speaking, given a starting point $p_0 \in C(N,v)$, a random line L through p_0 is selected (uniformly over all directions). The next chain observation is selected uniformly from the segment of the line inside $C(N,v)$. Time complexity analysis in Section 3.1.1 shows that the arbitrary choice of p_0 does not mutate the polynomial complexity.

The underlying ideas allow to formalise an algorithm for reconstructing the core $C(N,v)$ of a TU game (if it exists) for general dimension. Let $(N,v) \in G^N$ such that $C(N,v) \neq \emptyset$. Below, the steps of the estimation procedure are described in detail.

1. The set to be estimated is $C(N,v)$, the set of stable allocations for N.
2. $C(N,v)$ is considered as the support of a uniform distribution.
3. A sample of size m supported on $C(N,v)$, $\mathcal{X}_m$, has to be generated by using hit-and-run.
4. Compute the convex hull of $\mathcal{X}_m$, $H(\mathcal{X}_m)$, as the reconstruction of $C(N,v)$, $\hat{C}(N,v)$.

The pseudocode of the resulting procedure is summarised in Algorithm 1. Normalisation of all core allocations were considered in Algorithm 1. Specifically, we consider as $(|N|-1)$-simplex the points in $\mathbb{R}^{|N|-1}$ given by the first $|N|-1$ components of the canonical basis of $\mathbb{R}^{|N|}$. Possible computational problems in large-scale are avoided with this operation, that depends on the characteristic function v.

Figs. 2(a) and 3(a) shows the cores of two different TU games with 3 and 4 players, respectively. The estimators $\hat{C}(N,v)$ computed from two hit-and-run samples of sizes $m = 100$ and $m = 1000$ are also shown in (c) and (e). Of course, core reconstruction improves considerably as m increases, specially for 4 players. Once the estimator of $C(N,v)$ is determined, the volume of the core could be approximated by the volume of the estimator. But, also other characteristic elements such as facets or vertices could be analysed from the corresponding ones of the reconstruction.

One natural question that may arise from this methodological proposal for core estimation is related to check if its capabilities still offer a good performance on those TU games with $|N|$ players whose associated core, after the above-mentioned normalisation, is of dimension strictly less than $|N|-1$. Following Smith (1984), hit-and-run algorithm provides a solution for the problem of sample generation on a general bounded D'-dimensional surface contained in $\mathbb{R}^D$ with polynomial complexity for all $D' \leq D$. The convex hull computation in a lower dimensional space in which the $C(N,v)$ is contained does not present any limitation either.

Algorithm 1: Pseudocode of the algorithm for reconstructing $C(N,v)$.

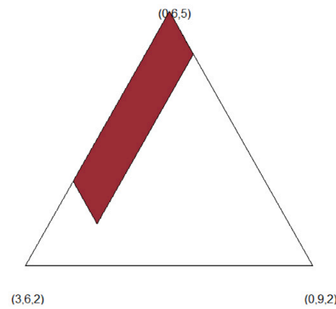
Inputs: Take a sampling size equal to m and let $\{u_0, u_1, \dots, u_{|N|-1}\}$ be a $(|N|-1)$-simplex.

Step 1: Generate the random sample $\mathcal{X}_m$ supported on $C(N,v)$ by using hit-and-run.

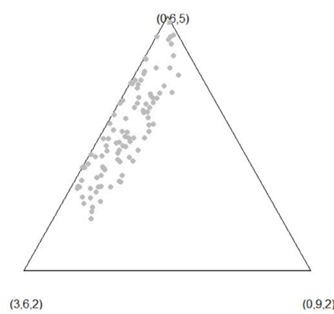
Set $l = 1$ and do $\mathcal{X}_m = \emptyset$
while $l \leq m$ **do**
Generate a random observation $X_l = (x_1^l, \dots, x_n^l)$ supported on $C(N,v)$.
Do $\hat{X}_l = \left(\frac{x_1^l}{v(N)}, \frac{x_2^l}{v(N)}, \dots, \frac{x_n^l}{v(N)}\right) \cdot \begin{pmatrix} u_0 \\ u_1 \\ \dots \\ u_{|N|-1} \end{pmatrix}$, $\mathcal{X}_m = \mathcal{X}_m \cup \hat{X}_l$ and
$l = l + 1$.

Step 2: Compute $H(\mathcal{X}_m)$.

Output: Do $\hat{C}(N,v) = H(\mathcal{X}_m)$.

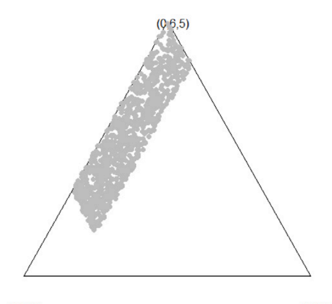


(a) Core $C(N, v)$ of a TU game (N, v) with 3 players.

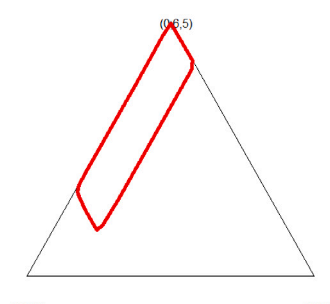


(b) Uniform sample $\mathcal{X}_{100}$ on $C(N, v)$.

(c) $\hat{C}(N, v) = H(\mathcal{X}_{100})$.



(d) Uniform sample $\mathcal{X}_{1000}$ on $C(N, v)$.



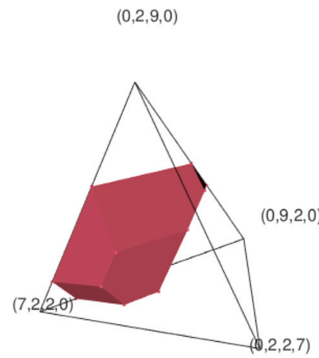
(e) $\hat{C}(N, v) = H(\mathcal{X}_{1000})$.

Fig. 2. Convex hull (right) of uniform samples (left) of sizes $m = 100$ (b, c) and $m = 1000$ (d, e) on $C(N, v)$ for a 3-player TU game.

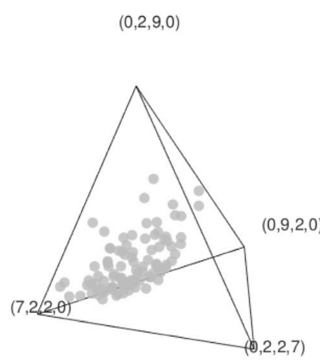
3.1.1. Analysis of time complexity for core estimation algorithm

Studying the time complexity of Algorithm 1 requires to analyse the two steps main steps of the procedure. The first step (step 1) is focused on the generation of the sample $\mathcal{X}_m$ supported on $C(N, v)$. Several works in the literature analyse the time complexity of hit-and-run algorithm. Dyer, Frieze, and Stougie (1994) obtain a polynomial bound on its mixing time. For a convex body in $\mathbb{R}^D$, Lovász (1999) shows that hit-and-run produces an approximately uniformly distributed sample of points in time $O^*(D^3)$[4] when p_0 is properly selected. Lovász and Vempala (2006) improve this result by showing that hit-and-run actually mixes rapidly from any interior starting point. Although the mixing time could be big if the starting point is located in a very tight corner, the derived bound is logarithmic in the distance.

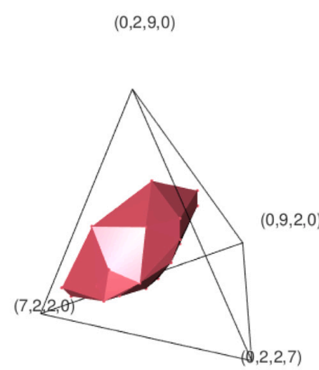
[4] The O^* notation eliminates factors that involve logarithms or error bounds.



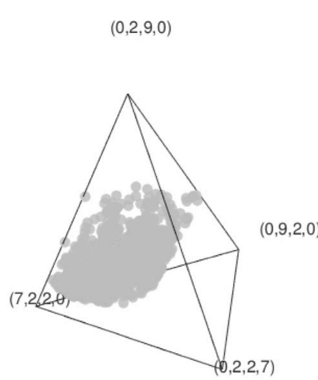
(a) Core $C(N, v)$ of a TU game (N, v) with 4 players.



