UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

UNIVERSIDADE DA CORUÑA

Universidade de Vigo

Trabajo Fin de Máster

# Problemas de Bancarrota con Bienes Indivisibles. Un Caso Práctico

Estefanía Martínez Pintos

**Máster en Técnicas Estadísticas**

**Curso 2018-2019**

# Propuesta de Trabajo Fin de Máster

| |
|---|
| **Título en galego:** Problemas de Bancarrota con Bens Indivisibles. Un Caso Práctico |
| **Título en español:** Problemas de Bancarrota con Bienes Indivisibles. Un Caso Práctico |
| **English title:** Bankruptcy problems with indivisible assets. A practical case |
| **Modalidad:** A |
| **Autor/a:** Estefanía Martínez Pintos, Universidad de Santiago de Compostela |
| **Director/a:** Leticia Lorenzo Picado, Universidade de Vigo |
| **Breve resumen del trabajo:**<br>Un problema clásico de bancarrota surge cuando se quiere dividir el valor de liquidación de una empresa (el estado) entre sus acreedores (demandantes) y la demanda global es superior al capital disponible. En este problema clásico se asume que tanto el estado como las demandas son perfectamente divisibles y se buscan reglas de reparto que satisfagan propiedades deseables. En este trabajo se estudiarán los problemas de bancarrota donde tanto las demandas como el estado son bienes indivisibles, como podría ser el caso de las listas de espera en los hospitales o la asignación de slots de tiempo en los aeropuertos para las diferentes aerolíneas. Se realizará un breve resumen de la literatura disponible y se procederá a su aplicación a un caso práctico como puede ser el reparto de plazas entre las diferentes áreas de la universidad. |
| **Recomendaciones:** Haber cursado la materia Teoría de Juegos |

# Índice general

# Resumen


## Resumen en español

En este trabajo se realiza un estudio de los problemas clásicos de bancarrota. Representan una situación en la que una cantidad de un bien perfectamente divisible debe repartirse entre diferentes agentes. Cada uno de los agentes tiene una demanda, y la suma de las demandas excede la cantidad del bien a repartir. Para solucionar estos problemas, se presentan diferentes reglas de reparto y algunas de las propiedades que verifican.

Se estudian también los problemas de bancarrota con bienes indivisibles, como puede ser el caso, por ejemplo, de las listas de espera para las cirugías en hospitales. Se presentan diferentes reglas de reparto y algunas de las propiedades que verifican. Estos problemas se aplican al caso particular del reparto de plazas de titular en la Universidade de Vigo.

Además, se programan en el software `R` algunas de las reglas de reparto y se crea una aplicación con el paquete `shiny` en la que se pueden aplicar las reglas programadas de forma sencilla.

## English abstract

In this paper a study of the classic problems of bankruptcy is made. They represent a situation in which a quantity of a perfectly divisible good must be divided among different agents. Each one of the agents has a demand, and the sum of the demands exceeds the amount of the good to be distributed. To solve these problems, different distribution rules and some of the properties that they verify are presented.

The problems of bankruptcy with indivisible goods are also studied, as it may be the case, for example, of the waiting lists for surgeries in hospitals. Different distribution rules and some of the properties that they verify are presented. These problems are applied to the particular case of the distribution of associate professor positions in the University of Vigo.

In addition, some of the distribution rules are programmed in the software `R` and an application with the package `shiny` is created in which the programmed rules can be applied easily.

# Prefacio

La humanidad lleva siglos enfrentándose al problema clásico de bancarrota que surge cuando se quiere dividir una cantidad limitada de un bien perfectamente divisible (estado) entre un conjunto de agentes cuya demanda supera la cantidad total a repartir. ¿Cómo debe repartirse el bien entre los agentes? Para resolver este problema, a lo largo de los años, se han buscado y planteado reglas de reparto que satisfagan ciertas propiedades deseables.

Las referencias más antiguas se encuentran en el Talmud, colección de textos tradicionales del judaísmo que recogen principalmente las discusiones rabínicas sobre leyes judías, tradiciones, costumbres, narraciones y dichos, parábolas, historias y leyendas. En el Talmud aparecen algunos ejemplos y diversos mecanismos para solucionar los problemas clásicos de bancarrota. Hoy en día, nos encontramos ante un problema de bancarrota en cualquier tipo de ámbito y situación como, por ejemplo, concurso de acreedores de empresas en quiebra, reparto de herencias, etc.

Sin embargo, en muchas ocasiones el bien a repartir es indivisible y por tanto las reglas de reparto propuestas para dar solución al problema clásico de bancarrota no son válidas. Este es el caso de las listas de espera para cirugías en el hospital o la asignación de slots de tiempo en los aeropuertos para las diferentes aerolíneas. Estamos ante un nuevo problema de bancarrota en el que tanto las demandas como el estado son indivisibles. Para resolver este nuevo problema, se necesitará modificar algunas de las reglas de reparto usadas en el caso del problema clásico de bancarrota y estudiar reglas de reparto nuevas.

En esta memoria, además de presentar las reglas de reparto para problemas de bancarrota con bienes divisibles e indivisibles, se considerará el problema de bancarrota dado por la asignación de plazas de profesores titulares entre las áreas de la Universidade de Vigo. Cada año salen a concurso un cierto número de plazas de profesores titulares de universidad y, en la actualidad, el número de profesores acreditados que optan a dichas plazas sobrepasa el número de plazas convocadas. Por lo tanto, nos encontramos ante un problema de bancarrota con bienes indivisibles para el que es necesario establecer un mecanismo que nos permita decidir en que áreas de conocimiento se convocarán las plazas. Se introducirá un nuevo problema de bancarrota con bienes indivisibles en el que los agentes poseen una puntuación para optar a cada una de las unidades del estado y se agrupan en grupos que tienen un derecho mínimo que debe otorgársele.

Este trabajo se divide en 4 capítulos. En el Capítulo 1 se presenta el problema clásico de bancarrota en el que el estado es perfectamente divisible, se explican las principales reglas de reparto, las propiedades que verifican y se caracterizan dichas reglas. Para ello se ha seguido principalmente el artículo de Thomson (2003).

En el Capítulo 2 se presenta el problema de bancarrota con bienes indivisibles, se explican las principales reglas de reparto, se ve que propiedades verifican y se caracterizan dichas reglas. Además, se estudiará un caso particular de estos problemas en los que en lugar de tener demandas los agentes tienen preferencias. Para la elaboración de este capítulo se han seguido principalmente los artículos de Chen (2015) y Herrero y Martínez (2004, 2008a y 2008b).

En el Capítulo 3 se define el problema de bancarrota que surge al querer asignar plazas de profesores titulares entre las áreas de la Universidade de Vigo. Se presenta la regla de reparto usada por la Universidade de Vigo y se propone una regla alternativa. Además, se comprueba qué propiedades verifica cada una de las reglas.

Para finalizar, en el Capítulo 4, se programan las reglas de reparto en problemas de bancarrota con bienes indivisibles usando el lenguaje de programación R. Se programan solamente aquellas reglas que verifican la propiedad de equilibrio, una propiedad de justicia. Además, se ha creado una aplicación con un interfaz amigable, utilizando el paquete shiny de R, en la que introduciendo los datos necesarios de manera sencilla y escogiendo la regla de reparto deseada, se obtiene la asignación final sin necesidad de manipular el código.

# Capítulo 1

# Introducción a los problemas de bancarrota

Los problemas de bancarrota surgen cuando se dispone de una cantidad limitada de un bien perfectamente divisible (dinero, agua, etc.) que hay que repartir entre un conjunto de agentes cuya demanda supera la cantidad total a repartir. Cada agente demandará una cantidad que puede ser inferior, igual o superior al total de la cuantía a repartir, pero la suma de las demandas supera la cantidad a repartir. En este caso, ¿cómo debe repartirse el bien entre los agentes? Este reparto debe hacerse de forma justa, empleando ciertos criterios éticos y procedimentales. Dichos criterios dependerán de la naturaleza del problema que se aborda.

Los problemas de bancarrota aparecen en casi todos los ámbitos cotidianos y afectan tanto a personas como a empresas o instituciones. Por ejemplo, a la hora de la comida un matrimonio y sus dos hijos demandan medio litro de agua cada uno para comer pero sólo tienen una botella de litro y medio de agua para todos. Aparece de esta forma un problema de bancarrota ya que la demanda total de agua supera a la cantidad disponible.

Otros problemas de bancarrota que se pueden dar en nuestra sociedad son los siguientes:

- Concurso de acreedores de empresas en quiebra: es el caso más recurrente y el primero que nos viene a la cabeza cuando hablamos de problemas de bancarrota. En este caso se tienen que repartir los activos entre los distintos acreedores.
- Recortes presupuestarios: el Gobierno elabora anualmente los Presupuestos Generales del Estado. Cuando la recaudación del Estado cae, los presupuestos deben reducirse para cumplir los objetivos de déficit que se marcan desde la Unión Europea. Por tanto, se lleva a cabo un reparto de pérdidas (problema de bancarrota) entre sus distintos organismos como consecuencia de la reducción del presupuesto.
- Distribución de recursos hídricos regulados por las Confederaciones Hidrográficas para sus posibles usos (riego, consumo doméstico, consumo industrial, etc.)
- Herencias: en este caso existe un problema de bancarrota cuando el testador fallece y la cantidad de los bienes a repartir tales como, hectáreas de tierra o dinero, es inferior a la cuantía de la suma de las demandas de los herederos legítimos.
- Subvenciones: tanto personas como empresas o instituciones desean, por lo general, obtener este tipo de ayudas cuya cantidad suele ser bastante limitada. En este punto podemos considerar, entre otros, la concesión de becas a estudiantes o las ayudas a ganaderos y agricultores.

Los antecedentes históricos de este tipo de situaciones comienzan en la tradición judía, donde aparecen algunos ejemplos y diversos mecanismos para solucionarlas. El Talmud es una colección de textos tradicionales del judaísmo tardío que se divide en dos secciones: el Mishná y el Gemara. Un problema que aparece resuelto en el Mishná es el siguiente: Un hombre tiene tres esposas cuyos contratos matrimoniales especifican que a su muerte deberían recibir 100, 200 y 300 unidades monetarias, respectivamente. El hombre muere y se encuentra que su patrimonio asciende solo a 100 unidades monetarias. ¿Cuánto le corresponde a cada esposa? El Talmud recomienda el reparto $(33.33, 33.33, 33.33)$. Más adelante veremos como se llega a esa solución.

Resolver un problema de bancarrota consiste en dar un pago, razonable en algún sentido, a cada uno de los agentes. Teniendo en cuenta la situación que modeliza un problema de bancarrota toda solución suya debe dividir el estado entre el conjunto de agentes y ningún agente debe recibir ni un pago negativo ni más de lo que demanda.

## 1.1. Planteamiento y notación

Dado el conjunto de números naturales $\mathbb{N}$, $\mathcal{N}$ denota la familia de todos los subconjuntos finitos no vacíos de $\mathbb{N}$. Llamaremos $E$ a la cantidad de un bien divisible (o estado) que debe repartirse entre un conjunto finito de agentes, $N \in \mathcal{N}$. Denotaremos por $n$ al número de elementos de $N$. A la demanda del agente $i$-ésimo la denotaremos por $c_i$ y el vector formado por las demandas de todos los agentes lo denotaremos por $c = (c_1, c_2, \cdots, c_n) \in \mathbb{R}^n_+$. A la suma de las demandas de los agentes la llamaremos demanda agregada y la denotaremos por $C = \sum_{i \in N} c_i$. Por último, a la diferencia entre la demanda agregada y el estado, $C - E$, la definimos como déficit.

Estaremos ante un problema de bancarrota, cuando se cumplan las siguientes condiciones:

- $C = \sum_{i \in N} c_i \geq E$. En caso contrario, no habría problema en satisfacer las demandas de todos los agentes.

- $c \geq 0$. En caso contrario, habría agentes que deberían dinero a la empresa quebrada, por lo que no serían demandantes.

- $0 < E$. En caso contrario, se estaría repartiendo deuda, en lugar del valor de un bien liquidado.

Por lo tanto, un problema de bancarrota es un par $(E, c) \in \mathbb{R}_+ \times \mathbb{R}^N_+$ tal que $\sum_{i \in N} c_i \geq E$.

Se denota por $\mathbb{A}^N$ el conjunto de todos los problemas de bancarrota con un conjunto de agentes fijo $N$, y $\mathbb{A}$ denota el conjunto de todos los problemas, esto es,

$$\mathbb{A}^N = \{(E, c) \in \mathbb{R} \times \mathbb{R}^N\}$$

y

$$\mathbb{A} = \bigcup_{N \in \mathcal{N}} \mathbb{A}^N.$$

La solución de un problema de bancarrota $(E, c)$ vendrá determinada por una regla de reparto. Una regla de reparto asociada a un problema de bancarrota es una función $F$ que asocia a cada par $(E, c)$ un vector $F(E, c) \in \mathbb{R}^N$, donde cada componente $F_i(E, c)$ denota la asignación que recibirá el agente $i$. Dicho vector debe cumplir las siguientes condiciones:

- $0 \leq F_i(E, c) \leq c_i$. La cantidad asignada a cada agente debe ser positiva y no superior a la demanda del agente.

- $\sum_{i \in N} F_i(E, c) = E$. La suma de todas las asignaciones debe ser igual a la cantidad del bien a repartir. Esta propiedad se conoce como *eficiencia*.

## 1.2. Reglas de reparto

En esta sección se presentarán algunas de las reglas de reparto más conocidas: regla de reparto de igual ganancia, de igual pérdida, proporcional, del Talmud y de llegada aleatoria. Explicaremos cada una de ellas aplicándola en el ejemplo del Talmud mencionado anteriormente.

Un hombre tiene tres esposas en cuyos contratos matrimoniales figura que si muere, sus mujeres deben recibir 100, 200 y 300 unidades monetarias, respectivamente. Entonces, $c = (100, 200, 300)$. En el momento de su defunción, su hacienda asciende a 100 unidades monetarias, $E = 100$, por lo que no llega para satisfacer las demandas. ¿Cómo se divide el dinero entre las esposas? Veremos qué asignación propone cada regla de reparto.

### 1.2.1. Regla de igual ganancia

La regla de reparto de igual ganancia, $CEA$ (*Constrained equal awards rule*), asigna a cada agente la misma cantidad, con la restricción de que ninguno de ellos puede recibir una cuantía superior a su demandada. En ese caso se les asignaría el total de su demanda. Esta regla favorece a los agentes con demandas más bajas, ya que reciben toda su demanda o reciben lo mismo que los agentes con demandas más altas. Formalmente,

$$CEA_i(E, c) = min\{\lambda, c_i\},$$

siendo $\lambda$ solución de $\sum_{i \in N} min\{\lambda, c_i\} = E$.

En definitiva, a todos los agentes se les asigna $\lambda$ excepto a quienes demandan menos de $\lambda$, que se les asigna su demanda.

Considerando los datos del ejemplo expuesto anteriormente, donde $E = 100$ y $c = (100, 200, 300)$, se tiene que $\lambda$ es igual a $\frac{100}{3}$ y las asignaciones aplicando la regla de igual ganancia son las siguientes:

$$CEA_1(E, c) = min\left\{\frac{100}{3}, 100\right\} = \frac{100}{3},$$

$$CEA_2(E, c) = min\left\{\frac{100}{3}, 200\right\} = \frac{100}{3},$$

$$CEA_3(E, c) = min\left\{\frac{100}{3}, 300\right\} = \frac{100}{3}.$$

En este caso, se reparte equitativamente el estado entre las tres esposas, $\frac{100}{3}$ unidades monetarias para cada una de ellas. Por lo que el vector de asignaciones es:

$$CEA(E, c) = \left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right).$$

Supongamos ahora que el estado es de 400 unidades monetarias y que las demandas siguen siendo las mismas. En este caso $\lambda = 150$ y las asignaciones obtenidas son las siguientes:

$$CEA_1(E, c) = min\{150, 100\} = 100,$$

$$CEA_2(E, c) = min\{150, 200\} = 150,$$

$$CEA_3(E, c) = min\{150, 300\} = 150.$$

Si dividiésemos el estado equitativamente entre las tres esposas, cada una se llevaría $\frac{400}{3}$ unidades monetarias. La primera esposa llevaría más de lo que demanda, por lo que no se respetaría la restricción citada anteriormente. Entonces, se le asigna el total de la demanda y a las otras esposas se les asigna $\lambda$, cantidad que equivale a repartir el estado sobrante entre las dos esposas restantes. Entonces,

$$CEA(E, c) = (100, 150, 150).$$

### 1.2.2. Regla de igual pérdida

La regla de igual pérdida, $CEL$ (*Constrained equal losses rule*), reparte el déficit, $(C - E)$, por igual entre cada agente con la restricción de que ninguno de ellos puede perder una cantidad superior a su demanda, es decir, su asignación no puede ser negativa. Esta regla, al contrario que la anterior, favorece a los agentes con demandas mayores. Formalmente,

$$CEL_i(E, c) = max\{0, c_i - \lambda\},$$

donde $\lambda$ es la solución de $\sum_{i \in N} max\{0, c_i - \lambda\} = E$.

Recuperando el ejemplo anterior con $E = 100$ y $c = (100, 200, 300)$. Hay que repartir un déficit de 500 unidades. Se tiene que $\lambda$ es igual a 200 y las asignaciones obtenidas aplicando la regla de reparto de igual pérdida son las siguientes:

$$CEL_1(E, c) = max\{0, 100 - 200\} = 0,$$

$$CEL_2(E, c) = max\{0, 200 - 200\} = 0,$$

$$CEL_3(E, c) = max\{0, 300 - 200\} = 100.$$

Si distribuyésemos el déficit a partes iguales entre las tres mujeres, $\frac{600-100}{3} = \frac{500}{3}$, la primera mujer perdería una cantidad superior a su demanda, por lo que no se cumpliría la restricción citada anteriormente. De esta forma, la primera mujer no recibe nada y se le asigna 100 unidades monetarias de déficit. A las otras dos mujeres se les asigna un déficit $\lambda$, que se obtiene al repartir el déficit sobrante, 400, de forma igualitaria entre ellas. Por lo que

$$CEL(E, c) = (0, 0, 100).$$

Consideremos ahora un estado de 400 unidades monetarias y las mismas demandas. Hay que repartir un déficit de 200 unidades. Se tiene $\lambda = \frac{200}{3}$ y las asignaciones obtenidas son las siguientes:

$$CEL_1(E, c) = max\left\{0, 100 - \frac{200}{3}\right\} = \frac{100}{3},$$

$$CEL_2(E, c) = max\left\{0, 200 - \frac{200}{3}\right\} = \frac{400}{3},$$

$$CEL_3(E, c) = max\left\{0, 300 - \frac{200}{3}\right\} = \frac{700}{3}.$$

En este caso se reparte el déficit equitativamente entre las tres esposas, a cada una le corresponden $\frac{200}{3}$ unidades monetarias. Entonces,

$$CEL(E, c) = \left(\frac{100}{3}, \frac{400}{3}, \frac{700}{3}\right).$$

### 1.2.3. Regla proporcional

La regla proporcional, $P$ (*Proportional rule*), es la más utilizada en la práctica. Distribuye el estado de manera proporcional a las demandas realizadas por cada agente. Se define un coeficiente de proporcionalidad $\lambda$ dividiendo el estado, $E$, entre la suma de las demandas de los agentes, $C$. Este coeficiente se multiplica por cada una de las demandas de los agentes para obtener sus asignaciones.

Formalmente, la regla de reparto proporcional viene definida de la siguiente manera:

$$P(E, c) = \lambda \cdot c,$$

donde $\lambda = E/C$.

Veamos qué reparto proporciona esta regla para el ejemplo anterior. Tenemos que el estado es $E = 100$ y el vector de demandas es $c = (100, 200, 300)$, por lo que la demanda agregada es $C = 600$. Entonces, el coeficiente de proporcionalidad es $\lambda = \frac{100}{600} = \frac{1}{6}$. Aplicando la regla de reparto proporcional, la asignación obtenida es:

$$P(E, c) = \lambda \cdot c = \left(\frac{100}{6}, \frac{200}{6}, \frac{300}{6}\right) = \left(\frac{50}{3}, \frac{100}{3}, 50\right).$$

Si consideramos ahora que el estado es de 400 unidades monetarias y las mismas demandas, el coeficiente de proporcionalidad es $\lambda = \frac{400}{600} = \frac{2}{3}$. Aplicando la regla de reparto proporcional, la asignación obtenida es:

$$P(E, c) = \left(\frac{200}{3}, \frac{400}{3}, \frac{600}{3}\right) = \left(\frac{200}{3}, \frac{400}{3}, 200\right).$$

### 1.2.4. Regla del Talmud

La regla del Talmud, $T$ (*Talmud rule*), es una mezcla de las reglas de igual ganancia e igual pérdida. Coincide con la regla de igual ganancia cuando $E < \frac{C}{2}$ y $\frac{E}{n} \leq \frac{min\{c_i\}}{2}$ y con la regla de igual pérdida si $E > \frac{C}{2}$ y además $\frac{C-E}{n} \leq \frac{min\{c_i\}}{2}$.

De esta forma, Aumann y Maschler (1985), definieron la regla del Talmud, T, de la siguiente manera:

$$T_i(E, c) = \begin{cases} min\{\frac{c_i}{2}, \lambda\} & \text{si } E \leq \frac{C}{2}, \text{ siendo } \lambda \text{ solución de } \sum_{i \in N} min\{\lambda, \frac{c_i}{2}\} = E \\ max\{\frac{c_i}{2}, c_i - \lambda\} & \text{si } E \geq \frac{C}{2}, \text{ siendo } \lambda \text{ solución de } \sum_{i \in N} max\{\frac{c_i}{2}, c_i - \lambda\} = E \end{cases}$$

En el ejemplo que estamos viendo, con $E = 100$ y $c = (100, 200, 300)$, el estado es menor a la mitad de la demanda agregada. Por lo que $\lambda = \frac{100}{3}$ y las asignaciones obtenidas son:

$$T_1(E, c) = min\left\{\frac{c_1}{2}, \lambda\right\} = \frac{100}{3},$$

$$T_2(E, c) = min\left\{\frac{c_2}{2}, \lambda\right\} = \frac{100}{3},$$

$$T_3(E, c) = min\left\{\frac{c_3}{2}, \lambda\right\} = \frac{100}{3}.$$

Como $E < \frac{C}{2}$, la regla del Talmud coincide con la regla de igual ganancia vista anteriormente. En este caso se quiere dividir el estado a partes iguales entre las esposas con la restricción de que ninguna reciba más de lo que demanda. Empezamos dividiendo el estado a partes iguales. Como $\frac{100}{3}$ no es mayor que la mitad de la demanda de cada una de las mujeres, la asignación es

$$T(E, c) = \left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right).$$

Si en lugar de tener $E = 100$, tenemos $E = 400$, el estado es mayor a la mitad de la demanda agregada, por lo que la regla del Talmud coincide con la regla de igual pérdida. En este caso $\lambda = 75$ y las asignaciones obtenidas son:

$$T_1(E, c) = max\left\{\frac{c_1}{2}, c_1 - \lambda\right\} = 50,$$

$$T_2(E,c) = max\left\{\frac{c_2}{2}, c_2 - \lambda\right\} = 125,$$
$$T_3(E,c) = max\left\{\frac{c_3}{2}, c_3 - \lambda\right\} = 225.$$

En primer lugar, el déficit a dividir entre las tres mujeres es de $\frac{200}{3}$ unidades monetarias a cada una, pero de esta forma la primera mujer pierde más de la mitad de su demanda, por lo que se le asigna un déficit igual a la mitad de su demanda, 50. En segundo lugar, se asignará el déficit sobrante equitativamente entre las otras dos mujeres ($\frac{150}{2} = 75$). De esta forma:

$$T(E,c) = (50, 125, 225).$$

### 1.2.5. Regla de llegada aleatoria

Para definir esta regla imaginemos que los agentes llegan de uno en uno y se les va asignando el mínimo entre la demanda y el estado disponible hasta que este se termina. Es decir, el agente 1 recibe el mínimo entre lo que demanda y el estado, luego el agente 2, y así sucesivamente hasta agotar el estado. Para que el reparto sea justo y que no beneficie a aquellos agentes que llegan primero cuando se ha declarado bancarrota, el reparto de llegada aleatoria, $RA$ (*Random arrival rule*), es el promedio de esas asignaciones en todos los posibles órdenes de llegada.

