I Congresso de Estatística e Investigação Operacional da Galiza e Norte de Portugal
VII Congreso Galego de Estatística e Investigación de Operacións
Guimarães 26, 27 e 28 de Outubro de 2005

TUGlab: A Matlab-based platform for teaching TU-games theory

Miguel Ángel Mirás Calvo[1], Estela Sánchez Rodríguez[2]

[1]Departamento de Matemáticas
Universidad de Vigo
[2]Departamento de Estadística e Investigación Operativa
Universidad de Vigo

ABSTRACT

We present an interactive platform for the teaching and learning of the theory of games with transferable utility. TUGlab is a Matlab-based program that covers the general topics of an introductory course on TU games theory. The program provides a dynamic and friendly environment, enriched with animations, simulations and 3D-graphs that bring to life on the computer screen the theoretical aspects of TU games.

Key words: Cooperative TU games, Matlab applications
AMS classification: 91-04, 91-01

1. INTRODUCTION

It is always a challenge to teach an introductory course on a given topic because, on the one hand, you must capture the attention and interest of the audience with fine examples and attractive applications while, at the same time, you have to lay down the mathematical foundations that usually are technical and complex. Cooperative game theory is not an exception to this rule. Take any book aimed at teaching cooperative TU games to the newcomer, for example Rafels i Pallarola et al. (1999), Curiel (1997), and you will find plenty of examples, exercises, pictures and diagrams whose purpose is to emphasize, clarify and consolidate the basic concepts of the theory. Matlab is a high-level language and interactive environment that enables you to perform computationally intensive tasks, see Mathworks (2005). The authors enjoy long personal experiences in teaching undergraduate and graduate courses on game theory and in using Matlab as a tool in education. Therefore, we thought about bringing our skills together and create a Matlab program that could serve as a helpful complement to the books and other materials used in introductory courses on cooperative game theory. The result is TUGlab (Transferable Utility Games laboratory) a Matlab-based program that follows the lines of similar programs developed in other areas such as Engineering, Science, and Mathematics.

Since our aim is to emphasize the geometrical aspects of cooperative game theory, TUGlab offers to both the instructor and the student a tool to compute and visualize basic concepts for any given 3 or 4 persons TU games. It allows the user to experiment at will with games without worrying about the mathematical complexity of the computations. That is the power of this platform: its direct and flexible way of going to the heart of the concepts overcoming the mathematical complexity. The platform is easy to use and no previous knowledge of Matlab is needed to work with it. It includes some tutorials, help menus, several animations and a library with many interesting examples of well-known games. Among the topics cover by TUGlab we can highlight: checking monotonicity, convexity or balancedness of a game; computing value solutions as the Shapley value, the nucleolus or the τ-value; drawing the imputation set, the Weber set and the core of a game; returning the Harsanyi dividends, the multi-linear extension and the normalization of any given game,...

This paper is organized as follows. In Section 2, we present the TUGlab platform with a brief description of its commands and the cooperative game theory topics that covers. We include, in Section 3, a couple of examples to illustrate how the platform can be used to unveil some beautiful geometrical aspects of the theory: knowing if a game is convex by looking at the picture of the core and presenting games where some marginal worth vectors are not extreme points of the Weber set. Finally, Section 3 shows how TUGlab can go beyond its basic goal and be used to study some advanced topics.

2. THE TUGlab PLATFORM

The TUGlab platform is a package of Matlab files (extension .m) that define several procedures to deal with TU games. It works on any implementation of the later releases of the Matlab product: Matlab 6 and Matlab 7 on Unix, PC or Macintosh. The procedures can be run directly from the Matlab Command Window if the user is acquainted with the program. Nevertheless, the platform provides a GUI (Graphic User Interface) friendly to use and that requires no knowledge of Matlab. In this section, we include a list of the most interesting commands defined in the TUGlab platform and a brief description of them.

1. **additivegame** Checks if a TU game is additive.
2. **admissiblegame** Checks if a TU game is compromise admissible.
3. **balancedgame** Checks if a TU game is balanced.
4. **belongtocore** Checks if a given point belongs to the core of a TU game.
5. **convexgame** Checks if a TU game is convex.
6. **corecenter** Computes the corecenter of a TU game.
7. **corecoverset** Draws the core-cover of a 4-person compromise admissible TU game.
8. **coreset** Draws the core of a balanced TU game.
9. **dualgame** Returns the dual game of a TU game.
10. **essentialgame** Checks if a TU game is essential.
11. **harsanyidividends** Computes the Harsanyi dividends of a TU game.
12. **harsanyiset** Draws the Harsanyi set of a TU game.
13. **imputationset** Draws the imputation set of an essential non-degenerate TU game.
14. **MLExtension** Returns the multi-linear extension of a TU game.
15. **monotonicgame** Checks if a TU game is monotonic.
16. **normalizedgame** Provides both the 0 and 0-1 normalizations of a TU game.