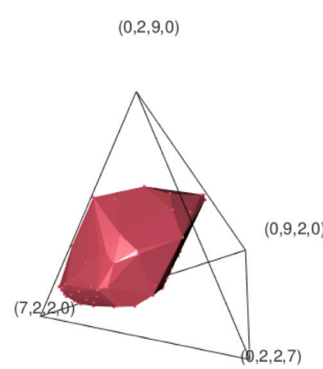
(b) Uniform sample $\mathcal{X}_{100}$ on $C(N, v)$.



(c) $\hat{C}(N, v) = H(\mathcal{X}_{100})$.



(d) Uniform sample $\mathcal{X}_{1000}$ on $C(N, v)$.



(e) $\hat{C}(N, v) = H(\mathcal{X}_{1000})$.

Fig. 3. Convex hull (right) of uniform samples (left) of sizes $m = 100$ (b, c) and $m = 1000$ (d, e) on $C(N, v)$ for a 4-player TU game.

Additionally, hit-and-run is already implemented in R (R Core Team, 2024). For instance, libraries `hitandrun`[5] and `volesti`[6] can be considered for hit-and-run sampling generation. Furthermore, users do not need to specify p_0 as an input in practice. For instance, package `volesti` computes p_0 as the center of the largest inscribed ball in $C(N, v)$.

The second step (step 2) of Algorithm 1 requires the convex hull computation for the sample $\mathcal{X}_m$ generated in Step 1. Deterministic algorithms for obtaining the convex hull of m points with running time $O(m \log m)$ were long established in the 2- and 3-dimensional spaces (Graham, 1972 and Preparata & Hong, 1977). For higher dimensions D, Seidel (1981) establishes a deterministic algorithm with time complexity $O(m \log m + m^{\lfloor D/2 \rfloor})$ for the case where D is even. Theorem 3.5 in Chazelle (1993) generalises this optimal result for general dimension D. Therefore, Theorem 3.1 ensures that the convex hull computation has polynomial complexity for a fixed space dimension.

Theorem 3.1 (*Chazelle, 1993*)**.** *Let $\{p_1, \ldots, p_m\}$ a set of m ($m > 1$) points in $\mathbb{R}^D$. Then, the convex hull of $\{p_1, \ldots, p_m\}$, $H(\{p_1, \ldots, p_m\})$, can be computed deterministically in $O(m \log m + m^{\lfloor D/2 \rfloor})$ time, which is optimal.*

In practice, the convex hull computation for a collection of points in spaces of general dimension is implemented in R package `geometry`.[7] The polynomial complexity of Algorithm 1 for core estimation is established in Proposition 3.2.

Proposition 3.2. *Let $(N, v) \in G^N$ be a TU game with a non-empty core $C(N, v)$ and characteristic function v obtainable in polynomial time. Then, the core estimator $\hat{C}(N, v) = H(\mathcal{X}_m)$ specified by Algorithm 1 has polynomial time complexity.*

Proof. From its definition in (3), we have ensured that $C(N, v) \subset \mathbb{R}^{|N|-1}$ is a convex polyhedron where $|N|$ is the number of players in N. Therefore, analysis in Lovász and Vempala (2006) ensure that time complexity of hit-and-run method applied on $C(N, v)$ is polynomial. Concretely, $O^*(|N|^3)$. Since v is also obtainable in polynomial time, polynomial complexity of step 1 in Algorithm 1 is checked. As for step 2, Theorem 3.1 ensures that the computation of $H(\mathcal{X}_m)$ is also polynomial fixed the space dimension $|N| - 1$. □

3.1.2. Estimation error and convergence rates

The problem of measuring the estimation error of the core reconstruction can be addressed by computing the distance between the theoretical (or exact) core and the corresponding estimator. The Euclidean distance in $\mathbb{R}^D$ is not the best choice to analyse discrepancies between sets. In particular, if two sets share a border, the Euclidean distance between them is zero. However, they could be quite different. Therefore, one of the most used distances to measure the estimation error when the estimator is a set, is the distance in measure.

Definition 3.3. The distance in measure between two bounded Borel sets A and B in $\mathbb{R}^D$ is defined by

$$d_\mu(A, B) = \mu(A \triangle B),$$

where the Lebesgue measure is denoted by μ, and the symmetric difference by $\triangle$, (see Fig. 4,b), that is,

$$A \triangle C = (A \setminus B) \cup (B \setminus A).$$

[5] More information available on: https://CRAN.R-project.org/package=hitandrun.

[6] More information available on: https://CRAN.R-project.org/package=volesti.

[7] More information available on: https://cran.r-project.org/package=geometry.

Remark 3.1. Distance d_μ does not satisfy the conditions to be a metric. Let A_1 and A_2 be two bounded Borel sets that differ by a finite set of points. Then, $d_\mu(A_1, A_2) = 0$ however $A_1 \neq A_2$. Therefore, d_μ does not penalise those set estimators that contain spurious connected components with null Lebesgue measure.

Fig. 4(a) shows the exact core $C(N, v)$ (red colour) for a TU game of 3 players and the convex hull (grey colour) of a random sample of size $m = 10$ supported on $C(N, v)$. As an illustration, the distance in measure between $C(N, v)$ and $H(\mathcal{X}_{10})$ is also represented in Fig. 4(b). The exact cores $C(N, v)$ represented in Figs. 2 and 3(a) have been also reconstructed as the convex hull of uniform samples of sizes $m = 100$ and $m = 1000$ shown in (b) and (d). For the example in Fig. 2, $d_\mu(C(N, v), H(\mathcal{X}_{100})) = 0.11648$ and $d_\mu(C(N, v), H(\mathcal{X}_{1000})) = 0.01482$. For the estimator shown in Fig. 3, we have that $d_\mu(C(N, v), H(\mathcal{X}_{100})) = 0.00340$ and $d_\mu(C(N, v), H(\mathcal{X}_{1000})) = 0.00092$. As natural, the error decreases considerably as the sample size increases.

Another classical choice to measure the approximation error in this setting, is the Hausdorff distance. Fig. 4(b) contains the graphical representation of the Hausdorff distance between $C(N, v)$ and $H(\mathcal{X}_{10})$ for a specific 3−player TU game.

Definition 3.4. The Hausdorff distance between two nonempty compact subsets A and B of $\mathbb{R}^D$ is defined by

$$d_H(A, B) = \max \left\{ \sup_{a \in A} \rho(a, B), \sup_{b \in B} \rho(b, A) \right\},$$

where

$$\rho(a, B) = \inf\{ \|a - b\| : b \in B \}$$

and $\| \cdot \|$ is the Euclidean norm.

Remark 3.2. Section 1.4 in Matheron (1975) shows that d_H is a metric. However, the metric d_H is not able to detect differences in shape properties. In fact, two sets with a small Hausdorff distance, could present very different shapes. This usually occurs when the boundaries ∂A and ∂B are far apart, no matter the proximity of A and B. Visual proximity of two sets can be validated through the Hausdorff distance between their frontiers.

Dümbgen and Walther (1996) provide the convergence rates in Hausdorff distance for convex supports under uniform distribution assumption on the sample observations.

Theorem 3.5 (*Dümbgen & Walther, 1996*)**.** *Let S be a convex compact set in $\mathbb{R}^D$ with non-empty interior. Let $\mathcal{X}_m$ be a uniform and independent, identically distributed random sample on S. Then,*

$$d_H(S, H(\mathcal{X}_m)) = O\left(\left(\frac{\log m}{m} \right)^{1/D} \right), \text{ almost surely.}$$

As a consequence, Corollary 3.6 establishes the mathematical consistency and also the estimation error rates for the core reconstruction proposed in this work.