Dicho de otra forma, la regla de llegada aleatoria, es una regla de asignación en la que cada agente recibe el promedio que puede obtener, es decir, el mínimo de la cantidad igual a su demanda y el remanente, si todos los agentes aparecen uno por uno al azar. Formalmente

$$RA_i(E,c) = \frac{1}{n!}\sum_{\pi\in\Pi^N} min\left\{c_i, max\left\{E - \sum_{j\in N:\pi(i)<\pi(j)} c_j, 0\right\}\right\}, \forall i \in N.$$

Resolvamos el problema del Talmud aplicando ahora la regla de llegada aleatoria. Recordemos que el estado era $E = 100$ y las demandas, $c = (100, 200, 300)$. En la tabla 1.1 mostramos los $n! = 3! = 6$ posibles órdenes de llegada y el reparto que establece la regla tomando el promedio.

| Demandas \ Órdenes | 123 | 132 | 213 | 231 | 312 | 321 | Regla de llegada aleatoria |
|---|---|---|---|---|---|---|---|
| 100 | 100 | 100 | 0 | 0 | 0 | 0 | 200/6=100/3 |
| 200 | 0 | 0 | 100 | 100 | 0 | 0 | 200/6=100/3 |
| 300 | 0 | 0 | 0 | 0 | 100 | 100 | 200/6=100/3 |

Tabla 1.1: Asignación obtenida mediante la regla de llegada aleatoria con $E = 100$.

En la primera columna, el orden de llegada es 123, lo que quiere decir que primero llega la primera esposa y se le asigna su demanda, después llega la segunda esposa y ya no hay estado para satisfacer su demanda y, por último, llega la tercera esposa para la que tampoco queda estado disponible. Se razona de igual forma para los demás órdenes posibles y, por último, se calcula el promedio de lo que recibe cada una en los diferentes órdenes.

Por lo que la asignación obtenida es la siguiente:

$$RA(E,c) = \left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right).$$

Si ahora consideramos $E = 400$ y las mismas demandas, en la tabla 1.2 se tiene la asignación obtenida mediante la regla de llegada aleatoria.

| Demandas \ Órdenes | 123 | 132 | 213 | 231 | 312 | 321 | Regla de llegada aleatoria |
|---|---|---|---|---|---|---|---|
| 100 | 100 | 100 | 100 | 0 | 100 | 0 | 400/6=200/3 |
| 200 | 200 | 0 | 200 | 200 | 0 | 100 | 700/6=350/3 |
| 300 | 100 | 300 | 100 | 200 | 300 | 300 | 1300/6=650/3 |

Tabla 1.2: Asignación obtenida mediante la regla de llegada aleatoria con $E = 400$.

En este caso, cuando el orden es 123, la primera esposa llega primero y se le asigna su demanda, después llega la segunda esposa y se le asigna su demanda y, por último, llega la tercera esposa y se le asigna la cantidad de estado que aún queda disponible, que son 100 unidades monetarias. Se razona de la misma manera para los distintos órdenes y, por último, se calcula el promedio de lo que recibe cada una en los diferentes órdenes. Por lo que se obtiene la siguiente asignación:

$$RA(E,c) = \left(\frac{200}{3}, \frac{350}{3}, \frac{650}{3}\right).$$

Una vez vistas todas las reglas de reparto, en la Tabla 1.3 se tiene de forma resumida la asignación que propone coda una de las reglas de reparto para el problema del Talmud.

| | Igual ganancia | Igual pérdida | Proporcional | Talmud | Llegada aleatoria |
|---|---|---|---|---|---|
| $E = 100$ | $\left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right)$ | (0,0,100) | $\left(\frac{50}{3}, \frac{100}{3}, 50\right)$ | $\left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right)$ | $\left(\frac{100}{3}, \frac{100}{3}, \frac{100}{3}\right)$ |
| $E = 400$ | (100,150,150) | $\left(\frac{100}{3}, \frac{400}{3}, \frac{700}{3}\right)$ | $\left(\frac{200}{3}, \frac{400}{3}, 200\right)$ | (50,125,225) | $\left(\frac{200}{3}, \frac{350}{3}, \frac{650}{3}\right)$ |

Tabla 1.3: Asignación obtenida con cada una de las reglas del reparto para el problema del Talmud.

## 1.3. Propiedades de las reglas de reparto

En esta sección recogeremos parte de las propiedades que se han propuesto en la literatura, asociadas a las reglas de bancarrota.

Algunas propiedades ya están incorporadas en la definición de regla de reparto. Una de ellas es la *factibilidad*: es simplemente el requerimiento de que la suma de las asignaciones no debe exceder la cantidad disponible. Otra de estas propiedades es la *eficiencia*: es el requerimiento de que la cantidad disponible debe ser repartida totalmente. Es obvio que no podemos distribuir más de lo que hay, pero es concebible que podamos distribuir menos. Sin embargo, insistiremos en la igualdad entre la suma de asignaciones y la cantidad disponible.

Las dos siguientes propiedades ponen límites a las asignaciones. Ambas son naturales y también las hemos incorporado en la definición de regla de reparto. *No negatividad* nos da un límite inferior, cada agente debe recibir una cantidad no negativa. *Acotación de demandas* nos da un límite superior, cada agente debe recibir como máximo su demanda.

Como consecuencia de las tres propiedades anteriores se tiene que cada demandante debe recibir al menos la diferencia entre la cantidad disponible y la suma de las demandas de los demás demandantes si la diferencia es no negativa, y 0 en otro caso. Nos referimos a esta cantidad como el *derecho mínimo del demandante*.

A continuación, veremos otras propiedades que ya no están incluidas en la definición de regla de reparto.

### Tratamiento igualitario o simetría

Esta propiedad nos dice que los agentes que reclaman lo mismo deben recibir la misma asignación, es decir, se trata a iguales por igual (Thomson, 2003).

Dado un problema de bancarrota $(E, c)$ y dos agentes $i, j \in N$ tal que $c_i = c_j$, $F$ verifica simetría si

$$F_i(E, c) = F_j(E, c).$$

Esto quiere decir que dos agentes son indistinguibles a efectos de reparto y asignación si demandan lo mismo.

Esta propiedad no está siempre justificada y, de hecho, en la práctica no se cumple a menudo. Por ejemplo, en los actuales procedimientos de bancarrota, algunos demandantes (por ejemplo, demandas garantizadas) tienen mayor prioridad que otros (como demandas sin garantía).

Las cinco reglas de reparto definidas en la sección anterior (igual ganancia, igual pérdida, proporcional, Talmud y llegada aleatoria) cumplen esta propiedad.

### Anonimato

Esta propiedad nos dice que la identidad de los agentes no debe importar, el vector de asignaciones sólo debe depender de la lista de demandas, no de quién las realiza (Thomson, 2003).

Dado un problema de bancarrota $(E, c)$, para cada $\pi \in \Pi^N$, y cada $i \in N$, $F$ verifica la propiedad de anonimato si

$$F_{\pi(i)}(E, (c_{\pi(i)})_{i \in N}) = F_i(E, c).$$

Todas las reglas de reparto vistas anteriormente cumplen esta propiedad.

### Independencia de escala

Si las demandas y el estado se multiplican por un número positivo, entonces también deberían hacerlo todas las asignaciones. Dicho de otro modo, las asignaciones no deben depender de la unidad de medida (ya sea monetaria al cambiar de divisa, de medida al cambiar del sistema anglosajón al métrico decimal, etc.) (Thomson, 2003).

Dado un problema de bancarrota $(E, c)$, diremos que $F$ verifica independencia de escala si

$$F(\lambda E, \lambda c) = \lambda F(E, c) \text{ para cualquier } \lambda > 0.$$

Todas las reglas de reparto definidas anteriormente cumplen esta propiedad.

### Composición hacia arriba

Consideremos la siguiente situación: después de haber dividido el valor de liquidación de una empresa en quiebra entre sus acreedores, sus activos son revaluados y se determina que valen más de lo que originalmente se pensaba, $E' > E$. Para tratar esta nueva situación tenemos dos opciones: cancelar la división inicial y aplicar la regla de reparto al problema revisado, o dejar que los agentes conserven sus adjudicaciones iniciales, ajustar las reclamaciones según estos valores y aplicar la regla para dividir el excedente $(E' - E)$. La propiedad de composición hacia arriba nos dice que ambas formas de proceder deberían dar como resultado las mismas asignaciones (Young, 1988).

Dicho de otro modo, ante una variación al alza de la cantidad a repartir, las asignaciones proporcionadas por la regla de reparto a los agentes serán las mismas tanto si se reparte la cantidad finalmente disponible $(E')$ , como si reparte primero la cantidad inicial y después el excedente $(E' - E)$.

Dado un problema de bancarrota $(E, c)$ y $E' > E$, decimos que $F$ verifica composición hacia arriba si

$$F(E', c) = F(E, c) + F(E' - E, c - F(E, c)).$$

De las reglas de reparto vistas anteriormente, cumplen esta propiedad las reglas de igual ganancia, igual pérdida y la regla proporcional.

## Composición hacia abajo

Ahora consideremos la situación contraria: después de dividir el valor de liquidación de una empresa entre sus acreedores, sus activos son revaluados y se determina que valen menos de lo que inicialmente se pensaba, $E' < E$. Para tratar esta nueva situación, tenemos dos opciones disponibles: cancelar la división inicial y aplicar la regla al problema revisado; o considerar las asignaciones iniciales como demandas sobre el valor revisado $(E')$ y aplicar la regla al problema definido de esta forma. La composición hacia abajo nos dice que ambas formas de proceder dan como resultado los mismos vectores de asignaciones (Moulin, 2000).

Dicho de otro modo, ante una variación a la baja de la cantidad inicial a repartir, cada agente percibirá lo mismo tanto si mantiene su reclamación inicial, como si reclama la cantidad que se le habría asignado con el presupuesto antiguo $E$.

Dado un problema de bancarrota $(E, c)$ y $E' < E$, diremos que $F$ verifica composición hacia abajo si

$$F(E', c) = F(E', F(E, c)).$$

De las reglas de reparto vistas anteriormente, cumplen esta propiedad las reglas de igual ganancia, igual pérdida y la regla proporcional.

## Consistencia

Esta propiedad expresa un cierto tipo de independencia de las reglas con respecto a cambios en el conjunto de agentes. Supongamos que después de aplicar una regla de reparto, algunos agentes se van con sus asignaciones. Esta propiedad nos dice que si ahora consideramos solo los agentes restantes y el estado que queda al irse los otros agentes, la regla debe otorgar a cada uno de ellos la misma cantidad que inicialmente (Aumann y Maschler, 1985).

El problema que enfrentan los demandantes restantes se define de la siguiente manera: sus demandas no se modifican y la cantidad a dividir es la diferencia entre la cantidad disponible inicialmente y la suma de las asignaciones a los agentes que se fueron. Se llama problema reducido en relación con el subgrupo y la recomendación inicial. La reducción produce un problema de demandas bien definido.

Dado un problema de bancarrota $(E, c)$ y un subconjunto $S \subset N$. Decimos que $F$ cumple la propiedad de consistencia si y solo si, para todo $i \in N \backslash S$

$$F_i(N, E, c) = F_i(N \backslash S, E - \sum_{i \in S} F_i(N, E, c), (c_i)_{i \in N \backslash S}).$$

Esta propiedad la cumplen las reglas de igual ganancia, igual pérdida, la regla proporcional y la regla del Talmud.

## Autodualidad

Un problema de bancarrota también puede ser abordado como un problema de reparto de pérdidas, asignando el déficit $(C-E)$ entre los agentes. La propiedad de autodualidad exige que las asignaciones deben ser las mismas tanto si se considera dividir la cantidad disponible como si se considera repartir el déficit (Aumann y Maschler, 1985).

Dado un problema de bancarrota $(E, c)$. Decimos que $F$ verifica autodualidad si

$$c - F(C - E, c) = F(E, c).$$

Las reglas de reparto que cumplen esta propiedad son la regla proporcional, la regla del Talmud y la regla de llegada aleatoria.

## Compensación completa

Compensar a los agentes con demandas pequeñas es relativamente más fácil y, en algunas ocasiones, puede ser tentador hacerlo primero.

El criterio que uno debe adoptar para decidir qué tan pequeño debe ser un reclamo para que su propietario merezca este trato preferencial es una cuestión de juicio, pero un valor crítico interesante de una demanda se obtiene de la siguiente forma: sustituyámosla por la demanda de cualquier otro agente cuya demanda sea mayor y verifiquemos si hay sificiente para compensar a todos. En este caso, el agente recibirá su demanda completa (Herrero y Villar, 2001b).

Formalmente, dado un problema de bancarrota $(E, c)$ y un agente $i \in N$ tal que $\sum_{j \in N} min\{c_j, c_i\} \leq E$, entonces

$$F_i(E, c) = c_i.$$

Por ejemplo, supongamos que tenemos el problema de bancarrota $(E, c)$ donde $N = \{1, 2, 3\}$, $c = (2, 6, 6)$ y $E = 9$. Bajo la perspectiva del primer agente, su demanda es razonable en el siguiente sentido: Si los agentes 2 y 3 no demandasen más de lo que demanda el agente 1, sus demandas podrían ser satisfechas totalmente. De esta forma, el agente 1 no es responsable de la escasez de recursos por lo que debe recibir lo que demanda.

Una opción alternativa para un valor crítico de un reclamo por debajo del cual podría requerirse una compensación total es simplemente $\alpha = \frac{E}{n}$ (Herrero y Villar, 2001a). Las demandas que no superen la cantidad $\alpha$ deben ser otorgadas de forma íntegra. Esta propiedad se conoce como "exención".

Formalente, dado un problema de bancarrota $(E, c)$ y un agente $i \in N$ tal que $c_i \leq \alpha$, entonces

$$F_i(E, c) = c_i.$$

Esta propiedad también puede entenderse como una prioridad en el reparto para los agentes con demandas pequeñas. Y, por ley, las reglas que se aplican en las bancarrotas de entidades financieras deben atender este principio para favorecer las pretensiones de los pequeños impositores.

De todas las reglas que hemos visto, sólo la regla igual ganancia cumple la propiedad de compensación completa.

## Compensación vacía

Simétricamente, se podría adoptar el punto de vista de que si una demanda es demasiado pequeña (demanda residual), no se le asigna nada al agente con esa demanda. Una demanda $c_i$ se puede considerar residual si $(E, (max\{0, c_1 - c_i\}, ..., max\{0, c_n - c_i\}))$ es un problema de bancarrota (Herrero y Villar, 2001b).

Se dice que una regla de reparto verifica la propiedad de compensación vacía si para cada $(E, c)$ y cada $i \in N$ con demanda residual se tiene que $F_i(E, c) = 0$.

Herrero y Villar (2001a), consideran como umbral el déficit per cápita, $\alpha = \frac{C-E}{m}$. Esta propiedad se conoce como "exclusión". Dado un problema de bancarrota $(E, c)$ y un agente $i \in N$ tal que $c_i \leq \alpha$, entonces

$$F_i(E, c) = 0.$$

La propiedad de compensación vacía se tiene en cuenta, por ejemplo, en ciertas situaciones relacionadas con la sanidad donde, por la cuantía que comportan, se pueden financiar medicamentos de primera necesidad o tratamientos médicos irrecusables, mientras que el paciente debe sufragar los correspondientes gastos que sean menos prioritarios y que entrañan un desembolso asequible. En el contexto de la quiebra, por ejemplo, el objetivo sería dar prioridad a los agentes que han arriesgado cantidades relativamente mayores. En el contexto de los impuestos, eximir a los agentes con ingresos más bajos es una característica de casi todas las leyes impositivas del mundo real.

De todas las reglas de reparto vistas anteriormente, la única que cumple esta propiedad es la regla de igual pérdida.

## Derechos mínimos primero

Esta propiedad indica que el vector de asignaciones se puede obtener de dos formas distintas: asignando primero a cada agente su derecho mínimo y después aplicando una regla de reparto para asignar el estado restante, o aplicando la regla de reparto directamente al problema de bancarrota (Curiel et al., 1987).

El derecho mínimo de cada agente, como ya se ha mencionado al inicio de la sección, es el máximo entre la diferencia entre el estado y la suma de las demandas de los demás agentes y 0, es decir, $m_i(E, c) = max\{E - \sum_{N\setminus\{i\}} c_j, 0\}$.

Dado un problema de bancarrota $(E, c)$, diremos que $F$ verifica la propiedad de derechos mínimos primero si

$$F(E, c) = m(E, c) + F\left(E - \sum_{i \in N} m_i(E, c), c - m(E, c)\right),$$

donde $m(E, c)$ es el vector de derechos mínimos.

Las reglas de reparto que satisfacen esta propiedad son la regla de igual pérdida, la regla de llegada aleatoria y la regla del Talmud.

## Aseguramiento

Esta propiedad asegura que todos los agentes recibirán una cierta cantidad de sus respectivas demandas, con independencia de las demandas de los otros agentes. La asignación mínima de cada agente va a depender de tres factores: su demanda, el número de agentes y la cantidad a dividir (Moreno y Villar, 2004).

Dado un problema de bancarrota $(E, c)$. $F$ verifica aseguramiento si

$$F_i(E, c) \geq \frac{1}{n} min\{c_i, E\}.$$

Si una demanda supera la cantidad disponible, la asignación será, al menos, la cantidad disponible dividida entre el número de agentes; en caso contrario, la asignación será, al menos, la demanda dividida entre el número de agentes.

Las reglas de igual ganancia, del Talmud y de llegada aleatoria, son las que cumplen esta propiedad.

## Compensación equilibrada

Una regla de reparto verifica esta propiedad si la asignación excluyendo a un agente determinado es la misma que la asignación excluyendo a otro agente distinto del primero (Hwang, 2015).

Formalmente, dado un problema de bancarrota $(E, c)$, decimos que $F$ verifica compensación equilibrada si

$$F_i(E, c) - F_i(N\backslash\{j\}, E_{N\backslash\{j\}}, c_{N\backslash\{j\}}) = F_j(E, c) - F_j(N\backslash\{i\}, E_{N\backslash\{i\}}, c_{N\backslash\{i\}}),$$

donde $E_{N\backslash\{k\}} = max\{E - c_k, 0\}, \forall k \in N$.

La regla de llegada aleatoria es la única que cumple esta propiedad.

## Preservación del orden

Una regla de reparto satisface esta propiedad si otorga mayores pagos y pérdidas a quienes demandan cantidades mayores; de modo que dichos pagos y pérdidas no son menores que los pagos y pérdidas que otorga a quienes demandan cantidades menores; por tanto, lo que recibe (y deja de ganar) el que demanda una cantidad mayor no es menor que lo que recibe (y deja de ganar) el que demanda una cantidad menor, respectivamente (Aumann y Maschler, 1985).

Dado un problema de bancarrota $(E, c)$. Para cualesquiera $i, j \in N$ tales que $c_i \leq c_j$, se dice que $F$ verifica preservación del orden si

$$F_i(E, c) \leq F_j(E, c) \text{ y } c_i - F_i(E, c) \leq c_j - F_j(E, c).$$

Esta propiedad la satisfacen todas las reglas vistas anteriormente.

## Monotonía respecto al estado

Esta propiedad indica que si la cantidad a dividir aumenta y las demandas de los agentes no sufren variación tras este aumento, cada agente debe recibir al menos tanto como inicialmente (Thomson, 2003).

Formalmente, dado un problema de bancarrota $(E, c)$ y $E' > E$, decimos que $F$ verifica monotonía respecto al estado si

$$F_i(E', c) \geq F_i(E, c), \forall i \in N.$$

Las cinco reglas de reparto vistas en la sección anterior cumplen esta propiedad.

## Monotonía respecto a las demandas

Si un agente aumenta su demanda, la regla de reparto verifica la propiedad de monotonía débil respecto a las demandas si el pago de dicho agente no disminuye (Thomson, 2003).

Formalmente, dado un problema de bancarrota $(E, c)$ y un agente $i \in N$ tal que $c'_i > c_i$, $F$ verifica monotonía respecto a las demandas si

$$F_i(E, c') \geq F_i(E, c).$$

En la monotonía respecto a las demandas denotamos por $c' = (c_1, ..., d'_N) \in \mathbb{R}^N_+$ al vector que tiene las mismas componentes que $c$ a excepción de $c'_i > c_i$.

Por definición de monotonía débil respecto de las demandas se deduce que si una regla verifica esta propiedad, el pago conjunto que reciben el resto de agentes no puede aumentar debido a la eficiencia,

$$\sum_{j \in N\backslash\{i\}} R_j(E, c) \geq \sum_{j \in N\backslash\{i\}} R_j(E, c').$$

En el caso de que cada uno de los agentes que no aumentan su demanda, si el pago otorgado a cada uno de ellos no aumenta, diremos que la regla verifica la propiedad de monotonía fuerte respecto a las demandas.

Diremos que una regla verifica la propiedad de monotonía fuerte respecto de las demandas si se tiene

$$F_j(E, c') \leq F_j(E, c), \forall j \in N \text{ tal que } i \neq j.$$

Todas las reglas de reparto vistas anteriormente cumplen esta propiedad.

### Invarianza al truncar demandas

Esta propiedad nos dice que la parte de una demanda que esté por encima del estado a repartir debe ignorarse. Como esta parte de la demanda no se puede reembolsar de todos modos, reemplazar $c_i$ por $E$ para cada $i \in N$ tal que $c_i > E$ no afecta al vector de asignaciones (Curiel et al., 1987).

Formalmente, dado un problema de bancarrota $(E, c)$, $F$ verifica invarianza al truncar demandas si

$$F(E, c) = F(E, (min\{E, c_i\})).$$

Las reglas de reparto que cumplen esta propiedad son la regla de igual ganancia, del Talmud y de llegada aleatoria.

En la Tabla 1.4 situada al final del capítulo, podemos ver de forma resumida que propiedades cumple cada una de las reglas de reparto.

## 1.4. Caracterización de las reglas de reparto

Combinando distintas propiedades de las vistas en la sección anterior, caracterizamos o determinamos las reglas de reparto vistas en la Sección 1.2.

La regla de igual ganancia es la única regla de reparto que verifica las propiedades de:

- **a)** Tratamiento igualitario, invarianza al truncar demandas y composición hacia arriba (Dagan, 1996).
- **b)** Compensación completa y composición hacia abajo (Herrero y Villar, 2001b).
- **c)** Compensación completa y monotonía respecto a las demandas (Yeh, 2001).
- **d)** Aseguramiento, composición hacia arriba y consistencia (Chun, 2006).
- **e)** Consistencia, exención y composición hacia abajo (Herrero y Villar, 2001a).

La regla de igual pérdida es la única que verifica:

- **a)** Tratamiento igualitario, derechos mínimos primero y composición hacia abajo (Herrero y Villar, 2001a).
- **b)** Compensación vacía y composición hacia arriba (Herrero y Villar, 2001b).
- **c)** Consistencia, exclusión y composición hacia arriba (Herrero y Villar, 2001a).

La regla proporcional es la única regla de reparto que verifica las propiedades de:

- **a)** Tratamiento igualitario, composición hacia arriba y autodualidad (Young, 1988).
- **b)** Tratamiento igualitario, composición hacia abajo y autodualidad (Herrero y Villar, 2001a).

La regla del Talmud es la única que verifica las propiedades:

- **a)** Invarianza al truncar demandas, consistencia y autodualidad (Herrero y Villar, 2001a).

**b)** Derechos mínimos primero, consistencia y autodualidad (Herrero y Villar, 2001a).

**c)** Aseguramiento, derechos mínimos primero y consistencia (Chun, 2006).

La regla de llegada aleatoria es la única que cumple:

**a)** Compensación equilibrada (Hwang, 2015).

| | Proporcional | Igual ganancia | Igual pérdida | Talmud | Llegada aleatoria |
|---|---|---|---|---|---|
| Tratamiento igualitario | ✓ | ✓ | ✓ | ✓ | ✓ |
| Anonimato | ✓ | ✓ | ✓ | ✓ | ✓ |
| Independencia de escala | ✓ | ✓ | ✓ | ✓ | ✓ |
| Composición hacia arriba | ✓ | ✓ | ✓ | | |
| Composición hacia abajo | ✓ | ✓ | ✓ | | |
| Consistencia | ✓ | ✓ | ✓ | ✓ | |
| Autodualidad | ✓ | | | ✓ | ✓ |
| Compensación completa | | ✓ | | | |
| Compensación vacía | | | ✓ | | |
| Derechos mínimos primero | | | ✓ | ✓ | ✓ |
| Aseguramiento | | ✓ | | ✓ | ✓ |
| Compensación equilibrada | | | | | ✓ |
| Conservación del orden | ✓ | ✓ | ✓ | ✓ | ✓ |
| Monotonía respecto al estado | ✓ | ✓ | ✓ | ✓ | ✓ |
| Monotonía respecto a las demandas | ✓ | ✓ | ✓ | ✓ | ✓ |
| Invarianza al truncar demandas | | ✓ | | ✓ | ✓ |

Tabla 1.4: Propiedades que verifica cada regla de reparto.

