

UNIVERSIDAD DE SEVILLA
DEPARTAMENTO DE MATEMÁTICA APLICADA II

COMPLEJIDAD Y ALGORITMOS
EN JUEGOS
COOPERATIVOS

Julio Rodrigo Fernández García
TESIS DOCTORAL

Memoria presentada por
Julio Rodrigo Fernández García
Para optar al grado de
DOCTOR por la UNIVERSIDAD DE SEVILLA.

Vº Bº de los Directores:

Fdo.: J. Mario Bilbao Arrese

Fdo.: Jorge J. López Vázquez

Sevilla, Febrero de 2000

*"Toda sabiduría viene del Señor,
y está con él por siempre".*

Libro del Eclesiástico 1.1

A Joaqui, Soledad y Elena

A mis padres

Agradecimientos:

A Jesús Mario Bilbao Arrese y a Jorge López Vázquez por encomendarme la realización de este trabajo, por la colaboración prestada y por la confianza depositada.

A los miembros del Grupo de Investigación de Métodos Combinatorios en Teoría de Juegos, especialmente a Nieves Jiménez Jiménez, por su colaboración en la redacción final de esta memoria.

A mi hermano Nicolás Luis, Matemático y Profesor T.E.U. del Área de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Córdoba, por sus orientaciones en cuestiones relativas a Algoritmia y Complejidad, y por su estímulo.

A todos los que me han animado a hacer la tesis desde que me incorporé al Departamento de Matemática Aplicada II de la Universidad de Sevilla y a la E.U.P. de Sevilla, en comisión de servicios, procedente del I.P.F.P. de Sevilla.

A mis compañeros.

Índice

1	Introducción	3
1.1	Conceptos básicos	4
1.1.1	Conjuntos parcialmente ordenados	4
1.1.2	Grafos	9
1.1.3	Geometrías convexas.	13
1.2	Algoritmos	16
1.2.1	Eficiencia o complejidad de los algoritmos	17
1.2.2	Programación dinámica	25
1.2.3	El sistema MATHEMATICA	27
1.3	Sumario	28
2	Algoritmos para el cálculo de valores en juegos cooperativos	31
2.1	Juegos cooperativos	31
2.1.1	Los valores de Shapley y Banzhaf	38
2.1.2	Índices de poder en juegos simples	41
2.2	Algoritmos para el cálculo del valor de Shapley	43
2.2.1	El potencial de Hart y Mas-Colell	43
2.2.2	Dividendos de Harsanyi	48
2.3	Algoritmos con MATHEMATICA	51
2.3.1	Cálculo del valor de Shapley	51
3	Algoritmos para el cálculo de valores en juegos restringidos	59
3.1	Juegos restringidos por situaciones de comunicación	59

3.2	El valor de Myerson en juegos restringidos por árboles	79
3.3	Generalización del modelo de Myerson. Estructuras \cup -estables	90
3.4	El valor de Myerson generalizado en geometrías convexas.	107
3.5	Algoritmos con MATHEMATICA	117
4	Funciones generatrices para calcular índices de poder	127
4.1	Funciones generatrices	127
4.2	Algoritmos para juegos de votación ponderada	130
4.2.1	Índice de poder de Banzhaf normalizado	130
4.2.2	Índice de poder de Shapley-Shubik	140
4.3	Algoritmos para juegos de votación con doble mayoría	149
4.3.1	Índice de poder de Banzhaf normalizado	149
4.3.2	Índice de poder de Shapley-Shubik	160
4.4	Algoritmos con MATHEMATICA	169
5	Funciones generatrices para juegos restringidos	175
5.1	Valor de Myerson generalizado mediante funciones generatrices	176
5.2	El valor de Myerson en juegos restringidos por grafos estrella .	187
5.3	Algoritmos con MATHEMATICA	210
	Referencias	223

Capítulo 1

Introducción

En principio, se quiere indicar que esta memoria de investigación, desarrollada en cinco capítulos, se enmarca en la teoría de juegos y, dentro de ella, en contextos relacionados con modelos de cooperación total y parcial. Además, ha tenido como objetivo principal el estudio e implementación algorítmica de algunos conceptos de solución establecidos para juegos cooperativos y, especialmente, para juegos simples de votación ponderada.

Evidentemente, en el transcurso del trabajo, ha sido necesaria la utilización de algunos conceptos y procedimientos que, no siendo específicos de la teoría de juegos, han supuesto un soporte técnico para ciertas definiciones, razonamientos, demostraciones y generalizaciones. Por ello, se entiende que este primer capítulo debe tener un carácter introductorio y, en él, establecer, con algún detalle, aquellas nociones no relacionadas directamente con la teoría de juegos que serán indispensables para la correcta comprensión de esta memoria.

Así, el capítulo se ha dividido en tres secciones principales: conceptos básicos, algoritmos y sumario. En la primera sección se incluyen conceptos sobre conjuntos parcialmente ordenados, grafos —en especial sobre aquellos grafos denominados árboles—, y geometrías convexas. Es obvio que estos contenidos dan lugar a subsecciones inconexas entre sí y cuya única inten-

ción es formular con precisión aquellos conceptos que serán necesarios en el estudio de juegos con cooperación parcial. En la segunda sección, se hace un resumen de nociones básicas de Algoritmia —fundamentales para el estudio de la complejidad espacial y temporal de algunos algoritmos— y una breve referencia al sistema de cálculo simbólico MATHEMATICA, que es el que ha sido utilizado en el estudio empírico de los diferentes resultados que se han obtenido. Por último, en la tercera sección, se realiza una síntesis de los contenidos y resultados principales de los demás capítulos del trabajo de investigación que se presenta.

1.1 Conceptos básicos

En cualquier juego cooperativo pueden formarse diferentes grupos de cooperación a partir del conjunto inicial de jugadores. Estas coaliciones o grupos de jugadores, junto con la relación de inclusión, forman un conjunto parcialmente ordenado. Por dicho motivo se definen a continuación algunos conceptos referentes a conjuntos parcialmente ordenados utilizándose, en lo que sigue, las notaciones de Stanley [56] y Birkhoff [14].

1.1.1 Conjuntos parcialmente ordenados

Un *conjunto parcialmente ordenado* es un par (P, \leq) donde P es un conjunto y \leq una relación binaria que satisface las propiedades reflexiva, antisimétrica y transitiva. Se dice que $\hat{1} \in P$ es *último elemento* de P si $x \leq \hat{1}$, para todo $x \in P$. Análogamente, se dice que $\hat{0} \in P$ es *primer elemento* de P si $\hat{0} \leq x$, para todo $x \in P$.

Un ejemplo de conjunto parcialmente ordenado sería el conjunto 2^N de todos los subconjuntos de un conjunto N , ordenado por inclusión; es decir, si $A, B \in 2^N$, entonces $A \leq B$ en 2^N si y sólo si $A \subseteq B$. Evidentemente, si N es finito, entonces $(2^N, \subseteq)$ es también finito.

En el caso de ser $Q \subseteq P$ puede definirse un orden parcial en Q , denominado *orden inducido*, de la siguiente forma: para $x, y \in Q$, $x \leq y$ en Q si y sólo si $x \leq y$ en P . Al conjunto Q se le llama *subconjunto parcialmente ordenado* inducido por el orden de P . Dos clases de subconjuntos parcialmente ordenados son los intervalos y las cadenas.

Sean x, y elementos del conjunto parcialmente ordenado P , con $x \leq y$. El conjunto

$$[x, y] = \{z \in P : x \leq z \leq y\},$$

se llama *intervalo*. Si cualquier intervalo de (P, \leq) es finito se dice que (P, \leq) es *localmente finito*.

Se dice que dos elementos $x, y \in P$ son *comparables* si $x \leq y$ o bien $y \leq x$. En otro caso, son *incomparables*.

Una *cadena* C de P es un subconjunto parcialmente ordenado inducido por el orden de P , en el cual no hay elementos incomparables; es decir,

$$C \subseteq P \text{ es una cadena si } x \leq y \text{ o bien } y \leq x, \text{ para todo par } \{x, y\} \subseteq C,$$

denominándose *longitud de la cadena* C al número $l(C) = |C| - 1$.

Otra clase particular de subconjuntos parcialmente ordenados inducidos por P , lo forman los llamados ideales del orden de P . Se dice que $I \subseteq P$ es un *ideal del orden de P* cuando

$$\forall x \in I, \text{ si } y \leq x \implies y \in I.$$

En particular, sea $x \in P$. El par (Λ_x, \leq) , donde

$$\Lambda_x = \{y \in P : y \leq x\},$$

se denomina *ideal principal del orden generado por x* .

Si $x, y \in P$, se dice que y *cubre* a x si $x < y$ y no hay ningún elemento $z \in P$ que cumpla la condición $x < z < y$. Esto es, y cubre a x si y sólo si

$$x < y \quad \text{y} \quad [x, y] = \{x, y\}.$$

Cuando se consideren conjuntos parcialmente ordenados mediante la inclusión, la relación de cubrir se denotará por \prec .

Un elemento *maximal* de un subconjunto X de P es un elemento $a \in X$ tal que no existe $x \in X$ verificando $a < x$. Dualmente, un elemento $b \in X$ es *minimal* si no existe $x \in X$ tal que $x < b$.

Se denomina *diagrama de Hasse* de un conjunto parcialmente ordenado y finito P a un grafo cuyos vértices son los elementos de P y las aristas vienen determinadas por la relación de cubrir.

Una *cota superior* de un subconjunto X de un conjunto parcialmente ordenado P es un elemento $a \in P$ tal que $x \leq a$ para todo $x \in X$. Si además, para cada y cota superior de X , se verifica $a \leq y$, entonces el elemento $a \in P$ es el *supremo* de X . Análogamente, se definen los conceptos de *cota inferior* y de *ínfimo* del conjunto X . Un *retículo* es un conjunto parcialmente ordenado P para el que cualquier par de elementos de P tiene supremo e ínfimo. Usualmente, se denota

$$a \wedge b = \inf \{a, b\}, \quad a \vee b = \sup \{a, b\}.$$

Un retículo es *completo* si cualquier subconjunto suyo no vacío tiene supremo e ínfimo.

Una clase importante de retículos desde el punto de vista combinatorio son los *retículos distributivos*. Un retículo es distributivo si verifica

$$\begin{aligned} x \vee (y \wedge z) &= (x \vee y) \wedge (x \vee z) \\ x \wedge (y \vee z) &= (x \wedge y) \vee (x \wedge z). \end{aligned}$$

Un tipo especial de retículo distributivo es el retículo 2^N de todos los subconjuntos de un conjunto arbitrario N . Cualquier cadena es también un retículo distributivo.

A continuación, se exponen algunos conceptos relativos al álgebra de incidencia de un conjunto parcialmente ordenado localmente finito y la fórmula de inversión de Möebius.

Sea P un conjunto parcialmente ordenado localmente finito y \mathbb{K} un cuerpo (usualmente \mathbb{R} o \mathbb{C}). Se dice que

$$f : P \times P \longrightarrow \mathbb{K}$$

es una *función de incidencia* de P sobre \mathbb{K} si $f(x, y) = 0$ cuando $x \not\leq y$. Esta definición implica que una función de incidencia es forzosamente nula cuando se evalúa sobre pares que no constituyen intervalos de P . Se denotará por $I(P, \mathbb{K})$ al conjunto formado por las funciones de incidencia de P sobre \mathbb{K} . Este conjunto tiene la estructura de espacio vectorial con las operaciones de suma de funciones y producto por un escalar. Además, puede definirse una segunda operación interna denominada *producto* o *convolución* de la siguiente forma

$$(f * g)(x, y) = \sum_{\{z: x \leq z \leq y\}} f(x, z) g(z, y)$$

para cualquier $(x, y) \in P \times P$. Esta operación interna $*$ tiene como elemento neutro a la *función δ de Kronecker*,

$$\delta(x, y) = \begin{cases} 1, & \text{si } x = y \\ 0, & \text{en otro caso.} \end{cases}$$

El conjunto $I(P, \mathbb{K})$ junto con las operaciones suma, producto por un escalar y convolución constituye un álgebra sobre el cuerpo \mathbb{K} , designada habitualmente por *álgebra de incidencia* $I(P, \mathbb{K})$.

Una función de $I(P, \mathbb{K})$ es la *función zeta* ζ , definida por

$$\zeta(x, y) = \begin{cases} 1, & \text{si } x \leq y \\ 0, & \text{en otro caso.} \end{cases}$$

Para el conjunto parcialmente ordenado y localmente finito de todos los subconjuntos de un conjunto finito N , fijado cualquier $S \in 2^N, S \neq \emptyset$, la función zeta daría lugar a la función $\zeta_S : 2^N \longrightarrow \mathbb{R}$,

$$\zeta_S(T) = \zeta(S, T) = \begin{cases} 1, & \text{si } S \subseteq T \\ 0, & \text{en otro caso.} \end{cases}$$

Dada $f \in I(P, \mathbb{K})$, la función inversa para la convolución existe si y sólo si $f(x, x) \neq 0$, para todo $x \in P$. Ello implica que la función zeta ζ posee inversa a la que se denomina *función de Möebius* de P y se simboliza por μ . Su cálculo puede realizarse mediante las siguientes fórmulas recurrentes

$$\mu(x, y) = \begin{cases} 1, & \text{si } x = y \\ - \sum_{\{z: x \leq z < y\}} \mu(x, z), & \text{si } x < y \text{ en } P. \end{cases}$$

Como resultado fundamental relacionado con la función de Möebius hay que resaltar la *fórmula de inversión de Möebius*, la cual establece que si P es un conjunto parcialmente ordenado en donde cualquier ideal principal del orden es finito y si $f, g : P \rightarrow \mathbb{C}$, entonces, para todo $x \in P$, se verifica que

$$g(x) = \sum_{y \leq x} f(y) \iff f(x) = \sum_{y \leq x} \mu(y, x) g(y).$$

Si P es el álgebra booleana 2^N , la función de Möebius de P viene dada por

$$\mu(T, S) = (-1)^{|S|-|T|}.$$

De aquí, que la fórmula de inversión de Möebius para 2^N establezca lo siguiente: si $f, g : P \rightarrow \mathbb{C}$, entonces

$$g(S) = \sum_{T \subseteq S} f(T) \iff f(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} g(T).$$

Por último, si P es un retículo finito y existe algún elemento $x \in P$ tal que $x \neq \hat{1}$, entonces, una consecuencia inmediata de [56, Corolario 3.9.3] es

$$\sum_{x \in P} \mu(x, \hat{1}) = 0.$$

1.1.2 Grafos

Tal como se indicó en la introducción, se incluye, en esta sección, un apartado en el que se tratan algunas nociones sobre grafos. En su desarrollo se seguirá, básicamente, la notación de Harary [34] y, de forma particular, se incluirán propiedades relativas a árboles enunciadas por Owen [44] y Brassard [23].

Un *grafo no dirigido* G es un par (N, E) constituido por un conjunto finito N de elementos denominados *nodos* o *vértices* y un conjunto finito E de elementos llamados *aristas* o *arcos*.

Cada arista se identifica con un par no ordenado de vértices, denominados *vértices finales*, indicándose que dos aristas son *paralelas* si tienen los mismos vértices finales. Además, si los vértices finales de una arista coinciden, entonces la arista recibe el nombre de *lazo*.

Un grafo no dirigido se denomina *simple* si no tiene lazos ni aristas paralelas. Dada una arista, se dice que es *incidente* con sus vértices finales. Dos vértices se denominan *adyacentes* si son los vértices finales de alguna arista. El número de aristas incidentes en un vértice $x \in N$, se denomina *grado* del vértice x . Los vértices con grado uno se denominan *vértices colgantes* y si tienen grado cero se denominan *vértices aislados*.

Un *paseo* en un grafo G es una sucesión finita de vértices $v_0, v_1, v_2, \dots, v_k$ tales que v_{i-1} es adyacente a v_i para $i = 1, \dots, k$. Los vértices v_0, v_k se denominan los vértices finales del paseo v_0-v_k , siendo el número k de aristas del paseo su *longitud*. Se puede observar que en un paseo, aristas y vértices pueden repetirse. Con carácter más específico, un *camino* es un paseo en el que no hay vértices repetidos, lo que implica que tampoco hay aristas repetidas, y un *ciclo* es un paseo con al menos tres vértices diferentes que no aparecen repetidos, exceptuando el primero y el último.

Sea $G = (N, E)$ un grafo no dirigido. Se dice que G es *conexo* si existe un camino entre cualquier pareja de vértices distintos. Por otra parte, un

grafo es *2-conexo* si para cualquier par de vértices x, y existen al menos dos caminos que los unen y sin vértices en común, salvo los extremos x, y .

El grafo $G' = (N', E')$ es un *subgrafo* de $G = (N, E)$ si $N' \neq \emptyset$, $N' \subseteq N$ y $E' \subseteq E$, donde cada arista de E' es incidente con vértices de N' . Dado $S \neq \emptyset$, $S \subseteq N$, el subgrafo de $G = (N, E)$ *inducido* por S es el subgrafo cuyo conjunto de vértices es S y contiene a todas las aristas de E incidentes con vértices de S . Dicho subgrafo se denota por $G/S = (S, E(S))$.

Sea $G = (N, E)$ un grafo y $S \subseteq N$. Se dice que S es un *subconjunto conexo* si el subgrafo inducido $G/S = (S, E(S))$ es conexo.

Ejemplo 1.1 Sea $G = (N, E)$ el grafo de la Figura 1.1, con $N = \{1, 2, 3, 4, 5\}$. La familia de subconjuntos conexos es

$$\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 3\}, \{1, 2\}, \{2, 4\}, \{2, 5\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 5\}, \{2, 4, 5\}, \{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 4, 5\}, \{1, 2, 3, 4, 5\}\}.$$

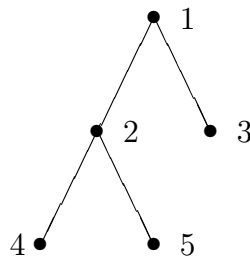


FIGURA 1.1

Se denomina *componente* de un grafo a todo subgrafo maximal conexo y un *punte* a una arista cuya eliminación aumenta el número de componentes.

Sea N un conjunto de n vértices. El *grafo completo* en N , denotado por K_n , es un grafo simple en el que, para cualquier $a, b \in N$, $a \neq b$, existe una arista $\{a, b\}$.

Un *bloque* de un grafo G es un puente o un subgrafo 2-conexo maximal de G . Un grafo G se denomina *ciclo-completo* o *grafo bloque* si cualquier bloque es un grafo completo; es decir, si contiene un ciclo, entonces el subgrafo inducido por los vértices del ciclo es completo. En particular, los grafos sin ciclos y los grafos completos son grafos ciclo-completos.

Un *árbol* es un grafo conexo que no tiene ciclos. En general, un grafo sin ciclos se denomina *bosque*. Los árboles tienen un cierto número de propiedades sencillas, entre las que es preciso destacar las siguientes:

- (a) Un árbol con n nodos posee exactamente $n - 1$ aristas.
- (b) Si se añade una única arista a un árbol, entonces el grafo resultante contiene un único ciclo.
- (c) Si se elimina una única arista de un árbol, entonces el grafo resultante ya no es conexo.

Sean G un bosque y S un subconjunto conexo. Se dice que i es un *punto extremal* de S si, en el subgrafo G/S , i tiene un grado 0 ó 1; es decir, hay a lo sumo otro $j \in S$ que está directamente conectado a i . El conjunto de los puntos extremales de S se representará por $ex(S)$.

En un árbol la intersección de subgrafos conexos es conexa. Asimismo, si $G = (N, E)$ es un árbol, para cualquier $T \subseteq N$, existe un único conjunto S minimal, en el sentido de la inclusión, conexo tal que $T \subseteq S$. Dicho conjunto, denominado *envoltura conexa* de T , viene dado por

$$H(T) = \bigcap \{S : T \subseteq S, S \text{ es conexo}\}.$$

En el Ejemplo 1.1, para $T = \{3, 4\}$ su envoltura convexa es $H(T) = \{1, 2, 3, 4\}$. Obsérvese que $S = H(S)$ si, y sólo si, S es conexo.

Sea $G = (N, E)$ un árbol y considérese un vértice cualquiera p del grafo G . La selección del vértice p , denominado *raíz*, permite la introducción de la siguiente relación de orden entre los nodos de G : $a \leq b$ si, y sólo si, existe un camino p - b que contiene al vértice a . Es inmediato que, en efecto, es una relación de orden en la que p es el primer elemento. En esta situación, el árbol $G = (N, E)$ suele denominarse *árbol con raíz p* , utilizándose como representación gráfica del mismo el diagrama de Hasse correspondiente.

Ejemplo 1.2 Sea $G = (N, E)$ el grafo del Ejemplo 1.1. Fijado el vértice $p = 3$, la representación gráfica del árbol con raíz 3 sería

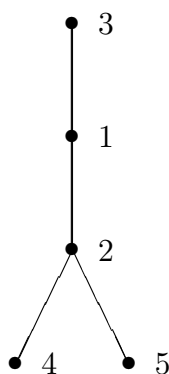


FIGURA 1.2

En un árbol con raíz p , una *hoja* es un nodo maximal, denominándose a los demás nodos, que no sean la raíz p o una hoja, *nodos internos*. Además, si x e y son dos vértices tales que $x < y$, entonces, utilizando la nomenclatura habitual en Algoritmia, se dice que x es *padre* e y es *hijo* en esta relación entre nodos adyacentes.

Para describir la posición de un nodo dentro de un árbol con raíz se utilizan los siguientes conceptos: la *altura de un nodo* es el máximo de las

longitudes de aquellas cadenas cuyos primer y último elemento son, respectivamente, el nodo en cuestión y los elementos maximales u hojas. La *profundidad* de un nodo es la longitud de la cadena que tiene a éste como último elemento y cuyo primer elemento es la raíz p , denominándose *nivel* del nodo a la diferencia entre la altura de la raíz del árbol y la profundidad de un determinado nodo.

Por último, se definen el *factor de ramificación* de un árbol con raíz p al número máximo de hijos que tiene un nodo y la *altura de un árbol* como la altura de su raíz.

1.1.3 Geometrías convexas.

El interés del estudio de las geometrías convexas proviene de la búsqueda de estructuras combinatorias que generalicen, si es posible, los resultados obtenidos con otras estructuras de cooperación utilizadas en la teoría de juegos. Aquí, se indicarán, únicamente, aquellos conceptos básicos que serán utilizados con posterioridad.

Un *espacio de clausura* [30] es un par (N, \mathcal{L}) donde \mathcal{L} es una familia de subconjuntos de un conjunto finito N , que cumple las siguientes propiedades:

(P1) $\emptyset \in \mathcal{L}$ y $N \in \mathcal{L}$.

(P2) Si $A \in \mathcal{L}$ y $B \in \mathcal{L}$, entonces $A \cap B \in \mathcal{L}$.

Los elementos de \mathcal{L} se denominan *conjuntos cerrados* [63].

Teniendo en cuenta que la intersección es cerrada para los elementos de \mathcal{L} y que cualquier subconjunto de N está incluido, al menos, en un conjunto cerrado, tiene sentido definir el *operador clausura*

$$- : 2^N \longrightarrow 2^N, \quad A \longmapsto \bar{A} = \bigcap \{C \in \mathcal{L} : A \subseteq C\} \in \mathcal{L},$$

que tiene las siguientes características:

(C1) $A \subseteq \overline{A}$, para todo $A \subseteq N$.

(C2) Si $A \subseteq B \subseteq N$, entonces $\overline{A} \subseteq \overline{B}$.

(C3) $\overline{\overline{A}} = \overline{A}$, para todo $A \subseteq N$.

Obviamente $\overline{\emptyset} = \emptyset$. Para $S \subseteq N$, \overline{S} se denominará *clausura* de S . Recíprocamente, a partir de cualquier operador clausura, se puede construir el espacio de clausura \mathcal{L} determinado por los cerrados del operador; es decir,

$$A \in \mathcal{L} \text{ si y sólo si } A = \overline{A}.$$

Una *base* para $S \in 2^N$ es un subconjunto minimal $B \subseteq S$ que verifica $\overline{B} = \overline{S}$. Un elemento i de un conjunto $S \subseteq N$ es un *punto extremal* de S si $i \notin \overline{S \setminus \{i\}}$. El conjunto de los puntos extremales de S se denota por $ex(S)$, el cual puede ser vacío. En general, si (N, \mathcal{L}) es un espacio de clausura, se verifican las siguientes condiciones:

- (a) Si $A \in \mathcal{L}$, entonces $i \in A$ es un punto extremal de A si y sólo si $A \setminus \{i\} \in \mathcal{L}$.
- (b) Si $S \subseteq A$ es una base de A , entonces $ex(A) \subseteq S$.

La propiedad (b) pone de manifiesto que cualquier base de $A \in 2^N$ contiene a los puntos extremales de A . Es evidente, que ello no implica que el conjunto de los puntos extremales de A constituya una base suya. Si esto ocurriese para cualquier conjunto cerrado; es decir,

$$\forall S \in \mathcal{L}, \quad \overline{ex(S)} = S,$$

se dice que el espacio de clausura (N, \mathcal{L}) satisface la *propiedad de Minkowski-Krein-Milman*. Así, el par (N, \mathcal{L}) es una *geometría convexa* si es un espacio de clausura que verifica la propiedad de Minkowski-Krein-Milman: cada conjunto cerrado es la clausura de sus puntos extremales o, equivalentemente,

para cualquier conjunto cerrado, el conjunto de sus puntos extremales es la única base.

Si (N, \mathcal{L}) es una geometría convexa y $S \in \mathcal{L}$, $S \neq \emptyset$, el conjunto de sus puntos extremales es no vacío ya que, en otro caso, resultaría una contradicción: si $ex(S) = \emptyset$, entonces $S = \overline{ex(S)} = \overline{\emptyset} = \emptyset$.

Edelman y Jamison establecen, [30, Teorema 3.7], que G es un grafo bloque conexo si y sólo si (N, \mathcal{L}) , con

$$\mathcal{L} = \{S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G \},$$

es una geometría convexa. Como ejemplo de esta última afirmación, si se tiene un grafo de n vértices correspondiente a una cadena y se consideran todos sus subconjuntos conexos, denotando por \mathcal{L}_n al conjunto formado por ellos, el par (N, \mathcal{L}_n) es una geometría convexa. El caso particular de \mathcal{L}_5 es ilustrado en la siguiente figura.

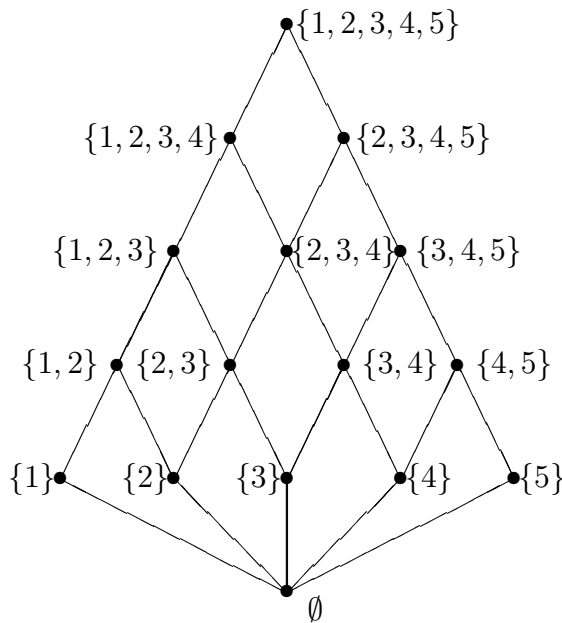


FIGURA 1.3 La geometría convexa \mathcal{L}_5 .

Por último, si (N, \mathcal{L}) es una geometría convexa y $S \in \mathcal{L}$, entonces el intervalo $[S^-, S]$, en el retículo (\mathcal{L}, \subseteq) , es un álgebra de Boole, siendo

$$S^- = S \setminus ex(S).$$

Este resultado, es una consecuencia directa de un teorema de Edelman y Jamison [30, Teorema 4.2] en el que se establece que el intervalo $[C, K]$, con $C, K \in \mathcal{L}$, es un álgebra de Boole en (\mathcal{L}, \subseteq) si y sólo si $K \setminus C \subseteq ex(K)$. Aquí, es inmediato por ser

$$S^- = \bigcap_{i \in ex(S)} (S \setminus \{i\}) = S \setminus ex(S) \in \mathcal{L},$$

y por tanto $S \setminus S^- = ex(S)$.

1.2 Algoritmos

Escribir un programa de ordenador para resolver un problema comprende varios pasos que van desde la formulación y especificación del problema, el diseño de la solución, su implementación, prueba y documentación, hasta la evaluación de la solución. Según Aho, Hopcroft y Ullman [1], “la mitad del trabajo es saber qué problema se va a resolver”. En general, se puede recurrir a casi cualquier rama de las matemáticas y de las ciencias para obtener un modelo de cierto tipo de problemas. Cuando ya se cuenta con un modelo matemático adecuado del problema, puede buscarse una solución en función de ese modelo. El objetivo inicial consiste en hallar una solución en forma de *algoritmo*, que es una secuencia finita de instrucciones, cada una de las cuales tiene un significado preciso y puede ejecutarse con una cantidad finita de esfuerzo en un tiempo finito. Las instrucciones de un algoritmo pueden ejecutarse cualquier número de veces, siempre que ellas mismas indiquen la repetición. No obstante, se exige que un algoritmo termine después de ejecutar un número finito de instrucciones sin importar cuáles fueron los

valores de entrada. Así, un programa es un algoritmo mientras no entre en un ciclo infinito con ninguna entrada.

Según G. Heileman [37] “un algoritmo es cualquier procedimiento computacional bien definido, junto con un conjunto de datos de entrada permitidos, que produce un valor o conjunto de valores como salida”.

Es importante reconocer la diferencia entre un problema y un algoritmo que resuelve un problema. Un problema tiene un sólo enunciado que lo describe en ciertos términos generales; sin embargo, puede haber muchas formas diferentes de resolver un problema, y algunas de las soluciones pueden ser más eficaces que otras.

1.2.1 Eficiencia o complejidad de los algoritmos

La resolución de un determinado problema se puede lograr por medio de diferentes algoritmos lo cual implica la necesidad de elegir el mejor entre ellos y, por tanto, se han de evaluar los recursos que requiere cada algoritmo para su ejecución. A este proceso de evaluación de los recursos se le denomina cálculo de *la eficiencia o complejidad de un algoritmo*. Se dice que un algoritmo es más eficiente o de menor complejidad que otro cuando utiliza menos recursos.

Los principales recursos que suelen tenerse en cuenta a la hora de evaluar la eficiencia de un algoritmo son el espacio de memoria que necesita y el tiempo de ejecución. La *eficiencia en memoria o complejidad espacial* de un algoritmo indica la cantidad de almacenamiento necesario para ejecutar el algoritmo; es decir, el espacio de memoria que ocupan todas las variables utilizadas por el algoritmo. Suele ser fácil calcular la memoria estática de un algoritmo, ya que sólo hay que sumar la memoria ocupada por todas las variables declaradas en el algoritmo. Sin embargo, el cálculo de la memoria dinámica no es tan fácil, debido a que ésta depende de cada ejecución concreta del algoritmo. La *eficiencia en tiempo o complejidad temporal* de un

algoritmo indica el tiempo que requiere para su ejecución.

La eficiencia en tiempo y espacio son objetivos contrapuestos por ser, generalmente, la optimización de uno de ellos a costa del otro. El programador es quien debe establecer la relación adecuada entre espacio y tiempo, para lo cual ha de tener en cuenta el uso pretendido del algoritmo y los recursos disponibles.

En este trabajo, aunque se darán algunos resultados de complejidad espacial, se abordará primordialmente el estudio de la eficiencia temporal.

El incesante aumento de la capacidad que poseen los nuevos ordenadores, tanto en velocidad de cálculo, como en almacenamiento de memoria, no implica que se tenga que desechar el estudio de la eficiencia de los algoritmos, ya que un mal algoritmo ejecutado en un ordenador muy potente puede tardar mucho más que otro algoritmo mejor que se ejecute en otra máquina menos potente.

Hay tres factores que influyen en el cálculo de la complejidad temporal: el tamaño de los datos de entrada, el contenido de los datos de entrada, y el ordenador y el código generado por el compilador, siendo el más importante, para medir la eficiencia de un algoritmo, el referente al tamaño de los datos de entrada. El tamaño de un ejemplar x se corresponde formalmente con el número de dígitos binarios necesarios para representarlo en el computador. No obstante lo anterior, se suele identificar el tamaño con el número de elementos lógicos contenidos en el ejemplar. Por ejemplo, un problema consistente en ordenar n enteros será considerado de tamaño n , aunque se necesite en la práctica más de un dígito binario para representar cada entero. Cuando se trate de problemas numéricos, como el cálculo del factorial de un número, la eficiencia se expresará en función del *valor* del dato considerado y no en función de su tamaño (que sería la longitud de la representación binaria de dicho valor).

En relación con el segundo factor, el contenido de los datos de entrada, hay

dos estrategias posibles en el cálculo de la complejidad temporal: estudiar la eficiencia en el *caso peor* y el estudio de la eficiencia en el *caso medio*. La primera de las estrategias consiste en, fijado un tamaño del problema, analizar la eficiencia del algoritmo en aquellas situaciones en las que emplea más tiempo y teniendo como objetivo la obtención de una cota superior del tiempo de ejecución del algoritmo. El estudio de la eficiencia en el caso medio plantea la necesidad de conocer el tiempo de ejecución del algoritmo en todas las situaciones y la frecuencia con que éstas se presentan; es decir, su distribución de probabilidades. Esta distribución suele ser difícil de conocer, por lo que han de realizarse hipótesis cuya fiabilidad es discutible. Además, las operaciones matemáticas necesarias para realizar el cálculo pueden llegar a hacer éste impracticable en muchos casos. Por todo ello, el análisis del caso medio es menos frecuente que el estudio del caso peor. En el presente trabajo siempre se realizará el análisis de la eficiencia en el peor de los casos.

El tercer factor, el ordenador y el código generado por el compilador, no suele tenerse en cuenta a la hora de calcular la eficiencia en tiempo de un algoritmo debido a que, por un lado, se pretende analizar la eficiencia de un algoritmo de un modo totalmente independiente de las máquinas y lenguajes existentes, y a que, por otro lado, se acepta el *principio de invariancia*: diferentes implementaciones de un mismo algoritmo diferirán en sus tiempos de ejecución a lo sumo en *una constante multiplicativa* positiva, para problemas de tamaño suficientemente grande. Más precisamente, si $t_1(n)$ y $t_2(n)$ son los tiempos de ejecución de dos implementaciones, siempre existe un número real positivo c y un número entero n_0 tales que, para todo $n \geq n_0$, se verifica que $t_1(n) \leq c t_2(n)$. Por ejemplo, un factor constante de 10, 100 ó 1000 en los tiempos de ejecución no se considera en general importante frente a una diferencia en la dependencia del tamaño n del problema, ya que, para un tamaño suficientemente grande, es dicha dependencia quien establece la dependencia real.

Ahora bien, es preciso indicar que existen tres enfoques metodológicos para evaluar la eficiencia de los algoritmos:

Empírico o a posteriori: los algoritmos se implementan en un ordenador y se comparan mediante la realización de pruebas con datos del problema de distinto tamaño.

Teórico o a priori: se determina, matemáticamente, la cantidad de recursos necesitados por cada algoritmo como una función cuya variable independiente es el tamaño de los datos del problema.

Híbrido: se determina teóricamente la función que describe la eficiencia de un algoritmo y luego se calculan empíricamente los parámetros numéricos requeridos por un programa y ordenador concretos. Este enfoque permite hacer predicciones, mediante extrapolación, sobre el tiempo que tardaría en ejecutarse un algoritmo en una implementación concreta con unos datos mucho más grandes que los usados en las pruebas.

El segundo enfoque es el más interesante, porque ofrece una medida independiente del ordenador, el lenguaje de programación y la capacidad del programador. Ahorra, por un lado, el tiempo que se tardaría en programar, sin necesidad, un algoritmo ineficiente y, por otro lado, el tiempo del ordenador que se desperdiciaría probándolo. Además, permite estudiar la eficiencia de un algoritmo cuando es usado con datos de cualquier tamaño, mientras que con los enfoques empírico o híbrido, las consideraciones prácticas pueden forzar a probar los algoritmos sólo con datos de tamaño moderado. Esta última reflexión es muy importante ya que muchos algoritmos comienzan a ser eficientes, en comparación con otros, cuando se usan con datos de gran tamaño.

Debido a lo anterior, aquí se va a utilizar un enfoque teórico o a priori como método para evaluar la complejidad temporal de un algoritmo considerándose, como factor principal, el tamaño de los datos de entrada, aunque también se tendrá en cuenta el factor relativo al contenido de los datos de entrada cuando éstos reflejen el peor de los casos posibles.

Puesto que, como se ha indicado, el cálculo de la complejidad temporal de un algoritmo se realiza en función del tamaño del problema, resulta que si $f : N \rightarrow \mathbb{R}^+ \cup \{0\}$ es la función que indica cuánto tarda en ejecutarse un algoritmo, entonces $f(n)$ denota el tiempo cuando el tamaño del problema es n . La eficiencia de los algoritmos se hace mediante un estudio de $f(n)$ para valores de n suficientemente grandes, lo cual se denomina *análisis asintótico*. Para poder realizar dicho estudio se requiere de la utilización de las notaciones asintóticas, las cuales establecen unas cotas superiores o inferiores del tiempo de ejecución de un algoritmo: $\mathcal{O}(\cdot)$ que da una cota superior, $\Omega(\cdot)$ que proporciona una cota inferior y la notación $\Theta(\cdot)$ que establece, simultáneamente, una cota inferior y superior.

El conjunto de las funciones *del orden de* $f(n)$, denotado por $\mathcal{O}(f(n))$, viene dado por

$$\mathcal{O}(f(n)) = \{g : N \rightarrow \mathbb{R}^+ : \exists c \in \mathbb{R}^+, n_0 \in N, \forall n \geq n_0 \quad g(n) \leq cf(n)\}.$$

La definición de $\mathcal{O}(f(n))$ garantiza el principio de invariancia descrito anteriormente; es decir, si el tiempo de ejecución de una implementación concreta de un algoritmo está descrito por una función $g(n)$ del orden de $f(n)$, el tiempo $g'(n)$ empleado por cualquier otra implementación del mismo que difiera de la anterior en el lenguaje, el compilador, y/o la máquina empleada, también será del orden de $f(n)$. En general, se dirá que el tiempo de ejecución de un algoritmo (y, por tanto, de todas sus implementaciones) es del orden de $f(n)$.

El conjunto $\mathcal{O}(f(n))$ define un *orden de complejidad* y se escogerá, como representante, la función $f(n)$ más sencilla posible dentro del mismo. De esta forma se tiene, por ejemplo, que $\mathcal{O}(n)$ representa el orden de *complejidad lineal*, $\mathcal{O}(n^2)$ el de *complejidad cuadrática* y $\mathcal{O}(1)$ el de las *funciones constantes*.

El conjunto de las funciones $\Omega(f(n))$ viene dado por

$$\Omega(f(n)) = \{g : N \rightarrow \mathbb{R}^+ : \exists c \in \mathbb{R}^+, n_0 \in N, \forall n \geq n_0 \quad g(n) \geq cf(n)\}.$$

Las notaciones $\mathcal{O}(f(n))$ y $\Omega(f(n))$ son independientes de que el análisis sea en el caso peor o en el caso medio. Si $t(n)$ representa el tiempo de ejecución de un algoritmo y $t(n) \in \Omega(f(n))$, entonces la medida de $\Omega(f(n))$ no se refiere al mejor tiempo del mismo, sino a una cota inferior de $t(n)$. Si se realiza un análisis en el peor caso, $t(n)$ representa el tiempo del peor ejemplo de tamaño n , mientras que $f(n)$ es una cota inferior del mismo.

Obsérvese que hay una asimetría entre las medidas de $\mathcal{O}(f(n))$ y $\Omega(f(n))$. La primera establece una cota superior al peor caso y, por tanto, a todos los ejemplos de tamaño n , mientras que la segunda, al establecer una cota inferior al peor caso, permite que puedan existir infinitos ejemplos de tamaño n con un tiempo mejor que $f(n)$. Ahora bien, se verifica que

$$f(n) \in \mathcal{O}(g(n)) \iff g(n) \in \Omega(f(n)),$$

denominándose, a esta equivalencia, *regla de dualidad*.

El conjunto de las funciones *del orden exacto de $f(n)$* , denotado por $\Theta(f(n))$, se define como

$$\Theta(f(n)) = \mathcal{O}(f(n)) \cap \Omega(f(n)).$$

Nótese que esta definición es equivalente a

$$\Theta(f(n)) = \{g : N \rightarrow \mathbb{R}^+ : \exists c, d \in \mathbb{R}^+, n_0 \in N, \forall n \geq n_0 \, df(n) \leq g(n) \leq cf(n)\}.$$

Lógicamente, la notación $\Theta(\cdot)$ es más precisa que las notaciones $\mathcal{O}(\cdot)$ y $\Omega(\cdot)$, ya que $g(n)$ está en el orden exacto de $f(n)$ ($g(n) \in \Theta(f(n))$) si, y sólo si, $f(n)$ es a la vez cota inferior y superior de $g(n)$.

A continuación, se indican algunas propiedades de la notación $\mathcal{O}(\cdot)$ (para un estudio exhaustivo, consúltese [28]). Si $f, g, h : N \rightarrow \mathbb{R}^+ \cup \{0\}$, entonces se verifica

$$(1) \, f(n) \in \mathcal{O}(f(n)).$$

- (2) Si $f(n) \in \mathcal{O}(g(n))$ y $g(n) \in \mathcal{O}(h(n))$, entonces $f(n) \in \mathcal{O}(h(n))$.
- (3) Si $g(n) \in \mathcal{O}(f(n))$ y $f(n) \in \mathcal{O}(g(n))$, entonces $\mathcal{O}(f(n)) = \mathcal{O}(g(n))$.
- (4) Para todo $c \in \mathbb{R}^+$, se tiene que $\mathcal{O}(cf(n)) = \mathcal{O}(f(n))$.
- (5) Regla del máximo: $\mathcal{O}(f(n) + g(n)) = \mathcal{O}(\max(f(n), g(n)))$.

Las tres primeras propiedades definen una ordenación parcial en el conjunto de las funciones y, consecuentemente, en el conjunto de las eficiencias relativas de los diferentes algoritmos para resolver un problema dado. Por otro lado, puede observarse que la regla del máximo (o de la suma) permite simplificar drásticamente el cálculo de la eficiencia de un algoritmo.

Otra propiedad importante es la *regla del límite* que se enuncia a continuación.

- (a) $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = l \in \mathbb{R}^+ \implies f(n) \in \mathcal{O}(g(n))$ y $g(n) \in \mathcal{O}(f(n))$.
- (b) $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0 \implies f(n) \in \mathcal{O}(g(n))$ y $g(n) \notin \mathcal{O}(f(n))$.
- (c) $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = +\infty \implies f(n) \notin \mathcal{O}(g(n))$ y $g(n) \in \mathcal{O}(f(n))$.

Una consecuencia inmediata de la regla del límite es que todo polinomio en n de grado m cuyo coeficiente de mayor grado sea positivo es del orden de n^m .

Las funciones de complejidad algorítmica más usuales, ordenadas de mayor a menor eficiencia (o de menor a mayor complejidad), son

$$\mathcal{O}(1) \subset \mathcal{O}(\log n) \subset \mathcal{O}(n) \subset \mathcal{O}(n \log n) \subset \mathcal{O}(n^2) \subset \mathcal{O}(n^k) \subset \mathcal{O}(2^n) \subset \mathcal{O}(n!).$$

Las dos últimas complejidades forman parte de las complejidades *superpolinómicas* o *intratables*, mientras que las restantes pertenecen al grupo de complejidades *polinómicas*.

A la hora de evaluar la eficiencia de un algoritmo se han de tener en cuenta las siguientes consideraciones.

El hecho de que un algoritmo tenga una eficiencia que sea “buena” en términos generales no implica que lo sea en términos particulares, ya que puede haber otro algoritmo que resuelva el mismo problema de una forma más eficiente.

El que un problema P se resuelva mediante un algoritmo A con una eficiencia lineal no implica que dicho algoritmo sea necesariamente el mejor, ya que puede haber otro algoritmo B con una eficiencia logarítmica que también resuelva dicho problema.

Si se comparan las eficiencias de dos algoritmos, se ha de tener en cuenta que el cálculo del orden de complejidad es una medida *asintótica*; es decir, se realiza un estudio teórico para valores de n suficientemente grandes. Puede ocurrir que un algoritmo tenga una eficiencia asintótica peor que la de otro, pero que tenga un mejor rendimiento para valores pequeños de n . Además, si el tamaño de los datos del problema es reducido, entonces no se puede despreciar el valor de las constante multiplicativas, ya que pueden afectar bastante a la eficiencia de los algoritmos. Por ejemplo, si se supone que un problema se puede resolver mediante dos algoritmos diferentes cuyas funciones de complejidad son, respectivamente, $f(n) = n^3 - 3n^2 + 10$ y $g(n) = 100n^2 \log n + 10$, la primera función posee complejidad asintótica peor que la segunda, pero posee un mejor rendimiento para valores de n menores que 651. Por tanto, si el tamaño de los datos del problema es inferior a 651, entonces conviene utilizar el primer algoritmo aunque su comportamiento asintótico sea peor que el del segundo algoritmo.

Los aspectos que se han tratado en esta sección se refieren a conceptos preliminares de Algoritmia, la cual se ocupa del diseño y análisis sistemático de algoritmos específicos, cada uno de los cuales será más eficiente que sus predecesores para resolver algún problema dado. Un campo diferente al de

la Algoritmia es el de la Complejidad Computacional. Este campo, que tiene una trayectoria paralela al de la Algoritmia, considera globalmente todos los posibles algoritmos para resolver un problema dado. Esto incluye aquellos algoritmos en los que ni siquiera se ha pensado todavía.

Empleando la Algoritmia, se puede demostrar, expresando y analizando un algoritmo explícito, que el problema objeto de estudio se puede resolver en un tiempo que está en $\mathcal{O}(f(n))$ para alguna función $f(n)$ que se intenta reducir lo más posible. Por otra parte, al emplear la complejidad se pretende hallar una función $g(n)$ tan grande como sea posible para la cual se pueda demostrar que cualquier algoritmo capaz de resolver correctamente el problema, en todos sus casos, debe requerir un tiempo que esté en $\Omega(g(n))$. A esta función $g(n)$ se le da el nombre de cota inferior de la complejidad del problema.

Por último, es preciso indicar que en esta memoria de investigación se van a describir algoritmos para resolver problemas relativos al cálculo de valores en juegos cooperativos, analizando la eficiencia de dichos algoritmos en el contexto de la Algoritmia.

1.2.2 Programación dinámica

En esta sección se hace referencia a una técnica, la *programación dinámica*, la cual se encuadra dentro de un contexto amplio de *técnicas algorítmicas* y que se empleará en el análisis de la complejidad de algunos algoritmos. Algunas clases generales de técnicas algorítmicas son: *algoritmos de fuerza bruta*, *algoritmos divide y vencerás*, *programación dinámica*, *algoritmos voraces* y *algoritmos probabilistas* (véase [23]), teniendo cada una de ellas propiedades especiales que las hacen apropiadas para resolver ciertos tipos de problemas. La estructura de cada problema es la que determina habitualmente la posibilidad de aplicación de cada técnica específica.

La programación dinámica resuelve un problema descomponiéndolo en

varios subproblemas que son similares al problema original y con un tamaño más pequeño. Es una técnica ascendente que comienza habitualmente resolviendo los subproblemas más pequeños, guardando estos resultados, y reutilizándolos para resolver subproblemas cada vez mayores hasta que se obtiene la solución al problema original.

Es importante resaltar que un planteamiento de programación dinámica sólo está justificado si existe cierto grado de solapamiento en los subproblemas. La idea subyacente es evitar calcular el mismo resultado dos veces lo que, en general, se consigue construyendo una tabla en memoria, y rellenándola con resultados obtenidos conforme se van calculando. Estos resultados son utilizados para resolver subproblemas más grandes. Téngase en cuenta que recuperar un resultado dado en esta tabla (es decir, realizar una consulta sobre la tabla) requiere un tiempo $\Theta(1)$.