17. **nucleolus** Returns the nucleolus of a TU game.
18. **Shapley** Computes the Shapley value and the marginal worth vectors of a TU game.
19. **superadditivegame** Checks if a TU game is superadditive.
20. **tauvalue** Computes the tau-value of a TU game.
21. **totalbalancedgame** Checks if a TU game is totally balanced.
22. **utopiapayoffs** Returns the utopia payoffs of a TU game.
23. **weberset** Draws the Weber set of an essential non-degenerate TU game.

The characteristic function of the game must be introduce as a vector A=[v(1) v(2) v(3) v(12) v(13) v(23) v(123)], for 3 persons games, or A=[v(1) v(2) v(3) v(4) v(12) v(13) v(14) v(23) v(24) v(34) v(123) v(124) v(134) v(234) v(1234)], for 4 persons games. So, for example, the next commands

```
A=[0 0 0 100 200 300 400];[control,info]=convexgame(A)
```

produce the outcome

```
control = 0
info = v{123}-v{23}<v{13}-v{3}
```

which tells us the the game given by A is not convex because the inequality $v(123)-v(23) < v(13)-v(3)$ holds.

2. SOME INTERESTING EXAMPLES

We will try to illustrate with two examples how the TUGlab platform can be used to explore the geometrical aspects behind some concepts of cooperative game theory.

Let us start by recalling, see Shapley (1971) and Ichiishi (1981), that a TU game is convex if and only if its marginal vectors are core elements. Now, take the 3 persons games given by vectors $A = [0, 0, 0, 0.2, 0.2, 0.2, 1]$ and $A = [0, 0, 0, 0.55, 0.55, 0.55, 1]$. One can check that only the first of these two games is convex (and, as a consequence, superadditive and balanced). Drawing the corresponding cores, see Figure 1, it is clear that a game is convex if its core "touches" the 3 sides of the imputations triangle. This geometrical interpretation does

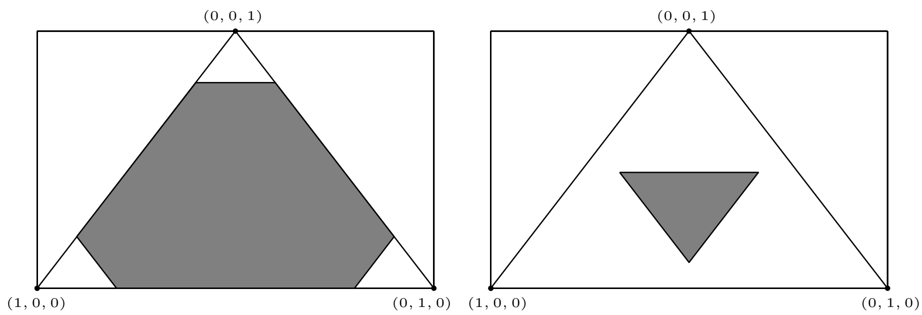

Figure 1: Cores of convex and non-convex 3 persons games

not hold for 4 persons games. Indeed, the games $A = [0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 6, 5, 5, 5, 10]$

and $B = [0,0,0,0,0,0,0,0,0,0,6,5,5,5,10]$ share the same imputation set and the same core, see the picture at the left in Figure 2. All the extreme points of the core belong to the faces of the imputations tetrahedron. But, game A is convex and B is not. However, if the core has a vertex that is interior to the imputation set then the game

