

Programas informáticos orientados a juegos TU

David Mirás Calvo

Programa de doctorado: "Estadística e investigación de
operaciones"

Septiembre 2008

Aplicaciones informáticas orientadas a juegos TU

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

- <http://www.gametheory.net/applets/solvers.html> (9 "applets" para no-cooperativos)
- <http://www.gametheorysociety.org/resources.html> (GAMBIT, lrs home page)
- <http://www.econ.usu.edu/acaplan/tugames1.html> (Juegos TU de 3 jugadores)
- <http://www.math.unimaas.nl/PERSONAL/jeand/home1.htm> (Jean Derks pre-nucleolo)
- <http://eio.usc.es/pub/io/xogos> (TUGlab)

TUGlab vs. TUGlabExtended

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

	TUGlab	TUGlabExtended
Jugadores	3 y 4	Arbitrario (25)
Funciones	31 (27)	16 (8)
Gráficas	Sí	No
Orientación	Docencia	Investigación

TUGlab y TUGlabExtended están programados en MATLAB[®].
Información y descarga en la página web del grupo

SaGaTh (Santiago Game Theory Group).

Función característica

Un juego cooperativo con utilidad transferible, o juego TU, es un par (N, v) , en donde $N = \{1, \dots, n\}$ es el conjunto finito de jugadores y v es la función característica

$$v: 2^N \rightarrow \mathbb{R}$$

con $v(\emptyset) = 0$. Cada subconjunto de N , $S \subset N$, se denomina coalición.

Consecuentemente, para dar la función característica de un juego de n jugadores necesitamos $2^n - 1$ valores

Codificación binaria de las coaliciones

Orden lexicográfico de las coaliciones

{1} {2} {3} {1,2} {1,3} {2,3} {1,2,3}

Orden binario de las coaliciones

natural	binario	coalicón
1	100	{1}
2	010	{2}
3	110	{1,2}
4	001	{3}
5	101	{1,3}
6	011	{2,3}
7	111	{1,2,3}

Operaciones conjuntistas

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

Jugador	S	T	$S \cup T \sim \text{OR}$	$S \cap T \sim \text{AND}$
1	0	1	1	0
2	1	1	1	1
3	1	1	1	1
4	1	0	1	0
5	0	1	1	0
	14	23	31	6

Ejemplo de código

```
function [respuesta,info]=superadditiveE(v)
% la sintaxis de la función en MATLAB

[n,nC]=characteristicdata(v);
% Calculamos el número de jugadores y de coaliciones.
for ii=1:nC
% Para toda coalición S (el natural ii)
    for jj=1:nC
% Para toda coalición T (el natural jj)
        if bitand(ii,jj)==0
% Si S y T son disjuntos.
            if v(bitior(ii,jj)) < (v(ii)+v(jj))
% Si  $v(S \cup T) < v(S) + v(T)$ 
                respuesta=0;info=[ii jj];return
% No se verifica la condición para las coaliciones S y T.
            end
        end
    end
end
respuesta=1;info=[];
% En caso contrario el juego es superaditivo.
```

Funciones de TUGlabExtended

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

Principales

<i>allmarginalsE</i>	<i>belong2coreE</i>	<i>convexE</i>
<i>dualE</i>	<i>essentialE</i>	<i>harsanyidividendsE</i>
<i>imputationverticesE</i>	<i>marginalvectorE</i>	<i>monotonicE</i>
<i>normalizedE</i>	<i>shapleyE</i>	<i>superadditiveE</i>
<i>tauvalueE</i>	<i>unanimityE</i>	<i>utopiapayoffsE</i>

Auxiliares

<i>bin2lex.m</i>	<i>characteristicfunction</i>	<i>characteristicdata</i>
<i>getcoalition</i>	<i>getcoalitionnumber</i>	<i>getplayers</i>
<i>lex2bin</i>	<i>permutationE</i>	<i>shapleymatrix</i>

Ejemplo

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Ejemplo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8 8] 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Ejemplo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Ejemplo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8] 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Ejemplo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8] 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Ejemplo

```
>> characteristicfunction
```

```
Number of players: 5
```

```
ans =
```

```
1-1 2-2 3-12 4-3 5-13 6-23 7-123 8-4 9-14 10-24 11-124
```

```
12-34 13-134 14-234 15-1234 16-5 17-15 18-25 19-125
```

```
20-35 21-135 22-235 23-1235 24-45 25-145 26-245
```

```
27-1245 28-345 29-1345 30-2345 31-12345
```

```
>> v=[0 0 5 0 5 5 8 0 6 6 8 6 8 8] 10 0 5 5 5 5 8 8
```

```
10 6 8 8 12 8 5 12 20];
```

```
>> sh=shapleyE(v)
```

```
sh =
```

```
3.3000 5.0500 3.5500 4.3000 3.8000
```

Tiempos de ejecución

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

Tiempos de ejecución (en segundos) de las funciones:
shapleymatrix, *convexE* y *tauvalueE*.

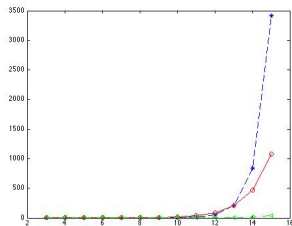
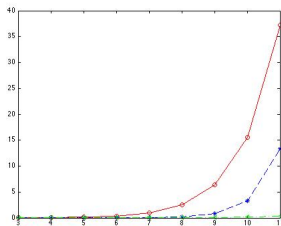
Jugadores	Shapley	τ -valor	Convexidad
3	0.016912	0.004057	0.000732
4	0.050173	0.001219	0.000766
5	0.141409	0.002493	0.002999
6	0.385369	0.005456	0.012129
7	1.012571	0.012438	0.049606
8	2.579563	0.028435	0.201198
9	6.418109	0.064519	0.801556
10	15.546376	0.145796	3.294879
11	37.162233	0.325364	13.314801
12	87.538653	0.842787	52.558724
13	203.704998	3.903356	209.971761
14	471.142136	12.000016	837.683950
15	1080.910514	37.075949	3419.695234

Gráficas de tiempos de ejecución

Programas
informáticos
orientados a
juegos TU

David Mirás
Calvo

Gráficas de los tiempos de ejecución (en segundos) de las funciones: *shapleymatrix* (en rojo), *convexE* (en azul) y *tauvalueE* (en verde).



Representación matricial del valor de Shapley

El valor de Shapley de un juego (N, v) asigna a cada $i \in N$,

$$\begin{aligned} Sh_i(N, v) &= \frac{1}{n!} \sum_{S \subset N: i \in S} (s-1)!(n-s)!(v(S) - v(S \setminus \{i\})) \\ &= \sum_{\emptyset \neq S \subset N} a(i, S)v(S) \end{aligned}$$

donde

$$a(i, S) = \begin{cases} \frac{(s-1)!(n-s)!}{n!} & \text{si } i \in S \\ -\frac{s!(n-s-1)!}{n!} & \text{si } i \notin S \end{cases}$$

Denotando $A(n) = (a(i, S))$, se tiene que

$$\boxed{Sh(N, v) = A(n)v}$$

Consecuencias de la representación matricial

- Cálculo del valor de Shapley en juegos paramétricos.
- Formulación matricial del valor de Shapley en términos de los dividendos de Harsanyi.
- Interpretación de los coeficientes de $A(n)$ como derivadas parciales:

$$a(i, S) = \frac{\partial Sh_i}{\partial x_S}(v)$$

donde x_S es la coordenada correspondiente al valor $v(S)$ en v .