En ocasiones, cuando se usa programación dinámica para resolver un problema, es posible que se tengan que resolver algunos casos irrelevantes puesto que sólo, posteriormente, se conocerá con exactitud las subsoluciones que se necesitan. Ahora bien, si se tiene en cuenta que un algoritmo descendente recursivo no tiene esta limitación, aunque sí tiene el inconveniente de resolver de manera independiente varios subcasos más de una vez, es posible combinar las ventajas de ambas técnicas —sencillez de una formulación recursiva y eficiencia que ofrece la programación dinámica— mediante la utilización de una *función con memoria*. Es una variación de la programación dinámica que a menudo ofrece la eficiencia de la misma [28]. La técnica consiste en memorizar el algoritmo recursivo exceptuando lo ineficiente y mantener —como en la programación dinámica ordinaria— una tabla con las soluciones de los subproblemas, siendo la estructura de control, para rellenar la tabla, semejante al algoritmo recursivo.

Al programa recursivo se le añade una tabla del tamaño necesario. Inicialmente, todas las entradas de esta tabla contienen un valor especial que muestra que todavía no se han calculado. En lo sucesivo, cada vez que se

llama a la función, se examina primero la tabla para ver si ya ha sido calculada con el mismo conjunto de parámetros. De ser así, se devuelve el valor de la tabla. En caso contrario, se sigue adelante y se calcula la función. Antes de devolver el valor calculado, sin embargo, se almacena en el lugar adecuado de la tabla. De esta forma nunca es necesario calcular dos veces la función para los mismos valores de sus parámetros.

1.2.3 El sistema Mathematica

Aunque el estudio de la eficiencia o complejidad de los algoritmos se hará empleando pseudocódigo, para las aplicaciones se utilizará el sistema MATHEMATICA. Este sistema hay que encuadrarlo dentro de los sistemas generales de cálculo simbólico los cuales, entre otras características, contienen un gran número de funciones y procedimientos predefinidos y permiten un uso interactivo.

MATHEMATICA es un sistema general diseñado por Stephen Wolfram (1988) para aplicaciones matemáticas, entre otras. Es un programa escrito en lenguaje C, que permite desarrollar las principales computaciones matemáticas; es decir, las operaciones numéricas, simbólicas y gráficas, y manejarlas de forma unificada. Además, incorpora diversos paquetes de software específico.

Por último, teniendo en cuenta que en esta memoria de investigación se desarrollan algoritmos específicos para el cálculo de valores en juegos cooperativos en contextos de cooperación parcial, es necesario resaltar, por su utilización, el paquete Cooperat'Cooperat' creado por Michael Carter [26] y el denominado DiscreteMath'Combinatorica', diseñado por Steven Skiena [55] e incluido en el libro *Implementing Discrete Mathematics: Combinatorics and Graphs Theory with Mathematica*.

1.3 Sumario

En el capítulo segundo, tras hacer una síntesis de las nociones fundamentales en juegos cooperativos, se describen los valores de Shapley y Banzhaf como soluciones de dichos juegos, y se aborda el estudio de los índices de poder en juegos simples.

En la segunda sección de dicho capítulo se describe el potencial de Hart y Mas-Colell, el cual permite obtener el valor de Shapley de un juego, con una eficiencia temporal mejor que la fórmula directa. Se demuestra que, para un juego cooperativo (N, v) , la complejidad temporal del cálculo del valor de Shapley, usando la función potencial de Hart y Mas-Colell es $\Theta(n2^n)$. A continuación, en la misma sección, se analiza la complejidad del cálculo del valor de Shapley mediante los dividendos de Harsanyi, obtenidos mediante un procedimiento recurrente y una técnica ascendente, resultando que se requiere un tiempo $\Theta(3^n)$. Como conclusión inmediata se tiene que este método, en un contexto de cooperación total, no mejora el algoritmo dado por Hart y Mas-Colell [36].

Finalmente, en la tercera sección de este capítulo se incluyen algoritmos escritos con MATHEMATICA y se muestran distintas alternativas para el cálculo del valor de Shapley en diversos juegos. En los restantes capítulos también se incluyen aplicaciones con MATHEMATICA al final de los mismos.

En la primera sección tercer capítulo se describen los juegos restringidos por situaciones de comunicación y se prueba que el procedimiento descrito en el capítulo segundo para calcular el valor de Shapley a partir de los dividendos de Harsanyi puede emplearse para calcular, de manera directa, el valor de Myerson sin más que considerar la familia \mathcal{F} de coaliciones conexas en vez del conjunto 2^N . Este método de cálculo evita obtener la función característica del juego restringido así como las componentes conexas maximales de cada coalición. De hecho, en algunos casos analizados, mejora sensiblemente los tiempos de ejecución con respecto a otros algoritmos que calculan el valor de

Myerson a partir de la función característica del juego restringido o mediante la función potencial de Hart y Mas-Colell [39].

A continuación, se analiza la complejidad del cálculo del valor de Myerson cuando el grafo de cooperación es un grafo camino y un grafo estrella. En el primer caso se demuestra que tiene una complejidad temporal polinómica; en el segundo caso, cuando la situación de comunicación está modelada por un grafo estrella, se prueba que la complejidad temporal es exponencial.

En la tercera sección se estudia el valor de Myerson en juegos restringidos por árboles a partir de los trabajos de Owen [44] y Bilbao y López [12], y se estudia el cálculo de los dividendos de Harsanyi del juego restringido (N, v^G) a partir de la función característica del juego (N, v) . Se dan resultados sobre la complejidad del cálculo del valor de Myerson, y se analizan, al igual que en la sección anterior, los casos particulares de situaciones de comunicación (N, v, G) en las que G es un grafo camino o un grafo estrella.

En las secciones tercera y cuarta se introducen las estructuras de cooperación \cup -estables [2] y las geometrías convexas [30], que tienen un antecedente en las estructuras de conferencias de Myerson y en las situaciones de comunicación mediante hipergrafos, y suponen una generalización de las situaciones de comunicación. En ellas, se prueba que pueden emplearse los algoritmos dados en las secciones anteriores para el cálculo del valor de Myerson del correspondiente juego restringido o valor de Myerson generalizado.

En el cuarto capítulo se estudia el cálculo del índice de poder de Banzhaf y del índice de Shapley-Shubik en juegos simples de votación ponderada, utilizando funciones generatrices. El uso de funciones generatrices para calcular el índice de poder de Shapley-Shubik en juegos de votación tiene un antecedente en los trabajos de David G. Cantor (1962) [40] [41]. Posteriormente, Brams y Affuso (1976) [21] emplearon funciones generatrices para calcular el índice de Banzhaf normalizado. Tras analizar estos algoritmos, se pasa al estudio de juegos de votación de doble mayoría y se introducen nuevos algoritmos, mediante el empleo de funciones generatrices, para el cál-

culo de los índices de Banzhaf normalizado y de Shapley-Shubik, estudiándose asimismo su complejidad temporal. Al igual que en el resto de los capítulos en la última sección se incluye la descripción de los algoritmos usando MATHEMATICA.

En la primera sección del capítulo quinto se introducen funciones generatrices para el cálculo de los dividendos en juegos simples de votación ponderada, restringidos por geometrías convexas de partición.

En la segunda sección se parte de una nueva fórmula para el cálculo del valor de Shapley en juegos restringidos por geometrías convexas de partición. Dicha fórmula se particulariza en situaciones de comunicación (N, v, G) , siendo (N, v) un juego simple de votación ponderada y G es un grafo estrella, surgiendo la necesidad de determinar el número de coaliciones factibles que incluyan a un mismo jugador y posean un tamaño y peso prefijados. Esta búsqueda del número de coaliciones factibles que posean ciertas características se consigue mediante funciones generatrices y a través de una modificación de un algoritmo de Jamison para calcular los números de Whitney de un árbol. Se estudia la complejidad temporal del algoritmo de Jamison, se modifica para la determinación, mediante funciones generatrices, de las coaliciones conexas o factibles necesarias, y se analiza la complejidad del cálculo del valor de Myerson generalizado. Finalmente, en la última sección del capítulo, se introducen tres ejemplificaciones de los algoritmos introducidos.

Capítulo 2

Algoritmos para el cálculo de valores en juegos cooperativos

2.1 Juegos cooperativos

La teoría de juegos fue fundada por John von Neumann en 1928 [60]. Esta aproximación a los problemas de cooperación y competición se redescubrió, en 1944, cuando von Neumann escribió un tratado con el economista Oskar Morgenstern, titulado *Theory of Games and Economic Behavior* [61]. En este libro, definieron el concepto de juego cooperativo de n personas e introdujeron algunas ideas relacionadas con la solución del juego. Todo el trabajo que se ha llevado a cabo desde entonces sobre este tipo de juegos ha estado fuertemente influenciado por esta obra. En ella, un juego cooperativo es una situación derivada de una actividad en la que los elementos o actores que intervienen (personas, instituciones, empresas, etc.) persiguen alcanzar un determinado objetivo como ganar una votación, buscar mayores beneficios empresariales o mejorar una gestión, mediante la colaboración entre ellos.

A diferencia de los denominados juegos competitivos o no cooperativos —caracterizados por las estrategias que pueden emplear cada uno de los jugadores y una función de pagos asociada a cada jugador, la cual depende

de las diferentes estrategias que se empleen—, en un juego cooperativo no es necesario analizar las estrategias de los jugadores; es suficiente conocer los pagos asociados a los resultados del juego.

Un *juego cooperativo de utilidad transferible* se define como un par (N, v) , donde N es un conjunto finito y v es una función $v : 2^N \rightarrow \mathbb{R}$ que asigna a cada $S \subseteq N$ un número real y verifica que $v(\emptyset) = 0$.

Los elementos de $N = \{1, 2, \dots, n\}$ se denominan *jugadores*, los subconjuntos $S \in 2^N$ *coaliciones* y $v(S)$ es el *valor* de la coalición S . La función v se denomina habitualmente *función característica del juego*, siendo identificado —siempre que no haya lugar a confusión— el juego (N, v) mediante su función característica.

Ejemplo 2.1 *Tres ciudades de la misma comarca necesitan un sistema de tratamiento de aguas residuales. Cada ayuntamiento ha hecho un estudio de los costes, individuales y colectivos, con los otros ayuntamientos, para ver la posibilidad de ahorro. El estudio se representa en la siguiente tabla, donde 1, 2 y 3, simbolizan a cada una de las ciudades.*

COALICIÓN	COSTE	BENEFICIO
{1}	150	0
{2}	200	0
{3}	550	0
{1, 2}	350	0
{1, 3}	610	90
{2, 3}	650	100
{1, 2, 3}	780	120

Esta situación se modela mediante un juego cooperativo de utilidad transferible (N, v) , donde $N = \{1, 2, 3\}$ y la función característica v del juego viene

dada por

$$v(\{1\}) = v(\{2\}) = v(\{3\}) = 0, \\ v(\{1, 2\}) = 0, \quad v(\{1, 3\}) = 90, \quad v(\{2, 3\}) = 100, \quad v(\{1, 2, 3\}) = 120.$$

Ejemplo 2.2 *En un órgano colegiado de una institución constituido por 40 personas con derecho a voz y voto, las decisiones se adoptan mediante el voto favorable de la mayoría absoluta de sus miembros.*

En este caso, el modelo de representación del juego es $N = \{1, 2, \dots, 40\}$ y $v : 2^N \rightarrow \mathbb{R}$ definida, para cualquier $S \subseteq N$, por

$$v(S) = \begin{cases} 1, & \text{si } |S| \geq 21 \\ 0, & \text{en otro caso.} \end{cases}$$

Aquí los valores 0 y 1 reflejan si una coalición de jugadores es perdedora o ganadora en una votación.

En general, se denota por Γ^N al conjunto de todos los juegos cooperativos de utilidad transferible sobre N ; es decir,

$$\Gamma^N = \{ (N, v), v : 2^N \rightarrow \mathbb{R}, v(\emptyset) = 0 \}.$$

En el conjunto Γ^N , se introducen las operaciones

$$+ : \Gamma^N \times \Gamma^N \rightarrow \Gamma^N, \quad (v, w) \mapsto v + w$$

$$\cdot : \mathbb{R} \times \Gamma^N \rightarrow \Gamma^N, \quad (\alpha, v) \mapsto \alpha \cdot v$$

definidas por

$$(v + w)(S) = v(S) + w(S), \quad (\alpha \cdot v)(S) = \alpha \cdot v(S),$$

para cualquier $S \subseteq N$. Con respecto a estas operaciones, la terna $(\Gamma^N, +, \cdot)$ constituye un espacio vectorial $(2^n - 1)$ -dimensional.

Una base está formada por el conjunto

$$\{u_T \in \Gamma^N : T \subseteq N, T \neq \emptyset\},$$

siendo, para cada $S \subseteq N$,

$$u_T(S) = \begin{cases} 1, & \text{si } T \subseteq S \\ 0, & \text{en otro caso.} \end{cases}$$

Estos juegos u_T se denominan *juegos de unanimidad*.

Otra base del espacio Γ^N está constituida por los *juegos de identidad*, denotados, para cada $T \subseteq N, T \neq \emptyset$, por δ_T y definidos, para cada $S \subseteq N$, por

$$\delta_T(S) = \begin{cases} 1, & \text{si } T = S \\ 0, & \text{en otro caso.} \end{cases}$$

Normalmente, las propiedades especiales que tenga la función característica correspondiente a un juego cooperativo son las que cualifican y dan nombre al juego. Así, cuando $v(S) \leq v(T)$, para todo $S \subseteq T \subseteq N$, se dice que el juego (N, v) es *monótono*. Puede observarse que las funciones características de los ejemplos anteriores cumplen esta condición.

El juego (N, v) es llamado *cero-normalizado*, *cero-monótono*, *aditivo* o *superaditivo*, respectivamente, si su función característica, v , verifica la correspondiente condición:

- (a) $v(\{i\}) = 0$, para todo elemento $i \in N$.
- (b) $v(S) + \sum_{i \in T \setminus S} v(\{i\}) \leq v(T)$, para cualesquiera $S \subseteq T \subseteq N$.
- (c) $v(S \cup T) = v(S) + v(T)$, para cualesquiera $S, T \subseteq N$, con $S \cap T = \emptyset$.
- (d) $v(S \cup T) \geq v(S) + v(T)$, para cualesquiera $S, T \subseteq N, S \cap T = \emptyset$.

Una clase especial de juegos superaditivos son los llamados juegos convexos. Estos fueron introducidos por Shapley [53] y se utilizan para modelar diversas situaciones que estudian las ciencias económicas. Un juego (N, v) se dice que es *convexo* si la función característica es *supermodular*; esto es,

$$v(S \cup T) + v(S \cap T) \geq v(S) + v(T), \text{ para cualesquiera } S, T \in 2^N.$$

Un juego es *simple* si es monótono y $v(S)$ sólo toma valores en el conjunto $\{0, 1\}$, para toda coalición $S \subseteq N$. Un juego simple se denomina *propio* si no existen coaliciones $S, T \subseteq N$, con $S \cap T = \emptyset$, que verifiquen $v(S) = v(T) = 1$.

Cuando se analiza un juego cooperativo, una primera idea consiste en buscar la mejor estrategia para cada jugador y encontrar el pago que podría esperarse que obtuvieran un grupo de jugadores racionales. Sin embargo, pronto uno puede darse cuenta que éste es un objetivo demasiado ambicioso ya que hasta los juegos más elementales son demasiado complejos para determinar un solo pago. Esto no debe suponer decepción alguna sino entender que cualquier teoría, que intente ser un reflejo de la realidad, debe admitir que siempre hay una gran variedad de resultados cuando se juega en la vida real y que dependen de diversas variables, como pueden ser, entre otras, la capacidad de negociación, la habilidad de los jugadores o las presiones de tipo social.

Sin embargo, cuando hay que decidir qué resultados del juego son plausibles, una de las ideas básicas en la teoría de juegos cooperativos de utilidad transferible es que, dado un juego (N, v) y suponiendo que se llega a algún tipo de entendimiento entre los jugadores, se reparte la ganancia total, $v(N)$, de la gran coalición N entre ellos. Esto parece lógico en la situación contemplada en el Ejemplo 2.1, donde es obvio que todas las ciudades desean cooperar entre ellas y obtener un ahorro de costes de 120 millones. La cuestión central sería cómo repercute ese beneficio/ahorro en cada ciudad o, dicho de otra forma, cuál es el coste para cada una de ellas.

Por tanto, es obligado repartir el beneficio/ahorro total entre todos y, de

ahí, si se representa por x_1, x_2, x_3 lo que corresponde a cada ciudad, se tiene

$$\{(x_1, x_2, x_3) : x_1 + x_2 + x_3 = 120\},$$

con lo que una solución razonable sería, en principio, cualquier vector de \mathbb{R}^3 que verifique esta condición. También, sería deseable imponerle —a los vectores de \mathbb{R}^3 que verifican la condición anterior— las restricciones siguientes

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0,$$

$$x_1 + x_2 \geq 0, \quad x_1 + x_3 \geq 90, \quad x_2 + x_3 \geq 100,$$

ya que ello significa que cada ciudad y cada coalición tiene asegurado un reparto final de beneficios que no les perjudica en ningún caso. De ahí, que el conjunto de pagos o *repartos razonables* estaría formado por

$$\{(x_1, x_2, x_3) \in \mathbb{R}_+^3 : x_1 + x_2 + x_3 = 120, x_1 + x_2 \geq 0, x_1 + x_3 \geq 90, x_2 + x_3 \geq 100\}$$

que constituye lo que se denomina el core del juego.

De lo anterior se deduce que, a la hora de buscar resultados posibles, debe hacerse una distribución de la cantidad $v(N)$ entre los jugadores. Ésta puede ser representada por una función x con valores reales sobre el conjunto de jugadores N y debe satisfacer el *principio de eficiencia*

$$\sum_{i \in N} x_i = v(N),$$

donde x_i representa el pago al jugador i con la función x . Normalmente se identifica la función x sobre N con una n -upla de números reales

$$x = (x_1, \dots, x_n).$$

Los vectores $x \in \mathbb{R}^n$ que satisfacen el principio de eficiencia son llamados *vectores de pago eficientes* o *preimputaciones* para el juego (N, v) y, atendiendo a esta idea, una *solución* o *concepto de solución* sobre una colección no vacía de juegos es una aplicación ψ que asocia a cada juego cooperativo (N, v) de dicha colección un subconjunto $\psi(v)$ del conjunto de preimputaciones.

Además, la mayoría de los conceptos de solución propuestos para juegos cooperativos requieren que los vectores de pago eficientes cumplan el llamado *principio de individualidad racional*, el cual exige que el pago a cada jugador i por el vector de pago x sea al menos la cantidad que el jugador puede obtener por sí mismo en el juego. Es decir,

$$x_i \geq v(\{i\}), \quad \text{para todo } i \in N.$$

Las preimputaciones que verifican este principio de individualidad racional se llaman *imputaciones* para el juego (N, v) , denotándose $I(v)$ el conjunto de todas ellas. Por último, podría plantearse, como una opción entre otras posibles, que los pagos sean *coalicionalmente razonables*; esto es, que los miembros de cada coalición reciban un pago total que sea mayor o igual que el valor de dicha coalición. Si se le exige a las imputaciones que verifiquen el *principio de racionalidad para todas las coaliciones*, se obtiene el concepto de solución denominado *core*.

Así, el *core* del juego (N, v) es el conjunto

$$C(v) = \{x \in \mathbb{R}^n : x(N) = v(N), x(S) \geq v(S), \text{ para toda } S \in 2^N\},$$

donde $x(S) = \sum_{i \in S} x_i$, y $x(\emptyset) = 0$.

Esta idea del *core* de un juego fue introducida por Gillies [32] y pueden darse ejemplos de juegos en los que el *core* es vacío. No obstante, hay clases de juegos cooperativos de utilidad transferible para los que el *core* es no vacío. A este respecto, se destaca el conjunto de juegos convexos.

En el estudio de las condiciones que determinan si el juego tiene o no un *core* vacío, Shapley [52] introdujo el concepto de coaliciones equilibradas y de juego equilibrado.

Dado un juego (N, v) , una colección $\{S_1, S_2, \dots, S_m\}$ de subconjuntos de N , distintos y no vacíos, se dice que es *equilibrada sobre N* si existen números

positivos $\alpha_1, \alpha_2, \dots, \alpha_m$ —denominados *pesos*— tales que, para todo $i \in N$,

$$\sum_{\{j: i \in S_j\}} \alpha_j = 1.$$

Si, para cualquier colección equilibrada sobre N , se verifica que

$$\sum_{j=1}^m \alpha_j v(S_j) \leq v(N),$$

entonces se dice que el juego (N, v) es *equilibrado*.

Bondareva [15] y Shapley [52] demuestran que la clase de juegos equilibrados coincide con la clase de juegos con core no vacío.

Un juego (N, v) se dice *totalmente equilibrado* si los subjuegos inducidos (S, v_S) son equilibrados para toda $S \subseteq N$, $S \neq \emptyset$. Aquí se entiende por subjuego inducido (S, v_S) aquel cuya función característica viene determinada por

$$v_S(T) = v(T), \text{ para toda } T \subseteq S.$$

Existen otros conceptos de solución diferentes al core como pueden ser los *conjuntos estables* de von Neuman [61] y el *conjunto de negociación de Aumann y Maschler* [5]. Estos conceptos dan como solución de un juego cooperativo de utilidad transferible un subconjunto del conjunto de imputaciones, cuando éste es no vacío. Por otro lado, existen otros conceptos de solución que asignan a cada juego un único elemento del conjunto de preimputaciones. Entre ellos se encuentran los valores de Shapley y Banzhaf que se describen a continuación.

2.1.1 Los valores de Shapley y Banzhaf

Lloyd S. Shapley analizó los juegos cooperativos intentando contestar a la siguiente cuestión: dada la función característica de un juego, ¿cuál es el pago esperado para un jugador individual? Teniendo en cuenta los comentarios anteriores, intentar predecir la solución de un juego cooperativo parece un

asunto bastante arriesgado ya que es lógico pensar que la personalidad de los jugadores, su entorno, sus facilidades de comunicación, etc., tengan efectos sobre la conclusión final. No obstante, Shapley [51], en 1953, propone un concepto de valor de un juego (N, v) , dado, para cada jugador $i \in N$, por la siguiente expresión combinatoria

$$\Phi_i(v) = \sum_{\{S \subseteq N: i \in S\}} \frac{(s-1)!(n-s)!}{n!} [v(S) - v(S \setminus \{i\})],$$

donde $n = |N|$ y $s = |S|$. Este valor es conocido como *valor de Shapley* para el jugador i y se puede observar que está determinado, de forma exclusiva y a priori, por la función característica del juego, haciendo abstracción de cualquiera de los factores anteriormente mencionados.

El valor de Shapley es un concepto de solución que puede interpretarse como la contribución marginal *esperada* del jugador i , o como un promedio de las contribuciones marginales $v(S) - v(S \setminus \{i\})$ de dicho jugador a todas las coaliciones no vacías $S \in 2^N$ cuando se supone que la coalición a la que pertenece el jugador i es equiprobable que sea de igual tamaño s ($1 \leq s \leq n$) y que todas las coaliciones de tamaño s tienen la misma probabilidad; es decir, una suma ponderada de las contribuciones marginales del jugador i , admitiendo que la probabilidad de que el jugador i pertenezca a una coalición de tamaño s , p_s^i , viene dada por

$$p_s^i \binom{n-1}{s-1} = \frac{1}{n} \implies p_s^i = \frac{1}{n} \binom{n-1}{s-1}^{-1} = \frac{(s-1)!(n-s)!}{n!}.$$

Otra manera de introducir el valor de Shapley corresponde a la siguiente interpretación. Se supone que los jugadores forman la gran coalición incorporándose de uno en uno, en un orden elegido aleatoriamente. De esta forma, cada jugador consigue la cantidad con la que él contribuye a la coalición ya formada cuando se incorpora. El valor de Shapley distribuye a cada jugador la cantidad esperada que él obtiene por este procedimiento, suponiendo que la coalición de n jugadores puede formarse, con la misma probabilidad, en todos los órdenes posibles.

Si se parte de la apreciación subjetiva de que para el jugador i es equiprobable pertenecer a cualquier coalición, surge otro concepto de solución denominado *valor de Banzhaf-Coleman* [8] [27],

$$\Psi_i(v) = \sum_{\{S \subseteq N: i \in S\}} \frac{1}{2^{n-1}} [v(S) - v(S \setminus \{i\})], \text{ para cada } i \in N.$$

La idea de Shapley es, en principio, solamente una de las que podrían servir para tomar una decisión sobre cuál debe ser la solución de un juego cooperativo de n personas. Sin embargo, Shapley justifica su elección indicando requisitos que debería cumplir cualquier solución razonable y demostrando que, el valor de Shapley, es el único que satisface estas condiciones. Así, el valor de Shapley satisface los siguientes axiomas.

Axioma de linealidad: Si (N, v) y (N, w) son dos juegos cooperativos cualesquiera de utilidad transferible y $\alpha, \beta \in \mathbb{R}$, entonces

$$\Phi_i(\alpha v + \beta w) = \alpha \Phi_i(v) + \beta \Phi_i(w), \text{ para todo } i \in N.$$

Axioma del jugador pasivo: Si $i \in N$ es un jugador pasivo en el juego v ; es decir, $v(S) - v(S \setminus \{i\}) = v(\{i\})$ para todo $S \subseteq N$, entonces

$$\Phi_i(v) = v(\{i\}).$$

Axioma de eficiencia: Se verifica

$$\sum_{i \in N} \Phi_i(v) = v(N).$$

Axioma de simetría: Para cada permutación π del conjunto N ,

$$\Phi_{\pi i}(\pi v) = \Phi_i(v),$$

donde πv es el juego definido por $\pi v(\pi S) = v(S)$, para toda $S \subseteq N$.

2.1.2 Índices de poder en juegos simples

Tal y como se ha indicado en la sección anterior, un juego simple es un juego cooperativo $v : 2^N \rightarrow \{0, 1\}$, tal que $v(S) \leq v(T)$ cuando $S \subseteq T \subseteq N$. Una coalición S se denomina *ganadora* si $v(S) = 1$, y *perdedora* si $v(S) = 0$.

Si se denota por \mathcal{MW} el conjunto de las *coaliciones ganadoras minimales*; esto es,

$$\mathcal{MW} = \{S \subseteq N : v(S) = 1, \quad v(T) = 0 \quad \text{para toda } T \subset S\},$$

entonces para un juego simple (N, v) el valor de Shapley del jugador i viene dado por

$$\begin{aligned} \Phi_i(v) &= \sum_{\{S \subseteq N : i \in S\}} \frac{(s-1)!(n-s)!}{n!} [v(S) - v(S \setminus \{i\})] \\ &= \sum_{\{S \in \mathcal{MW} : i \in S\}} \frac{(s-1)!(n-s)!}{n!} \end{aligned}$$

y se denomina *índice de poder de Shapley–Shubik* [54]. Este índice puede interpretarse como la contribución marginal esperada de un jugador que convierte una coalición perdedora en ganadora, y su fórmula expresa una suma extendida a todas las coaliciones ganadoras minimales a las que pertenezca el jugador.

Análogamente, el valor de Banzhaf para un juego simple (N, v) viene dado por

$$\begin{aligned} \Psi_i(v) &= \sum_{\{S \subseteq N : i \in S\}} \frac{1}{2^{n-1}} [v(S) - v(S \setminus \{i\})] \\ &= \sum_{\{S \in \mathcal{MW} : i \in S\}} \frac{1}{2^{n-1}}. \end{aligned}$$

En un juego simple, el valor de Banzhaf recibe el nombre de *índice de poder de Banzhaf*. Este índice de poder, a diferencia del índice de Shapley–Shubik, no verifica la propiedad de eficiencia. Una variante es el denominado

índice de Banzhaf normalizado que sí es eficiente. Para definirlo son necesarios algunos conceptos previos que se introducen a continuación, haciendo notar que, en lo que sigue, por simplicidad en la notación, se escribirá $S \setminus i$ en lugar de $S \setminus \{i\}$ y $S \cup i$ en vez de $S \cup \{i\}$.

Un *swing* para el jugador i es un par de coaliciones $(S \cup i, S)$ tales que $S \cup i$ es ganadora y S no lo es. Para cada $i \in N$, se denota por $\eta_i(v)$ el número de swings para el jugador i en el juego (N, v) ; es decir, el número de coaliciones para las que el jugador i es crítico. El número total de swings es

$$\bar{\eta}(v) = \sum_{i \in N} \eta_i(v),$$

y el índice de Banzhaf normalizado del jugador i viene dado por

$$\beta_i(v) = \frac{\eta_i(v)}{\bar{\eta}(v)}.$$

Los índices de poder tienen una aplicación fundamental en las ciencias políticas y sociales a través de los denominados juegos de votación ponderada. Estos juegos constituyen una clase especial de juegos simples que surgen en un contexto en el que se pretende asignar índices o cuotas de poder a los partidos, grupos parlamentarios o naciones [22]. El *índice* o *cuota de poder* de cada partido mide su capacidad para lograr coaliciones que superen la mayoría absoluta o formen una coalición ganadora.

Un *juego de votación ponderada* se define en un conjunto de jugadores $N = \{1, 2, \dots, n\}$ en el que cada jugador i , que puede ser un grupo político o una nación, tiene un número de votos que se denota por w_i , con $w_i > 0$, para todo $i \in N$. Cada coalición de jugadores $S \subseteq N$, reúne la suma de los votos de sus componentes

$$w(S) = \sum_{i \in S} w_i.$$

Así, fijado un número $q > \frac{1}{2}w(N)$, la función característica del juego de votación ponderada (N, v) , $v : 2^N \rightarrow \{0, 1\}$, se define, para cada $S \subseteq N$, por

$$v(S) = \begin{cases} 1, & \text{si } w(S) \geq q \\ 0, & \text{en otro caso.} \end{cases}$$

A la vista de la definición, un juego de votación ponderada se representa por

$$v = [q; w_1, w_2, \dots, w_n],$$

y es inmediato probar que es simple y propio. En efecto, es monótono ya que

$$S \subseteq T \subseteq N \implies w(S) = \sum_{i \in S} w_i \leq \sum_{i \in T} w_i = w(T) \implies v(S) \leq v(T)$$

y, además, es propio ya que si existieran dos coaliciones ganadoras disjuntas $S, T \subseteq N$, se tendría

$$\sum_{i \in S} w_i \geq q, \sum_{i \in T} w_i \geq q \implies \sum_{i \in S \cup T} w_i \geq 2q > w(N),$$

lo cual es contradictorio.

El hecho de que sea simple y propio, permite afirmar que es superaditivo. Por último, es necesario precisar que en esta memoria se considerará que w_i , para todo $i \in N$, es un entero positivo y estrictamente menor que q .

2.2 Algoritmos para el cálculo del valor de Shapley

2.2.1 El potencial de Hart y Mas-Colell

En general, el cálculo de valores e índices de poder exige la resolución de un problema combinatorio que involucra un elevadísimo número de operaciones. Por ejemplo, si se quisieran estudiar los índices de poder en el seno del Consejo de Gobierno de la Unión Europea, que está formado por 15 miembros, se

debería tener en cuenta que el número de coaliciones que se pueden formar es superior a 32.000. El cálculo directo del valor de Shapley de un jugador tiene una complejidad temporal $\mathcal{O}(n2^n)$ [29]. Es por ello que se han de considerar nuevos algoritmos para los que se tenga una cota temporal más baja. Un resultado importante es el debido a Hart y Mas-Colell [36], los cuales introducen una función de potencial para el cálculo del valor de Shapley. Esta función se describe en los siguientes párrafos.

Sea Γ el conjunto de todos los juegos de utilidad transferible. La *función potencial de Hart y Mas-Colell* es una función $P : \Gamma \rightarrow \mathbb{R}$ que asigna a cada juego (N, v) un número real $P(N, v)$, y satisface las condiciones

$$P(\emptyset, v) = 0, \quad \sum_{i \in N} D^i P(N, v) = v(N),$$

donde $D^i P(N, v) = P(N, v) - P(N \setminus i, v_{N \setminus i})$.

En la expresión anterior, se entiende por $v_{N \setminus i}$ la restricción de la función característica v a las coaliciones $S \in 2^{N \setminus i}$. Es decir, $v_{N \setminus i}(S) = v(S)$ para toda coalición $S \subseteq N \setminus i$ y, siempre que no exista confusión, el juego $(N \setminus i, v_{N \setminus i})$ se denotará por $(N \setminus i, v)$.

El número real $D^i P(N, v)$ se denomina *contribución marginal* del jugador i al juego (N, v) y permite indicar que la función potencial exige, para todo juego (N, v) , que el vector formado por las contribuciones marginales de todos los jugadores sea eficiente. De esta condición de eficiencia se deduce

$$P(S, v) = \frac{1}{|S|} \left[v(S) + \sum_{i \in S} P(S \setminus i, v) \right], \text{ para cada } S \subseteq N, S \neq \emptyset,$$

lo que genera, empezando con $P(\emptyset, v) = 0$, un algoritmo recursivo de cálculo para determinar $P(N, v)$.

Hart y Mas-Colell [36] demuestran que la contribución marginal de un jugador i al juego (N, v) coincide con su valor de Shapley; esto es,

$$D^i P(N, v) = P(N, v) - P(N \setminus i, v) = \Phi_i(N, v), \text{ para todo } i \in N.$$

Además, prueban fórmulas explícitas para el potencial como

$$P(N, v) = \sum_{S \subseteq N} \frac{(s-1)!(n-s)!}{n!} v(S).$$

A continuación se describe el algoritmo *potencial* que determina los potenciales de todos los subjuegos del juego (N, v) , usando programación dinámica, mediante un proceso de cálculo ascendente según el tamaño de las coaliciones y almacenando todos los valores calculados.

Algoritmo *potencial* (N, v)

$P(\emptyset, v) \leftarrow 0$

[para i de 1 hasta n

[para j de 1 hasta $\binom{n}{i}$

$P(S_i^j, v) \leftarrow \frac{1}{|S_i^j|} \left[v(S_i^j) + \sum_{k \in S_i^j} P(S_i^j \setminus k, v) \right]$

fin_para

fin_para

donde S_i^j es la j -ésima coalición de tamaño i , $|S_i^j| = i$.

El siguiente resultado se utilizará en el análisis de la eficiencia temporal de este algoritmo.

Lema 2.3 Para cualquier número natural n , se verifica

$$\sum_{i=0}^n \binom{n}{i} (i+1) = 2^n + n2^{n-1}.$$

Demostración: Al ser $\sum_{i=0}^n \binom{n}{i} x^{i+1} = (1+x)^n x$, derivando se obtiene que

$$\sum_{i=0}^n \binom{n}{i} (i+1)x^i = (1+x)^n + xn(1+x)^{n-1}.$$

Evaluando la expresión anterior en $x = 1$, resulta la igualdad

$$\sum_{i=0}^n \binom{n}{i} (i+1) = 2^n + n2^{n-1}.$$

□

Proposición 2.4 *Sea (N, v) un juego cooperativo. El cálculo de la función de potencial de Hart y Mas-Colell para todos los subjuegos del juego (N, v) requiere un tiempo $\Theta(n2^n)$.*

Demostración: El tiempo de ejecución del algoritmo *potencial* es

$$\begin{aligned} t(\text{potencial}) &= 1 + t(\text{bucle1}) \\ &= 1 + \sum_{i=1}^n t(\text{bucle2}) \\ &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} t(\text{asignación}) \\ &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} (1 + t(\text{suma})) \\ &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} \left(1 + \sum_{k=1}^i 1\right) \\ &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} (1 + i) \end{aligned}$$

$$\begin{aligned}
&= 1 + \sum_{i=1}^n (1+i) \binom{n}{i} \\
&= \sum_{i=0}^n (1+i) \binom{n}{i} \\
&= n2^{n-1} + 2^n.
\end{aligned}$$

De aquí, $t(\text{potencial}) = f(n) = n2^{n-1} + 2^n \in \Theta(n2^n)$.

□

Durante el proceso de cálculo del potencial del juego (N, v) , es necesario almacenar los potenciales de todos los subjuegos. Como hay 2^n subjuegos, la siguiente proposición es inmediata.

Proposición 2.5 *Sea (N, v) un juego cooperativo. El cálculo de la función de potencial de Hart y Mas-Colell para todos los subjuegos del juego (N, v) requiere un espacio $\Omega(2^n)$.*

Para la complejidad del cálculo del valor de Shapley, a partir de la función de potencial de Hart y Mas-Colell, se tiene el siguiente resultado.

Proposición 2.6 *Sea (N, v) un juego cooperativo. El cálculo del valor de Shapley de todos los jugadores, utilizando la función de potencial de Hart y Mas-Colell, requiere un tiempo $\Theta(n2^n)$.*

Demostración: En primer lugar, con el algoritmo *potencial*, se obtienen los potenciales de todos los subjuegos en un tiempo $\Theta(n2^n)$. A continuación, para evaluar el valor de Shapley de cada jugador i basta realizar una consulta en la tabla que almacena los potenciales y calcular

$$\Phi_i(v) = P(N, v) - P(N \setminus i, v).$$

Por tanto, como este cálculo se realiza en un tiempo $\Theta(1)$, la complejidad del cálculo del valor de Shapley, para todos los jugadores, viene determinada por la complejidad del cálculo de los potenciales. Es decir, se requiere un tiempo $\Theta(n2^n)$.

□

2.2.2 Dividendos de Harsanyi

En esta sección se describe un algoritmo para el cálculo del valor de Shapley a partir de los dividendos de Harsanyi. Aunque, como se observará más adelante, la complejidad temporal de este algoritmo es $\Theta(3^n)$, se hace una descripción del mismo con el objetivo de mostrar, en el capítulo siguiente, cómo mejora notablemente esta eficiencia en contextos de cooperación parcial.

En el espacio vectorial Γ^N , es conocido que el conjunto de los juegos de unanimidad $\{u_T : T \subseteq N, T \neq \emptyset\}$ constituye una base. Por tanto, cualquier juego $v \in \Gamma^N$ puede expresarse como combinación lineal de ellos, resultando que [51]

$$v = \sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) u_T, \quad \text{con } \Delta_v(T) = \sum_{H \subseteq T} (-1)^{|T|-|H|} v(H).$$

Tal como indica Harsanyi [35], a cada una de las coordenadas del juego v en la base formada por los juegos de unanimidad, $\Delta_v(T)$, se la denomina *dividendo de T en el juego v* , verificándose que

$$v(S) = \sum_{T \subseteq S} \Delta_v(T), \quad \text{para } S \in 2^N \setminus \{\emptyset\}, \text{ y } \Delta_v(\emptyset) = 0.$$

Los dividendos se pueden calcular, para toda coalición no vacía $S \subseteq N$, a partir de la siguiente fórmula recursiva

$$\Delta_v(S) = v(S) - \sum_{T \subset S} \Delta_v(T),$$

y el valor de Shapley [51], para cada $i \in N$, resulta ser

$$\Phi_i(v) = \sum_{\{S \subseteq N: i \in S\}} \frac{\Delta_v(S)}{|S|}.$$

Como consecuencia de las ideas anteriores, se puede establecer un algoritmo denominado *dividendo* que permite, utilizando programación dinámica, determinar los dividendos de Harsanyi del juego (N, v) . La descripción del algoritmo es la siguiente.

Algoritmo *dividendo* (N, v)

$\Delta_v(\emptyset) \leftarrow 0$

```

[ para  $i$  de 1 hasta  $n$ 
  [
    para  $j$  de 1 hasta  $\binom{n}{i}$ 
       $\Delta_v(S_i^j) \leftarrow v(S_i^j) - \sum_{T \subset S_i^j} \Delta_v(T)$ 
    fin_para
  ]
fin_para

```

donde S_i^j es la j -ésima coalición de tamaño i , $|S_i^j| = i$.

Tal como se recoge en la siguiente proposición, la complejidad temporal de este algoritmo es de orden exponencial.

Proposición 2.7 *Sea un juego cooperativo (N, v) . El cálculo de todos los dividendos de Harsanyi requiere un tiempo $\Theta(3^n)$.*

Demostración: De la descripción del algoritmo *dividendo*, se deduce que el tiempo de ejecución verifica

$$t(\text{dividendo}) = 1 + t(\text{bucle1})$$

$$\begin{aligned}
 &= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} t(\text{asignación}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} (1 + t(\text{suma})) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} \left(1 + \sum_{k=1}^{2^i-1} 1\right) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} (1 + 2^i - 1) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{\binom{n}{i}} 2^i \\
 &= 1 + \sum_{i=1}^n 2^i \binom{n}{i} \\
 &= \sum_{i=0}^n 2^i \binom{n}{i} = 3^n.
 \end{aligned}$$

Por tanto, $t(\text{dividendo}) = f(n) = 3^n \in \Theta(3^n)$.

□

Si se considera que los dividendos se calculan mediante un proceso ascendente que exige almacenar los dividendos de cada una de las coaliciones, se obtiene, de forma inmediata, el siguiente resultado.

Proposición 2.8 *Sea un juego cooperativo (N, v) . El cálculo de todos los dividendos de Harsanyi requiere un espacio $\Omega(2^n)$.*

Además, la Proposición 2.7 junto con la expresión del valor de Shapley

del juego (N, v) en función de los dividendos dan, como consecuencia, el resultado siguiente.

Proposición 2.9 *Sea un juego cooperativo (N, v) . El cálculo del valor de Shapley de todos los jugadores, utilizando los dividendos de Harsanyi, requiere un tiempo $\Theta(3^n)$.*

Demostración: En primer lugar, con el algoritmo *dividendo* se calculan todos los dividendos de Harsanyi y se almacenan en una tabla. A continuación, para calcular el valor de Shapley de cada jugador se realiza una consulta en la tabla que almacena los dividendos y se evalúa, para cada $i \in N$, la expresión

$$\Phi_i(v) = \sum_{\{S \subseteq N: i \in S\}} \frac{\Delta_v(S)}{|S|}.$$

Si se tiene en cuenta que cada jugador pertenece a 2^{n-1} coaliciones, el tiempo requerido para evaluar la expresión anterior, para cada jugador, es $\Theta(2^n)$. Para los n jugadores el tiempo sería $\Theta(n2^n)$. Como el cálculo de todos los dividendos se realiza en un tiempo $\Theta(3^n)$, se concluye que la complejidad temporal del cálculo del valor de Shapley para todos los jugadores es $\Theta(3^n)$. \square

2.3 Algoritmos con MATHEMATICA

2.3.1 Cálculo del valor de Shapley

Como ya se ha indicado anteriormente para realizar los cálculos se va a utilizar el sistema MATHEMATICA con los paquetes `DiscreteMath'Combinatorica'` y `Cooperat'Cooperat'` realizados por Skiena [55] y Carter [26]. El paquete `Cooperat'Cooperat'` incluye tres funciones para obtener el valor de Shapley: las dos primeras funciones, denominadas `ShapleyValue1` y `ShapleyValue2`, implementan la fórmula directa del valor de Shapley; la tercera función, llamada

ShapleyValue3, calcula el valor de Shapley a partir de la función potencial de Hart y Mas-Colell.

La función ShapleyValue1 obtiene el valor de Shapley a partir de la fórmula combinatoria

$$\Phi_i(v) = \sum_{\{S \subseteq N: i \in S\}} \gamma(S)[v(S) - v(S \setminus i)], \text{ para todo } i \in N,$$

donde

$$\gamma(S) = \frac{(s-1)!(n-s)!}{n!}.$$

Carter define, usando MATHEMATICA, las siguientes expresiones:

```
In[1] :=
  gamma[S_List]:=(Length[S]-1)!(Length[N]-length[S])! /Length[N]!
```

```
In[2] :=
  ShapleyValue1[game_,i_]:= Plus
  (gamma[#] (v[#]- v[DeleteCases[#,i]]) & /@ Rest[Coalitions])
```

```
In[3] :=
  Attributes[ShapleyValue1]={Listable};
  ShapleyValue1[game_]:= ShapleyValue1[game,N]
```

La función ShapleyValue2 desarrolla un algoritmo específico para el cálculo del coeficiente $\gamma(S)$, ligeramente distinto del anterior, que mejora el tiempo de cálculo. Ahora, en este caso, la función $\gamma(S)$ y el valor de Shapley se escriben en MATHEMATICA del siguiente modo

```
In[4] :=
  g[s_Integer]:=g[s]=Module[{n=Lentgh[N]},(s-1)! (n-s)!/n!]
```

```
In[5] :=
```

```
ShapleyValue2[game_,i_]:= Plus @@
(v[Length[#]] (v[#] -v[DeleteCases[#,i]] )& /@ Rest[Coalitions])
```

```
In[6] :=
Attributes[ShapleyValue2]={Listable};
ShapleyValue2[game_]:= ShapleyValue2[game,N]
```

El tercer algoritmo que suministra Carter viene dado por la función `ShapleyValue3`, y calcula el valor de Shapley utilizando la función de potencial de Hart y Mas-Colell. La función potencial y el valor de Shapley se definen del siguiente modo

```
In[7] :=
p[{i_}]:= p[{i}]=v[{i}];
p[S_]:= p[S] = (v[S]
+Plus @@ Map[p[Complement[S,{#}] @, S])/ Length[S] }
```

```
In[8] :=
ShapleyValue3[game_,List:N]:= Module[{value},
value=p[N]-(p[DeleteCases[N,#]] & /@ S); p[#] = . & /@
Rest[Coalitions];
Return[value] ]
```

```
In[9] :=
ShapleyValue3[game_] := ShapleyValue3[game,N];
ShapleyValue3[ ] := ShapleyValue3[Null,N];
ShapleyValue3[game_:Null,i_?AtomQ] := ShapleyValue3[game,{i}]
```

En los siguientes ejemplos se ilustra cómo se introduce un juego en MATHEMATICA y cómo se calcula el valor de Shapley con las funciones anteriormente descritas. En el primero de ellos se utiliza únicamente la función `ShapleyValue3`. En los otros ejemplos, se comparan los tiempos de ejecución de

las tres funciones incluidas en el paquete `Cooperat'Cooperat'` que permiten calcular el valor de Shapley, determinando el índice de Shapley-Shubik en el juego de votación correspondiente al Consejo de Gobierno de la Unión Europea. En todos los ejemplos hay que cargar previamente los paquetes `DiscreteMath'Combinatorica'` y `Cooperat'Cooperat'`.

Ejemplo 2.10 Sea el juego (N, v) , donde $N = \{1, 2, 3\}$ y $v(S) = |S| - 1$, para toda coalición no vacía $S \subseteq N$.

El número de jugadores y la función característica del juego se introducen del siguiente modo.

```
In[1] :=
  Juego:=(Clear[v,T];
  T=Range[3];
  v[{}]:=0;
  v[S_]:=Length[S]-1;)
```

y el valor de Shapley se obtiene escribiendo

```
In[2] :=
  ShapleyValue3[Juego]
```

```
Out[2] =
  {2/3, 2/3, 2/3}
```

Ejemplo 2.11 El sistema de votación para el Consejo de Gobierno de la Unión Europea se modela mediante el juego (N, v) , donde

$$N = \{Alemania, Reino Unido, Francia, Italia, España, Holanda, Grecia, Bélgica, Portugal, Suecia, Austria, Dinamarca, Finlandia, Irlanda, Luxemburgo\},$$

$$v = [q ; 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2], \quad 61 \leq q \leq 68.$$

```

In[1] :=
  JuegoUE15[n_Integer]:= (Clear[T,x,y,v];
  T=Range[15];
  x[i_Integer/;1<=i && i<=4]:=10;
  x[5]:=8;
  x[i_Integer/;6<=i && i<=9]:=5;
  x[10]:=4;x[11]:=4;
  x[i_Integer/;12<=i && i<=14]:=3;
  x[15]:=2;
  y[S_List]:=Apply[Plus,x /@S]; v[{}]:=0;
  v[S_/;y[S]>=n]:=1;v[S_]:=0;)

```

A continuación, las salidas de resultados indicarán el valor de Shapley del juego usando las tres funciones y precedidas del tiempo de ejecución.

```

In[2] :=
  N[Timing[ShapleyValue1[JuegoUE15[62]]]]

```

```

Out[2] =
  {1499.3 Second, {0.116667,0.116667,0.116667,0.116667,0.0954601,
  0.0551754,0.0551754,0.0551754,0.0551754,0.0453602,0.0453602,
  0.0352536,0.0352536, 0.0352536,0.0206904}}

```

```

In[3] :=
  N[Timing[ShapleyValue2[JuegoUE15[62]]]]

```

```

Out[3] =
  {1383.19 Second, {0.116667,0.116667,0.116667,0.116667,0.0954601,
  0.0551754,0.0551754,0.0551754,0.0551754,0.0453602,0.0453602,
  0.0352536,0.0352536, 0.0352536,0.0206904}}

```

```

In[4] :=

```

N[Timing[ShapleyValue3[JuegoUE15[62]]]]

Out[4] =
 {112.32 Second, {0.116667, 0.116667, 0.116667, 0.116667, 0.0954601,
 0.0551754, 0.0551754, 0.0551754, 0.0551754, 0.0453602, 0.0453602,
 0.0352536, 0.0352536, 0.0352536, 0.0206904}}

En el siguiente ejemplo se calcula el índice de poder de Shapley-Shubik en el juego de votación correspondiente al Congreso de los Diputados de España. El cálculo se hace con las tres funciones incorporadas en el paquete Cooperat'Cooperat' y con la función ShapleyValue4, la cual se introduce en este trabajo y que efectúa el cálculo del valor de Shapley a partir de los dividendos con el algoritmo descrito en la sección 2.2.2 de este capítulo.

