

Enjoying cooperative games: The R package GameTheory

Cano-Berlanga, S.
Universitat Rovira i Virgili
GRODE

Giménez-Gómez, J.M.
Universitat Rovira i Virgili
CREIP

Vilella, C.
Universitat Rovira i Virgili
CREIP

Abstract

This paper focuses on cooperative games with transferable utility. We propose the computation of two solutions, the *Shapley value* for n agents and the *nucleolus* with a maximum of four agents. The current approach is also focused on conflicting claims problems, a particular case of coalitional games. We provide the computation of the most well-known and used claims solutions: the *proportional*, the *constrained equal awards*, the *constrained equal losses*, the *Talmud* and the *random arrival* rules.

Keywords: Cooperative game, Shapley value, nucleolus, claims problem, claims rule, bankruptcy.

Col·lecció “DOCUMENTS DE TREBALL DEL DEPARTAMENT D'ECONOMIA - CREIP”

Document de treball n.06- 2015

Dipòsit Legal: T - 220 - 2015

ISSN edició en paper: 1576 - 3382

ISSN edició electrònica: 1988 - 0820

Working paper available at <http://goo.gl/fvb43h>

1. Introduction

Game theory is the discipline that studies how agents make strategic decisions. It was initially developed in economics to understand a large collection of economic behaviors, including firms, markets and consumers. Specifically, a game is the mathematical formalization of such conflicts, originated by Antoine Augustine Cournot (1801-1877) in 1838 with his solution of the Cournot duopoly.

Modern game theory begins with the publication of the book “Theory of Games and Economic Behavior” written by [Morgenstern and Von Neumann \(1953\)](#), who considered cooperative games with several players. Indeed, according to [Maschler \(1992\)](#) after this initial point, game theory was developed extensively in the 1950s by numerous authors. Later on, the application field of game theory was not unique to Economics and we may find game theory in social network formation, behavioral economics, ethical behavior and biology, among others.

Game theory is divided into two branches, called the non-cooperative and cooperative branches. Actually, in the words of Aumann (1989, pp. 8-9):

“Cooperative theory starts with a formalization of games that abstracts away altogether from procedures and [...] concentrates, instead, on the possibilities for agreement [...] There are several reasons that explain why cooperative games came to be treated separately. One is that when one does build negotiation and enforcement procedures explicitly into the model, then the results of a non-cooperative analysis depend very strongly on the precise form of the procedures, on the order of making offers and counter-offers and so on. This may be appropriate in voting situations in which precise rules of parliamentary order prevail, where a good strategist can indeed carry the day. But problems of negotiation are usually more amorphous; it is difficult to pin down just what the procedures are. More fundamentally, there is a feeling that procedures are not really all that relevant; that it is the possibilities for coalition forming, promising and threatening that are decisive, rather than whose turn it is to speak [...] Detail distracts attention from essentials. Some things are seen better from a distance; the Roman camps around Metzarda are indiscernible when one is in them, but easily visible from the top of the mountain.”

These two branches of game theory differ in how they formalize interdependence among the players. In non-cooperative game theory, a game is a detailed model of all the moves available to the players. In contrast, cooperative game theory abstracts away from this level of detail, and describes only the outcomes that result when the players come together in different combinations. This research usually centers its interest on particular sets of strategies known as “solution concepts” or “equilibria” based on what is required by norms of (ideal) rationality. Among the several types of games, this paper focuses on *cooperative games with transferable utility*.

A coalitional game with transferable utility involving a set of agents (hereinafter a coalitional game) is a cooperative game that can be described as a function that associates with each group of agent (or coalition), a real number which the worth of the coalition. If a coalition forms, then it can divide its worth in any possible way among its members. This is possible if money is available as a medium of exchange, and if each player’s utility for money is linear (see [Aumann \(1960\)](#)). A solution on coalitional games is a correspondence that associates with each game a non-empty set of payoff vectors in \mathbb{R}^N whose coordinates add up to $v(N)$. One of the most important solutions is the core and it selects for each game all the payoff vectors such that no coalition could simultaneously provide a higher payoff to each of its members. The core is a multi-valued solution but the ones we present here, the Shapley value ([Shapley 1953](#)) and the *nucleolus* ([Schmeidler 1969](#)), are single-valued. We propose the computation of the *Shapley value* for n agents and the *nucleolus* with a maximum of four agents. As noted by [Guajardo and Jörnsten \(2015\)](#) it is usual to find mistakes in computing the *nucleolus*, but our results coincide with theirs.

The current approach is also focused on a particular case of coalitional games, the *conflicting claims problems*. This model describes the situation faced by a court that has to distribute the net worth of a bankrupt firm among its creditors. But, it also corresponds with cost-sharing, taxation, or rationing problems. Given a conflicting claims problem, a rule associates within each problem a distribution of the available resources among the agents. In this sense, we provide the computation of the most well-known and used claims solutions: the *proportional*, the *constrained equal awards*, the *constrained equal losses*, the *Talmud* and the *random arrival*

rules.

Finally, the aim of this paper is to provide a toolbox which includes common solutions to cooperative games. Currently, there is no package available covering such algorithms. Lately, [Kenkel and Signorino \(2014\)](#) have developed the package **Games**, which focus on models of strategic interaction.

This paper is organized as follows. Section 2 has a methodology review for coalitional games and for the conflicting claims problem. In Section 3 we have the library structure and Section 4 provides some examples and illustrations.