Corollary 3.6. *Let $(N, v) \in G^N$ be a TU game with a non-empty core $C(N, v)$. Let $\mathcal{X}_m$ be a uniform and independent, identically distributed random sample on $C(N, v)$. Then, $\hat{C}(N, v) = H(\mathcal{X}_m)$ satisfies that*

$$d_H(C(N, v), \hat{C}(N, v)) = O\left(\frac{\log m}{m} \right)^{(1/(|N|-1))}, \text{ almost surely.} \tag{6}$$

Proof. Since $\hat{C}(N, v)$ is defined as $H(\mathcal{X}_m) \subset \mathbb{R}^{|N|-1}$, Eq. (6) is a straightforward of convergence rate (in Hausdorff distance) for the convex hull established in Theorem 3.5. □

As expected, convergence rate in Corollary 3.6 shows that the convex hull convergence is slower as the space dimension increases

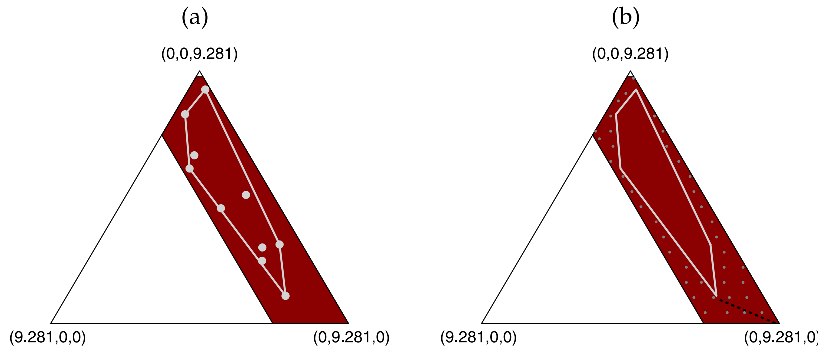


Fig. 4. (a) $C(N,v)$ (red colour) for a 3-player TU game and a core reconstruction, $H(\mathcal{X}_{10})$ (grey colour). (b) Distance in measure (dotted area) and Hausdorff distance (black dashed line) between $C(N,v)$ and $H(\mathcal{X}_{10})$.

(Hughes effect) or, equivalently, the number of players is bigger. However, consistency of $\hat{C}(n,v)$ is guaranteed for a general number of players. Moreover, under assumptions in Corollary 3.6, it is verified that $\mathcal{X}_m \subset C(N,v)$ with probability one. Consequently, $\hat{C}(N,v) = H(\mathcal{X}_m) \subset C(N,v)$. Therefore, $\hat{C}(N,v)$ underestimates $C(N,v)$ and, as consequence, allocations outside $C(N,v)$ do not belong to the proposed estimator $\hat{C}(N,v)$.

As far as we are aware, neither consistency results of convex hull frontier for Hausdorff distance nor convex hull convergence rates for distance in measure have been derived yet. As a consequence, consistency of the estimator introduced in Algorithm 1 for these both error criteria will be analysed from simulations in Section 4.

3.2. Algorithm for core estimation of convex TU games

In this section, we propose an alternative procedure for reconstructing the core $C(N,v)$ for the subclass of convex games. Let $(N,v) \in G^N$ be a TU game. Formally, (N,v) is said to be convex if it satisfies the condition in (4).

This class of TU games received much attention because of the important properties that it guarantees. Among others, the Shapley value always belongs to its associated core (see Shapley & Shubik, 1971). This fact ensures the non-emptiness of $C(N,v)$ under convexity. As we have previously mentioned, the vector of marginal contributions induced by a given order σ always belongs to $C(N,v)$ under the convexity of (N,v). That is, the vector $m^\sigma(v) \in C(N,v)$, where $m^\sigma(v)$ is given by

$$m_i^\sigma(v) = v(P_i^\sigma \cup i) - v(P_i^\sigma), \text{ for all } i \in N,$$

with $\Pi(N)$ the set of permutations of N and $P_i^\sigma = \{k \in N : \sigma(k) < \sigma(i)\}$ for a fixed $\sigma \in \Pi(N)$. In this scenario, we use the fact that the core of convex games in (5) is the convex hull of all marginal contributions, i.e. we have that

$$C(N,v) = H(\{m_i^\sigma(v) : \sigma \in \Pi(N)\}),$$

with $H(\cdot)$ the operator convex hull of a collection of points and that bases our sampling proposal.

Let $(N,v) \in G^N$ be a convex TU game whose characteristic function is obtained in polynomial time. The sampling procedure steps are detailed below.

1. The core of (N,v), $C(N,v)$, is considered as the unknown set to be estimated.
2. The sampling population is the set of permutations of N, i.e. $\Pi(N)$.
3. For each sampling unit $\sigma \in \Pi(N)$, we obtain the vector $(m_i^\sigma(v))_{i\in N}$ given by $m_i^\sigma(v) = v(P_i^\sigma \cup i) - v(P_i^\sigma)$ for all $i \in N$.
4. The sampling procedure takes each permutation $\sigma \in \Pi(N)$ with equal probability. In this way we obtain a sample $\{\sigma_1, \dots, \sigma_m\}$ of m orders of N obtained with replacement.
5. A sample of size m supported on $C(N,v)$, $\mathcal{X}_m$, is obtained as the marginal contribution vectors over the sample of permutations $\mathcal{S}$.
6. Compute the convex hull of $\mathcal{X}_m$, $H(\mathcal{X}_m)$, as the reconstruction of $C(N,v)$, $\hat{C}(N,v)$.

Algorithm 2: Pseudocode of the algorithm for reconstructing $C(N,v)$, when (N,v) is convex.

Inputs: Take a sampling size equal to m and let $\{u_0, u_1, \dots, u_{|N|-1}\}$ be a $(|N|-1)$-simplex.

Step 1: Generate the random sample $\mathcal{X}_m$ of vertices of $C(N,v)$.

Set $l = 1$ and do $\mathcal{X}_m = \emptyset$
while $l \le m$ **do**
  Randomly, generate a permutation $\sigma \in \Pi(N)$.
  Obtain $m_i^\sigma(v) = v(P_i^\sigma \cup i) - v(P_i^\sigma)$, for all $i \in N$.
  Do $X_l = (x_1^l, \dots, x_n^l)$, such that $x_i^l = m_i^\sigma(v)$ for all $i \in N$.
  Do $\hat{X}_l = \left(\frac{x_1^l}{v(N)}, \frac{x_2^l}{v(N)}, \dots, \frac{x_n^l}{v(N)}\right) \cdot \begin{pmatrix} u_0 \\ u_1 \\ \dots \\ u_{|N|-1} \end{pmatrix}$, $\mathcal{X}_m = \mathcal{X}_m \cup \hat{X}_l$ and
  $l = l + 1$.

Step 2: Compute $H(\mathcal{X}_m)$.

Output: Do $\hat{C}(N,v) = H(\mathcal{X}_m)$.

Algorithm 2 formalises the steps of this procedure. Through a sample of permutations of N, we obtain an associated sample of core vertices for the convex TU game (N,v) by considering the marginal contributions over such sample. Thus, the estimation of the core $C(N,v)$, namely $\hat{C}(N,v)$, is obtained as the convex hull of the sampled vertices. Again, it follows that $\hat{C}(N,v)$ is always a subset of $C(N,v)$, almost surely. In contrast to Derks and Kuipers (2002), it is important to remark that the computation time for obtaining a sample of vertices is still polynomial.

4. Simulation study

The performance of the method for reconstructing the core of a TU game in Section 3 is explored next from simulations. A total of 10 simulation scenarios have been considered and they correspond to the different TU games that are described below. All of them have non-empty cores. In these reduced scenarios, $C(N,v)$ can be obtained exactly through specific game-theoretical tools. We will analyse the estimation error of the methodology proposed along this work, by measuring the (frontiers) Hausdorff distance and the distance in measure

Table 3
TU game associated to Model 3.A.

T	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	N
$v(S)$	0	0	0	1500	500	500	2000

Table 4
TU game associated to Model 3.B.

T	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	N
$v(S)$	0	0	0	0.219	0	6.924	9.281

Table 5
TU game associated to Model 3.C.

T	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	N
$v(S)$	0	0	0	0	0	100	200

Table 6
TU game associated to Model 4.A.

T	$\emptyset$	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$
$v(T)$	0	0	0	0	0	3	5	7
T	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$	$\{1,2,3\}$	$\{1,2,4\}$	$\{1,3,4\}$	$\{2,3,4\}$	N
$v(T)$	5	4	6	20	30	40	50	100

Table 7
TU game associated to Model 4.B.

T	$\emptyset$	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$
$v(T)$	0	0	0	0	0	1	2	4
T	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$	$\{1,2,3\}$	$\{1,2,4\}$	$\{1,3,4\}$	$\{2,3,4\}$	N
$v(T)$	5	7	7	12	14	17	12	22

between the exact cores and the corresponding core estimator. In the simulation study, we have restricted our scenarios up to 10 players. For more players, the exact determination of the core, required to measure the estimation error, involves an unavoidable computational cost.