Ejemplo 2.12 *La actual composición del Congreso de los Diputados de España, derivada de las elecciones de 1996, permite considerar el conjunto de jugadores*

$$N = \{PP, PSOE, IU, CiU, PNV, CC, BNG, HB, ERC, EA, UV\},$$

estando el juego representado por el siguiente esquema de votación

$$[176; 156, 141, 21, 16, 5, 4, 2, 2, 1, 1, 1].$$

El cálculo efectivo se realiza con MATHEMATICA utilizando una técnica similar a la empleada anteriormente para la Unión Europea.

Definición de la función característica del juego, previa introducción del número de jugadores.

```
In[1] :=
  JuegoCongreso[n_Integer]:= (Clear[T,x,y,v];
  T=Range[11];
```

```

x[1]:=156;
x[2]:=141;
x[3]:=21;
x[4]:=16;
x[5]:=5;
x[6]:=4;
x[i_Integer/;7<=i && i<=8]:=2;
x[i_Integer/;9<=i && i<=11]:=1;
y[S_List]:=Apply[Plus,x/@S]; v[{}]:=0;
v[S_/:y[S]>=n]:=1;v[S_]:=0;)

```

Cálculo de los índices de poder, indicando los tiempos de ejecución para cada una de las funciones.

```

In[2] :=
N[Timing[ShapleyValue1[JuegoCongreso[176]]]]

```

```

Out[2] =
{46.58 Second, {0.452092, 0.183838, 0.183838,
0.118759, 0.0171717, 0.0171717, 0.00804473,
0.00804473, 0.00367965, 0.00367965, 0.00367965}}

```

```

In[3] :=
N[Timing[ShapleyValue2[JuegoCongreso[176]]]]

```

```

Out[3] =
{38.72 Second, {0.452092, 0.183838, 0.183838,
0.118759, 0.0171717, 0.0171717, 0.00804473,
0.00804473, 0.00367965, 0.00367965, 0.00367965}}

```

```

In[4] :=
N[Timing[ShapleyValue3[JuegoCongreso[176]]]]

```

```
Out[4] =
{4.83 Second, {0.452092, 0.183838, 0.183838, 0.118759,
0.0171717, 0.0171717, 0.00804473, 0.00804473,
0.00367965, 0.00367965, 0.00367965}}
```

```
In[5] :=
div[{i_}]:=div[{i}]=v[{i}];
div[S_]:=div[S]=
v[S]-(Apply[Plus,div[#]& /@
Drop[Intersection[Subsets[S],Coalitions],-1]]);
```

```
In[6] :=
ShapleyValue4[game_:=Null]:=Module[{value},
value=Table[Apply[Plus,
(div[#]/Length[#])& /@
Select[F,(MemberQ[#,i])&]],{i,Length[T]}];
div[#]=. & /@ Rest[Coalitions];
Return[value];
```

```
Out[6] =
{47.56 Second, {0.452092, 0.183838, 0.183838, 0.118759, 0.0171717,
0.0171717, 0.00804473, 0.00804473, 0.00367965, 0.00367965,
0.00367965}}
```

Como era de esperar la función `ShapleyValue4` no proporciona una mejora en el tiempo de cálculo. Ahora bien, como se probará en el siguiente capítulo, dicha función puede utilizarse, de manera directa, para calcular el valor de Shapley en los que se denominarán juegos restringidos, en el caso en el que la cooperación parcial esté modelada por grafos e hipergrafos, ya que, en esta situación, bastará considerar únicamente las coaliciones que puedan formarse en el juego.

Capítulo 3

Algoritmos para el cálculo de valores en juegos restringidos

3.1 Juegos restringidos por situaciones de comunicación

En principio, en el estudio de los juegos cooperativos, se supone que cualquiera de los jugadores quiere cooperar con los demás o, en otro caso, el juego se desarrollará en forma no cooperativa. Es decir, será posible formar cualquier coalición entre jugadores y, por tanto, existirá una cooperación universal entre todos ellos.

El interés por el estudio de juegos de n personas en los que se incorporan restricciones a las coaliciones entre los jugadores, se inicia con el modelo de Aumann y Maschler [5] sobre *juegos con estructuras de coalición*. En esta aproximación a una cooperación más restringida o limitada, los jugadores son distribuidos formando una partición del conjunto de los mismos, $\mathcal{B} = \{B_1, B_2, \dots, B_k\}$, denominada *estructura de coalición*. Así, dada una estructura de coalición, las relaciones entre jugadores pueden realizarse, únicamente, dentro de las coaliciones que constituyen la estructura.

Las estructuras de coalición no pueden emplearse en aquellas situaciones en las que la relación entre los jugadores no sea transitiva. Por ello, Myerson, en su trabajo *Graphs and Cooperation in Games* [42], propone un nuevo punto de vista para modelar la conducta cooperativa entre los jugadores. Dado un juego (N, v) , Myerson le asocia un grafo de cooperación $G = (N, E)$ cuyo conjunto de vértices N es el formado por todos los jugadores y cuyo conjunto de aristas no ordenadas E viene dado por los acuerdos bilaterales entre los jugadores. El grafo indica las posibilidades de comunicación entre parejas de jugadores y lleva implícito que no todas las coaliciones son factibles. Así, serán consideradas *coaliciones factibles* aquellas coaliciones de jugadores que son conexas en el grafo; es decir, una coalición de jugadores va a ser factible si, dados dos elementos cualesquiera de la misma, existe un camino en el grafo que los relaciona y que está completamente incluido en dicha coalición. Por tanto, dado el grafo de cooperación $G = (N, E)$, el conjunto de coaliciones factibles, que se denotará por \mathcal{F} , es

$$\begin{aligned}\mathcal{F} &= \{ S \subseteq N : S \text{ es conexo en } G \} \\ &= \{ S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G \},\end{aligned}$$

donde se considera siempre que la coalición vacía es conexo.

Teniendo en cuenta lo anterior, la cooperación parcial entre jugadores viene determinada por una terna (N, v, G) que, habitualmente, es denominada *situación de comunicación* y, para Myerson, la existencia de coaliciones factibles y de otras que no lo son, obliga a que la función característica del juego (N, v) tenga que ser modificada, dando lugar a lo que se denomina *juego restringido por el grafo de cooperación*, el cual se define a continuación.

Definición 3.1 *Sea la terna (N, v, G) , donde G es un grafo y (N, v) un juego de utilidad transferible. Se denomina juego restringido por el grafo de cooperación G , al par (N, v^G) con*

$$v^G : 2^N \longrightarrow \mathbb{R}, \quad v^G(S) = \sum_{T \in G/S} v(T),$$

donde la suma está extendida a todas las componentes conexas maximales del subgrafo inducido por la coalición S .

Obsérvese que si $S \subseteq N$ es una coalición factible, entonces $v^G(S) = v(S)$. Por tanto, si G fuera el grafo completo de cooperación K_n , resultaría que $(N, v^G) = (N, v)$.

En el siguiente ejemplo se describe una situación de comunicación junto con el juego restringido correspondiente.

Ejemplo 3.2 Sea la situación de comunicación (N, v, G) , con (N, v) tal que, para cualquier coalición no vacía $S \subseteq N$, $v(S) = |S|^2 - 1$ y $G = (N, E)$ es el grafo de cooperación entre los jugadores del conjunto $N = \{1, 2, 3, 4, 5\}$, siendo $E = \{\{1, 2\}, \{2, 3\}, \{4, 5\}\}$.

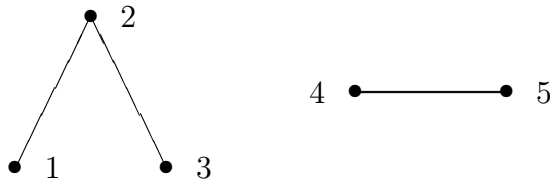


FIGURA 3.1

El conjunto de coaliciones factibles viene dado por

$$\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 2\}, \{2, 3\}, \{4, 5\}, \{1, 2, 3\}\}$$

y los valores de la función característica asociada al juego restringido por el grafo de cooperación, $v^G : 2^N \rightarrow \mathbb{R}$, son

$$v^G(\{i\}) = 0, \text{ para todo } i \in N.$$

$$v^G(\{1, 2\}) = v^G(\{2, 3\}) = v^G(\{4, 5\}) = 3,$$

$$v^G(\{i, j\}) = 0, \text{ en otro caso.}$$

$$v^G(\{1, 2, 3\}) = 8, v^G(\{1, 3, 4\}) = v^G(\{1, 3, 5\}) = 0,$$

$$v^G(\{i, j, k\}) = 3, \text{ en otro caso.}$$

$$v^G(\{1, 2, 3, 4\}) = v^G(\{1, 2, 3, 5\}) = 8,$$

$$v^G(\{1, 2, 4, 5\}) = v^G(\{2, 3, 4, 5\}) = 6, v^G(\{1, 3, 4, 5\}) = 3,$$

$$v^G(N) = 11.$$

Este modelo, que se utiliza en problemas de asignación de costes/beneficios y redes de comunicación, ha suscitado una línea de investigación desarrollada en los trabajos de Owen [44], Borm [16], van den Nouweland y Borm [58], Carreras [25], Borm, Owen y Tijs [18], Borm, van den Nouweland, Owen y Tijs [17], Borm, van den Nouweland y Tijs [20], Potters y Reijnierse [45], Grafe, Mauleón e Iñarra [33] y Bilbao y López [12], entre otros. Trabajos relacionados también con esta línea, pueden ser encontrados en Rosenthal [48], [49] y Voshtina [62]. Por otra parte, la formación endógena de coaliciones en situaciones de comunicación es analizada por Aumann y Myerson [6] y van den Nouweland [59]. Finalmente, otros modelos que guardan una estrecha relación con el planteado inicialmente por Myerson, en tanto que las relaciones entre los jugadores se modela mediante un grafo no dirigido, son estudiados por Bergantiños, Carreras y García-Jurado [9], Calvo y Lasaga [24].

Estos trabajos sobre las situaciones de comunicación de Myerson han seguido, básicamente, tres direcciones de estudio.

1. Propiedades del juego (N, v) que se transmiten o son heredadas por el juego restringido (N, v^G) .

2. Conceptos de solución de conjunto para el juego restringido (N, v^G) y existencia de relaciones entre éstas y las del juego (N, v) .
3. Conceptos de valores o reglas de asignación de pagos para las situaciones de comunicación.

Obviamente, presentar todos los resultados obtenidos en los campos de estudio indicados para las situaciones de comunicación es una tarea que escapa de los objetivos que se plantean en este trabajo. No obstante, a continuación, se indican brevemente algunos de los resultados más interesantes relacionados con las reglas de asignación de pagos para las situaciones de comunicación.

En el capítulo anterior se puso de manifiesto que uno de los conceptos más clásicos en la búsqueda de soluciones para un juego cooperativo era el valor de Shapley. Entonces es lógico que, cuando se abordan ideas de cooperación parcial modeladas a través de situaciones de comunicación, se piense en utilizar el valor de Shapley asociado a la función característica del juego restringido por el grafo de cooperación. A su definición, como regla de asignación, y a sus propiedades se hace referencia a continuación.

Sea (N, v) un juego cooperativo y sea SC^N el conjunto de todas las situaciones de comunicación definidas sobre el conjunto N , es decir:

$$SC^N = \{(N, v, G) : v \in \Gamma^N, G \text{ es un grafo}\}.$$

Se denomina *regla de asignación para el juego v* a cualquier función

$$\gamma : SC^N \rightarrow \mathbb{R}^n, (N, v, G) \mapsto (\gamma_1(N, v, G), \gamma_2(N, v, G), \dots, \gamma_n(N, v, G)),$$

tal que

$$\forall (N, v, G) \in SC^N, \forall S \in G/N, \sum_{k \in S} \gamma_k(N, v, G) = v(S).$$

Es decir, una regla de asignación para el juego v es una función que a cada situación de comunicación (N, v, G) le asigna un vector n -dimensional que es eficiente para las coaliciones factibles maximales de la coalición N .

La regla de asignación, $\gamma : SC^N \rightarrow \mathbb{R}^n$, es *justa* si, para cualquier $(N, v, G) \in SC^N$ y cada arista $\{i, j\} \in E$, se tiene

$$\gamma_j(N, v, G) - \gamma_j(N, v, G \setminus \{i, j\}) = \gamma_i(N, v, G) - \gamma_i(N, v, G \setminus \{i, j\}).$$

La regla de asignación es *estable* si, para cualquier $(N, v, G) \in SC^N$ y cada arista $\{i, j\} \in E$,

$$\gamma_j(N, v, G) \geq \gamma_j(N, v, G \setminus \{i, j\}), \quad \gamma_i(N, v, G) \geq \gamma_i(N, v, G \setminus \{i, j\}).$$

Definición 3.3 Sea (N, v, G) una situación de comunicación. El valor de Myerson de (N, v, G) , denotado por $\mu(N, v, G) \in \mathbb{R}^n$, es el valor de Shapley asociado al correspondiente juego restringido por el grafo de cooperación; esto es,

$$\mu(N, v, G) = \Phi(v^G).$$

Los siguientes teoremas ponen de manifiesto que el valor de Myerson es la única regla de asignación justa que puede definirse en el conjunto de las situaciones de comunicación asociadas a un juego (N, v) , y que, en el caso de que el juego (N, v) sea superaditivo es, además, una regla de asignación estable. Las pruebas originales de Myerson pueden consultarse en su trabajo *Graphs and Cooperation in Games*.

Teorema 3.4 (Myerson (1977)) Dado un juego (N, v) , existe una única regla de asignación justa $\gamma : SC^N \rightarrow \mathbb{R}^n$, dada por $\gamma(N, v, G) = \mu(N, v, G)$.

Teorema 3.5 (Myerson (1977)) Si el juego (N, v) es superaditivo entonces el valor de Myerson, $\mu(N, v, G)$, es una regla de asignación justa y estable.

Borm, van den Nouweland y Tijs han dado otras caracterizaciones del valor de Myerson que puede consultarse en el libro *Imperfections and Behavior in Economic Organizations*.

Si $G = (N, E)$ es un grafo de cooperación, puede definirse una aplicación de Γ^N en sí mismo, que se denotará por L_G [44], que asocia a cada juego v el correspondiente juego restringido

$$L_G : \Gamma^N \longrightarrow \Gamma^N, \quad L_G(v) = v^G.$$

Si G fuera el grafo completo de cooperación K_n , resultaría que $\mathcal{F} = 2^N$, $v^G = v$ y L_G sería la aplicación identidad. Por tanto, se considerará que G no es un grafo completo.

Dado que Γ^N es un espacio vectorial de dimensión $2^n - 1$ y que el juego restringido por G está definido sobre una suma extendida a las componentes conexas maximales de cada coalición, L_G es un endomorfismo en Γ^N .

Proposición 3.6 *El operador L_G es lineal y no biyectivo.*

Demostración: Es fácil ver, por definición de juego restringido, que

$$L_G(\alpha v + \beta w) = \alpha L_G(v) + \beta L_G(w), \quad \forall \alpha, \beta \in \mathbb{R}, \quad \forall v, w \in \Gamma^N.$$

En efecto, para cualquier $S \subseteq N$,

$$\begin{aligned} (\alpha v + \beta w)^G(S) &= \sum_{T \in G/S} (\alpha v + \beta w)(T) \\ &= \alpha \sum_{T \in G/S} v(T) + \beta \sum_{T \in G/S} w(T) \\ &= (\alpha v^G + \beta w^G)(S). \end{aligned}$$

La aplicación L_G no es biyectiva ya que su núcleo no queda reducido al juego nulo ($\theta(S) = 0, \forall S \subseteq N$). Como $\mathcal{F} \subset 2^N$ ($\mathcal{F} \neq 2^N$, al ser G no completo), existen subconjuntos de N que no son coaliciones factibles. De ahí, el juego definido, para toda $S \subseteq N$, por

$$v(S) = \begin{cases} 0, & \text{si } S \text{ es conexa en } G \\ 1, & \text{en otro caso,} \end{cases}$$

es no nulo, y su imagen es el juego nulo; es decir, para cualquier $S \subseteq N$,

$$v^G(S) = \sum_{T \in G/S} v(T) = 0.$$

□

Los siguientes resultados permiten expresar la función característica del juego restringido en función de los dividendos de éste y de los juegos de unanimidad de las coaliciones conexas en el grafo de cooperación.

Teorema 3.7 (*Owen (1986)*) *Sea la terna (N, v, G) , donde G es un grafo y (N, v) un juego de utilidad transferible. Entonces, el conjunto de los juegos de unanimidad $\{u_T : T \text{ conexas en } G, T \neq \emptyset\}$ constituye una base del espacio $L_G(\Gamma^N)$.*

Como consecuencia de este resultado, el juego restringido puede expresarse como una combinación lineal de los juegos de unanimidad correspondientes a las coaliciones conexas en el grafo G ,

$$v^G = \sum_{\{T \text{ conexas: } T \neq \emptyset\}} \Delta_{v^G}(T) u_T, \quad \text{con } \Delta_{v^G}(\emptyset) = 0.$$

Se verifica, para toda coalición conexas S , que

$$v(S) = v^G(S) = \sum_{\{T \text{ conexas: } T \subseteq S\}} \Delta_{v^G}(T),$$

y, a partir de esta expresión, se pueden calcular, para cualquier coalición S conexas, los dividendos del juego restringido para todas las coaliciones conexas con la fórmula recurrente

$$\Delta_{v^G}(S) = v(S) - \sum_{\{T \text{ conexas: } T \subset S\}} \Delta_{v^G}(T).$$

Esta fórmula es similar a la dada anteriormente, en la sección 2.2.2, para el cálculo de los dividendos del juego (N, v) . La única diferencia radica en el

hecho de que, en este caso, sólo se calculan los dividendos para las coaliciones conexas y supone una alternativa a la propuesta hecha por Owen [44] para calcular los dividendos del juego restringido en función de los dividendos del juego (N, v) dado.

Utilizando el algoritmo *dividendo*, descrito en la sección 2.2.2, se tienen los siguientes resultados para las complejidades espacial y temporal del cálculo de dividendos del juego restringido.

Proposición 3.8 *Sea la terna (N, v, G) , donde G es un grafo y (N, v) un juego de utilidad transferible. El cálculo de los dividendos del juego restringido requiere un espacio $\Omega(|\mathcal{F}|)$, donde \mathcal{F} es el conjunto de coaliciones conexas en el grafo G .*

Demostración: El resultado es inmediato pues sólo es necesario almacenar los dividendos de todas las coaliciones conexas, cuyo número es $|\mathcal{F}|$.

□

Proposición 3.9 *Sea la terna (N, v, G) , donde G es un grafo y (N, v) un juego de utilidad transferible. El cálculo de los dividendos del juego restringido requiere un tiempo $\mathcal{O}(3^n)$.*

Demostración: Teniendo en cuenta que únicamente es necesario calcular los dividendos de las coaliciones conexas, cuyo número es $|\mathcal{F}| \leq 2^n$, y que el número de coaliciones de tamaño i , denotado por $S(i)$, verifica la desigualdad $S(i) \leq \binom{n}{i}$, para $i = 1, \dots, n$, el resultado se deduce de manera inmediata.

□

Dada una situación de comunicación (N, v, G) el valor de Myerson en función de los dividendos del juego restringido, para cada jugador $i \in N$, viene dado por

$$\mu_i(N, v, G) = \Phi(v^G) = \sum_{\{S \text{ conexas: } i \in S\}} \frac{\Delta_{v^G}(S)}{|S|}.$$

Para situaciones de comunicación en las que el grafo G no es conexo, van den Nouweland, en [59], establece que el valor de Myerson de un jugador puede determinarse considerando únicamente la situación de comunicación derivada de la componente maximal a la que pertenezca el jugador y al correspondiente subjuego inducido. Es decir, el valor de Myerson de un jugador i , perteneciente a la j -ésima componente maximal, viene dado por

$$\mu_i(N, v, G) = \mu_i(M_j, v_{M_j}, G/M_j), \text{ donde } M_j \in G/M_j, i \in M_j.$$

Este método de cálculo del valor de Myerson a partir de los dividendos puede ser útil, en determinadas situaciones de comunicación, para estudiar juegos restringidos por grafos de comunicación ya que evita obtener la función característica del juego restringido así como las componentes conexas maximales de cada coalición. De hecho, en algunos casos analizados, mejora sensiblemente los tiempos de ejecución con respecto a otros algoritmos que calculan el valor de Myerson a partir de la función característica del juego restringido o mediante los potenciales restringidos [39][13][12].

A continuación, se analiza la complejidad del cálculo del valor de Myerson en dos situaciones de comunicación en las cuales el grafo de cooperación tiene una estructura particular. En el primer caso, se estudiará la complejidad del cálculo del valor de Myerson en una situación de comunicación en la que el grafo es un camino, demostrándose que tiene una complejidad temporal polinómica. En el segundo caso, se analizará la complejidad del cálculo del valor de Myerson cuando la situación de comunicación está modelada mediante un grafo estrella, probándose que la complejidad temporal es exponencial.

Sea la terna (N, v, G) , donde G es un grafo camino con n vértices y (N, v) un juego de utilidad transferible. Si se enumeran los vértices de izquierda a derecha, el conjunto \mathcal{F} de coaliciones factibles viene determinado por

$$\mathcal{F} = \{[i, j], 1 \leq i \leq j \leq n\} \cup \{\emptyset\},$$

donde $[i, j] = \{k \in N : i \leq k \leq j\}$.

Obsérvese que, exceptuando el conjunto vacío, se pueden disponer las coaliciones factibles como entradas de una matriz $A \in \mathcal{M}_n(\mathbb{R})$ triangular superior de modo que las coaliciones unitarias se almacenen en la diagonal. En la fila k , para $k = 1, \dots, n$, se almacenarían ordenadamente las coaliciones del conjunto $\{[k, j], k \leq j \leq n\}$. El jugador k pertenece a todas las coaliciones de la submatriz (a_{ij}) , para $i = 1, \dots, k$ y $j = k, \dots, n$.

Previo al análisis de la complejidad del cálculo del valor de Myerson, a partir de la fórmula de los dividendos, se describen algunos resultados.

Lema 3.10 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El número de coaliciones factibles con i jugadores es $S(i) = n - i + 1$.*

Demostración: Teniendo en cuenta la disposición de las coaliciones factibles en la matriz $A \in \mathcal{M}_n(\mathbb{R})$, triangular superior, como se ha indicado anteriormente, se puede observar que en la i -ésima superdiagonal, que posee un número de entradas igual a $(n - i + 1)$, estarían almacenadas todas las coaliciones de i jugadores. Por tanto, el número de coaliciones factibles con i jugadores es $S(i) = n - i + 1$.

□

Lema 3.11 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El número total de coaliciones factibles es $|\mathcal{F}| = \frac{n^2 + n + 2}{2}$.*

Demostración: Obviamente $|\mathcal{F}| = 1 + \sum_{i=1}^n S(i)$ y, teniendo en cuenta el lema anterior, se obtiene, de forma inmediata,

$$|\mathcal{F}| = 1 + \sum_{i=1}^n S(i) = 1 + \sum_{i=1}^n (n - i + 1) = 1 + \frac{n^2 + n}{2} = \frac{n^2 + n + 2}{2}.$$

□

Lema 3.12 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El número total de coaliciones factibles a las que pertenece el jugador k es $F_k = k(n - k + 1)$.*

Demostración: El jugador k pertenece a todas las coaliciones del conjunto $\{[k, j], k \leq j \leq n\}$, cuyo número es $(n - k + 1)$. También pertenece a igual número de coaliciones del conjunto $\{[i, j], i \leq j \leq n\}$, para $i = 1, \dots, k - 1$. Por tanto, el número total de coaliciones a las que pertenece el jugador k es $F_k = k(n - k + 1)$.

□

Ejemplo 3.13 *Sea un grafo camino con 5 vértices. Si se enumeran los vértices de izquierda a derecha, las coaliciones factibles son*

$$\mathcal{F} = \{[i, j], 1 \leq i \leq j \leq 5\} \cup \{\emptyset\}.$$

Las coaliciones no vacías se pueden almacenar como entradas de una matriz triangular superior tal y como se indica a continuación.

$$\begin{pmatrix} 1 & 12 & 123 & 1234 & 12345 \\ & 2 & 23 & 234 & 2345 \\ & & 3 & 34 & 345 \\ & & & 4 & 45 \\ & & & & 5 \end{pmatrix}$$

Obsérvese que el número de coaliciones factibles con i jugadores es

$$S(i) = 5 - i + 1 = 6 - i, \text{ para } i = 1, \dots, 5.$$

Asimismo, el número de coaliciones a las que pertenece un jugador k viene dado por $F_k = k(5 - k + 1) = k(6 - k)$, para $k = 1, \dots, 5$. La siguiente tabla recoge el número de coaliciones a las que pertenece cada jugador.

Jugador	nº de coaliciones
1	5
2	8
3	9
4	8
5	5

Lema 3.14 Sea n cualquier número natural y sea $F_k = k(n - k + 1)$, para $k = 1, \dots, n$. Se verifica que

$$n \leq F_k \leq \frac{1}{4}(n+1)^2.$$

Demostración: Se tiene que $F_k = k(n+1-k) = \frac{1}{4}(n+1)^2 - (\frac{1}{2}(n+1)-k)^2$ y es inmediato que, para cualquier valor de k , se verifica que $F_k \leq \frac{1}{4}(n+1)^2$. El valor mínimo de F_k se tiene cuando $(\frac{1}{2}(n+1)-k)^2$ alcanza el valor máximo, y este valor será máximo en el caso $k = 1$, para el cual se tiene que $F_k = n$. \square

Obsérvese, además, que el valor mínimo de F_k se alcanza también para $k = n$. Es decir, $F_1 = F_n = n$. El valor máximo de F_k se alcanza para los valores centrales de k distinguiéndose dos casos: si n es impar, $n = 2p + 1$ y el valor central corresponde a $k = p + 1$, con lo que $F_{p+1} = \frac{1}{4}(n+1)^2$; si n es par, $n = 2p$ y los valores centrales son $k = p$ y $k = p + 1$, resultando en esta situación que $F_p = F_{p+1} < \frac{1}{4}(n+1)^2$.

Lema 3.15 Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. Si T es una coalición factible, con $|T| = i$,

el número total de coaliciones factibles contenidas estrictamente en T viene dado por $C(i) = \frac{i^2 + i}{2}$.

Demostración: Sea T una coalición factible tal que $|T| = i$. El número de coaliciones factibles del correspondiente subgrafo camino viene dado por $\frac{i^2 + i + 2}{2}$. Si se excluye a la propia coalición T se obtiene $C(i) = \frac{i^2 + i}{2}$. \square

Proposición 3.16 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\Theta(n^4)$.*

Demostración: Para el cálculo de los dividendos se utiliza el algoritmo *dividendo* descrito en la sección 2.2.2. Como ahora se consideran únicamente las coaliciones factibles, el número de coaliciones de tamaño i viene dado por $S(i) = n - i + 1$, y el número total de coaliciones factibles contenidas estrictamente en una coalición factible T de tamaño i es $C(i) = \frac{i^2 + i}{2}$. Por tanto, el tiempo de ejecución del algoritmo es

$$\begin{aligned}
 t(\text{dividendo}) &= 1 + t(\text{bucle1}) \\
 &= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{asignación}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} (1 + t(\text{suma})) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} \left(1 + \sum_{k=1}^{C(i)} 1\right) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} (1 + C(i))
 \end{aligned}$$

$$\begin{aligned}
&= 1 + \sum_{i=1}^n \left(1 + \frac{i^2 + i}{2}\right) S(i) \\
&= 1 + \sum_{i=1}^n \left(\frac{i^2 + i + 2}{2}\right) (n - i + 1) \\
&= 1 + \frac{1}{2} \sum_{i=1}^n (i^2 + i + 2)(n - i + 1) \\
&= 1 + \frac{1}{2} \sum_{i=1}^n [(2 + 2n) + (n - 1)i + ni^2 - i^3] \\
&= 1 + \frac{1}{2} \left((2 + 2n) \sum_{i=1}^n 1 + (n - 1) \sum_{i=1}^n i + n \sum_{i=1}^n i^2 - \sum_{i=1}^n i^3 \right) \\
&= 1 + \frac{1}{2} \left((2 + 2n)n + (n - 1) \frac{n^2 + n}{2} + n \frac{n(n + 1)(2n + 1)}{6} - \frac{n^2(n + 1)^2}{4} \right) \\
&= 1 + \frac{1}{2} \left(2n + 2n^2 + \frac{n^3 - n}{2} + \frac{(n^3 + n^2)(2n + 1)}{6} - \frac{(n^4 + 2n^3 + n^2)}{4} \right) \\
&= 1 + \frac{1}{2} \left(2n + 2n^2 + \frac{n^3 - n}{2} + \frac{2n^4 + 3n^3 + n^2}{6} - \frac{n^4 + 2n^3 + n^2}{4} \right) \\
&= 1 + \frac{1}{2} \frac{24n + 24n^2 + 6n^3 - 6n + 4n^4 + 6n^3 + 2n^2 - 3n^4 - 6n^3 - 3n^2}{12} \\
&= \frac{n^4 + 6n^3 + 23n^2 + 18n + 24}{24}.
\end{aligned}$$

Por tanto, $t(\text{dividendo}) = f(n) \in \Theta(n^4)$.

□

Proposición 3.17 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un espacio $\Omega(n^2)$.*

Demostración: El proceso de cálculo exige almacenar los dividendos correspondientes a todas las coaliciones factibles cuyo número es $\frac{n^2 + n + 2}{2}$. Por tanto, el espacio requerido es $\Omega(n^2)$.

□

De la expresión del valor de Myerson en función de los dividendos del juego restringido se deduce, usando programación dinámica, el siguiente resultado.

Proposición 3.18 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson de todos los jugadores requiere un tiempo $\Theta(n^4)$.*

Demostración: En primer lugar, con el algoritmo *dividendo* se calculan todos los dividendos de Harsanyi. A continuación, para calcular el valor de Myerson del jugador k , hay que considerar que este jugador pertenece a $F_k = k(n - k + 1)$ coaliciones. Como

$$n \leq F_k \leq \frac{1}{4}(n+1)^2,$$

se tiene que el tiempo requerido para evaluar la suma en la expresión del valor de Myerson, para cada jugador, es $\mathcal{O}(n^2)$. Por tanto, la complejidad del cálculo del valor de Myerson viene determinada por la complejidad del cálculo de los dividendos. Es decir, se requiere un tiempo $\Theta(n^4)$.

□

A continuación, tal y como se había indicado anteriormente, se estudia la complejidad del cálculo del valor de Myerson, a partir de los dividendos de Harsanyi, cuando la situación de comunicación viene determinada por un grafo estrella.

Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. Si n es el vértice central, entonces el conjunto \mathcal{F} de coaliciones factibles viene determinado por

$$\mathcal{F} = \{\{n\} \cup S : S \in 2^{N \setminus \{n\}}\} \cup \{\{1\}, \dots, \{n-1\}\} \cup \{\emptyset\}.$$

Antes de analizar la complejidad del cálculo del valor de Myerson a partir de la fórmula de los dividendos se describen algunos resultados.

Lema 3.19 Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El número de coaliciones factibles con i jugadores, siendo $i \geq 2$, viene dado por

$$S(i) = \binom{n-1}{i-1}.$$

Demostración: Para $i \geq 2$, el número de coaliciones con i jugadores viene determinado por el número de coaliciones con $(i-1)$ jugadores en el conjunto $2^{N \setminus \{n\}}$. Este número es $\binom{n-1}{i-1}$ y, de aquí, $S(i) = \binom{n-1}{i-1}$. □

Lema 3.20 Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El número total de coaliciones factibles viene dado por $|\mathcal{F}| = n + 2^{n-1}$.

Demostración: Obviamente $|\mathcal{F}| = 1 + \sum_{i=1}^n S(i)$ y, teniendo en cuenta el Lema 3.19, se obtiene de forma inmediata

$$\begin{aligned} |\mathcal{F}| &= 1 + \sum_{i=1}^n S(i) \\ &= 1 + n + \sum_{i=2}^n \binom{n-1}{i-1} \\ &= 1 + n + \sum_{l=1}^{n-1} \binom{n-1}{l} \\ &= 1 + n + \sum_{l=0}^{n-1} \binom{n-1}{l} - 1 \\ &= n + 2^{n-1}. \end{aligned}$$

□

Nótese que este resultado se podría haber obtenido teniendo en cuenta como está constituido el conjunto \mathcal{F} . Así $|\mathcal{F}| = 2^{n-1} + (n-1) + 1 = n + 2^{n-1}$.

Lema 3.21 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El número total de coaliciones factibles a las que pertenece un jugador viene dado por*

$$F_k = \begin{cases} (n-1) + 2^{n-2}, & \text{si } k = 1, \dots, n-1 \\ 2^{n-1}, & \text{si } k = n, \text{ siendo } n \text{ el jugador central.} \end{cases}$$

Demostración: Si un jugador k no es central, entonces pertenece a tantas coaliciones como resulta de considerar un subgrafo en el que se ha suprimido el vértice correspondiente a ese jugador. Luego, para $k = 1, \dots, n-1$, se verifica que

$$F_k = (n-1) + 2^{n-2}.$$

Para el jugador central el resultado es inmediato. □

Lema 3.22 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. Si T es una coalición factible con $|T| = i$, el número total de coaliciones factibles contenidas estrictamente en T viene dado por $C(i) = i + 2^{i-1} - 1$.*

Demostración: Sea T una coalición factible de tamaño i . Esta coalición tiene asociado un subgrafo, que también es un grafo estrella, con i vértices. El número de coaliciones factibles correspondiente a este subgrafo es $i + 2^{i-1}$. Si se excluye a la propia coalición T , se obtiene $C(i) = i + 2^{i-1} - 1$. □

Proposición 3.23 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\Theta(3^n)$.*

Demostración: Para el cálculo de los dividendos se utiliza el algoritmo *dividendo* descrito en la sección 2.2.2. Como ahora se consideran únicamente

las coaliciones factibles, el número de coaliciones de tamaño i viene dado por $S(i) = \binom{n-1}{i-1}$, y el número total de coaliciones factibles contenidas estrictamente en una coalición factible T de tamaño i es $C(i) = i + 2^{i-1} - 1$. Por tanto, el tiempo de ejecución del algoritmo es

$$\begin{aligned}
t(\text{dividendo}) &= 1 + t(\text{bucle1}) \\
&= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{asignación}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} (1 + t(\text{suma})) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} \left(1 + \sum_{k=1}^{C(i)} 1\right) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} (1 + i + 2^{i-1} - 1) \\
&= 1 + \sum_{i=1}^n (i + 2^{i-1})S(i) \\
&= 1 + 2n + \sum_{i=2}^n (i + 2^{i-1})S(i) \\
&= 1 + 2n + \sum_{i=2}^n (i + 2^{i-1}) \binom{n-1}{i-1} \\
&= 1 + 2n + \sum_{i=2}^n \binom{n-1}{i-1} i + \sum_{i=2}^n \binom{n-1}{i-1} 2^{i-1} \\
&= 1 + 2n + (2^{n-1} + (n-1)2^{n-2} - 1) + (3^{n-1} - 1) \\
&= 3^{n-1} + (n+1)2^{n-2} + 2n - 1.
\end{aligned}$$

Por tanto, $t(\text{dividendo}) = f(n) \in \Theta(3^n)$.

□

Para la complejidad espacial se tiene el siguiente resultado.

Proposición 3.24 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un espacio $\Omega(2^n)$.*

Demostración: El proceso de cálculo exige almacenar los dividendos correspondientes a todas las coaliciones factibles cuyo número es $2^{n-1} + n$. Por tanto, el espacio requerido es $\Omega(2^n)$.

□

Como consecuencia de la Proposición 3.23 se obtiene el orden de la complejidad temporal del cálculo del valor de Myerson.

Proposición 3.25 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson de todos los jugadores requiere un tiempo $\Theta(3^n)$.*

Demostración: En primer lugar, con el algoritmo *dividendo*, se calculan todos los dividendos de Harsanyi. A continuación, para calcular el valor de Myerson del jugador k , hay que considerar que este jugador pertenece a F_k coaliciones. Como $F_k \leq 2^{n-1}$, se tiene que el tiempo requerido para evaluar la suma en la expresión del valor de Myerson es $\mathcal{O}(2^n)$ para cada jugador. Por tanto, la complejidad del cálculo del valor de Myerson viene determinada por la complejidad del cálculo de los dividendos. Es decir, se requiere un tiempo $\Theta(3^n)$.

□

3.2 El valor de Myerson en juegos restringidos por árboles

Anteriormente, se ha descrito un método para el cálculo del valor de Myerson a partir de los dividendos del juego restringido evitando obtener la función característica del juego restringido, así como las componentes conexas maximales de cada coalición.

Los dividendos del juego restringido son determinantes para calcular el valor de Myerson. Para una situación de comunicación (N, v, G) pueden obtenerse con la siguiente fórmula

$$\Delta_{v^G}(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} v^G(T).$$

Owen, en su trabajo *Values of Graph-Restricted Games*, demuestra que los dividendos del juego restringido se pueden determinar a través de los dividendos del juego (N, v) y de los juegos de unanimidad restringidos.

Teorema 3.26 (Owen (1986)) *Sea la terna (N, v, G) , donde G es un grafo y (N, v) un juego de utilidad transferible. Los dividendos del juego restringido, para toda coalición conexas S , vienen determinados por*

$$\Delta_{v^G}(S) = \sum_{\{T \subseteq S, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^G}(S).$$

Puede observarse que la expresión resultante es, en la práctica, bastante difícil de evaluar. Por esta razón, Owen [44] describe una alternativa para simplificar el cálculo, estudiando los dividendos del juego restringido cuando el grafo es un árbol. De forma análoga, Bilbao y López [12] obtienen simplificaciones significativas que permiten hacer un cálculo directo de los dividendos del juego restringido mediante los valores del juego (N, v) reduciendo así la dificultad del cálculo del valor de Myerson.

Teorema 3.27 (Bilbao y López (1996)). Sea la terna (N, v, G) , donde G es un árbol y (N, v) un juego. Los dividendos del juego restringido (N, v^G) verifican, para toda coalición S conexa,

$$\Delta_{v^G}(S) = \sum_{\{T: S \setminus \text{ex}(S) \subseteq T \subseteq S\}} (-1)^{|S|-|T|} v(T),$$

donde $\text{ex}(S)$ es el conjunto de los puntos extremales de S .

Corolario 3.28 Sea la terna (N, v, G) , donde G es un árbol y (N, v) un juego. El valor de Myerson viene dado, para todo $i \in N$, por

$$\mu_i(N, v, G) = \sum_{\{S \text{ conexa: } i \in S\}} \frac{1}{|S|} \left[\sum_{T \in [S^-, S]} (-1)^{|S|-|T|} v(T) \right],$$

donde, para cada coalición conexa, $S^- = S \setminus \text{ex}(S)$.

Ejemplo 3.29 Sea la terna (N, v, G) , siendo $N = \{1, 2, 3, 4, 5\}$, G el árbol de la Figura 3.2, y el juego v dado por

$$v : 2^N \longrightarrow \mathbb{R}, \quad v(S) = \begin{cases} |S| - 1, & \text{si } S \neq \emptyset \\ 0, & \text{si } S = \emptyset. \end{cases}$$

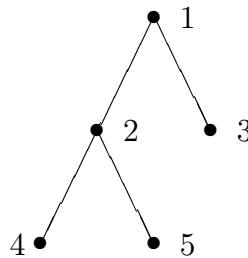


FIGURA 3.2

En este caso, las coaliciones factibles son:

$$\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 3\}, \{1, 2\}, \{2, 4\}, \{2, 5\}, \{1, 2, 3\}, \{1, 2, 4\}, \\ \{1, 2, 5\}, \{2, 4, 5\}, \{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 4, 5\}, \{1, 2, 3, 4, 5\}\}.$$

Utilizando resultados anteriores, el valor de Myerson viene determinado por

$$\mu_i(N, v, G) = \sum_{\{S \text{ conexa: } i \in S\}} \frac{\Delta_{vG}(S)}{|S|}, \text{ para cada } i \in N,$$

$$\text{con } \Delta_{vG}(S) = \sum_{T \in [S^-, S]} (-1)^{|S|-|T|} v(T) \text{ para } S \text{ conexa.}$$

Por tanto, teniendo en cuenta que el intervalo $[S^-, S]$ es un álgebra de Boole en el retículo (\mathcal{F}, \subseteq) , se calculan en primer lugar los dividendos del juego restringido

$$i) \text{ Para todo } i \in N, \Delta_{vG}(\{i\}) = \sum_{T \in [\emptyset, \{i\}]} (-1)^{1-|T|} v(T) = 0.$$

ii) Para las coaliciones $\{i, j\} \in \mathcal{F}$, se tiene

$$\Delta_{vG}(\{i, j\}) = \sum_{T \in [\emptyset, \{i, j\}]} (-1)^{2-|T|} v(T) = v(\{i, j\}) = 1.$$

iii) Para las coaliciones factibles de cardinal 3,

$$\begin{aligned} \Delta_{vG}(\{1, 2, 3\}) &= \sum_{T \in [\emptyset, \{1, 2, 3\}]} (-1)^{3-|T|} v(T) \\ &= -v(\{1, 2\}) - v(\{1, 3\}) + v(\{1, 2, 3\}) \\ &= 0. \end{aligned}$$

Análogamente,

$$\Delta_{vG}(\{1, 2, 4\}) = \sum_{T \in [\emptyset, \{1, 2, 4\}]} (-1)^{3-|T|} v(T) = 0,$$

$$\Delta_{vG}(\{1, 2, 5\}) = \sum_{T \in [\emptyset, \{1, 2, 5\}]} (-1)^{3-|T|} v(T) = 0,$$

$$\Delta_{v^G}(\{2, 4, 5\}) = \sum_{T \in [\{2\}, \{2, 4, 5\}]} (-1)^{3-|T|} v(T) = 0.$$

iv) Para las coaliciones factibles de 4 jugadores,

$$\Delta_{v^G}(\{1, 2, 3, 4\}) = \sum_{T \in [\{1, 2\}, \{1, 2, 3, 4\}]} (-1)^{4-|T|} v(T) = 0,$$

$$\Delta_{v^G}(\{1, 2, 3, 5\}) = \sum_{T \in [\{1, 2\}, \{1, 2, 3, 5\}]} (-1)^{4-|T|} v(T) = 0,$$

$$\Delta_{v^G}(\{1, 2, 4, 5\}) = \sum_{T \in [\{2\}, \{1, 2, 4, 5\}]} (-1)^{4-|T|} v(T) = 0.$$

v) Si $S = N$,

$$\Delta_{v^G}(\{1, 2, 3, 4, 5\}) = \sum_{T \in [\{1, 2\}, \{1, 2, 3, 4, 5\}]} (-1)^{5-|T|} v(T) = 0.$$

De ahí,

$$\begin{aligned} \mu_1(N, v, G) &= \sum_{\{S \text{ conexa: } 1 \in S\}} \frac{\Delta_{v^G}(S)}{|S|} \\ &= \Delta_{v^G}(\{1\}) + \frac{1}{2} [\Delta_{v^G}(\{1, 2\}) + \Delta_{v^G}(\{1, 3\})] \\ &\quad + \frac{1}{3} [\Delta_{v^G}(\{1, 2, 3\}) + \Delta_{v^G}(\{1, 2, 4\}) + \Delta_{v^G}(\{1, 2, 5\})] \\ &\quad + \frac{1}{4} [\Delta_{v^G}(\{1, 2, 3, 4\}) + \Delta_{v^G}(\{1, 2, 3, 5\}) + \Delta_{v^G}(\{1, 2, 4, 5\})] \\ &\quad + \frac{1}{5} [\Delta_{v^G}(\{1, 2, 3, 4, 5\})] \\ &= 1, \end{aligned}$$

y análogamente:

$$\mu_2(N, v, G) = \sum_{\{S \text{ conexa: } 2 \in S\}} \frac{\Delta_{v^G}(S)}{|S|} = \frac{3}{2},$$

$$\mu_3(N, v, G) = \sum_{\{S \text{ conexa: } 3 \in S\}} \frac{\Delta_{v^G}(S)}{|S|} = \frac{1}{2},$$

$$\mu_4(N, v, G) = \sum_{\{S \text{ conexa: } 4 \in S\}} \frac{\Delta_{v^G}(S)}{|S|} = \frac{1}{2},$$

$$\mu_5(N, v, G) = \sum_{\{S \text{ conexa: } 5 \in S\}} \frac{\Delta_{v^G}(S)}{|S|} = \frac{1}{2}.$$

Nótese que
$$\sum_{i \in N} \mu_i(N, v, G) = 1 + \frac{3}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} = 4 = v(N).$$

El resultado obtenido en el Teorema 3.27 establece un nuevo algoritmo al que se denominará *dividendo2* que permite determinar los dividendos de Harsanyi del juego (N, v) . La descripción del algoritmo es la siguiente.

Algoritmo *dividendo2*

$\Delta_{vG}(\emptyset) \leftarrow 0$

[para i de 1 hasta n

[para j de 1 hasta $S(i)$

$$\Delta_{vG}(S_i^j) \leftarrow \sum_{\{T: S_i^j \setminus ex(S_i^j) \subseteq T \subseteq S_i^j\}} (-1)^{|S_i^j| - |T|} v(T)$$

[fin_para

[fin_para

donde S_i^j es la j -ésima coalición de tamaño i y $S(i)$ es el número de coaliciones de tamaño i .

La complejidad temporal de este algoritmo se analiza en la proposición siguiente. En este análisis se considera un almacenamiento previo, en una tabla, de los extremales de cada coalición conexa.

Proposición 3.30 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\mathcal{O}(2^D |\mathcal{F}|)$, donde \mathcal{F} es el conjunto de coaliciones conexas en el árbol y $D = \max(\{|ex(S)| : S \in \mathcal{F}\})$.*

Demostración: De la descripción del algoritmo *dividendo2* se deduce que su tiempo de ejecución es

$$\begin{aligned}
 t(\text{dividendo2}) &= 1 + t(\text{bucle1}) \\
 &= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{asignación}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{suma}) \\
 &= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} 2 \cdot 2^{|ex(S_i^j)|} \\
 &\leq 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} 2 \cdot 2^D \\
 &= 1 + 2^{D+1} \sum_{i=1}^n \sum_{j=1}^{S(i)} 1 \\
 &< 1 + 2^{D+1} |\mathcal{F}|.
 \end{aligned}$$

Por tanto, $t(\text{dividendo2}) = f(n) \in \mathcal{O}(2^{D+1} |\mathcal{F}|)$.

□

Si se considera que los dividendos se calculan mediante un proceso ascendente que exige almacenar los dividendos de cada una de las coaliciones, se obtiene, de forma inmediata, el siguiente resultado.

Proposición 3.31 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un espacio $\Omega(|\mathcal{F}|)$.*

Además, la Proposición 3.30 da como consecuencia el resultado siguiente.

Proposición 3.32 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson de todos los jugadores, utilizando los dividendos de Harsanyi, requiere un tiempo $\mathcal{O}(\max(n|\mathcal{F}|, 2^D|\mathcal{F}|)$ siendo $D = \max(\{|ex(S)| : S \in \mathcal{F}\})$.*

Demostración: En primer lugar, con el algoritmo *dividendo2* se calculan todos los dividendos de Harsanyi de las coaliciones conexas, lo cual requiere un tiempo $\mathcal{O}(2^D|\mathcal{F}|)$ con $D = \max(\{|ex(S)| : S \in \mathcal{F}\})$. A continuación, para calcular el valor de Myerson del jugador i ,

$$\mu_i(N, v, G) = \sum_{\{S \text{ conexas: } i \in S\}} \frac{\Delta_{v^G}(S)}{|S|},$$

hay que considerar que cada jugador pertenece a $S(i)$ coaliciones, siendo $S(i) < |\mathcal{F}|$. El tiempo requerido para evaluar la suma en la expresión del valor de Myerson para un jugador es $\mathcal{O}(|\mathcal{F}|)$. Por tanto, el cálculo del valor de Myerson para los n jugadores requiere un tiempo $\mathcal{O}(\max(n|\mathcal{F}|, 2^D|\mathcal{F}|)$. \square

Nótese que, en los resultados anteriores, se ha supuesto realizado el cálculo del conjunto de puntos extremales de cada coalición conexas y su almacenamiento. Dado que esto tiene una repercusión notable en la complejidad del algoritmo, es por lo que, en la siguiente proposición, se estudia, para cualquier $S \in \mathcal{F}$, la complejidad temporal del cálculo de $ex(S)$ y $[S^-, S]$.