2. Methodology Review

2.1. TU-cooperative games

TU-cooperative games are used to model situations where cooperation benefits the agents (in terms of profits or costs). The different solutions given for the model propose distributions of the profits obtained after cooperation. Some examples of situations where this model applies are the construction of a motorway in which different agents are involved, the neighbors who must pay costs of an elevator, common cables, antennas, etc., cooperation between countries (European Union, UN, etc.) or between political parties to form governments (governments in coalitions).

The situations where conflicts of interest arise are called games and the agents involved in the game are called players, who may be individuals, nations, political parties, companies, firms, etc. In these models we assume that players can make binding agreements about the distribution of the payoffs or the choice of strategies. In addition, players are able to compensate each other by transferring utility, for example, through a perfectly divisible good, which is usually identified with money.

A TU-game involving a set of agents $N \in \mathcal{N}$ can be described as a function v , known as the characteristic function, which associates a real number to each subset of agents, or coalition, S contained in N . Formally, for each $N \in \mathcal{N}$, a **TU-game** is a pair (N, v) , where $v : 2^N \rightarrow \mathbb{R}$. For each coalition $S \subseteq N$, $v(S)$ is commonly called its worth and denotes the quantity that agents in S can guarantee for themselves if they cooperate. Therefore, it is assumed that $v(\emptyset) = 0$. It is also often supposed that (N, v) is **superadditive**, i.e., for any pair of coalitions $S, T \subset N$ such that $S \cap T = \emptyset$, $v(S \cup T) \geq v(S) + v(T)$, so that there is incentive for the grand coalition forms. Let \mathcal{G}^N denote the family of TU-games with agents set N .

A **solution for TU-games** is a correspondence which for each $N \in \mathcal{N}$ and each $(N, v) \in \mathcal{G}^N$, selects a set of allocations of the worth of the grand coalition among the agents. If a TU-game solution consists of a unique allocation, it is called a **TU-value**. Next we introduce the *Shapely value* (plus its natural extension as a power index) and the *nucleolus*.

The **Shapley value** ([Shapley 1953](#)). To present this solution, we need to define the marginal contribution of an agent. Given $(N, v) \in \mathcal{G}^N$, for each $i \in N$ and each $S \subset N$, we call the **marginal contribution of agent i to coalition S** , denoted by $\Delta_i v(S)$, the amount which his adherence contributes to the value of the coalition, that is, $\Delta_i v(S) = v(S \cup \{i\}) - v(S)$. According to this solution the worth of the grand coalition is distributed assuming that all orders of agents' arrivals to the grand coalition are equally probable and in each order, each

agent gets his marginal contribution from the coalition that he joins. Formally, for each $(N, v) \in \mathcal{G}^N$, the **Shapley value**, γ^{Sh} , associates to each $i \in N$, the amount $\gamma_i^{Sh}(N, v) = \sum_{S \subseteq N \setminus \{i\}} [(s!(n-s-1)!)/n!] \Delta_i v(S)$.

The **Shapley and Shubik index** (Shapley and Shubik 1954). This solution proposes the specialization of the *Shapley value* to voting games that measures the real power of a coalition.¹ The Shapley and Shubik index works as follows. There is a group of individuals all willing to vote on a proposal. They vote in order and as soon as a majority has voted for the proposal, it is declared passed and the member who voted last is given credit for having passed it. Let us consider that the members are voting randomly. Then we compute the frequency with which an individual is the one that gets the credit for passing the proposal. That measures the number of times that the action of that individual joining the coalition of their predecessors makes it a winning coalition. Note that if this index reaches the value of 0, then it means that this player is a dummy. When the index reaches the value of 1, the player is a dictator.

The **nucleolus** (Schmeidler 1969). To introduce this solution, some additional notation is needed. For each $(N, v) \in \mathcal{G}^N$, $I(N, v) = \{x \in \mathbb{R}^n : \sum_{i \in N} x_i = v(N), x_i \geq v(\{i\}) \text{ for all } i \in N\}$ is the set of imputations. For each $x \in \mathbb{R}^n$ and each coalition $S \subseteq N$, $e(x, S) = v(S) - \sum_{i \in S} x_i$ is the excess of coalition S with respect to x and represents a measure of dissatisfaction of such a coalition. The vector $e(x) = \{e(x, S)\}_{S \subseteq N}$ provides the excesses of all the coalitions in reference to x . Given $x \in \mathbb{R}^n$, $\theta(x)$ is the vector that results from x by permuting coordinates in decreasing order, $\theta_1(x) \geq \theta_2(x) \geq \dots \geq \theta_n(x)$. Finally, \leq_L stands for the lexicographic order, that is, given $x, y \in \mathbb{R}^n$, $x \leq_L y$ if there is $k \in N$ such that for all $j \leq k$, $x_j = y_j$ and $x_{k+1} \leq y_{k+1}$.

The *nucleolus* looks for an individually rational distribution of the worth of the grand coalition in which the maximum dissatisfaction is minimized. Formally, for each $(N, v) \in \mathcal{G}^N$, the **nucleolus** γ^{nu} is the vector $\gamma^{nu}(N, v) = x \in I(N, v)$ such that $\theta(e(x)) \leq_L \theta(e(y))$ for all $y \in I(N, v)$. That is, the *nucleolus* selects the element in the core, if this is nonempty, that lexicographically minimizes the vector of non-increasing ordered excesses of coalitions. In order to compute this solution we consider the following linear programming model, which looks for an imputation that minimizes the maximum excess ε among all coalitions. Formally,

$$\begin{aligned} & \min_x \varepsilon \\ & \text{subject to } v(S) - \sum_{i \in S} x_i \leq \varepsilon, \quad \forall S \subset N, S \neq \emptyset \\ & \sum_{i \in N} x_i = v(N) \\ & \varepsilon \in \mathbb{R}, x_j \in \mathbb{R}; \forall j \in N \end{aligned}$$

¹Voting games are modeled by simple games. Those are cooperative games that can model various voting systems and the characteristic function is $v(S) \in \{0, 1\}$, for all coalitions $S \subseteq N$, where $v(N) = 1$ and $v(S) \leq v(T)$ if $S \subseteq T$.