# Capítulo 2

# Problemas de bancarrota con bienes indivisibles

Como vimos en el Capítulo 1, un problema de bancarrota representa una situación en la que una cantidad de un bien debe ser dividida entre diferentes agentes cuyas demandas suman más que la cantidad del bien a repartir. En el Capítulo 1 el bien a repartir era perfectamente divisible, sin embargo, en muchas ocasiones, el bien a repartir es indivisible. Por ejemplo, los casos de listas de espera para las cirugías en hospitales, contratar a un cierto número de secretarias en una universidad, asignación de slots de tiempo en los aeropuertos para las diferentes aerolíneas, etc.

Para resolver los problemas de bancarrota con bienes indivisibles, se suelen utilizar las llamadas "prioridades"(Moulin, 2000). Dependiendo de como se definan estas prioridades, se obtienen diferentes familias de reglas de reparto. La primera familia de reglas de reparto se obtiene considerando un orden de prioridad definido sobre el conjunto de agentes. Se establece un orden en el conjunto de agentes y se va asignando a cada uno de ellos su demanda, hasta agotar el estado. Es decir, supongamos que van llegando uno a uno siguiendo el orden de prioridad, se le asigna lo que demandan hasta que ya no queda más estado. Dependiendo del orden de prioridad tenemos diferentes asignaciones. A estas reglas de reparto las llamaremos reglas de prioridad secuencial.

Otro tipo de orden de prioridad, introducido por Young (1994), se define sobre los pares agente-demanda. Estos órdenes de prioridad se llaman estándar de comparación. Dado un estándar de comparación, hay dos formas posibles de asignar el estado. La primera de ellas son las llamadas reglas hacia arriba, que asignan el estado unidad a unidad teniendo en cuenta el orden de prioridad de los pares agente-demanda. Otra forma de asignar el estado es mediante las reglas hacia abajo, que asignan el déficit unidad a unidad entre los agentes teniendo en cuenta el orden de prioridad.

Se pueden distinguir dos subclases de estándar de comparación. La primera subclase se enfoca en la demanda de un par agente-demanda. Los pares con demandas más elevadas tienen mayor prioridad. A esta subclase se le llama estándar de comparación orientado a las demandas. La segunda subclase se enfoca en el agente de un par agente-demanda. La idea es que cuando la componente demanda de dos pares agente-demanda son iguales, el único factor que importa es la identidad de los agentes. A esta subclase la llamamos estándar de comparación orientado a los agentes.

Otra forma de asignar el estado entre los agentes es utilizando las reglas de reparto vistas en el capítulo anterior adaptadas al caso de bienes indivisibles. Asignamos la parte entera de la asignación que nos dan las reglas y lo que falta por asignar se reparte entre los agentes que todavía no han recibido toda su demanda siguiendo un orden de prioridad dado. Aunque se podría llevar a cabo este método con cualquiera de las reglas del capítulo anterior, veremos lo que sucede con la regla de igual ganancia y con la regla de igual pérdida.

Un problema de bancarrota con bienes indivisibles se puede tratar también de forma diferente cuando los agentes en lugar de tener una demanda fija tienen una preferencia. La forma de resolver estos problemas es muy similar ya que se vuelven a utilizar prioridades.

## 2.1. Planteamiento y notación

Dado el conjunto de números naturales $\mathbb{N}$, $\mathcal{N}$ denota la familia de todos los subconjuntos finitos no vacíos de $\mathbb{N}$. En un problema de bancarrota con bienes indivisibles, una cantidad fija de unidades indivisibles, $E \in \mathbb{Z}_+$, a la que llamaremos estado, debe distribuirse en un grupo de agentes, $N \in \mathcal{N}$, que demandan una cierta cantidad del estado. Para cada $i \in N$, $c_i \in \mathbb{Z}_+$ denota la demanda del agente $i$-ésimo. El vector $c = (c_1, ..., c_n) \in \mathbb{Z}_+^N$ es el vector de demandas. Entonces, un problema de bancarrota con bienes indivisibles es un par $(E, c) \in \mathbb{Z}_+ \times \mathbb{Z}_+^N$ tal que la demanda agregada es mayor que el estado, $C = \sum_{i \in N} c_i \geq E$.

Se denota por $\mathbb{A}^N$ el conjunto de todos los problemas de bancarrota con un conjunto de agentes fijo $N$, y $\mathbb{A}$ denota el conjunto de todos los problemas, esto es,

$$\mathbb{A}^N = \{(E, c) \in \mathbb{Z}_+ \times \mathbb{Z}_+^N\}$$

y

$$\mathbb{A} = \bigcup_{N \in \mathcal{N}} \mathbb{A}^N.$$

Una regla de reparto para un problema de bancarrota con bienes indivisibles se define de forma similar a la vista para el caso de bienes divisibles. La diferencia es que en este caso imponemos que las asignaciones sean números enteros.

Una regla de reparto discreta es una función $F$ que asocia a cada par $(E, c)$ un vector de asignaciones $F(E, c) \in \mathbb{Z}_+^N$ tal que:

- $0 \leq F_i(E, c) \leq c_i$, $\forall i \in N$. Cada agente debe recibir una cantidad no negativa y no mayor que su demanda.
- $\sum_{i \in N} F_i(E, c) = E$. El estado debe ser repartido completamente.

## 2.2. Reglas de reparto

En esta sección veremos reglas de reparto que usan ordenes de prioridad predeterminados para asignar el estado.

### 2.2.1. Reglas de prioridad secuencial

Un orden de prioridad es una función $\sigma : \mathbb{N} \longrightarrow \mathbb{N}$ que asigna a cada agente un valor que representa su posición en el orden. Decimos que el agente $i$ tiene prioridad sobre el agente $j$, y lo denotaremos por $i\sigma j$, cuando $\sigma(i) < \sigma(j)$. $\Sigma$ denota el conjunto de todos los órdenes de prioridad en $\mathbb{N}$.

La regla de prioridad secuencial (*Sequential priority rule*) asociada al orden $\sigma$, $SP^\sigma$, asigna a cada agente su demanda de acuerdo al orden de prioridad. Es decir, imaginemos que los agentes van llegando de uno en uno según el orden de prioridad y se le va asignando su demanda hasta que el estado se agote.

Para cada $N \in \mathcal{N}$, cada $(E, c)$ y cada $i \in N$,

$$SP^\sigma(E, c) = \begin{cases} c_i & \text{si } i\sigma k \\ 0 & \text{si } k\sigma i \\ E - \sum_{j\sigma k} c_i & \text{si } i = k \end{cases}$$

donde $k \in \mathbb{N}$ se elige de forma que $E - \sum_{j\sigma k} c_j \in (0, c_k]$.

***Ejemplo*:** $N = \{1, 2, 3\}$, $E = 9$ y $c = (2, 6, 6)$. Supongamos además el siguiente orden de prioridad, $2\sigma 1\sigma 3$. Entonces, el agente 2 llega primero y le asignamos su demanda, $x_2 = 6$. Después llega el agente 1 y le asignamos su demanda, $x_1 = 2$. Por último llega el agente 3, su demanda es de 6 unidades pero sólo queda 1 unidad de estado por asignar, por lo que se le asigna la unidad restante, $x_3 = 1$. Entonces, la asignación obtenida mediante la regla de prioridad secuencial es

$$SP^\sigma(E, c) = (2, 6, 1).$$

### 2.2.2. Reglas asociadas al estándar de comparación

Un *estándar de comparación*, $\succ$, es un orden lineal sobre los pares agente-demanda tal que, si $i, j \in \mathbb{N}$ son dos agentes, $x, y \in \mathbb{Z}_+$ sus respectivas demandas y $(i, x) \succ (j, y)$, significa que el par que involucra al agente $i$ demandando $x$ unidades tiene preferencia sobre el par que involucra al agente $j$ demandando $y$ unidades. Además, para cada $i \in \mathbb{N}$ y cada $x \in \mathbb{Z}_+$, $(i, x + 1) \succ (i, x)$, es decir, para un agente fijo, tiene mayor prioridad el par con mayor demanda.

Se pueden distinguir dos subclases de estándar de comparación. La primera subclase se centra en la demanda de un par; los pares con mayor demanda tienen mayor prioridad. Llamaremos a esta subclase estándar de comparación orientado a demandas. La segunda subclase se centra en el agente de un par; cuando las componentes demanda de dos pares son iguales, el único factor que importa es la identidad de los agentes. Llamaremos a esta subclase estándar de comparación orientado a agentes.

- *Estándar de comparación asociado a demandas:* Es un estándar de comparación, tal que, para cada par $i, j \in \mathbb{N}$ y cada par $x, y \in \mathbb{Z}_+$, si $x > y$, entonces $(i, x) \succ (j, y)$. Los pares con demandas más altas tienen mayor prioridad sin importar la identidad del agente.

- *Estándar de comparación asociado a agentes:* Es un estándar de comparación, tal que, para cada par $i, j \in \mathbb{N}$ y cada par $x, y \in \mathbb{Z}_+$, si $(i, x) \succ (j, x)$, entonces $(i, y) \succ (j, y)$. En este caso, cuando las demandas de dos agentes coinciden, lo único que nos interesa es la identidad de los agentes para establecer la prioridad.

Dado un problema de bancarrota con bienes indivisibles, considerando un estándar de comparación específico, podemos ordenar todos los pares agente-demanda según su prioridad.

Hay dos maneras de resolver un problema usando un estándar de comparación. La primera, es asignar el estado unidad a unidad teniendo en cuenta el orden de prioridad de los pares agente-demanda. Son las llamadas reglas ascendentes. La segunda, es sustraer el déficit unidad a unidad teniendo en cuenta el orden de prioridad de los pares agente-demanda. Son las llamadas reglas descendentes.

#### Reglas ascendentes

Dado un estándar de comparación, podemos ordenar todos los pares agente-demanda de un problema particular. Una vez que tenemos todos los pares agente-demanda ordenados, asignamos una unidad al agente cuyo par sea el de mayor prioridad, y disminuimos su demanda en una unidad. Volvemos a ordenar los pares agente-demanda de acuerdo con $\succ$, asignamos una unidad al agente cuyo par sea el de mayor prioridad y disminuimos su demanda en una unidad. Repetimos este proceso hasta que el estado esté totalmente asignado.

***Ejemplos:***

- **Regla ascendente asociada al estándar de comparación.** Sea $N = \{1, 2, 3\}$, $E = 6$ y $c = (2, 5, 3)$. Consideramos el estándar de comparación que ordena los pares agente-demanda de la siguiente forma:

$$(2, 5) \succ (2, 4) \succ (3, 3) \succ (2, 3) \succ (3, 2) \succ (1, 2) \succ (3, 1) \succ (2, 2) \succ (2, 1) \succ (1, 1).$$

De acuerdo con $\succ$ el par con la mayor prioridad es $(2,5)$, por lo que le asignamos una unidad del estado al agente 2 y reducimos en una unidad su demanda. El siguiente par con la mayor prioridad es el $(2,4)$, por lo que le asignamos otra unidad del estado al agente 2 y reducimos su demanda en una unidad. Seguimos con el proceso hasta asignar el estado completamente. En la siguiente tabla tenemos resumido todo el proceso:

| $E$ | $x^{(k)}$ | $c^{(k)}$ | Par con la prioridad más alta |
|---|---|---|---|
| 0 | $(0,0,0)$ | $(2,5,3)$ | $(2,5)$ |
| 1 | $(0,1,0)$ | $(2,4,3)$ | $(2,4)$ |
| 2 | $(0,2,0)$ | $(2,3,3)$ | $(3,3)$ |
| 3 | $(0,2,1)$ | $(2,3,2)$ | $(2,3)$ |
| 4 | $(0,3,1)$ | $(2,2,2)$ | $(3,2)$ |
| 5 | $(0,3,2)$ | $(2,2,1)$ | $(1,2)$ |
| 6 | $(1,3,2)$ | | |

La primera columna muestra la $k$-ésima unidad del estado. Empezando por 0, asignamos unidad a unidad hasta llegar a 6 unidades. La segunda columna muestra la asignación hasta esa unidad, $x^{(k)}$. La tercera columna muestra el vector de demandas actualizado, $c^{(k)}$. Finalmente, la cuarta columna muestra el par con la prioridad más alta en cada etapa del proceso.

- **Regla ascendente orientada a demandas.** Sea $N = \{1,2,3\}$, $E = 6$ y $c = (2,5,3)$. Consideramos el estándar de comparación orientado a demandas. Además, consideramos $(3,x) \succ (1,x) \succ (2,x)$ si $x$ es par, y $(1,x) \succ (3,x) \succ (2,x)$ si $x$ es impar. De acuerdo con $\succ$, el par con la mayor prioridad es $(2,5)$, por lo que le asignamos una unidad del estado al agente 2 y reducimos en una unidad su demanda. Ahora, el par con la mayor prioridad es el $(2,4)$, por lo que asignamos otra unidad al agente 2 y reducimos en otra unidad su demanda. Se sigue el mismo procedimiento hasta asignar completamente el estado. En la siguiente tabla se recoge todo el procedimiento:

| $E$ | $x^{(k)}$ | $c^{(k)}$ | Par con la prioridad más alta |
|---|---|---|---|
| 0 | $(0,0,0)$ | $(2,5,3)$ | $(2,5)$ |
| 1 | $(0,1,0)$ | $(2,4,3)$ | $(2,4)$ |
| 2 | $(0,2,0)$ | $(2,3,3)$ | $(3,3)$ |
| 3 | $(0,2,1)$ | $(2,3,2)$ | $(2,3)$ |
| 4 | $(0,3,1)$ | $(2,2,2)$ | $(3,2)$ |
| 5 | $(0,3,2)$ | $(2,2,1)$ | $(1,2)$ |
| 6 | $(1,3,2)$ | | |

La primera columna muestra la $k$-ésima unidad del estado. Empezando por 0, asignamos unidad a unidad hasta llegar a 6 unidades. La segunda columna muestra la asignación hasta esa unidad, $x^{(k)}$. La tercera columna muestra el vector de demandas actualizado, $c^{(k)}$. Finalmente, la cuarta columna muestra el par con la prioridad más alta en cada etapa del proceso.

- **Regla ascendente orientada a agentes.** Sea $N = \{1, 2, 3\}$, $E = 6$ y $c = (2, 5, 3)$. Consideramos el estándar de comparación orientado a agentes que ordena los pares agente-demanda de la siguiente manera:

$$(1, 2) \succ (2, 5) \succ (2, 4) \succ (3, 3) \succ (1, 1) \succ (2, 3) \succ (3, 2) \succ (2, 2) \succ (3, 1) \succ (2, 1).$$

De acuerdo con $\succ$ el par con la mayor prioridad es $(1, 2)$, por lo que le asignamos una unidad del estado al agente 1 y reducimos en una unidad su demanda. El siguiente par con la mayor prioridad es el $(2, 5)$, por lo que le asignamos una unidad del estado al agente 2 y reducimos su demanda en una unidad. Seguimos con el proceso hasta asignar el estado completamente. En la siguiente tabla podemos ver todo el proceso:

| $E$ | $x^{(k)}$ | $c^{(k)}$ | Par con la prioridad más alta |
|---|---|---|---|
| 0 | $(0, 0, 0)$ | $(2, 5, 3)$ | $(1, 2)$ |
| 1 | $(1, 0, 0)$ | $(1, 5, 3)$ | $(2, 5)$ |
| 2 | $(1, 1, 0)$ | $(1, 4, 3)$ | $(2, 4)$ |
| 3 | $(1, 2, 0)$ | $(1, 3, 3)$ | $(3, 3)$ |
| 4 | $(1, 2, 1)$ | $(1, 3, 2)$ | $(1, 1)$ |
| 5 | $(2, 2, 1)$ | $(0, 3, 2)$ | $(2, 3)$ |
| 6 | $(2, 3, 1)$ | | |

Al igual que en los ejemplos anteriores, la primera columna muestra la $k$-ésima unidad del estado. Empezando por 0, asignamos unidad a unidad hasta llegar a 6 unidades. La segunda columna muestra la asignación hasta esa unidad, $x^{(k)}$. La tercera columna muestra el vector de demandas actualizado, $c^{(k)}$. Finalmente, la cuarta columna muestra el par con la prioridad más alta en cada etapa del proceso.

**Regla descendente**

Empezamos asignando a cada agente su demanda y vamos asignando el déficit unidad a unidad.

Una vez que se tienen todos los pares agente-demanda ordenandos según su prioridad, asignamos una unidad de déficit al agente cuyo par tenga mayor prioridad (siempre que sea posible, no podemos asignar déficit si la demanda es de cero unidades), i.e., le quitamos una unidad al agente cuyo par tenga mayor prioridad. Volvemos a ordenar todos los pares agente-demanda de acuerdo con $\succ$, y le quitamos una unidad al agente cuyo par tenga mayor prioridad. Repetimos este proceso hasta asignar completamente el estado.

***Ejemplos:***

- **Regla descendente asociada al estándar de comparación.** Sea $N = \{1, 2, 3\}$, $E = 6$ y $c = (2, 5, 3)$. Consideramos el estándar de comparación que ordena los pares agente-demanda de la siguiente forma:

$$(2,5) \succ (2,4) \succ (3,3) \succ (2,3) \succ (3,2) \succ (1,2) \succ (3,1) \succ (2,2) \succ (2,1) \succ (1,1).$$

Empezamos asignando a cada agente su demanda y vamos restando unidad a unidad el déficit. De acuerdo con $\succ$ el par con la mayor prioridad es $(2,5)$, por lo que le restamos una unidad a la asignación del agente 2. El siguiente par con la mayor prioridad es el $(2,4)$, por lo que le restamos otra unidad a la asignación del agente 2. Seguimos con el proceso hasta asignar completamente el déficit. En la siguiente tabla tenemos resumido todo el proceso:

| $E$ | $x^{(k)}$ | Par con la prioridad más alta |
|---|---|---|
| 10 | $(2,5,3)$ | $(2,5)$ |
| 9 | $(2,4,3)$ | $(2,4)$ |
| 8 | $(2,3,3)$ | $(3,3)$ |
| 7 | $(2,3,2)$ | $(2,3)$ |
| 6 | $(2,2,2)$ | |

En la primera columna aparece el estado que se estaría repartiendo en cada etapa. En la segunda columna se muestra la asignación en cada etapa y en la tercera columna, se muestra el par agente demanda con la prioridad más alta.

- **Regla descendente orientada a demandas.** Sea $N = \{1,2,3\}$, $E = 6$ y $c = (2,5,3)$. Consideramos el estándar de comparación orientado a demandas. Además, consideramos $(3,x) \succ (1,x) \succ (2,x)$ si $x$ es par, y $(1,x) \succ (3,x) \succ (2,x)$ si $x$ es impar. Empezamos asignando a cada agente su demanda y vamos restando unidad a unidad el déficit. De acuerdo con $\succ$ el par con la mayor prioridad es $(2,5)$, por lo que le restamos una unidad a la asignación del agente 2 y le restamos una unidad a su demanda. Ahora, el par con mayor prioridad es el $(2,4)$, por lo que le restamos otra unidad a la asignación del agente 2 y a su demanda. Seguimos con el proceso hasta asignar completamente el déficit. En la siguiente tabla tenemos resumido todo el proceso:

| $E$ | $x^{(k)}$ | Par con la prioridad más alta |
|---|---|---|
| 10 | $(2,5,3)$ | $(2,5)$ |
| 9 | $(2,4,3)$ | $(2,4)$ |
| 8 | $(2,3,3)$ | $(3,3)$ |
| 7 | $(2,3,2)$ | $(2,3)$ |
| 6 | $(2,2,2)$ | |

En la primera columna aparece el estado que se estaría repartiendo en cada etapa. En la segunda columna se muestra la asignación en cada etapa y en la tercera columna, se muestra el par agente demanda con la prioridad más alta.

- **Regla descendente orientada a agentes.** Sea $N = \{1, 2, 3\}$, $E = 6$ y $c = (2, 5, 3)$. Consideramos el estándar de comparación orientado a agentes que ordena los pares agente-demanda de la siguiente forma:

$$(1, 2) \succ (2, 5) \succ (2, 4) \succ (3, 3) \succ (1, 1) \succ (2, 3) \succ (3, 2) \succ (2, 2) \succ (3, 1) \succ (2, 1).$$

Empezamos asignando a cada agente su demanda y vamos restando unidad a unidad el déficit. De acuerdo con $\succ$ el par con la mayor prioridad es $(2, 5)$, por lo que le restamos una unidad a la asignación del agente 2. El siguiente par con la mayor prioridad es el $(2, 4)$, por lo que le restamos otra unidad a la asignación del agente 2. Seguimos con el proceso hasta asignar completamente el déficit. En la siguiente tabla tenemos resumido todo el proceso:

| $E$ | $x^{(k)}$ | Par con la prioridad más alta |
|---|---|---|
| 10 | $(2, 5, 3)$ | $(1, 2)$ |
| 9 | $(1, 5, 3)$ | $(2, 5)$ |
| 8 | $(1, 4, 3)$ | $(2, 4)$ |
| 7 | $(1, 3, 3)$ | $(3, 3)$ |
| 6 | $(1, 3, 2)$ | |

En la primera columna aparece el estado que se estaría repartiendo en cada etapa. En la segunda columna se muestra la asignación en cada etapa y en la tercera columna, se muestra el par agente demanda con la prioridad más alta.

### 2.2.3. Regla de igual ganancia discreta

De la misma forma que en la regla de igual ganancia vista en el capítulo anterior, el objetivo es repartir el estado de la forma más igualitaria posible, por lo que se puede utilizar dicha regla para obtener la asignación en este caso.

El proceso de asignación se lleva a cabo en dos etapas. Dado el problema de bancarrota con bienes indivisibles $(E, c)$, en la primera etapa asignamos al agente $i$-ésimo, $\lfloor CEA_i(E, c)\rfloor$ unidades, es decir, la parte entera de la asignación que le correspondería al usar la regla de igual ganancia. Si en esta etapa aún quedan unidades por repartir ($E' = E - \sum_{i\in N}\lfloor CEA_i(E, c)\rfloor > 0$) vamos a la segunda etapa.

Podemos distinguir dos tipos de agentes: aquellos que ya han recibido una cantidad entera de acuerdo con $CEA$, es decir, $CEA_i(E, c) \in \mathbb{Z}_+$ (y entonces $\lfloor CEA_i(E, c)\rfloor = CEA_i(E, c)$); y aquellos agentes cuya asignación dada por $CEA$ no es un número entero. Denotaremos por $Q(CEA; E, c)$ este último grupo de agentes: $Q(CEA; E, c) = \{j \in N : CEA_j(E, c) \notin \mathbb{Z}_+\}$.

Dado un orden de prioridad $\sigma \in \Sigma$. En la segunda etapa distribuimos las $E'$ unidades restantes entre los agentes de $Q(CEA; E, c)$. Daremos una unidad extra a los $E'$ agentes con mayor prioridad en $Q(CEA; E, c)$ de acuerdo al orden $\sigma$. Denotamos por $Q^\sigma(CEA; E, c)$ el conjunto ordenado $Q(CEA; E, c)$. Denotamos por $Q^\sigma_a(CEA; E, c)$ al conjunto de los $a$ primeros agentes en $Q^\sigma(CEA; E, c)$.

Formalmente,

$$CEA^\sigma_i(E, c) = \begin{cases} \lfloor CEA_i(E, c)\rfloor + 1 & \text{si } i \in Q^\sigma_{E'}(CEA; E, c) \\ \lfloor CEA_i(E, c)\rfloor & \text{en otro caso} \end{cases}$$

donde $E' = E - \sum_{i \in N} \lfloor CEA_i(E,c) \rfloor > 0$.