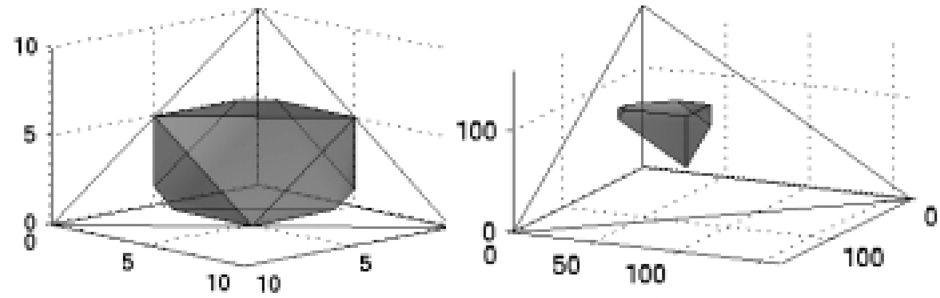

Figure 2: Core that touches all the imputation set faces and core with an interior vertex

is not convex. That is the case portrayed at the right in Figure 2 for the game $A = [0,0,0,0,10,40,30,60,10,20,90,90,120,130,160]$.

The Weber set, Weber (1988), is the polytope defined as the convex hull of the finite set of marginal worth vectors. This set might not be minimal, that is, not all the marginal vectors have to be vertices of the Weber polytope. Here, we present a couple of examples where this situation occurs, corresponding to the 3 persons game $A = [0,0,0,10,4,4,10]$ and the 4 persons game $A = [0,0,0,0,0,1,2,2,3,0,2,3,2,3,4]$ whose Weber sets are shown in Figure 3. Observe that some of the marginal worth vectors, the points marked with an asterisk, are not vertices of the Weber set. Using the command `webervertices` we can

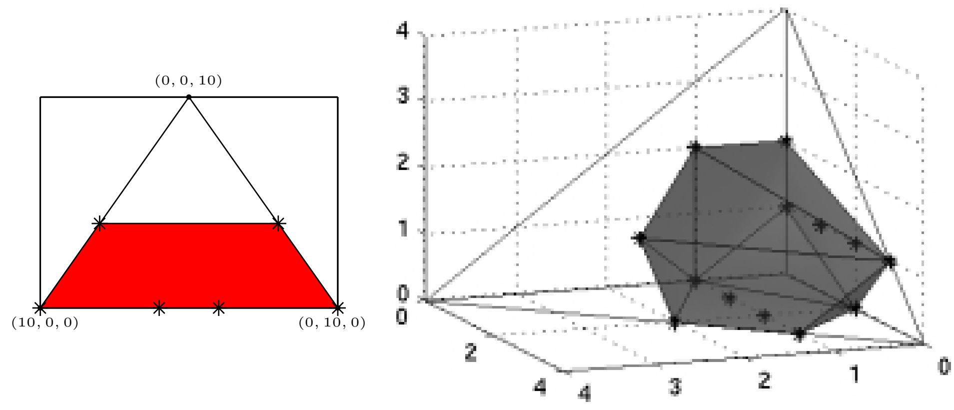

Figure 3: Weber sets with worth vectors that are not vertices

know which marginal worth vectors are extreme points of the Weber set. For the previous 3 persons game, the extreme points of the Weber set are $(6,0,4), (10,0,0), (0,10,0)$ and $(0,6,4)$ while the marginal vectors $(6,4,0)$ and $(4,6,0)$ are not extreme points of the Weber polyhedron.

3. ADVANCED TOPICS

Though TUGlab is mainly a teaching tool, we have included some specialized features that could serve as an indication of how the program can be improved to cover advanced topics. For instance, the platform computes the corecenter of a balanced game, i.e., the centroid of the core. This concept has been introduce by González-Díaz and Sánchez-Rodríguez (2003) and can be seen as the expectation of the uniform distribution over the whole core of a game. It is well known that the Shapley value of a n persons game can be interpreted as the center of mass of a finite system of exactly $n!$ particles of the same mass located at the positions indicated by the marginal vectors. The corecenter is the center of mass of the core polytope as a whole (assuming that the mass is uniformly distributed on the body). Both values behave very differently.

Using TUGlab one can easily make an animation of the following parametric game $A = [0, 0, 0, a, 5, 5, 10]$, $a \in [0, 10]$, and compare the Shapley value and the corecenter among them and both with the nucleolus. Here, we just make an sketch of the situation. The game is convex in the parameter range $[0, 5]$. The nucleolus, N, the corecenter, CC, and the Shapley value, Sh, coincide for $a = 0$. The nucleolus remains unchanged, $N = (2.5, 2.5, 5)$, for all $a \in [0, 2.5]$ while CC and Sh varied. The three values converge to $(\frac{10}{3}, \frac{10}{3}, \frac{10}{3})$ when the parameter a moves from 2.5 to 5. The game is not convex in the parameter range $[5, 10]$, the core is a triangle that shrinks to the point $(5, 5, 0)$ as a approaches 10, so CC and N coincide and converge to that point. But, the Weber set grows so S moves away from the other two values. The next table presents the values obtained from TUGlab for some choices of the parameter a.