2.2. The conflicting claims problem

A conflicting claims problem is a particular case of the distribution problem, in which the amount to be distributed, the *endowment* E , is not enough to satisfy the agents' claims on it. This model describes the situation faced by a court that has to distribute the net worth of a bankrupt firm among its creditors. But, it also corresponds with cost-sharing, taxation, or rationing problems. The formal analysis of situations like these, which originates in a seminal paper by O'Neill (1982), shows that a vast number of well-behaved solutions have been defined for solving conflicting claims problems, being the *proportional*, the *constrained equal awards*, the *constrained equal losses*, the *Talmud* and the *random arrival* rules the prominent concepts used.²

An illustrative example of conflicting claims problems is the fishing quotas reduction, in which the agent's claim can be understood as the previous captures, and the endowment is the new (lower) level of joint captures (Gallastegui, Iñarra, and Prellezo (2003); Iñarra and Skonhof (2008)). A similar example is given by milk quotas among European Union (EU) members.³ In both examples, *proportionality* is the main principle used. Another example of conflicting claims situations is the September 11th Victim Compensation Fund (VCF), where the income each victim would have earned in a full lifetime was estimated and the individual claim is the legal right to be compensated. Similarly, bankruptcy laws consider the claimants identity to establish a priority rule. Specifically, bankruptcy codes normally list all claims that should be treated identically in various categories and assigns to them lexicographic priorities (Kamiski 2006). Pulido, Borm, Hendrickx, Llorca, and Sánchez-Soriano (2002, 2008) analyze, under the name of bankruptcy problems with references, the real-life case of allocating a given amount of money among the various degree courses that are offered at a (public) Spanish university. The (verifiable) monetary needs of each course constitute their claims. Additionally, there exist reference values for each course, which are set by the government independently, below their claims. Other relevant practical cases also involving more complex rationing situations could be protocols for the reduction of pollution (Giménez-Gómez, Teixidó-Figueras, and Vilella 2014), water distribution in drought periods, or even some resource allocation procedures in the public health care sector, in which past consumption could be considered as an entitlement, and current needs as a claim (see, for instance, Hougaard, J., and Osterdal 2012 and Moreno-Tertero and Roemer 2012). The formalization of such problems is as follows.

Consider a set of agents $N = \{1, 2, \dots, n\}$ and amount $E \in \mathbb{R}_+$ of an infinite divisible resource, the **endowment**, that has to be allocated among them. Each agent has a **claim**, $c_i \in \mathbb{R}_+$ on it. Let $c \equiv (c_i)_{i \in N}$ be the claims vector.

A **conflicting claims problem** is a pair (E, c) with $\sum_{i=1}^n c_i > E$. Without loss of generality, we will order the agents according to their claims $c_1 \leq c_2 \leq \dots \leq c_n$ and we will denote by \mathcal{B} the set of all conflicting claims problems.

Given a conflicting claims problem, a rule associates within each problem a distribution of the endowment among the agents. A **rule** is a single-valued function $\varphi : \mathcal{B} \rightarrow \mathbb{R}_+^n$ such that $0 \leq$

²The reader is referred to Moulin (2002) and Thomson (2003, 2013) for reviews of this literature.

³Quotas were introduced in 1984. Each member state was given a reference quantity which was then allocated to individual producers. The initial quotas were not sufficiently restrictive to remedy the surplus situation and so the quotas were cut in the late 1980s and early 1990s. Quotas will end on April 1, 2015.

$\varphi_i(E, c) \leq c_i$, for all $i \in N$ (**non-negativity** and **claim-boundedness**); and $\sum_{i=1}^n \varphi_i(E, c) = E$ (**efficiency**). Those rules used throughout the present approach are introduced below.

The **proportional (P)** rule recommends a distribution of the endowment which is proportional to the claims: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $P_i(E, c) \equiv \lambda c_i$, where $\lambda = \frac{E}{\sum_{i \in N} c_i}$.

The **constrained equal awards (CEA)** rule (Maimonides, 12th century), proposes equal awards to all agents subject to no one receiving more than his claim: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $CEA_i(E, c) \equiv \min\{c_i, \mu\}$, where μ is such that $\sum_{i \in N} \min\{c_i, \mu\} = E$.

The **constrained equal losses (CEL)** rule (Maimonides, 12th century ([Aumann and Maschler 1985](#))), chooses the awards vector at which all agents incur equal losses, subject to no one receiving a negative amount: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $CEL_i(E, c) \equiv \max\{0, c_i - \mu\}$, where μ is such that $\sum_{i \in N} \max\{0, c_i - \mu\} = E$.