In what follows, these TU games considered will be referred to in the form $n.X$ where n will denote the number of players and X will distinguish each of the cases under study with this number of players.

- **Model 3.A.** Firstly, we take the saving game associated to the problem of the visiting professor described in González-Díaz et al. (2010). It is a 3-player situation modelled by the TU game (N, v) which assigns, for each $T \subseteq N$, the values in Table 3.
- **Model 3.B.** Now, we take Example 4.6 from Saavedra-Nieves, Schouten, and Borm (2020). It corresponds to the 3-player general sequencing game (N, v) depicted in Table 4.
- **Model 3.C.** The last 3-player TU game we consider is a bankruptcy game. Recall that a *bankruptcy game* is the TU game (N, v) that assigns, for each coalition $T \subseteq N$,

$$v(T) = \max\left\{0, E - \sum_{i \notin T} d_i\right\}, \tag{7}$$

  being E the estate, that is, the amount to be distributed among the claimants in N with demands d_i, for every $i \in N$. This problem was initially introduced in O'Neill (1982). We consider the bankruptcy situation in González-Díaz et al. (2010), that results from the estate $E = 200$ and from the individual demands given by $d_1 = 100$, $d_2 = 200$, and $d_3 = 300$. Table 5 depicts the considered TU game.
- **Model 4.A.** The first 4-player TU game used in this simulation study is again a saving game, that is displayed in Table 6.
- **Model 4.B.** Next, we take the 4-player TU game displayed in Table 7, also a saving game.
- **Model 4.C.** The concluding 4-player example is the bankruptcy TU game shown in Table 8.

Table 8
TU game associated to Model 4.C.

T	$\emptyset$	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$
$v(T)$	0	0	0	0	0	0	0	0
T	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$	$\{1,2,3\}$	$\{1,2,4\}$	$\{1,3,4\}$	$\{2,3,4\}$	N
$v(T)$	0	1	4	1	3	6	8	10

- **Model 5.** In spite of the computational difficulties that may arise when considering 5 players, we also consider the bankruptcy game with 5 players associated to the bankruptcy problem arisen when dividing an estate $E = 600$ with vector of claims $d = (200, 300, 120, 110, 230)$. The associated bankruptcy game (N, v) is fully given by Expression (7) for each possible coalition.
- **Model 6.** Below, we take a standard sequencing game with 6 players. A standard sequencing situation (cf. Smith, 1956) is a multi-agent sequencing situation given by $(N, \sigma_0, \{p_i\}_{i\in N}, \{c_i\}_{i\in N})$, where σ_0 prescribes the initial order for the jobs of N and, for every $i \in N$, p_i is the i's job processing time and c_i a linear time-dependent function $c_i : [0, \infty) \longrightarrow \mathbb{R}$ such that $c_i(t) = \alpha_i t$, with α_i being the i's linear cost coefficient. Thus, a saving TU game (N, v) is associated to any sequencing situation as follows. For each $T \subseteq N$ connected by σ_0,

$$v(T) = \sum_{(i,j)\in MP, i,j\in T} (\alpha_j p_i - \alpha_i p_j), \tag{8}$$

  with $MP = \left\{(i, j) \in N \times N \mid \alpha_j p_i - \alpha_i p_j > 0\right\}$. If T is disconnected, $v(T)$ is obtained by the sum of the savings of its components in σ_0. To check the performance of our proposal, we consider the case with $N = \{1, 2, 3, 4, 5, 6\}$, $\sigma_0 = (1, 2, 3, 4, 5, 6)$, $p = (3, 4, 6, 1, 3, 4)$ and $\alpha = (1, 2, 4, 2, 5, 2)$.
- **Model 8.** Now, we consider the standard sequencing game with 8 players associated $(N, \sigma_0, \{p_i\}_{i\in N}, \{c_i\}_{i\in N})$, with $N = \{1, 2, 3, 4, 5, 6, 7, 8\}$, $\sigma_0 = (1, 2, 3, 4, 5, 6, 7, 8)$, $p = (3, 4, 6, 1, 3, 4, 5, 4)$ and $\alpha = (1, 2, 4, 2, 5, 2, 1, 4)$.
- **Model 10.** Finally, we consider the 10-players TU game (N, v) defined as follows.

$$v(T) = \begin{cases} 0, & \text{if } T = \{i\}, \text{ with } i \in N, \\ \frac{3}{4}\frac{|T|}{|N|}, & \text{if } T \subset N \text{ such that } \{10\} \in T, \\ \frac{|T|}{|N|}, & \text{otherwise.} \end{cases}$$

Firstly, we consider the case of reconstructing the core of general TU game through a uniform sample of core-elements. More specifically, a total of 250 samples (of size m) supported on the core of the 10 simulation models have been generated by using hit-and-run approach. Different sample sizes m have been also considered: 10, 25, 50, 100, 200, 250, 500, 1000, 2000 and 5000. For each sample, we have calculated the core estimator $\hat{C}(N, v)$ established in Algorithm 1 and, then, we have computed the estimation error as the distance in measure between $C(N, v)$ and $H(\mathcal{X}_m)$. Since the convergence rates are not available for this error criterion, this simulation study allows to analyse the estimator consistency for this particular error criterion. The Hausdorff distance between $C(N, v)$ and $H(\mathcal{X}_m)$ is not considered in these simulations because theoretical convergence rates were fully established in Corollary 3.6. Moreover, its determination (specially in high dimensions) would be really complex. However, the computation of Hausdorff distance between $\partial C(N, v)$ and $\partial H(\mathcal{X}_m)$ is an addressable problem with dimensions smaller than 3 (4 players). Therefore, it will be approximated for Models 3.A, 3.B and 3.C (with m from 10 to 500) to study the consistency of the proposed core estimator under this error criterion.

Table 9 shows the means of the 250 estimation errors $d_\mu(\hat{C}(N, v), C(N, v))$ obtained for the considered simulation models. Additionally, it contains the means of the 250 quotients $\mu(\hat{C}(N, v))/\mu(C(N, v))$. Since $C(N, v)$ is a convex set, it is possible to ensure that $\hat{C}(N, v) \subset C(N, v)$,

Table 9
Averages of $d_{\mu}(\hat{C}(N,v), C(N,v))$ **(D)** and $\mu(\hat{C}(N,v)/\mu(C(N,v)))$ **(Q)** for 250 estimations of the core of the considered TU games.

	m	10		25		50		100		200	
		D	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**
Model	3.A	0.09038	0.42157	0.05266	0.66295	0.03268	0.79083	0.02031	0.87004	0.01138	0.92716
	3.B	0.12779	0.42301	0.07200	0.67490	0.04462	0.79851	0.02626	0.88143	0.01580	0.92868
	3.C	0.21353	0.43058	0.12040	0.67895	0.07456	0.80118	0.04454	0.88122	0.02661	0.92904
	4.A	0.10809	0.14797	0.08349	0.34188	0.06230	0.50887	0.04432	0.65060	0.03114	0.75455
	4.B	0.01114	0.15683	0.00824	0.37643	0.00601	0.54529	0.00424	0.67874	0.00282	0.78656
	4.C	0.04390	0.14761	0.03250	0.36903	0.02410	0.53204	0.01682	0.67340	0.01118	0.78286
	5	0.00277	0.03213	0.00243	0.15168	0.00205	0.28396	0.00165	0.42458	0.00125	0.56360
	6	0.00016	0.00141	0.00018	0.01882	0.00020	0.05806	0.00019	0.11912	0.00017	0.20198
	10	≈ 0	≈ 0	≈ 0	0.00001	≈ 0	0.00019	≈ 0	0.00203	≈ 0	0.01013
	m	250		500		1000		2000		5000	
		D	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**
Model	3.A	0.00962	0.93841	0.00538	0.96555	0.00295	0.98110	0.00162	0.98963	0.00074	0.99529
	3.B	0.01282	0.94212	0.00726	0.96721	0.00418	0.98111	0.00232	0.98952	0.00108	0.99513
	3.C	0.02152	0.94263	0.01248	0.96673	0.00697	0.98140	0.00368	0.99019	0.00167	0.99556
	4.A	0.02777	0.78106	0.01839	0.85504	0.01223	0.90362	0.00793	0.93746	0.00433	0.96589
	4.B	0.00246	0.81402	0.00159	0.87968	0.00100	0.92394	0.00062	0.95275	0.00032	0.97577
	4.C	0.00989	0.80802	0.00628	0.87814	0.00398	0.92273	0.00244	0.95257	0.00126	0.97560
	5	0.00113	0.60476	0.00082	0.71385	0.00058	0.79921	0.00039	0.86435	0.00023	0.92123
	6	0.00016	0.23162	0.00014	0.33865	0.00012	0.44427	0.00010	0.55019	0.00007	0.67444
	10	≈ 0	0.01531	≈ 0	0.04398	≈ 0	0.05201	≈ 0	0.08674	≈ 0	0.17511