Proposición 3.33 *Sean $G = (N, E)$ un árbol y $S \in \mathcal{F}$, siendo \mathcal{F} el conjunto de coaliciones conexas de G .*

- a) *El cálculo de $ex(S)$ requiere un tiempo $\mathcal{O}(n|\mathcal{F}|)$.*
- b) *El cálculo de $[S^-, S]$ requiere un tiempo $\mathcal{O}(|\mathcal{F}|2^D)$, siendo $D = \max(\{|ex(S)| : S \in \mathcal{F}\})$.*

Demostración: a) Dada una coalición conexa S , para obtener el conjunto de todos sus extremales, hay que verificar si $S \setminus i \in \mathcal{F}$ para cada $i \in S$. Por tanto, hay que hacer $|S|$ comprobaciones. Como $|S| \leq n$ y hay $|\mathcal{F}|$ coaliciones conexas, el resultado es inmediato.

b) Al ser $[S^-, S] = \{T : T = (S \setminus ex(S)) \cup R, \quad R \in 2^{ex(S)}\}$, el cálculo de $[S^-, S]$ viene determinado por el tiempo de cálculo del conjunto $2^{ex(S)}$, el cual es $\mathcal{O}(2^{|ex(S)|})$. De aquí, el resultado se sigue de considerar que hay $|\mathcal{F}|$ coaliciones conexas y de ser $D = \max(\{|ex(S)| : S \in \mathcal{F}\})$. □

Como consecuencia de este resultado, si se analiza el cómputo global de los conjuntos $ex(S)$, $[S^-, S]$ y el dividendo correspondiente a cada coalición conexa $S \in \mathcal{F}$, se obtiene un resultado análogo al de la Proposición 3.32 para el cálculo del valor de Myerson de todos los jugadores mediante la utilización de los dividendos de Harsanyi. Obviamente las constantes multiplicativas serían mayores.

A continuación, se analiza la complejidad en dos casos particulares de árboles, que ya han sido estudiados en la sección anterior, y derivados de considerar un grafo camino y un grafo estrella. La intencionalidad de estudiar, de nuevo, estas dos situaciones de comunicación es poner de manifiesto la importancia de utilizar las fórmulas propuestas, en esta sección, en juegos restringidos por árboles ya que es posible, en algunos casos, obtener una reducción en los tiempos de cálculo, tanto en los dividendos del juego restringido como en el del valor de Myerson.

Previo al estudio de la complejidad temporal del cálculo de los dividendos en el grafo camino, se indica el número de jugadores extremales de cada coalición conexa, omitiéndose, por evidente, la demostración.

Lema 3.34 *Sea G un grafo camino. El número de jugadores extremales de una coalición conexa S viene dado por $|ex(S)| = 2$, si $|S| \geq 2$, y $|ex(S)| = 1$ si $|S| = 1$.*

Proposición 3.35 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\Theta(n^2)$.*

Demostración: Para el cálculo de los dividendos se utiliza el algoritmo *dividendo2*. Como ahora se consideran únicamente las coaliciones factibles, el número de coaliciones de tamaño i viene dado por $S(i) = n - i + 1$, y el número total de coaliciones factibles contenidas estrictamente en una coalición factible T , con $|T| = i$, es $C(i) = \frac{i^2 + i}{2}$. Por tanto, el tiempo de ejecución del algoritmo es

$$\begin{aligned}
t(\text{dividendo2}) &= 1 + t(\text{bucle1}) \\
&= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{asignación}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{suma}) \\
&= 1 + \sum_{j=1}^{S(1)} t(\text{suma}) + \sum_{i=2}^n \sum_{j=1}^{S(i)} t(\text{suma}) \\
&= 1 + \sum_{j=1}^{S(1)} 2 \cdot 2^1 + \sum_{i=2}^n \sum_{j=1}^{S(i)} 2 \cdot 2^2 \\
&= 1 + 2^2 \sum_{j=1}^n 1 + 2^3 \sum_{i=2}^n \sum_{j=1}^{S(i)} 1 \\
&= 1 + 2^2 n + 2^3 \sum_{i=2}^n S(i) \\
&= 1 + 4n + 2^3 \sum_{i=2}^n (n - i + 1) \\
&= 1 + 4n + 8 \frac{(n-1) + 1}{2} (n-1)
\end{aligned}$$

$$\begin{aligned}
&= 1 + 4n + 4n(n - 1) \\
&= 1 + 4n^2.
\end{aligned}$$

Por tanto, $t(\text{dividendo2}) = f(n) \in \Theta(n^2)$.

□

Proposición 3.36 *Sea la terna (N, v, G) , donde G es un grafo camino y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson de todos los jugadores requiere un tiempo $\mathcal{O}(n^3)$.*

Demostración: En primer lugar, con el algoritmo *dividendo2* se calculan todos los dividendos de Harsanyi. El cálculo requiere un tiempo $\Theta(n^2)$. A continuación, para calcular el valor de Myerson del jugador k ,

$$\mu_k(N, v, G) = \sum_{\{S \text{ conexa: } k \in S\}} \frac{\Delta_{v^G}(S)}{|S|},$$

hay que considerar que este jugador pertenece a $F_k = k(n - k + 1)$ coaliciones. Como

$$n \leq F_k \leq \frac{1}{4}(n + 1)^2,$$

se tiene que el tiempo requerido para evaluar la suma, para cada jugador, es $\mathcal{O}(n^2)$. Por tanto, el cálculo del valor de Myerson para los n jugadores requiere un tiempo $\mathcal{O}(n^3)$.

□

Obsérvese que este resultado se podría haber obtenido como una consecuencia de la Proposición 3.32, considerando que $D = 2$ y $|\mathcal{F}| = \frac{n^2 + n + 2}{2}$, pues $\mathcal{O}(\max(n|\mathcal{F}|, 2^D|\mathcal{F}|)) = \mathcal{O}(n^3)$.

A continuación, se estudia la complejidad del cálculo del valor de Myerson, a partir de los dividendos de Harsanyi, cuando la situación de comunicación viene determinada por un grafo estrella. Se seguirá un esquema de trabajo similar al estudio particular anterior y se podrá observar que, ahora, no van a producirse ventajas computacionales en cuanto al tiempo.

Lema 3.37 Sea G un grafo estrella. El número de jugadores extremos de una coalición conexa S viene dado por $|ex(S)| = |S| - 1$, si $|S| \geq 3$, y por $|ex(S)| = |S|$, si $|S| \leq 2$.

Proposición 3.38 Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\Theta(3^n)$.

Demostración: Para el cálculo de los dividendos se utiliza el algoritmo *dividendo2*. Como ahora se consideran únicamente las coaliciones factibles, el número de coaliciones de tamaño i viene dado por $S(i) = \binom{n-1}{i-1}$, y el número total de coaliciones factibles contenidas estrictamente en una coalición factible T , con $|T| = i$, es $C(i) = i + 2^{i-1} - 1$. Por tanto, el tiempo de ejecución del algoritmo es

$$\begin{aligned}
t(\text{dividendo2}) &= 1 + t(\text{bucle1}) \\
&= 1 + \sum_{i=1}^n t(\text{bucle2}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} t(\text{asignación}) \\
&= 1 + \sum_{i=1}^n \sum_{j=1}^{S(i)} (1 + t(\text{suma})) \\
&= 1 + 2n + 2 \binom{n-1}{1} 2^2 + \sum_{i=3}^n \sum_{j=1}^{S(i)} 2 \cdot 2^{i-1} \\
&= 1 + 2n + 8(n-1) + \sum_{i=3}^n 2^i \sum_{j=1}^{S(i)} 1 \\
&= 1 + 2n + 8(n-1) + \sum_{i=3}^n 2^i S(i) \\
&= 10n - 7 + \sum_{i=3}^n 2^i \binom{n-1}{i-1}
\end{aligned}$$

$$= 6n - 5 + 23^{n-1}.$$

Por tanto, $t(\text{dividendo2}) = f(n) \in \Theta(3^n)$.

□

Proposición 3.39 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson de todos los jugadores requiere un tiempo $\Theta(3^n)$.*

Demostración: En primer lugar, con el algoritmo *dividendo2*, se calculan todos los dividendos de Harsanyi. A continuación, para calcular el valor de Myerson del jugador k hay que considerar que este jugador pertenece a F_k coaliciones. Como $F_k \leq 2^{n-1}$, se tiene que el tiempo requerido para evaluar la suma en la expresión del valor de Myerson es $\mathcal{O}(2^n)$ para cada jugador. Por tanto, la complejidad del cálculo del valor de Myerson viene determinada por la complejidad del cálculo de los dividendos. Es decir, se requiere un tiempo $\Theta(3^n)$.

□

3.3 Generalización del modelo de Myerson. Estructuras \cup -estables

Aunque el modelo de Myerson supone un primer paso en el estudio de la cooperación parcial, éste no cubre todas las situaciones posibles que pueden plantearse. Por ejemplo, es inmediato observar que si no fuera posible formar coaliciones factibles de menos de tres jugadores, entonces sería imposible modelar las relaciones entre los jugadores mediante un grafo de cooperación. Por ello, en 1980, en su trabajo *Conference structures and fair allocation rules*, Myerson plantea la generalización de su modelo de cooperación parcial con la utilización de *hipergrafos de comunicación* basándose en la existencia de un conjunto de coaliciones factibles que no tiene porqué derivarse de las

coaliciones conexas que se forman en un grafo de cooperación entre los jugadores. Este estudio es continuado por van den Nouweland, Borm y Tijs [19].

Siguiendo la línea de investigación de Myerson, López [39], Bilbao [10] [11], y Algaba [2] desarrollan un modelo de cooperación parcial basado en los denominados *sistemas \cup -estables*, los cuales generalizan las situaciones de comunicación. A continuación se exponen resultados relativos a este modelo y su relación con el de Myerson.

Definición 3.40 *Un sistema \cup -estable o estable para la unión es un par (N, \mathcal{F}) , con $\mathcal{F} \subseteq 2^N$, que verifica*

$$A, B \in \mathcal{F}, A \cap B \neq \emptyset \implies A \cup B \in \mathcal{F}.$$

Los sistemas \cup -estables tienen una interpretación lógica dentro de la cooperación parcial ya que, si dos coaliciones factibles tienen elementos comunes, éstos pueden ejercer de intermediarios entre ambas para formar una coalición factible más amplia, constituida por el grupo total. Como puede observarse, en uno de los siguientes ejemplos, un caso particular de familias de coaliciones factibles \cup -estables lo constituyen las situaciones de comunicación.

Ejemplo 3.41 *Sea $N = \{1, 2, \dots, n\}$, para algún número natural n y considérese la colección \mathcal{L}_n , constituida por todas las coaliciones conexas del grafo camino. En este modelo, que corresponde a una situación de votación en un orden político unidimensional estudiada por Edelman [31], y que tiene un antecedente en el policy order de Axelrod [7], el par (N, \mathcal{L}_n) es un sistema \cup -estable.*

Ejemplo 3.42 Sea la terna (N, v, G) una situación de comunicación, donde $G = (N, E)$. El par (N, \mathcal{F}) , con

$$\mathcal{F} = \{S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G\},$$

es un sistema \cup -estable.

Es importante resaltar que, dado un sistema \cup -estable, éste no siempre puede ser modelado mediante un grafo de comunicación. En efecto, considérese el par (N, \mathcal{F}) , donde $N = \{1, 2, 3, 4\}$ y la familia

$$\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 2, 3\}, \{2, 3, 4\}, N\}.$$

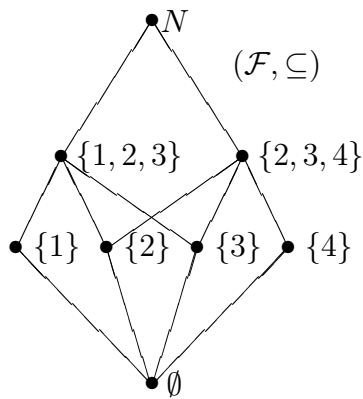


FIGURA 3.3

Puede comprobarse que (N, \mathcal{F}) constituye un sistema estable para la unión que no coincide con la familia de subgrafos conexos de ningún grafo. Esto es debido a que cualquier grafo (N, E) se define mediante parejas $\{i, j\}$, lo cual implica que deben existir forzosamente coaliciones factibles formadas por dos elementos y esta propiedad no se verifica para la familia de coaliciones \mathcal{F} considerada.

Ejemplo 3.43 Sea $N = \{1, 2, 3, 4\}$ y considérense los sistemas de coaliciones (N, \mathcal{F}_1) y (N, \mathcal{F}_2) dados por

$$\mathcal{F}_1 = \{\{1\}, \{4\}, \{1, 2\}, \{2, 3\}, \{2, 4\}, \{1, 2, 3\}, \{1, 2, 4\}, \{2, 3, 4\}, N\},$$

$$\mathcal{F}_2 = \{\{1\}, \{2\}, \{3\}, \{4\}, \{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, N\}.$$

A través de la representación de $(\mathcal{F}_1, \subseteq)$ y $(\mathcal{F}_2, \subseteq)$ puede examinarse fácilmente si los sistemas son \cup -estables. Así, (N, \mathcal{F}_1) constituye un sistema \cup -estable ya que, como puede comprobarse (ver diagramas de la Figura 3.4), siempre que dos coaliciones factibles tienen intersección no vacía, su unión es una coalición factible. El par (N, \mathcal{F}_2) no es un sistema \cup -estable puesto que, por ejemplo,

$$\{1, 2\}, \{1, 3\} \in \mathcal{F}_2, \text{ verifican } \{1, 2\} \cap \{1, 3\} \neq \emptyset,$$

y, sin embargo,

$$\{1, 2\} \cup \{1, 3\} = \{1, 2, 3\} \notin \mathcal{F}_2.$$

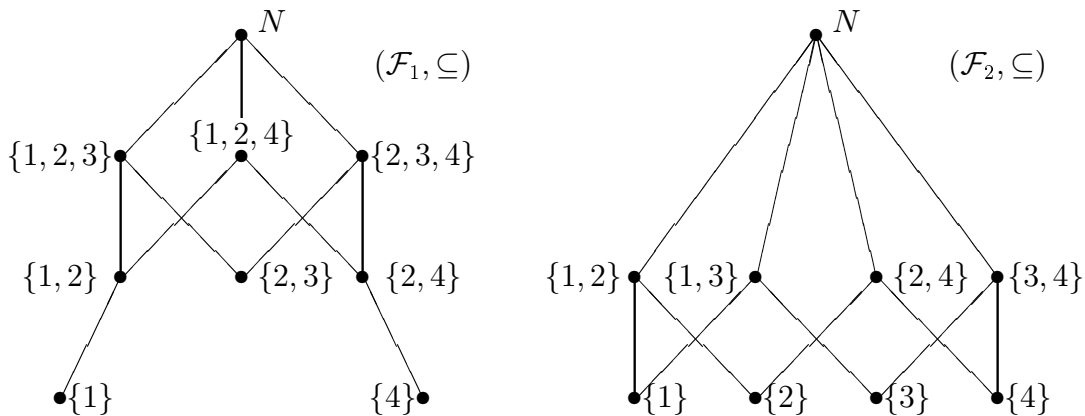


FIGURA 3.4

Definición 3.44 Sean $\mathcal{F} \subseteq 2^N$ y $S \subseteq N$. Se dice que $T \subseteq S$ es una \mathcal{F} -componente de S si $T \in \mathcal{F}$ y no existe $T' \in \mathcal{F}$ tal que $T \subset T' \subseteq S$.

Es decir, las \mathcal{F} -componentes de S son las coaliciones factibles maximales pertenecientes a \mathcal{F} y contenidas en S .

En adelante, para toda $S \subseteq N$, se denotará por $C_{\mathcal{F}}(S)$ al conjunto de coaliciones factibles maximales de S en (N, \mathcal{F}) ; es decir,

$$C_{\mathcal{F}}(S) = \{T : T \text{ es } \mathcal{F}\text{-componente de } S\} \subseteq 2^S.$$

Obsérvese que el conjunto $C_{\mathcal{F}}(S)$ puede ser vacío. Esto significa que no existe ninguna coalición factible contenida en S y, este hecho, tendrá que ser tenido en cuenta a la hora de introducir el juego restringido por la cooperación parcial definida por la familia \mathcal{F} .

En el Ejemplo 3.42 se ha puesto de manifiesto que, dada una situación de comunicación (N, v, G) , el par (N, \mathcal{F}) , donde

$$\mathcal{F} = \{S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G\},$$

es un sistema de coaliciones factibles \cup -estable. En este caso, las \mathcal{F} -componentes de cualquier coalición $S \subseteq N$ son las componentes conexas del subgrafo $(S, E(S))$ y constituyen una partición de S .

En general, se demuestra que, para cualquier coalición $S \subseteq N$ tal que $C_{\mathcal{F}}(S) \neq \emptyset$, las \mathcal{F} -componentes de S son siempre disjuntas entre sí y constituyen una partición de un subconjunto de S . Ahora bien, si todas las coaliciones unitarias fueran factibles, entonces el conjunto $C_{\mathcal{F}}(S)$ forma una partición de S . En este caso, el sistema de coaliciones factibles \cup -estable (N, \mathcal{F}) se denomina *sistema de partición*.

Una *estructura de cooperación estable para la unión* es una terna (N, v, \mathcal{F}) donde (N, v) es un juego cooperativo de utilidad transferible y el par (N, \mathcal{F}) es un sistema de coaliciones estable para la unión.

Definición 3.45 Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión. Se denomina \mathcal{F} -juego restringido al par $(N, v^{\mathcal{F}})$, con

$$v^{\mathcal{F}} : 2^N \longrightarrow \mathbb{R}, \quad v^{\mathcal{F}}(S) = \begin{cases} \sum_{T \in C_{\mathcal{F}}(S)} v(T), & \text{si } C_{\mathcal{F}}(S) \neq \emptyset \\ 0, & \text{en otro caso.} \end{cases}$$

Teniendo en cuenta el Ejemplo 3.42, la Definición 3.44 y las observaciones posteriores, puede afirmarse que la definición anterior constituye una generalización del concepto de juego restringido por un grafo de comunicación dado por Myerson [42] (véase Algaba y López [2][39]).

Teorema 3.46 *Sea la terna (N, v, \mathcal{F}) una estructura de cooperación estable para la unión. Si (N, v) es un juego simple, propio y cero-normalizado, se verifica*

- a) $(N, v^{\mathcal{F}})$ es simple y propio.
- b) $v^{\mathcal{F}}(S) = \max\{v(T) : T \in C_{\mathcal{F}}(S)\}$, para cada $S \subseteq N$.

Para estructuras de cooperación \cup -estables, se define el *valor de Myerson generalizado*, el cual generaliza el valor de Myerson definido en situaciones de comunicación.

Definición 3.47 *Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión. El valor de Myerson generalizado asociado a (N, v, \mathcal{F}) , denotado por $\mu(N, v, \mathcal{F})$, está definido por*

$$\mu(N, v, \mathcal{F}) = \Phi(v^{\mathcal{F}}) \in \mathbb{R}^n.$$

Ejemplo 3.48 *Considérese el conjunto de jugadores $N = \{1, 2, 3, 4\}$ y el sistema estable para la unión dado por $\mathcal{F} = \{\{1\}, \{1, 2, 3\}, \{2, 3, 4\}, N\}$.*

Sea $v : 2^N \longrightarrow \mathbb{R}$ definido por $v(S) = |S| - 1$, para toda coalición no vacía. En este caso, el juego restringido está definido por

$$v^{\mathcal{F}}(S) = \begin{cases} |S| - 1, & \text{si } S \in \mathcal{F} \\ 0, & \text{en otro caso.} \end{cases}$$

El valor de Myerson generalizado, $\mu(N, v, \mathcal{F})$, viene dado por

$$\begin{aligned} \mu_1(N, v, \mathcal{F}) &= \sum_{\{S \subseteq N : 1 \in S\}} \gamma(S) [v^{\mathcal{F}}(S) - v^{\mathcal{F}}(S \setminus \{1\})] = \frac{5}{12}, \\ \mu_2(N, v, \mathcal{F}) &= \sum_{\{S \subseteq N : 2 \in S\}} \gamma(S) [v^{\mathcal{F}}(S) - v^{\mathcal{F}}(S \setminus \{2\})] = \frac{13}{12}, \\ \mu_3(N, v, \mathcal{F}) &= \sum_{\{S \subseteq N : 3 \in S\}} \gamma(S) [v^{\mathcal{F}}(S) - v^{\mathcal{F}}(S \setminus \{3\})] = \frac{13}{12}, \\ \mu_4(N, v, \mathcal{F}) &= \sum_{\{S \subseteq N : 4 \in S\}} \gamma(S) [v^{\mathcal{F}}(S) - v^{\mathcal{F}}(S \setminus \{4\})] = \frac{5}{12}, \end{aligned}$$

siendo $\gamma(S) = \frac{(s-1)!(n-s)!}{n!}$. Así, $\mu(N, v, \mathcal{F}) = \left(\frac{5}{12}, \frac{13}{12}, \frac{13}{12}, \frac{5}{12} \right)$.

Con el objetivo de analizar las características del valor de Myerson generalizado, se introducen progresivamente, y siguiendo la terminología de Myerson, el concepto de *regla de asignación* y algunas propiedades de las mismas.

Para ello, fijado un conjunto finito $N = \{1, 2, \dots, n\}$, se denotará por SE^N al conjunto de todas las estructuras de cooperación estables para la unión que tienen a N como conjunto de jugadores. Es decir,

$$SE^N = \{(N, v, \mathcal{F}) : (N, \mathcal{F}) \text{ es } \cup\text{-estable y } (N, v) \text{ un juego cooperativo}\}.$$

Definición 3.49 Una regla de asignación es una función $\gamma : SE^N \longrightarrow \mathbb{R}^n$, $(N, v, \mathcal{F}) \longmapsto (\gamma_1(N, v, \mathcal{F}), \dots, \gamma_n(N, v, \mathcal{F}))$, que verifica

1. $\sum_{k \in M} \gamma_k(N, v, \mathcal{F}) = v(M), \forall (N, v, \mathcal{F}) \in SE^N, \forall M \in C_{\mathcal{F}}(N).$
2. $\gamma_i(N, v, \mathcal{F}) = 0, \forall i \notin \bigcup_{M \in C_{\mathcal{F}}(N)} M.$

Es decir, una regla de asignación es una función que, a cada estructura de cooperación estable para la unión definida sobre N , le asigna un vector n -dimensional, que es eficiente para las \mathcal{F} -componentes de la coalición N y que asocia un valor nulo a aquellos jugadores que no pertenecen a ninguna coalición factible maximal.

Myerson [42] demostró, en el conjunto de situaciones de comunicación, que el valor de Myerson es la única regla de asignación justa. Con el propósito de dar un resultado similar en estructuras de cooperación estables para la unión se tendrá que definir previamente lo que se entiende por regla de asignación justa en estructuras de cooperación \cup -estables. Para ello es necesario introducir el concepto de base de un sistema estable para la unión.

Considérese un sistema \cup -estable (N, \mathcal{F}) y sea $\mathcal{G} \subseteq \mathcal{F}$. A partir de \mathcal{G} se procede a realizar la siguiente construcción

$$\begin{aligned}
\mathcal{G}^{(0)} &= \mathcal{G} \\
\mathcal{G}^{(1)} &= \{ S \cup T : S, T \in \mathcal{G}^{(0)}, S \cap T \neq \emptyset \} \\
&\vdots \\
\mathcal{G}^{(k)} &= \{ S \cup T : S, T \in \mathcal{G}^{(k-1)}, S \cap T \neq \emptyset \} \\
&\vdots
\end{aligned}$$

Obsérvese que

$$\mathcal{G}^{(0)} \subseteq \mathcal{G}^{(1)} \subseteq \dots \subseteq \mathcal{G}^{(k)} \subseteq \dots \subseteq \mathcal{F},$$

puesto que $\mathcal{G} \subseteq \mathcal{F}$ y (N, \mathcal{F}) es un sistema \cup -estable. Además, el proceso es siempre finito ya que, al ser \mathcal{F} finito, se obtendrían, a lo sumo, todas las coaliciones factibles de \mathcal{F} . Esta observación permite introducir el siguiente concepto.

Definición 3.50 Sean (N, \mathcal{F}) un sistema \cup -estable y $\mathcal{G} \subseteq \mathcal{F}$. Se denomina sistema generado por \mathcal{G} , al conjunto $SG(\mathcal{G}) = \mathcal{G}^{(k)}$, donde k es el menor entero tal que $\mathcal{G}^{(k+1)} = \mathcal{G}^{(k)}$.

Obviamente, $SG(\emptyset) = \emptyset$, $SG(\{S\}) = \{S\}$ para cualquier $S \in \mathcal{F}$, y $SG(\mathcal{F}) = \mathcal{F}$. En el siguiente ejemplo se ilustra este concepto.

Ejemplo 3.51 Sea $N = \{1, 2, 3, 4, 5\}$ y considérese el sistema \cup -estable (N, \mathcal{F}) dado por

$$\mathcal{F} = \{\{1\}, \{3\}, \{1, 2\}, \{2, 3\}, \{2, 4\}, \{3, 5\}, \{1, 2, 3\}, \{1, 2, 4\}, \\ \{2, 3, 4\}, \{2, 3, 5\}, \{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{2, 3, 4, 5\}, N\}.$$

Si $\mathcal{G} = \{\{1, 2\}, \{2, 3\}, \{3, 5\}\} \subseteq \mathcal{F}$, entonces

$$\begin{aligned} \mathcal{G}^{(0)} &= \mathcal{G} = \{\{1, 2\}, \{2, 3\}, \{3, 5\}\}, \\ \mathcal{G}^{(1)} &= \{\{1, 2\}, \{2, 3\}, \{3, 5\}, \{1, 2, 3\}, \{2, 3, 5\}\}, \\ \mathcal{G}^{(2)} &= \{\{1, 2\}, \{2, 3\}, \{3, 5\}, \{1, 2, 3\}, \{2, 3, 5\}, \{1, 2, 3, 5\}\}, \\ \mathcal{G}^{(3)} &= \mathcal{G}^{(2)}. \end{aligned}$$

Luego,

$$SG(\mathcal{G}) = \mathcal{G}^{(2)} = \{\{1, 2\}, \{2, 3\}, \{3, 5\}, \{1, 2, 3\}, \{2, 3, 5\}, \{1, 2, 3, 5\}\} \subseteq \mathcal{F}$$

y, en este caso, $k = 2$.

Sería deseable encontrar, para cada sistema de coaliciones estable para la unión, un subconjunto minimal que, por el proceso anteriormente descrito, genere toda la familia inicial. Esta idea conduce a los siguientes conceptos.

Sean (N, \mathcal{F}) un sistema \cup -estable y $\mathcal{G} \subseteq \mathcal{F}$. Si (N, \mathcal{G}) es \cup -estable, existirán, posiblemente, coaliciones factibles que puedan expresarse como unión de dos coaliciones factibles con intersección no vacía. Esto da lugar a considerar el conjunto

$$D(\mathcal{G}) = \{G \in \mathcal{G} : G = A \cup B, A \neq G, B \neq G, A, B \in \mathcal{G}, A \cap B \neq \emptyset\}.$$

Es decir, $D(\mathcal{G})$ está formado por aquellas coaliciones factibles que se pueden expresar como unión de otras dos coaliciones factibles distintas y no disjuntas.

Definición 3.52 Sea (N, \mathcal{F}) un sistema \cup -estable. Al conjunto

$$B(\mathcal{F}) = \mathcal{F} \setminus D(\mathcal{F}),$$

se le denomina base de \mathcal{F} . Sus elementos se llaman soportes de \mathcal{F} .

El siguiente ejemplo ilustra el concepto de base.

Ejemplo 3.53 Sea $N = \{1, 2, 3, 4\}$ y considérese la familia de coaliciones factibles

$$\mathcal{F} = \{\{1\}, \{3\}, \{1, 2\}, \{1, 3\}, \{3, 4\}, \{1, 2, 3\}, \{1, 3, 4\}, \{2, 3, 4\}, \{1, 2, 3, 4\}\}.$$

El sistema (N, \mathcal{F}) es estable para la unión. Para calcular la base de \mathcal{F} se seleccionan, del conjunto de coaliciones factibles, aquéllas que pueden expresarse como unión de dos coaliciones factibles distintas con intersección no vacía; esto es,

$$\begin{aligned} \{1, 2, 3, 4\} &= \{1, 3, 4\} \cup \{2, 3, 4\}, \\ \{1, 3, 4\} &= \{1, 3\} \cup \{3, 4\}, \\ \{1, 2, 3\} &= \{1, 2\} \cup \{1, 3\}, \end{aligned}$$

y así,

$$D(\mathcal{F}) = \{\{1, 2, 3\}, \{1, 3, 4\}, \{1, 2, 3, 4\}\}.$$

Finalmente, se eliminan de \mathcal{F} dichas coaliciones, quedando

$$B(\mathcal{F}) = \{\{1\}, \{3\}, \{1, 2\}, \{1, 3\}, \{3, 4\}, \{2, 3, 4\}\}.$$

Es evidente que, por construcción, el conjunto $B(\mathcal{F})$ es único y no vacío, siempre que $\mathcal{F} \neq \emptyset$. Además, se verifica que si (N, \mathcal{F}) es un sistema \cup -estable, entonces $B(\mathcal{F})$ es el subconjunto minimal de \mathcal{F} tal que $SG(B(\mathcal{F})) = \mathcal{F}$ (véase [2]).

Un estudio exhaustivo de las propiedades de las bases de un sistema \cup -estable así como las relaciones entre las \mathcal{F} -componentes, las coaliciones factibles y los soportes puede verse en [3][4]. De entre ellas hay una de especial significación ya que relaciona estrechamente las estructuras de cooperación \cup -estables con el modelo de estructuras de conferencias de Myerson. Esta relación constituye un resultado fundamental en la caracterización axiomática del valor de Myerson generalizado.

Definición 3.54 *Una regla de asignación, $\gamma : SE^N \rightarrow \mathbb{R}^n$, es justa si, para toda $(N, v, \mathcal{F}) \in SE^N$ y cualesquiera $B \in B(\mathcal{F})$ y $j \in B$,*

$$\gamma_j(N, v, \mathcal{F}) - \gamma_j(N, v, \mathcal{F}') = c,$$

donde $\mathcal{F}' = SG(B(\mathcal{F}) \setminus \{B\})$ y $c \in \mathbb{R}$.

Así, si una regla de asignación es justa, todos los jugadores que pertenecen a un mismo soporte B pierden o ganan la misma cantidad si este soporte B es eliminado de la estructura de cooperación.

Myerson [42] probó que si el juego (N, v) es superaditivo, entonces el valor de Myerson es, además, una regla de asignación estable en el conjunto de situaciones de comunicación. Este resultado se puede generalizar a estructuras de cooperación estables para la unión.

Definición 3.55 *Una regla de asignación, $\gamma : SE^N \rightarrow \mathbb{R}^n$, es estable si, para toda $(N, v, \mathcal{F}) \in SE^N$ y cualesquiera $B \in B(\mathcal{F})$ y $j \in B$,*

$$\gamma_j(N, v, \mathcal{F}) \geq \gamma_j(N, v, \mathcal{F}'),$$

siendo $\mathcal{F}' = SG(B(\mathcal{F}) \setminus \{B\})$.

Esta condición asegura que todos los jugadores de cualquier soporte siempre se benefician de llegar a un acuerdo y cooperar, si se consideran juegos de beneficios.

Teorema 3.56 *Sea $(N, v, \mathcal{F}) \in SE^N$ tal que (N, v) es superaditivo y cero-normalizado. Entonces, el valor de Myerson generalizado, $\mu(N, v, \mathcal{F})$, es una regla de asignación justa y estable.*

El hecho de que el valor de Myerson generalizado sea una regla de asignación implica, de forma inmediata, que constituye siempre una preimputación para el juego restringido $(N, v^{\mathcal{F}})$. En efecto, para cualquier estructura de cooperación estable para la unión,

$$\sum_{i \in N} \mu_i(N, v, \mathcal{F}) = \sum_{M \in C_{\mathcal{F}}(N)} \left[\sum_{i \in M} \mu_i(N, v, \mathcal{F}) \right] = \sum_{M \in C_{\mathcal{F}}(N)} v(M) = v^{\mathcal{F}}(N).$$

Al igual que en situaciones de comunicación existen otras caracterizaciones axiomáticas para el valor de Myerson debidas a Borm, van den Nouweland y Tijs [20], en sistemas \cup -estables hay también caracterizaciones análogas para el valor de Myerson generalizado [2].

Para el cálculo del valor de Myerson generalizado en un juego restringido por una estructura de cooperación estable para la unión, se va a seguir una línea similar a la presentada para el cálculo del valor de Myerson.

La característica de linealidad del valor de Shapley hace que el valor de Myerson generalizado pueda ser expresado en términos de los dividendos de Harsanyi. De forma análoga a las situaciones de comunicación [39][12], el valor de Myerson generalizado se expresa en términos de los dividendos del juego restringido. Así, se tiene que, para cualquier $i \in N$,

$$\mu_i(N, v, \mathcal{F}) = \sum_{\{S \in \mathcal{F}: i \in S\}} \frac{\Delta_{v^{\mathcal{F}}}(S)}{|S|},$$

ya que dada una estructura de cooperación estable para la unión (N, v, \mathcal{F}) , el conjunto de los juegos de unanimidad $\{u_T : T \in \mathcal{F}, T \neq \emptyset\}$ constituye una base del espacio $L_{\mathcal{F}}(\Gamma^N)$, donde $L_{\mathcal{F}} : \Gamma^N \rightarrow \Gamma^N$, $L_{\mathcal{F}}(v) = v^{\mathcal{F}}$.

Como consecuencia de este resultado, el juego restringido puede expresarse como una combinación lineal de los juegos de unanimidad correspondientes a las coaliciones factibles.

$$v^{\mathcal{F}} = \sum_{\{T \in \mathcal{F}: T \neq \emptyset\}} \Delta_{v^{\mathcal{F}}}(T) u_T, \quad \text{con } \Delta_{v^{\mathcal{F}}}(\emptyset) = 0,$$

verificándose, para toda coalición factible S ,

$$v(S) = v^{\mathcal{F}}(S) = \sum_{\{T \in \mathcal{F}: T \subseteq S\}} \Delta_{v^{\mathcal{F}}}(T).$$

A partir de esta expresión se pueden calcular, para cualquier coalición S factible, todos los dividendos del juego restringido con la siguiente fórmula recursiva

$$\Delta_{v^{\mathcal{F}}}(S) = v(S) - \sum_{\{T \in \mathcal{F}: T \subset S\}} \Delta_{v^{\mathcal{F}}}(T).$$

Esta fórmula es similar a la dada anteriormente para el cálculo de los dividendos de un juego restringido por una situación de comunicación, cambiando las coaliciones conexas por las coaliciones factibles.

Ejemplo 3.57 Sea la terna (N, v, \mathcal{F}) , siendo $N = \{1, 2, 3, 4\}$, $v(S) = |S| - 1$ para toda coalición no vacía $S \subseteq N$ y $\mathcal{F} = \{\{1\}, \{1, 2, 3\}, \{2, 3, 4\}, N\}$.

Utilizando resultados anteriores, el valor de Myerson generalizado viene determinado, para todo $i \in N$, por

$$\mu_i(N, v, \mathcal{F}) = \sum_{\{S \in \mathcal{F}: i \in S\}} \frac{\Delta_{v^{\mathcal{F}}}(S)}{|S|},$$

$$\text{con } \Delta_{v^{\mathcal{F}}}(S) = v(S) - \sum_{\{T \in \mathcal{F}: T \subset S\}} \Delta_{v^{\mathcal{F}}}(T) \text{ para } S \in \mathcal{F}.$$

$$\Delta_{v^{\mathcal{F}}}(\{1\}) = v(\{1\}) = 0,$$

$$\Delta_{v^{\mathcal{F}}}(\{1, 2, 3\}) = v(\{1, 2, 3\}) - \Delta_{v^{\mathcal{F}}}(\{1\}) = 2 - 0 = 2,$$

$$\Delta_{v^{\mathcal{F}}}(\{2, 3, 4\}) = v(\{2, 3, 4\}) = 2,$$

$$\Delta_{v^{\mathcal{F}}}(N) = v(N) - (\Delta_{v^{\mathcal{F}}}(\{1\}) + \Delta_{v^{\mathcal{F}}}(\{1, 2, 3\}) + \Delta_{v^{\mathcal{F}}}(\{2, 3, 4\})) = -1.$$

De ahí,

$$\begin{aligned} \mu_1(N, v, \mathcal{F}) &= \sum_{\{S \in \mathcal{F}: 1 \in S\}} \frac{\Delta_{v^{\mathcal{F}}}(S)}{|S|} \\ &= \Delta_{v^{\mathcal{F}}}(\{1\}) + \frac{1}{3}\Delta_{v^{\mathcal{F}}}(\{1, 2, 3\}) + \frac{1}{4}\Delta_{v^{\mathcal{F}}}(N) \\ &= 0 + \frac{1}{3}2 + \frac{1}{4}(-1) \\ &= \frac{5}{12}. \end{aligned}$$

Análogamente,

$$\begin{aligned} \mu_2(N, v, \mathcal{F}) &= \mu_3(N, v, \mathcal{F}) \\ &= \frac{1}{3}\Delta_{v^{\mathcal{F}}}(\{1, 2, 3\}) + \frac{1}{3}\Delta_{v^{\mathcal{F}}}(\{2, 3, 4\}) + \frac{1}{4}\Delta_{v^{\mathcal{F}}}(N) \\ &= \frac{1}{3}2 + \frac{1}{3}2 + \frac{1}{4}(-1) \\ &= \frac{13}{12}. \\ \mu_4(N, v, \mathcal{F}) &= \frac{1}{3}\Delta_{v^{\mathcal{F}}}(\{2, 3, 4\}) + \frac{1}{4}\Delta_{v^{\mathcal{F}}}(N) \\ &= \frac{1}{3}2 + \frac{1}{4}(-1) \\ &= \frac{5}{12}. \end{aligned}$$

Nótese que $\sum_{i \in N} \mu_i(N, v, \mathcal{F}) = 3 = v^{\mathcal{F}}(N)$.

Utilizando el algoritmo *dividendo* para el cálculo de los dividendos de las coaliciones factibles, para las complejidades espacial y temporal se tienen resultados idénticos a los descritos anteriormente para el valor de Myerson. Por dicho motivo se omiten las pruebas de los mismos.

Proposición 3.58 *Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión. El cálculo de todos los dividendos del juego restringido requiere un espacio $\Omega(|\mathcal{F}|)$.*

Proposición 3.59 *Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión. El cálculo de todos los dividendos del juego restringido requiere un tiempo $\mathcal{O}(3^n)$.*

Si se tiene en cuenta la posibilidad de que la coalición N no sea una coalición factible, puede realizarse el cálculo del valor de Myerson generalizado considerando la restricción a las coaliciones factibles maximales de N a la que pertenezca cada uno de los jugadores.

Teorema 3.60 *Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión tal que (N, v) es un juego cero-normalizado. Si $M \in C_{\mathcal{F}}(N)$ es tal que $i \in M$, entonces*

$$\mu_i(N, v, \mathcal{F}) = \mu_i(M, v_M, \mathcal{F}_M),$$

donde $\mathcal{F}_M = \{F \in \mathcal{F} : F \subseteq M\}$.

Este resultado, dado por Algaba [2], supone una generalización del dado por Van den Nouweland [59] para situaciones de comunicación.

Teorema 3.61 *Sea (N, v, \mathcal{F}) una estructura de cooperación estable para la unión con $|C_{\mathcal{F}}(N)| = p$.*

- a) El cálculo del valor de Myerson generalizado para un jugador i , mediante los dividendos de Harsanyi, requiere un espacio $\Omega(|\mathcal{F}_M|)$ y un tiempo $\mathcal{O}(3^{|M|})$, siendo $M \in C_{\mathcal{F}}(N)$ tal que $i \in M$.
- b) El cálculo del valor de Myerson generalizado para todos los jugadores, mediante los dividendos de Harsanyi, requiere un espacio $\Omega(|\mathcal{F}|)$ y un tiempo $\mathcal{O}(p \cdot 3^{|M|})$, siendo $|M| = \max\{|M_j| : M_j \in C_{\mathcal{F}}(N)\}$.

Demostración: a) Si la coalición N no es una coalición factible, el cálculo del valor de Myerson generalizado, para el jugador i , se puede realizar considerando la restricción a la coalición factible maximal de N a la que pertenezca este jugador. Esto es, si $M \in C_{\mathcal{F}}(N)$ tal que $i \in M$, entonces (M, v_M, \mathcal{F}_M) , donde v_M es la restricción de v a M y $\mathcal{F}_M = \{F \in \mathcal{F} : F \subseteq M\}$, es una estructura de cooperación \cup -estable y $\mu_i(N, v, \mathcal{F}) = \mu_i(M, v_M, \mathcal{F}_M)$ por el teorema anterior. Por tanto, se requiere un espacio $\Omega(|\mathcal{F}_M|)$ y un tiempo $\mathcal{O}(3^{|M|})$ para el cálculo de los dividendos de todas las coaliciones $S \in \mathcal{F}_M$.

Como $\mu_i(N, v, \mathcal{F}) = \mu_i(M, v_M, \mathcal{F}_M) = \sum_{\{S \in \mathcal{F}_M : i \in S\}} \frac{\Delta_{v_{M_j}^{\mathcal{F}}}(S)}{|S|}$, y se necesita un tiempo $\mathcal{O}(|\mathcal{F}_M|)$ para evaluar esta suma, resulta que el cálculo del valor de Myerson de un jugador $i \in M$ requiere un tiempo $\mathcal{O}(3^{|M|})$.

b) Para realizar el cálculo del valor de Myerson generalizado de todos los jugadores, habría que establecer dos etapas claramente diferenciadas. En la primera, se determinan, para cada \mathcal{F} -componente de la coalición N , los dividendos de todas las coaliciones factibles según el siguiente esquema: sea $C_{\mathcal{F}}(N) = \{M_1, \dots, M_p\}$, entonces

$$\Delta_{v^{\mathcal{F}}}(\emptyset) \leftarrow 0$$

```

para  $j$  de 1 hasta  $p$ 
    dividendo( $M_j, v_{M_j}^{\mathcal{F}}$ )
        { Calcula  $\Delta_{v_{M_j}^{\mathcal{F}}}(S)$ , para toda  $S \in \mathcal{F}_{M_j}$  }
        {  $\Delta_{v^{\mathcal{F}}}(S) = \Delta_{v_{M_j}^{\mathcal{F}}}(S)$  }
    fin_para

```

En la segunda etapa, se determina, una vez conocidos todos los dividendos, el valor de Myerson generalizado. El algoritmo es como se indica

```

para  $j$  de 1 hasta  $p$ 
    para  $i$  de 1 hasta  $|M_j|$ 
         $\mu_i^j(N, v, \mathcal{F}) \leftarrow \sum_{\{S \in \mathcal{F}_{M_j} : i \in S\}} \frac{\Delta_{v_{M_j}^{\mathcal{F}}}(S)}{|S|}$ 
        {  $\mu_i^j(N, v, \mathcal{F}) = \mu_i(M_j, v_{M_j}, \mathcal{F}_{M_j})$  }
    fin_para
fin_para

```

Por tanto, $t(Myerson) = f(n) + g(n)$, siendo $f(n)$ y $g(n)$ las funciones que indican, respectivamente, el tiempo de cálculo de las dos etapas establecidas; es decir, el tiempo correspondiente a la determinación de los dividendos y, una vez conocidos éstos, el tiempo de cálculo del valor de Myerson de cada uno de los jugadores.

Ahora bien,

$$f(n) = t(bucle) \leq p t(\text{dividendo}(M, v_M^{\mathcal{F}})) \leq p 3^{|M|},$$

donde $M \in C_{\mathcal{F}}(N)$ y verifica que $|M| = \max\{|M_j|, 1 \leq j \leq p\}$. De igual

forma,

$$g(n) = t(\text{bucle1}) = pt(\text{bucle2}) \leq p|M|t(\text{suma}) \leq p|M|2^{|M|}.$$

Luego,

$$\mathcal{O}(t(\text{Myerson})) = \mathcal{O}(f(n) + g(n)) = \mathcal{O}(\max(p3^{|M|}, p|M|2^{|M|})).$$

Finalmente, como

$$\lim_{|M| \rightarrow \infty} \frac{|M|2^{|M|}}{3^{|M|}} = 0,$$

se tiene que el cálculo del valor de Myerson para todos los jugadores requiere un tiempo $\mathcal{O}(\max(p3^{|M|}, p|M|2^{|M|})) = \mathcal{O}(p3^{|M|})$.

Es inmediato que el espacio requerido es $\Omega(|\mathcal{F}|)$ debido a que, necesariamente, han de obtenerse todos los dividendos de Harsanyi. \square

3.4 El valor de Myerson generalizado en geometrías convexas.

Anteriormente se ha descrito que, para juegos restringidos por situaciones de comunicación, Owen encuentra simplificaciones para el cálculo de los dividendos cuando el grafo es un árbol [44]. Bilbao y López [12] ponen de manifiesto que estos resultados de Owen se generalizan cuando el sistema de coaliciones factibles es una geometría convexa de partición, entendiéndose por la misma a un par (N, \mathcal{L}) que verifica el ser una geometría convexa y un sistema de partición.

Los dividendos del juego restringido son determinantes para calcular el valor de Myerson generalizado. Cuando el sistema de coaliciones factibles es una geometría convexa (N, \mathcal{L}) , éstos pueden obtenerse con la siguiente fórmula

$$\Delta_{v^c}(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} v^{\mathcal{L}}(T).$$

Sin embargo interesa estudiar bajo qué condiciones el valor de Myerson puede calcularse en términos de los dividendos del juego (N, v) y, si es posible, de su función característica.

Teorema 3.62 Sean (N, \mathcal{F}) un sistema de partición y $u_T \in \Gamma^N$ un juego de unanimidad. Entonces, el juego restringido $u_T^{\mathcal{F}}$ viene definido, para toda $S \subseteq N$, por

$$u_T^{\mathcal{F}}(S) = \begin{cases} 1, & \text{si existe } F \in \mathcal{F} \text{ tal que } T \subseteq F \subseteq S \\ 0, & \text{en otro caso.} \end{cases}$$

Además, para cada $S \subseteq N$,

$$\Delta_{u_T^{\mathcal{F}}}(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} u_T^{\mathcal{F}}(L).$$

Demostración: Al ser los juegos de unanimidad u_T , simples y propios (ya que son superaditivos), se tiene que los juegos restringidos, $u_T^{\mathcal{F}}$, también son simples y propios.

Además,

$$u_T^{\mathcal{F}}(S) = \max\{u_T(R) : R \in C_{\mathcal{F}}(S)\}.$$

Luego $u_T^{\mathcal{F}}(S) = 1$ si y sólo si $u_T(R) = 1$ para una única $R \in C_{\mathcal{F}}(S)$. Ello será cierto únicamente cuando $T \subseteq R$, lo cual es equivalente a que exista un $F \in \mathcal{F}$ tal que $T \subseteq F \subseteq S$. Por tanto, para toda $S \subseteq N$, se verifica

$$u_T^{\mathcal{F}}(S) = \begin{cases} 1, & \text{si existe } F \in \mathcal{F} \text{ tal que } T \subseteq F \subseteq S \\ 0, & \text{en otro caso.} \end{cases}$$

Al ser $\{u_T : T \in 2^N, T \neq \emptyset\}$ base de Γ^N formada por los juegos de unanimidad, se tiene que, fijada $T \subseteq N, T \neq \emptyset$, el juego restringido $u_T^{\mathcal{F}} \in \Gamma^N$

y, por tanto, puede expresarse como combinación lineal de los juegos de unanimidad:

$$u_T^{\mathcal{F}} = \sum_{\{L \subseteq N, L \neq \emptyset\}} \Delta_{u_T^{\mathcal{F}}}(L) u_L$$

Así, para cualquier coalición $S \subseteq N$,

$$u_T^{\mathcal{F}}(S) = \sum_{\{L \subseteq N, L \neq \emptyset\}} \Delta_{u_T^{\mathcal{F}}}(L) u_L(S) = \sum_{L \subseteq S} \Delta_{u_T^{\mathcal{F}}}(L)$$

ya que $u_L(S) = 1$ si y sólo si $L \subseteq S$.