The **Talmud (T)** rule ([Aumann and Maschler 1985](#)) proposes to apply the constrained equal awards rule, if the endowment is not enough to satisfy the half-sum of the claims. Otherwise, each agent receives the half of his claim and the constrained equal losses rule is applied to distribute the remaining endowment: for each $(E, c) \in \mathcal{B}$, and each $i \in N$, $T_i(E, c) \equiv CEA_i(E, c/2)$ if $E \leq \sum_{i \in N} c_i/2$; or $T_i(E, c) \equiv c_i/2 + CEL_i(E - \sum_{i \in N} c_i/2, c/2)$, otherwise.

The **random arrival (RA)** rule ([O'Neill 1982](#)). Suppose that each claim is fully honored until the endowment runs out following the order of the claimants' arrival. In order to remove the unfairness of the first-come first-served scheme associated with any particular order of arrival, the rule proposes to take the average of the awards vectors calculated in this way when all orders are equally probable: for each $(E, c) \in \mathcal{B}$, and each $i \in N$, $RA_i(E, c) \equiv \frac{1}{|N|!} \sum_{\prec \in \mathbb{R}^N} \min\{c_i, \max\{E - \sum_{j \in N, j \prec i} c_j, 0\}\}$.

3. Library structure

The **GameTheory** package is designed to implement common solutions to cooperative games. In particular, we focus on *transferable utility games*, *conflicting claims problems* and *voting power index*. **GameTheory** is available in source code format.⁴ In order to install the library, download the source code in your current working directory and execute,

```
> system("R CMD build GameTheory")
> system("R CMD INSTALL GameTheory_1.0.tar.gz")
> library("GameTheory")
```

⁴The package will be available in the near future in Comprehensive R Archive Network. If you compile the code from the source, be sure to install first all dependencies mentioned in the DESCRIPTION file.

Game class	Command	Output	Max players
TU-COOPERATIVE	Nucleolus()	Numerical	4
	NucleolusCost()	Numerical	4
	ShapleyValue()	Numerical	n
CONFLICTING CLAIMS	Proportional()	Numerical	n
	CEA()	Numerical	n
	CEL()	Numerical	n
	Talmud()	Numerical	n
	RandomArrival()	Numerical	n
	AllRules()	Numerical	n
	PlotAll()	Graphical	
	LorenzRules()	Graphical	
VOTING POWER	ShapleyShubik()	Numerical	n

Table 1: Summary of available instructions.

GameTheory package depends on **lpSolveAPI** to perform linear programming optimization, **kappalab**, **combinat**, and **ineq**. The results presented in this paper have been obtained using R version 3.1.2 on a Mac running OS X 10.10.2. The main commands (a brief summary of all available instructions is displayed in Table 1) of the library are :

Nucleolus(N,coalitions): Obtains the *nucleolus* of TU-game with a maximum of 4 players. The needed arguments are number of players and values of every coalition. This command is designed to be used with a gains game. To calculate the *nucleolus* of a cost game execute **NucleolusCost()**.

ShapleyValue(coalitions,names): Performs the calculation of the *Shapley value* for an N-player TU-game. The extension to voting power index is made by **ShapleyShubik()**.

AllRules(E,claims,names): Obtains the allocations for a N-agents following the all the conflicting claims rules presented in Section 2.2. This command performs **Proportional()**, **CEA()**, **CEL()**, **Talmud()** and **RandomArrival()** simultaneously and includes the Gini Index to check inequality among them. Results can be displayed with **PlotAll()** and **LorenzRules()**.

4. Examples and Illustrations

4.1. TU-cooperative games

In order to illustrate *TU*-cooperative games we first take the example proposed by [Lemaire \(1991\)](#) where three individuals can collaborate by investing in common funds. This particular game is defined by the following function,

$$\begin{array}{lll}
v(\emptyset) = \emptyset & v(1) = 46125 & v(12) = 69187.5 \\
v(N) = 90000 & v(2) = 17437.5 & v(13) = 53812.5 \\
& v(3) = 5812.5 & v(23) = 30750
\end{array}$$

and with this data we can compute the *Shapley value* and the *nucleolus* solutions. We calculate both solutions using the commands `ShapleyValue()` and `Nucleolus()`, respectively. We proceed in the following manner,

```
> COALITIONS <- c(46125,17437.5,5812.5,69187.5,53812.5,30750,90000)
> NAMES <- c("Investor 1","Investor 2","Investor 3")
> LEMAIRESHAPLEY <- ShapleyValue(COALITIONS,NAMES)
> LEMAIRESHAPLEY
```

	Shapley Value
Investor 1	51750
Investor 2	25875
Investor 3	12375

```
> LEMAIRE <- Nucleolus(3,c(
46125, # v(1)
17437.5, # v(2)
69187.5, # v(12)
5812.5, # v(3)
53812.5, # v(13)
30750, # v(23)
90000 # v(123)
))
```

Model name: Nucleolus of a gains game

	C1	C2	C3	C4		
Minimize	0	0	0	-1		
R1	0	0	0	1	>=	0
R2	1	0	0	-1	>=	46125
R3	0	1	0	-1	>=	17437.5
R4	1	1	0	-1	>=	69187.5
R5	0	0	1	-1	>=	5812.5
R6	1	0	1	-1	>=	53812.5
R7	0	1	1	-1	>=	30750
R8	1	1	1	0	=	90000
Kind	Std	Std	Std	Std		
Type	Real	Real	Real	Real		
Upper	Inf	Inf	Inf	Inf		
Lower	0	0	0	0		

Model name: 'Nucleolus of a gains game' - run #1

Objective: Minimize(R0)

SUBMITTED

Model size: 8 constraints, 4 variables, 19 non-zeros.
Sets: 0 GUB, 0 SOS.