Table 10
Standard deviations of $d_{\mu}(\hat{C}(N,v), C(N,v))$ **(D)** and $\mu(\hat{C}(N,v))/\mu(C(N,v))$ **(Q)** for 250 estimations of the core of the considered TU games.

	m	10		25		50		100		200	
		D	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**
Model	3.A	0.01698	0.10867	0.01281	0.08196	0.00849	0.05434	0.00566	0.03625	0.00316	0.02022
	3.B	0.02333	0.10535	0.01682	0.07597	0.01089	0.04915	0.00690	0.03117	0.00369	0.01665
	3.C	0.04252	0.11340	0.03032	0.08086	0.02042	0.05445	0.01111	0.02963	0.00736	0.01963
	4.A	0.00716	0.05647	0.00800	0.06308	0.00682	0.05373	0.00503	0.03964	0.00344	0.02714
	4.B	0.00070	0.05310	0.00089	0.06704	0.00075	0.05681	0.00054	0.04097	0.00036	0.02756
	4.C	0.00253	0.04910	0.00320	0.06210	0.00284	0.05520	0.00195	0.03795	0.00121	0.02343
	5	0.00005	0.01741	0.00011	0.03787	0.00013	0.04436	0.00010	0.03654	0.00009	0.03307
	6	0.00009	0.00151	0.00007	0.01093	≈ 0	0.01543	≈ 0	0.02158	0.00001	0.02436
	10	≈ 0	≈ 0	≈ 0	0.00001	≈ 0	0.00011	≈ 0	0.00067	≈ 0	0.00202
	m	250		500		1000		2000		5000	
		D	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**	**D**	**Q**
Model	3.A	0.00277	0.01771	0.00149	0.00953	0.00080	0.00509	0.00042	0.00270	0.00019	0.00122
	3.B	0.00329	0.01486	0.00180	0.00814	0.00102	0.00462	0.00047	0.00213	0.00024	0.00109
	3.C	0.00583	0.01554	0.00306	0.00816	0.00165	0.00441	0.00087	0.00233	0.00043	0.00115
	4.A	0.00308	0.02426	0.00205	0.01614	0.00124	0.00980	0.00085	0.00669	0.00044	0.00345
	4.B	0.00031	0.02328	0.00019	0.01441	0.00013	0.00948	0.00007	0.00568	0.00003	0.00256
	4.C	0.00128	0.02494	0.00077	0.01497	0.00048	0.00940	0.00027	0.00526	0.00013	0.00252
	5	0.00008	0.02826	0.00010	0.03398	0.00006	0.02076	0.00005	0.01642	0.00003	0.01086
	6	≈ 0	0.02572	0.00001	0.02268	≈ 0	0.02027	≈ 0	0.01753	≈ 0	0.01210
	10	≈ 0	0.00280	≈ 0	0.00179	≈ 0	0.00232	≈ 0	0.00232	≈ 0	0.00163

almost surely (see Section 3 for more details). Therefore, the value of $\mu(\hat{C}(N,v))/\mu(C(N,v))$ is smaller than one, almost surely. Roughly speaking, the mean of these quotients allow to quantify the percentage of the Lebesgue measure of $C(N,v)$ that is underestimated $\hat{C}(N,v)$. These results show the good performance of the proposed core reconstructions as m increases, specially for games with larger amount of players involved. In particular, for Model 5, we estimate around the 80% of the Lebesgue measure of the true core when $m = 1000$ or $m = 2000$. A sample size of $m = 5000$ is necessary in this example to undertake the 90%. But, this is only the dimensionality effect of sampling. If the number of players increases, accuracy considerably decreases for the sample sizes fixed in this study (see Model 6 and Model 10).

Table 10 contains the standard deviations of the 250 estimation errors $d_{\mu}(\hat{C}(N,v), C(N,v))$ computed for the considered simulation models. It also shows the standard deviations of the 250 quotients $\mu(\hat{C}(N,v))/\mu(C(N,v))$. No large variability is observed from results obtained. Besides, when the sample size m is larger, standard deviations tend to decrease.

Table 11 shows the means and the standard deviations calculated from the 250 estimation errors $d_H(\partial\hat{C}(N,v), \partial C(N,v))$ for Models 3.A, 3.B and 3.C. According to the results exposed, the error clearly decreases as the sample size m enlarges. Therefore, the core reconstruction seems consistent also for this error measurement. As for the standard deviations, they also reduce considerably when m increases.

Up to this point, we have studied the performance of the approach based on the generation of uniform samples in $C(N,v)$. Except for Model 8, the number of vertices of the exact cores of the associated TU games is small. For this reason, the use of the specific estimation proposal in Algorithm 2 for convex TU games is not justified. However, Model 8 has a core $C(N,v)$ with 2304 vertices (it is still manageable). In this sense, recall that sampling permutations of N is equivalent to sampling vertices of the core by the convexity property that the class of sequencing games satisfies. We mainly follow Algorithm 2 to obtain these reconstructions of the core of the considered sequencing game.

In view of the results in Table 12, we can state the following. In general, the algorithm based on sampling vertices of the core (through the sampling of marginal contributions) always improves the results, both with the criterion of $d_{\mu}(\hat{C}(N,v), C(N,v))$ and with the volume ratio criterion $\mu(\hat{C}(N,v))/\mu(C(N,v))$. This example clearly reflects the

Table 11
Averages (A) and standard deviations (SD) of $d_H(\partial\hat{C}(N,v),\partial C(N,v))$ for 250 estimations of the core of Models 3.A, 3.B and 3.C.

Model	m	10	25	50	100	200	250	500
3.A	A	380.9152	264.8531	190.6026	141.7226	98.8281	91.1785	82.3688
	SD	145.7894	95.1708	78.2313	56.6666	32.3702	27.8250	22.0149
3.B	A	2.0514	1.2986	0.9267	0.6667	0.4751	0.4700	0.4788
	SD	0.6982	0.4307	0.2940	0.2152	0.1318	0.1119	0.1091
3.C	A	59.3007	38.4014	27.4604	20.9127	14.5224	12.8129	10.9830
	SD	17.4189	12.5341	9.2306	7.2480	4.6874	3.9456	2.7506

Table 12
Averages of $d_\mu(\hat{C}(N,v),C(N,v))$ **(D)** and $\mu(\hat{C}(N,v))/\mu(C(N,v))$ **(Q)** for 250 estimations of the core of $C(N,v)$ in Model 8 under uniform sampling (HR) and under a sample of vertices (V).

m	**D**		**Q**	
	HR	V	HR	V
25	$3.984 \cdot 10^{-6}$	$3.833 \cdot 10^{-6}$	0.00249	0.04029
50	$3.942 \cdot 10^{-6}$	$3.280 \cdot 10^{-6}$	0.01307	0.17881
100	$3.842 \cdot 10^{-6}$	$2.341 \cdot 10^{-6}$	0.03817	0.41398
200	$3.665 \cdot 10^{-6}$	$1.324 \cdot 10^{-6}$	0.08228	0.66857
250	$3.590 \cdot 10^{-6}$	$1.042 \cdot 10^{-6}$	0.10120	0.73916
500	$3.321 \cdot 10^{-6}$	$4.075 \cdot 10^{-7}$	0.16842	0.89798
1000	$2.973 \cdot 10^{-6}$	$1.118 \cdot 10^{-7}$	0.25572	0.97202
2000	$2.605 \cdot 10^{-6}$	$1.970 \cdot 10^{-8}$	0.34788	0.99507

Table 13
Exact core-centers $cc(v)$ the TU games associated to Models 3.B, 4.C and 5.

Model	$cc(v)$
3.B	$(1.12290, 4.08473, 4.07348)$
4.C	$(0.97667, 1.90220, 3.28068, 3.84044)$
5	$(120.9932, 206.8919, 67.2481, 61.0542, 143.8126)$

Table 14
Average absolute errors per component in 250 estimations of the core-center $cc(v)$ in Model 3.B.