Si se considera el conjunto parcialmente ordenado $(2^N, \subseteq)$ y las funciones

$$u_T^{\mathcal{F}}, \Delta_{u_T^{\mathcal{F}}} : 2^N \longrightarrow \mathbb{R},$$

resulta que, aplicando la fórmula de inversión de Möebius [56, p. 116]

$$u_T^{\mathcal{F}}(S) = \sum_{L \subseteq S} \Delta_{u_T^{\mathcal{F}}}(L) \iff \Delta_{u_T^{\mathcal{F}}}(S) = \sum_{L \subseteq S} \mu(L, S) u_T^{\mathcal{F}}(L),$$

siendo $\mu(L, S) = (-1)^{|S|-|L|}$.

□

Teorema 3.63 *Sea la terna (N, v, \mathcal{F}) , donde (N, \mathcal{F}) es un sistema de partición y (N, v) un juego. Para cualquier $S \in \mathcal{F}$, los dividendos del juego restringido vienen determinados por*

$$\Delta_{v^{\mathcal{F}}}(S) = \sum_{\{T \subseteq S, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^{\mathcal{F}}}(S).$$

Demostración: Por un lado, teniendo en cuenta la linealidad del operador $L_{\mathcal{F}} : \Gamma^N \rightarrow \Gamma^N$, $L_{\mathcal{F}}(v) = v^{\mathcal{F}}$, y las bases de juegos de unanimidad existentes en los espacios Γ^N y $L_{\mathcal{F}}(\Gamma^N)$, resulta que las coordenadas o dividendos $\Delta_{v^{\mathcal{F}}}(S)$ del juego restringido $v^{\mathcal{F}}$ vienen determinados por

$$v^{\mathcal{F}} = L_{\mathcal{F}}(v) = L_{\mathcal{F}} \left(\sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) u_T \right) = \sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) L_{\mathcal{F}}(u_T)$$

$$\begin{aligned}
 &= \sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) u_T^{\mathcal{F}} = \sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) \left[\sum_{\{S \in \mathcal{F}, S \neq \emptyset\}} \Delta_{u_T^{\mathcal{F}}}(S) u_S \right] \\
 &= \sum_{\{S \in \mathcal{F}, S \neq \emptyset\}} \left[\sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^{\mathcal{F}}}(S) \right] u_S.
 \end{aligned}$$

De otra parte,

$$v^{\mathcal{F}} = \sum_{\{S \in \mathcal{F}, S \neq \emptyset\}} \Delta_{v^{\mathcal{F}}}(S) u_S,$$

y de ahí se verifica que

$$\Delta_{v^{\mathcal{F}}}(S) = \sum_{\{T \subseteq N, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^{\mathcal{F}}}(S), \quad \text{para cada } S \in \mathcal{F}.$$

Debido a la proposición anterior,

$$\Delta_{u_T^{\mathcal{F}}}(S) = \sum_{L \subseteq S} \mu(L, S) u_T^{\mathcal{F}}(L)$$

y $u_T^{\mathcal{F}}(L) = 1$ si y sólo si existe $F \in \mathcal{F}$ tal que $T \subseteq F \subseteq L \subseteq S$. Por tanto, si $T \not\subseteq S$ entonces $u_T^{\mathcal{F}}(L) = 0$ para toda $L \subseteq S$ y $\Delta_{u_T^{\mathcal{F}}}(S) = 0$. Queda finalmente

$$\Delta_{v^{\mathcal{F}}}(S) = \sum_{\{T \subseteq S, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^{\mathcal{F}}}(S).$$

□

Puede observarse que la expresión resultante es, en la práctica, bastante difícil de evaluar. Sin embargo, en el caso de que el sistema (N, \mathcal{F}) sea una geometría convexa de partición se obtienen simplificaciones significativas. Ello se pone de manifiesto en los siguientes teoremas en los que, en primer lugar, se generaliza el resultado de Owen [44] que relaciona los dividendos del juego restringido $(N, v^{\mathcal{F}})$ con los del juego (N, v) . El segundo teorema permitirá hacer un cálculo directo de los dividendos del juego restringido mediante los valores del juego (N, v) .

Proposición 3.64 Sea (N, \mathcal{L}) una geometría convexa de partición. Para toda coalición no vacía $T \subseteq N$, se tiene que

$$u_T^{\mathcal{L}} = u_{\bar{T}}.$$

Demostración: Por el Teorema 3.62, para toda coalición $S \subseteq N$,

$$u_T^{\mathcal{L}}(S) = \begin{cases} 1, & \text{si existe } C \in \mathcal{L} \text{ tal que } T \subseteq C \subseteq S \\ 0, & \text{en otro caso,} \end{cases}$$

y, aplicando el concepto de clausura, se obtiene que $u_T^{\mathcal{L}}(S) = 1$ si y sólo si $T \subseteq \bar{T} \subseteq S$ si y sólo si $u_{\bar{T}}(S) = 1$, con lo que $u_T^{\mathcal{L}} = u_{\bar{T}}$, para toda $T \subseteq N$, $T \neq \emptyset$.

□

Teorema 3.65 Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego. Para cualquier $S \in \mathcal{L}$, los dividendos del juego restringido por la geometría convexa de partición, $(N, v^{\mathcal{L}})$, verifican

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T: \bar{T}=S\}} \Delta_v(T) = \sum_{\{T: ex(S) \subseteq T \subseteq S\}} \Delta_v(T).$$

Demostración: Haciendo un razonamiento análogo al de la demostración del Teorema 3.27 y aplicando, además, la Proposición 3.64, se obtiene

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T \subseteq S, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_T^{\mathcal{L}}}(S) = \sum_{\{T \subseteq S, T \neq \emptyset\}} \Delta_v(T) \Delta_{u_{\bar{T}}}(S),$$

para cada $S \in \mathcal{L}$. Ahora bien,

$$u_{\bar{T}} = \sum_{\{S \in \mathcal{L}, S \neq \emptyset\}} \Delta_{u_{\bar{T}}}(S) u_S,$$

y como $\bar{T} \in \mathcal{L}$, resulta que

$$\Delta_{u_{\bar{T}}}(S) = 1 \iff S = \bar{T}$$

Ello, junto con el hecho de ser

$$\Delta_v(\emptyset) = \sum_{H \subseteq \emptyset} (-1)^{|H|} v(H) = v(\emptyset) = 0,$$

hace que se obtenga que

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T: \overline{T}=S\}} \Delta_v(T).$$

Finalmente, quedaría probar que $\{T : \overline{T} = S\} = \{T : ex(S) \subseteq T \subseteq S\}$. En efecto, sea T tal que $\overline{T} = S$. Como $S = \overline{S}$, por ser $S \in \mathcal{L}$, es obvio que $T \subseteq \overline{T} = S$. Al ser $S \in \mathcal{L}$, la propiedad de Minkowski–Krein–Milman indica que $ex(S)$ es base de S por lo que $ex(S)$ es un conjunto minimal tal que $\overline{ex(S)} = S$. El ser minimal asegura que $ex(S) \subseteq T \subseteq S$.

Para establecer la inclusión contraria, se supone que $ex(S) \subseteq T \subseteq S$. Se verifica entonces

$$S = \overline{ex(S)} \subseteq \overline{T} \subseteq \overline{S} = S,$$

por lo que $\overline{T} = S$.

□

Teorema 3.66 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego. Si $S \in \mathcal{L}$, entonces*

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T: S \setminus ex(S) \subseteq T \subseteq S\}} (-1)^{|S|-|T|} v(T).$$

Demostración: La colección $\{u_T : T \in \mathcal{L}, T \neq \emptyset\}$ es una base del espacio de los juegos restringidos. De aquí, para cada $S \in \mathcal{L}$, se obtiene

$$v(S) = v^{\mathcal{L}}(S) = \sum_{\{T \in \mathcal{L}: T \subseteq S\}} \Delta_{v^{\mathcal{L}}}(T).$$

Si se consideran las funciones

$$v, \Delta_{v^{\mathcal{L}}} : \mathcal{L} \longrightarrow \mathbb{R},$$

y se aplica la fórmula de inversión de Möebius al conjunto parcialmente ordenado (\mathcal{L}, \subseteq) , los dividendos de $v^{\mathcal{L}}$ quedan expresados en función de los valores de v . En efecto, para toda coalición $S \in \mathcal{L}$,

$$v(S) = \sum_{\{T \in \mathcal{L}: T \subseteq S\}} \Delta_{v^{\mathcal{L}}}(T) \iff \Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T \in \mathcal{L}: T \subseteq S\}} v(T) \mu(T, S).$$

La función de Möebius en el caso de una geometría convexa viene dada, en [30, Teorema 4.3], por

$$\mu(T, S) = \begin{cases} (-1)^{|S|-|T|}, & \text{si } S \setminus T \subseteq \text{ex}(S) \\ 0, & \text{en otro caso.} \end{cases}$$

Para $T \subseteq S$, se tiene que $S \setminus T \subseteq \text{ex}(S)$ si y sólo si $S \setminus \text{ex}(S) \subseteq T$. En consecuencia,

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T: S \setminus \text{ex}(S) \subseteq T \subseteq S\}} (-1)^{|S|-|T|} v(T).$$

□

Nótese que si $\mathcal{L} = 2^N$, entonces todos los elementos de cualquier coalición son puntos extremales de la misma. Es decir, para toda $S \in 2^N$, se tiene que $S \setminus \text{ex}(S) = \emptyset$, y como $v^{\mathcal{L}} = v$, se obtiene, debido al Teorema 3.66, que para toda $S \in 2^N$,

$$\Delta_v(S) = \Delta_{v^{\mathcal{L}}}(S) = \sum_{T \subseteq S} (-1)^{|S|-|T|} v(T),$$

que es la expresión correspondiente a los dividendos de Harsanyi [35].

Corolario 3.67 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego. El valor de Myerson generalizado viene dado por,*

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{S \in \mathcal{L}: i \in S\}} \frac{1}{|S|} \left[\sum_{T \in [S^-, S]} (-1)^{|S|-|T|} v(T) \right],$$

para todo $i \in N$.

Este resultado una consecuencia inmediata del Teorema 3.66 ya que para cada $S \in \mathcal{L}$, el intervalo $[S^-, S]$ es un álgebra de Boole y $S^- = S \setminus ex(S)$.

Ejemplo 3.68 Sea la terna (N, v, \mathcal{L}) , siendo $N = \{1, 2, 3, 4\}$, $v(S) = |S| - 1$ para toda coalición no vacía $S \subseteq N$, y \mathcal{L} la familia dada por

$$\mathcal{L} = \{ \emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 2\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{1, 2, 3\}, \{1, 2, 4\}, \{2, 3, 4\}, N \}.$$

Puede observarse (véase Figura 3.5) que el par (N, \mathcal{L}) es una geometría convexa de partición.

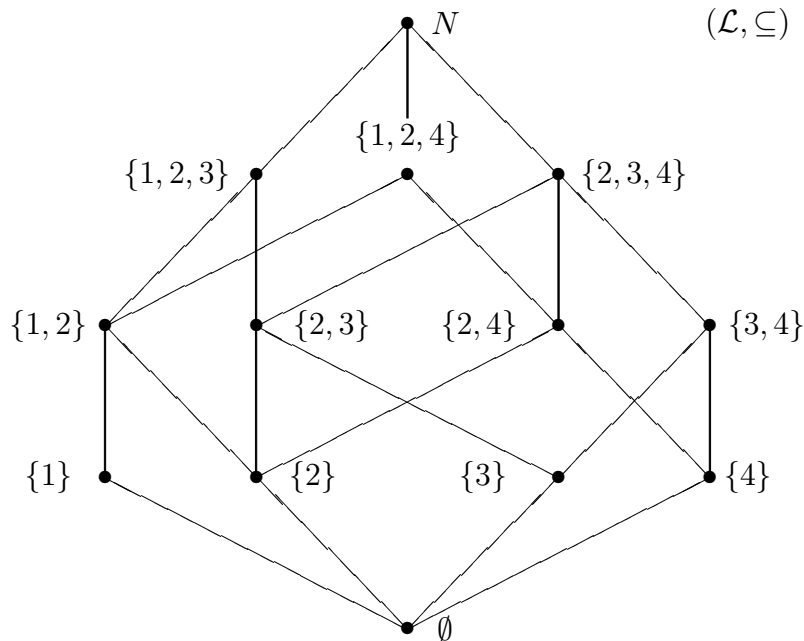


FIGURA 3.5

Utilizando resultados anteriores, el valor de Myerson generalizado viene determinado, para cada $i \in N$, por

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{S \in \mathcal{L}: i \in S\}} \frac{\Delta_{v, \mathcal{L}}(S)}{|S|},$$

con $\Delta_{v\mathcal{L}}(S) = \sum_{T \in [S^-, S]} (-1)^{|S|-|T|} v(T)$ para $S \in \mathcal{L}$.

Por tanto, teniendo en cuenta que el intervalo $[S^-, S]$ es un álgebra de Boole en el retículo (\mathcal{L}, \subseteq) , se calculan en primer lugar los dividendos del juego restringido.

i) Para todo $i \in N$, $\Delta_{v\mathcal{L}}(\{i\}) = \sum_{T \in [\emptyset, \{i\}]} (-1)^{1-|T|} v(T) = 0$.

ii) Si $S \in \mathcal{L}$, con $|S| = 2$, se tiene

$$\Delta_{v\mathcal{L}}(\{i, j\}) = \sum_{T \in [\emptyset, \{i, j\}]} (-1)^{2-|T|} v(T) = v(\{i, j\}) = 1.$$

iii) Para las coaliciones factibles $\{1, 2, 3\}$, $\{1, 2, 4\}$ y $\{2, 3, 4\}$:

$$\Delta_{v\mathcal{L}}(\{1, 2, 3\}) = \sum_{T \in [\{2\}, \{1, 2, 3\}]} (-1)^{3-|T|} v(T) = 0,$$

$$\Delta_{v\mathcal{L}}(\{1, 2, 4\}) = \sum_{T \in [\{2\}, \{1, 2, 4\}]} (-1)^{3-|T|} v(T) = 0,$$

$$\Delta_{v\mathcal{L}}(\{2, 3, 4\}) = \sum_{T \in [\emptyset, \{2, 3, 4\}]} (-1)^{3-|T|} v(T) = -1.$$

iv) Si $S = N$, entonces $\Delta_{v\mathcal{L}}(N) = \sum_{T \in [\{2\}, N]} (-1)^{4-|T|} v(T) = 0$.

De ahí,

$$\begin{aligned} \mu_1(N, v, \mathcal{L}) &= \sum_{\{S \in \mathcal{L}: 1 \in S\}} \frac{\Delta_{v\mathcal{L}}(S)}{|S|} \\ &= \Delta_{v\mathcal{L}}(\{1\}) + \frac{1}{2} \Delta_{v\mathcal{L}}(\{1, 2\}) \\ &\quad + \frac{1}{3} [\Delta_{v\mathcal{L}}(\{1, 2, 3\}) + \Delta_{v\mathcal{L}}(\{1, 2, 4\})] + \frac{1}{4} \Delta_{v\mathcal{L}}(N) \\ &= \frac{1}{2}. \end{aligned}$$

Análogamente,

$$\mu_3(N, v, \mathcal{L}) = \mu_4(N, v, \mathcal{L}) = \frac{2}{3}, \quad \mu_2(N, v, \mathcal{L}) = \frac{7}{6}.$$

Nótese que
$$\sum_{i \in N} \mu_i(N, v, \mathcal{L}) = 3 = v^{\mathcal{L}}(N).$$

Obsérvese que estos resultados generalizan, para juegos restringidos por geometrías convexas de partición, los resultados dados para juegos restringidos por árboles. Aquí las coaliciones convexas desempeñan el papel que allí jugaban las coaliciones conexas. Por tanto, se puede emplear el algoritmo *dividendo2* para el cálculo del valor de Myerson generalizado en juegos restringidos por geometrías convexas de partición, siendo similar el análisis de la complejidad espacial y temporal para el cálculo de los dividendos y del valor de Myerson generalizado.

Los dividendos se calculan mediante un proceso ascendente que exige almacenar los dividendos de cada una de las coaliciones. En el análisis se considera un almacenamiento previo en una tabla de los extremales de cada coalición convexa. Son inmediatos los siguientes resultados.

Proposición 3.69 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un tiempo $\mathcal{O}(2^D |\mathcal{L}|)$, donde $D = \max \{|ex(S)| : S \in \mathcal{L}\}$.*

Proposición 3.70 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego de utilidad transferible. El cálculo de todos los dividendos de Harsanyi del juego restringido requiere un espacio $\Omega(|\mathcal{L}|)$.*

Proposición 3.71 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego de utilidad transferible. El cálculo del valor de Myerson generalizado de todos los jugadores, utilizando los dividendos de Harsanyi, requiere un tiempo $\mathcal{O}(\max(n |\mathcal{L}|, 2^D |\mathcal{L}|))$.*

3.5 Algoritmos con MATHEMATICA

En esta sección se describe el cálculo del valor de Myerson en diversos juegos de votación, utilizando las funciones `MyersonValue` y `ConvexValue` que implementan respectivamente los algoritmos *dividendo* y *dividendo2*.

Obsérvese que la sintaxis de `MyersonValue` es exactamente la misma que la de la función `ShapleyValue4` introducida en el capítulo segundo, con la salvedad de que ahora se consideran únicamente las coaliciones conexas.

Ejemplo 3.72 *Sea el juego de votación correspondiente al actual pleno del Ayuntamiento de Sevilla. Los jugadores, con expresión del número de votos, son los partidos con representación municipal:*

Jugador 1: PP, 13 votos.

Jugador 2: PSOE, 12 votos.

Jugador 3: PA, 6 votos.

Jugador 4: IU, 2 votos.

Para la elección de Alcalde/sa en primera votación se requieren 17 votos. Este juego simple se representa por

$$v = [q; 13, 12, 6, 2], \text{ siendo } q = 17.$$

Se considera que no todas las coaliciones son posibles y que hay un alineamiento de los grupos de izquierda a derecha

$$PP - PA - PSOE - IU.$$

*Esta situación conduce al juego restringido por la situación de comunicación (N, v, G) , donde G es el grafo camino anterior. A continuación se indica, usando MATHEMATICA, el proceso de cálculo del valor de Myerson, usando la función `MyersonValue` que incorpora el algoritmo *dividendo*. Recuérdese que, previamente, habría que cargar los paquetes `DiscreteMath'Combinatorica'` y `Cooperat'Cooperat'`.*

En primer lugar se introduce el número de jugadores, el número de votos de cada uno de ellos y la función característica del juego.

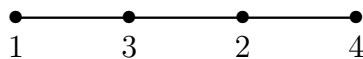
```
In[1] :=
  Sevilla[n_] := (Clear[T,w,p,v]; T=Range[4];
  w[1]:=13;
  w[2]:=12;
  w[3]:=6;
  w[4]:=2;
  p[S_List] := Apply[Plus, w/@S]; v[{}]:=0;
  v[S_ /; p[S]>=n]:=1; v[S_ /; p[S]<n]:=0;)
```

A continuación se muestra el grafo de cooperación junto con la familia de coaliciones factibles.

```
In[2] :=
  grafo=FromUnorderedPairs[{{1,3},{3,2},{2,4}}];
```

```
In[3] :=
  grafo[[2]]={{0,0},{2,0},{1,0},{3,0}}; ShowLabeledGraph[grafo]
```

Out[3] =



```
In[4] :=
  Clear[conexas];
  conexas=Select[Rest[Subsets[Range[V[grafo]]]],
```

```
(ConnectedQ[InduceSubgraph[grafo,#]])&];
```

```
Out[4] =
```

```
{ {1}, {2}, {3}, {4}, {1, 3}, {2, 3}, {2, 4}, {1, 2, 3}, {2, 3, 4},
  {1, 2, 3, 4} }
```

```
In[5] :=
```

```
div[{i_}] := div[{i}] = v[{i}];
div[S_] := div[S] =
v[S] - (Apply[Plus, div[#] & /@
Drop[Intersection[Subsets[S], conexas], -1]]);
```

```
In[6] :=
```

```
MyersonValue[game_ : Null] :=
Module[{value},
value = Table[Apply[Plus,
(div[#]/Length[#]) & /@
Select[conexas, (MemberQ[#, i]) &]], {i, Length[T]}];
div[#] =. & /@ conexas;
Return[value];
```

```
In[7] :=
```

```
MyersonValue[Sevilla[17]]
```

```
Out[7] =
```

```
{ 1/6, 1/6, 2/3, 0 }
```

Como se observa, el jugador número 3 (PA) incrementa notablemente su poder en este contexto de cooperación parcial y es el auténtico árbitro de la situación.

En el juego con cooperación total, el valor de Shapley viene dado por

```
In[8] :=
    ShapleyValue3[Sevilla[17]]
```

```
Out[8] =
    {1/3, 1/3, 1/3, 0}
```

Nótese que cuando el grafo es un camino y el número de jugadores es elevado, las coaliciones conexas no vacías pueden obtenerse implementando la expresión

$$\mathcal{F} = \{[i, j], 1 \leq i \leq j \leq n\}.$$

Obviamente, ello exigiría reenumerar los jugadores conforme a su disposición en el grafo. Así, en MATHEMATICA, sería de la siguiente forma

```
In[ ] :=
    LinOrder[n_Integer]:=Flatten[Table[Range[i,j],{i,n},{j,i,n}],1]
```

```
In[ ] :=
    Clear[conexas];conexas=Sort[LinOrder[4]]
```

Ejemplo 3.73 *El método de votación para el Consejo de Gobierno de la Unión Europea se modela mediante el juego (N, v) , donde*

$$N = \{Alemania(1), Reino Unido(2), Francia(3), Italia(4), España(5), \\ Holanda(6), Grecia(7), Bélgica(8), Portugal(9), Suecia(10), \\ Austria(11), Dinamarca(12), Finlandia(13), Irlanda(14), \\ Luxemburgo(15)\},$$

$$v = [q ; 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2], \quad 61 \leq q \leq 68.$$

*Se considera el juego restringido por la situación de comunicación (N, v, G) , donde G es el grafo denominado grafoUE. A continuación, se indica el cálculo del valor de Myerson usando las funciones *MyersonValue* y *ConvexValue*.*

In[1] :=

```
JuegoUE[n_]:= (Clear[T,w,p,v];T=Range[15];
w[i_Integer/;1<=i && i<=4]:=10; w[5]:=8;
w[i_Integer/;6<=i && i<=9]:=5; w[10]:=4; w[11]:=4;
w[12]:=3; w[13]:=3; w[14]:=3; w[15]:=2;
p[S_List]:=Apply[Plus, w/@S]; v[{}]:=0;
v[S_ /; p[S]>=n]:=1;v[S_ /; p[S]<n]:=0;)
```

In[2] :=

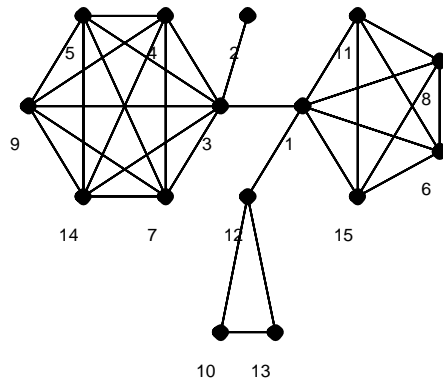
```
grafoUE=FromUnorderedPairs[{{1,3},{1,6},{1,8},
{1,11},{1,12},{1,15},{2,3},{3,4},{3,5},{3,7},{3,9},
{3,14},{4,5},{4,7},{4,9},{4,14},{5,7},{5,9},{5,14},
{6,8},{6,11},{6,15},{7,9},{7,14},{8,11},{8,15},
{9,14},{10,12},{10,13},{11,15},{12,13}}];
```

In[3] :=

```
grafoUE[[2]]={{2,0},{0,1},{-1,0},{-3,1},{-6,1},
{7,-0.5},{-3,-1},{7,0.5},{-8,0},{-1,-2.5},{4,1},
{0,-1},{1,-2.5},{-6,-1},{4,-1}};
```

In[4] :=

```
ShowLabeledGraph[grafoUE];
```



```
In[5] :=
Clear[conexas];
Timing[conexas=Select[Rest[Subsets[Range[V[grafoUE]]]],
(ConnectedQ[InduceSubgraph[grafoUE,#]])&];]
```

```
Out[5] =
{370.85 Second, Null}
```

```
In[6] := div[{i_}]:=div[{i}]=v[{i}];
div[S_]:=div[S]=
v[S]-(Apply[Plus,div[#]& /@
Drop[Intersection[Subsets[S],conexas],-1]]);
```

```
In[7] :=
MyersonValue[game_ :Null]:=
Module[{value},
value=Table[Apply[Plus,
(div[#]/Length[#])& /@
Select[conexas,(MemberQ[#,i])&]],{i,Length[T]}];
div[#]=. & /@ conexas;
Return[value];]
```

```
In[8] :=
  N[Timing[MyersonValue[JuegoUE[62]]]]
```

```
Out[8] =
  {590.39 Second {0.292469, 0.0628427, 0.292469, 0.0628427,
  0.0511711, 0.0282384, 0.0282384, 0.0282384, 0.0282384, 0.0149073,
  0.0246587, 0.0449939, 0.0110612, 0.0182873, 0.0113442}}
```

La familia de coaliciones factibles derivada de la situación de comunicación (N, v, G) considerada en este ejemplo es una geometría convexa de partición, pues G es un grafo bloque conexo. Ello permite calcular el valor de Myerson considerando la función `ConvexValue` que implementa el algoritmo *dividendo2*. Una vez determinadas las coaliciones conexas, el proceso de cálculo se indica a continuación.

```
In[9] :=
  F=Union[conexas,{{}];
```

```
In[10] :=
  exS[S_List,F_List]:=exS[S,F]=
  Select[S,MemberQ[F,Complement[S,{#}]]&];
```

```
In[11] :=
  Smenos[S_List,F_List]:=Smenos[S,F]=Module[{S3},
  S3=Complement[S,exS[S,F]];
  Flatten[Append[S3,#]]& /@Subsets[exS[S,F]]];
```

```
In[12] :=
  div2[{i_}]:=div2[{i}]=v[{i}];
  div2[S_]:=div2[S]=(Apply[Plus,
  ((-1)^(Length[S]-Length[#]))v[#]& /@Smenos[S,F]]);
```

```
In[13] :=
```



```
ConvexValue[game_ :Null]:=Module[{value},
value=Table[Apply[Plus,(div2[#]/Length[#])&
/@Select[F,(MemberQ[#,i])&]],{i,Length[T]}];
div2[#]=. & /@ Rest[F];Return[value];
```

```
In[14] :=
N[Timing[ConvexValue[JuegoUE[62]]]]
```

```
Out[14] =
{4432.33 Second {0.292469, 0.0628427, 0.292469, 0.0628427,
0.0511711, 0.0282384, 0.0282384, 0.0282384, 0.0282384, 0.0149073,
0.0246587, 0.0449939, 0.0110612, 0.0182873, 0.0113442}}
```

Durante el proceso de cálculo se han almacenado los extremales de cada coalición. Si ahora se realiza el cálculo con una cuota diferente, $q = 65$, el tiempo de cálculo baja sensiblemente.

```
In[15] :=
N[Timing[ConvexValue[JuegoUE[65]]]]
```

```
Out[15] =
{1613.77 Second,{0.265346, 0.0727384, 0.265346, 0.0727384,
0.0567544, 0.0327561, 0.0327561, 0.0327561, 0.0327561, 0.0155012,
0.0228827, 0.057265, 0.0121712, 0.0195527, 0.00868021}}
```

Como se observa, para este ejemplo de la Unión Europea, el cálculo con MyersonValue ha requerido menos tiempo que el cálculo con ConvexValue. Es decir, el cálculo de los dividendos con el algoritmo *dividendo* ha sido más eficaz que con *dividendo2*. La razón está en el hecho de que para usar el algoritmo *dividendo2* hay que determinar previamente los extremales de cada coalición $S \in \mathcal{F}$, y los intervalos $[S^-, S]$.

Sin embargo, cuando se calcula el valor de Myerson en el juego de votación

correspondiente a la actual composición del Congreso de los Diputados de España, considerando un alineamiento de los partidos de izquierda a derecha, los tiempos de cálculo del valor de Myerson usando `MyersonValue` y `ConvexValue` son similares, e inferiores a un segundo (0.71 seg. y 0.62 seg. respectivamente). En este caso, los jugadores extremales de cada coalición factible coinciden con el primer y el último jugador de cada coalición conexa considerando el orden establecido por el grafo.

Capítulo 4

Funciones generatrices para calcular índices de poder

4.1 Funciones generatrices

El cálculo de los índices de poder en juegos de votación es un problema que puede afrontarse utilizando los algoritmos para calcular el valor de Shapley descritos en los capítulos anteriores. Obviamente, la complejidad de estos algoritmos limita sobremanera la capacidad de cálculo, y hace muy difícil estudiar un juego de votación ponderada con más de 20 jugadores. Ahora bien, en el caso de juegos simples de votación ponderada, los índices de poder vienen determinados por sumas extendidas sobre las coaliciones ganadoras minimales a las que pertenece cada jugador y, afortunadamente, hay una forma de contar este número sin tener que inspeccionar todas las coaliciones posibles, 2^n . Ello implica utilizar un procedimiento que involucra el uso de funciones generatrices y que constituye, en análisis combinatorio, un método indirecto para el cálculo del número de configuraciones de determinados tipos. En esta introducción se describe, brevemente, su concepto.

Considérese la sucesión de números reales $\{a_j\}_{j \geq 0}$. A esta sucesión le

corresponde biunívocamente la serie

$$\sum_{j=0}^{\infty} a_j x^j,$$

la cual es cómoda y simple en las operaciones, particularmente cuando converge a una función real $f(x)$ que posee una forma analítica conveniente. Es decir,

$$\sum_{j=0}^{\infty} a_j x^j = f(x).$$

La función $f(x)$ se denomina *función generatriz ordinaria* de la sucesión $\{a_j\}_{j \geq 0}$.

Por ejemplo, sea la sucesión de números reales $\{1, 1, 1, \dots\}$. A esta sucesión se le puede asociar la serie

$$1 + x + x^2 + x^3 + \dots = \sum_{j=0}^{\infty} x^j,$$

que es convergente a la función $f(x) = \frac{1}{1-x}$, siempre que $|x| < 1$. Entonces, la función $f(x)$ genera la sucesión $\{a_j\}_{j \geq 0}$, con $a_j = 1$ para todo j , ya que su desarrollo en serie de potencias da lugar a identificar los coeficientes numéricos de los infinitos sumandos con los elementos de la sucesión. Por tanto, puede afirmarse que una función generatriz ordinaria genera una sucesión $\{a_j\}_{j \geq 0}$ cuyos elementos son números reales.

En general, en el estudio y utilización de las funciones generatrices ordinarias, la variable x no tiene un significado propio por sí misma y sólo se introduce para identificar a_j como el coeficiente correspondiente a x^j en el desarrollo de $f(x)$. Por tanto, la cuestión relativa a la convergencia de la serie no es relevante si x se interpreta como una variable formal.

Las funciones generatrices ordinarias proporcionan un método para contar el número de elementos $c(k)$ de un conjunto finito, cuando estos elementos poseen una determinada configuración dependiente de una variable k , para

lo cual se construye la serie de potencias formal —ya que se ignora tanto la evaluación en valores particulares como los problemas de convergencia—

$$\sum_{k \geq 0} c(k)x^k,$$

y se investiga su función generatriz ordinaria. Así, por ejemplo, para cualquier $n \in \mathbb{N}$, el número de subconjuntos de k elementos de $N = \{1, 2, \dots, n\}$ viene dado por la fórmula explícita de los coeficientes binomiales

$$\binom{n}{k} = \frac{n(n-1) \cdots (n-k+1)}{k!},$$

y una función generatriz ordinaria que aproxima los coeficientes binomiales viene dada por

$$\sum_{k \geq 0} \binom{n}{k} x^k = (1+x)^n.$$

Para un estudio exhaustivo de los fundamentos del método de funciones generatrices, tipos de funciones generatrices y aparato operacional del método, pueden consultarse los trabajos de Stanley [56] y Ribnikov [47]. En lo sucesivo, para simplificar la notación, $c(k)$ se escribirá como c_k .

Por último, dado que van a utilizarse en secciones posteriores, es preciso indicar que se pueden considerar funciones generatrices ordinarias de varias variables, como las correspondientes a la serie de potencias formal

$$\sum_{k \geq 0} \sum_{j \geq 0} \sum_{l \geq 0} c(k, j, l) x^k y^j z^l,$$

donde, en este caso, $c(k, j, l)$, que se denotará como c_{kjl} , quiere indicar el número de elementos de un conjunto finito con una determinada configuración dependiente de tres variables k , j y l .

4.2 Algoritmos para juegos de votación ponderada

En esta sección se presentan funciones generatrices para calcular los índices de poder de Banzhaf y de Shapley-Shubik en un juego de votación ponderada (N, v) , definido por

$$v = [q; w_1, \dots, w_n].$$

La utilización de funciones generatrices para calcular el índice de poder de Shapley-Shubik en juegos de votación tiene un antecedente en los trabajos de David G. Cantor (1962) [40] [41]. Posteriormente, Brams y Affuso (1976) [21] emplearon funciones generatrices para calcular el índice de Banzhaf normalizado. Se hace notar que, con objeto de facilitar la exposición de resultados, el orden establecido será diferente al histórico y, por dicho motivo, se describirá en primer lugar el cálculo del índice de Banzhaf.

4.2.1 Índice de poder de Banzhaf normalizado

Como se indicó en la sección 2.1.3, el índice de Banzhaf normalizado viene dado por el vector

$$\beta(v) = (\beta_1(v), \dots, \beta_n(v)), \text{ con } \beta_i(v) = \frac{\eta_i(v)}{\bar{\eta}(v)},$$

donde $\eta_i(v)$ es el número de swings del jugador i en el juego (N, v) , y $\bar{\eta}(v)$ es el número total de swings de todos los jugadores,

$$\bar{\eta}(v) = \sum_{i \in N} \eta_i(v).$$

Recuérdese que un swing para el jugador i está constituido por un par de coaliciones $(S \cup i, S)$ tal que $S \cup i$ es ganadora y S es perdedora. Así, el número de swings del jugador i es

$$\eta_i(v) = |\{S \notin \mathcal{W} : S \cup i \in \mathcal{W}\}|,$$

donde \mathcal{W} denota el conjunto de coaliciones ganadoras.

Proposición 4.1 *Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, el número de swings del jugador i viene dado por*

$$\eta_i(v) = \sum_{k=q-w_i}^{q-1} b_k^i,$$

siendo b_k^i el número de coaliciones $S \subseteq N$ tales que $i \notin S$ y $w(S) = k$.

Demostración: De entre todas las coaliciones que no incluyen al jugador i , aquellas coaliciones S con peso comprendido entre $q - w_i$ y $q - 1$ son coaliciones perdedoras. Si a estas coaliciones se incorpora el jugador i , el peso de la coalición $S \cup i$ será mayor o igual que la cuota y, por tanto, la coalición resultante sería ganadora.

Si se denota por b_k^i el número de coaliciones que no incluyen al jugador i y que tienen un peso k , resulta que, para calcular los swings del jugador i , basta con sumar los números $\{b_k^i\}_{k \geq 0}$ para valores de k comprendidos entre $q - w_i$ y $q - 1$. Por tanto, $\eta_i(v) = \sum_{k=q-w_i}^{q-1} b_k^i$.

□

El siguiente resultado, debido a Brams y Affuso [21] establece una función generatriz para obtener los números $\{b_k^i\}_{k \geq 0}$.

Proposición 4.2 *(Brams-Affuso (1976)) Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, para cualquier $i \in N$, la función generatriz de los números $\{b_k^i\}_{k \geq 0}$, donde, para cada k , b_k^i es el número de coaliciones $S \subseteq N$ tal que $i \notin S$, y $w(S) = k$, viene dada por*

$$B_i(x) = \prod_{j=1, j \neq i}^n (1 + x^{w_j}).$$

Demostración: Para los pesos $\{w_1, \dots, w_n\}$, se considera la función generatriz

$$\begin{aligned} (1 + x^{w_1}) \cdots (1 + x^{w_n}) &= \sum_{S \subseteq N} \prod_{i \in S} x^{w_i} \\ &= \sum_{S \subseteq N} (x^{\sum_{i \in S} w_i}) \\ &= \sum_{S \subseteq N} x^{w(S)}. \end{aligned}$$

Agrupando las potencias del mismo orden, y denotando $w(S) = k$, resulta que

$$(1 + x^{w_1}) \cdots (1 + x^{w_n}) = \sum_{k=0}^{w(N)} b_k x^k,$$

donde b_k , obviamente, es el número de coaliciones con peso k . Es inmediato que para obtener los números $\{b_k^i\}_{k \geq 0}$, basta suprimir el factor $(1 + x^{w_i})$. □

En el ejemplo siguiente se describe la utilización de la función generatriz indicada en la proposición anterior.

Ejemplo 4.3 *Considérese el juego de votación ponderada del Ejemplo 3.72, correspondiente al pleno del Ayuntamiento de Sevilla, y determinado por*

$$N = \{1, 2, 3, 4\}, \quad v = [q; 13, 12, 6, 2], \quad q = 17.$$

Si se calcula el índice de poder de Banzhaf normalizado mediante un procedimiento de cálculo directo habrá que determinar, primero, el peso y el valor de cada coalición.

Coalición	$w(S)$	$v(S)$
{}	0	0
{1}	13	0
{2}	12	0
{3}	6	0
{4}	2	0
{1, 2}	25	1
{1, 3}	19	1
{1, 4}	15	0

Coalición	$w(S)$	$v(S)$
{2, 3}	18	1
{2, 4}	14	0
{3, 4}	8	0
{1, 2, 3}	31	1
{1, 2, 4}	27	1
{1, 3, 4}	21	1
{2, 3, 4}	20	1
{1, 2, 3, 4}	33	1

A continuación se indican, para cada jugador i , las coaliciones $S \subseteq N$ tales que $v(S) = 0$ y $v(S \cup i) = 1$.

Jugador	Coaliciones
1	{{2}, {3}, {2, 4}, {3, 4}}
2	{{1}, {3}, {1, 4}, {3, 4}}
3	{{1}, {2}, {1, 4}, {2, 4}}
4	

y, por tanto, el número de swings para cada jugador es

Jugador	$\eta_i(v)$
1	4
2	4
3	4
4	0

Como consecuencia, el número total de swings es

$$\bar{\eta}(v) = \sum_{i \in N} \eta_i(v) = 12,$$

y el valor de Banzhaf normalizado es

$$\beta(v) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0 \right), \text{ donde } \beta_i(v) = \frac{\eta_i(v)}{\bar{\eta}(v)}.$$

Si se quiere realizar el cálculo del valor de Banzhaf utilizando las funciones generatrices indicadas en la proposición anterior habrá que determinar, en primer lugar, las funciones $B_i(x) = \prod_{j=1, j \neq i}^n (1 + x^{w_j})$.

$$\begin{aligned} B_1(x) &= (1 + x^{12})(1 + x^6)(1 + x^2) \\ &= 1 + x^2 + x^6 + x^8 + x^{12} + x^{14} + x^{18} + x^{20}, \\ B_2(x) &= (1 + x^{13})(1 + x^6)(1 + x^2) \\ &= 1 + x^2 + x^6 + x^8 + x^{13} + x^{15} + x^{19} + x^{21}, \\ B_3(x) &= (1 + x^{13})(1 + x^{12})(1 + x^2) \\ &= 1 + x^2 + x^{12} + x^{13} + x^{14} + x^{15} + x^{25} + x^{27}, \\ B_4(x) &= (1 + x^{13})(1 + x^{12})(1 + x^6) \\ &= 1 + x^6 + x^{12} + x^{13} + x^{18} + x^{19} + x^{25} + x^{31}. \end{aligned}$$

Para obtener el número de swings del jugador i basta considerar que

$$\eta_i(v) = \sum_{k=q-w_i}^{q-1} b_k^i.$$

Obsérvese que al sumar entre $q - w_i$ y $q - 1$ se tendrá el número total de coaliciones que pasan de perdedoras a ganadoras cuando se incorpora el jugador i . Por tanto,

$$\begin{aligned} \eta_1(v) &= \sum_{k=4}^{16} b_k^1 = 4, & \eta_2(v) &= \sum_{k=5}^{16} b_k^2 = 4, \\ \eta_3(v) &= \sum_{k=11}^{16} b_k^3 = 4, & \eta_4(v) &= \sum_{k=15}^{16} b_k^4 = 0. \end{aligned}$$

Finalmente, el número total de swings es

$$\bar{\eta}(v) = \sum_{i \in N} \eta_i(v) = 12,$$

y el valor de Banzhaf normalizado resulta ser

$$\beta(v) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0 \right).$$

Nótese que, en el ejemplo anterior, el número de coeficientes no nulos de los polinomios $B_i(x)$ está acotado por el número de coeficientes no nulos del polinomio $B(x) = (1 + x^{13})(1 + x^{12})(1 + x^6)(1 + x^2)$. Además, se verifica el siguiente resultado.

Proposición 4.4 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El número de coeficientes no nulos, denotado por c , del polinomio $B(x) = \prod_{j=1}^n (1 + x^{w_j})$, verifica

$$n + 1 \leq c \leq \min(2^n, w(N) + 1).$$

Demostración: En primer lugar, si se considera que los pesos de todos los jugadores son iguales entre sí, $w_1 = \dots = w_n = w$, el número de coeficientes no nulos del polinomio

$$(1 + x^w)^n \tag{1}$$

es siempre menor o igual que el número de coeficientes no nulos del polinomio

$$B(x) = \prod_{j=1}^n (1 + x^{w_j})$$

con los w_1, \dots, w_n cualesquiera.

Por tanto, teniendo en cuenta que el número de coeficientes del polinomio (1) viene dado por el desarrollo de la potencia n -ésima del binomio, resulta que $n + 1 \leq c$.

Para determinar una cota superior de c , obsérvese, por un lado, que

$$B(x) = \prod_{j=1}^n (1 + x^{w_j}) = \sum_{k=0}^{w(N)} b_k x^k,$$

es un polinomio de grado $w(N)$. Luego, de ahí, $c \leq w(N) + 1$. Por otro lado, $c \leq 2^n$ ya que, en el peor caso, cuando todos los exponentes de los términos del polinomio $B(x)$ son distintos, dicho número c coincide con el número de subconjuntos del conjunto $\{w_1, \dots, w_n\}$ que es 2^n .

□

Corolario 4.5 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, para cada $i \in N$, el número de coeficientes no nulos del polinomio $B_i(x) = \prod_{j=1, j \neq i}^n (1 + x^{w_j})$ está acotado por c , siendo c el número de coeficientes no nulos del polinomio $B(x)$.

Ejemplo 4.6 Para el juego de votación considerado en el Ejemplo 4.3, correspondiente a la toma de decisiones en el Ayuntamiento de Sevilla, el polinomio $B(x)$ viene dado por

$$B(x) = (1 + x^{13})(1 + x^{12})(1 + x^6)(1 + x^2),$$

y se verifica que

$$c = 16, \quad \sum_{j=1}^4 w_j + 1 = 34, \quad 2^4 = 16.$$

Ejemplo 4.7 Considérese el juego de votación del Ejemplo 3.73, correspondiente a la toma de decisiones en el Consejo de Gobierno de la Unión Europea. En dicho juego, representado por

$$v = [q, 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2], \quad 61 \leq q \leq 68,$$

el polinomio $B(x)$ resulta ser

$$B(x) = (1 + x^{10})^4(1 + x^8)(1 + x^5)^4(1 + x^4)^2(1 + x^3)^3(1 + x^2),$$

y se verifica que

$$c = 86, \quad \sum_{j=1}^{15} w_j + 1 = 88, \quad 2^{15} = 32768.$$

Ejemplo 4.8 La composición del Congreso de los Diputados de España resultante de las elecciones de 1996 es la siguiente

1: PP, 156 escaños.	7: BNG, 2 escaños.
2: PSOE, 141 escaños.	8: HB, 2 escaños.
3: IU, 21 escaños.	9: ERC, 1 escaño.
4: CiU, 16 escaños.	10: EA, 1 escaño.
5: PNV, 5 escaños.	11: UV, 1 escaño.
6: CC, 4 escaños.	

Conforme a esta distribución de escaños, la toma de decisiones en el Congreso puede modelarse como un juego simple de votación ponderada (N, v) , siendo N el conjunto formado por todos los partidos con representación parlamentaria y

$$v = [q; 156, 141, 21, 16, 5, 4, 2, 2, 1, 1, 1], \quad q = 176.$$

El polinomio $B(x)$ viene dado por

$$B(x) = (1 + x^{156})(1 + x^{141})(1 + x^{21})(1 + x^{16})(1 + x^5)(1 + x^4)(1 + x^2)^2(1 + x)^3,$$

y se verifica que

$$c = 177, \quad \sum_{j=1}^{11} w_j + 1 = 351, \quad 2^{11} = 2048.$$

Proposición 4.9 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. La expansión del polinomio $B(x) = \prod_{j=1}^n (1 + x^{w_j})$ requiere un tiempo $\mathcal{O}(nC)$, siendo $C = \min(2^n, w(N) + 1)$.

Demostración: Sea $f(n)$ el número de operaciones necesarias para expandir el polinomio $B(x) = B^{(n)}(x) = \prod_{j=1}^n (1 + x^{w_j})$. Siguiendo la notación introducida, $f(n-1)$ indica el número de operaciones necesarias para expandir el polinomio $B^{(n-1)}(x) = \prod_{j=1}^{n-1} (1 + x^{w_j})$, y puede establecerse la siguiente relación de recurrencia

$$\mathcal{O}(f(n)) = \begin{cases} \mathcal{O}(1), & \text{si } n = 1 \\ \mathcal{O}(f(n-1) + 3C), & \text{si } n \geq 2, \end{cases}$$

ya que $f(1) = 0$, pues el polinomio ya está expandido, y para $n \geq 2$, en el peor caso, el cálculo del polinomio $B^{(n)}(x) = B^{(n-1)}(x)(1 + x^{w_n})$ requiere un número de productos y de sumas acotado superiormente por $2C$ y C , respectivamente. Esto último es consecuencia de ser C una cota superior del número de coeficientes no nulos del polinomio $B(x)$. Omitiendo la notación $\mathcal{O}(\cdot)$ y expandiendo la recurrencia anterior, se tiene

$$\begin{aligned} f(n) &= f(n-1) + 3C = f(n-2) + 3C + 3C \\ &= f(n-2) + 2(3C) \\ &\vdots \\ &= f(n-k) + k(3C). \end{aligned}$$

Para $k = n-1$, resulta

$$f(n) = f(1) + (n-1)(3C).$$

Es decir, $\mathcal{O}(f(n)) = \mathcal{O}(nC)$. □

Una consecuencia inmediata de esta proposición y del Corolario 4.5 es el siguiente resultado.

Corolario 4.10 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. La expansión del polinomio $B_i(x) = \prod_{j=1, j \neq i}^n (1 + x^{w_j})$, para cada $i \in N$, requiere un tiempo $\mathcal{O}(nc)$, siendo c el número de coeficientes no nulos del polinomio $B(x)$.

A continuación se describe la función *banzhafPowerPlus*, que se utilizará para calcular el índice de poder de Banzhaf normalizado en un juego (N, v) , de votación ponderada, con $v = [q; w_1, \dots, w_n]$.

Función *banzhafPowerPlus*(*pesos*, *q*)

{*pesos*: lista de n enteros; *q*: entero}

```

para  $i$  de 1 hasta  $n$ 
   $lista\_aux \leftarrow borra(\{w_1, \dots, w_n\}, w_i)$ 
   $B_i(x) \leftarrow \prod_{w_j \in lista\_aux} (1 + x^{w_j})$ 
   $\left\{ B_i(x) = \sum_{k=0}^{w(N \setminus i)} b_k^i x^k \right\}$ 
   $\eta_i \leftarrow \sum_{k=q-w_i}^{q-1} b_k^i$ 
fin_para

 $\eta \leftarrow \sum_{i=1}^n \eta_i$ 

para  $i$  de 1 hasta  $n$ 
   $\beta_i \leftarrow \eta_i / \eta$ 
fin_para

```


devolver $\{\beta_1, \dots, \beta_n\}$

En la siguiente proposición se analiza la complejidad temporal de este algoritmo.

Proposición 4.11 *Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El cálculo del índice de Banzhaf normalizado de todos los jugadores con la función *banzhafPowerPlus* requiere un tiempo $\mathcal{O}(n^2c)$, siendo c el número de coeficientes no nulos del polinomio $B(x)$.*

Demostración: Sea $f(n)$ la función que determina el tiempo de ejecución de la función *banzhafPowerPlus*. Considerando las propiedades de la notación $\mathcal{O}(\cdot)$ —en particular, la regla del máximo—, el orden de complejidad de la función $f(n)$ viene dado por la del primer bucle. En consecuencia,

$$\begin{aligned}
 \mathcal{O}(f(n)) &= \mathcal{O}(t(\text{bucle1}) + t(\text{asignación}) + t(\text{bucle2})) \\
 &= \mathcal{O}(\max(t(\text{bucle1}), n, 2n)) \\
 &= \mathcal{O}(t(\text{bucle1})) \\
 &= \mathcal{O}(n(t(\text{lista_aux}) + t(\text{polinomio}) + t(\text{suma}))) \\
 &= \mathcal{O}(nt(\text{lista_aux}) + nt(\text{polinomio}) + nt(\text{suma})) \\
 &= \mathcal{O}(n + n^2c + nc) \\
 &= \mathcal{O}(n^2c).
 \end{aligned}$$

Por tanto, $\mathcal{O}(f(n)) = \mathcal{O}(n^2c)$.