Using DUAL simplex for phase 1 and DUAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Found feasibility by dual simplex after 4 iter.

Optimal solution -6562.5 after 5 iter.

Excellent numeric accuracy ||*|| = 0

MEMO: lp_solve version 5.5.2.0 for 64 bit OS, with 64 bit LPSREAL variables.

In the total iteration count 5, 0 (0.0) were bound flips.

There were 2 refactorizations, 0 triggered by time and 0 by density.

... on average 2.5 major pivots per refactorization.

The largest [LUSOL v2.2.1.0] fact(B) had 18 NZ entries, 1.0x largest basis.

The constraint matrix inf-norm is 1, with a dynamic range of 1.

Time to load data was 0.010 seconds, presolve used 0.002 seconds,

... 0.002 seconds in simplex solver, in total 0.014 seconds.

[...some output omitted...]

> LEMAIRE

	v(S)	x(S)	Ei
1	46125.0	52687.50	-6562.50
2	17437.5	24468.75	-7031.25
3	5812.5	12843.75	-7031.25

Next, by analyzing costs instead of gains, we introduce cost allocation problems, usually called airport problems (Littlechild and Thompson 1977). Consider, for instance, several airlines that are jointly using an airstrip. Obviously, different airlines will have different needs for the airstrip. The larger the planes an airline flies, the longer the airstrip it needs. An airstrip that accommodates a given plane accommodates any smaller airplane at no extra cost. The airstrip is large enough to accommodate the largest plane any airline flies. How should its cost be divided among the airlines?

Note that under this illustration, several situations may be considered. For instance, consider farmers that are distributed along an irrigation drain. The farmer closest to the water gate only needs that the segment to his field would be maintained. Accordingly, the second closest farmer needs that the first two segments be maintained (the segment that goes from the water gate and the first farmer, and that segment from the first farmer to his field), and so on. The cost of maintaining a segment used by several farmers is incurred only once, independently of how many farmers use it. How should the total cost of maintaining the ditch be shared?

$$\begin{array}{llll}
v(\emptyset) = \emptyset & v(1) = 26 & v(12) = 53 & v(123) = 108 \\
v(N) = 110 & v(2) = 27 & v(13) = 81 & v(124) = 110 \\
& v(3) = 55 & v(14) = 83 & v(134) = 110 \\
& v(4) = 57 & v(23) = 82 & v(234) = 110 \\
& & v(24) = 84 & \\
& & v(34) = 110 &
\end{array}$$

In order to illustrate this, consider the following cost airport game. First we compute the *Shapley value*,

```

> COALITIONS <- c(26,27,55,57,53,81,83,82,84,110,108,110,110,110,110)
> NAMES <- c("Airline 1","Airline 2","Airline 3","Airline 4")
> ShapleyValue(COALITIONS,NAMES)
      Shapley Value
Airline 1      17.33333
Airline 2      18.00000
Airline 3      36.33333
Airline 4      38.33333

```

and now we can see what would be the imputations using the *nucleolus*,

```

NucleolusCost(4,c(
26, # v(1)
27, # v(2)
53, # v(12)
55, # v(3)
81, # v(13)
82, # v(23)
108, # v(123)
57, # v(4)
83, # v(14)
84, # v(24)
110, # v(124)
110, # v(34)
110, # v(134)
110, # v(234)
110 # v(1234)
))

```

Model name: Nucleolus of a cost game

	C1	C2	C3	C4	C5	C6		
Maximize	0	0	0	0	1	-1		
R1	0	0	0	0	0	1	>=	0
R2	1	0	0	0	1	-1	<=	26
R3	0	1	0	0	1	-1	<=	27

R4	1	1	0	0	1	-1	<=	53
R5	0	0	1	0	1	-1	<=	55
R6	1	0	1	0	1	-1	<=	81
R7	0	1	1	0	1	-1	<=	82
R8	1	1	1	0	1	-1	<=	108
R9	0	0	0	1	1	-1	<=	57
R10	1	0	0	1	1	-1	<=	83
R11	0	1	0	1	1	-1	<=	84
R12	1	1	0	1	1	-1	<=	110
R13	0	0	1	1	1	-1	<=	110
R14	1	0	1	1	1	-1	<=	110
R15	0	1	1	1	1	-1	<=	110
R16	1	1	1	1	0	0	=	110
Kind	Std	Std	Std	Std	Std	Std		
Type	Real	Real	Real	Real	Real	Real		
Upper	Inf	Inf	Inf	Inf	Inf	Inf		
Lower	0	0	0	0	0	0		

Model name: 'Nucleolus of a cost game ' - run #1

Objective: Maximize(R0)

SUBMITTED

Model size: 16 constraints, 6 variables, 61 non-zeros.
Sets: 0 GUB, 0 SOS.

Using PRIMAL simplex for phase 1 and DUAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Found feasibility by primal simplex after 5 iter.

Optimal solution 13 after 7 iter.

Excellent numeric accuracy ||*|| = 0

MEMO: lp_solve version 5.5.2.0 for 64 bit OS, with 64 bit LPSREAL variables.
In the total iteration count 7, 0 (0.0) were bound flips.
There were 2 refactorizations, 0 triggered by time and 0 by density.
... on average 3.5 major pivots per refactorization.
The largest [LUSOL v2.2.1.0] fact(B) had 59 NZ entries, 1.0x largest basis.
The constraint matrix inf-norm is 1, with a dynamic range of 1.
Time to load data was 0.021 seconds, presolve used 0.000 seconds,
... 0.002 seconds in simplex solver, in total 0.023 seconds.