a	N	CC	Sh
0	$(2.5, 2.5, 5)$	$(2.5, 2.5, 5)$	$(2.5, 2.5, 5)$
1	$(2.5, 2.5, 5)$	$(2.5442, 2.5442, 4.9116)$	$(2.6667, 2.6667, 4.6667)$
2.5	$(2.5, 2.5, 5)$	$(2.7381, 2.7381, 4.5238)$	$(2.9167, 2.9167, 4.1667)$
4	$(3, 3, 4)$	$(3.0490, 3.0490, 3.9020)$	$(3.1667, 3.1667, 3.6667)$
5	$(3.3333, 3.3333, 3.3333)$	$(3.3333, 3.3333, 3.3333)$	$(3.3333, 3.3333, 3.3333)$
7	$(4, 4, 2)$	$(4, 4, 4)$	$(3.6667, 3.6667, 2.6667)$
10	$(5, 5, 0)$	$(5, 5, 0)$	$(4.1667, 4.1667, 1.6667)$

We have included among the TUGlab capabilities that of drawing the Harsanyi set of a game, also called the selectope. The Harsanyi set is the set of payoff vectors obtained by all possible distributions of the Harsanyi dividends of a coalition amongst its members, see Vasil´ev and van der Laan (2002). The Harsanyi set is related to the core and the Weber set as it is illustrated in the following example. Let us consider the game $A = [1, 1, 2, 1, 2, 4, 10]$, see Figure 4. This game is not superadditive, the Weber set (red) is not a subset of the imputation set but contains the core (grey) and is contained in the Harsanyi set (blue). Other features related to the Harsanyi set already implemented in the TUGlab platform are, for example, checking if a game A is almost positive or computing the mingame associated with a game A whose core coincides with the Harsanyi set of A.

4. ACKNOWLEDGEMENTS

Please, report any bugs or suggestions about TUGlab to mmiras@uvigo.es. All comments would be welcome.

Financially supported by the Xunta de Galicia, grant PGIDT00PXI30001PN.

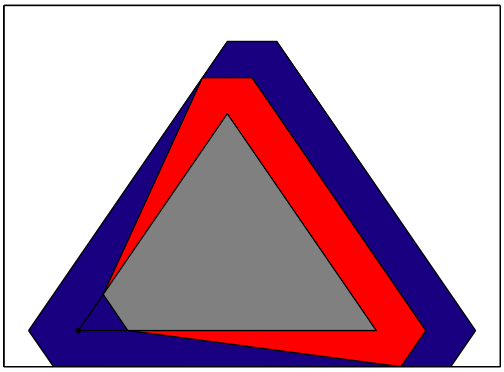

Figure 4: The Harsanyi set, the core and the Weber set

References

Curiel, I., 1997. Cooperative game theory and applications. Kluwer Academic Publishers, Dordrecht. The Netherlands.

González-Díaz, J., Sánchez-Rodríguez, E., 2003. From set-valued solutions to single-valued solutions: the centroid and the core-center. Working Paper 03-09. Reports in Statistics and Operations Research, Universidad de Santiago de Compostela.

Ichiishi, T., 1981. Supermodularity: applications to convex games and to the greedy algorithm for LP. Journal of Economic Theory 25, 283–286.

Mathworks, 2005. http://www.mathworks.com.

Rafels i Pallarola, C., Izquierdo i Aznar, J. M., Marín Solano, J., Martínez de Albéniz Salas, F. J., Núñez Oliva, M., Ybern Carballo, N., 1999. Jocs cooperatius i aplicacions econònomiques. Edicions de la Universitat de Barcelona, Barcelona.

Shapley, L., 1971. Cores of convex games. International Journal of Game Theory 1 (3), 11–26.

Vasil´ev, V., van der Laan, G., 2002. The Harsanyi set for cooperative TU-games. Siberian Advances in Mathematics 12, 97–125.

Weber, R. J., 1988. Probabilistic values for games. In: Roth, A. E. (Ed.), The Shapley value. Essays in honor of L. S. Shapley. Cambridge University Press, Cambridge, pp. 101–119.