□

4.2.2 Índice de poder de Shapley-Shubik

Ya se ha indicado que Cantor utilizó funciones generatrices para calcular el índice de poder de Shapley-Shubik en juegos simples de votación y, como mencionan Mann y Shapley [41], contribuyó con un resultado que se expone

en la siguiente proposición (véase Lucas [40, pp. 214–216]). Previo al enunciado de la misma, obsérvese que el índice de poder de Shapley-Shubik, para cada jugador $i \in N$, en un juego simple, verifica que

$$\begin{aligned}\Phi_i(v) &= \sum_{\{S \notin \mathcal{W}: S \cup i \in \mathcal{W}\}} \frac{s!(n-s-1)!}{n!} \\ &= \sum_{j=0}^{n-1} \frac{j!(n-j-1)!}{n!} d_j^i,\end{aligned}$$

donde d_j^i representa el número de swings del jugador i en coaliciones de tamaño j ; es decir, el número de coaliciones de tamaño j que pasan a ser ganadoras cuando se incorpora el jugador i .

Si, además, se considera un juego simple de votación ponderada (N, v) , caracterizado por $v = [q; w_1, \dots, w_n]$, puede escribirse, siguiendo el mismo razonamiento establecido en la Proposición 4.1, que, para todo valor de j , $0 \leq j \leq n-1$,

$$d_j^i = \sum_{k=q-w_i}^{k=q-1} a_{kj}^i,$$

siendo a_{kj}^i el número de swings del jugador i en coaliciones formadas con j jugadores, distintos de i , y con un peso coalicional k . Por tanto, dentro de la clase de juegos simples de votación ponderada, el índice de poder de Shapley-Shubik, para cada jugador $i \in N$, viene determinado por

$$\Phi_i(v) = \sum_{j=0}^{n-1} \frac{j!(n-j-1)!}{n!} d_j^i, \quad \text{con} \quad d_j^i = \sum_{k=q-w_i}^{k=q-1} a_{kj}^i.$$

El resultado dado por Cantor, y aludido al principio de la sección, permite calcular los coeficientes $\{a_{kj}^i\}_{k \geq 0, j \geq 0}$, y, con ellos, los números $\{d_j^i\}_{j \geq 0}$ necesarios para determinar el valor de Shapley-Shubik.

Proposición 4.12 (Cantor (1962)) Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Para cualquier $i \in N$, la función generatriz de los números $\{a_{kj}^i\}_{k \geq 0, j \geq 0}$, donde, para cada k y j , a_{kj}^i es el número de coaliciones $S \subseteq N$, formadas por j jugadores, con $i \notin S$ y $w(S) = k$, viene dada por

$$S_i(x, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} z).$$

Demostración: Sea $\{w_1, \dots, w_n\}$ el conjunto de los pesos de todos los jugadores. Se considera la función

$$S(x, z) = (1 + x^{w_1} z) \cdots (1 + x^{w_n} z).$$

Desarrollando los productos se obtiene

$$\begin{aligned} (1 + x^{w_1} z) \cdots (1 + x^{w_n} z) &= \sum_{S \subseteq N} \prod_{i \in S} x^{w_i} z \\ &= \sum_{S \subseteq N} (x^{\sum_{i \in S} w_i} z^{|S|}) \\ &= \sum_{S \subseteq N} x^{w(S)} z^{|S|}, \end{aligned}$$

y, agrupando las potencias, resulta

$$(1 + x^{w_1} z) \cdots (1 + x^{w_n} z) = \sum_{k=0}^{w(N)} \sum_{j=0}^n a_{kj} x^k z^j,$$

donde j indica el tamaño de una coalición S y k el peso de la misma.

El coeficiente a_{kj} indica el número de coaliciones con peso k y tamaño j . Por tanto, la función $S(x, z)$ es una función generatriz para los números $\{a_{kj}\}_{k \geq 0, j \geq 0}$.

Finalmente, es inmediato que, para obtener los números $\{a_{kj}^i\}_{k \geq 0, j \geq 0}$, basta eliminar el factor $(1 + x^{w_i} z)$ en la definición de $S(x, z)$. De ahí, se origina una nueva función $S_i(x, z)$ que es generadora de los mismos.

□

Obsérvese que el conocimiento de los coeficientes $\{a_{kj}^i\}_{k \geq 0, j \geq 0}$, mediante la función generatriz $S_i(x, z)$, posibilita la determinación de los números $\{d_j^i\}_{j \geq 0}$ necesarios para el cálculo del índice de poder de Shapley-Shubik en juegos simples de votación ponderada. Ahora bien, para determinarlos, es preciso notar que estos números d_j^i pueden identificarse con los coeficientes de un polinomio de la forma

$$g_i(z) = \sum_{j=0}^{n-1} d_j^i z^j,$$

y, teniendo en cuenta que $d_j^i = \sum_{k=q-w_i}^{q-1} a_{kj}^i$, resulta

$$g_i(z) = \sum_{j=0}^{n-1} d_j^i z^j = \sum_{j=0}^{n-1} \left[\sum_{k=q-w_i}^{q-1} a_{kj}^i \right] z^j.$$

Como la Proposición 4.12 establece que

$$S_i(x, z) = \sum_{j=0}^{n-1} \left[\sum_{k=0}^{w(N)} a_{kj}^i x^k \right] z^j,$$

se deduce que para determinar los coeficientes de $g_i(z)$ es suficiente, para cada potencia de la variable z en $S_i(x, z)$, seleccionar los coeficientes de los monomios $x^k z^j$ en los que el exponente k de la variable x varíe entre $q - w_i$ y $q - 1$.

A continuación, se ejemplifica lo anteriormente expuesto.

Ejemplo 4.13 *Considérese el juego de votación (N, v) del Ejemplo 4.3, correspondiente a la toma de decisiones en el Ayuntamiento de Sevilla. En este caso,*

$$v = [q; 13, 12, 6, 2], \text{ siendo } q = 17.$$

Para calcular el índice de poder de Shapley-Shubik, con funciones generatrices, se procede del siguiente modo. En primer lugar, se calculan las funciones

$$S_i(x, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} z).$$

$$\begin{aligned} S_1(x, z) &= (1 + x^{12}z)(1 + x^6z)(1 + x^2z) \\ &= 1 + x^2z + x^6z + x^{12}z + x^8z^2 + x^{14}z^2 + x^{18}z^2 + x^{20}z^3, \\ S_2(x, z) &= (1 + x^{13}z)(1 + x^6z)(1 + x^2z) \\ &= 1 + x^2z + x^6z + x^{13}z + x^8z^2 + x^{15}z^2 + x^{19}z^2 + x^{21}z^3, \\ S_3(x, z) &= (1 + x^{13}z)(1 + x^{12}z)(1 + x^2z) \\ &= 1 + x^2z + x^{12}z + x^{13}z + x^{14}z^2 + x^{15}z^2 + x^{25}z^2 + x^{27}z^3, \\ S_4(x, z) &= (1 + x^{13}z)(1 + x^{12}z)(1 + x^6z) \\ &= 1 + x^6z + x^{12}z + x^{13}z + x^{18}z^2 + x^{19}z^2 + x^{25}z^2 + x^{31}z^3. \end{aligned}$$

Los polinomios $g_i(z)$ son los siguientes

$$\begin{aligned} g_1(z) &= \sum_{j=0}^3 \left[\sum_{k=4}^{16} a_{kj}^1 \right] z^j = \sum_{j=0}^3 d_j^1 z^j = 2z + 2z^2, \\ g_2(z) &= \sum_{j=0}^3 \left[\sum_{k=5}^{16} a_{kj}^2 \right] z^j = \sum_{j=0}^3 d_j^2 z^j = 2z + 2z^2, \\ g_3(z) &= \sum_{j=0}^3 \left[\sum_{k=11}^{16} a_{kj}^3 \right] z^j = \sum_{j=0}^3 d_j^3 z^j = 2z + 2z^2, \\ g_4(z) &= \sum_{j=0}^3 \left[\sum_{k=15}^{16} a_{kj}^4 \right] z^j = \sum_{j=0}^3 d_j^4 z^j = 0. \end{aligned}$$

Finalmente, como

$$\Phi_i(v) = \frac{1}{4!} \sum_{j=0}^3 j!(4-j-1)! d_j^i,$$

sustituyendo, resulta

$$\Phi_1(v) = \frac{1}{4!} \sum_{j=0}^3 j!(4-j-1)! d_j^1$$

$$\begin{aligned}
&= \frac{1}{4!}(1!(4-1-1)!d_1^1 + 2!(4-2-1)!d_2^1) \\
&= \frac{1}{4!}(2! \cdot 2 + 2! \cdot 1! \cdot 2) \\
&= \frac{1}{24}(4 + 4) \\
&= \frac{1}{3}.
\end{aligned}$$

Análogamente,

$$\Phi_2(v) = \Phi_3(v) = \frac{1}{3} \quad y \quad \Phi_4(v) = 0.$$

Proposición 4.14 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El número de coeficientes no nulos, denotado por c , del polinomio $S(x, z) = \prod_{j=1}^n (1 + x^{w_j} z)$ verifica

$$n + 1 \leq c \leq \min(2^n, n w(N) + 1).$$

Demostración: Si $w_1 = \dots = w_n = w$, entonces el número de coeficientes no nulos del polinomio

$$(1 + x^w z)^n \tag{1}$$

es siempre menor o igual que el número de coeficientes no nulos del polinomio $S(x, z)$, con w_1, \dots, w_n cualesquiera.

Por tanto, teniendo en cuenta que el número de coeficientes no nulos del polinomio (1) viene dado por el desarrollo de la potencia n -ésima del binomio, resulta que $n + 1 \leq c$.

Para determinar una cota superior basta considerar que

$$S(x, z) = \prod_{j=1}^n (1 + x^{w_j} z) = \sum_{k=0}^{w(N)} \sum_{j=0}^n a_{kj} x^k z^j,$$

es un polinomio de grado $w(N)$ en x , de grado n en z , y en él no hay términos de la forma x^k o z^j . Por tanto, $c \leq n w(N) + 1$. Además, $c \leq 2^n$ ya que, en el peor caso, cuando todos los exponentes de los términos del polinomio $S(x, z)$ son distintos, dicho número c coincide con el número de subconjuntos del conjunto $\{w_1, \dots, w_n\}$.

□

Como consecuencia inmediata, se tiene el siguiente resultado.

Corolario 4.15 *Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, para cada $i \in N$, el número de coeficientes no nulos del polinomio $S_i(x, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} z)$ está acotado por c , siendo c el número de coeficientes no nulos del polinomio $S(x, z)$.*

Al igual que se hizo en la sección correspondiente al índice de Banzhaf, se enuncian resultados referentes a la complejidad temporal de la expansión de los polinomios $S(x, z)$ y $S_i(x, z)$. En ellos, se omiten las demostraciones por ser análogas a las de la Proposición 4.9 y del Corolario 4.10.

Proposición 4.16 *Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. La expansión del polinomio $S(x, z) = \prod_{j=1}^n (1 + x^{w_j} z)$ requiere un tiempo $\mathcal{O}(nC)$, siendo $C = \min(2^n, n w(N) + 1)$.*

Corolario 4.17 *Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. La expansión del polinomio $S_i(x, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} z)$, para cada $i \in N$, requiere un tiempo $\mathcal{O}(nc)$, siendo c el número de coeficientes no nulos del polinomio $S(x, z)$.*

A continuación se describe la función *ssPowerPlus* que se utilizará para el cálculo del índice de poder de Shapley-Shubik en un juego de votación ponderada (N, v) , con $v = [q; w_1, \dots, w_n]$.

Función *ssPowerPlus*(*pesos*, *q*)

{*pesos*: lista de n enteros; *q*: entero}

```

para i de 1 hasta  $n$ 
  lista_aux  $\leftarrow$  borra( $\{w_1, \dots, w_n\}, w_i$ )
  
$$S_i(x, z) \leftarrow \prod_{w_j \in \text{lista\_aux}} (1 + x^{w_j} z)$$

  
$$\left\{ S_i(x, z) = \prod_{l=1, l \neq i}^n (1 + x^{w_l} z) = \sum_{k=0}^{w(N \setminus i)} \sum_{j=0}^{n-1} a_{kj}^i x^k z^j \right\}$$

  
$$g_i(z) \leftarrow \sum_{j=0}^{n-1} d_j^i z^j$$

  
$$\left\{ g_i(z) = \sum_{j=0}^{n-1} \left[ \sum_{k=q-w_i}^{q-1} a_{kj}^i \right] z^j = \sum_{j=0}^{n-1} d_j^i z^j \right\}$$

  
$$\phi_i \leftarrow \sum_{j=0}^{n-1} d_j^i j!(n-j-1)!/n!$$

fin_para

devolver  $\{\phi_1, \dots, \phi_n\}$ 

```

Proposición 4.18 Sea (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El cálculo del índice de poder de Shapley-Shubik de todos los jugadores con la función *ssPowerPlus* requiere un tiempo $\mathcal{O}(n^2 c)$, siendo c el número de coeficientes no nulos del polinomio $S(x, z)$.

Demostración: Sea $f(n)$ la función que determina el tiempo de ejecución de la función *ssPowerPlus*. Entonces, el orden de complejidad de la función $f(n)$ viene dado por la complejidad del bucle

$$\mathcal{O}(f(n)) = \mathcal{O}(t(\text{bucle})) = \mathcal{O}(n(t(\text{asignación1}) + \dots + t(\text{asignación4})))$$

Ahora bien, dentro del bucle, la complejidad de la primera asignación sería

$$\mathcal{O}(t(\text{asignación1})) = \mathcal{O}(t(\text{lista_aux})) = \mathcal{O}(1),$$

ya que la eliminación de cualquier elemento de una lista se realiza en tiempo constante. La complejidad de la segunda asignación,

$$\mathcal{O}(t(\text{asignación2})) = \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(nc),$$

es consecuencia del Corolario 4.17.

Por último, las complejidades de las dos asignaciones siguientes se justifican por ser c una cota superior del número de coeficientes no nulos del polinomio $S_i(x, z)$, y porque el cálculo de los factoriales se realiza en tiempo lineal. En consecuencia,

$$\begin{aligned} \mathcal{O}(t(\text{asignación3})) &= \mathcal{O}(t(\text{suma})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación4})) &= \mathcal{O}(t(\text{suma})) = \mathcal{O}(n^2). \end{aligned}$$

Finalmente, considerando la regla del máximo de la notación $\mathcal{O}(\cdot)$, se sigue que

$$\mathcal{O}(f(n)) = \mathcal{O}(n(t(\text{asignación2}))) = \mathcal{O}(n(nc)) = \mathcal{O}(n^2c).$$

Por tanto, $\mathcal{O}(f(n)) = \mathcal{O}(n^2c)$.

□

4.3 Algoritmos para juegos de votación con doble mayoría

Tal como su nombre indica, en esta sección se tratarán juegos simples de votación con doble mayoría, que siempre pueden interpretarse como una composición de dos juegos simples de votación ponderada.

En el conjunto

$$\Omega^N = \{(N, v), v \text{ simple}\},$$

se considera la siguiente operación interna

$$\wedge : \Omega^N \times \Omega^N \longrightarrow \Omega^N, \quad (v_1, v_2) \longmapsto v = v_1 \wedge v_2,$$

definida, para toda $S \subseteq N$, por

$$v(S) = (v_1 \wedge v_2)(S) = \min \{v_1(S), v_2(S)\}.$$

Si los juegos simples (N, v_1) y (N, v_2) son juegos de votación ponderada, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$, el juego (N, v) , construido de la forma anteriormente descrita, se denomina *juego de votación ponderada con doble mayoría* y, en este caso, se verifica

$$v(S) = (v_1 \wedge v_2)(S) = \begin{cases} 1, & \text{si } w(S) \geq q \text{ y } p(S) \geq p \\ 0, & \text{en otro caso.} \end{cases}$$

El objetivo fundamental de esta sección consistirá en determinar funciones generatrices que permitan calcular, para este tipo de juegos, el índice de Banzhaf normalizado y el índice de Shapley-Shubik.

4.3.1 Índice de poder de Banzhaf normalizado

Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. Es necesario recordar que el cálculo del índice de Banzhaf normalizado exige conocer, para cada jugador i ,

su número de swings. Este número se puede calcular considerando el conjunto de coaliciones ganadoras a las que pertenece el jugador i y, dentro de él, el subconjunto de coaliciones en las que no sea necesaria su presencia para que sean ganadoras.

Proposición 4.19 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. Entonces, el número de swings del jugador i viene dado por*

$$\eta_i(v) = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} b_{kr}^i - \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} b_{kr}^i,$$

donde b_{kr}^i es el número de coaliciones S que no incluyen al jugador i tales que $w(S) = k$ y $p(S) = r$.

Demostración: En primer lugar se considera el conjunto de todas las coaliciones S que no incluyen al jugador i y que tienen pesos $w(S) \geq q - w_i$ y $p(S) \geq p - p_i$. El cardinal de este conjunto viene dado por

$$s_1^i = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} b_{kr}^i.$$

Como $w(S \cup i) \geq q$ y $p(S \cup i) \geq p$, se tiene que s_1^i coincide con el número de coaliciones ganadoras en las que participa el jugador i .

Por otra parte, dentro del conjunto de coaliciones ganadoras en las que participa el jugador i , se considera el subconjunto formado por aquellas coaliciones en las que no es necesaria su presencia para que sean ganadoras. El cardinal de este subconjunto coincide con el del conjunto de todas las coaliciones S que no incluyen al jugador i y que tienen pesos $w(S) \geq q$ y $p(S) \geq p$. Este cardinal viene dado por

$$s_2^i = \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} b_{kr}^i.$$

Para obtener el número de swings del jugador i basta con calcular la diferencia entre s_1^i y s_2^i . Es decir, $\eta_i(v) = s_1^i - s_2^i$.

□

A continuación, se establece una función generatriz para obtener los números $\{b_{kr}^i\}_{k \geq 0, r \geq 0}$.

Proposición 4.20 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. Entonces, para cualquier $i \in N$, la función generatriz de los números $\{b_{kr}^i\}_{k \geq 0, r \geq 0}$, donde, para cada k y r , b_{kr}^i es el número de coaliciones $S \subseteq N$ tales que $i \notin S$, $w(S) = k$, y $p(S) = r$, viene dada por*

$$B_i(x, y) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j}).$$

Demostración: Sean $\{w_1, \dots, w_n\}$ y $\{p_1, \dots, p_n\}$ los conjuntos constituidos por los pesos de los diferentes jugadores en (N, v_1) y (N, v_2) . Considérese la función

$$B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j}).$$

Desarrollando los productos resulta

$$\begin{aligned} B(x, y) &= \sum_{S \subseteq N} \prod_{i \in S} x^{w_i} y^{p_i} \\ &= \sum_{S \subseteq N} (x^{\sum_{i \in S} w_i}) (y^{\sum_{i \in S} p_i}) \\ &= \sum_{S \subseteq N} x^{w(S)} y^{p(S)}, \end{aligned}$$

y, agrupando, se verifica que

$$B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j}) = \sum_{k=0}^{w(N)} \sum_{r=0}^{p(N)} b_{kr} x^k y^r.$$

Por tanto, $B(x, y)$ es una función generatriz para los números $\{b_{kr}\}_{k \geq 0, r \geq 0}$ donde cada b_{kr} indica el número de coaliciones $S \subseteq N$ tales que $w(S) = k$ y $p(S) = r$. Por último, para obtener los números $\{b_{kr}^i\}_{k \geq 0, r \geq 0}$, es suficiente con eliminar el factor $(1 + x^{w_i} y^{p_i})$ en la función $B(x, y)$ dando lugar a la función generatriz $B_i(x, y)$.

□

Ejemplo 4.21 *Supóngase que el Patronato de los Jardines de Almutamid, dependiente del Ayuntamiento de Sevilla, está formado por un representante de cada uno de los grupos políticos municipales. Los acuerdos de dicho organismo deben contar con el respaldo de la mayoría absoluta de sus miembros, los cuales, además, deben representar la mayoría absoluta de los concejales. Se trata, por tanto, de un juego de doble mayoría que puede modelarse del siguiente modo.*

Considérese $N = \{1, 2, 3, 4\}$ el conjunto de jugadores,

Jugador 1: PP	Jugador 3: PA
Jugador 2: PSOE	Jugador 4: IU

y sean (N, v_1) y (N, v_2) , con $v_1 = [17; 13, 12, 6, 2]$ y $v_2 = [3; 1, 1, 1, 1]$. Entonces, la función característica del juego de doble mayoría verifica

$$v(S) = (v_1 \wedge v_2)(S) = \begin{cases} 1, & \text{si } w(S) \geq 17 \text{ y } p(S) \geq 3 \\ 0, & \text{en otro caso.} \end{cases}$$

A continuación se describe el cálculo del índice de Banzhaf mediante funciones generatrices.

En primer lugar, se calculan las funciones $B_i(x, y) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j})$.

$$\begin{aligned} B_1(x, y) &= (1 + x^{12}y)(1 + x^6y)(1 + x^2y) \\ &= 1 + x^2y + x^6y + x^8y^2 + x^{12}y + x^{14}y^2 + x^{18}y^2 + x^{20}y^3, \\ B_2(x, y) &= (1 + x^{13}y)(1 + x^6y)(1 + x^2y) \end{aligned}$$

$$\begin{aligned}
&= 1 + x^2y + x^6y + x^{13}y + x^8y^2 + x^{15}y^2 + x^{19}y^2 + x^{21}y^3, \\
B_3(x, y) &= (1 + x^{13}y)(1 + x^{12}y)(1 + x^2y) \\
&= 1 + x^2y + x^{12}y + x^{13}y + x^{14}y^2 + x^{15}y^2 + x^{25}y^2 + x^{27}y^3, \\
B_4(x, y) &= (1 + x^{13}y)(1 + x^{12}y)(1 + x^6y) \\
&= 1 + x^6y + x^{12}y + x^{13}y + x^{18}y^2 + x^{19}y^2 + x^{25}y^2 + x^{31}y^3.
\end{aligned}$$

Para calcular los swings de cada jugador se evalúa

$$\eta_i(v) = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} b_{kr}^i - \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} b_{kr}^i,$$

y se obtiene

$$\begin{aligned}
\eta_1(v) &= \sum_{k=4}^{20} \sum_{r=2}^3 b_{kr}^1 - \sum_{k=17}^{20} \sum_{r=3}^3 b_{kr}^1 = 4 - 1 = 3, \\
\eta_2(v) &= \sum_{k=5}^{21} \sum_{r=2}^3 b_{kr}^2 - \sum_{k=17}^{21} \sum_{r=3}^3 b_{kr}^2 = 4 - 1 = 3, \\
\eta_3(v) &= \sum_{k=11}^{27} \sum_{r=2}^3 b_{kr}^3 - \sum_{k=17}^{27} \sum_{r=3}^3 b_{kr}^3 = 4 - 1 = 3, \\
\eta_4(v) &= \sum_{k=15}^{31} \sum_{r=2}^3 b_{kr}^4 - \sum_{k=17}^{31} \sum_{r=3}^3 b_{kr}^4 = 4 - 1 = 3.
\end{aligned}$$

Como el número total de swings es $\bar{\eta}(v) = \sum_{i=1}^n \eta_i(v) = 12$, resulta que el valor de Banzhaf normalizado es

$$\beta(v) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4} \right), \text{ donde } \beta_i(v) = \frac{\eta_i(v)}{\bar{\eta}(v)}.$$

Con objeto de facilitar el cómputo de los coeficientes $\{b_{kr}^i\}_{k \geq 0, r \geq 0}$ necesarios para determinar el número de swings, se pueden utilizar tablas para representar los polinomios $B_i(x, y)$. Así, el almacenamiento en una matriz de los coeficientes del polinomio $B_i(x, y)$ sería de la forma que se indica en la

Una vez construida la matriz, para calcular el número de swings del jugador i , se determina, primero, el cardinal del conjunto de coaliciones ganadoras a las que pertenece dicho jugador,

$$s_1^i = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} b_{kr}^i,$$

el cual puede obtenerse sumando los elementos no nulos del bloque constituido por las filas que van desde la $q - w_i + 1$ hasta la $w(N \setminus i) + 1$, y por las columnas comprendidas entre la $p - p_i + 1$ hasta la columna $p(N \setminus i) + 1$, ambas inclusive.

Seguidamente, se determina el cardinal del subconjunto de coaliciones ganadoras, a las que pertenece el jugador i , pero en las que su presencia no es necesaria para que sean ganadoras,

$$s_2^i = \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} b_{kr}^i.$$

Este cardinal puede obtenerse sumando los elementos no nulos del bloque constituido por las filas que van desde la $q + 1$ hasta $w(N \setminus i) + 1$, y por las columnas comprendidas entre la $p + 1$ hasta la columna $p(N \setminus i) + 1$, ambas inclusive.

Por último, el número de swings del jugador i se obtiene evaluando la diferencia

$$\eta_i = s_1^i - s_2^i.$$

Proposición 4.22 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El número de coeficientes no nulos, denotado por c , del polinomio $B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j})$ verifica

$$n + 1 \leq c \leq \min(2^n, w(N)p(N) + 1).$$

Demostración: Una cota inferior de c se obtiene en el caso en que se consideren todos los pesos de todos los jugadores iguales entre sí; es decir, si $w_i = w$ y $p_i = u$, para todo $1 \leq i \leq n$, el número de coeficientes no nulos del polinomio

$$(1 + x^w y^u)^n,$$

que viene determinado por el desarrollo de la potencia n -ésima del binomio, es siempre menor o igual que el número de coeficientes no nulos del polinomio

$$B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j}).$$

Para determinar una cota superior basta considerar que

$$B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j}) = \sum_{k=0}^{w(N)} \sum_{r=0}^{p(N)} b_{kr} x^k y^r,$$

es un polinomio de grado $w(N)$ en x , de grado $p(N)$ en y , y en el que no existen términos de la forma x^k ó y^r . Por tanto, $c \leq w(N)p(N) + 1$. Además, en el peor caso, $c \leq 2^n$ ya que éste se produce cuando todos los exponentes de los términos del polinomio $B(x, y)$ son distintos y, entonces, el número c coincide con el número de subconjuntos del conjunto N de jugadores. \square

A continuación se enuncian resultados que son similares a los ya establecidos para juegos simples de votación ponderada. Las pruebas de los mismos se omiten por ser análogas.

Corolario 4.23 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El número de coeficientes no nulos del polinomio $B_i(x, y) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j})$, para cada $i \in N$, está acotado por c , siendo c el número de coeficientes no nulos del polinomio $B(x, y)$.*

Proposición 4.24 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. La expansión del polinomio $B(x, y) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j})$ requiere un tiempo $\mathcal{O}(nC)$, siendo $C = \min(2^n, w(N)p(N) + 1)$.

Corolario 4.25 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. Para cada $i \in N$, la expansión del polinomio $B_i(x, y) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j})$ requiere un tiempo $\mathcal{O}(nc)$, siendo c el número de coeficientes no nulos del polinomio $B(x, y)$.

Por último, se describe la función *banzhafTwoPower* que se utilizará para calcular el índice de Banzhaf normalizado de todos los jugadores en un juego de votación ponderada de doble mayoría. Además, se estudia su complejidad temporal.

Función *banzhafTwoPower*(*pesos1*, *pesos2*, *q*, *p*)

{ *pesos1*: lista de n enteros; *pesos2*: lista de n enteros; *q*: entero; *p*: entero }

para i de 1 hasta n

lista_aux1 \leftarrow borra($\{w_1, \dots, w_n\}, w_i$)

lista_aux2 \leftarrow borra($\{p_1, \dots, p_n\}, p_i$)

$B_i(x, y) \leftarrow \prod_{\substack{w_j \in \text{lista_aux1} \\ p_j \in \text{lista_aux2}}}^n (1 + x^{w_j} y^{p_j})$

$$\left\{ B_i(x, y) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j}) = \sum_{k=0}^{w(N \setminus i)} \sum_{r=0}^{p(N \setminus i)} b_{kr}^i x^k y^r \right\}$$

coefi \leftarrow Coeficientes de $B_i(x, y)$

{*coefi* es una matriz de orden $(w(N \setminus i) + 1) \times (p(N \setminus i) + 1)$ }

```

 $s_1^i \leftarrow \text{Suma}[\text{coef}i[k, r], \text{ con } k \text{ desde } q - w_i + 1 \text{ hasta } w(N \setminus i) + 1, \text{ y con } r$ 
 $\text{ desde } p - p_i + 1 \text{ hasta } p(N \setminus i) + 1]$ 

$$\left\{ s_1^i = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} b_{kr}^i \right\}$$

 $s_2^i \leftarrow \text{Suma}[\text{coef}i[k, r], \text{ con } k \text{ desde } q + 1 \text{ hasta } w(N \setminus i) + 1, \text{ y con } r$ 
 $\text{ desde } p + 1 \text{ hasta } p(N \setminus i) + 1]$ 

$$\left\{ s_2^i = \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} b_{kr}^i \right\}$$

 $\eta_i \leftarrow s_1^i - s_2^i$ 
fin para

$$\eta \leftarrow \sum_{i=1}^n \eta_i$$

para  $i$  de 1 hasta  $n$ 
 $\beta_i \leftarrow \eta_i / \eta$ 
fin para
devolver  $\{\beta_1, \dots, \beta_n\}$ 

```

Proposición 4.26 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El cálculo del índice de Banzhaf normalizado de todos los jugadores, con la función *banzhafTwo-Power*, requiere un tiempo $\mathcal{O}(n^2c)$, siendo c el número de coeficientes no nulos del polinomio $B(x, y)$.*

Demostración: En esta situación, el orden de complejidad de la función $f(n)$, que determina el tiempo de ejecución de la función *banzhafTwoPower*,

viene dado por

$$\begin{aligned}\mathcal{O}(f(n)) &= \mathcal{O}(t(\text{bucle1}) + t(\text{asignación}) + t(\text{bucle2})) \\ &= \mathcal{O}(\max(t(\text{bucle1}), n, 2n)) \\ &= \mathcal{O}(t(\text{bucle1})).\end{aligned}$$

Ahora bien,

$$\mathcal{O}(t(\text{bucle1})) = \mathcal{O}(n(t(\text{asignación1}) + \dots + t(\text{asignación7})))$$

y, teniendo en cuenta que

$$\begin{aligned}\mathcal{O}(t(\text{asignación1})) &= \mathcal{O}(t(\text{lista_aux1})) = \mathcal{O}(1), \\ \mathcal{O}(t(\text{asignación2})) &= \mathcal{O}(t(\text{lista_aux2})) = \mathcal{O}(1), \\ \mathcal{O}(t(\text{asignación3})) &= \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(nc), \\ \mathcal{O}(t(\text{asignación4})) &= \mathcal{O}(t(\text{coefi})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación5})) &= \mathcal{O}(t(\text{suma1})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación6})) &= \mathcal{O}(t(\text{suma2})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación7})) &= \mathcal{O}(t(\text{diferencia})) = \mathcal{O}(1),\end{aligned}$$

resulta ser

$$\mathcal{O}(f(n)) = \mathcal{O}(t(\text{bucle1})) = \mathcal{O}(n t(\text{asignación3})).$$

Por tanto, finalmente,

$$\mathcal{O}(f(n)) = \mathcal{O}(n t(\text{asignación3})) = \mathcal{O}(n^2 c).$$

□

4.3.2 Índice de poder de Shapley-Shubik

Al igual que sucede con el índice de Banzhaf en juegos de doble mayoría, también es posible obtener, usando funciones generatrices, un resultado similar para el cálculo del índice de poder de Shapley-Shubik.

Ahora bien, antes de establecer los oportunos resultados, recuérdese que, en la sección 4.2.2 se indicó que el índice de Shapley-Shubik en juegos simples venía determinado, para cada jugador $i \in N$, por

$$\Phi_i(v) = \sum_{j=0}^{n-1} \frac{j!(n-j-1)!}{n!} d_j^i$$

donde cada d_j^i representa el número de swings del jugador i en coaliciones de tamaño j .

Si, en la situación que aquí se estudia, resulta ser el juego $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$, entonces, según los razonamientos establecidos al inicio de la sección anterior, se tiene que

$$d_j^i = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} a_{krj}^i - \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} a_{krj}^i,$$

donde a_{krj}^i es el número de coaliciones S que no incluyen al jugador i y que tienen pesos $w(S) = k$ y $p(S) = r$.

Proposición 4.27 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. Para cualquier $i \in N$, la función generatriz de los números $\{a_{krj}^i\}_{k \geq 0, r \geq 0, j \geq 0}$ donde, para cada k, r y j , a_{krj}^i es el número de coaliciones S de j jugadores con $w(S) = k$, $p(S) = r$ y tales que $i \notin S$, viene dada por*

$$S_i(x, y, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j} z).$$

Demostración: Sean $\{w_1, \dots, w_n\}$ y $\{p_1, \dots, p_n\}$ los conjuntos de los pesos correspondientes a todos los jugadores en los juegos (N, v_1) y (N, v_2) . Considérese la siguiente función

$$S(x, y, z) = (1 + x^{w_1} y^{p_1} z) \cdots (1 + x^{w_n} y^{p_n} z).$$

Desarrollando los productos

$$\begin{aligned}
S(x, y, z) &= \sum_{S \subseteq N} z^{|S|} \prod_{i \in S} x^{w_i} y^{p_i} \\
&= \sum_{S \subseteq N} z^{|S|} \left(x^{\sum_{i \in S} w_i} \right) \left(y^{\sum_{i \in S} p_i} \right) \\
&= \sum_{S \subseteq N} x^{w(S)} y^{p(S)} z^{|S|} \\
&= \sum_{k=0}^{w(N)} \sum_{r=0}^{p(N)} \sum_{j=0}^n a_{krj} x^k y^r z^j,
\end{aligned}$$

donde cada coeficiente a_{krj} es el número de coaliciones $S \subseteq N$, tales que $|S| = j$, $w(S) = k$ y $p(S) = r$. Obviamente, para obtener los números $\{a_{krj}^i\}_{k \geq 0, r \geq 0, j \geq 0}$, es suficiente eliminar el factor $(1 + x^{w_i} y^{p_i} z)$ en la función $S(x, y, z)$. Ello da lugar a la función $S_i(x, y, z)$ que, por tanto, es generatriz de los mismos.

□

Obsérvese que el conocimiento de los coeficientes $\{a_{krj}^i\}_{k \geq 0, r \geq 0, j \geq 0}$, mediante la función $S_i(x, y, z)$ posibilita la determinación de los números $\{d_j^i\}_{j \geq 0}$ necesarios para calcular el índice de Shapley-Shubik. Ahora bien, para ello, se utilizará una técnica similar a la empleada en los juegos simples de votación ponderada de la sección 4.2.2, y a la descrita en la sección 4.3.1 para almacenar los coeficientes de los polinomios $B_i(x, y)$ en el cálculo de índice de Banzhaf en juegos de votación de doble mayoría.

Así, los números $\{d_j^i\}_{j \geq 0}$ pueden identificarse con los coeficientes de un polinomio de la forma

$$g_i(z) = \sum_{j=0}^{n-1} d_j^i z^j$$

y, teniendo en cuenta el hecho de ser

$$d_j^i = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} a_{krj}^i - \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} a_{krj}^i,$$

Posteriormente, procediendo de modo análogo al cálculo de índice de Banzhaf, se determina, primero, un polinomio en z cuyos coeficientes representan, clasificados según los diferentes cardinales de las coaliciones, las coaliciones ganadoras a las que pertenece el jugador i ,

$$s_1^i(z) = \sum_{k=q-w_i}^{w(N\setminus i)} \sum_{r=p-p_i}^{p(N\setminus i)} \sum_{j=0}^{n-1} a_{krj}^i z^j.$$

En segundo lugar, se determina también un polinomio en z cuyos coeficientes representan, clasificados según los diferentes cardinales de las coaliciones, el número de coaliciones ganadoras a las que pertenece el jugador i pero en las que su presencia no es necesaria para que sean ganadoras,

$$s_2^i(z) = \sum_{k=q}^{w(N\setminus i)} \sum_{r=p}^{w(N\setminus i)} \sum_{j=0}^{n-1} a_{krj}^i z^j.$$

Estos polinomios $s_1^i(z)$ y $s_2^i(z)$ se obtienen, respectivamente, sumando los elementos no nulos del bloque constituido por las filas que van desde la $(q - w_i)$ -ésima a la última y desde la $(p - p_i)$ -ésima columna a la última, y sumando los elementos no nulos del bloque formado por las $w(N\setminus i) - q$ últimas filas y las $p(N\setminus i) - p$ últimas columnas.

Finalmente,

$$g_i(z) = \sum_{j=0}^{n-1} d_j^i z^j = s_1^i(z) - s_2^i(z).$$

A continuación se ejemplifican los razonamientos desarrollados en los párrafos anteriores. Posteriormente, se estudia la complejidad del algoritmo para el cálculo de los coeficientes $\{d_j^i\}_{j \geq 0}$.

Ejemplo 4.28 *Considérese el Ejemplo 4.21 correspondiente al Patronato de los Jardines de Almutamid. Teniendo en cuenta que, en este caso, los pesos correspondientes a todos los jugadores en los juegos (N, v_1) y (N, v_2) son*

$\{13, 12, 6, 2\}$ y $\{1, 1, 1, 1\}$, la función generatriz correspondiente al jugador 1 viene dada por

$$\begin{aligned} S_1(x, y, z) &= (1 + x^{12}yz)(1 + x^6yz)(1 + x^2yz) \\ &= 1 + x^2yz + x^6yz + x^8y^2z^2 + x^{12}yz + x^{14}y^2z^2 + x^{18}y^2z^2 + x^{20}y^3z^3, \end{aligned}$$

y la representación matricial del polinomio es

$$\begin{array}{c} 1 \\ x \\ x^2 \\ x^3 \\ x^4 \\ x^5 \\ x^6 \\ x^7 \\ x^8 \\ x^9 \\ x^{10} \\ x^{11} \\ x^{12} \\ x^{13} \\ x^{14} \\ x^{15} \\ x^{16} \\ x^{17} \\ x^{18} \\ x^{19} \\ x^{20} \end{array} \begin{pmatrix} 1 & y & y^2 & y^3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & z & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & z & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & z & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & z^2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & z^2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z^3 \end{pmatrix}$$

FIGURA 4.4

Nótese que, tal y como se ha razonado previamente, el elemento que ocupa la posición $(k+1, r+1)$ en esta matriz es un polinomio de la forma $\sum_{j=0}^{n-1} a_{krj}^1 z^j$.

Proposición 4.29 *Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El número de coeficientes no nulos, denotado por c , del polinomio $S(x, y, z) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j} z)$ verifica*

$$n + 1 \leq c \leq \min(2^n, n w(N) p(N) + 1).$$

Demostración: Una cota inferior se obtiene en el caso en que se consideren los pesos de todos los jugadores iguales entre sí; es decir, si $w_i = w$ y $p_i = u$, para todo $1 \leq i \leq n$, el número de coeficientes no nulos del polinomio

$$(1 + x^w y^u z)^n,$$

que viene determinado por el desarrollo de la potencia n -ésima del binomio, es siempre menor o igual que el número de coeficientes no nulos del polinomio

$$S(x, y, z) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j} z).$$

Por tanto, $c \geq n + 1$.

Para determinar una cota superior basta considerar que

$$S(x, y, z) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j} z) = \sum_{k=0}^{w(N)} \sum_{r=0}^{p(N)} \sum_{j=0}^n a_{krj} x^k y^r z^j,$$

es un polinomio de grado $w(N)$ en x , de grado $p(N)$ en y , de grado n en z , y en el que no hay términos de la forma x^k , y^r ó z^j . Por tanto,

$$c \leq n w(N) p(N) + 1.$$

Además, en el peor caso, $c \leq 2^n$ ya que éste se produce cuando todos los exponentes de los términos del polinomio $S(x, y, z)$ son distintos y, entonces,

dicho número c coincide con el número de subconjuntos del conjunto N de jugadores.

□

De la misma forma que se hizo en la sección anterior, se indican resultados similares a los enunciados para juegos de votación de una mayoría. Las pruebas de los mismos se omiten por ser análogas.

Corolario 4.30 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El número de coeficientes no nulos del polinomio $S_i(x, y, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j} z)$, para cada $i \in N$, está acotado por c , siendo c el número de coeficientes no nulos del polinomio $S(x, y, z)$.

Proposición 4.31 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. La expansión del polinomio $S(x, y, z) = \prod_{j=1}^n (1 + x^{w_j} y^{p_j} z)$ requiere un tiempo $\mathcal{O}(nC)$, siendo, $C = \min(2^n, n w(N) p(N) + 1)$.

Corolario 4.32 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. La expansión del polinomio $S_i(x, y, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j} z)$, para cada $i \in N$, requiere un tiempo $\mathcal{O}(nc)$, siendo c el número de coeficientes no nulos del polinomio $S(x, y, z)$.

A continuación se describe la función *ssTwoPower* que se utilizará para calcular el índice de poder de Shapley-Shubik de todos los jugadores en un juego (N, v) de votación con doble mayoría, siendo la función característica $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$.

Función $ssTwoPower(pesos1, pesos2, q, p)$

{ $pesos1$: lista de n enteros; $pesos2$: lista de n enteros; q, p : enteros }

para i de 1 hasta n

$lista_aux1 \leftarrow borra(\{w_1, \dots, w_n\}, w_i)$

$lista_aux2 \leftarrow borra(\{p_1, \dots, p_n\}, p_i)$

$S_i(x, y, z) \leftarrow \prod_{\substack{w_j \in lista_aux1 \\ p_j \in lista_aux2}}^n (1 + x^{w_j} y^{p_j} z)$

$$\left\{ S_i(x, y, z) = \prod_{j=1, j \neq i}^n (1 + x^{w_j} y^{p_j} z) = \sum_{k=0}^{w(N \setminus i)} \sum_{r=0}^{p(N \setminus i)} \sum_{j=0}^{n-1} a_{krj}^i x^k y^r z^j \right\}$$

$coefi \leftarrow$ Coeficientes de $S_i(x, y, z)$

{ $coefi$ es una matriz que almacena los coeficientes del polinomio,

$$coefi(k+1, r+1) = \sum_{j=0}^{n-1} a_{krj}^i z^j \}$$

$s_1^i(z) \leftarrow$ Suma[$coefi[k, r]$, con k desde $q - w_i + 1$ hasta $w(N \setminus i) + 1$,
y con r desde $p - p_i + 1$ hasta $p(N \setminus i) + 1$]

$$\left\{ s_1^i(z) = \sum_{k=q-w_i}^{w(N \setminus i)} \sum_{r=p-p_i}^{p(N \setminus i)} \sum_{j=0}^{n-1} a_{krj}^i z^j \right\}$$

$s_2^i(z) \leftarrow$ Suma[$coefi[k, r]$, con k desde $q + 1$ hasta $d_w + 1$,
y con r desde $p + 1$ hasta $d_p + 1$]

$$\left\{ s_2^i(z) = \sum_{k=q}^{w(N \setminus i)} \sum_{r=p}^{p(N \setminus i)} \sum_{j=0}^{n-1} a_{krj}^i z^j \right\}$$

$g_i(z) \leftarrow s_1^i(z) - s_2^i(z)$

$$\left\{ g_i(z) = \sum_{j=0}^{n-1} d_j^i z^j \right\}$$

$\phi_i \leftarrow \sum_{j=0}^{n-1} d_j^i j! (n - j - 1)! / n!$

fin para

devolver $\{\phi_1, \dots, \phi_n\}$

Proposición 4.33 Sea (N, v) un juego de votación con doble mayoría, siendo $v = v_1 \wedge v_2$, con $v_1 = [q; w_1, \dots, w_n]$ y $v_2 = [p; p_1, \dots, p_n]$. El cálculo del índice de Shapley-Shubik de todos los jugadores con la función *ssTwoPower* requiere un tiempo $\mathcal{O}(n^2c)$, siendo c el número de coeficientes no nulos del polinomio $S(x, y, z)$.

Demostración: Sea $f(n)$ la función que determina el tiempo de ejecución de la función *ssTwoPower*. Se verifica que

$$\mathcal{O}(f(n)) = \mathcal{O}(t(\text{bucle})) = \mathcal{O}(n(t(\text{asignación1}) + \dots + t(\text{asignación8}))).$$

Ahora bien,

$$\begin{aligned} \mathcal{O}(t(\text{asignación1})) &= \mathcal{O}(t(\text{lista_aux1})) = \mathcal{O}(1), \\ \mathcal{O}(t(\text{asignación2})) &= \mathcal{O}(t(\text{lista_aux2})) = \mathcal{O}(1), \\ \mathcal{O}(t(\text{asignación3})) &= \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(nc), \\ \mathcal{O}(t(\text{asignación4})) &= \mathcal{O}(t(\text{coefi})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación5})) &= \mathcal{O}(t(\text{suma1})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación6})) &= \mathcal{O}(t(\text{suma2})) = \mathcal{O}(c), \\ \mathcal{O}(t(\text{asignación7})) &= \mathcal{O}(t(\text{diferencia})) = \mathcal{O}(n), \\ \mathcal{O}(t(\text{asignación8})) &= \mathcal{O}(t(\text{suma})) = \mathcal{O}(n^2). \end{aligned}$$

Por tanto,

$$\mathcal{O}(f(n)) = \mathcal{O}(n(t(\text{asignación3}))) = \mathcal{O}(n(nc)) = \mathcal{O}(n^2c).$$

□

4.4 Algoritmos con MATHEMATICA

En esta sección se describen algunos ejemplos en los que se calculan los índices de poder de Banzhaf y de Shapley-Shubik en diversos juegos de votación.

Para ello, se emplean los algoritmos anteriormente descritos implementados con MATHEMATICA.

Ejemplo 4.34 *Sea el juego de votación del Ejemplo 4.7, correspondiente a la toma de decisiones en el Consejo de Gobierno de la Unión Europea, y cuya función característica se representa por*

$$v = [62; 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2].$$

Para este juego, se describe la implementación con MATHEMATICA del cálculo de los índices de Banzhaf normalizado y de Shapley-Shubik mediante funciones generatrices. Este procedimiento es similar al presentado por Tannenbaum (1997) [57] aunque con algunas modificaciones.

In[1] :=

```
<<DiscreteMath'Combinatorica'
```

In[2] :=

```
banzhafG[weights_List]:=Times @@ (1+x^weights)
```

In[3] :=

```
banzhafIndexPlus[i_, weights_List, q_Integer]:=
Module[{delw, sw, poly, coefi}, delw>Delete[weights,i];
sw=Apply[Plus,delw]; poly=banzhafG[delw];
coefi=CoefficientList[poly,x];
Apply[Plus,coefi[[Range[Max[1,q-weights[[i]]+1],Min[q,sw]]]]]]
```

In[4] :=

```
banzhafPowerPlus[weights_List, q_]:=#/(Plus @@ #)& @
Table[banzhafIndexPlus[i, weights, q],{i,Length[weights]}
```

In[5] :=

```
votosUE={10,10,10,10,8,5,5,5,5,4,4,3,3,3,2}
```

In[6] :=

```
Timing[banzhafPowerPlus[votosUE,62]]//N
```

```
Out[6] =
```

```
{0.726 Second, {0.111621, 0.111621, 0.111621, 0.111621,
0.0924238, 0.0587383, 0.0587383, 0.0587383, 0.0587383,
0.047872, 0.047872, 0.0359191, 0.0359191, 0.0359191,
0.0226381}}
```

```
In[7] :=
```

```
ssG[pesos_List]:=Times @@ (1+ z x^weights)
```

```
In[8] :=
```

```
ssPowerPlus[weights_List,q_Integer]:=
Module[{n=Length[weights], delw, sw, poly, g, coefi},
Table[delw=Delete[weights,i]; sw=Apply[Plus,delw]+1;
poly=ssG[delw]; coefi=CoefficientList[poly,x];
g=Apply[Plus,coefi[[
Range[Max[1,q-weights[[i]]+1],Min[q,sw]]]]];
Sum[Coefficient[g,z,j] j! (n-j-1)!,{j,n-1}},{i,n}]/n!]
```

```
In[9] :=
```

```
Timing[ssPowerPlus[votosUE,62]]//N
```

```
Out[9] =
```

```
{3.778 Second, {0.116667, 0.116667, 0.116667, 0.116667,
0.0954601, 0.0551754, 0.0551754, 0.0551754, 0.0551754, 0.0453602,
0.0453602, 0.0352536, 0.0352536, 0.0352536, 0.0206904}}
```

Ejemplo 4.35 *Supóngase que, para adoptar algunas decisiones en el Consejo de Gobierno de la Unión Europea, se considera un juego de votación (N, v) de doble mayoría, siendo $v = v_1 \wedge v_2$, con*

$$v_1 = [62; 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2],$$

$$v_2 = [60; 22, 16, 16, 15, 11, 4, 3, 3, 3, 2, 2, 1, 1, 1, 0].$$

Obsérvese que el juego (N, v_1) es el juego estudiado en el ejemplo anterior. Por otro lado se hace notar que los pesos correspondientes al juego (N, v_2) vienen determinados por el porcentaje, redondeado, de población de cada uno de los países de la Unión Europea.