[1] 13 14 42 41 13 0

Using PRIMAL simplex for phase 1 and DUAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Found feasibility by primal simplex after 5 iter.

```
Optimal solution          13.5 after          8 iter.
```

```
Excellent numeric accuracy ||*|| = 0
```

```
[...some output omitted...]
```

```

      v(S)  x(S)    Ei
1    26 13.00 13.00
2    27 13.50 13.50
3    55 40.75 14.25
4    57 42.75 14.25
```

Voting Power

During Autumn 2014 Artur Mas (member of the Democratic Party of Catalunya (CiU) and President of Catalunya) said to Oriol Junqueras (leader of the Republican Party of Catalunya (ERC)) that “alternative majorities are possible” after discussing the referendum proposal of *November 9*.⁵ To conclude our paper we analyze these words through Shapley-Shubik power index. As mentioned in section 2, this voting power index often reveals surprising power distribution that is not obvious on the surface. In order to compare the power index of CiU and ERC we use the results of the elections of 2003, 2006 and 2012, whose results are displayed in Table 2.

<i>year</i>	CiU	PSC	ERC	PP	ICV	C's	CUP
2003	46	42	23	15	9	-	-
2006	48	37	21	14	12	3	-
2012	50	20	21	19	13	9	3

Table 2: Catalan seats distribution after elections of 2003, 2006 and 2012.

Having a look to the data of 2003 it might seem that PSC might have much more power than ERC (19 less seats in the camera), and the same should apply to year 2006. After executing `ShapleyShubik()` (results displayed in Table 3) one can see there are no differences in power among ERC and PSC for the chosen years. Another interesting case is the *dummy player*, both C's (in 2006) and CUP (in 2012) parties, never become pivotal players. Furthermore, one might consider that President Mas was right as there are two more parties with the same Shapley - Shubik power index.

To perform the Shapley - Shubik power index one simply provides the number of members of each party and the minimum amount of votes needed to pass a vote. The R session to obtain magnitudes of Table 3 is as follows,

```
# 2003 Elections
> SEATS<-c(46,42,23,15,9)
```

⁵Manchón, M. (2014) “Mas a Junqueras: Other majorities are possible in the Parliament,” *Economía Digital*, September 16, 2014 [on-line].

<i>year</i>	CiU	PSC	ERC	PP	ICV	C's	CUP
2003	0.400	0.233	0.233	0.067	0.067	-	-
2006	0.400	0.233	0.233	0.067	0.067	0.000	-
2012	0.533	0.133	0.133	0.133	0.033	0.033	0.000

Table 3: Shapley - Shubik power index of the Catalan parliament.

```

> PARTIES<-c("CiU","PSC","ERC","PP","ICV")
> ShapleyShubik(68,SEATS,PARTIES)
      CiU    PSC    ERC    PP    ICV
Votes    46.000 42.000 23.000 15.0000 9.0000
Votes (%)  0.341  0.311  0.170  0.1111 0.0667
Shapley-Shubik 0.400 0.233 0.233 0.0667 0.0667

# 2006 Elections
> SEATS<-c(48,37,21,14,12,3)
> PARTIES<-c("CiU","PSC","ERC","PP","ICV","C's")
> ShapleyShubik(68,SEATS,PARTIES)
      CiU    PSC    ERC    PP    ICV    C's
Votes    48.000 37.000 21.000 14.0000 12.0000 3.0000
Votes (%)  0.356  0.274  0.156  0.1037  0.0889 0.0222
Shapley-Shubik 0.400 0.233 0.233 0.0667 0.0667 0.0000

# 2012 Elections
> SEATS<-c(50,20,21,19,13,9,3)
> PARTIES<-c("CiU","PSC","ERC","PP","ICV","C's","CUP")
> ShapleyShubik(68,SEATS,PARTIES)
      CiU    PSC    ERC    PP    ICV    C's    CUP
Votes    50.000 20.000 21.000 19.000 13.0000 9.0000 3.0000
Votes (%)  0.370  0.148  0.156  0.141  0.0963 0.0667 0.0222
Shapley-Shubik 0.533 0.133 0.133 0.133 0.0333 0.0333 0.0000

```

4.2. Conflicting claims problem

The usefulness of the mechanisms proposed in the current paper covers several contexts, as mentioned in the introduction. As an example we replicate [Gallastegui et al. \(2003\)](#). They analyze the distribution of Northern European Anglerfish Fishery quotas among EU countries in terms of the allocations recommended by different solutions and how this may affect the sustainable growth of the fishing catches. Specifically, they consider seven countries (France, Spain, U.K., Ireland, Belgium, Netherlands and Germany). Each country has a claim, which depends on its historical fishing catches (13,952; 6,256; 4,348; 2,196; 927; 299; 158, respectively).

To replicate the study of [Gallastegui et al. \(2003\)](#) we can execute the commands one by one or use `Allrules()` to run all of them at once. To do so, we create objects containing the

	Claim	Proportional	Nucleolus	Shapley
Germany	158	76	79	74
Netherlands	299	143	145	140
Belgium	927	445	463	437
Ireland	2,196	1,054	1,098	1,071
UK	4348	2,086	2,174	2,147
Spain	6,256	3,002	3,128	3,101
France	13,952	6,694	6,408	6,530

Table 4: Fishing captures allocations. $E=13,500$; claim = average catches 1986-93. Source: Gallastegui *et al.* (2003).