Model 3.B			
m	Player 1	Player 2	Player 3
100	0.02449	0.09600	0.09794
1000	0.00378	0.01289	0.01337

problem that arises when increasing the dimension. From the simulation results, this effect is not observed when using the specific procedure for convex games as the results improve radically. See the case of a sample size $m = 1000$, where we obtain estimations that occupy, on average, 97.2% of the overall volume compared to 25.58% provided by the more general method. Therefore, a natural extension of this work arises: generating samples close to the boundary of the polyhedron is enough to estimate the core. However, it should be noted that the convex hull of samples of vertices can just be used as an estimator of the core of a TU game when it satisfies convexity. Otherwise, the mechanism required for reconstructing the core of a general TU game is the one detailed in Algorithm 1.

Finally, we mention that the results included along this section and in the following one have been obtained by processing our algorithms in R software, on a PC with Intel(R) Core(TM) i9-9980HK and 32 GB of memory and with a single 2.40 GHz CPU processor. We list those R packages that have been used. The R package `CoopGame`[8] has been required to determinate the exact core of the simulation models. The R library `hitandrun`[9] can be considered for generating uniform samples. The R package `geometry` has been used for calculating the convex hulls. Note that this library allows to the convex hull computation in general dimension. The final R code used can be obtained from the authors upon request.

5. A geometrical application: Estimating the core-center

Geometry becomes fundamental for motivating some allocation rules proposed for TU games in literature. As González-Díaz and Sánchez-Rodríguez (2007) commented, the Shapley value for a TU game is considered as the gravity center of its marginal contributions and, if it exists, the core-center $cc(v)$ can be interpreted as the real center of the core $C(N,v)$.

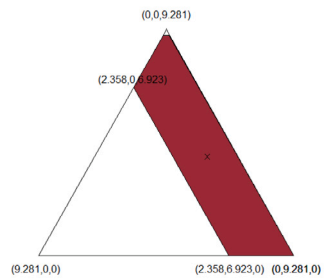
As we mentioned along the introductory section of this paper, sampling methodologies for estimating point-valued solutions in Cooperative Game Theory have been already used in literature. As an application of the methodology in Section 3, we also introduce a novel sampling-based algorithm for estimating the core-center $cc(v)$ of a TU game. Since $cc(v)$ coincides with the barycenter of the convex and compact polyhedron in $\mathbb{R}^{|N|-1}$ that determines the non-empty core of the TU game, a natural estimator $\hat{cc}_H(v)$ of this allocation rule emerges by considering the centroid of the core estimation $\hat{C}(N,v)$ established in Algorithm 1. Formally, $\hat{cc}_H(v)$ can be computed (in any dimension) as the center of gravity of $\hat{C}(N,v) = H(\mathcal{X}_m)$. In Appendix, our R code implementation of the function that calculates the barycenter of any polyhedron in general dimension is provided. Mainly, it depends on the function `delaunayn()` and `convhulln()` developed in the R package `geometry`. As far as we know, this function is not available in any library on the R CRAN.[10]

Fig. 5(a) and (b) show the exact core (black colour) of (N,v) for Models 3.B and 4.C presented in Section 4. Additionally, Fig. 5(c) and (d) depicts the corresponding representations of 250 estimations $\hat{cc}_H(v)$ when $m = 100$. For the case of $m = 1000$, the representations of $\hat{cc}_H(v)$ are respectively included in (e) and in (f). In view of these graphical results, we check that the fact of increasing the sample size tends to reduce the variability of the estimations considerably.

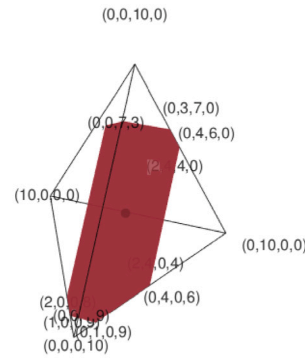
The performance of our proposal of estimator for the core-center is analysed through a small simulation study taking Model 3.B, Model 4.C and Model 5 in Section 4 as models. The core-center for the associated TU games are exactly given in Table 13. For each of the considered models, we reevaluate the 250 samples considered in Section 4 to compute $\hat{cc}_H(v)$. Since the only purpose is to illustrate how they perform, we only use as sampling sizes $m = 100$ and $m = 1000$.

Tables 14–16 contain the average of absolute estimation errors per component for Models 3.B, 4.C and 5, respectively. Results obtained show that the estimator introduced in this work, offers very positive results for all considered scenarios. In fact, the incurred absolute error per component considerably reduces when increasing the size of the sample when estimating the core-center as the centroid of the convex hull. In this sense, our proposal is very competitive from a computational perspective.

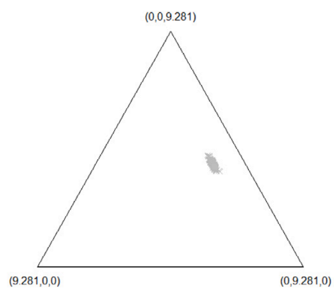
[8] More information available on: https://cran.r-project.org/package=CoopGame.
[9] More information available on: https://CRAN.R-project.org/package=hitandrun.
[10] More information available on: https://cran.r-project.org/.



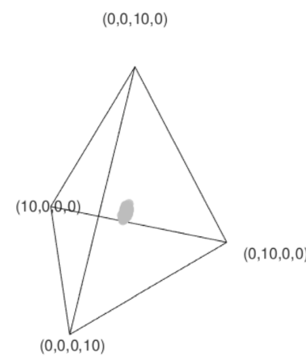
(a) $C(N,v)$ and $cc(v)$ for the TU game associated to Model 3.B.



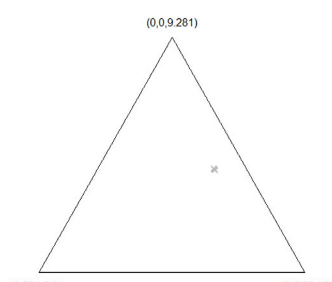
(b) $C(N,v)$ and $cc(v)$ for the TU game associated to Model 4.C.



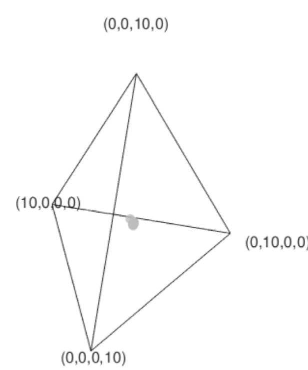
(c) 250 estimations $\hat{cc}_H(v)$ of $cc(v)$ when $m = 100$ for Model 3.B.



(d) 250 estimations $\hat{cc}_H(v)$ of $cc(v)$ when $m = 100$ for Model 4.C.



(e) 250 estimations $\hat{cc}_H(v)$ of $cc(v)$ when $m = 1000$ for Model 3.B.



(f) 250 estimations $\hat{cc}_H(v)$ of $cc(v)$ when $m = 1000$ for Model 4.C.

Fig. 5. Graphical representation of $\hat{cc}_H(v)$ for Models 3.B (left) and 4.C (right).

To complete the comparison, Tables 17–19 contain the estimated bias and variance of $\hat{cc}_H(v)$ for Models 3.B, 4.C and 5, respectively. Results obtained do not contradict the consistency of the estimator for the core-center. Furthermore, it is important to mention that $\hat{cc}_H(v)$ presents an adequate performance in terms of bias and variance. Again, when the sample size is larger, $\hat{cc}_H(v)$ shows remarkable reductions of these two error measurements.

Results obtained from simulations allows to confirm that the estimator of the core-center introduced in this work is not affected by the Hughes effect of sampling as usual in other multidimensional settings.

6. Concluding remarks

The exact computation of the core of a TU game and its characteristic elements, as the facets or the vertices, become a challenging computational task as the amount of players increases by reaching exponential time complexity. Therefore, proposing a polynomial-time estimator of the (non-empty) core of a TU game in general dimension is the main goal of this work. The route designed can be summarised as follows: (1) Support estimation theory allows to introduce an estimator for the core of a TU game as the convex hull of a uniform sample

Table 15
Average absolute errors per component in 250 estimations of the core-center $cc(v)$ in Model 4.C.