El cálculo con MATHEMATICA de los índices de Banzhaf y de Shapley-Shubik, usando las funciones introducidas en este capítulo, es como sigue.

In[1] :=

```
<<DiscreteMath'Combinatorica'
```

In[2] :=

```
banzhafTwoG[weights_List, pop_List] := Times @@ (1 + x^weights
y^pop)
```

In[3] :=

```
banzhafTwoIndex[i_ , weights_List, pop_List, q_Integer, p_Integer] :=
Module[{poly, coefi, delwe, delpo, s1, s2, sw, sp},
delwe = Delete[weights, i]; delpo = Delete[pop, i];
poly = banzhafTwoG[delwe, delpo];
sw = Apply[Plus, delwe] + 1; sp = Apply[Plus, delpo] + 1;
coefi = CoefficientList[poly, {x, y}] /. {} -> Table[0, {sp}];
s1 = Apply[Plus, Flatten[coefi[[
Range[Max[1, q - weights[[i]] + 1], sw], Range[Max[1, p - pop[[i]] + 1], sp]]]];
s2 = If[((q + 1) > sw) || ((p + 1) > sp), 0, Apply[Plus,
Flatten[coefi[[Range[q + 1, sw], Range[p + 1, sp]]]]]];
s1 - s2]
```

In[4] :=

```
banzhafTwoPower[weights_List, pop_List, q_ , p_] := # / (Plus @@ #) &
@ Table[banzhafTwoIndex[i, weights, pop, q, p], {i, Length[weights]}
```

In[5] :=

```
votosUE = {10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2};
```

```
In[6] :=
  popUE{22, 16, 16, 15, 11, 4, 3, 3, 3, 2, 2, 1, 1, 1, 0};
```

```
In[7] :=
  Timing[banzhafTwoPower[votosUE,popUE,62,60]]//N
```

```
Out[7] =
  {5.99 Second, {0.112715, 0.112109, 0.112109, 0.111745,
  0.092332, 0.0588449, 0.0587236, 0.0587236, 0.0587236, 0.0475613,
  0.0475613, 0.0355496, 0.0355496, 0.0355496, 0.0222033}}
```

```
In[8] :=
  ssTwoG[weights_List,pop_List]:=
  Times @@ (1+x^weights y^pop z)
```

```
In[9] :=
  ssTwoPower[weights_List,pop_List,q_Integer,p_Integer]:=
  Module[{n=Length[weights],poly,delw,delp,sw,sp,s1,s2,g,coefi},
  Table[delw>Delete[weights,i]; delp>Delete[pop,i];
  poly=ssTwoG[delw,delp];
  sw=Apply[Plus,delw]+1; sp=Apply[Plus,delp]+1;
  coefi=CoefficientList[poly,{x,y}]/.{x,y}->Table[0,{sp}];
  s1=Apply[Plus,Flatten[coefi[
  [Range[Max[1,q-weights[[i]]+1],sw],
  Range[Max[1,p-pop[[i]]+1],sp]]]];
  s2=If[((q+1)>sw) || ((p+1)>sp),0,Apply[
  Plus,Flatten[coefi[[Range[q+1,sw],Range[p+1,sp]]]]];
  g=s1-s2;
  Sum[Coefficient[g,z,j] j! (n-j-1)!,{j,0,n-1}]/n!,{i,n}]
```

```
In[9] :=
  Timing[ssTwoPower[votosUE,popUE,62,60]]//N
```

```
Out[9] =
```

*{11.21 Second, {0.12326, 0.118681, 0.118681, 0.115934,
0.0938117, 0.0562743, 0.0553585, 0.0553585, 0.0553585, 0.0437118,
0.0437118, 0.0336053, 0.0336053, 0.0336053, 0.0190421}}*

Por último, se quiere resaltar que el número de coeficientes no nulos de los polinomios $B(x, y)$ y $S(x, y, z)$ se calculan, con MATHEMATICA, del siguiente modo

In[10] :=
`Length[banzhafTwoG[votosUE,popUE]//Expand]`

Out[10] =
1644

In[11] :=
`Length[ssTwoG[votosUE,popUE]//Expand]`

Out[11] =
2206

Capítulo 5

Funciones generatrices para juegos restringidos

Las funciones generatrices ya han sido introducidas en el capítulo anterior, con el objetivo de obtener e implementar nuevos algoritmos que permitan calcular índices de poder en juegos simples de votación ponderada y en contextos de cooperación total entre todos los jugadores.

Ahora, se considerarán algunas situaciones de cooperación parcial y, con el mismo objetivo del capítulo anterior, se estudiará la complejidad del cálculo del valor de Myerson generalizado en juegos simples de votación ponderada.

En la primera sección, el trabajo se centra en la utilización de funciones generatrices cuando el valor de Myerson generalizado se quiere determinar mediante los dividendos del juego restringido. En la segunda sección, se introduce una nueva fórmula explícita para el cálculo del valor de Myerson en juegos simples de votación ponderada y en situaciones de comunicación (N, v, G) en las que G es un grafo estrella. Esta nueva fórmula permitirá la creación e implementación de un algoritmo que incorpora la utilización de funciones generatrices.

5.1 Valor de Myerson generalizado mediante funciones generatrices

En el tercer capítulo se ha descrito un algoritmo, dado por Bilbao y López [12], para calcular los dividendos de Harsanyi en juegos restringidos por geometrías convexas de partición. Dicho resultado, que supone una generalización del obtenido por Owen [44] para árboles, permite calcular los dividendos del juego restringido en términos de la función característica del juego (N, v) . Además, mediante esta expresión de los dividendos se obtienen simplificaciones en el cálculo del valor de Myerson generalizado.

Así, al considerar la terna (N, v, \mathcal{L}) donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego cooperativo, los dividendos del juego restringido $(N, v^{\mathcal{L}})$ venían dados, para cualquier coalición $S \in \mathcal{L}$, por

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{\{T: S \setminus \text{ex}(S) \subseteq T \subseteq S\}} (-1)^{|S|-|T|} v(T),$$

y, como consecuencia, el valor de Myerson generalizado, para cada jugador $i \in N$, se obtenía evaluando la expresión

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{S \in \mathcal{L}: i \in S\}} \frac{1}{|S|} \sum_{T \in [S^-, S]} (-1)^{|S|-|T|} v(T),$$

donde $[S^-, S] = \{T : S \setminus \text{ex}(S) \subseteq T \subseteq S\}$.

En esta sección, haciendo uso de las igualdades anteriores y de las funciones generatrices, se obtendrá una expresión del valor de Myerson generalizado en un juego de votación ponderada (N, v) , con $v = [q; w_1, \dots, w_n]$, cuando la cooperación está restringida por una geometría convexa de partición (N, \mathcal{L}) . Para ello, se establece, en primer lugar, una fórmula para el cálculo de los dividendos del juego restringido $(N, v^{\mathcal{L}})$.

Proposición 5.1 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Si $S \in \mathcal{L}$, entonces*

$$\Delta_{v\mathcal{L}}(S) = \sum_{r=0}^{|ex(S)|} (-1)^r a_r^S,$$

donde cada a_r^S es el número de coaliciones $T \in [S^-, S]$, con $|T| = |S| - r$ y peso $w(T) \geq q$.

Demostración: Como el conjunto $[S^-, S] = \{T : S \setminus ex(S) \subseteq T \subseteq S\}$ es isomorfo al álgebra de Boole $2^{ex(S)}$, se tiene que toda coalición $T \in [S^-, S]$ puede expresarse como $T = S \setminus R$, con $R \in 2^{ex(S)}$ y, por tanto,

$$\begin{aligned} \Delta_{v\mathcal{L}}(S) &= \sum_{\{T: S \setminus ex(S) \subseteq T \subseteq S\}} (-1)^{|S|-|T|} v(T) \\ &= \sum_{\{R: \emptyset \subseteq R \subseteq ex(S)\}} (-1)^{|R|} v(S \setminus R) \\ &= v(S) - \sum_{i \in ex(S)} v(S \setminus i) \\ &\quad + \sum_{\{i,j\} \subseteq ex(S)} v(S \setminus \{i,j\}) + \dots + (-1)^{|ex(S)|} v(S \setminus ex(S)). \end{aligned}$$

Al verificarse que $v(S \setminus R) = 1$ si, y sólo si, $w(S \setminus R) \geq q$, resulta

$$\Delta_{v\mathcal{L}}(S) = \sum_{r=0}^{|ex(S)|} (-1)^r a_r^S,$$

siendo cada a_r^S el número de coaliciones del intervalo $[S^-, S]$, de tamaño $|S| - r$ y con peso mayor o igual que la cuota q .

□

A continuación, con el objetivo de determinar, para cada coalición factible $S \in \mathcal{L}$, los números $\{a_r^S\}_{r \geq 0}$ de la proposición anterior, se establece, previamente, la expresión de la función generatriz de los números $\{b_{kj}^S\}_{k \geq 0, j \geq 0}$,

donde cada b_{kj}^S representa el número de coaliciones del intervalo $[S^-, S]$ de peso k y tamaño j .

Proposición 5.2 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, dada una coalición $S \in \mathcal{L}$, la función generatriz de los números $\{b_{kj}^S\}_{k \geq 0, j \geq 0}$ donde cada b_{kj}^S representa el número de coaliciones $T \in [S^-, S]$ tales que $|T| = j$ y $w(T) = k$, viene dada por el polinomio*

$$P_S(x, z) = x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \prod_{i \in ex(S)} (1 + x^{w_i} z).$$

Demostración: El intervalo $[S^-, S]$ es isomorfo a $2^{ex(S)}$. Por tanto, se tiene

$$\begin{aligned} [S^-, S] &= \{T : S \setminus ex(S) \subseteq T \subseteq S\} \\ &= \{T : T = (S \setminus ex(S)) \cup R, \quad R \in 2^{ex(S)}\}. \end{aligned}$$

También, obsérvese que cada coeficiente del producto $\prod_{i \in ex(S)} (1 + x^{w_i} z)$ determina el número de coaliciones de $2^{ex(S)}$ de un tamaño y peso determinados. Por otra parte, teniendo en cuenta que los jugadores no extremales de S pertenecen a todas las coaliciones del intervalo $[S^-, S]$, y que cada coalición de dicho intervalo puede obtenerse como la unión de la coalición $S \setminus ex(S)$ con una coalición $R \in 2^{ex(S)}$, resulta que al multiplicar el producto anterior por el monomio $x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|}$ se obtiene

$$\begin{aligned} P_S(x, z) &= x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \prod_{i \in ex(S)} (1 + x^{w_i} z) \\ &= x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \sum_{k=0}^{w(ex(S))} \sum_{j=0}^{|ex(S)|} b_{kj}^S x^k z^j \\ &= \sum_{k=w(S \setminus ex(S))}^{w(S)} \sum_{j=|S \setminus ex(S)|}^{|S|} b_{kj}^S x^k z^j, \end{aligned}$$

donde, finalmente, cada b_{kj}^S representa el número de coaliciones $T \in [S^-, S]$ con un tamaño y peso determinados.

□

Este resultado se ejemplifica a continuación.

Ejemplo 5.3 Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es la geometría convexa de partición correspondiente al grafo de cooperación de la Figura 5.1, y (N, v) un juego de votación ponderada, con $v = [176; 156, 4, 16, 5, 21, 141]$.

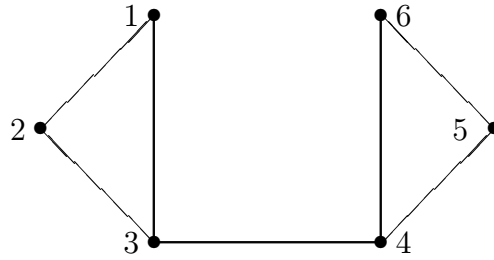


FIGURA 5.1

Si se considera la coalición $S \in \mathcal{L}$ dada por $S = \{2, 3, 4, 5\}$, se verifica que $ex(S) = \{2, 5\}$, $2^{ex(S)} = \{\emptyset, \{2\}, \{5\}, \{2, 5\}\}$, $S \setminus ex(S) = \{3, 4\}$ y

$$[S^-, S] = \{\{3, 4\}, \{3, 4, 2\}, \{3, 4, 5\}, \{3, 4, 2, 5\}\}.$$

En este caso,

$$\begin{aligned} P_S(x, z) &= x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \prod_{i \in ex(S)} (1 + x^{w_i} z) \\ &= x^{21} z^2 (1 + x^4 z) (1 + x^{21} z) \\ &= x^{21} z^2 (1 + x^4 z + x^{21} z + x^{25} z^2) \\ &= x^{21} z^2 + x^{25} z^3 + x^{42} z^3 + x^{46} z^4, \end{aligned}$$

y obsérvese que los coeficientes del polinomio $P_S(x, z)$ se corresponden con

el número de coaliciones del intervalo $[S^-, S]$ con un determinado peso y tamaño

$T \in [S^-, S]$	$ T $	$w(T)$	$b_{kj}^S x^k z^j$
$\{3, 4\}$	2	21	$x^{21} z^2$
$\{3, 4, 2\}$	3	25	$x^{25} z^3$
$\{3, 4, 5\}$	3	42	$x^{42} z^3$
$\{3, 4, 2, 5\}$	4	46	$x^{46} z^4$

El conocimiento de los números $\{b_{kj}^S\}_{k \geq 0, j \geq 0}$, mediante la función generatriz $P_S(x, z)$, posibilita la determinación de los números $\{a_r^S\}_{r \geq 0}$ necesarios para el cálculo del dividendo correspondiente a la coalición S en un juego simple de votación ponderada restringido por una geometría convexa de partición.

Para determinarlos, se puede considerar, por una parte, que estos números $\{a_r^S\}_{r \geq 0}$ se identifican con los coeficientes de un polinomio de la forma

$$C_S(z) = \sum_{r=0}^{|ex(S)|} a_r^S z^{|S|-r}$$

y, por otra parte, la Proposición 5.2 permite establecer que

$$P_S(x, z) = \sum_{j=|S \setminus ex(S)|}^{|S|} \left[\sum_{k=w(S \setminus ex(S))}^{w(S)} b_{kj}^S x^k \right] z^j,$$

con lo que, haciendo $r = |S| - j$, con $0 \leq r \leq ex(S)$, resulta

$$P_S(x, z) = \sum_{r=0}^{|ex(S)|} \left[\sum_{k=w(S \setminus ex(S))}^{w(S)} b_{kr}^S x^k \right] z^{|S|-r}.$$

De ahí, se deduce que para determinar los coeficientes de $C_S(z)$ y, en consecuencia, los números $\{a_r^S\}_{r \geq 0}$, es suficiente, para cada potencia de la

variable z en $P_S(x, z)$, seleccionar los coeficientes de los monomios $x^k z^{s-r}$, con $s = |S|$, en los que el exponente k de la variable x sea mayor o igual que q . Es decir,

$$C_S(z) = \sum_{r=0}^{|ex(S)|} a_r^S z^{s-r} = \sum_{r=0}^{|ex(S)|} \left[\sum_{k=q}^{w(S)} b_{kr}^S \right] z^{s-r},$$

y, por tanto, $a_r^S = \sum_{k=q}^{w(S)} b_{kr}^S$.

Es lógico que, a partir de los resultados y comentarios anteriores, se obtenga una expresión de los dividendos de Harsanyi en términos de las funciones generatrices. Ello se expone en el siguiente resultado.

Proposición 5.4 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Entonces, dada una coalición $S \in \mathcal{L}$, el dividendo de la coalición S en el juego restringido $(N, v^{\mathcal{L}})$ viene dado por*

$$\Delta_{v^{\mathcal{L}}}(S) = (-1)^{|S|} C_S(-1),$$

donde $C_S(z)$ es la función generatriz de los números $\{a_r^S\}_{r \geq 0}$ donde cada a_r^S es el número de coaliciones del intervalo $[S^-, S]$, de tamaño $|S| - r$ y con peso mayor o igual que q .

Demostración: Teniendo en cuenta la Proposición 5.1,

$$\Delta_{v^{\mathcal{L}}}(S) = \sum_{r=0}^{|ex(S)|} (-1)^r a_r^S.$$

Esta expresión puede reescribirse como

$$\Delta_{v^{\mathcal{L}}}(S) = (-1)^s \sum_{r=0}^{|ex(S)|} (-1)^{s-r} a_r^S,$$

con $s = |S|$. Si se considera que $C_S(z) = \sum_{r=0}^{|ex(S)|} a_r^S z^{s-r}$, el resultado es inmediato. □

Ejemplo 5.5 *Considérese la terna (N, v, \mathcal{L}) del Ejemplo 5.3. Para la coalición factible $S = \{2, 3, 4, 5\}$, el polinomio $P_S(x, z)$ viene dado por*

$$P_S(x, z) = x^{21} z^2 + x^{25} z^3 + x^{42} z^3 + x^{46} z^4,$$

y el polinomio $C_S(z)$ se obtiene como sigue

$$C_S(z) = \sum_{r=0}^{|ex(S)|} \left[\sum_{k=q}^{w(S)} b_{kr}^S \right] z^{s-r} = \sum_{r=0}^2 \left[\sum_{k=176}^{46} b_{kr}^S \right] z^{4-r} = 0,$$

al ser la cuota $q = 176$ y $w(S) = 46$. Por tanto,

$$\Delta_{v\mathcal{L}}(\{2, 3, 4, 5\}) = (-1)^4 C_S(-1) = 0.$$

Si se considera ahora la coalición factible $S = \{1, 2, 3\}$, el cálculo del dividendo $\Delta_{v\mathcal{L}}(S)$ sería de la siguiente manera: como $ex(S) = \{1, 2, 3\}$, resulta que

$$\begin{aligned} P_S(x, z) &= \prod_{i=1}^3 (1 + x^{w_i} z) \\ &= (1 + x^{156} z)(1 + x^4 z)(1 + x^{16} z) \\ &= 1 + x^4 z + x^{16} z + x^{156} z + x^{20} z^2 + x^{160} z^2 + x^{172} z^2 + x^{176} z^3, \end{aligned}$$

y, entonces,

$$C_S(z) = \sum_{r=0}^{|ex(S)|} \left[\sum_{k=q}^{w(S)} b_{kr}^S \right] z^{s-r} = \sum_{r=0}^3 \left[\sum_{k=176}^{176} b_{kr}^S \right] z^{3-r} = z^3.$$

Finalmente, en este caso,

$$\Delta_{v\mathcal{L}}(\{1, 2, 3\}) = (-1)^s C_S(-1) = (-1)^3 (-1)^3 = 1.$$

Corolario 5.6 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El valor de Myerson generalizado viene dado, para cada jugador $i \in N$, por*

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{S \in \mathcal{L}: i \in S\}} \frac{1}{|S|} [(-1)^{|S|} C_S(-1)],$$

donde $C_S(z)$ es la función generatriz de los números $\{a_r^S\}_{r \geq 0}$ de coaliciones del intervalo $[S^-, S]$, de tamaño $|S| - r$, con peso mayor o igual que q .

Una vez establecido un nuevo procedimiento para el cálculo del valor de Myerson generalizado en juegos simples de votación ponderada, se inicia el estudio de la complejidad temporal del correspondiente algoritmo. Para ello, se analiza, en primer lugar, para cada coalición factible S , la complejidad temporal asociada a la expansión del polinomio $P_S(x, z)$.

Proposición 5.7 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. Para cada $S \in \mathcal{L}$, el número de coeficientes no nulos, denotado por c , del polinomio $P_S(x, z)$ verifica*

$$|ex(S)| + 1 \leq c \leq \min(2^{|ex(S)|}, |ex(S)| w(ex(S)) + 1).$$

Demostración: Dada una coalición $S \in \mathcal{L}$, el polinomio $P_S(x, z)$ viene dado por

$$P_S(x, z) = x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \prod_{i \in ex(S)} (1 + x^{w_i} z).$$

El número de coeficientes no nulos de $P_S(x, z)$ coincide con el número de coeficientes no nulos del polinomio $\prod_{i \in ex(S)} (1 + x^{w_i} z)$. Considerando la Proposición 4.14 el resultado es inmediato.

□

Proposición 5.8 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. La expansión del polinomio $P_S(x, z)$, para cada $S \in \mathcal{L}$, requiere un tiempo $\mathcal{O}(|ex(S)|^C)$, siendo $C = \min(2^{|ex(S)|}, |ex(S)| w(ex(S)) + 1)$.*

Demostración: Sea $S \in \mathcal{L}$, la expansión del polinomio

$$P_S(x, z) = x^{w(S \setminus ex(S))} z^{|S \setminus ex(S)|} \prod_{i \in ex(S)} (1 + x^{w_i} z)$$

viene determinada por la expansión del producto

$$\prod_{i \in ex(S)} (1 + x^{w_i} z)$$

la cual, por la Proposición 4.16, requiere un tiempo $\mathcal{O}(|ex(S)|^C)$, siendo $C = \min(2^{|ex(S)|}, |ex(S)| w(ex(S)) + 1)$.

□

A continuación, en la siguiente página, se describe el algoritmo *geomdiv* que se utilizará para calcular el valor de los dividendos en un juego de votación ponderada (N, v) , con $v = [q; w_1, \dots, w_n]$, restringido por una geometría convexa de partición.

Algoritmo *geomdiv*(*pesos*, *q*)

{*pesos*: lista de n enteros; q : entero}

$\Delta_{v\mathcal{L}}(\emptyset) \leftarrow 0$

para p **de** 1 **hasta** $|\mathcal{L}|$

lista_aux $\leftarrow ex(S_p)$

$P_{S_p}(x, z) \leftarrow x^{w(S_p \setminus ex(S_p))} z^{|S_p \setminus ex(S_p)|} \prod_{i \in lista_aux} (1 + x^{w_i} z)$

$$\left\{ P_{S_p}(x, z) = \sum_{k=w(S_p \setminus ex(S_p))}^{w(S_p)} \sum_{j=|S_p \setminus ex(S_p)|}^{|S_p|} b_{kj}^{S_p} x^k z^j \right\}$$

$C_{S_p}(z) \leftarrow \sum_{r=0}^{|ex(S_p)|} a_r^{S_p} z^{|S_p|-r}$

$$\left\{ C_{S_p}(z) = \sum_{r=0}^{|ex(S_p)|} \left[\sum_{k=q}^{w(S_p)} b_{kr}^{S_p} \right] z^{|S_p|-r} = \sum_{r=0}^{|ex(S_p)|} a_r^{S_p} z^{|S_p|-r} \right\}$$

$\Delta_{v\mathcal{L}}(S_p) \leftarrow (-1)^{|S_p|} C_{S_p}(-1)$

fin_para

Proposición 5.9 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. El cálculo de los dividendos, mediante el algoritmo *geomdiv*, requiere un tiempo $\mathcal{O}(|\mathcal{L}| DE)$, siendo $D = \max\{|ex(S)| : S \in \mathcal{L}\}$, y $E = \max\{\min(2^{|ex(S)|}, |ex(S)| w(ex(S)) + 1) : S \in \mathcal{L}\}$.*

Demostración: Sea $f(n)$ la función que determina el tiempo de ejecución del algoritmo *geomdiv*. Entonces, el orden de la complejidad de la función $f(n)$ viene dado por

$$\mathcal{O}(f(n)) = 1 + \mathcal{O}(t(\text{bucle})) = \mathcal{O}(|\mathcal{L}| (t(\text{asignación1}) + \dots + t(\text{asignación4}))).$$

Ahora bien, dentro del bucle, la complejidad de la primera asignación sería

$$\mathcal{O}(t(\text{asignación1})) = \mathcal{O}(t(\text{lista_aux})) = \mathcal{O}(1),$$

ya que se supone que los extremales se han calculado previamente, y se encuentran almacenados. Para la complejidad de la segunda asignación, se tiene que, en el peor caso,

$$\mathcal{O}(t(\text{asignación2})) = \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(D E),$$

lo cual es consecuencia de la Proposición 5.8. Del mismo modo, y considerando el peor caso,

$$\mathcal{O}(t(\text{asignación3})) = \mathcal{O}(t(\text{suma})) = \mathcal{O}(E),$$

al ser E una cota superior del número de coeficientes no nulos de cualquier polinomio $P_S(x, z)$. Análogamente, teniendo en cuenta que, para toda coalición $S \in \mathcal{L}$, la evaluación de los polinomios $C_S(z)$, que son de grado $|ex(S)|$, en $z = -1$, se realiza en un tiempo $\mathcal{O}(|ex(S)|)$, y $|ex(S)| \leq D$, resulta ser

$$\mathcal{O}(t(\text{asignación4})) = \mathcal{O}(D).$$

Finalmente, si se considera la regla del máximo de $\mathcal{O}(\cdot)$, se sigue que,

$$\mathcal{O}(f(n)) = \mathcal{O}(|\mathcal{L}| (t(\text{asignación2}))) = \mathcal{O}(|\mathcal{L}| D E).$$

□

Proposición 5.10 *Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. El cálculo del valor de Myerson generalizado de todos los jugadores, utilizando los dividendos de Harsanyi, requiere un tiempo $\mathcal{O}(\max(n |\mathcal{L}|, |\mathcal{L}| D E)$.*

Demostración: En primer lugar, con el algoritmo *geomdiv* se calculan todos los dividendos de Harsanyi de las coaliciones convexas, lo cual requiere, según la proposición anterior, un tiempo $\mathcal{O}(|\mathcal{L}| DE)$. A continuación, para calcular el valor de Myerson del jugador i ,

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{S \in \mathcal{L}: i \in S\}} \frac{\Delta_{v^{\mathcal{L}}}(S)}{|S|},$$

hay que considerar que cada jugador pertenece a $S(i)$ coaliciones, siendo $S(i) < |\mathcal{L}|$. Como el tiempo requerido para evaluar la suma en la expresión del valor de Myerson generalizado es, para cada jugador, $\mathcal{O}(|\mathcal{L}|)$, se deduce, por tanto, que el cálculo del valor de Myerson generalizado para los n jugadores requiere un tiempo $\mathcal{O}(\max(n|\mathcal{L}|, |\mathcal{L}| DE)$.

□

5.2 El valor de Myerson en juegos restringidos por grafos estrella

Tal como se indicó en los primeros párrafos introductorios al capítulo, se presenta, en esta sección, una nueva fórmula explícita, usando funciones generatrices, para el cálculo el valor de Myerson en juegos simples de votación ponderada y en situaciones de comunicación, (N, v, G) , en las que G es un grafo estrella.

Nótese, en primer lugar, que, en esta situación, el par (N, \mathcal{L}) , donde \mathcal{L} es la familia de coaliciones factibles

$$\mathcal{L} = \{S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G \},$$

es una geometría convexa de partición. Por dicho motivo se exponen, previamente, resultados dados por Bilbao [10] para el cálculo del valor de Myerson generalizado en juegos restringidos por geometrías convexas de partición.

Definición 5.11 Sea (N, \mathcal{L}) una geometría convexa. Para $T \in \mathcal{L}$, se denota por T^+ al subconjunto de N constituido por

$$T^+ = \{i \in N : T \cup i \in \mathcal{L}\}$$

De la definición se desprende que, para cada $T \in \mathcal{L}$, T^+ es no vacío por ser \mathcal{L} una geometría convexa. Además, puede demostrarse que si (N, \mathcal{L}) es una geometría convexa de partición, se verifica que, para cada $T \in \mathcal{L}$ no vacía, el intervalo $[T, T^+]$ es un álgebra de Boole isomorfa a $2^{T^+ \setminus T}$, y que si $T = \emptyset$, entonces $[T, T^+] = \mathcal{L}$.

Teorema 5.12 (Bilbao (1998)) Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego. El valor de Myerson generalizado, para cada jugador $i \in N$, viene dado por

$$\begin{aligned} \mu_i(N, v, \mathcal{L}) &= \sum_{T \in \mathcal{L}_i^+} \frac{(t-1)!(t^+ - t)!}{t^+!} [v(T) - v(T \setminus i)] \\ &+ \sum_{T \in \mathcal{L}_i \setminus \mathcal{L}_i^+} \frac{(t-1)!(t^+ - t)!}{t^+!} v(T) \\ &- \sum_{T \in \mathcal{L}_i^*} \frac{t!(t^+ - t + 1)!}{t^+!} v(T), \end{aligned}$$

donde $\mathcal{L}_i = \{T \in \mathcal{L} : i \in T\}$, $\mathcal{L}_i^+ = \{T \in \mathcal{L} : i \in \text{ex}(T), (T \setminus i)^+ = T^+\}$, $\mathcal{L}_i^* = \{T \in \mathcal{L} : i \notin T, T \cup i \in \mathcal{L}, T^+ \neq (T \cup i)^+\}$, $t = |T|$ y $t^+ = |T^+|$.

Teorema 5.13 (Bilbao (1998)) Sea la terna (N, v, \mathcal{L}) , donde (N, \mathcal{L}) es una geometría convexa de partición y (N, v) un juego cero-normalizado. Si el jugador $i \in \text{ex}(N)$, entonces

$$\mu_i(N, v, \mathcal{L}) = \sum_{\{T \in \mathcal{L}, |T| \geq 2\}} \frac{(t-1)!(t^+ - t)!}{t^+!} [v(T) - v(T \setminus i)],$$

donde $t = |T|$ y $t^+ = |T^+|$.

Considérese, ahora, la situación de comunicación (N, v, G) , donde G es un grafo estrella y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. Recuérdese que, en esta situación, el par (N, \mathcal{L}) , siendo

$$\mathcal{L} = \{S \subseteq N : (S, E(S)) \text{ es un subgrafo conexo de } G \},$$

es una geometría convexa de partición y el valor de Myerson generalizado es, obviamente, el valor de Myerson correspondiente a la situación de comunicación considerada. Así, en este contexto, si n es el nodo que ocupa la posición central del grafo estrella, se verifica que $ex(N) = \{1, \dots, n-1\}$ y, en consecuencia, es posible calcular el valor de Myerson de estos jugadores mediante la fórmula del Teorema 5.13. De forma análoga, puede utilizarse el Teorema 5.12 para determinar el valor de Myerson correspondiente al jugador central, aunque también sería suficiente, para su cálculo, considerar que, en este caso, $\sum_{i=1}^n \mu_i(N, v, G) = 1$, ya que (N, v) es un juego simple de votación ponderada. Estas últimas ideas se formalizan en los resultados que, a continuación, se exponen.

Teorema 5.14 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. Si $i \in ex(N)$, entonces*

$$\mu_i(N, v, G) = \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{q-1+w_i} c_{kj}^i \right),$$

donde c_{kj}^i es el número de coaliciones factibles T , con $|T| = j$, que contienen al jugador i y cuyo peso es $w(T) = k$.

Demostración: Considérese, sin pérdida de generalidad, que el jugador n ocupa la posición central y que \mathcal{L} es la familia de coaliciones factibles correspondiente al grafo estrella G . Entonces, el par (N, \mathcal{L}) es una geometría convexa de partición, $ex(N) = \{1, \dots, n-1\}$ y se verifica, para toda coalición $T \in \mathcal{L}$ con $|T| \geq 2$, que $n \in T$ y $T^+ = N$. Por tanto, por el Teorema 5.13, se

tiene, para cualquier $i \in ex(N)$, que

$$\begin{aligned} \mu_i(N, v, G) &= \sum_{\{T \in \mathcal{L}: |T| \geq 2\}} \frac{(t-1)!(t^+ - t)!}{t^+!} [v(T) - v(T \setminus i)] \\ &= \sum_{\{T \in \mathcal{L}: |T| \geq 2\}} \frac{(t-1)!(n-t)!}{n!} [v(T) - v(T \setminus i)] \\ &= \sum_{\{T \in \mathcal{L} \cap \mathcal{W}: |T| \geq 2, T \setminus i \notin \mathcal{W}\}} \frac{(t-1)!(n-t)!}{n!}, \end{aligned}$$

siendo \mathcal{W} el conjunto de coaliciones tales que $v(T) = 1$. Ahora bien, si T es una coalición que incluye al jugador i y, además, verifica que $v(T) = 1$, entonces $w(T) \geq q$. Por otro lado, si la coalición $T \setminus i$ es tal que $v(T \setminus i) = 0$, entonces $0 \leq w(T \setminus i) < q$ o, lo que es lo mismo, $w_i \leq w(T) < q + w_i$. Por tanto, $w_i < q \leq w(T) < q + w_i$, por lo que se deduce que los pesos de las coaliciones para las que se verifica la igualdad $v(T) - v(T \setminus i) = 1$ satisfacen que $w(T) \in [q, q + w_i - 1]$. Luego,

$$\mu_i(N, v, G) = \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{q-1+w_i} c_{kj}^i \right),$$

donde c_{kj}^i es el número de coaliciones factibles T , con $|T| = j$, que contienen al jugador i y cuyo peso es $w(T) = k$. □

A continuación, se introduce una fórmula que permite calcular el valor del Myerson del jugador central de manera directa, sin necesidad de determinar el valor de Myerson correspondiente a los demás jugadores.

Teorema 5.15 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El valor de Myerson del jugador central n viene dado por*

$$\mu_n(N, v, G) = \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{w(N)} c_{kj}^n \right),$$

donde c_{kj}^n es el número de coaliciones factibles T , con $|T| = j$, que contienen al jugador n y cuyo peso es $w(T) = k$.

Demostración: De igual forma que se razonó en el Teorema 5.14, el par (N, \mathcal{L}) , siendo \mathcal{L} la familia de coaliciones factibles correspondiente al grafo estrella G , es una geometría convexa de partición. Así, por el Teorema 5.12, se tiene

$$\begin{aligned} \mu_n(N, v, G) &= \sum_{T \in \mathcal{L}_n^+} \frac{(t-1)!(t^+ - t)!}{t^+!} [v(T) - v(T \setminus n)] \\ &\quad + \sum_{T \in \mathcal{L}_n \setminus \mathcal{L}_n^+} \frac{(t-1)!(t^+ - t)!}{t^+!} v(T) \\ &\quad - \sum_{T \in \mathcal{L}_n^*} \frac{t!(t^+ - t + 1)!}{t^+!} v(T). \end{aligned}$$

Puesto que (N, v) es un juego cero-normalizado resulta que

$$\begin{aligned} \mu_n(N, v, G) &= \sum_{\{T \in \mathcal{L}_n^+ : |T| \geq 2\}} \frac{(t-1)!(t^+ - t)!}{t^+!} [v(T) - v(T \setminus n)] \\ &\quad + \sum_{\{T \in \mathcal{L}_n \setminus \mathcal{L}_n^+ : |T| \geq 2\}} \frac{(t-1)!(t^+ - t)!}{t^+!} v(T) \\ &\quad - \sum_{\{T \in \mathcal{L}_n^* : |T| \geq 2\}} \frac{t!(t^+ - t + 1)!}{t^+!} v(T). \end{aligned}$$

Ahora bien, los conjuntos $\{T \in \mathcal{L}_n^+ : |T| \geq 2\}$ y $\{T \in \mathcal{L}_n^* : |T| \geq 2\}$ son vacíos. En efecto, $\{T \in \mathcal{L}_n^+ : |T| \geq 2\}$ es vacío ya que, por definición

$$\{T \in \mathcal{L}_n^+ : |T| \geq 2\} = \{T \in \mathcal{L} : n \in \text{ex}(T), (T \setminus n)^+ = T^+, |T| \geq 2\},$$

las únicas coaliciones factibles T que tienen a n como elemento extremal son aquellas tales que $|T| = 2$ y, para éstas, $(T \setminus n)^+ = T$ y $T^+ = N$, con lo que $(T \setminus n)^+ \neq T^+$. Por otra parte, $\{T \in \mathcal{L}_n^* : |T| \geq 2\}$ también es vacío ya que, por definición, estaría constituido por

$$\{T \in \mathcal{L} : n \notin T, T \cup n \in \mathcal{L}, T^+ \neq (T \cup n)^+, |T| \geq 2\},$$

y $n \in T$, para toda coalición factible con $|T| \geq 2$.

Entonces, resulta

$$\mu_n(N, v, G) = \sum_{\{T \in \mathcal{L}_n : |T| \geq 2\}} \frac{(t-1)!(n-t)!}{n!} v(T)$$

Como $\{T \in \mathcal{L}_n : |T| \geq 2\} = \{T \in \mathcal{L} : |T| \geq 2\}$, se sigue que

$$\begin{aligned} \mu_n(N, v, G) &= \sum_{\{T \in \mathcal{L} : |T| \geq 2\}} \frac{(t-1)!(n-t)!}{n!} v(T) \\ &= \sum_{\{T \in \mathcal{L} \cap \mathcal{W} : |T| \geq 2\}} \frac{(t-1)!(n-t)!}{n!} \\ &= \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{w(N)} c_{kj}^n \right), \end{aligned}$$

siendo c_{kj}^n el número de coaliciones factibles que incluyen al jugador central n , y que tienen peso k y tamaño j .

□

Hasta aquí se puede establecer, como conclusión, que si se considera una situación de comunicación (N, v, G) en la que G es un grafo estrella, el hecho de que las coaliciones factibles configuren una geometría convexa de partición ha permitido demostrar que

$$\begin{aligned} \mu_i(N, v, G) &= \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{q-1+w_i} c_{kj}^i \right), \\ \mu_n(N, v, G) &= \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{w(N)} c_{kj}^n \right), \end{aligned}$$

siendo $i \in ex(N)$, y n el jugador que ocupa la posición central.

Lógicamente, para la realización del cálculo efectivo del valor de Myerson de cada uno de los jugadores mediante las expresiones indicadas en el párrafo

anterior, es exigible el conocimiento del número de coaliciones factibles en las que participa cada jugador que, además, tienen un determinado peso y tamaño. En general, este conocimiento es difícil de obtener aunque, dentro de las situaciones de comunicación, es posible determinar el número de coaliciones conexas o factibles cuando el grafo que representa las relaciones entre los jugadores es un árbol. Ello ha sido dado a conocer por Jamison cuando estudia el cálculo de los *números de Whitney* de un árbol.

De ahí que se dediquen las siguientes páginas a la enumeración de subgrafos conexas en árboles y de las coaliciones conexas de un tamaño y peso determinados. Posteriormente, en la última parte de la sección, se tratará el estudio de la complejidad temporal del algoritmo, para el cálculo del valor de Myerson, que surge de incorporar las correspondientes técnicas de enumeración que se desarrollan a continuación.

Definición 5.16 Sea $G = (N, E)$ un árbol con $|N| = n$ nodos. Se denominan *números de Whitney* a los números $A_i(G)$, con $i = 1, \dots, n$, donde cada $A_i(G)$ representa el número de subárboles con i vértices.

Para el cálculo de los números de Whitney de un árbol G , Jamison [38] considera la función generatriz

$$\Phi_G(z) = \sum_{j=1}^n A_j(G) z^j,$$

obtenida de forma recursiva tras fijar un nodo cualquiera del árbol como raíz y considerar, a su vez, otras funciones generatrices las cuales, junto con algunas notaciones, se describen en lo que sigue para una correcta interpretación del proceso de cálculo.

Sea p cualquier nodo de G , y considérese la función

$$\varphi_G(p; z) = \sum_{j=1}^n \alpha_j(G; p) z^j.$$

Cada coeficiente $\alpha_j(G; p)$ expresa el número de subárboles de G con j nodos que contienen al vértice p .

Se denotará por G_p el árbol G considerando el nodo p como raíz y por $D(i; G_p)$ el conjunto *descendientes* de i en el árbol G_p ; esto es

$$D(i; G_p) = \{l \in N : i \text{ se encuentra en el camino de } l \text{ a } p\}.$$

Además, por convenio, $\varphi_G(i|p; z)$ representa la función generatriz $\varphi_D(i; z)$, donde $D = (D(i; G_p), E(D(i; G_p)))$, y, siempre que no haya lugar a confusión, se suprimirán los subíndices de las funciones generatrices φ .

Teorema 5.17 (*Jamison (1987)*). *Para cualquier nodo p en un árbol G , se verifica*

- a) $\varphi(p; z) = z \prod (1 + \varphi(i|p; z))$, donde el producto recorre todos los hijos i del nodo p en G .
- b) $\Phi_G(z) = \sum \varphi(i|p; z)$, donde la suma recorre todos los nodos de G .

Nótese que, del resultado anterior, se desprende un algoritmo para determinar de forma recursiva los números de Whitney del árbol G . Para ello, se considera el árbol enraizado en un vértice cualquiera p . A continuación, en primer lugar, se aplica el apartado a) del Teorema 5.17 al nodo p para el cálculo de $\varphi(p; z)$ quedando, este cálculo, pendiente de la determinación de las funciones $\varphi(i|p; z)$, donde la variable i recorre los hijos del nodo p en el árbol G_p . Para la determinación de las funciones $\varphi(i|p; z)$ se vuelve a aplicar, tomando a cada nodo i como raíz del subárbol constituido por $D(i; G_p)$, el apartado a) del Teorema 5.17. La aplicación reiterada de éste apartado a) da lugar a un proceso de recorrido en profundidad desde los diferentes vértices que permite calcular las funciones generatrices $\varphi(i|p; z)$ para todos los nodos. Cada función $\varphi(i|p; z)$ indica, tomando al nodo i como raíz, los números de Whitney de los respectivos subárboles en este proceso descendente —desde

el nodo raíz p hasta las hojas— y quedan perfectamente determinadas al ser el proceso establecido finito, ya que la función $\varphi(i|p; z)$, cuando el vértice i es una hoja, es $\varphi(i|p; z) = z$, lo cual permite el cálculo de todas las demás. Por último, para obtener los números de Whitney del árbol G bastará con efectuar la suma de todas funciones $\varphi(i|p; z)$ extendida a todos los nodos del árbol con raíz G_p .

Ejemplo 5.18 Sea el árbol $G = (N, E)$, con $N = \{1, 2, 3, 4, 5\}$ y

$$E = \{\{2, 4\}, \{1, 2\}, \{2, 5\}, \{1, 3\}\}.$$

En este caso, para determinar de forma recursiva los números de Whitney del árbol G , se considera —véase Figura 5.2— el árbol enraizado en el vértice 1.

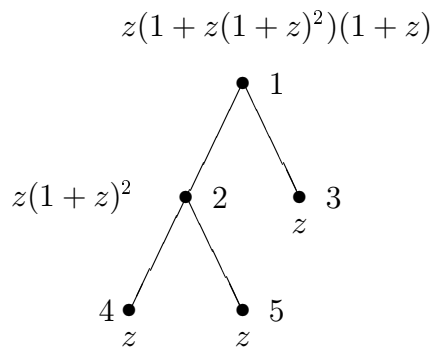


FIGURA 5.2

La aplicación del Teorema de Jamison y del proceso de recorrido en profundidad, desde el vértice raíz, para el cálculo de todas las funciones genera-

trices $\varphi(i|p; z)$ sería como sigue

$$\begin{aligned}
 \varphi(1; z) &= z(1 + \varphi(2|1; z))(1 + \varphi(3|1; z)) \\
 &= z(1 + z(1 + \varphi(4|1; z))(1 + \varphi(5|1; z)))(1 + z) \\
 &= z(1 + z(1 + z)^2)(1 + z), \\
 \varphi(2|1; z) &= z(1 + \varphi(4|1; z))(1 + \varphi(5|1; z)) \\
 &= z(1 + z)^2, \\
 \varphi(3|1; z) &= z, \\
 \varphi(4|1; z) &= z, \\
 \varphi(5|1; z) &= z.
 \end{aligned}$$

Finalmente, para obtener la función generatriz $\Phi_G(z)$, que determina los números de Whitney del árbol G , será suficiente con efectuar la suma de las funciones $\varphi(i|p; z)$ extendida a todos los nodos del árbol con raíz G_1 . Es decir,

$$\begin{aligned}
 \Phi_G(z) &= \sum_{i=1}^5 \varphi(i|p; z) \\
 &= z(1 + z(1 + z)^2)(1 + z) + z(1 + z)^2 + z + z + z \\
 &= z^5 + 3z^4 + 4z^3 + 4z^2 + 5z.
 \end{aligned}$$

Nótese que si se considera el sistema de coaliciones factibles o conexas asociado al árbol G , se tiene

$$\begin{aligned}
 \mathcal{L} = & \{ \emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 3\}, \{1, 2\}, \{2, 4\}, \{2, 5\}, \{1, 2, 3\}, \{1, 2, 4\}, \\
 & \{1, 2, 5\}, \{2, 4, 5\}, \{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 4, 5\}, \{1, 2, 3, 4, 5\} \},
 \end{aligned}$$

y puede observarse que, en efecto, el número de coaliciones factibles de tamaño j se corresponde con el número de Whitney $A_j(G)$ del árbol G , el cual coincide con el coeficiente del término de grado j en el polinomio $\Phi_G(z) = z^5 + 3z^4 + 4z^3 + 4z^2 + 5z$.

El Teorema 5.17 permite considerar, para calcular los números de Whitney de un árbol, la función denominada *Jamison*, cuya descripción se indica

a continuación. Posteriormente, se estudiará la complejidad temporal del algoritmo.

Función $Jamison(p)$

{ p es el nodo raíz}

$\varphi(p; z) \leftarrow z$

[**Para** cada i hijo de la raíz **hacer**
 $\varphi(p; z) \leftarrow \varphi(p; z)(1 + Jamison(i))$
fin_para]

devolver $\varphi(p; z)$

Proposición 5.19 Sean G un árbol y p un nodo cualquiera de G . La obtención de la función $\varphi(p; z) = z \prod (1 + \varphi(i|p; z))$, donde el producto recorre todos los hijos i de p en G , requiere un tiempo $\mathcal{O}(r^h)$, siendo h y r , respectivamente, la altura y el factor de ramificación del árbol con raíz G_p .

Demostración: Sea el árbol con raíz G_p . Como se va a realizar un análisis en el peor caso, se considera que, excepto para las hojas del árbol, todos los nodos tienen un número de hijos igual al factor de ramificación r .

El tiempo de ejecución de la función $Jamison$ para el nodo p , que se encuentra a una altura h , viene dado por

$$t(h) = 2 + 2r + r t(h - 1),$$

ya que en la ejecución de $Jamison(p)$ se requieren dos asignaciones y hay que hacer r llamadas recursivas, para cada uno de los r hijos del nodo i , a la función $Jamison$. Además, junto a las r llamadas, hay que contabilizar la devolución del valor de la función para cada una de ellas y, a esto, hay que

añadir el tiempo de ejecución de la función *Jamison* para cada uno de los hijos del nodo p , los cuales se encuentran a una altura $h - 1$.

Expandiendo la recurrencia anterior,

$$\begin{aligned}
 t(h) &= 2 + 2r + r t(h - 1) \\
 &= 2 + 2r + r(2 + 2r + r t(h - 2)) \\
 &= 2 + 2r + 2r + 2r^2 + r^2 t(h - 2) \\
 &= 2 + 4r + 2r^2 + r^2 (2 + 2r + r t(h - 3)) \\
 &= 2 + 4r + 4r^2 + 2r^3 + r^3 t(h - 3) \\
 &= 2 + 4r + 4r^2 + 2r^3 + r^3 (2 + 2r + r t(h - 4)) \\
 &= 2 + 4r + 4r^2 + 4r^3 + 2r^4 + r^4 t(h - 4) \\
 &\vdots \\
 &= 2 + 4r + 4r^2 + 4r^3 + \dots + 2r^k + r^k t(h - k).
 \end{aligned}$$

Como, para $k = h$, $t(0) = 2$, resulta que

$$\begin{aligned}
 t(h) &= 2 + 4r + 4r^2 + 4r^3 + \dots + 2r^h + r^h t(0) \\
 &= 2 + 4r + 4r^2 + 4r^3 + \dots + 4r^h.
 \end{aligned}$$

Por tanto, $\mathcal{O}(t(h)) = \mathcal{O}(4r^h) = \mathcal{O}(r^h)$.

□

Proposición 5.20 Sean G un árbol y p un nodo cualquiera de G . La expansión del polinomio $\varphi(p; z) = z \prod (1 + \varphi(i|p; z))$, donde el producto recorre todos los hijos i de p en G , requiere un tiempo $\mathcal{O}(r^h)$, siendo h y r , respectivamente, la altura y el factor de ramificación del árbol con raíz G_p .