***Ejemplo:*** Consideremos un problema de bancarrota con bienes indivisibles en el que $N = \{1,2,3\}$, $E = 9$ y $c = (2,6,6)$. La regla de igual ganancia vista en el capítulo anterior nos da la siguiente asignación, $CEA(E,c) = (2, 3.5, 3.5)$. Nos quedamos con la parte entera, $\lfloor CEA(E,c) \rfloor = (2,3,3)$. Todavía nos queda una unidad por asignar. Podemos dársela al agente 2 o al agente 3, ya que el agente 1 tiene asignada su demanda completa. Aquí es donde entra el orden de prioridad. Si $2\sigma3$, le asignamos la unidad restante al agente 2, en caso contrario, se la asignamos al agente 3. Tendríamos entonces dos posibles soluciones, $CEA^\sigma(E,c) = (2,4,3)$ o $CEA^\sigma(E,c) = (2,3,4)$, respectivamente.

### 2.2.4. Regla de igual pérdida discreta

Igual que en la regla de igual pérdida vista en el capítulo anterior, se pretende asignar el estado de forma que las pérdidas de los agentes sean lo más iguales posible.

Dado un problema de bancarrota con bienes indivisibles, la asignación final se puede obtener en dos etapas. En la primera etapa, cada agente $i \in N$ recibe $\lfloor CEL_i(E,c) \rfloor$ unidades, es decir, la parte entera de la asignación que le correspondería al usar la regla de igual pérdida. Si todavía quedan unidades por repartir ($E' = E - \sum_{i \in N} \lfloor CEL(E,c) \rfloor > 0$) vamos a la segunda etapa.

Dividimos el conjunto de agentes en dos grupos: aquellos que ya han recibido una cantidad entera de acuerdo con $CEL$, es decir, $CEL_i(E,c) \in \mathbb{Z}_+$ (entonces $\lfloor CEL_i(E,c) \rfloor = CEL_i(E,c)$); y los que no. $Q(CEL; E, c) = \{j \in N : CEL_j(E,c) \notin \mathbb{Z}_+\}$ denota este último subconjunto de agentes.

En la segunda etapa distribuimos las $E'$ unidades restantes entre los agentes de $Q(CEL; E, c)$. Daremos una unidad extra a los $E'$ agentes con la mayor prioridad en $Q(CEL; E, c)$ de acuerdo al orden $\sigma$. Denotamos por $Q_a^\sigma(CEL; E, c)$ a conjunto de los $a$ primeros agentes en $Q^\sigma(CEL; E, c)$.

Formalmente,

$$CEL_i^\sigma(E,c) = \begin{cases} \lfloor CEL_i(E,c) \rfloor + 1 & \text{si } i \in Q_{E'}^\sigma(CEL; E, c) \\ \\ \lfloor CEL_i(E,c) \rfloor & \text{en otro caso} \end{cases}$$

donde $E' = E - \sum_{i \in N} \lfloor CEL_i(E,c) \rfloor > 0$.

***Ejemplo:*** Consideremos $N = \{1,2,3\}$, $c = (2,4,6)$ y $E = 5$. La demanda agregada es de 12 unidades y el estado es de 5 unidades, por lo que debemos distribuir 7 unidades de déficit entre los agentes. Empezamos aplicando la regla de igual pérdida vista en el capítulo anterior. Obtenemos así $CEL(E,c) = (0, 1.5, 3.5)$ y nos quedamos con la parte entera, $\lfloor CEL(E,c) \rfloor = (0,1,3)$. Nos queda por asignar una unidad del estado entre los agentes 2 y 3. Aquí es donde entra el orden de prioridad. Si $2\sigma3$, le asignamos la unidad restante al agente dos, en caso contrario, se la asignamos al agente 3. Tendríamos entonces dos posibles soluciones, $CEL^\sigma(E,c) = (0,2,3)$ o $CEL^\sigma(E,c) = (0,1,4)$, respectivamente.

## 2.3. Propiedades de las reglas de reparto

En esta sección se verán algunas de las propiedades que cumplen las reglas de reparto que se acaban de introducir. Algunas de estas propiedades ya han sido estudiadas en el caso de bienes divisibles. Otras propiedades, en cambio, requieren una modificación para que tengan sentido en el caso de bienes indivisibles.

Las propiedades de composición hacia arriba, composición hacia abajo, consistencia, compensación completa, compensación vacía, conservación del orden, monotonía respecto al estado y monotonía respecto a las demandas se enuncian de manera equivalente para el caso de bienes indivisibles por lo tanto se remite su definición a la Sección 1.3.

A continuación se introducen dos nuevas propiedades específicas para el caso de bienes indivisibles.

## Equilibrio

Es una adaptación de la propiedad de tratamiento igualitario vista para el caso de problemas de bancarrota con bienes divisibles. La regla de tratamiento igualitario nos dice que dos agentes con la misma demanda deben recibir la misma asignación. Pero en el caso de los problemas de bancarrota con bienes indivisibles esto no siempre se cumple. Supongamos que tenemos un problema con dos agentes que demandan lo mismo y que $E = 1$. Sólo uno de los agentes puede recibir la unidad del estado. Entonces se hace una modificación en la propiedad para permitir una diferencia de una unidad entre las asignaciones de los agentes con la misma demanda.

Formalmente, dado un problema de bancarrota con bienes indivisibles $(E, c)$ y dos agentes $i, j \in N$ tal que $c_i = c_j$, decimos que $F$ verifica la propiedad de equilibrio si

$$|F_i(E, c) - F_j(E, c)| \leq 1.$$

## Monotonía vinculada demanda-estado

Esta propiedad se aplica a situaciones en las que el estado y la demanda de un agente se incrementan simultáneamente.

Se requiere que cualquier agente cuya demanda aumenta cuando la pérdida agregada permanezca sin cambios pierda al menos tantas unidades como lo hizo inicialmente.

Dado un problema de bancarrota con bienes indivisibles $(E, c)$ y in agente $i \in N$ tal que $c'_i > c_i$, entonces,

$$c_i - F_i(E, c) \leq c'_i - F_i(E', (c_{-i}, c'_i)).$$

Una vez introducidas las nuevas propiedades, veamos que propiedades cumple cada regla de reparto.

- La **regla de prioridad secuencial** satisface las propiedades de composición hacia arriba, composición hacia abajo, consistencia, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla ascendente asociada al estándar de comparación** satisface las propiedades de composición hacia arriba, consistencia, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla de descendente asociada al estándar de comparación:** satisface las propiedades de composición hacia abajo, consistencia, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla ascendente orientada a demandas** satisface las propiedades de equilibrio, composición hacia arriba, consistencia, compensación vacía, conservación del orden, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla descendente orientada a demandas** satisface las propiedades de equilibrio, composición hacia abajo, consistencia, compensación completa, conservación del orden, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla ascendente orientada a agentes** satisface las propiedades de composición hacia arriba, consistencia, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla descendente orientada a agentes** satisface las propiedades de composición hacia abajo, consistencia, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla de igual ganancia discreta** satisface las propiedades de equilibrio, composición hacia abajo, consistencia, compensación completa, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

- La **regla de igual pérdida discreta** satisface las propiedades de equilibrio, composición hacia arriba, consistencia, compensación vacía, monotonía respecto al estado, monotonía respecto a las demandas y monotonía vinculada demanda-estado.

En la tabla 2.1 se recoge de forma resumida las propiedades que cumple cada una de las reglas de reparto vistas en este capítulo.

## 2.4. Caracterización

Combinando distintas propiedades de las mencionadas, se caracterizan las reglas de reparto vistas anteriormente.

Las reglas de prioridad secuencial son las únicas reglas que satisfacen:

- Composición hacia arriba, composición hacia abajo y consistencia (Moulin, 2000).

Las reglas ascendentes se caracterizan por las propiedades de:

- Monotonía de las demandas, composición hacia arriba y consistencia (Chen, 2015).

Las reglas descendentes se caracterizan por las propiedades de:

- Monotonía vinculada demanda-estado, composición hacia abajo y consistencia (Moulin y Stong, 2002).
- Compensación completa, composición hacia abajo y consistencia (Herrero y Martínez, 2008a).
- Equilibrio, monotonía vinculada demanda-estado, composición hacia abajo y consistencia.

Las reglas ascendentes orientadas a demandas se caracterizan por las propiedades de:

- Compensación vacía, composición hacia arriba y consistencia (Chen, 2015).

Las reglas descendentes orientadas a demandas se caracterizan por las propiedades de:

- Equilibrio, monotonía vinculada demanda-estado, composición hacia abajo y consistencia (Chen, 2015).
- Compensación completa, composición hacia abajo y consistencia (Herrero y Martínez, 2008a).

La regla de igual ganancia discreta es la única regla que verifica las propiedades de

- Equilibrio, compensación completa y composición hacia abajo (Herrero y Martínez, 2004).

La regla de igual pérdida discreta es la única regla que verifica las propiedades de

- Equilibrio, compensación vacía y composición hacia arriba (Herrero y Martínez, 2004).

| | Regla de prioridad secuencial | Regla ascendente | Regla descendente | Regla ascendente orientada a demandas | Regla descendente orientada a demandas | Regla ascendente orientada a agentes | Regla descendente orientada a agentes | Regla de igual ganancia discreta | Regla de igual pérdida discreta |
|---|---|---|---|---|---|---|---|---|---|
| Equilibrio | | | | ✓ | ✓ | | | ✓ | ✓ |
| Composición hacia arriba | ✓ | ✓ | | ✓ | | ✓ | | | ✓ |
| Composición hacia abajo | ✓ | | ✓ | | ✓ | | ✓ | ✓ | |
| Consistencia | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Compensación completa | | | | | ✓ | | | ✓ | |
| Compensación vacía | | | | ✓ | | | | | ✓ |
| Conservación del orden | | | | ✓ | ✓ | | | | |
| Monotonía respecto al estado | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Monotonía respecto a las demandas | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Monotonía vinculada demanda-estado | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

Tabla 2.1: Propiedades que cumple cada una de las reglas de reparto.

## 2.5. Preferencias

Consideremos ahora un problema de bancarrota con bienes indivisibles en el que los agentes tienen una preferencia en lugar de tener una demanda (Herrero y Martínez, 2008b). Esta situación surge cuando hay una tarea que debe realizar un grupo de agentes y el tiempo para realizar la tarea viene en unidades indivisibles. Cada agente tiene una oferta de trabajo preferida, llamada pico, y tener que trabajar más o menos que la cantidad ideal disminuye su utilidad. Además, el total de unidades de tiempo necesarias para realizar la tarea deben ser distribuidas.

Ejemplos de este tipo de situación son la asignación de turnos entre médicos o enfermeras en centros de salud, asignación de vuelos entre miembros de la tripulación en una compañía aérea, o incluso asignación de horas de clase entre los profesores de un departamento, entre otros.

Los médicos, miembros de la tripulación o profesores universitarios en los ejemplos anteriores se llaman agentes; el número de unidades indivisibles que se asignarán (turnos, vuelos, horas de clase) se denomina tarea. Finalmente, también tenemos información sobre el perfil de preferencias de cada uno de los agentes, donde se supone que hay un pico, que es la opción preferida y, a medida que se alejas de dicho pico, su utilidad disminuye. Este tipo de preferencias se denominan de un solo pico. Cuando la oferta agregada de trabajo (es decir, la suma de las cantidades preferidas) es diferente de la tarea a realizar, se debe idear algún procedimiento para asignar la diferencia.

Formalmente, sea $\mathbb{N}$ una población infinita de agentes potenciales y $\mathcal{N}$ la familia de todos los subconjuntos no vacíos finitos de $\mathbb{N}$. Para un agente $i \in \mathbb{N}$, sea $R_i$ una preferencia sobre $\mathbb{Z}_+$, y sea $P_i$ la preferencia estricta asociada. Decimos que $R_i$ tiene un solo pico si existe un entero $p(R_i) \in \mathbb{Z}_+$ (llamado el pico de $R_i$) tal que, para cada $a$, $b \in \mathbb{Z}_+$,

$$aP_ib \Leftrightarrow [(b < a < p(R_i)) \text{ o } (p(R_i) < a < b)].$$

Es decir, $a$ será preferido a $b$ si se encuentra más cerca del pico.

$\mathbb{S}$ denota la clase de todas la preferencias de un solo pico definidas sobre $\mathbb{Z}_+$.

Un problema de asignación con preferencias de un solo pico es una terna $e = (N, T, R)$ en la cual un conjunto de agentes, $N = \{1, ..., n\}$, cuyas preferencias se extraen de $\mathbb{S}^N$, deben proporcionar conjuntamente una serie de unidades de trabajo $T$, llamadas tarea. $\mathbb{A}^N$ denota la clase de problemas con un conjunto de agentes fijo $N$, y $\mathbb{A}$ la clase de todos los problemas, esto es,

$$\mathbb{A}^N = \{(N, T, R) \in \{N\} \times \mathbb{Z}_+ \times \mathbb{S}^N\}$$

y

$$\mathbb{A} = \bigcup_{N \in \mathcal{N}} \mathbb{A}^N.$$

Para cada problema dado, una asignación es una distribución de la tarea entre los agentes. Entonces, una asignación es un vector , $x \in \mathbb{Z}_+^N$, tal que $\sum_{i \in N} x_i = T$. Una asignación es *eficiente* si no hay otra asignación en la que todos los agentes estén en mejores condiciones. Esto es, para un problema dado, una asignación $y$ es eficiente si no hay otra asignación $x$ tal que, para cada $i \in N$, $x_iR_iy_i$, y para algún $j \in N$, $x_jP_jy_j$. Denotaremos por $P(e)$ el conjunto de todas las asignaciones eficientes para $e \in \mathbb{A}$.

### 2.5.1. Reglas de reparto

Una *regla de reparto* es una forma eficiente de asignar la tarea entre los agentes, esto es, es una función, $F : \mathbb{A} \longrightarrow \bigcup_{e \in \mathbb{A}} P(e)$, que selecciona, para cada problema $e \in \mathbb{A}$, una única asignación $F(e) \in P(e)$.

Las reglas, como se definen arriba, recomiendan que todos los agentes estén localizados en el mismo lado de sus picos. Esto es, $F(e) \leq p(R)$ si $\sum_{i \in N} p(R_i) \geq T$, y $F(e) \geq p(R)$, en otro caso. Es decir, si estamos ante un problema de exceso de oferta de mano de obra, en asignaciones eficientes todos deberían trabajar menos que su cantidad preferida, y si el problema es de exceso de demanda, todos los agentes deberían que trabajar más de lo que les gustaría.

Una forma natural de resolver problemas de asignación con indivisibilidades cuando las preferencias son de un solo pico, es considerar un proceso secuencial en el que la tarea se asigna unidad por unidad. Para hacer eso, usamos una especie de orden, llamado estándar de prioridad.

El estándar de prioridad, $\succ$, es un orden lineal en el conjunto de todos los pares $(i, a) \in \mathbb{N} \times \mathbb{Z}$ de manera que, para todos $i$, $j \in \mathbb{N}$ y todo $a \in \mathbb{Z}$, $(i, a+1) \succ (j, a)$. Es decir, los pares con números más altos tienen prioridad.

Para cualquier problema dado y para cada agente en $e$, podemos considerar todos los pares que involucran a ese agente junto con cualquier número menor o igual que su oferta de mano de obra preferida. El estándar de prioridad entonces ordena todos esos pares. Ahora, asignamos la tarea por unidad según el estándar de prioridad. Diferentes estándares de prioridad dan lugar a diferentes formas de resolver problemas, es decir, diferentes reglas. Llamamos a estas reglas métodos ascendentes.

**Método ascendente asociado a $\succ$, $U^{\succ}$:** Sea $e \in \mathbb{A}$. Para cada agente en $e$, se considera su oferta de trabajo preferida en $e$, es decir, su pico $p(R_i)$, y se toman todos los pares $\{(i, a_i)\}_{i \in N}$ con $a_i \leq p(R_i)$. Se identifica el par con la prioridad más alta, de acuerdo con *succ*, y se asigna la primera unidad de la tarea al agente en ese par. Se elimina este par, se identifica el agente en el par con la siguiente prioridad más alta y se le asigna la segunda unidad de la tarea. Se procede de la misma manera hasta tener distribuida toda la tarea.

***Ejemplo*:** Sea $\succ$ un estándar de prioridad tal que, restringido a los agentes en $N = \{1, 2, 3\}$, $(1, x) \succ (2, x) \succ (3, x)$ si $x$ es impar, y $(1, x) \succ (3, x) \succ (2, x)$ si $x$ es par. Ahora, consideremos el problema donde $N = \{1, 2, 3\}$, $T = 8$, y $R = (R_1, R_2, R_3)$ son tales que $R_2 = R_3$ y $p(R) = (1, 3, 3)$. Entonces tomamos todos los pares $\{(1, 1), (1, 0), (1, -1), (1, -2), ..., (2, 3), (2, 2), (2, 1), (2, 0), (2, -1), (2, -2), ..., (3, 3), (3, 2), (3, 1), (3, 0), (3, -1), (3, -2), ...\}$. De acuerdo con *succ*, el par con la mayor prioridad es $(2, 3)$. Entonces asignamos la segunda unidad al agente 3, y así sucesivamente. Repitiendo este proceso hasta asignar las 8 unidades pedidas, concluimos que $U^{\succ}(e) = (2, 3, 3)$. La siguiente tabla presenta el procedimiento.

| Par con mayor prioridad | Asignación parcial |
|---|---|
| $(2, 3)$ | $(0, 1, 0)$ |
| $(3, 3)$ | $(0, 1, 1)$ |
| $(3, 2)$ | $(0, 1, 2)$ |
| $(2, 2)$ | $(0, 2, 2)$ |
| $(1, 1)$ | $(1, 2, 2)$ |
| $(2, 1)$ | $(1, 3, 2)$ |
| $(3, 1)$ | $(1, 3, 3)$ |
| $(1, 0)$ | $(2, 3, 3)$ |

### 2.5.2. Propiedades

En esta sección se verán las propiedades que cumplen las reglas ascendentes aplicadas a los problemas de asignación con indivisibilidades cuando las preferencias son de un solo pico.

**Equidad**

Esta propiedad es similar a la propiedad de equilibrio vista en el caso de problemas de demandas con bienes indivisibles. Al igual que en ese caso, no se puede tratar agentes con preferencias iguales de tal manera que reciban la misma parte de la tarea. En su lugar, podemos pedir que se asigne a agentes con preferencia iguales la cantidad más parecida posible.

Para cada $e \in \mathbb{A}$ y cada $\{i, j\} \subseteq N$, si $R_i = R_j$ entonces

$$|F_i(e) - F_j(e)| \leq 1.$$

**Pico único**

Las preferencias de los agentes son información privada y difícil obtener. Pico único es una condición de eficiencia informativa. Requiere que las asignaciones de los agentes dependan solo de sus picos.

Para cada $e = (N, T, (R_i, R_{-i})) \in \mathbb{A}$ y cada $e' = (N, T, (R'_i, R'_{-i})) \in \mathbb{A}$ tal que $p(R'_i) = p(R_i)$,

$$F_i(e) = F_i(e').$$

**Consistencia**

La consistencia se refiere a una situación en la que se ha realizado una distribución tentativa de la tarea, y un grupo de agentes dejan el problema después de aceptar su parte de la tarea. Esta propiedad indica que el problema reducido debe resolverse de tal manera que a todos los agentes restantes se les asignen exactamente los mismos deberes que antes.

Dado un problema $(N, T, R) \in \mathbb{A}$ y un subconjunto $S \in N$, para cada $i \in N \backslash S$,

$$F_i(N, T, R) = F_i(N \backslash S, \sum_{i \in S} F_i(N, T, R), R_{N \backslash S}).$$

**Independencia de agenda**

Se tiene que dividir una tarea que viene dada en dos partes, $T_1$ y $T_2$, entre los agentes. Al final deben asignarse $T = T_1 + T_2$ unidades. Hay dos posibilidades de resolver el problema. La primera es asignar directamente las $T$ unidades. La segunda posibilidad es distribuir primero $T_1$, ajustar las preferencias de los agentes teniendo en cuenta lo que ya han recibido y después distribuir $T_2$. La independencia de agenda nos dice que los agentes deben recibir lo mismo independientemente de la opción elegida.

Para cada $e = (N, T, R) \in \mathbb{A}$, sean $T_1$, $T_2 \in \mathbb{Z}_{++}$ tal que $T_1 + T_2 = T$. Entonces

$$F(N, T, R) = F(N, T_1, R) + F(N, T_2, R'),$$

donde $R'$ son las preferencias modificadas una vez repartido $T_1$.

Esta propiedad es equivalente a la propiedad de composición hacia arriba vista en el capítulo anterior.

**R-dummy**

Esta propiedad es similar a la propiedad de compensación vacía considerada en los problemas clásicos de bancarrota.

Si estamos en una situación de exceso de oferta, i.e., $T < \sum_{j \in N} p(R_j)$, y para un agente particular $i$, su oferta es tan pequeña que incluso si consideramos $n$ agentes idénticos al agente $i$ y sumamos su oferta agregada a la tarea, los agentes aún tendrían que ser racionados en el trabajo, i.e., $T + np(R_i) < \sum_{j \in N} p(R_j)$. La propiedad r-dummy requiere que se ignore la oferta del agente $i$ y que el problema se resuelva considerando solo los agentes restantes, o, en otras palabras, al agente $i$ se le asignan cero unidades.

Para cada $e = (N, T, R) \in \mathbb{A}$ y cada $i \in N$, si $T + np(R_i) < \sum_{j \in N} p(R_j)$, entonces $F_i(e) = 0$.

Las reglas ascendentes aplicadas a los problemas de asignación con indivisibilidades cuando las preferencias son de un solo pico satisfacen todas las propiedades mencionadas anteriormente. Además, las reglas ascendentes son las únicas reglas que las satisfacen, por lo que se caracterizan por estas propiedades (Herrero y Martínez, 2008b).

# Capítulo 3

# Aplicación práctica: Reparto de plazas de titular entre las áreas en la Universidade de Vigo

Cada año, la Universidade de Vigo presenta la convocatoria de oferta de empleo público de personal docente e investigador. En dicha convocatoria, se establece el número de plazas de profesores titulares de universidad que saldrán a concurso ese año.

Para poder optar a las plazas de profesores titulares de universidad es necesario estar acreditado para dicha figura de profesorado por la ANECA (Agencia Nacional de Evaluación de la Calidad y Acreditación). En la actualidad, el número de profesores acreditados sobrepasa el número de plazas convocadas y por ello es necesario establecer un mecanismo a la hora de decidir en que áreas de conocimiento se convocarán dichas plazas. Por lo tanto, nos encontramos ante un problema de bancarrota donde tanto las demandas (profesores acreditados en el área) como el estado (plazas convocadas) son indivisibles y así ha de ser también la propuesta de reparto de plazas.

A la hora de seleccionar las áreas, la universidad establece el criterio de convocar las plazas en aquellas áreas donde se observa un mayor déficit de profesorado funcionario, y con este fin calcula un indicador $Q$ que relaciona la carga docente del área (número de horas de docencia impartidas durante el curso, denotada por $H$) con el número de profesores funcionarios del área, denotado por $F$. Calculándose $Q = \frac{F}{H} \times 100$, quedándonos con dos cifras decimales. Cuanto menor sea el valor de $Q$, mayor será la necesidad de profesorado funcionario en dicha área.

Por otro lado, la universidad también quiere tratar de manera igualitaria a cada uno de los ámbitos presentes en la misma: científico, tecnológico, jurídico-social y humanístico. Para ello se garantiza en su normativa una plaza por ámbito en el caso de que haya solicitudes.

El procedimiento para decidir en qué áreas se convocarán las plazas es el siguiente. Se elabora una lista ordenada de menor a mayor valor de $Q$. En el caso de existir más de una solicitud por área, se recalculará el valor de $Q$ como resultado de la primera asignación, es decir, incluyendo la plaza ya asignada. Una vez elaborada la lista se seleccionan las primeras áreas de la misma garantizando una plaza por ámbito, siempre que existan solicitudes. En el caso de empate en el indicador $Q$ tendrá preferencia el área con más carga docente.

Las plazas se asignarán al campus al que pertenezca la persona solicitante. En el caso de haber solicitantes de varios campus en una misma área, se aplicará el cálculo por separado a los campus, y resultará seleccionado aquel con menor valor de $Q$.