Model 4.C				
m	Player 1	Player 2	Player 3	Player 4
100	0.03805	0.07134	0.10294	0.11646
1000	0.00971	0.01778	0.02088	0.02339

Table 16
Average absolute errors per component in 250 estimations of the core-center $cc(v)$ in Model 5.

Model 5					
m	Player 1	Player 2	Player 3	Player 4	Player 5
100	9.49104	10.35219	2.24029	3.62077	16.17735
1000	3.95465	3.47677	0.70187	1.30669	6.08828

Table 17
Bias and variance per component in 250 estimations of the core-center $cc(v)$ in Model 3.B.

Model 3.B	m	Player 1	Player 2	Player 3
Bias	100	0.0090	0.0103	−0.0193
	1000	0.0024	−0.0002	−0.0022
Variance	100	0.0010	0.0148	0.0147
	1000	0.00002	0.0003	0.0003

Table 18
Bias and variance per component in 250 estimations of the core-center $cc(v)$ in Model 4.C.

Model 4.C	m	Player 1	Player 2	Player 3	Player 4
Bias	100	0.0268	0.0559	−0.0328	−0.0498
	1000	0.0090	0.0159	−0.0111	−0.0138
Variance	100	0.0022	0.0080	0.0160	0.0216
	1000	0.0003	0.0005	0.0007	0.0010

Table 19
Bias and variance per component in 250 estimations of the core-center $cc(v)$ in Model 5.

Model 5	m	Player 1	Player 2	Player 3	Player 4	Player 5
Bias	100	9.4649	10.3300	−0.2597	−3.3578	−16.1774
	1000	3.9269	3.3023	0.1058	−1.2467	−6.0883
Variance	100	107.0646	140.6046	7.7609	18.4486	286.9150
	1000	17.4629	15.5348	3.0865	2.1265	39.2297

supported on the core. (2) The algorithm for determining the core estimator is fully described in Algorithm 1. (3) Polynomial time complexity of Algorithm 1 is established. (4) Theoretical convergence rates for the estimation error in Hausdorff metric are derived. (5) Consistency for distance in measure is checked through an extensive simulation study. (6) A specific core reconstruction for convex games is introduced also with polynomial time complexity. (7) Finally, a new estimator of the core-center of a TU game is defined as the barycenter of the core reconstruction proposed in this work. Simulations show the consistency of this approximation that is able to manage the dimensionality effect.

Further research on the established core estimator and several natural extensions need to be discussed. Additionally, the positive results of the specific core estimator introduced for convex games show that generating samples close to the boundary of the polyhedron is clearly enough to estimate the core. Therefore, research in this line could be explored. Moreover, theoretical results on the consistency of the estimator proposed for the core-center of a TU game in this work need to be proved. A comparative analysis with other well-known methodologies in the literature should be considered. Among others, we mention the case of Growing Neural Gas, used for the Pareto front approximation in high-dimensional settings in Huang, Wu, Chen, Chen, and Cao (2021). Also, a simulation study on least core estimation would be very valuable. The methodology introduced in this work could be directly applied in the reconstruction of the least core because properties of compactness and convexity are guaranteed. In particular, if $\varepsilon = 0$ core reconstruction would be addressed. In fact, the performance of our proposal could be compared to the coalition-based sampling approach presented in Yan and Procaccia (2021). Although its running time is asserted to be polynomial in $\log n$, this least core estimator does not solve the problem of core emptiness either and, in practice, the final estimator could contain points outside the true least core. Finally, this work can be seen as the first interaction between game theory and set estimation. However, we think that other contributions could arise by considering, for instance, the estimation of other allocation rules such as the Shapley value under convexity of TU games.

CRediT authorship contribution statement

Alejandro Saavedra-Nieves: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing. **Paula Saavedra-Nieves:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Funding

This work has been supported under grants PID2021-124030NBC32, PID2020-118101GB-I00 and PID2020-116587GB-I00, and under grant GRC ED431C 2021/24, funded by Consellería de Cultura, Educación e Universidades, Spain, Xunta de Galicia, Spain.

Authors also acknowledge Professors Justo Puerto, Alberto Rodríguez-Casal and Rosa M. Crujeiras for their insightful comments on this paper and the referees for the constructive suggestions which have improved the manuscript.

Appendix. R code for obtaining of the centroid of a convex polyhedron

The procedure in R software (R Core Team, 2024) required to estimate the core-center of any TU game is based in the obtaining of the centroid of a convex polyhedron. In particular, we have implemented a function that computes this point in any dimension. Specifically, the function determines the centroid by partitioning into simplices and computing the weighted sum of their centroids.

```
centroid<-function(P){
    T<-delaunayn(P)
    n<-dim(T)[1]
    W=numeric(n)
    C=0
    for (m in 1:n){
        sp=P[T[m,],]
        W[m]=convhulln(sp,output.options = TRUE)$vol
        C = C + W[m] * apply(sp,2,mean)
    }
    return(C/sum(W))
}
```

Note that the only input argument that the user has to provide is `P`, a matrix containing the vertices of the convex polyhedron for which its centroid is required. In dimension D, this matrix has D columns and as many rows as the number vertices of the polyhedron. Besides, it is important to mention that the usage of `centroid()` function in R software depends on the functions `delaunay()` and `convhulln()` developed in the `geometry` library.

References

Algaba, E., Fragnelli, V., Llorca, N., & Sánchez-Soriano, J. (2019). Horizontal cooperation in a multimodal public transport system: The profit allocation problem. *European Journal of Operational Research*, *275*(2), 659–665.

Aumann, R. J., & Dreze, J. H. (1974). Cooperative games with coalition structures. *International Journal of Game Theory*, *3*(4), 217–237.

Avis, D., & Fukuda, K. (1992). A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete & Computational Geometry*, *8*(3), 295–313.

Bachrach, Y., Markakis, E., Resnick, E., Procaccia, A. D., Rosenschein, J. S., & Saberi, A. (2010). Approximating power indices: theoretical and empirical analysis. *Autonomous Agents and Multi-Agent Systems*, *20*(2), 105–122.

Benati, S., López-Blázquez, F., & Puerto, J. (2019). A stochastic approach to approximate values in cooperative games. *European Journal of Operational Research*, *279*(1), 93–106.

Bondareva, O. N. (1963). Some applications of linear programming methods to the theory of cooperative games. *Problemy Kibernetiki*, *10*, 119–139.

Castro, J., Gómez, D., & Tejada, J. (2009). Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research*, *36*(5), 1726–1730.

Chalkiadakis, G., Elkind, E., & Wooldridge, M. (2011). Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *5*(6), 1–168.

Chazelle, B. (1993). An optimal convex hull algorithm in any fixed dimension. *Discrete & Computational Geometry*, *10*(4), 377–409.

Cuevas, A., & Fraiman, R. (2010). Set estimation. *New perspectives in stochastic geometry*, 374–397.

Derks, J., & Gilles, R. P. (1995). Hierarchical organization structures and constraints on coalition formation. *International Journal of Game Theory*, *24*(2), 147–163.

Derks, J., & Kuipers, J. (2002). On the number of extreme points of the core of a transferable utility game. In *Chapters in game theory* (pp. 83–97). Springer.

Drechsel, J., & Kimms, A. (2010). Computing core allocations in cooperative games with an application to cooperative procurement. *International Journal of Production Economics*, *128*(1), 310–321.

Dümbgen, L., & Walther, G. (1996). Rates of convergence for random approximations of convex sets. *Advances in Applied Probability*, *28*(2), 384–393.

Dyer, M., Frieze, A., & Kannan, R. (1991). A random polynomial-time algorithm for approximating the volume of convex bodies. *Journal of the ACM*, *38*(1), 1–17.

Dyer, M., Frieze, A., & Stougie, L. (1994). *Talk at the math. prog. symp*. Ann Arbor, Mich.

Espinoza Burgos, N. H. (2020). *Comparación de métodos exactos y aproximados para calcular el core-center del juego del aeropuerto* (Master thesis), Retrieved from http://eio.usc.es/pub/mte/descargas/ProyectosFinMaster/Proyecto_1791.pdf.

Faigle, U. (1989). Cores of games with restricted cooperation. *Zeitschrift für Operations Research*, *33*(6), 405–422.

Fernández-García, F., & Puerto-Albandoz, J. (2006). *Teoría de Juegos Multiobjetivo*. Sevilla: Imagraf Impresores SA.

Gillies, D. B. (1953). *Some theorems on n-person games* (Ph.D. thesis), Princeton University.