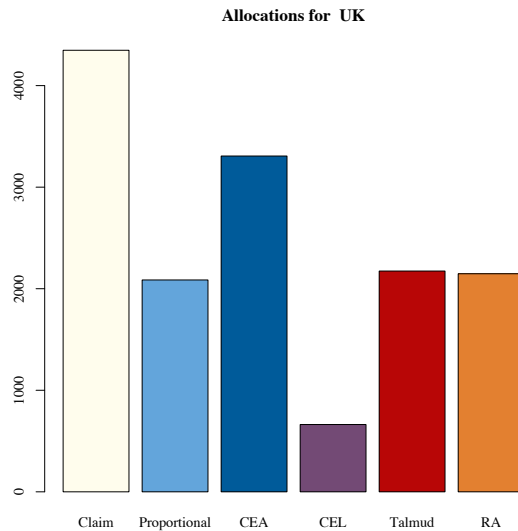


Figure 1: Fishing captures allocations for United Kingdom.

individual claims and labels of the countries. After that, running `Allrules()` is straightforward. Displaying the output of the allocations is undertaken by running `PlotAll()` (Figure 1). Graphical analysis of the inequality among rules is performed by `LorenzRules()` (Figure 2).

```
> CLAIMS <- c(158,299,927,2196,4348,6256,13952)
> COUNTRIES <- c("Germany","Netherlands","Belgium","Ireland","UK","Spain","France")

> INARRA <- AllRules(13500,CLAIMS,COUNTRIES)
> INARRA
```

	Claim	Proportional	CEA	CEL	Talmud	RA
Germany	158	75.81	158.00	0.00	79.00	73.73
Netherlands	299	143.46	299.00	0.00	149.50	139.53

Belgium	927	444.79	927.00	0.00	463.50	436.92
Ireland	2196	1053.67	2196.00	0.00	1098.00	1071.42
UK	4348	2086.22	3306.67	662.67	2174.00	2147.42
Spain	6256	3001.71	3306.67	2570.67	3128.00	3101.42
France	13952	6694.34	3306.67	10266.67	6408.00	6529.57
Gini Index	NA	0.58	0.38	0.77	0.56	0.57

```
> PlotAll(INARRA,5) ## Display allocations for UK
> LorenzRules(INARRA) ## Inequality graph
```

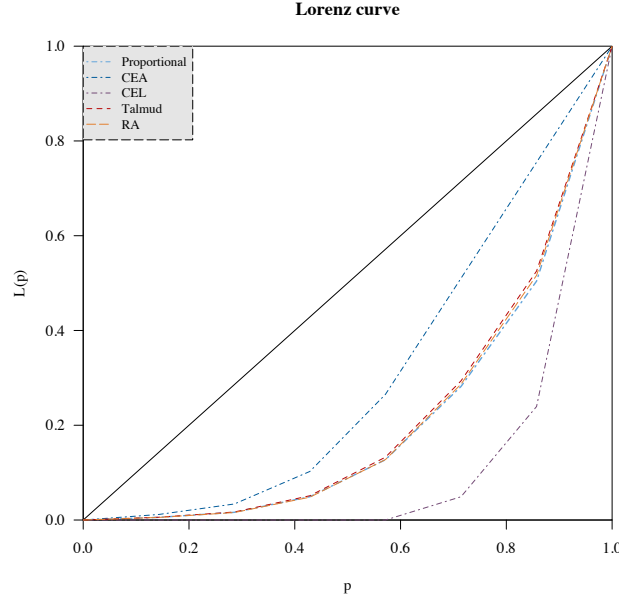


Figure 2: Inequality analysis for the different rules.

A note on the nucleolus computation

Validating the nucleolus solution might be an issue. Indeed, there is a fruitful discussion in the literature on what strategy is optimal to obtain the nucleolus, how it should be calculated, or what algorithm could achieve a better solution. For instance, [Guajardo and Jörnsten \(2015\)](#) discuss several nucleolus solutions among different papers, finding that the published results might not be optimal. Therefore, for the sake of robustness, we check our *nucleolus* commands through the correspondence between conflicting claims problems and bankruptcy games.

Following [O'Neill \(1982\)](#), a bankruptcy game is the *TU*-game associated with a conflicting claims problem. Formally, for each $(E, c) \in \mathcal{B}$, the cooperative game induced by (E, c) is the pair (N, v) , where the function $v : 2^N \rightarrow \mathbb{R}_+$ associates to each coalition $S \subseteq N$ the real

$$\text{number } v(S) = \max \left\{ 0, E - \sum_{i \in N \setminus S} c_i \right\}.$$

According to O'Neill (1982) and Aumann and Maschler (1985), for each conflicting claims problem, the *Talmud* and the *random arrival* rules coincide with the *nucleolus* and the *Shapley value* solutions of the associated bankruptcy game, respectively. A fact that we can replicate by computing the *Talmud* and the *random arrival* rules to the considered conflicting claims problem. For instance, the example of the airport cost game presented in Section 4.1, is associated to the following conflicting claims problem $(E, c) = (110, (26, 27, 55, 57))$. Hence, we can transform this particular problem into a gains game by applying the aforementioned definition of O'Neill (1982). The new game adopts the form: $v(1) = 0, v(2) = 0, v(3) = 0, v(4) = 2, v(12) = 0, v(13) = 26, v(14) = 28, v(23) = 27, v(24) = 29, v(34) = 57, v(123) = 53, v(124) = 55, v(134) = 83, v(234) = 84, v(N) = 110$. Applying `Nucleolus()` over this game coincides with the result of `Talmud(110, c(26, 27, 55, 57))`.