Demostración: Sea el árbol con raíz G_p . Para analizar la complejidad de la expansión del polinomio $\varphi(p; z) = z \prod (1 + \varphi(i|p; z))$, donde el producto recorre todos los hijos i de p en G , se considera que, para el árbol con raíz G_p , en el peor caso, el número de hijos de la raíz y de cada nodo interno coincide con el factor de ramificación, y que las hojas del árbol se encuentran todas a la misma profundidad.

Así, si l es un nodo de altura 1, entonces

$$\varphi(l; z) = z \prod_{j=1}^r (1 + \varphi(l_j; z)) = z(1 + z)^r,$$

donde los nodos l_j , que son hojas del árbol, son los hijos del nodo l . La expansión de $(1 + z)^r$ requiere, utilizando la Proposición 4.9, un número de operaciones $\mathcal{O}(r^2)$. Por tanto, la expansión de $\varphi(l; z)$, para un nodo de altura 1, requiere un tiempo $\mathcal{O}(t(1)) = \mathcal{O}(r^2)$.

Teniendo en cuenta que la raíz del árbol se encuentra a una altura h , se puede plantear la siguiente recurrencia

$$t(h) = \begin{cases} r^2, & \text{si } h = 1 \\ r t(h - 1), & \text{si } h \geq 2 \end{cases}$$

De lo anterior, se obtiene

$$\begin{aligned} t(h) &= r t(h - 1) \\ &= r^2 t(h - 2) \\ &= r^3 t(h - 3) \\ &\vdots \\ &= r^k t(h - k). \end{aligned}$$

Si $k = h - 1$, entonces $t(h) = r^{h-1} t(1) = r^{h-1} r^2 = r^{h+1}$. Por tanto, finalmente, $\mathcal{O}(t(h)) = \mathcal{O}(r^h)$.

□

Recuérdese que la determinación del número de coaliciones factibles o conexas existentes en una situación de comunicación (N, v, G) , siendo G un grafo estrella y (N, v) un juego simple de votación ponderada, era necesaria para el cálculo del valor de Myerson de cada uno de los diferentes jugadores. Con más precisión: se requiere el conocimiento, para cada jugador, del número de coaliciones factibles a las que pertenezca y que posean un determinado peso y tamaño. Por ello, el análisis realizado es aún insuficiente y se introduce, ahora, la introducción de un juego simple de votación

ponderada para asignar a cada jugador un peso determinado y proseguir el estudio planteado con los resultados correspondientes a los Teoremas 5.14 y 5.15.

Sin embargo, antes de continuar, hay que subrayar la importancia de la función *Jamison*. En efecto, nótese, que en el tercer capítulo de esta memoria, para situaciones de comunicación (N, v, G) en las que G era un árbol, se dieron resultados de complejidad temporal en función del cardinal del conjunto \mathcal{F} , formado por todas las coaliciones factibles, y, por tanto, tiene bastante interés el conocimiento de dicho número. Utilizando el algoritmo de Jamison es posible conocerlo, a priori, sin necesidad de obtener toda la familia de coaliciones factibles o conexas ya que únicamente es necesaria la información referente a las aristas de G . En definitiva, si además se tiene en cuenta —como pone de manifiesto Jamison— que, para árboles con un mismo número de vértices, los números de coaliciones conexas pueden ser muy dispares, el conocimiento del número $|\mathcal{F}|$ proporciona, por anticipado, un dato fundamental y revelador de la complejidad del problema que se esté considerando.

Para determinar el número de coaliciones factibles de un tamaño y peso prefijados considérese la situación de comunicación (N, v, G) , siendo $G = (N, E)$ un árbol y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. El procedimiento a seguir será análogo al dado por Jamison para la obtención de los números de Whitney.

Así, para obtenerlos se considera la función generatriz

$$\Psi_G(x, z) = \sum_{k \geq 0} \sum_{j \in N} A_{kj}(G) x^k z^j,$$

donde, cada coeficiente $A_{kj}(G)$ representa el número de coaliciones factibles S , con $|S| = j$, y cuyo peso es $w(S) = k$. Por otra parte, la función $\Psi_G(x, z)$ será determinada por medio de las funciones generatrices

$$\psi_G(p; x, z) = \sum_{k \geq 0} \sum_{j \in N} c_{kj}(G; p) x^k z^j,$$

donde p es cualquier nodo de G , y los coeficientes $c_{kj}(G; p)$ del polinomio indican el número de coaliciones conexas que contienen al nodo p con un peso y tamaño determinados. Por simplicidad en la notación, fijado un árbol G , se escribirá $c_{kj}^p = c_{kj}(G; p)$ y $\psi_G(i|p; x, z)$ representará a la función generatriz $\psi_D(i; x, z)$, con $D = (D(i; G_p), E(D(i; G_p)))$. Además, al igual que en el anterior proceso estudiado para los números de Whitney, se suprimirán los subíndices siempre que no haya lugar a confusión.

En esta situación, el cálculo de las funciones generatrices $\Psi_G(x, z)$ y $\psi_G(p; x, z)$ es como sigue.

Teorema 5.21 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. Si p es un nodo cualquiera del árbol G , entonces*

- a) $\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z))$, donde el producto recorre todos los hijos i del nodo p en G .
- b) $\Psi_G(x, z) = \sum \psi(i|p; x, z)$, donde la suma recorre todos los nodos i de G .

Demostración: a) Sea $\{w_1, \dots, w_n\}$ el conjunto de los pesos de todos los jugadores, y considérese la función

$$\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z)).$$

Como esta recursión es similar a la utilizada en el apartado a) del Teorema 5.17 (obsérvese que, para $x = 1$, $\psi(p; 1, z) = \varphi(p; z)$), se tiene que

$$\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z)) = \sum_{\{S \in \mathcal{L}: p \in S\}} \prod_{i \in S} x^{w_i} z,$$

siendo \mathcal{L} el conjunto de coaliciones factibles o conexas correspondiente al árbol G . De ahí,

$$\psi(p; x, z) = \sum_{\{S \in \mathcal{L}: p \in S\}} (x^{\sum_{i \in S} w_i} z^{|S|}) = \sum_{\{S \in \mathcal{L}: p \in S\}} x^{w(S)} z^{|S|}.$$

Agrupando las potencias, resulta

$$\psi(p; x, z) = \sum_{k \geq 0} \sum_{j \in N} c_{kj}^p x^k z^j,$$

con $|S| = j$ y $w(S) = k$.

Teniendo en cuenta que el coeficiente c_{kj}^p indica el número de coaliciones factibles que incluyen al jugador p , con peso k y tamaño j , se deduce que la función $\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z))$ es una función generatriz para los números $\{c_{kj}^p\}_{k \geq 0, j \geq 0}$.

b) Es consecuencia inmediata del apartado b) del Teorema 5.17.

□

Es obvio que el teorema anterior proporciona un algoritmo para determinar, para cada j y k , el número $A_{kj}(G)$ de coaliciones factibles S , con $|S| = j$, y cuyo peso es $w(S) = k$. Puesto que el proceso a establecer es similar al del cálculo de los números de Whitney, se procede directamente a una ejemplificación del mismo.

Ejemplo 5.22 Sea (N, v, G) una situación de comunicación en la que G es el árbol definido en el Ejemplo 5.18 y (N, v) , con $v = [q; 3, 2, 2, 1, 1]$, un juego simple de votación ponderada.

Para determinar de forma recursiva los números $\{A_{kj}(G)\}_{k \geq 0, j \geq 0}$ se considera el árbol enraizado en un vértice cualquiera. En este caso —véase la Figura 5.3— se ha elegido el vértice 1. A continuación, en un proceso de recursivo, se calculan las funciones generatrices $\psi(i|1; x, z)$ para todos los vértices.

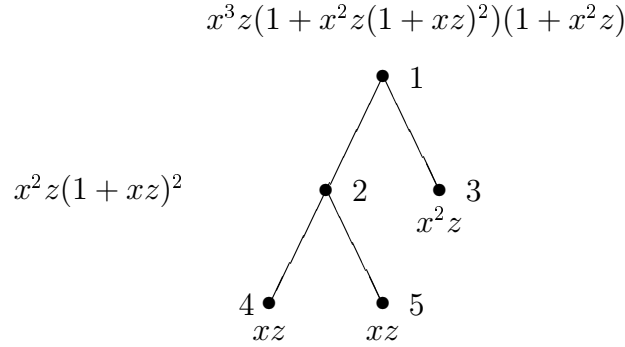


FIGURA 5.3

$$\begin{aligned}
\psi(1; x, z) &= x^3 z(1 + \psi(2|1; x, z))(1 + \psi(3|1; x, z)) \\
&= x^3 z(1 + x^2 z(1 + \psi(4|1; x, z))(1 + \psi(5|1; x, z)))(1 + x^2 z) \\
&= x^3 z(1 + x^2 z(1 + xz)^2)(1 + x^2 z), \\
&= x^3 z(1 + x^2 z + 2x^3 z^2 + x^4 z^3)(1 + x^2 z) \\
&= (x^3 z + x^5 z^2 + 2x^6 z^3 + x^7 z^4)(1 + x^2 z) \\
&= x^3 z + x^5 z^2 + 2x^6 z^3 + x^7 z^4 + x^5 z^2 + x^7 z^3 + 2x^8 z^4 + x^9 z^5 \\
&= x^3 z + 2x^5 z^2 + 2x^6 z^3 + x^7 z^3 + x^7 z^4 + 2x^8 z^4 + x^9 z^5, \\
\psi(2|1; x, z) &= x^2 z(1 + \psi(4|1; x, z))(1 + \psi(5|1; x, z)) \\
&= x^2 z(1 + xz)^2 \\
&= x^2 z(1 + 2xz + x^2 z^2) \\
&= x^2 z + 2x^3 z^2 + x^4 z^3, \\
\psi(3|1; x, z) &= x^2 z, \\
\psi(4|1; x, z) &= xz, \\
\psi(5|1; x, z) &= xz.
\end{aligned}$$

Obsérvese que cada función $\psi(i|1; x, z)$ determina los números $\{c_{kj}^{i|1}\}_{k \geq 0, j \geq 0}$ que corresponden al subárbol de G_1 que tiene como raíz al nodo i .

Como consecuencia del Teorema 5.21, para obtener la función $\Psi_G(x, z)$ y, como consecuencia, los números $\{A_{kj}(G)\}_{k \geq 0, j \geq 0}$ bastará con efectuar la suma de todas funciones ψ extendida a todos los vértices del árbol con raíz G_1 .

$$\begin{aligned}\Psi_G(x, z) &= \sum_{i=1}^n \psi(i|p; x, z) \\ &= x^3z + 2x^5z^2 + 2x^6z^3 + x^7z^3 + x^7z^4 + 2x^8z^4 + x^9z^5 \\ &\quad + 2x^2z + 2x^3z^2 + x^4z^3 + 2xz.\end{aligned}$$

Recuérdese que el sistema de coaliciones factibles asociado al árbol G viene dado por

$$\begin{aligned}\mathcal{L} = \{ &\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{1, 3\}, \{1, 2\}, \{2, 4\}, \{2, 5\}, \{1, 2, 3\}, \{1, 2, 4\}, \\ &\{1, 2, 5\}, \{2, 4, 5\}, \{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 4, 5\}, \{1, 2, 3, 4, 5\}\},\end{aligned}$$

y puede comprobarse, con la tabla que se expone a continuación, que el número de coaliciones $T \in \mathcal{L}$, con $|T| = j$ y peso $w(T) = k$, se corresponde con el coeficiente $A_{kj}(G)$ del polinomio $\Psi_G(x, z) = \sum_{k \geq 0} \sum_{j \in N} A_{kj}(G)x^k z^j$.

Coalición	Peso	Tamaño
$\{1\}$	3	1
$\{2\}$	2	1
$\{3\}$	2	1
$\{4\}$	1	1
$\{5\}$	1	1
$\{1, 2\}$	5	2
$\{1, 3\}$	5	2
$\{2, 4\}$	3	2
$\{2, 5\}$	3	2

Coalición	Peso	Tamaño
$\{1, 2, 3\}$	7	3
$\{1, 2, 4\}$	6	3
$\{1, 2, 5\}$	6	3
$\{2, 4, 5\}$	4	3
$\{1, 2, 3, 4\}$	8	4
$\{1, 2, 3, 5\}$	8	4
$\{1, 2, 4, 5\}$	7	4
$\{1, 2, 3, 4, 5\}$	9	5

Si se deseara conocer el número de coaliciones conexas con un peso y tamaño determinados, a las que pertenece un jugador dado, ha de calcularse, para cada jugador p , la función generatriz $\psi(p; x, z)$, considerando, en cada caso, el árbol G enraizado en dicho nodo p .

Ejemplo 5.23 *Considérese la situación de comunicación (N, v, G) del ejemplo anterior. El cálculo de los números $\{c_{kj}^i\}_{k \geq 0, j \geq 0}$, con $i \in N$, es como sigue:*

- 1) *Para obtener los números $\{c_{kj}^1\}_{k \geq 0, j \geq 0}$, se considera el árbol enraizado en el nodo 1 y, al igual que en el ejemplo anterior, se obtiene*

$$\begin{aligned}\psi(1; x, z) &= x^3 z (1 + \psi(2; x, z)) (1 + \psi(3; x, z)) \\ &= x^3 z + 2x^5 z^2 + 2x^6 z^3 + x^7 z^3 + x^7 z^4 + 2x^8 z^4 + x^9 z^5.\end{aligned}$$

- 2) *Para el nodo 2 se procede de forma análoga, considerando, ahora, el árbol G enraizado en dicho nodo.*

$$\begin{aligned}\psi(2; x, z) &= x^2 z (1 + \psi(1|2; x, z)) (1 + \psi(4|2; x, z)) (1 + \psi(5|2; x, z)) \\ &= x^2 z (1 + x^3 z (1 + \psi(3|2; x, z))) (1 + xz)^2 \\ &= x^2 z (1 + x^3 z (1 + x^2 z)) (1 + xz)^2 \\ &= x^2 z + 2x^3 z^2 + x^5 z^2 + x^4 z^3 + 2x^6 z^3 \\ &\quad + x^7 z^3 + x^7 z^4 + 2x^8 z^4 + x^9 z^5, \\ \psi(1|2; x, z) &= x^3 z (1 + \psi(3|2; x, z)) \\ &= x^3 z (1 + x^2 z) \\ &= x^3 z + x^5 z^2, \\ \psi(3|2; x, z) &= x^2 z, \\ \psi(4|2; x, z) &= xz, \\ \psi(5|2; x, z) &= xz.\end{aligned}$$

Se obtiene, por tanto, que los números $\{c_{kj}^2\}_{k \geq 0, j \geq 0}$ son los coeficientes del polinomio

$$\psi(2; x, z) = x^3 z + 2x^5 z^2 + 2x^6 z^3 + x^7 z^3 + x^7 z^4 + 2x^8 z^4 + x^9 z^5.$$

3) Reiterando el proceso para los nodos restantes se obtiene los números $\{c_{kj}^i\}_{k \geq 0, j \geq 0}$, con $i \in \{3, 4, 5\}$. En cada caso habrá que considerar el árbol G enraizado en el nodo correspondiente. El resultado obtenido es el que sigue.

$$\begin{aligned} \psi(3; x, z) &= x^2 z(1 + x^3 z(1 + x^2 z(1 + xz)^2)) \\ &= x^2 z + x^5 z^2 + x^7 z^3 + x^7 z^4 + 2x^8 z^4 + x^9 z^5. \end{aligned}$$

$$\begin{aligned} \psi(4; x, z) &= xz(1 + x^2 z(1 + xz)(1 + x^3 z(1 + x^2 z))) \\ &= xz + x^3 z^2 + x^4 z^3 + x^6 z^3 + x^7 z^4 + x^8 z^4 + x^9 z^5. \end{aligned}$$

$$\begin{aligned} \psi(5; x, z) &= xz(1 + x^2 z(1 + xz)(1 + x^3 z(1 + x^2 z))) \\ &= xz + x^3 z^2 + x^4 z^3 + x^6 z^3 + x^7 z^4 + x^8 z^4 + x^9 z^5. \end{aligned}$$

Nótese, teniendo en cuenta la implementación del algoritmo correspondiente, que todos los valores $\psi(i|p; x, z)$, con $i \neq p$, una vez determinados y utilizados, pueden eliminarse en su almacenamiento de memoria por ser valores que corresponden a subárboles y, en consecuencia, no tendrán ningún interés posterior.

Dado que este procedimiento recursivo, para calcular las funciones generatrices $\psi(p; x, z)$, es similar al descrito para obtener las funciones $\varphi(p; z)$, se establecen, de manera inmediata, los siguientes resultados.

Proposición 5.24 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. Si p es un*

nodo cualquiera del árbol G , entonces la obtención de la función $\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z))$, donde el producto recorre todos los hijos i de p en G , requiere un tiempo $\mathcal{O}(r^h)$, donde h es la altura y r el factor de ramificación del árbol con raíz G_p .

Proposición 5.25 *Sea la terna (N, v, G) , donde G es un árbol y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego simple de votación ponderada. Si p es un nodo cualquiera del árbol G , entonces la expansión del polinomio $\psi(p; x, z) = x^{w_p} z \prod (1 + \psi(i|p; x, z))$, donde el producto recorre todos los hijos i de p en G , requiere un tiempo $\mathcal{O}(r^h)$, donde h es la altura y r el factor de ramificación del árbol con raíz G_p .*

Una vez que se ha realizado un estudio, utilizando funciones generatrices, sobre la enumeración de subgrafos conexos en árboles y de las coaliciones conexas de un tamaño y peso determinados es lógico que, en la última parte de la sección, se trate la complejidad temporal del algoritmo, para el cálculo del valor de Myerson, que surge de incorporar las técnicas de enumeración estudiadas a los resultados correspondientes de los Teoremas 5.14 y 5.15. De ahí que, en la siguiente página, se presente la función *MyersonStar* que permitirá el cálculo del valor de Myerson en situaciones de comunicación (N, v, G) en las que G es un grafo estrella y (N, v) es un juego simple de votación ponderada.

Función *MyersonStar*(*pesos*, *q*)

{*pesos*: lista de n enteros; q : entero}

para l **de** 1 **hasta** $n - 1$
 $\psi(l; x, z) \leftarrow x^{w_l} z \prod (1 + \psi(i|l; x, z))$
 $\left\{ \psi(l; x, z) = \sum_{k \geq 0} \sum_{j \in N} c_{kj}^l x^k z^j \right\}$
 $\mu_l \leftarrow \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{q-1+w_l} c_{kj}^l \right)$
fin_para
 $\psi(n; x, z) \leftarrow x^{w_n} z \prod_{j=1}^{n-1} (1 + x^{w_j} z)$
 $\left\{ \psi(n; x, z) = \sum_{k \geq 0} \sum_{j \in N} c_{kj}^n x^k z^j \right\}$
 $\mu_n \leftarrow \sum_{j=2}^n \frac{(j-1)!(n-j)!}{n!} \left(\sum_{k=q}^{w(N)} c_{kj}^n \right)$

devolver $\{\mu_1, \dots, \mu_n\}$

Proposición 5.26 *Sea la terna (N, v, G) , donde G es un grafo estrella y (N, v) , con $v = [q; w_1, \dots, w_n]$, un juego de votación ponderada. El cálculo del valor de Myerson de todos los jugadores con la función *MyersonStar* requiere un tiempo $\mathcal{O}(n^2c)$, siendo c el número de coeficientes no nulos del polinomio $\Psi_G(x, z)$.*

Demostración: Sea $f(n)$ la función que determina el tiempo de ejecución de la función *MyersonStar*. Entonces, se verifica

$$\mathcal{O}(f(n)) = \mathcal{O}(t(\text{bucle}) + t(\text{asignación3}) + t(\text{asignación4}))$$

y, además,

$$\mathcal{O}(t(\text{bucle})) = \mathcal{O}((n-1)(t(\text{asignación1}) + t(\text{asignación2}))).$$

Ahora bien, por las Proposiciones 5.24 y 5.25, se verifica que

$$\mathcal{O}(t(\text{asignación1})) = \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(r^h),$$

donde r es el factor de ramificación y h la altura del grafo estrella. Como, en el bucle, el árbol se enraíza en un nodo que no es el central, entonces $r = n - 2$ y $h = 2$. Por tanto,

$$\mathcal{O}(t(\text{asignación1})) = \mathcal{O}(n^2).$$

Por otra parte,

$$\mathcal{O}(t(\text{asignación2})) = \mathcal{O}((n-1)(t(\text{coefi}) + t(\text{suma1}))),$$

teniéndose que $\mathcal{O}(t(\text{coefi})) = \mathcal{O}(n)$, pues los factoriales se calculan en tiempo lineal, y $\mathcal{O}(t(\text{suma1})) = \mathcal{O}(c)$, donde c es el número de coeficientes no nulos del polinomio $\Psi_G(x, z)$ y constituye una cota superior del número de coeficientes no nulos de los polinomios $\psi(p; x, z)$, con $p \in N$. Por tanto, al ser $c \geq n$,

$$\mathcal{O}(t(\text{asignación2})) = \mathcal{O}(n^2 + nc) = \mathcal{O}(nc).$$

De lo anterior se deduce que el tiempo de ejecución del bucle es

$$\begin{aligned} \mathcal{O}(t(\text{bucle})) &= \mathcal{O}((n-1)t(\text{asignación1}) + nt(\text{asignación2})) \\ &= \mathcal{O}(n^3 + n^2c) \\ &= \mathcal{O}(n^2c). \end{aligned}$$

Las asignaciones restantes, que se refieren al nodo central, verifican, por un lado,

$$\mathcal{O}(t(\text{asignación3})) = \mathcal{O}(t(\text{polinomio})) = \mathcal{O}(r^h) = \mathcal{O}(n),$$

ya que, para el vértice central, $r = n - 1$ y $h = 1$. Por otro lado,

$$\begin{aligned}\mathcal{O}(t(\text{asignación4})) &= \mathcal{O}(n(t(\text{coefi}) + t(\text{suma2}))) \\ &= \mathcal{O}(n^2 + nc) \\ &= \mathcal{O}(nc).\end{aligned}$$

Finalmente, resulta $\mathcal{O}(f(n)) = \mathcal{O}(\max(n^2c, n, nc)) = \mathcal{O}(n^2c)$.

□

5.3 Algoritmos con Mathematica

En esta última sección se implementan los algoritmos descritos en las secciones anteriores particularizándolos en tres contextos concretos. En el primero de ellos, se trata la función *geomdiv* para calcular el valor de Myerson generalizado mediante los dividendos de Harsanyi y utilizando funciones generatrices. El segundo ejemplo muestra la determinación de los números de Whitney, y en el último ejemplo se realiza el cálculo del valor de Myerson en una situación de comunicación en la que el grafo, que modela la cooperación entre los jugadores, es un grafo estrella y se ha considerado un juego simple de votación ponderada.

Ejemplo 5.27 *Sea el juego de votación correspondiente a la composición del Congreso de los Diputados de España resultante de las elecciones de 1996, y considerando únicamente los seis grandes grupos.*

En esta situación, los jugadores, con expresión del número de votos, son los partidos políticos:

Jugador 1: PP, 156 votos.

Jugador 2: CC, 4 votos.

Jugador 3: CIU, 16 votos.

Jugador 4: PNV, 5 votos.

Jugador 5: IU, 21 votos.

Jugador 6: PSOE, 141 votos.

Este juego simple de votación ponderada se representa por

$$v = [q; 156, 4, 16, 5, 21, 141], \text{ siendo } q = 176.$$

Supóngase, además, que no todas las coaliciones son posibles y que la colaboración entre estos grupos viene restringida por el grafo de la Figura 5.1 con lo que, teniendo en cuenta que se trata de un grafo bloque conexo, el par (N, \mathcal{L}) , siendo \mathcal{L} la familia de coaliciones conexas, constituye una geometría convexa de partición.

A continuación, se indica, usando MATHEMATICA, el proceso de cálculo del valor de Myerson generalizado. Previamente, para calcular los dividendos de Harsanyi del correspondiente juego restringido, se emplea el algoritmo geomdiv.

```
In [1] :=
  <<DiscreteMath'Combinatorica'
```

```
In [2] :=
  <<Cooperat'Cooperat'
```

```
In [3] :=
  Congreso:=(Clear[T,w,q];T=Range[6];
  q=176;
  w[1]:=151;
  w[2]:=4;
```

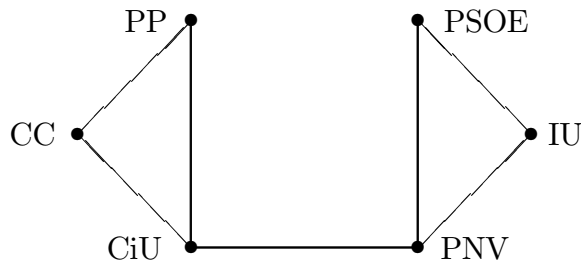


```
w[3]:=16;
w[4]:=5;
w[5]:=21;
w[6]:=141;
w[S_List]:=Apply[Plus, w/@S];)
```

```
In[4] :=
grafo=FromUnorderedPairs[{{1,2},{1,3},{2,3},{3,4}, {4,5},{4,6},{5,6}}];
grafo[[2]]={{0,0},{-1,-1},{0,-2},{2,-2},{3,-1}, {2,0}};
```

```
In[5] :=
ShowLabeledGraph[grafo,
{"PP","CC"," CiU","PNV","IU","PSOE "};
```

Out[5] =



```
In[6] :=
Clear[F];
F=Sort[Select[Rest[Subsets[Range[V[grafo]]]],
(ConnectedQ[InduceSubgraph[grafo,#]])&]];
PrependTo[F,{}]
```

Out[6] =

```

{{}, {1}, {2}, {3}, {4}, {5}, {6}, {1, 2}, {1, 3},
{2, 3}, {3, 4}, {4, 5}, {4, 6}, {5, 6}, {1, 2, 3},
{1, 3, 4}, {2, 3, 4}, {3, 4, 5}, {3, 4, 6}, {4, 5, 6},
```

```
{1, 2, 3, 4}, {1, 3, 4, 5}, {1, 3, 4, 6}, {2, 3, 4, 5},
{2, 3, 4, 6}, {3, 4, 5, 6}, {1, 2, 3, 4, 5}, {1, 2, 3, 4, 6},
{1, 3, 4, 5, 6}, {2, 3, 4, 5, 6}, {1, 2, 3, 4, 5, 6}}
```

In[7] :=

```
ex[S_List, F_List] := ex[S, F] = Select[S,
(MemberQ[F, Complement[S, {#}]) &]
```

In[8] :=

```
Clear[dividen];
dividen[S_] := dividen[S] =
Module[{A, A1, A2, P, Cs, Cs1},
A = w /@ ex[S, F]; A1 = Complement[S, ex[S, F]];
A2 = w[A1];
P = Apply[Times, 1 + z * x^A] * z^Length[A1] * x^A2;
Cs = (-1)^Length[S] * Apply[Plus, CoefficientList[
Expand[P], x][[Range[q + 1, w[S] + 1]]]];
Cs1 = ReplaceAll[B, z: > -1]; Return[Cs1];
ex[#, F] =. & /@ F]
```

La función GeomValue, la cual se describe en el siguiente párrafo, permite calcular el valor de Myerson generalizado de todos los jugadores y tiene una sintaxis similar a la de la función ShapleyValue4 con la salvedad de que sólo considera a las coaliciones factibles.

In[9] :=

```
GeomValue[Game_ : Null] := Module[{value},
value = Table[Apply[Plus,
(dividen[#]/Length[#]) & /@ Select[F,
(MemberQ[#, i] &)], {i, Length[T]}];
dividen[#] =. & /@ Rest[F]; Return[value];
```

```
In[10] :=
Timing[GeomValue[Congreso]]
```

```
Out[10] =
{0.822 Second, { $\frac{13}{60}, \frac{1}{12}, \frac{7}{15}, \frac{2}{15}, \frac{1}{20}, \frac{1}{20}$ }}
```

Ejemplo 5.28 *Considérese el árbol definido en el Ejemplo 5.18. La determinación de los números de Whitney usando MATHEMATICA es como sigue.*

```
In[1] :=
<<DiscreteMath'Combinatorica'
```

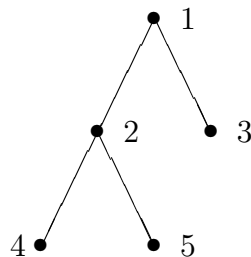
```
In[2] :=
<<Cooperat'Cooperat'
```

```
In[3] :=
grafo=FromUnorderedPairs[{{1,2},{1,3},{2,4},{2,5}}];
```

```
In[4] :=
grafo[[2]]={{0,0},{-1,-1},{1,-1},{-2,-2},{0,-2}};
```

```
In[5] :=
ShowLabeledGraph[grafo];
```

```
Out[5] =
```



A continuación se implementa la función $\varphi(p; z)$, denotada por f . Para ello, se utiliza programación dinámica con objeto de almacenar en una tabla $\varphi(p; z)$ junto con las funciones $\varphi(i|p; z)$.

```
In[6] :=
  Clear[f]
  arcos=aristas;
  f[{}]=0;
  f[{i_}]:=f[{i}]=Module[{hijos, factores},
    hijos=Complement[Flatten[Select[arcos, (MemberQ[#, i])&]], {i}];
    arcos=Complement[arcos, Select[arcos, (MemberQ[#, i])&]];
    factores=(1+f[#])& /@ Partition[hijos, 1];
    z*Times @@ factores]
```

Para calcular $\Phi_G(z)$ se considera, tal como se expuso en la sección anterior, que el árbol se enraíza en un nodo cualquiera. En este caso se elige el nodo 1 y, por dicho motivo, se calcula $\varphi(1; z)$.

```
In[7] :=f[{1}]
```

```
Out[7] =
  z(1 + z)(1 + z(1 + z)^2)
```

Como consecuencia del proceso de cálculo, el valor $\varphi(1; z)$ queda almacenado junto con los valores $\varphi(i|1; z)$, para $i \in \{2, 3, 4, 5, \dots\}$. Este hecho se pone de manifiesto solicitando ayuda al programa.

```
In[8] :=??f
```

```
Out[8] =
  f[{}] = 0
  f[{1}] = z*(1 + z)*(1 + z*(1 + z)^2)
  f[{2}] = z*(1 + z)^2
```

```

f[{3}] = z
f[{4}] = z
f[{5}] = z
f[{i_}] :=
f[{i}] = Module[{hijos, factores},
hijos = Complement[Flatten[Select[arcos, MemberQ[#1, i] & ]], {i}];
arcos = Complement[arcos, Select[arcos, MemberQ[#1, i] & ]];
factores = (1 + f[#1] & ) /@ Partition[hijos, 1]; z*Times @@ factores]

```

En las siguientes tablas se muestran los valores obtenidos para los diversos polinomios almacenados. Aparecen los resultados, correspondientes a cada nodo, sin expandir y en forma expandida.

```

In[9] :=
TableForm[Transpose[{Range[5], Table[f[{i}], {i, 1, 5}]}]]

```

```

Out[9]//TableForm =
1      z(1 + z)(1 + z(1 + z)2)
2      z(1 + z)2
3      z
4      z
5      z

```

```

In[10] :=
TableForm[Transpose[{Range[5], Expand[Table[f[{i}], {i, 1, 5}]}]}]]

```

```

Out[10]//TableForm =
1      z + 2z2 + 3z3 + 3z4 + z5
2      z + 2z2 + z3
3      z
4      z
5      z

```

El polinomio $\Phi_G(x, z)$ se obtiene sumando todos los polinomios φ almacenados.

```
In[11] :=
  Polinomio=Apply[Plus,Expand[Table[f[{i}],{i,1,5}]]]
```

```
Out[11] =
  5z+4z2+4z3+3z4+z5
```

Por último, en la tabla siguiente, aparecen determinados los números de Whitney.

```
In[12] :=
  TableHeadings->{{" Coaliciones" }, {" Tamaño" , " Número" }}}
```

```
Out[12]//TableForm =
```

	Tamaño	Número
	1	5
Coaliciones	2	4
	3	4
	4	3
	5	1

Ejemplo 5.29 Sea el juego de votación ponderada, descrito en el Ejemplo 2.11, que corresponde al sistema de votación en el Consejo de Gobierno de la Unión Europea. Además, considérese el juego restringido por la situación de comunicación (N, v, G) , donde G es un grafo estrella cuyo nodo central, denotado por n , corresponde a Alemania.

Como consecuencia de esta elección, el conjunto de jugadores se ordena del siguiente modo

$N = \{ \text{Gran Bretaña (GB), Francia (FR), Italia (IT), España (ES), Países Bajos (PB), Grecia (GR), Bélgica (BE), Portugal (PO), Suecia (SU), Aus-}$

tria (AU), Dinamarca (DI), Finlandia (FI), Irlanda (IR), Luxemburgo (LU), Alemania (AL)},

y, para esta ordenación de los jugadores, el juego (N, v) queda caracterizado mediante

$$v = [q ; 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2, 10], \quad 61 \leq q \leq 68.$$

A continuación, se describe, empleando MATHEMATICA, el proceso de cálculo del valor de Myerson utilizando los resultados de la Sección 5.3.

In[1] :=

<<DiscreteMath'Combinatorica'

In[2] :=

<<Cooperat'Cooperat'

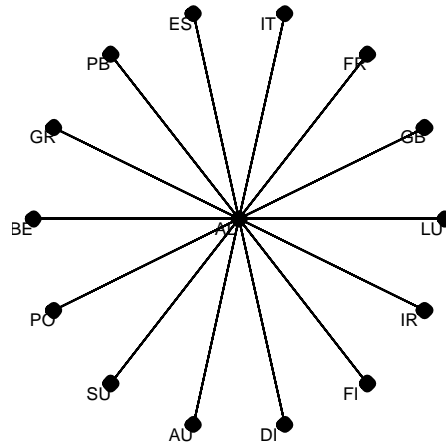
In[3] :=Clear[n, grafo, aristas, pesos, q, f];

In[4] :=

n=15; grafo=Star[n];

In[5] :=

ShowLabeledGraph[grafo, {"GB", "FR", "IT", "ES", "PB", "GR",
"BE", "PO", "SU", "AU", "DI", "FI", "IR", "LU", "AL"}];



```
In[6] :=
  aristas=Flatten[Prepend[{n},#]]&/@Partition[Range[n-1],1];
```

```
In[7] :=
  pesos={10,10,10,10,8,5,5,5,5,4,4,3,3,2,10}
```

A continuación se implementan la función $\psi(p; x, z)$, denotada por f , y las fórmulas que permiten determinar el valor de Myerson de los jugadores.

```
In[8] :=
  f[{i_}, pesos_List, aristas_List]:=
  Module[{A, hijos, factores, value, arcos},
  arcos=aristas;
  A=Select[arcos, (MemberQ[#, i])&];
  hijos=Complement[Flatten[A], {i}];
  arcos=Complement[arcos, A];
  factores=(1+f[#, pesos, arcos])& /@ Partition[hijos, 1];
  value=z*x^(pesos[[i]])*Times @@ factores;
  factores=. & /@ Partition[hijos, 1];
  Return[value]];
```

```
In[9] :=
```



```

MyersonStarPlayer[i_, pesos_List, q_] :=
Module[{m, g, gg, coefi},
g = Expand[f[{i}], pesos, aristas];
coefi = CoefficientList[g, x]; m = Exponent[g, x] + 1;
gg = Apply[Plus, coefi[[Range[q + 1, Min[m, q + pesos[[i]]]]]]];
Sum[Coefficient[gg, z^j] (j - 1)! (n - j)!, {j, 2, n}] / n!

```

```

In[10] :=
MyersonStarCenter[pesos_List, q_] :=
Module[{m, g, gg, coefi},
g = Expand[f[{n}], pesos, aristas];
coefi = CoefficientList[g, x]; m = Exponent[g, x] + 1;
gg = Apply[Plus, coefi[[Range[q + 1, m]]]];
Sum[Coefficient[gg, z^j] (j - 1)! (n - j)!, {j, 2, n}] / n!

```

```

In[11] :=
MyersonStarExtreme[pesos_List, q_] := Module[{value},
value = Table[MyersonStarPlayer[i, pesos, q], {i, n - 1}];
Return[value]

```

```

In[12] :=
MyersonStar[pesos_List, q_] :=
Join[MyersonStarExtreme[pesos, q],
{MyersonStarCenter[pesos, q]}]

```

```

In[13] :=
Timing[MyersonStar[pesos, 62]] // N

```

```

Out [13] =
{2.85 Second, {0.0882839, 0.0882839, 0.0882839,
0.0708236, 0.0406288, 0.0406288, 0.0406288,
0.0406288, 0.0328172, 0.0328172, 0.0251637,
0.0251637, 0.0251637, 0.0146964, 0.345987}}

```

Finalmente, con el objeto de hacer una comparación, se indica como se evalúa el valor de Myerson para este mismo juego cuando se utiliza la función *MyersonValue* descrita en el tercer capítulo.

Puesto que la función *MyersonValue* implementa el algoritmo dividiendo es preciso determinar todas las coaliciones conexas habida cuenta que hay que calcular los dividendos de todas ellas.

Si se considera el grafo estrella con n nodos, y se supone que el nodo n es el nodo central, las coaliciones factibles pueden determinarse de manera directa, tal y como se indicó en el Capítulo 3, a partir de la expresión

$$\mathcal{F} = \{\{n\} \cup S : S \in 2^{N \setminus \{n\}}\} \cup \{\{1\}, \dots, \{n-1\}\} \cup \{\emptyset\}.$$

Usando MATHEMATICA, la familia \mathcal{F} se implementa del siguiente modo

```
In[ ] :=
F=Union[Flatten[Prepend[{n},#]]&/@Union[
Subsets[Range[n-1]],Partition[Range[n-1],1]],{{}}];
```

Procediendo de forma similar a la descrita en el Ejemplo 3.73, se requiere un tiempo de 2845 segundos para calcular el valor de Myerson de todos los jugadores, con la función *MyersonValue*.

Evidentemente la función *MyersonStar* mejora considerablemente el tiempo de ejecución ya que éste no llega a tres segundos. Además, la utilización de dicha función requiere, únicamente, introducir los pesos y las aristas del árbol, y no hay que determinar ni las coaliciones factibles ni la función característica del juego.

Los tiempos de ejecución obtenidos, al calcular el valor de Myerson con ambos algoritmos, son coherentes con los resultados establecidos para la complejidad de los mismos en las Proposiciones 3.25 y 5.26.

Referencias

- [1] AHO, A.V., HOPCROFT, J.E., ULLMAN, J.D. (1983) *Estructuras de datos y algoritmos*, Addison-Wesley Iberoamericana, S.A., Wilmington, Delaware.
- [2] ALGABA, E. (1998) Extensión de juegos definidos en sistemas de conjuntos, *Tesis Doctoral*, Universidad de Sevilla.
- [3] ALGABA, E., BILBAO, J.M., BORM, P., LÓPEZ, J.J. (1998) The Myerson value for union stable systems. Research Memorandum FEW 773, Tilburg University.
- [4] ALGABA, E., BILBAO J.M., BORM, P., LÓPEZ, J.J. (2000) The position value for union stable systems. Aparecerá en *Mathematical Methods of Operations Research*, Vol. 52, Issue 3.
- [5] AUMANN, R., MASCHLER, M. (1964) The bargaining set for cooperative games, *Advances in Game Theory*, M. Dresher, L. Shapley, A. Tucker (eds.), Princeton University Press, Princeton, 443–471.
- [6] AUMANN, R., MYERSON, R. (1988) Endogenous formation of links between players and coalitions: an application of the Shapley value, *The Shapley value*, A. Roth (ed.), Cambridge University Press, Cambridge, 175–191.
- [7] AXELROD, R. (1970) *Conflict of Interest*, Markham, Chicago.

- [8] BANZHAF, J.F. III (1965) Weighted Voting doesn't Work Mathematical Analysis, *Rutgers Law Review* 19, 317–343.
- [9] BERGANTIÑOS, G., CARRERAS, F., GARCÍA-JURADO, I. (1993) Cooperation when Some Players are Incompatible, *ZOR-Methods and Models of Operations Research* 38, 187–201.
- [10] BILBAO, J.M. (1998) Values and potential of games with cooperation structure, *International Journal of Game Theory* 27, 131–145.
- [11] BILBAO, J.M. (1998) Closure spaces and restricted games, *Mathematical Methods of Operations Research* 48 (1), 57–69.
- [12] BILBAO, J.M., LÓPEZ, J.J. (1996) El potencial de Hart y Mas-Colell para juegos restringidos por grafos, *Quaderns d'Estadística i Investigació Operativa* 20 (1), 71–86.
- [13] BILBAO, J.M., FERNÁNDEZ, F.R. (eds.) (1999) *Avances en teoría de juegos con aplicaciones económicas y sociales*. Servicio de Publicaciones de la Universidad de Sevilla.
- [14] BIRKHOFF, G., BENNETT, M.K. (1985). The Convexity Lattice of a Poset, *Order* 2, 223-242.
- [15] BONDAREVA, O.(1963) Certain Applications of the Methods of Linear Programming to the Theory of Cooperative Games, *Problemy kibernetiki* 10, 119-139.
- [16] BORM, P. (1990) On game theoretic models and solution concepts, *Ph.D. Thesis*, University of Nijmegen, The Netherlands.
- [17] BORM, P., VAN DE NOUWELAND, A., OWEN, G., TIJS, S. (1993) Cost Allocation and Communication, *Naval Research Logistics* 40, 733–744.

-
- [18] BORM, P., OWEN, G., TIJS, S. (1992) On the position value for communication situations, *SIAM J. Disc. Math.* 5, 305–320.
- [19] BORM, P., VAN DEN NOUWELAND, A., TIJS, S. (1992) Allocation Rules for Hypergraph Communication Situations, *International Journal of Game Theory* 20, 255–268.
- [20] BORM, P., VAN DEN NOUWELAND, A., TIJS, S. (1994) Cooperation and Communication restrictions: a survey, *Imperfections and Behavior in Economic Organizations*, R.P. Gilles, P. Ruys (eds.), Kluwer Academic Publishers, 195–227.
- [21] BRAMS, S.F., AFFUSO, P.J. (1976). Power and Size: A New Paradox, *Theory and Decision* 7, 29–56.
- [22] BRAMS, S.J., LUCAS, W.F., STRAFFIN, P.D. (eds.) (1983) *Political and Related Models*, Springer-Verlag, New York.
- [23] BRASSARD, G., BRATLEY, P. (1997) *Fundamentos de Algoritmia*, Prentice Hall, Madrid.
- [24] CALVO, E., LASAGA, J. (1997) Probabilistic Graphs and Power Indices: an Application to the Spanish Parliament, *Journal of Theoretical Politics* 9 (4), 477–501.
- [25] CARRERAS, F. (1991) Restriction of Simple Games, *Mathematical Social Sciences* 21, 245–260.
- [26] CARTER, M. (1993) Cooperative games, en: *Economic and Financial Modeling with Mathematica* H. R. Varian (ed.), Springer-Verlag. Berlin, 167–191.
- [27] COLEMAN, J. (1971) Control of collectivities and the Power of a collectivity to act, *Social Choice*, B. Lieberman (ed.), Gordon and Breach, Londres, 269–300.

- [28] CORMEN, T.H., LEISERSON, C.E., RIVEST, R.L. (1996) *Introduction to Algorithms*, The MIT Press, Cambridge.
- [29] DRAGAN, I., DRIESSEN, T., FUNAKI Y. (1996) Collinearity between the Shapley value and the egalitarian division rules for cooperative games, *OR Spektrum* 18, 97-105.
- [30] EDELMAN, P.H., JAMISON, R.E. (1985) The theory of convex geometries, *Geometriae Dedicata*, 19, 247-270.
- [31] EDELMAN, P.H. (1997) A note on voting, *Mathematical Social Sciences*, 34, 37-50.
- [32] GILLIES, D.B. (1953) Some Theorems on n -Person Games, *Ph. D. Thesis*, Princeton University Press, Princeton, New Jersey.
- [33] GRAFE, F., MAULEON, A., IÑARRA, E. (1995) A Simple Procedure to Compute the Nucleolus of Γ -Component Additive Games, *TOP* 3 (2), 235-245.
- [34] HARARY, F. (1969) *Graph theory*, Addison-Wesley, Massachusettes.
- [35] HARSANYI, J.C. (1963) A Simplified Bargaining Model for the n -Person Cooperative Game, *International Economic Review* 4, 194-220.
- [36] HART, S., MAS-COLELL, A. (1988) The Potential of the Shapley value, *The Shapley value*, A. Roth (ed.), Cambridge University Press, Cambridge, 127-137.
- [37] HEILEMAN, G. (1998) *Estructuras de Datos, Algoritmos, Programación Orientada a Objetos*, McGraw-Hill. Madrid.
- [38] JAMISON, R.E. (1987) Alternating Whitney sums and matchings in trees. Part. 1, *Discrete Mathematics* 67, 177-189.
- [39] LÓPEZ, J.J. (1996) Cooperación parcial en juegos de n -personas, *Tesis Doctoral*, Universidad de Sevilla.

- [40] LUCAS, W.F. (1983) Measuring Power in Weighted Voting Systems, en: *Political and Related Models*, S. Brams (ed.), Springer-Verlag, New York.
- [41] MANN, I., SHAPLEY, L. S. (1962). Value of Large Games, VI: Evaluating the Electoral College Exactly, *RM-3158-PR, The Rand Corporation*, Santa Mónica, California.
- [42] MYERSON, R.B. (1977) Graphs and Cooperation in Games, *Mathematics of Operations Research* 2, 225–229.
- [43] MYERSON, R.B. (1980) Conference Structures and Fair Allocation Rules, *International Journal of Game Theory* 9, 169–182.
- [44] OWEN, G. (1986) Values of Graph-Restricted Games, *SIAM Journal of Algebraic and Discrete Methods* 7, 210–220.
- [45] POTTERS, J., REIJNIERSE, H. (1995) Γ -Component Additive Games, *International Journal of Game Theory* 24, 49–56.
- [46] REIJNIERSE, J. (1995) Games, Graphs and Algorithms, *Ph. D. Thesis*, Nijmegen University, The Netherlands.
- [47] RIBNIKOV, K. (1985) *Análisis Combinatorio*, Mir, Moscú.
- [48] ROSENTHAL, E.C. (1988) Communication networks and their role in cooperative games, *Social Networks* 10, 255–263.
- [49] ROSENTHAL, E.C. (1988) Communication and its costs in graph-restricted games, *Theory and Decision* 25, 275–286.
- [50] SCHRIJVER, A. (1986) *Theory of Linear and Integer Programming*, John Wiley & Sons, New York.
- [51] SHAPLEY, L.S. (1953) A Value for n -Person Games, *Contributions to the Theory of Games*, Vol. II, H.W. Kuhn, A.W. Tucker (eds.), Princeton, New Jersey, 307–317.

- [52] SHAPLEY, L.S. (1967) On Balanced sets and Cores, *Naval Research Logistics Quarterly* 14, 453-460.
- [53] SHAPLEY, L.S. (1971) Cores of convex games, *International Journal of Game Theory* 1, 11-26.
- [54] SHAPLEY, L.S., SHUBIK, M. (1954) A Method for Evaluating the Distribution of Power in a Committee System, *American Political Science Review* 48, 787-792.
- [55] SKIENA, S. (1990) *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*, Addison-Wesley, Redwood City.
- [56] STANLEY, R.P. (1986) *Enumerative Combinatorics*, Vol I, Wadsworth.
- [57] TANNENBAUM, P. (1997) Power in Weighted Voting Systems, *The Mathematica Journal* 7 (1), 58-63.
- [58] VAN DEN NOUWELAND, A., BORM, P. (1991) On the Convexity of Communication Games, *International Journal of Game Theory* 19, 421-430.
- [59] VAN DEN NOUWELAND, A. (1993) Games and Graphs in Economic Situations, *Ph. D. Thesis*, Tilburg University, The Netherlands.
- [60] VON NEUMANN, J. (1928) Zur Theorie der Gesellschaftsspiele, *Mathematische Annalen* 100, 295-320.
- [61] VON NEUMANN, J., MÖRGENSTERN, O. (1944) *Theory of Games and Economic Behavior*, Princeton University Press, Princeton, New Jersey.
- [62] VOSHTINA, O. (1997) Solution Concepts for some generalized cooperative game, *Ph. D. Thesis*, University of Texas at Arlington.

- [63] WILD, M. (1994) A theory of finite closure spaces based on implications, *Advances in Mathematics* 108, 118-139.
- [64] WOLFRAM, S. (1991) *Mathematica: A System for Doing Mathematics by Computer*, Addison-Wesley, Massachusetts.