En el caso de haber empate entre campus, se utilizará el criterio de desempate por género, (calculando el cociente entre el número de TU del mismo género que la persona solicitante y el total de TU del área de conocimiento en el campus, y se seleccionará el de menor cociente). Si sigue el empate, se deshará aplicando sucesivamente los criterios de mayor antigüedad en la acreditación, mayor antigüedad como profesorado permanente en la UVigo y mayor edad.

A continuación estudiamos este problema mediante un nuevo modelo de problemas de bancarrota con bienes indivisibles en el que los agentes poseen una puntuación para optar a cada una de las unidades del estado y, además, éstos se agrupan en grupos que tienen un derecho mínimo que debe cumplirse. Vamos a considerar un caso simplificado, en el cual asumiremos que hay un único campus y simplificaremos el mecanismo de desempate.

## 3.1. Planteamiento y notación

Se quiere distribuir una cantidad fija de unidades indivisibles, $E \in \mathbb{Z}_+$, a la que llamaremos estado, entre un conjunto finito de agentes, $N$, cuyas demandas vienen representadas en el vector $c = (c_i)_{i \in N}$. Para cada $i \in N$, $c_i \in \mathbb{Z}_+$ denota la demanda del agente $i$-ésimo. Además, cada agente posee una puntuación. Estas puntuaciones vienen recogidas en el vector $p = \{p_i^k, i \in N, k \in \mathbb{N}\}$, donde $p_i^k \in \mathbb{R}$ denota la puntuación del agente $i \in N$ para optar a la $k-$ésima unidad del estado. Se asume que $p_i^k$ es creciente en $k$.

Los agentes se agrupan en grupos que tienen un derecho mínimo que debe ser cumplido, por lo que se tiene un vector de restricciones $S = (S_r, l_r)_{r=1}^m$, donde $S_r \subset N$ y $l_r \in \mathbb{Z}_+$ tal que $0 \leq l_r \leq E$. El vector de restricciones nos dice que cada grupo $S_r$ debe recibir como mínimo $l_r$ unidades del estado. Asumiremos:

- $(S_r)_{r=1}^m$ es una partición de $N$.
- $\sum_{r \in \{1,\ldots,m\}} l_r < E$. A cada grupo se le puede dar su derecho mínimo sin exceder el estado.
- $\sum_{i \in S_r} c_i > l_r$. La demanda de cada grupo es mayor que el derecho mínimo, por lo que son compatibles.

Entonces, el problema es una tupla $(N, c, E, p, S)$ tal que $E < \sum_{i \in N} c_i$.

La solución al problema viene dada por una regla de reparto. Una regla de reparto es una función $f$ que asigna a cada problema $(N, c, E, p, S)$ un vector $f(N, c, E, p, S) \in \mathbb{Z}^N$ que satisface las siguientes condiciones:

- $\sum_{i \in N} f_i(N, c, E, p, S) = E$. Se asigna el estado completamente.
- $0 \leq f_i(N, c, E, p, S) \leq c_i$ para todo $i \in N$. Todos los agentes reciben una cantidad no negativa pero no mayor que su demanda.
- $\sum_{i \in S_r} f_i(N, c, E, p, S) \geq l_r$ para todo $r = 1, ..., m$. La regla de reparto debe respetar las restricciones impuestas por $S$.

Trasladando este modelo al caso particular de reparto de plazas de titular entre las diferentes áreas de una universidad, nos encontramos con que:

- $N$ es el conunto de áreas de la universidad.
- Para cada área $i \in N$, $c_i$ denota el número de profesores acreditados para titular.
- $p$ recoge las puntaciones de cada área de acuerdo con el índice $Q$.
- $S = (S_r, l_r)_{r=1,\ldots,m}$ recoge las restricciones impuestas sobre cada ámbito en la universidad. En el caso particular de la Universidade de Vigo, hay cuatro ámbitos, $m = 4$, y a cada uno se le asegura una plaza, $l_r = 1$ para $r = 1, 2, 3, 4$.

## 3.2. Reglas de reparto

En esta sección se presentarán dos reglas de reparto para resolver este tipo de problemas en los que la prioridad viene dada por las puntuaciones. Ambas reglas dan prioridad a los agentes con puntuaciones menores.

La primera regla de reparto que se verá, que es la usada por la Universidad de Vigo, asigna el estado unidad a unidad siguiendo la prioridad dada por las puntuaciones, teniendo en cuenta que se cumplan los derechos mínimos de los grupos. En la segunda regla de reparto, el estado se asigna en dos etapas. En la primera etapa se asigna el derecho mínimo a cada uno de los grupos y después se asigna el estado restante entre los diferentes grupos teniendo en cuenta las puntuaciones agregadas de los agentes de cada grupo. En la segunda etapa se distribuye la asignación de cada grupo entre los agentes que lo forman siguiendo el orden de prioridad dado por las puntuaciones de cada agente.

A continuación se definen formalmente ambas reglas:

### 3.2.1. Regla aplicada por la Universidad de Vigo

Dado un problema $(N, c, E, p, S)$, esta regla de reparto asigna el estado unidad a unidad siguiendo la prioridad dada por las puntuaciones teniendo en cuenta los derechos mínimos de los grupos. Denotamos por $A_N = \{(i, k_i) : i \in N, k_i \leq c_i\}$ el conjunto de pares formados por los agentes y el número de unidades del estado que demandan. Definimos $\pi_{A_N}$ como un orden sobre los pares en el conjunto $A_N$ tal que $\pi_{A_N}(i,k) < \pi_{A_N}(j,l)$ si y sólo si $p_i^k < p_j^l$ o $p_i^k = p_j^l$ y $i < j$. Por lo tanto, en caso de empate, tendrá prioridad el agente con menor índice.

Para cada subconjunto, $S_r \subset N$ denotamos por $A_{S_r} = \{(i, k_i) : i \in S_r, k_i \leq c_i\}$ el conjunto de pares formados por los agentes en $S_r$ y el número de unidades que demandan. Definimos el orden $\pi_{A_{S_r}}$ asociado con $A_{S_r}$ de igual forma que antes.

Empezamos asignando a cada grupo, $S_r$, su derecho mínimo $l_r$, teniendo en cuenta el orden $\pi_{A_{S_r}}$. Entonces, cada agente $i \in S_r$ recibirá $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}|$ unidades. En otras palabras, $x_i^1$ es el número de pares agente-demanda que involucran al agente $i$ que, en el orden restringido a $S_r$, ocupan los $l_r$ primeros puestos.

A continuación asignamos el estado restante, $E - \sum_{r=1}^m l_r$, entre todos los agentes teniendo en cuenta el orden $\pi_{A'}$, donde $A' = \{(i, k_i) \in A_N : k_i > x_i^1\}$.

Entonces, para cada $i \in N$ tal que $i \in S_r$ la asignación final será:

$$f_i^1(N, c, E, p, S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^m l_r\}|.$$

***Ejemplo:*** Consideremos el problema $(N, c, E, p, S)$ donde:

- $N = \{1, 2, 3, 4, 5\}$. Departamentos en los que hay solicitantes.
- $c = (2, 1, 3, 3, 2)$. Número de solicitantes en cada departamento.
- $E = 5$. Plazas ofertadas.

- Las puntuaciones $p = \{p_i^k, i \in N, k \in \mathbb{N}\}$ vienen dadas en la siguiente tabla.

| Ámbito 1 | | Ámbito 2 | | |
|---|---|---|---|---|
| Área 1 | Área 2 | Área 3 | Área 4 | Área 5 |
| 1.3 | 0.7 | 0.4 | 1 | 0.8 |
| 3.1 | 2 | 1.2 | 2.7 | 1.5 |
| 6 | 2.9 | 2.5 | 4.5 | 2.8 |
| 9.4 | 4.1 | 3.6 | 5.7 | 4.3 |
| 10 | 4.7 | 4 | 6.1 | 5 |

  El ámbito 1 está formada por 2 áreas y el ámbito 2 por 3 áreas. En las columnas tenemos las puntuaciones de cada área para conseguir de la primera a la quinta plaza ofertada.

- $S = (S_r, l_r)_{r=1}^2$ donde $l_1 = l_2 = 1$. Esto es, cada ámbito debe recibir al menos una unidad del estado.

  El orden $\pi_{A_N}$ viene dado por

$$(3,1),(2,1),(5,1),(4,1),(3,2),(1,1),(5,2),(3,3),(4,2),(1,2),(4,3)$$

  Y los órdenes $\pi_{A_{S_1}}$ y $\pi_{A_{S_2}}$ son respectivamente:

$$(2,1),(1,1),(1,2)$$

$$(3,1),(5,1),(4,1),(3,2),(5,2),(3,3),(4,2),(4,3)$$

Primero asignamos las $l_r$ plazas de cada ámbito $S_r$ de acuerdo a los órdenes $\pi_{A_{S_1}}$ y $\pi_{A_{S_2}}$. Así, $x^1 = (0,1,1,0,0)$.

A continuación, asignamos las tres plazas restantes siguiendo el orden dado por $\pi_{A'}$, donde $A' = A_N \backslash \{(2,1),(3,1)\}$,

$$(5,1),(4,1),(3,2),(1,1),(5,2),(3,3),(4,2),(1,2),(4,3).$$

Así, $f^1(N,c,E,p,S) = (0,1,1,0,0) + (0,0,1,1,1) = (0,1,2,1,1)$.

Por lo tanto, el área 2 se lleva una plaza, el área 3 se lleva dos plazas, el área 4 una y el área 5 se lleva otra.

La Universidade de Vigo, para cada área $i \in N$, calcula las puntuaciones $p_i^k$ teniendo en cuenta el número de profesores/as funcionarios y las horas de docencia en el área. Así, esta puntuación mide las necesidades docentes de cada área. Como se puede ver en la tabla de puntuaciones del ejemplo, el área 3 necesita realmente más profesores que otras áreas, y entonces la regla aplicada por la universidad favorece a este área sobre las demás.

### 3.2.2. Regla propuesta

Supongamos ahora que, en lugar de fijarnos en las áreas, nos fijamos en los ámbitos y consideramos la suma de las puntuaciones de sus áreas. Se observa que ambos ámbitos tienen necesidades similares, pero el ámbito 1 está ligeramente más necesitado de profesorado funcionario que el ámbito 2.

| Ámbito 1 | Ámbito 2 |
|---|---|
| 2 | 2.2 |
| 5.1 | 5.4 |
| 8.9 | 9.8 |
| 13.5 | 13.6 |
| 14.7 | 15.1 |

Esto permite introducir una forma diferente de asignar las plazas. Esta nueva regla de reparto asigna las plazas en dos etapas.

En la primera etapa se reparten las 5 plazas entre los ámbitos teniendo en cuenta las puntuaciones agregadas y el derecho mínimo de cada ámbito. En caso de que las puntuaciones de dos ámbitos coincidan se considera la siguiente regla de desempate: se asocia a cada ámbito un vector $y^r \in \{0,1\}^N$ tal que $y_i^r = 1$ si $i \in S_r$ y $y_i^r = 0$ en otro caso; tendrá prioridad el ámbito que esté primero de acuerdo al orden lexicográfico aplicado a los vectores $y^r$.

En la segunda etapa, se distribuyen las plazas asignadas a cada ámbito entre las áreas que lo forman siguiendo el orden dado por las puntuaciones, de la misma forma que en la regla $f^1$.

Para la primera etapa debemos considerar un problema en el que los agentes son los grupos. Dado un problema $(N, c, E, p, S)$ definimos el problema entre grupos $(\tilde{N}, \tilde{c}, \tilde{E}, \tilde{p}, \tilde{S})$, donde:

- $\tilde{N} = \{1, ..., m\}$. Los agentes son los grupos.
- $\tilde{c} = (c_r)_{r\in\{1,\ldots,m\}}$ donde $c_r = \sum_{i\in S_r} c_i$. Cada grupo reclama la suma de las demandas de los agentes pertenecientes al grupo.
- $\tilde{E} = E$.
- $\tilde{p} = \{p_r^k = \sum_{i\in S_r} p_i^k$ para cada $r \in \tilde{N}$ y $k \in \mathbb{N}\}$. La puntuación de cada grupo $r$ para obtener la $k-$ésima unidad del estado es la suma de las puntuaciones de los agentes pertenecientes a ese grupo para obtener tal unidad.
- $\tilde{S} = (\{r\}, l_r)_{r\in\{1,\ldots,m\}}$.

Considérese el conjunto $A_{\tilde{N}} = \{(r, k_r) : r \in \tilde{N}$ y $k_r \leq c_r\}$ y su orden asociado $\pi_{A_{\tilde{N}}}$ donde $\pi_{A_{\tilde{N}}}(r,k) < \pi_{A_{\tilde{N}}}(s,l)$ si $p_r^k < p_s^l$ o $p_r^k = p_s^l$ e $y_r \leq_L y_s$. Asignamos el estado entre los grupos siguiendo el orden dado por $\pi_{A_{\tilde{N}}}$ de la siguiente forma: para cada $r \in \{1, ..., m\}$,

$$x_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : \pi_{A'_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^{m} l_r\}|,$$

donde $A'_{\tilde{N}} = \{(r, k_r) \in A_{\tilde{N}} : k_r > l_r\}$.

Una vez que tenemos asignado el estado entre los grupos, pasamos a la segunda etapa. Para cada grupo, se distribuye $x_r$ entre sus miembros. Entonces, para cada grupo $r \in \tilde{N}$, definimos el problema $(S_r, c_{|S_r}, x_r, p^r, S_{|S_r})$ donde $p^r = \{p_i^k : i \in S_r, k \in \mathbb{N}\}$ y $S_{|S_r} = (S_r, l_r)$. Se considera el orden $\pi_{A_{S_r}}$, definido como en la regla anterior, y se asigna el estado unidad a unidad siguiendo dicho orden. Así, la asignación final será, para cada $r \in \tilde{N}$ y cada $i \in S_r$.

$$f_i^2(N, c, E, p, S) = |\{(i, k_i) \in A_{S_r} : \pi_{A_{S_r}}(i, k_i) \leq x_r\}|.$$

Si aplicamos esta regla al ejemplo anterior, la nueva asignación de plazas será: $x = (2, 1, 1, 0, 1)$. En la asignación anterior, inicialmente el ámbito 1 se llevaba 1 plaza y el ámbito 2 las 4 restantes. En este caso se observa un reparto más igualitario entre los ámbitos, llevándose 3 plazas el ámbito 1 y las dos restantes el ámbito 2.

## 3.3. Propiedades

En esta sección se introducen propiedades para este nuevo modelo y se estudia cuáles de ellas son satisfechas por las reglas definidas en la sección anterior. La mayoría de estas propiedades son adaptaciones de propiedades conocidas de la literatura de bancarrota a este nuevo marco.

**Equilibrio:** Esta propiedad, como ya vimos en el capítulo anterior, es una adaptación de la propiedad de tratamiento igualitario. La propiedad de tratamiento igualitario dice que dos agentes con la misma demanda deben recibir la misma asignación. Pero en el caso de problemas con bienes indivisibles esto no siempre se cumple. Por lo que se modifica la propiedad para permitir una diferencia de una unidad en las asignaciones de los agentes con la misma demanda y puntuación.

Para cada $(N, c, E, p, S)$ y cada $i, j \in S_r$, si $c_i = c_j$ y $p_i^k = p_j^k$ para todo $k \in \mathbb{N}$, entonces

$$|f_i(N, c, E, p, S) - f_j(N, c, E, p, S)| \leq 1.$$

**Conservación del orden:** La propiedad de conservación del orden en modelos clásicos de bancarrota establece que una regla debe preservar el orden de las demandas, en el sentido de que si un agente $i$ tiene una demanda mayor que otro agente $j$, entonces la regla no debe asignar más al agente $j$ que al agente $i$. Además, este orden también debe respetarse al calcular las pérdidas sufridas por esos agentes. Adaptamos esta propiedad a nuestro marco imponiendo que el orden que debe respetarse es el impuesto por las puntuaciones $p$.

Así que, dado un problema de bancarrota $(N, c, E, p, S)$, para todo $i, j \in N$ tal que $f_i(N, c, E, p, S) < c_i$,

$$p_i^{f_i(N,c,E,p,S)+1} \geq p_j^{f_j(N,c,E,p,S)}.$$

Es trivial ver que podría darse el caso de que ninguna regla satisfaga esta propiedad debido a las restricciones $S$ del modelo.

Supongamos que tenemos que tenemos el problema de bancarrota $(N, c, E, p, S)$ donde $N = \{1, 2, 3\}$, $c = (3, 2, 3)$, $E = 5$ y $S = (S_r, l_r)_{r=1}^2$ con $l_r = 2$. Supongamos que en el primer grupo está formado por el agente 1 y el segundo grupo está formado por los agentes 2 y 3.

Ya que el derecho mínimo del primer grupo es de 2 unidades, el agente 1 recibirá 2 unidades, aunque su puntuación sea mayor que la puntuación de los demás agentes.

Así que restringimos esta propiedad a los agentes pertenecientes al mismo grupo.

**Conservación del orden dentro de grupos:** Sea $i, j \in S_r$ tal que $f_i(N, c, E, p, S) < c_i$. Entonces,

$$p_i^{f_i(N,c,E,p,S)+1} \geq p_j^{f_j(N,c,E,p,S)}.$$

**Composición hacia arriba:** Supongamos que después de haber asignado el estado, se determina que su valor es mayor, $E' > E$. La propiedad de composición hacia arriba determina que cancelar la asignación inicial y resolver el nuevo problema es lo mismo que mantener la asignación inicial, ajustar las demandas teniendo en cuenta esa asignación y aplicar la regla para dividir el excedente, $E' - E$.

Formalmente, si $E' > E > \sum_{r=1}^m l_r$, entonces

$$f(N, c, E', p, S) = f(N, c, E, p, S) + f(N, c - f(N, c, E, p, S), E' - E, p', S'),$$

donde

- $S'_r = S_r$, $l'_r = 0$ para cada $r \in \{1, ..., m\}$.
- $p_i'^k = p_i^{k+f_i(N,c,E,p,S)}$ para cada $i \in N$ y cada $k \in \mathbb{N}$.

**Composición hacia abajo:** Consideremos ahora el caso contrario, el estado es menor de lo esperado, $E' < E$. Tenemos dos opciones: cancelar la asignación inicial y aplicar la regla al nuevo problema; o considerar las asignaciones iniciales como demandas sobre $E'$ y aplicar la regla al problema definido de esta forma. La propiedad de composición hacia abajo nos dice que el resultado es el mismo independientemente de la opción elegida.

Formalmente, si $\sum_{i \in N} c_i > E > E' > \sum_{r=1}^{m} l_r$, entonces

$$f(N, c, E', p, S) = f(N, f(N, c, E, p, S), E', p, S).$$

**Monotonía respecto a las demandas:** Esta propiedad nos dice que si la demanda de un agente aumenta, la asignación de dicho agente debe ser mayor o igual a la asignación inicial.

Dados dos problemas $(N, c, E, p, S)$ y $(N, c', E, p, S)$ tal que existe $i \in N$ con $c'_i > c_i$ y $c'_j = c_j$ para todo $j \in N \setminus \{i\}$, entonces $f_i(N, c', E, p, S) \geq f_i(N, c, E, p, S)$.

**Monotonía respecto a las puntuaciones.** Si la puntuaciones de un agente disminuyen, la asignación de dicho agente debe ser mayor o igual que la asignación inicial.

Para cada $(N, c, E, p, S)$ y para cada $i \in N$ si $p_i^k > p_i'^k$ y para cada $k \in \{1, ..., E\}$ entonces,

$$f_i(N, c, E, p, S) \leq f_i(N, c, E, p', S).$$

**Monotonía respecto al estado.** Si el estado aumenta, cada agente debe recibir al menos tanto como inicialmente.

Para cada $(N, c, E, p, S)$ y $(N, c, E', p, S)$ si $\sum_{i \in N} c_i > E' \geq E$ entonces,

$$f(N, c, E', p, S) \geq f(N, c, E, p, S).$$

**Teorema 3.1.** *Las dos reglas presentadas en esta sección verifican las propiedades de: equilibrio, conservación del orden dentro de los grupos, composición hacia abajo, monotonía respecto a las demandas, monotonía respecto a las puntuaciones y monotonía respecto al estado.*

*Demostración.*

- **Equilibrio**. Para cada $(N, c, E, p, S)$ y cada $i, j \in S_r$, si $c_i = c_j$ y $p_i^k = p_j^k$ para todo $k \in \mathbb{N}$, entonces $|f_i(N, c, E, p, S) - f_j(N, c, E, p, S)| \leq 1$.

  **Regla aplicada por la universidad.**

  Para cada $i \in S_r$

  $$f_i^1(N, c, E, p, S) = x_i^1 + |\{(i, k) \in A' : \pi_{A'}(i, k) \leq E - \sum_{r=1}^{m} l_r\}|,$$

  donde $x_i^1 = |\{(i, k) \in A_N : \pi_{A_{S_r}}(i, k) \leq l_r\}|$ y $A' = \{(i, k_i) \in A_N : k_i > x_i^1\}$.

  Empezaremos viendo que sucede con $x_i^1$ y $x_j^1$. Supongamos que $|x_i^1 - x_j^1| > 1$. Vamos a considerar que $i < j$, por lo que $x_i^1 > x_j^1 + 1$.

  Teniendo en cuenta que $p_i^k = p_j^k$ y $p_i^k < p_i^{k+1}$ para todo $k \in \mathbb{N}$, se tiene

  $$p_j^{x_j^1+1} = p_i^{x_j+1} < p_i^{x_i^1}.$$

  Teniendo en cuenta como se define el orden $\pi$ sobre $A_{S_r}$, se tiene

  $$\pi_{A_{S_r}}(j, x_j^1 + 1) < \pi_{A_{S_r}}(i, x_i^1).$$

Entonces, el agente $j$ recibiendo $x_j^1+1$ unidades tiene preferencia sobre el agente $i$ recibiendo $x_i^1$ unidades. Se llega a una contradicción, por lo que $|x_i^1 - x_j^1| \leq 1$.

Denotaremos por $x_i^2 = |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^m l_r\}|$.

Veamos ahora que sucede con $x_i^1 + x_i^2$ y $x_j^1 + x_j^2$. Tenemos dos posibles situaciones:

- $x_i^1 = x_j^1$. Razonaremos de la misma manera que en el caso anterior.
  Supongamos que $|x_i^1 + x_i^2 - x_j^1 - x_j^2| > 1$. Como consideramos $i < j$, $x_i^1 + x_i^2 > x_j^1 + x_j^2 + 1$. Teniendo en cuenta que $p_i^k = p_j^k$ y $p_i^k < p_i^{k+1}$ para todo $k \in \mathbb{N}$, se tiene

  $$p_j^{x_j^1+x_j^2+1} = p_i^{x_j^1+x_j^2+1} < p_i^{x_i^1+x_i^2}.$$

  Teniendo en cuenta como se define el orden $\pi$ sobre $A'$, se tiene

  $$\pi_{A'}(j, x_j^1 + x_j^2 + 1) < \pi_{A'}(i, x_i^1 + x_i^2).$$

  Entonces, el agente $j$ recibiendo $x_j^1 + x_j^2 + 1$ unidades tiene preferencia sobre el agente $i$ recibiendo $x_i^1 + x_i^2$ unidades. Se llega a una contradicción, por lo que

  $$|x_i^1 + x_i^2 - x_j^1 - x_j^2| = |f_i^1(N,c,E,p,S) - f_j^1(N,c,E,p,S)| \leq 1.$$

- $x_i^1 = x_j^1 + 1$. Hay que comprobar que se cumple $|f_i^1(N,c,E,p,S) - f_j^1(N,c,E,p,S)| \leq 1$. Es decir, que $|x_i^1 + x_i^2 - x_j^1 - x_j^2| = |x_j^1 + 1 + x_i^2 - x_j^1 - x_j^2| = |x_i^2 + 1 - x_j^2| \leq 1$. Esto quiere decir que el agente $j$ recibe en esta segunda etapa, lo mismo que el agente $i$ o una unidad más.
  Consideremos

  $$A_i = \{(i,k) : k = x_i^1 + 1, ..., c_i\},$$
  $$A_j = \{(j,k) : k = x_j^1 + 1, ..., c_j\} = \{(j,k) : k = x_i^1, ..., c_i\}.$$

  Teniendo en cuenta que $x_i^1 = x_j^1 + 1$ y que $c_i = c_j$, se tiene que $A_i$ contiene un par menos que $A_j$.
  Empecemos viendo que en la segunda etapa el agente $j$ no puede recibir menos que el agente $i$. Supongamos que recibe menos, es decir, $x_i^2 > x_j^2$, que es equivalente a $x_i^1 + x_i^2 > x_j^1 + 1 + x_j^2$. Teniendo en cuenta que $p_i^k = p_j^k$ y que $p_i^k < p_i^{k+1}$, se tiene que $p_i^{x_i^1+x_i^2} > p_j^{x_j^1+1+x_j^2}$. Entonces,

  $$\pi_{A_j}(j, x_j^1 + 1 + x_j^2) < \pi_{A_i}(i, x_i^1 + x_i^2) \leq E - \sum_{r=1}^m l_r.$$

  Entonces, en la segunda etapa, el agente $j$ debe recibir, al menos, lo mismo que el agente $i$, es decir $x_j^2 \geq x_i^2$.
  Veamos ahora que el agente $j$ no puede recibir más de una unidad más que el agente $i$.
  Si el agente $j$ recibe en la segunda etapa $x_j^2$ unidades, se tiene que

  $$\pi_{A_j}(j, x_j^1 + 1 + x_j^2) \leq E - \sum_{r=1}^m l_r.$$

  Además, $p_j^{x_j^1+1+x_j^2} = p_i^{x_j^1+1+x_j^2} > p_i^{x_j^1+1+x_j^2-1} = p_i^{x_i^1+x_j^2-1}$. Entonces,

  $$\pi_{A_i}(i, x_i^1 + x_j^2 - 1) < \pi_{A_j}(j, x_j^1 + 1 + x_j^2) \leq E - \sum_{r=1}^m l_r.$$

Por lo que si el agente $j$ recibe $x_j^2$ unidades en la segunda etapa, el agente $i$ recibe, como mínimo, una unidad menos que $j$, es decir, $x_i^2 \geq x_j^2 - 1$.