González-Díaz, J., García-Jurado, I., & Fiestras-Janeiro, M. G. (2010). *Graduate studies in mathematics*: *vol. 115*, *An Introductory Course on Mathematical Game Theory*. Providence: American Mathematical Society.

González-Díaz, J., Mirás-Calvo, M. Á., Quinteiro-Sandomingo, C., & Sánchez-Rodríguez, E. (2016). Airport games: the core and its center. *Mathematical Social Sciences*, *82*, 105–115.

González-Díaz, J., & Sánchez-Rodríguez, E. (2007). A natural selection from the core of a TU game: the core-center. *International Journal of Game Theory*, *36*(1), 27–46.

Goyal, T., & Kaushal, S. (2017). An intelligent scheduling scheme for real-time traffic management using cooperative game theory and AHP-TOPSIS methods for next generation telecommunication networks. *Expert Systems with Applications*, *86*, 125–134.

Grabisch, M. (2013). The core of games on ordered structures and graphs. *Annals of Operations Research*, *204*(1), 33–64.

Grabisch, M., & Sudhölter, P. (2018). On a class of vertices of the core. *Games and Economic Behavior*, *108*, 541–557.

Graham, R. L. (1972). An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, *1*, 132–133.

Guardiola, L. A., Meca, A., & Timmer, J. (2007). Cooperation and profit allocation in distribution chains. *Decision Support Systems*, *44*(1), 17–27.

Hadas, Y., Gnecco, G., & Sanguineti, M. (2017). An approach to transportation network analysis via transferable utility games. *Transportation Research, Part B (Methodological)*, *105*, 120–143.

Huang, W., Wu, M., Chen, L., Chen, X., & Cao, W. (2021). Multi-objective drilling trajectory optimization using decomposition method with minimum fuzzy entropy-based comprehensive evaluation. *Applied Soft Computing*, *107*, Article 107392.

Lovász, L. (1999). Hit-and-run mixes fast. *Mathematical Programming*, *86*(3), 443–461.

Lovász, L., & Vempala, S. (2006). Hit-and-run from a corner. *SIAM Journal on Computing*, *4*, 985–1005.

Luo, C., Zhou, X., & Lev, B. (2022). Core, Shapley value, nucleolus and Nash bargaining solution: A survey of recent developments and applications in operations management. *Omega*, Article 102638.

Mahdiraji, H. A., Razghandi, E., & Hatami-Marbini, A. (2021). Overlapping coalition formation in game theory: A state-of-the-art review. *Expert Systems with Applications*, *174*, Article 114752.

Maschler, M., Peleg, B., & Shapley, L. S. (1979). Geometric properties of the kernel, nucleolus, and related solution concepts. *Mathematics of Operations Research*, *4*(4), 303–338.

Matheron, G. (1975). Random sets and integral geometry.

Mirás-Calvo, M. Á., Quinteiro-Sandomingo, C., & Sánchez-Rodríguez, E. (2020). The boundary of the core of a balanced game: face games. *International Journal of Game Theory*, 1–21.

Mirás-Calvo, M. Á., & Sánchez-Rodríguez, E. (2006). TUGlab: A cooperative game theory toolbox. http://mmiras.webs.uvigo.es/TUGlab/.

Myerson, R. B. (1977). Values of games in partition function form. *International Journal of Game Theory*, *6*(1), 23–31.

Núñez, M., & Rafels, C. (1998). On extreme points of the core and reduced games. *Annals of Operations Research*, *84*, 121–133.

Núñez, M., & Rafels, C. (2003). Characterization of the extreme core allocations of the assignment game. *Games and Economic Behavior*, *44*(2), 311–331.

Omrani, H., Shafaat, K., & Emrouznejad, A. (2018). An integrated fuzzy clustering cooperative game data envelopment analysis model with application in hospital efficiency. *Expert Systems with Applications*, *114*, 615–628.

O'Neill, B. (1982). A problem of rights arbitration from the talmud. *Mathematical Social Sciences*, *2*(4), 345–371.

Owen, G. (1977). Values of games with a priori unions. In *Mathematical economics and game theory* (pp. 76–88). Springer.

Perea, F., & Puerto, J. (2019). A heuristic procedure for computing the nucleolus. *Computers & Operations Research*, *112*, Article 104764.

Preparata, F. P., & Hong, S. J. (1977). Convex hulls of finite sets of points in two and three dimensions. *Communications of the ACM*, *20*(2), 87–93.

R Core Team (2024). *R: A language and environment for statistical computing [computer software manual]*. Vienna, Austria: R Foundation for Statistical Computing, Retrieved from https://www.R-project.org/.

Reitzner, M., et al. (2003). Random polytopes and the Efron–Stein jackknife inequality. *The Annals of Probability*, *31*(4), 2136–2166.

Saavedra-Nieves, A. (2023). On stratified sampling for estimating coalitional values. *Annals of Operations Research*, *320*(1), 325–353.

Saavedra-Nieves, A., & Fiestras-Janeiro, M. G. (2021). Sampling methods to estimate the Banzhaf–Owen value. *Annals of Operations Research*, *301*(1), 199–223.

Saavedra-Nieves, A., & Fiestras-Janeiro, M. G. (2022). Analysis of the impact of DMUs on the overall efficiency in the event of a merger. *Expert Systems with Applications*, *195*, Article 116571.

Saavedra-Nieves, A., García-Jurado, I., & Fiestras-Janeiro, M. G. (2018). Estimation of the Owen Value based on Sampling. In E. Gil, E. Gil, J. Gil, & M. A. Gil (Eds.), *The mathematics of the uncertain: A tribute to pedro gil* (pp. 347–356). Springer.

Saavedra-Nieves, A., Schouten, J., & Borm, P. (2020). On interactive sequencing situations with exponential cost functions. *European Journal of Operational Research*, *280*(1), 78–89.

Schmeidler, D. (1969). The nucleolus of a characteristic function game. *SIAM Journal on Applied Mathematics*, *17*(6), 1163–1170.

Schneider, R. (1988). Random approximation of convex sets. *Journal of Microscopy*, *151*(3), 211–227.

Schneider, R. (2014). *Convex bodies: The Brunn–Minkowski theory (No. 151)*. Cambridge University Press.

Schouten, J., Saavedra-Nieves, A., & Fiestras-Janeiro, M. G. (2021). Sequencing situations and games with non-linear cost functions under optimal order consistency. *European Journal of Operational Research*, *294*(2), 734–745.

Seidel, R. (1981). *A convex hull algorithm optimal for point sets in even dimensions* (Ph.D. thesis), University of British Columbia.

Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, *2*(28), 307–317.

Shapley, L. S. (1967). On balanced sets and cores. *Naval Research Logistics*, *14*, 453–460.

Shapley, L. S. (1971). Cores of convex games. *International Journal of Game Theory*, *1*, 11–26.

Shapley, L. S., & Shubik, M. (1971). The assignment game I: The core. *International Journal of Game Theory*, *1*(1), 111–130.

Smith, W. (1956). Various optimizers for single-stage production. *Naval Research Logistics*, *3*, 59–66.

Smith, R. L. (1984). Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, *32*(6), 1296–1308.

Staudacher, J., & Anwander, J. (2019). Using the R package CoopGame for the analysis, solution and visualization of cooperative games with transferable utility [computer software manual]. Retrieved from https://cran.r-project.org/package=CoopGame (R Vignette).

Tijs, S. H. (2005). The first steps with Alexia, the average lexicographic value. Available at SSRN 870405.

Trudeau, C., & Vidal-Puga, J. (2017). On the set of extreme core allocations for minimal cost spanning tree problems. *Journal of Economic Theory*, *169*, 425–452.

Weber, R. J. (1988). Probabilistic values for games. In *The Shapley value. Essays in honor of Lloyd S. Shapley* (pp. 101–119).

Yan, T., & Procaccia, A. D. (2021). If you like Shapley then you'll love the core. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 35* (pp. 5751–5759).

Yang, S., Zhang, J., & Zhou, S. (2023). The cost transportation game for collaboration among transportation companies. *Annals of Operations Research*, 1–25.

Zhou, L., Lú, K., Yang, P., Wang, L., & Kong, B. (2015). An approach for overlapping and hierarchical community detection in social networks based on coalition formation game theory. *Expert Systems with Applications*, *42*(24), 9634–9646.