Acknowledgements

We are grateful to Generalitat de Catalunya under project 2014SGR631 and Ministerio de Ciencia e Innovación under project ECO2011-22765 for its support.

References

- Aumann RJ (1960). "Linearity of unrestrictedly transferable utilities." *Naval Research Logistics Quarterly*, **7**, 281–284.
- Aumann RJ, Maschler M (1985). "Game Theoretic Analysis of a bankruptcy from the Talmud." *Journal of Economic Theory*, **36**, 195–213.
- Gallastegui M, Iñarra E, Prellezo R (2003). "Bankruptcy of Fishing Resources: The Northern European Anglerfish Fishery." *Marine Resource Economics*, **17**, 291–307.
- Giménez-Gómez JM, Teixidó-Figueras J, Vilella C (2014). "The global carbon budget: a conflicting claims problem." *CREIP - Working papers*.
- Guajardo M, Jörnsten K (2015). "Common mistakes in computing the nucleolus." *European Journal of Operational Research*, **241**(3), 931–935.
- Hougaard JL, J MT, Osterdal LP (2012). "A unifying framework for the problem of adjudicating conflicting claims." *Journal of Mathematical Economics*, **48**, 107–114.
- Iñarra E, Skonhof A (2008). "Restoring a Fish Stock: A Dynamic Bankruptcy Problem." *Land Economics*, **84**(2), 327–339.
- Kamiski M (2006). "Parametric rationing methods." *Games and Economic Behavior*, **54**, 115–133.
- Kenkel B, Signorino CS (2014). "Estimating Extensive Form Games in R." *Journal of Statistical Software*, **56**.

- Lemaire J (1991). “Cooperative game theory and its insurance applications.” *Astin Bulletin*, **21**(01), 17–40.
- Littlechild SC, Thompson G (1977). “Aircraft landing fees: a game theory approach.” *The Bell Journal of Economics*, pp. 186–204.
- lpSolve, Konis K (2014). *lpSolveAPI: R Interface for lpSolve version 5.5.2.0*. R package version 5.5.2.0-12, URL <http://CRAN.R-project.org/package=lpSolveAPI>.
- Maschler M (1992). *The bargaining set, kernel and nucleolus*. In Aumann R and Hart S (Eds.), *Handbook of game theory with applications*. North-Holland, Amsterdam, pp. 591–668.
- Moreno-Ternero JD, Roemer JE (2012). “A common ground for resource and welfare egalitarianism.” *Games and Economic Behavior*, **75**, 832–841.
- Morgenstern O, Von Neumann J (1953). “Theory of games and economic behavior.”
- Moulin H (2002). *Axiomatic cost and surplus sharing*. In Arrow, A. K., and Sen, K. (eds), *Handbook of social choice and welfare*. Vol. 1. Elsevier. North Holland, Amsterdam, pp. 289–357.
- O’Neill B (1982). “A problem of rights arbitration from the Talmud.” *Mathematical Social Sciences*, **2**(4), 345–371.
- Plummer M, Best N, Cowles K, Vines K (2006). “CODA: Convergence Diagnosis and Output Analysis for MCMC.” *R News*, **6**(1), 7–11. URL <http://CRAN.R-project.org/doc/Rnews/>.
- Pulido M, Borm P, Hendrickx R, Llorca N, Sánchez-Soriano J (2002). “Game theory techniques for university management: an extended bankruptcy model.” *Annals of Operations Research*, **109**, 129–142.
- Pulido M, Borm P, Hendrickx R, Llorca N, Sánchez-Soriano J (2008). “Compromise solutions for bankruptcy situations with references.” *Annals of Operations Research*, **158**(1), 133–141.
- R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.
- Schmeidler D (1969). “The Nucleolus of a characteristic function game.” *SIAM Journal of Applied Mathematics*, **17**, 1163–1170.
- Shapley L (1953). *A value for n-person games*. In Tucker A, Kuhn H (Eds.), *Contributions to the theory of games II* (pp. 307–317). Princeton University Press: Princeton NJ.
- Shapley L, Shubik M (1954). “A Method for Evaluating the Distribution of Power in a Committee System.” *The American Political Science Review*, **48**(3), 787–792.
- Thomson W (2003). “Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey.” *Mathematical Social Sciences*, **45**(3), 249–297.
- Thomson W (2013). “Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: an update.” *mimeo*.

Warnes GR, Bolker B, Lumley T (2014). *gtools: Various R programming tools*. R package version 3.4.1, URL <http://CRAN.R-project.org/package=gtools>.

Zeileis A (2014). *ineq: Measuring Inequality, Concentration, and Poverty*. R package version 0.2-13, URL <http://CRAN.R-project.org/package=ineq>.

Affiliation:

Sebastián Cano-Berlanga
Department d'Economia
Facultat d'Economia i Empresa
Universitat Rovira i Virgili / GRODE
43204 Reus, Spain
E-mail: sebastian.cano@urv.cat

José-Manuel Giménez-Gómez
Department d'Economia
Facultat d'Economia i Empresa
Universitat Rovira i Virgili / CREIP
43204 Reus, Spain
E-mail: josemanuel.gimenez@urv.cat

Cori Vilella
Department de Gestió d'Empreses
Facultat d'Economia i Empresa
Universitat Rovira i Virgili / CREIP
43204 Reus, Spain
E-mail: cori.vilella@urv.cat