Se tiene entonces que $x_j^2 \geq x_i^2$ y $x_i^2 \geq x_j^2 - 1$, por lo tanto, $x_i^2 \leq x_j^2 \leq x_i^2 + 1$, es decir, $x_i^2 = x_j^2$ o $x_i^2 + 1 = x_j^2$.

Conclusión,

$$|x_i^1 + x_i^2 - x_j^1 - x_j^2| = |f_i^1(N, c, E, p, S) - f_j^1(N, c, E, p, S)| \leq 1.$$

**Regla propuesta.**

Para cada $r \in \tilde{N}$ y cada $i \in S_r$,

$$f_i^2(N, c, E, p, S) = |\{(i, k_i) \in A_{S_r} : \pi_{A_{S_r}}(i, k_i) \leq x_r\}|,$$

donde $x_r = l_r + |\{(r, k) \in A'_{\tilde{N}} : \pi_{A'_{\tilde{N}}}(r, k) \leq E - \sum_{r=1}^m l_r\}|$ siendo $A'_{\tilde{N}} = \{(r, k_r) \in A_{\tilde{N}} : k_r > l_r\}$.

En la primera etapa, $x_r$ es lo que se lleva cada grupo, por lo que tenemos que ver que sucede al repartir la asignación del grupo al que pertenecen los agentes $i$ y $j$ entre sus componentes.

Usando un razonamiento análogo al ussado en la regla anterior para demostrar que $|x_i^1 - x_j^1| \leq 1$, se demuestra que

$$|f_i^2(N, c, E, p, S) - f_j^2(N, c, E, p, S)| \leq 1.$$

- **Conservación del orden dentro de los grupos.** Sea $i, j \in S_r$ tal que $f_i(N, c, E, p, S) < c_i$. Entonces, $p_i^{f_i(N,c,E,p,S)+1} \geq p_j^{f_j(N,c,E,p,S)}$.

  **Regla aplicada por la universidad.**

  Para cada $i \in S_r$

  $$f_i^1(N, c, E, p, S) = x_i^1 + |\{(i, k) \in A' : \pi_{A'}(i, k) \leq E - \sum_{r=1}^{m} l_r\}|,$$

  donde $x_i^1 = |\{(i, k) \in A_N : \pi_{A_{S_r}}(i, k) \leq l_r\}|$ y $A' = \{(i, k_i) \in A_N : k_i > x_i^1\}$.

  Vamos a distinguir varios casos:

  - $f_i^1(N, c, E, p, S) = x_i^1$ y $f_j^1(N, c, E, p, S) = x_j^1$.

    Supongamos que $p_i^{f_i^1(N,c,E,p,S)+1} < p_j^{f_j^1(N,c,E,p,S)}$, es decir, $p_i^{x_i^1+1} < p_j^{x_j^1}$.

    Entonces, se tiene que $\pi_{A_{S_r}}(i, x_i^1 + 1) < \pi_{A_{S_r}}(j, x_j^1)$.

    Esto quiere decir que si el agente $j$ recibe $x_j^1$ unidades ($\pi_{A_{S_r}}(j, x_j^1) \leq l_r$), el agente $i$ tiene que recibir $x_i^1 + 1$ ya que en el orden definido sobre los pares en el conjunto $A_{S_r}$, el agente $i$ recibiendo $x_i^1 + 1$ unidades tiene un valor menor ($\pi_{A_{S_r}}(i, x_i^1 + 1) < \pi_{A_{S_r}}(j, x_j^1) \leq l_r$).

    Se llega a una contradicción, por lo que

    $$p_i^{x_i^1+1} \geq p_j^{x_j^1},$$

    es decir,

    $$p_i^{f_i^1(N,c,E,p,S)+1} \geq p_j^{f_j^1(N,c,E,p,S)}.$$

- $f_i^1(N,c,E,p,S) = x_i^1 + x_i^2$ y $f_j^1(N,c,E,p,S) = x_j^1$.

  Supongamos que $p_i^{f_i^1(N,c,E,p,S)+1} < p_j^{f_j^1(N,c,E,p,S)}$, es decir, $p_i^{x_i^1+x_i^2+1} < p_j^{x_j^1}$. Por lo tanto $p_i^{x_i^1+1} < p_j^{x_j^1}$.

  Entonces, se tiene que $\pi_{A_{S_r}}(i, x_i^1+1) < \pi_{A_{S_r}}(j, x_j^1) \leq l_r$.

  Esto quiere decir que el agente $i$ debería recibir en la primera etapa del problema $x_i^1+1$ unidades, que contradice su asignación inicial.

  Por lo tanto,
  $$p_i^{x_i^1+x_i^2+1} \geq p_j^{x_j^1},$$
  es decir,
  $$p_i^{f_i^1(N,c,E,p,S)+1} \geq p_j^{f_j^1(N,c,E,p,S)}.$$

- $f_i^1(N,c,E,p,S) = x_i^1 + x_i^2$ y $f_j^1(N,c,E,p,S) = x_j^1 + x_j^2$.

  Supongamos que $p_i^{f_i^1(N,c,E,p,S)+1} < p_j^{f_j^1(N,c,E,p,S)}$, es decir, $p_i^{x_i^1+x_i^2+1} < p_j^{x_j^1+x_j^2}$.

  Entonces, se tiene que $\pi_{A'}(i, x_i^1+x_i^2+1) < \pi_{A'}(j, x_j^1+x_j^2)$.

  Esto quiere decir que si el agente $j$ recibe $x_j^1+x_j^2$ unidades ($\pi_{A'}(j, x_j^1+x_j^2) \leq E - \sum_{r=1}^m l_r$), el agente $i$ tiene que recibir $x_i^1+x_i^2+1$ ya que en el orden definido sobre los pares en el conjunto $A'$, el agente $i$ recibiendo $x_i^1+x_i^2+1$ unidades tiene un valor menor ($\pi_{A'}(i, x_i^1+x_i^2+1) < \pi_{A'}(j, x_j^1+x_j^2) \leq E - \sum_{r=1}^m l_r$).

  Se llega a una contradicción, por lo que
  $$p_i^{x_i^1+x_i^2+1} \geq p_j^{x_j^1+x_j^2},$$
  es decir,
  $$p_i^{f_i^1(N,c,E,p,S)+1} \geq p_j^{f_j^1(N,c,E,p,S)}.$$

- $f_i^1(N,c,E,p,S) = x_i^1$ y $f_j^1(N,c,E,p,S) = x_j^1 + x_j^2$.

  Supongamos que $p_i^{f_i^1(N,c,E,p,S)+1} < p_j^{f_j^1(N,c,E,p,S)}$, es decir, $p_i^{x_i^1+1} < p_j^{x_j^1+x_j^2}$.

  Entonces, se tiene que $\pi_{A'}(i, x_i^1+1) < \pi_{A'}(j, x_j^1+x_j^2)$.

  Esto quiere decir que si el agente $j$ recibe $x_j^1+x_j^2$ unidades, el agente $i$ tiene que recibir $x_i^1+1$, lo que contradice su asignación inicial. Por lo que
  $$p_i^{x_i^1+1} \geq p_j^{x_j^1+x_j^2},$$
  es decir,
  $$p_i^{f_i^1(N,c,E,p,S)+1} \geq p_j^{f_j^1(N,c,E,p,S)}.$$

**Regla propuesta.** Para cada $r \in \tilde{N}$ y cada $i \in S_r$,
$$f_i^2(N,c,E,p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \leq x_r\}|,$$
donde $x_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : k > l_r, \pi_{A'_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^m l_r\}|$ siendo $A'_{\tilde{N}} = \{(r,k_r) \in A_{\tilde{N}} : k_r > l_r\}$.

Supongamos que $p_i^{f_i^2(N,c,E,p,S)+1} < p_j^{f_j^2(N,c,E,p,S)}$.

Teniendo en cuenta como se define el orden $\pi_{A_{S_r}}$, se tiene
$$\pi_{A_{S_r}}(i, f_i^2(N,c,E,p,S)+1) < \pi_{A_{S_r}}(j, f_j^2(N,c,E,p,S)).$$

Por lo que si el agente $j$ recibe $f_j^2(N,c,E,p,S)$ unidades, el agente $i$ debe recibir $f_i^2(N,c,E,p,S)+1$ unidades. Se llega así a una contradicción. Entonces

$$p_i^{f_i(N,c,E,p,S)+1} \geq p_j^{f_j(N,c,E,p,S)}.$$

- **Composición hacia abajo.** Si $\sum_{i\in N} c_i > E > E' > \sum_{r=1}^m l_r$, entonces

$$f(N,c,E',p,S) = f(N, f(N,c,E,p,S), E', p, S).$$

**Regla aplicada por la universidad.**

Para cada $i \in S_r$

$$f_i^1(N,c,E,p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^m l_r\}|,$$

donde $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}|$.

Además, dado que $E' > \sum_{r=1}^m l_r$, se puede ver fácilmente que

$$f_i^1(N,c,E',p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E' - \sum_{r=1}^m l_r\}|,$$

donde $A_N = \{(i,k_i) : i \in N, k_i \leq c_i\}$ y $A' = \{(i,k_i) \in A_N : k_i > x_i^1\}$.

Por otro lado

$$f_i^1(N, f^1(N,c,E,p,S), E', p, S) = x_i'^1 + |\{(i,k) \in A'' : \pi_{A''}(i,k) \leq E' - \sum_{r=1}^m l_r\}|,$$

donde $x_i'^1 = |\{(i,k) \in A_N^* : \pi_{A_{S_r}^*}(i,k) \leq l_r\}|$, $A_N^* = \{(i,k_i) : i \in N, k_i \leq f_i^1(N,c,E,p,S)\}$, $A_{S_r}^* = \{(i,k_i) : i \in S_r, k_i \leq f_i^1(N,c,E,p,S)\}$ y $A'' = \{(i,k_i) \in A_N^* : k_i > x_i'^1\}$.

Es trivial ver que $x_i^1 = x_i'^1$. Basta notar que $A_N^* \subset A_N$ y además los pares en $A_N^*$ preceden en el orden $\pi$ a aquellos en $A_N \setminus A_N^*$. Por otro lado, dado que $\sum_{i\in N} f_i^1(N,c,E,p,S) = E > E'$, tenemos pares suficientes para repartir los derechos mínimos que son equivalentes en ambos problemas.

Por lo tanto $A'' = \{(i,k) \in A_N^* : k > x_i'^1\} \subset \{(i,k) \in A_N : k > x_i^1\} = A'$, además los pares en $A''$ preceden en el orden $\pi$ a aquellos en $A' \setminus A''$. Por otro lado, dado que $\sum_{i\in N} f_i^1(N,c,E,p,S) = E > E'$, tenemos pares suficientes para repartir $E - \sum_{r=1}^m$ unidades y, por lo tanto, $E' - \sum_{r=1}^m$. De ese modo $f_i^1(N, f(N,c,E,p,S), E', p, S) = f_i^1(N,c,E',p,S)$.

**Regla propuesta:** Para cada $r \in \tilde{N}$ y cada $i \in S_r$,

$$f_i^2(N,c,E',p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \leq x_r\}|,$$

donde $x_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : \pi_{A'_{\tilde{N}}}(r,k) \leq E' - \sum_{r=1}^m l_r\}|$, $A'_{\tilde{N}} = \{(r,k_r) : r \in \tilde{N}, k_r > l_r\}$ y $A_{S_r} = \{(i,k_i) : i \in S_r, k_i \leq c_i\}$.

Por otro lado

$$f_i^2(N, f(N,c,E,p,S), E', p, S) = |\{(i,k_i) \in A'_{S_r} : \pi_{A'_{S_r}}(i,k_i) \leq x'_r\}|,$$

donde $x'_r = l_r + |\{(r,k) \in A''_{\tilde{N}} : k > l_r, \pi_{A''_{\tilde{N}}}(r,k) \leq E' - \sum_{r=1}^m l_r\}|$, $A''_{\tilde{N}} = \{(r,k_r) : r \in \tilde{N}, l_r < k_r \leq f_i^2(N,c,E,p,S)\}$ y $A'_{S_r} = \{(i,k_i) : i \in S_r, k_i \leq f_i^2(N,c,E,p,S)\}$.

Empecemos viendo que $x_r = x'_r$.

Observamos que $A''_{\tilde{N}} \subset A'_{\tilde{N}}$ además todas aquellos pares que no están en $A''_{\tilde{N}}$ ocupan posiciones posteriores en el orden inducido por las puntuaciones. Por otro lado, por definición de regla de reparto sabemos que $\sum_{i\in N} f_i^2(N,c,E,p,S) = E > E'$ así que dichos pares son suficientes para poder repartir entre los estados las $E'$ unidades disponibles. Por ello $x_r = x'_r$.

Siguiendo un razonamiento similar comparando los conjuntos $A'_{S_r}$ y $A_{S_r}$ podemos concluir que

$$f_i^2(N, f(N,c,E,p,S), E', p, S) = f_i^2(N,c,E',p,S).$$

- **Monotonía respecto a las demandas.** Dados dos problemas $(N,c,E,p,S)$ y $(N,c',E,p,S)$ tal que existe $i \in N$ con $c'_i > c_i$ y $c'_j = c_j$ para todo $j \in N \setminus \{i\}$, entonces $f_i(N,c',E,p,S) \geq f_i(N,c,E,p,S)$.

  **Regla aplicada por la universidad.**

  Para cada $i \in S_r$

$$f_i^1(N,c,E,p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^{m} l_r\}|,$$

  donde $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}|$ y $A' = \{(i,k) \in A_N : k > x_i^1\}$.

  Cuando aumenta la demanda del agente $i$ el nuevo reparto será,

$$f_i^1(N,c',E,p,S) = x_i'^1 + |\{(i,k) \in A'' : \pi_{A''}(i,k) \leq E - \sum_{r=1}^{m} l_r\}|$$

  donde $x_i'^1 = |\{(i,k) \in A'_N : \pi_{A'_{S_r}}(i,k) \leq l_r\}|$, siendo $A'_N = \{(i,k_i) : i \in N, k_i \leq c'_i\}$, $A'_{S_r} = \{(i,k_i) \in A'_N : i \in S_r\}$ y $A'' = \{(i,k_i) \in A'_N : k_i > x_i'^1\}$.

  Veamos primero que sucede con $x_i^1$ y $x_i'^1$. Dado que $c'_i > c_i$, se verifica que $A_{S_r} \subset A'_{S_r}$. Entonces, puede existir un par $(i,k) \in A'_{S_r} \backslash A_{S_r}$ de forma que $\pi_{A'_{S_r}}(i,k) \leq l_r$. Lo que implica que $x_i'^1 \geq x_i^1$.

  Veamos ahora que sucede con $x_i^2$ y $x_i'^2$.

  Supongamos que $x_i'^1 = x_i^1 + n$, con $n \in \{0, ..., c'_i - c_i\}$. Se tiene que

$$\begin{aligned} A' &= \{(i,k_i) \in A_N : x_i^1 < k \leq c_i\}, \\ A'' &= \{(i,k_i) \in A'_N : x_i'^1 < k \leq c'_i\} = \{(i,k_i) \in A'_N : x_i^1 + n < k \leq c'_i\}. \end{aligned}$$

  Por lo que $A'$ y $A''$ se van a diferenciar en $c'_i - c_i - n$ pares. $A''$ tendrá $c'_i - c_i - n$ pares más.

  Entonces, puede existir un $(i,k) \in A'' \backslash A'$ de forma que $\pi_{A''}(i,k) \leq E - \sum_{r=1}^{m} l_r$. Por lo que $x_i'^2 \geq x_i^2$.

  Teniendo en cuenta que $x_i'^1 \geq x_i^1$ y que $x_i'^2 \geq x_i^2$ se llega a que

$$f_i^1(N,c',E,p,S) = x_i'^1 + x_i'^2 \geq x_i^1 + x_i^2 = f_i^1(N,c,E,p,S).$$

  **Regla propuesta.**

  Para cada $r \in \tilde{N}$ y cada $i \in S_r$,

$$f_i^2(N,c,E,p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \leq x_r\}|,$$

  donde $x_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : \pi_{A'_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^{m} l_r\}|$.

Además,

$$f_i^2(N,c',E,p,S) = |\{(i,k_i) \in A'_{S_r} : \pi_{A'_{S_r}}(i,k_i) \leq x'_r\}|,$$

donde $x'_r = l_r + |\{(r,k) \in A''_{\tilde{N}} : \pi_{A''_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^m l_r\}|$.

Al aumentar la demanda del agente $i$ también aumenta la demanda agregada del grupo al que pertenece dicho agente. Esto es $c'_r > c_r$, por lo tanto,

$$A'_{\tilde{N}} = \{(r,k_r) : r \in \tilde{N}, l_r < k_r \leq c_r\},$$

$$A''_{\tilde{N}} = \{(r,k_r) : r \in \tilde{N}, l_r < k_r \leq c'_r\}.$$

Como $c'_r > c_r$, $A'_{\tilde{N}} \subset A''_{\tilde{N}}$. Por lo que puede existir un par $(r,k_r) \in A''_{\tilde{N}} \setminus A'_{\tilde{N}}$ tal que $\pi_{A''_{\tilde{N}}}(r,k_r) \leq E - \sum_{r=1}^m l_r$. Entonces $x'_r \geq x_r$.

Teniendo esto en cuenta, y que, además $A_{S_r} \subset A'_{S_r}$, ya que

$$A_{S_r} = \{(i,k_i) : i \in S_r, k_i \leq c_i\},$$

$$A'_{S_r} = \{(i,k_i) : i \in S_r, k_i \leq c'_i\}.$$

se llega a que puede existir un $(i,k) \in A'_{S_r} \setminus A_{S_r}$ tal que $\pi_{A'_{S_r}}(i,k) \leq x'_r$.

Entonces

$$f_i^2(N,c',E,p,S) \geq f_i^2(N,c,E,p,S).$$

- **Monotonía respecto a las puntuaciones.** Para cada $(N,c,E,p,S)$ y para cada $i \in N$ si $p_i^k > p_i'^k$ y para cada $k \in \{1,...,E\}$ entonces,

  $$f_i(N,c,E,p,S) \leq f_i(N,c,E,p',S).$$

  **Regla aplicada por la universidad.**

  Para cada $i \in S_r$

  $$f_i^1(N,c,E,p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^m l_r\}|,$$

  donde $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}|$.

  Se tiene también

  $$f_i^1(N,c,E,p',S) = x_i'^1 + |\{(i,k) \in A'' : \pi'_{A''}(i,k) \leq E - \sum_{r=1}^m l_r\}|,$$

  donde $x_i'^1 = |\{(i,k) \in A_N : \pi'_{A_{S_r}}(i,k) \leq l_r\}|$.

  Veamos que sucede con $x_i^1$. Como $p_i^k > p_i'^k$ se tiene que $\pi'_{A_{S_r}}(i,k) < \pi_{A_{S_r}}(i,k)$. Entonces

  $$x_i'^1 = |\{(i,k) \in A_N : \pi'_{A_{S_r}}(i,k) \leq l_r\}| \geq |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}| = x_i^1.$$

  Supongamos que $x_i'^1 = x_i^1 + n$ con $n \geq 0$. Veamos que sucede con $x_i^2$. Dados

  $$A' = \{(i,k_i) \in A_N : x_i^1 < k_i \leq c_i\} \text{ y } A'' = \{(i,k_i) \in A_N, x_i'^1 < k_i \leq c_i\}.$$

$A'' \subset A'$ y contiene $n$ pares menos que $A'$. Además para cada par $(i,k) \in A''$ se verifica que $\pi'_{A''}(i,k) \leq \pi_{A''}(i,k)$. Entonces, $x_i'^2 \geq x_i^2 - n$.

Por lo que

$$x_i'^1 + x_i'^2 \geq x_i^1 + n + x_i^2 - n = x_i^1 + x_i^2,$$

o lo que es lo mismo,

$$f_i(N,c,E,p,S) \leq f_i(N,c,E,p',S).$$

**Regla propuesta.**

Para cada $r \in \tilde{N}$ y cada $i \in S_r$,

$$f_i^2(N,c,E,p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \leq x_r\}|,$$

donde $x_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : \pi_{A'_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^m l_r\}|$ siendo $A'_{\tilde{N}} = \{(r,k_r) \in A_{\tilde{N}} : k_r > l_r\}$.

En el nuevo problema, tenemos que

$$f_i^2(N,c,E,p',S) = |\{(i,k_i) \in A_{S_r} : \pi'_{A_{S_r}}(i,k_i) \leq x'_r\}|,$$

donde $x'_r = l_r + |\{(r,k) \in A'_{\tilde{N}} : \pi'_{A'_{\tilde{N}}}(r,k) \leq E - \sum_{r=1}^m l_r\}|$ siendo $A'_{\tilde{N}} = \{(r,k_r) \in A_{\tilde{N}} : k_r > l_r\}$.

Al ser las puntaciones en el nuevo problema inferiores a las iniciales, en el problema entre grupos se verifica que las puntuaciones agregadas del grupo $r$ también serán inferiores y por tanto $\pi'_{A'_{\tilde{N}}}(r,k) \leq \pi_{A'_{\tilde{N}}}(r,k)$. Entonces, $x_r \leq x'_r$.

Dado que $\pi'_{A_{S_r}}(i,k) \leq \pi_{A_{S_r}}(i,k)$ se tiene que

$$f_i^2(N,c,E,p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \leq x_r\}| \leq$$

$$\leq |\{(i,k_i) \in A_{S_r} : \pi'_{A_{S_r}}(i,k_i) \leq x'_r\}| = f_i^2(N,c,E,p',S).$$

- **Monotonía respecto al estado.** Para cada $(N,c,E,p,S)$ y $(N,c,E',p,S)$ si $\sum_{i\in N} c_i > E' \geq E$ entonces,

$$f(N,c,E',p,S) \geq f(N,c,E,p,S).$$

**Regla aplicada por la universidad.**

Para cada $i \in S_r$

$$f_i^1(N,c,E,p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \leq E - \sum_{r=1}^m l_r\}|,$$

donde $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \leq l_r\}|$ y $A' = \{(i,k_i) \in A_N : k > x_i^1\}$.

Al aumentar el estado, $x_i^1$ sigue siendo igual, ya que el estado no está involucrado en su cálculo.

Denotaremos por $x_i^2$ al segundo sumando de $f_i^1(N,c,E,p,S)$. Al aumentar el estado, puede suceder que exista algún par $(i,k)$ de forma que $E - \sum_{r=1}^m l_r \leq \pi_{A'}(i,k) \leq E' - \sum_{r=1}^m l_r$. Por lo que $x_i^2 \leq x_i'^2$.

Entonces,

$$f^1(N,c,E',p,S) \geq f^1(N,c,E,p,S).$$

- **Regla propuesta:** Para cada $r \in \tilde{N}$ y cada $i \in S_r$,

$$f_i^2(N,c,E,p,S) = |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \le x_r\}|,$$

donde $x_r = l_r + |\{(r,k) \in A_{\tilde{N}} : \pi_{A_{\tilde{N}}}(r,k) \le E - \sum_{r=1}^m l_r\}|$ siendo $A'_{\tilde{N}} = \{(r,k_r) \in A_{\tilde{N}} : k_r > l_r\}$. Al aumentar el estado, puede suceder que exista algún par $(r,k)$ tal que $E - \sum_{r=1}^m l_r \le \pi_{A'_{\tilde{N}}}(r,k) \le E' - \sum_{r=1}^m l_r$ por lo que el nuevo $x_r$, al que denotaremos por $x'_r$ es mayor o igual. Entonces,

$$|\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \le x'_r\}| \ge |\{(i,k_i) \in A_{S_r} : \pi_{A_{S_r}}(i,k_i) \le x_r\}|,$$

Por lo tanto,

$$f^2(N,c,E',p,S) \ge f^2(N,c,E,p,S).$$

□

**Teorema 3.2.** *La regla aplicada por la Universidade de Vigo verifica la propiedad de composición hacia arriba, sin embargo, la regla propuesta no verifica dicha propiedad.*

*Demostración.*

- **Composición hacia arriba.** Si $E' > E > \sum_{r=1}^m l_r$, entonces

$$f(N,c,E',p,S) = f(N,c,E,p,S) + f(N,c - f(N,c,E,p,S), E' - E, p', S'),$$

donde

  - $S'_r = S_r$, $l'_r = 0$ para cada $r \in \{1,...,m\}$.
  - $p_i'^k = p_i^{k+f_i(N,c,E,p,S)}$ para cada $i \in N$ y cada $k \in \mathbb{N}$.

**Regla aplicada por la universidad.**

Sea $i \in N$ tal que $i \in S_r$, entonces

$$f_i^1(N,c,E,p,S) = x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \le E - \sum_{r=1}^m l_r\}|,$$

donde $x_i^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \le l_r\}|$

Además,

$$f_i^1(N,c,E',p,S) = x_i'^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \le E' - \sum_{r=1}^m l_r\}|$$

donde $x_i'^1 = |\{(i,k) \in A_N : \pi_{A_{S_r}}(i,k) \le l_r\}| = x_i^1$ y $A' = \{(i,k_i) \in A_N : k_i > x_i^1\}$.

Por otro lado,

$$f_i^1(N,c - f^1(N,c,E,p,S), E' - E, p', S') = |\{(i,k) \in A'' : \pi_{A''}(i,k) \le E' - E\}|,$$

donde $A'' = \{(i,k_i) : i \in N, k_i \le c_i - f_i^1(N,c,E,p,S)\}$ ya que, en este último caso, no hay derechos mínimos ($S'_r = S_r$, $l'_r = 0$ para cada $r \in \{1,...,m\}$).

Si sumamos,

$$f_i^1(N,c,E,p,S) + f_i^1(N,c - f(N,c,E,p,S), E' - E, p', S') =$$

$$= x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \le E - \sum_{r=1}^m l_r\}| + |\{(i,k) \in A'' : \pi_{A''}(i,k) \le E' - E\}|.$$

A continuación relacionamos los órdenes $\pi_{A''}$ y $\pi_{A'}$. Para ello utilizamos que $p_i'^k = p_i^{k+f_i(N,c,E,p,S)}$ para cada $i \in N$ y cada $k \in \mathbb{N}$.

Sean $(i,k)$, $(j,l) \in A''$,

$$\pi_{A''}(i,k) < \pi_{A''}(j,l) \Leftrightarrow p_i^{k+f_i^1(N,c,E,p,S)} < p_j^{l+f_j(N,c,E,p,S)} \Leftrightarrow$$
$$\Leftrightarrow \pi_{A'}(i,k+f_i^1(N,c,E,p,S)) < \pi_{A'}(j,l+f_j(N,c,,E,p,S)).$$

Por lo tanto, todos los pares $(i,k) \in A'$ tal que $k > f_i^1(N,c,E,p,S)$ se ordenan del mismo modo en $A'$ y $A''$.

Podemos separar $A'$ en dos subconjuntos,

$$\begin{aligned} A' &= \{(i,k_i) : i \in N, k_i = x_i^1+1,...,c_i\} \\ &= \{(i,k_i) : i \in N, k_i = x_i^1+1,...,f_i^1(N,c,E,p,S)\} \\ &\cup \{(i,k_i) : i \in N, k_i = f_i^1(N,c,E,p,S)+1,...,c_i\} \end{aligned}$$

En el primer subconjunto están los pares que se seleccionan cuando se reparten las $E - \sum_{r=1}^m l_r$ unidades del estado.

Los pares del segundo subconjunto coinciden con los pares en $A''$ y se ordenan de igual modo, por lo que es un conjunto equivalente a $A''$.

Entonces,

$$\begin{aligned} f_i^1(N,c,E,p,S) &+ f_i^1(N,c-f(N,c,E,p,S),E'-E,p',S') = \\ &= x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \le E - \sum_{r=1}^m l_r\}| + \\ &+ |\{(i,k) \in A' : E - \sum_{r=1}^m l_r \le \pi_{A'}(i,k) \le E' - \sum_{r=1}^m l_r\}| = \\ &= x_i^1 + |\{(i,k) \in A' : \pi_{A'}(i,k) \le E' - \sum_{r=1}^m l_r\} = f_i^1(N,c,E',p,S). \end{aligned}$$

**Regla propuesta.**

***Contraejemplo:*** Consideremos el problema $(N,c,E,p,S)$ donde:

- $N = \{1,2,3\}$.
- $c = (3,2,4)$.
- $E = 3$.
- $S = (S_r,l_r)_{r=1}^2$ donde $l_1 = l_2 = 1$. Esto es, cada grupo debe recibir al menos una unidad del estado.
- Las puntuaciones $p = \{p_i^k, i \in N, k \in \mathbb{N}\}$ vienen dadas en la siguiente tabla.

| Grupo 1 | | Grupo 2 |
|---|---|---|
| Agente 1 | Agente 2 | Agente 3 |
| 0.2 | 1.3 | 1.4 |
| 1.2 | 2.5 | 2.3 |
| 2.4 | 3.6 | 5 |
| 3.8 | 4.1 | 5.3 |

Para empezar, debemos considerar un problema en el que los agentes son los grupos. Dado un problema $(N, c, E, p, S)$ definimos el problema entre grupos $(\tilde{N}, \tilde{c}, \tilde{E}, \tilde{p}, \tilde{S})$, donde:

- $\tilde{N} = \{1, 2\}$. Los agentes son los grupos.
- $\tilde{c} = (5, 4)$. Demandas agregadas.
- $\tilde{E} = E = 3$.
- $\tilde{S} = (\{r\}, l_r)_{r \in \{1,\ldots,m\}}$.
- Las puntuaciones vienen dadas en la siguiente tabla:

| Grupo 1 | Grupo 2 |
|---|---|
| 1.5 | 1.4 |
| 3.7 | 2.3 |
| 6 | 5 |
| 7.9 | 5.3 |

Empezamos asignando el estado entre los grupos teniendo en cuenta que cada uno de ellos debe recibir al menos una unidad de estado (derecho mínimo). Entonces, $x_r = (1, 2)$.

Ahora, hay que repartir la asignación de cada grupo entre sus miembros, teniendo en cuenta sus puntuaciones. Por lo que

$$f^2(N, c, E, p, S) = (1, 0, 2).$$

Supongamos que el estado aumenta en dos unidades, $E' = 5$. Volvemos a calcular la asignación. Empezamos asignando el estado entre los grupos, $x'_r = (2, 3)$.

A continuación, se reparte la asignación de cada grupo entre sus miembros,

$$f^2(N, c, E', p, S) = (2, 0, 3).$$

Calculemos ahora $f^2(N, c - f(N, c, E, p, S), E' - E, p', S')$. En este caso el estado a repartir es $E' - E = 5 - 3 = 2$, el vector de demandas es $(3, 2, 4) - (1, 0, 2) = (2, 2, 2)$, no hay derechos mínimos para los grupos y las puntuaciones son las siguientes:

| Grupo 1 | | Grupo 2 |
|---|---|---|
| Agente 1 | Agente 2 | Agente 3 |
| 1.2 | 1.3 | 5 |
| 2.4 | 2.5 | 5,3 |
| 3.8 | 3,6 | |

Por lo que las puntuaciones agregadas son:

| Grupo 1 | Grupo 2 |
|---|---|
| 2.5 | 5 |
| 4.9 | 5.3 |

Empezamos asignando el estado entre los grupos. Entonces, $x_r'' = (2, 0)$.

Ahora, hay que repartir la asignación de cada grupo entre sus miembros, teniendo en cuenta sus puntuaciones. Por lo que

$$f^2(N, c - f(N, c, E, p, S), E' - E, p', S') = (1, 1, 0).$$

Sumando

$$f^2(N, c, E, p, S) + f^2(N, c - f(N, c, E, p, S), E' - E, p', S') =$$
$$= (1, 0, 2) + (1, 1, 0) = (2, 1, 2) \neq f^2(N, c, E', p, S) = (2, 0, 3).$$

Por lo que no se cumple la propiedad de composición hacia arriba.

□

En la Tabla 3.1 se puede ver de forma resumida que propiedades verifica cada una de las reglas de reparto vistas en este capítulo.

| | Regla de la Universidade de Vigo | Otra regla |
|---|---|---|
| Equilibrio | ✓ | ✓ |
| Conservación del orden | | |
| Conservación del orden dentro de grupos | ✓ | ✓ |
| Composición hacia arriba | ✓ | |
| Composición hacia abajo | ✓ | ✓ |
| Monotonía respecto a las demandas | ✓ | ✓ |
| Monotonía de pérdidas | ✓ | ✓ |
| Monotonía respecto a las puntuaciones | ✓ | ✓ |
| Monotonía respecto al estado | ✓ | ✓ |

Tabla 3.1: Propiedades que cumplen la regla de la Universidade de Vigo y la regla propuesta.

# Capítulo 4

# Reglas de reparto en R

En este capítulo se introduce el funcionamiento y la programación en `R` de algunas de las reglas de reparto para problemas con bienes indivisibles vistas anteriormente. Se van a considerar únicamente las que cumplan la propiedad de equilibrio, ya que es una propiedad de justicia. Por lo tanto consideramos:

- Regla ascendente orientada a demandas.
- Regla descendente orientada a demandas.
- Regla de igual ganancia discreta.
- Regla de igual pérdida discreta.
- Regla aplicada por la Universidade de Vigo.
- Regla propuesta.

En el caso de las reglas de igual ganancia e igual pérdida discretas, como ya se ha explicado anteriormente, se pueden utilizar las reglas de igual ganancia e igual pérdida vistas en el caso de bienes divisibles para obtener la asignación. Dichas reglas están ya programadas el paquete `GameTheory` de `R`, por lo que se va a hacer uso de dicho paquete. En Cano y Giménez (2015), se puede ver toda la información referente al paquete.

El paquete `GameTheory` tiene implementadas las reglas de reparto más usadas en los problemas clásicos de bancarrota: regla proporcional, regla de igual ganancia, regla de igual pérdida, Talmud y llegada aleatoria. Para obtener las asignaciones dadas por estas reglas de reparto necesitaremos proporcionar el estado y un vector con las demandas. También podemos proporcionar un vector con etiquetas que nombren a los agentes. Después de esto podemos ejecutar el comando `AllRules(E, c, names)`, donde $E$ es el estado, $c$ el vector de demandas y *names* el vector con los nombres de los agentes, para obtener las asignaciones dadas por todas las reglas o podemos ejecutar regla por regla con los comandos correspondientes: `Proportional()`, `CEA()`, `CEL()`, `Talmud()` y `RandomArrival()`.

En el caso de bienes divisibles, para el ejemplo del Talmud en el que hay que repartir 100 unidades monetarias entre las 3 esposas del hombre fallecido, utilizando el siguiente código se obtiene la asignación de cada una de las esposas mediante las 5 reglas de reparto vistas en el Capítulo 1:

```
> AllRules(100,c(100,200,300),c("Esposa 1","Esposa 2","Esposa 3"))

           Claim Proportional   CEA    CEL Talmud    RA
Esposa 1     100        16.67 33.33   0.00  33.33 33.33
Esposa 2     200        33.33 33.33   0.00  33.33 33.33
Esposa 3     300        50.00 33.33 100.00  33.33 33.33
Gini Index    NA         0.22  0.00   0.67   0.00  0.00
```

Si en lugar de tener $E = 100$, se tiene $E = 400$, se obtienen las siguientes asignaciones:

```
> AllRules(400,c(100,200,300),c("Esposa 1","Esposa 2","Esposa 3"))

Claim Proportional    CEA    CEL Talmud     RA
Esposa 1     100        66.67 100.00  33.33  50.00  66.67
Esposa 2     200       133.33 150.00 133.33 125.00 116.67
Esposa 3     300       200.00 150.00 233.33 225.00 216.67
Gini Index    NA         0.22   0.08   0.33   0.29   0.25
```

Se puede ver que las asignaciones coinciden con las calculadas en el primer capítulo, recogidas en la Tabla 1.3.

**Observación:** Hay que tener en cuenta, que en las reglas programadas en el paquete "`GameTheory`", el vector de demandas debe estar ordenado de menor a mayor.

Una vez explicado el funcionamiento del paquete `GameTheory` procedemos a programar cada una de las reglas de reparto mencionadas al inicio del capítulo. A continuación se muestra el código utilizado para programar cada una de ellas.

## 4.1. Regla de igual ganancia discreta

La primera función que se ha programado es la regla de igual ganancia discreta, la cual se ha denominado DCEA. Calcula la asignación mediante la regla de igual ganancia para bienes divisibles programada en el paquete `GameTheory`. Cabe destacar que para poder usar las funciones del paquete `GameTheory` es necesario ordenar de menor a mayor el vector de demandas. Los argumentos de la función son el estado, `E`, el vector de demandas, `c` y el vector de prioridades, `p`. La función DCEA empieza asignando a cada agente la parte entera de la asignación de la función CEA. Para los agentes que no habían recibido una cantidad entera mediante CEA se hace un reparto de lo que resta de estado tras restar las partes enteras asignadas. Esto se hace asignando unidad a unidad hasta agotar el estado siguiendo el orden de prioridad dado.

```
DCEA=function(E,c,p){
ord=order(c)  #Ordenamos el vector de demandas
c=c[ord]
CEA=CEA(E,c) #Función del paquete GameTheory
CEA=CEA$Results
CEA=CEA[ord]
DCEA=floor(CEA)
x=numeric() #Agentes que no recibieron una cantidad entera
for (i in 1:length(c)){
   if (DCEA[i]!=CEA[i]) {
      x[i]=i
   } else {x[i]=0}
}
#Asignar unidad a unidad siguiendo el orden de prioridad hasta agotar el estado
for (j in 1:length(c)) {
   if ((sum(p[j]==x))==1){
     if (sum(DCEA)<E) {
        DCEA[p[j]]=DCEA[p[j]]+1}
   }
}
```

```
DCEA=as.data.frame(DCEA)
colnames(DCEA)=c("DCEA")
return(DCEA)
}
```

Ejecutamos la función con los datos del ejemplo visto para esta regla de reparto en la Sección 2.2.3, en el que se quiere dividir un estado de 9 unidades entre 3 agentes cuyo vector de demandas es $c = (2, 6, 6)$. Además, consideraremos el vector de prioridades, $p = (2, 3, 1)$, es decir, $2\sigma 3\sigma 1$. La regla programada nos devuelve la siguiente asignación:

```
> DCEA(9,c(2,6,6),c(2,3,1))
DCEA
1    2
2    4
3    3
```

## 4.2. Regla de igual pérdida discreta

A continuación, se ha programado la regla de igual pérdida discreta, la cual se ha denominado DCEL. Calcula la asignación mediante la regla de igual pérdida para bienes divisibles programada en el paquete `GameTheory`. Los argumentos de la función son el estado, `E`, el vector de demandas, `c` y el vector de prioridades, `p`. La función DCEL empieza asignando a cada agente la parte entera de la asignación de la función CEL. Para los agentes que no habían recibido una cantidad entera mediante CEL se hace un reparto del resto del estado tras restar las partes enteras asignadas. Esto se hace asignando unidad a unidad hasta agotar el estado siguiendo el orden de prioridad dado.

```
DCEL=function(E,c,p){
ord=order(c) #Ordenamos el vector de demandas
c=c[ord]
CEL=CEL(E,c) #Función del paquete GameTheory
CEL=CEL$Results
CEL=CEL[ord]
DCEL=floor(CEL)
x=numeric() #Agentes que no recibieron una cantidad entera
for (i in 1:length(c)){
   if (DCEL[i]!=CEL[i]) {
     x[i]=i
   } else {x[i]=0}
}
#Se asignan unidad a unidad siguiendo el orden de prioridad hasta agotar el estado
for (j in 1:length(c)) {
   if ((sum(p[j]==x))==1){
      if (sum(DCEL)<E) {
         DCEL[p[j]]=DCEL[p[j]]+1}
   }
}
DCEL=as.data.frame(DCEL)
colnames(DCEL)=c("DCEL")
return(DCEL)
}
```

Ejecutando la función con los datos del ejemplo de la Sección 2.2.4 en el que se quiere dividir un estado de 5 unidades entre 3 agentes cuyo vector de demandas es $c = (2, 4, 6)$ y con vector de prioridades $p = (2, 3, 1)$, se obtiene la siguiente asignación:

```
> DCEL(5,c(2,4,6),c(2,3,1))
DCEL
1    0
2    2
3    3
```

## 4.3. Regla ascendente orientada a demandas

Otra regla de las que se ha programado es la regla ascendente orientada a demandas, la cual se ha denominado RAD. Los argumentos de la función son el estado, `E`, el vector de demandas, `c` y el vector de prioridades, `p`. En este caso, el vector de prioridades se considera como un vector de puntuaciones, cuanto más baja sea la puntuación del agente, mayor prioridad. La función RAD empieza creando una tabla con los datos (agente, demanda, prioridad y asignación). Al inicio del proceso, los agentes no tienen nada asignado. Se ordenan los datos según la demanda y la prioridad. Al agente con mayor prioridad, se le asigna una unidad del estado y se le resta una unidad de su demanda. Se vuelven a ordenar los datos para ver que agente tiene mayor prioridad y se repite el proceso hasta agotar el estado.

```
RAD=function(E,c,p){
n=seq(1:length(c)) #vector de agentes
x=rep(0,length(c)) #vector de asignaciones
pares=data.frame(Agente=n,Demanda=c,Prioridad=p,Asignación=x) #Tabla de datos
while (sum(pares$Asignación)<E) {
   pares=pares[order(-pares$Demanda,pares$Prioridad),]
   pares$Asignación[1]=pares$Asignación[1]+1
   pares$Demanda[1]=pares$Demanda[1]-1
}
pares$Asignación=as.data.frame(pares$Asignación)
colnames(pares$Asignación)=c("RAD")
return(pares$Asignación)}
```

Supongamos que queremos repartir 6 unidades de estado entre 3 agentes cuyo vector de demandas es $c = (2, 5, 3)$ y cuyo vector de prioridades es $p = (3, 1, 2)$, es decir, $2\sigma 3\sigma 1$. Ejecutamos la función con estos datos y obtenemos la siguiente asignación:

```
> RAD(6,c(2,5,3),c(3,1,2))
RAD
1   0
2   4
3   2
```

## 4.4. Regla descendente orientada a demandas

De forma similar a la regla ascendente orientada a demandas, se ha programado la regla descendente orientada a demandas, a la cual se ha denominado RDD. Al igual que en caso anterior, los argumentos de la función son el estado, `E`, el vector de demandas, `c` y el vector de prioridades, `p`. En este caso, el vector de prioridades se considera como una puntuación a cada agente, cuanto más baja sea la puntuación, mayor prioridad. La función RDD empieza creando una tabla con los datos (agente, demanda, prioridad y asignación). En este caso cada agente empieza con su demanda como asignación. Se ordenan los datos según la demanda y la prioridad. Al agente de mayor prioridad, se le asigna una unidad de déficit y se le resta una unidad de su demanda. Se vuelven a ordenar los datos para ver que agente tiene mayor prioridad y se repite el proceso hasta tener asignado completamente el estado.

```
RDD=function(E,c,p){
n=seq(1:length(c)) #Vector de agentes
pares=data.frame(Agente=n,Demanda=c,Prioridad=p,Asignación=c) #Tabla de datos
while (sum(pares$Asignación)>E) {
   pares=pares[order(-pares$Demanda,pares$Prioridad),]
   pares$Asignación[1]=pares$Asignación[1]-1
   pares$Demanda[1]=pares$Demanda[1]-1
}
pares$Asignación=as.data.frame(pares$Asignación)
colnames(pares$Asignación)=c("RDM")
return(pares$Asignación)}
```

Vamos a considerar el mismo ejemplo que para la regla ascendente orientada a demandas. Entonces, $E = 6$ ,$c = (2, 5, 3)$ y $p = (3, 1, 2)$. En este caso, ejecutando la función con estos datos se obtiene la siguiente asignación:

```
> RDD(6,c(2,5,3),c(3,1,2))
RDD
1   2
2   2
3   2
```

## 4.5. Regla de la Universidade de Vigo

Otra de las reglas que se ha programado es la regla de la Universidade de Vigo, la cual se ha denominado UV. Los argumentos de la función son el estado, `E`, el vector de derechos mínimos de los grupos, `S` y un archivo .csv que contiene los datos necesarios (agente, grupo, demanda y puntuaciones para optar a cada una de las unidades del estado). La función UV empieza creando una tabla con los datos (agente, grupo, demanda, puntuación para optar a la primera unidad del estado y asignación). Cada agente empieza con una asignación de 0 unidades. A continuación se asigna a cada grupo su derecho mínimo. Para ello se ordenan los agentes de cada grupo según su puntuación y se asigna al agente con menor puntuación. Finalmente, se asigna el estado restante entre los agentes sin tener en cuenta el grupo al que pertenecen. Para ello se ordenan los datos según la puntuación de cada agente y se va asignando unidad a unidad al agente con menor puntuación. Hay que tener en cuenta que cada vez que se asigna una unidad de estado a un agente hay que actualizar su puntuación.

```
UV=function(E,S,dat){
levA=levels(dat$Agente)
levels(dat$Agente)=c(1:length(levels(dat$Agente)))
levG=levels(dat$Grupo)
levels(dat$Grupo)=c(1:length(levels(dat$Grupo)))
p=dat[,4:ncol(dat)] #Tabla con las puntuaciones
x=rep(0,nrow(dat))  #Vector de asignaciones
datos=data.frame(dat[1:3],Puntuación=dat[,4],Asignación=x) #Tabla de datos
#Se asigna a cada grupo su derecho mínimo
for(i in 1:length(S)) {
   while (sum(datos[datos$Grupo==i,]$Asignación)<S[i]) {
      datos$Puntuación[datos$Demanda==datos$Asignación]=1000
      datos=datos[order(datos$Puntuación),]
      datos[datos$Grupo==i,]$Asignación[1]=datos[datos$Grupo==i,]$Asignación[1]+1
      datos[datos$Grupo==i,]$Puntuación[1]=p[datos[datos$Grupo==i,]$Agente[1],
```

```
      datos[datos$Grupo==i,]$Asignación[1]+1]
   }
}
#Se asigna el estado restante entre los agentes
while (sum(datos$Asignación)<E) {
   datos$Puntuación[datos$Demanda==datos$Asignación]=1000
   datos=datos[order(datos$Puntuación),]
   datos$Asignación[1]=datos$Asignación[1]+1
   datos$Puntuación[1]=p[datos$Agente[1],datos$Asignación[1]+1]
}
datos=datos[order(datos$Agente),]
levels(datos$Agente)=levA
levels(datos$Grupo)=levG
sol=data.frame(datos$Agente,datos$Asignación)
colnames(sol)=c("Agente","Asignación")
#Si hay más de 10 agentes, sólo se mostrarán los que reciben alguna unidad del estado
if (length(sol$Agente)<10){
return(sol)  } else {
sol=data.frame(datos$Agente[datos$Asignación!=0],datos$Asignación[datos$Asignación!=0])
colnames(sol)=c("Agente","Asignación")
return(sol)}}
```

Vamos a ejecutar la función con los datos reales de las áreas de la Universidade de Vigo que solicitan plazas de titular en la convocatoria del 2017. Se quieren repartir 7 plazas de profesor titular entre las áreas de la universidad que se agrupan en 4 ámbitos (científico, tecnológico, jurídico-social y humanístico). Los datos necesarios, junto con la resolución real de las áreas seleccionadas, se recogen en el documento que se encuentra en el Apéndice B. Antes de ejecutar la función, se crea un archivo .csv en donde se recoge el agente, el grupo al que pertenece, su demanda y las puntuaciones para optar a cada una de las unidades del estado (Figura 4.1). Además, hay que indicar en derecho mínimo de cada grupo, en este caso, una unidad por grupo.

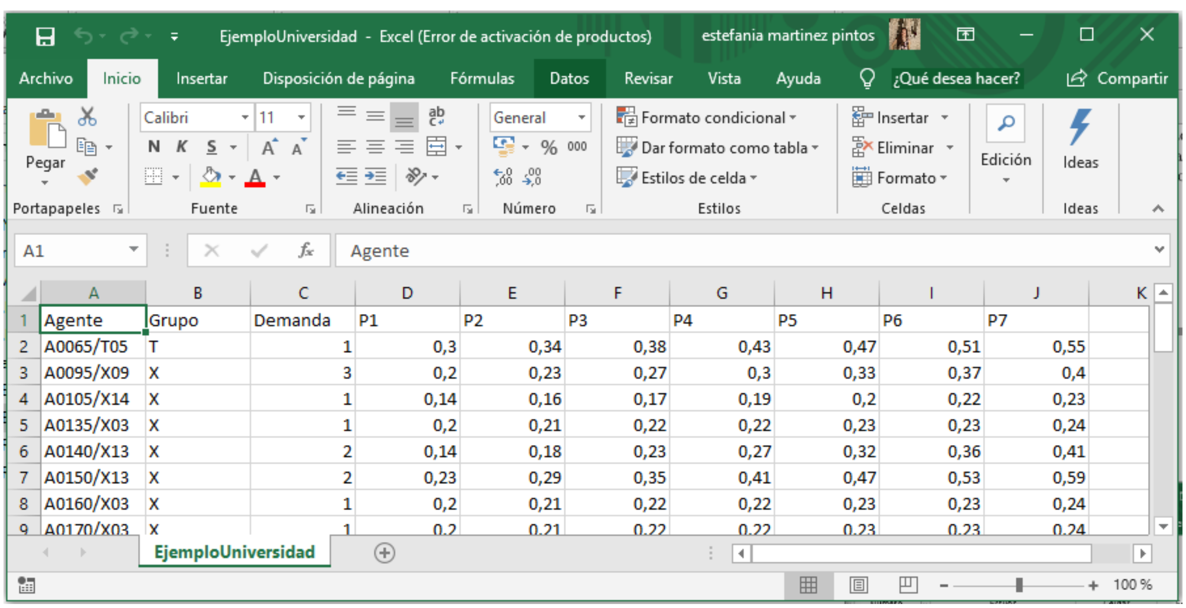



Figura 4.1: Archivo .csv con los datos

```
> UV(7,c(1,1,1,1),dat)
Agente Asignación
1 A0105/X14          1
2 A0140/X13          1
3 A0245/X05          1
4 A0265/C05          1
5 A0545/T03          1
6 A0650/X09          1
7 A0814/H12          1
```

Cabe destacar que el resultado obtenido coincide con las áreas de conocimiento seleccionadas en realidad.

## 4.6. Regla propuesta

Por último se ha programado la regla propuesta, la cual se ha denominado RP. Los argumentos de la función son el estado, `E`, el vector de derechos mínimos de los grupos, `S` y un archivo .csv que contiene los datos necesarios (agente, grupo, demanda y puntuaciones para optar a cada una de las unidades del estado). La función RP empieza creando una tabla con los datos (agente, grupo, demanda, puntuación para optar a la primera unidad del estado y asignación) y otra tabla con los datos de los grupos (grupo, demanda agregada, puntuación agregada para la primera unidad del estado y asignación). Cada agente y cada grupo empieza con una asignación de 0 unidades. Primero se asigna a cada grupo su derecho mínimo y a continuación se reparte el estado restante entre los grupos. Para ello se ordenan los grupos según su puntuación y se asigna una unidad al grupo con menor puntuación. Se repite el proceso hasta tener asignado el estado completamente. Después, hay que repartir el estado asignado a cada grupo entre los agentes que lo forman. Para ello se ordenan los datos correspondientes a cada grupo según la puntuación de cada agente y se va asignando unidad a unidad al agente con menor puntuación. Hay que tener en cuenta que cada vez que se asigna una unidad de estado a un agente o a un grupo hay que actualizar su puntuación.

```
RP=function(E,S,dat){
levA=levels(dat$Agente)
levels(dat$Agente)=c(1:length(levels(dat$Agente)))
levG=levels(dat$Grupo)
levels(dat$Grupo)=c(1:length(levels(dat$Grupo)))
g=dat[,2]
p=dat[,4:ncol(dat)] #Tabla con las puntuaciones
ca=vector() #Vector de demandas agregadas
for (i in 1:length(S)) {
   ca[i]=sum(dat$Demanda[g==i])
}
pa=matrix(nrow = length(S),ncol=ncol(p)) #Matriz de puntuciones agregadas
for (i in 1:length(S)) {
   pa[i,]=colSums(p[g==i,])
}
x=rep(0,length(dat$Agente)) #Vector de asignaciones de los agentes
# Tabla con los datos de los agentes
datos0=data.frame(dat[1:3],Puntuación=dat[,4],Asignación=x)
y=rep(0,length(S)) #Vector de asignaciones de los grupos
#Tabla con los datos de los grupos
datos=data.frame(Grupos=seq(1:length(S)),Demanda=ca,Puntuación=pa[,1],Asignación=y)
#Se asigna el derecho mínimo a cada grupo
```

```
for (i in 1:length(S)) {
   datos$Asignación[i]=S[i]
   datos$Puntuación[i]=pa[i,datos$Asignación[i]+1]
}
#Se reparte el estado restante entre los grupos
while (sum(datos$Asignación)<E) {
   datos$Puntuación[datos$Demanda==datos$Asignación]=9999
   datos=datos[order(datos$Puntuación,datos$Grupos),]
   datos$Asignación[1]=datos$Asignación[1]+1
   datos$Puntuación[1]=pa[datos$Grupos[1],datos$Asignación[1]+1]
}
datos=datos[order(datos$Grupos),]
#Se distribuye la asignación de cada grupo entre los agentes que lo forman
E=numeric() #Vector con la asignación de cada grupo
for (i in 1:length(S)) {
   E[i]=datos$Asignación[i]
   while(sum(datos0[datos0$Grupo==i,]$Asignación)<E[i]){
      datos0$Puntuación[datos0$Demanda==datos0$Asignación]=9999
      datos0[datos0$Grupo==i,]=datos0[datos0$Grupo==i,][order(datos0[datos0$Grupo==i,
      ]$Puntuación),]
      datos0[datos0$Grupo==i,]$Asignación[1]=datos0[datos0$Grupo==i,]$Asignación[1]+1
      datos0[datos0$Grupo==i,]$Puntuación[1]=p[datos0[datos0$Grupo==i,]$Agente[1],
      datos0[datos0$Grupo==i,]$Asignación[1]+1]
   }
}
datos0=datos0[order(datos0$Agente),]
levels(datos0$Agente)=levA
levels(datos0$Grupo)=levG
sol=data.frame(datos0$Agente,datos0$Asignación)
colnames(sol)=c("Agente","Asignación")
#Si hay más de 10 agentes, sólo se mostrarán los que reciban alguna unidad del estado
if (length(sol$Agente)<10){
   return(sol)  } else {
   sol=data.frame(datos0$Agente[datos0$Asignación!=0],datos0$Asignación[
   datos0$Asignación!=0])
   colnames(sol)=c("Agente","Asignación")
return(sol)
}
}
```

Veamos ahora que sucede si utilizamos la regla propuesta en lugar de la regla de la Universidade de Vigo.

```
> RP(7,c(1,1,1,1),dat)
Agente Asignación
1 A0105/X14          1
2 A0185/H09          1
3 A0265/C05          1
4 A0545/T03          1
5 A0814/H12          3
```

Se puede ver que las áreas A0105/X14, A0265/C05 y A0545/T03 siguen recibiendo lo mismo; el área A0814/H12 ahora recibe 3 plazas; las áreas A0140/X13, A0245/X05 y A0650/X09 ya no reciben nada y el área A0185/H09 que antes no recibía nada, ahora recibe una plaza.

## 4.7. Aplicación

Una vez programadas todas las reglas de reparto, se ha creado una aplicación utilizando el paquete `shiny` de R. Dicho paquete permite construir aplicaciones web interactivas a partir de los scripts de R que permiten a los usuarios interactuar con sus datos sin tener que manipular el código.

Esta aplicación permite resolver problemas de bancarrota con bienes indivisibles mediante las reglas de reparto programadas introduciendo únicamente los datos (estado, demandas, prioridades).

Una aplicación de `shiny` consta, al menos, de dos archivos. En el primero, `ui.R`, se controla el diseño y el aspecto de la aplicación. Y en el segundo, `server.R`, se realizan los cálculos necesarios.

El código de una aplicación `shiny` puede estar también dentro de un solo script, como es el caso de la aplicación que se ha programado. El código utilizado para la creación de la aplicación se encuentra en el Apéndice A. Una vez se ejecuta el código se obtiene la aplicación que se muestra en la Figura 4.2. Se puede ver que en la parte superior, debajo del título, hay tres pestañas, en cada una de ellas se tienen dos reglas de reparto. La Figura 4.2 recoge precisamente las reglas de igual ganancia e igual pérdida discretas. En el lado izquierdo de la pantalla debemos indicar el estado que queremos repartir, el vector de demandas, el vector de prioridades y por último, cual de las dos reglas queremos utilizar. En el lado derecho de la pantalla, se tiene una pequeña explicación de como funcionan las reglas de reparto y debajo, la asignación obtenida.

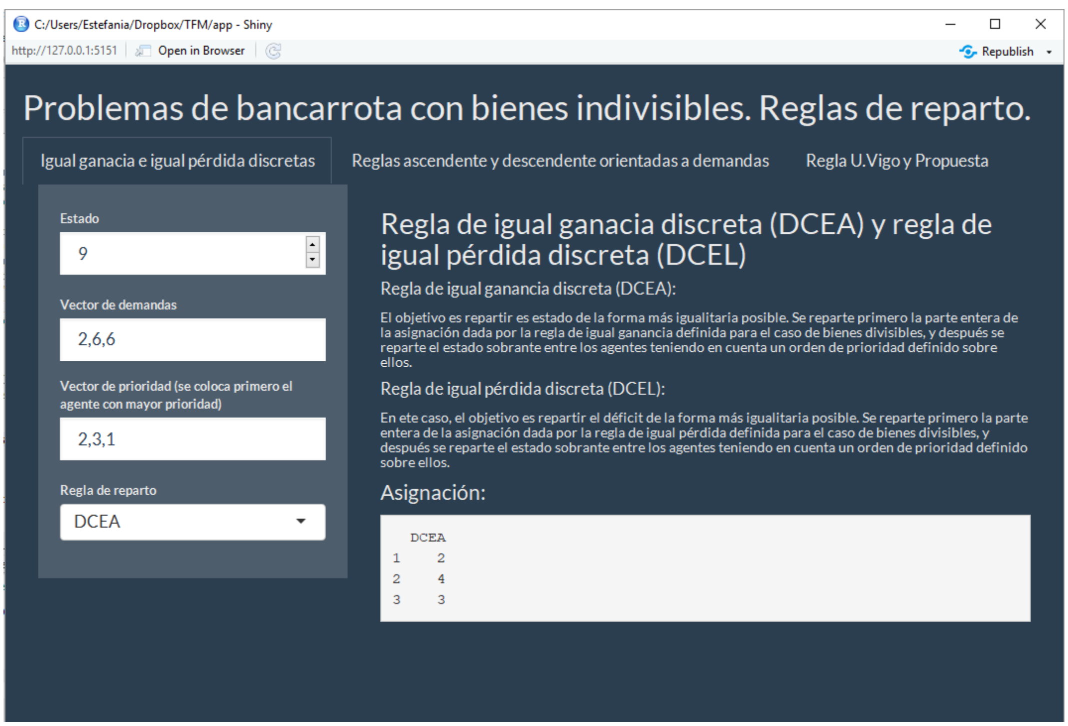



Figura 4.2: Aplicación shiny con la selección para las reglas de igual ganancia e igual pérdida discretas.

En la segunda pestaña (Figura 4.3), se tienen las reglas de reparto ascendente y descendente orientadas a demandas. Al igual que en la pestaña anterior, en el lado izquierdo de la pantalla se indicará el estado, el vector de demandas, el vector de prioridades y la regla que se quiere utilizar. En el lado derecho se tiene una pequeña explicación de las reglas y por último la asignación obtenida.

Por último, en la tercera pestaña (Figuras 4.5 y 4.6 ) se tiene la regla aplicada por la Universidade

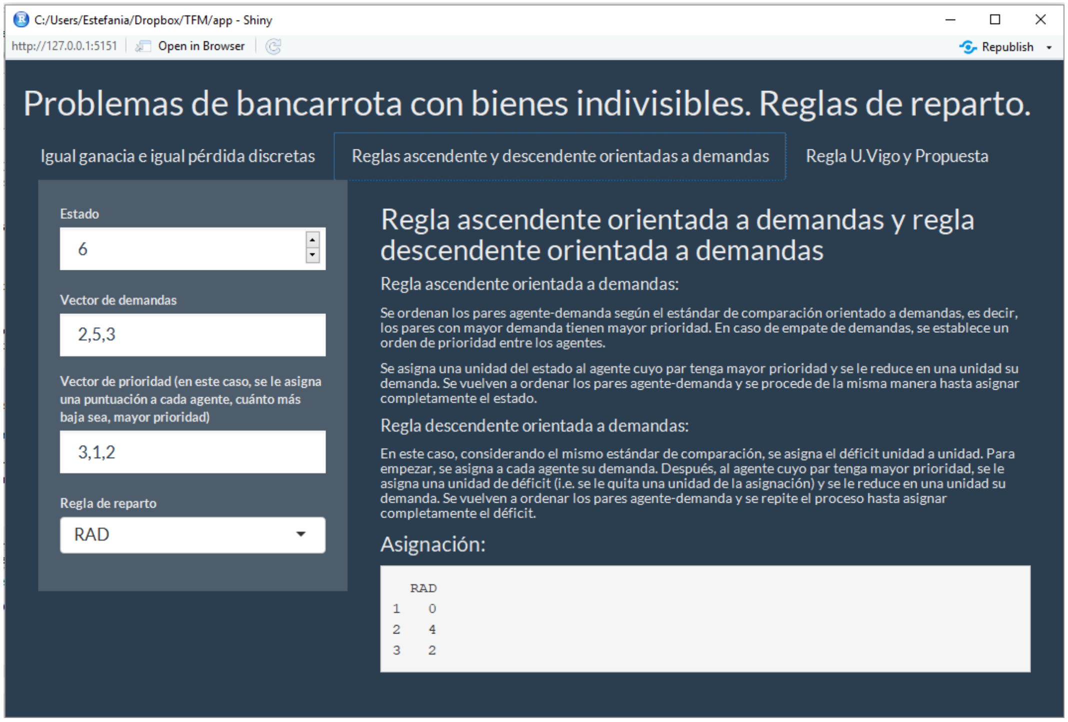
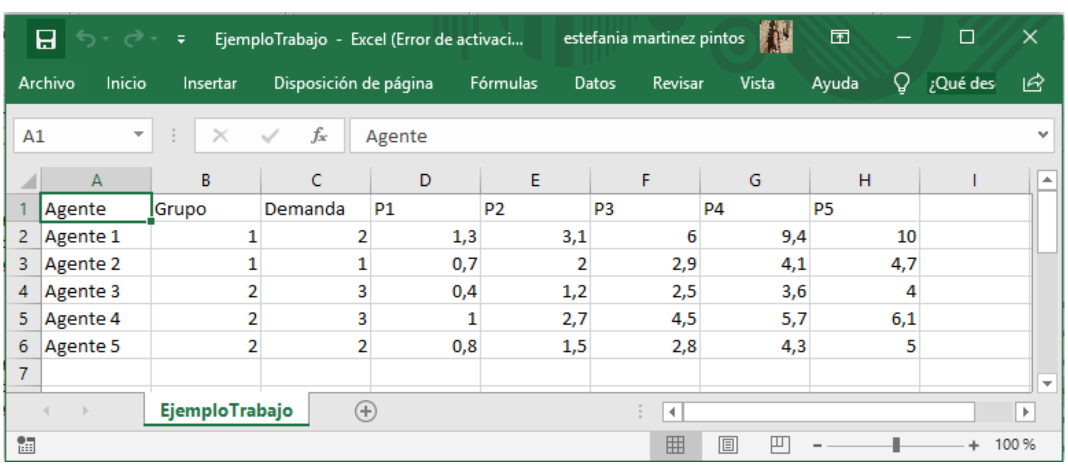



Figura 4.3: Aplicación shiny con la selección para las reglas ascendente y descendente orientadas a demandas.

de Vigo y la regla propuesta. En este caso se debe indicar el estado, el derecho mínimo de cada grupo, la regla que se desea utilizar, y a diferencia de las reglas anteriores, se debe cargar un archivo .csv en el que se proporcionen los datos (el agente, el grupo al que pertenece, su demanda, y su puntuación para conseguir cada una de las unidades del estado). En la Figura 4.4 se muestra un ejemplo de como debe ser el archivo .csv. Al igual que en las pestañas anteriores, en el lado derecho de la pantalla se tiene una breve explicación de la reglas de reparto y la asignación obtenida. En caso de tener un problema con más de diez agentes, se mostrarán sólo los agentes que que reciben alguna unidad del estado.




| Agente | Grupo | Demanda | P1 | P2 | P3 | P4 | P5 |
|---|---|---|---|---|---|---|---|
| Agente 1 | 1 | 2 | 1,3 | 3,1 | 6 | 9,4 | 10 |
| Agente 2 | 1 | 1 | 0,7 | 2 | 2,9 | 4,1 | 4,7 |
| Agente 3 | 2 | 3 | 0,4 | 1,2 | 2,5 | 3,6 | 4 |
| Agente 4 | 2 | 3 | 1 | 2,7 | 4,5 | 5,7 | 6,1 |
| Agente 5 | 2 | 2 | 0,8 | 1,5 | 2,8 | 4,3 | 5 |

Figura 4.4: Archivo .csv con los datos para el caso de la regla aplicada por la Universidade de Vigo y la regla propuesta.


Figura 4.5: Aplicación shiny con la selección para la regla aplicada por la Universidade de Vigo y la regla propuesta.

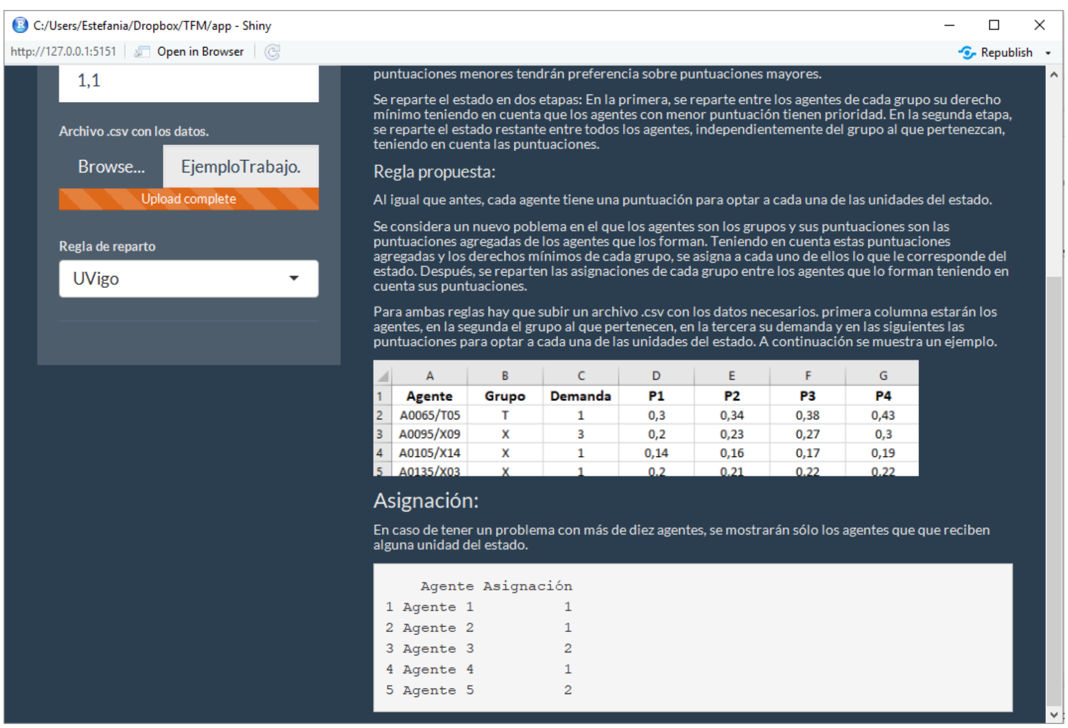



Figura 4.6: Aplicación shiny con la selección para la regla aplicada por la Universidade de Vigo y la regla propuesta.

# Apéndice A

# Código de la aplicación shiny.

```
#Se cargan los paquetes necesarios
library(shiny)
library(GameTheory)
library(shinythemes)
#Se ejecutan los archivos con las reglas programadas
source('ReglasReparto.R')
source('ReglasUniversidade.R')

#Se crea la aplicación
u <- shinyUI(fluidPage(theme = shinytheme("superhero"),
  titlePanel("Problemas de bancarrota con bienes indivisibles. Reglas de reparto."),
  tabsetPanel(
    tabPanel("Igual ganacia e igual pérdida discretas",
      sidebarPanel(
        numericInput('E1','Estado',"9",min = 0),
        textInput('vec1', 'Vector de demandas', "2,6,6"),
        textInput('p1', 'Vector de prioridad (se coloca primero el agente con mayor
        prioridad)',"2,3,1"),
        selectInput('regla1',label = 'Regla de reparto',choices = c("DCEA"="DCEA",
        "DCEL"="DCEL"))
      ),
      mainPanel(
        h3('Regla de igual ganacia discreta (DCEA) y regla de igual pérdida discreta
          (DCEL)'),
        h5('Regla de igual ganancia discreta (DCEA):'),
        h6('El objetivo es repartir es estado de la forma más igualitaria posible.
          Se reparte primero la parte entera de la asignación dada por la regla de
          igual ganancia definida para el caso de bienes divisibles, y después se
          reparte el estado sobrante entre los agentes teniendo en cuenta un orden
          de prioridad definido sobre ellos.'),
        h5('Regla de igual pérdida discreta (DCEL):'),
        h6('En ete caso, el objetivo es repartir el déficit de la forma más igualitaria
          posible. Se reparte primero la parte entera de la asignación dada por la
          regla de igual pérdida definida para el caso de bienes divisibles, y después
          se reparte el estado sobrante entre los agentes teniendo en cuenta un orden
          de prioridad definido sobre ellos.'),
        h4('Asignación:'),
```

```
      verbatimTextOutput("oid1")
    )
  ),
  tabPanel("Reglas ascendente y descendente orientadas a demandas",
    sidebarPanel(
      numericInput('E2','Estado',"6",min = 0),
      textInput('vec2', 'Vector de demandas', "2,5,3"),
      textInput('p2', 'Vector de prioridad (en este caso, se le asigna una
        puntuación a cada agente, cuánto más baja sea, mayor prioridad)', "3,1,2"),
      selectInput('regla2',label='Regla de reparto',choices = c("RAD"="RAD",
        "RDD"="RDD"))
    ),
    mainPanel(
      h3('Regla ascendente orientada a demandas y regla descendente orientada a
        demandas'),
      h5('Regla ascendente orientada a demandas:'),
      h6('Se  ordenan los pares agente-demanda según el estándar de comparación
        orientado a demandas, es decir, los pares con mayor demanda tienen mayor
        prioridad. En caso de empate de demandas, se establece un orden de
        prioridad entre los agentes.'),
      h6('Se asigna una unidad del estado al agente cuyo par tenga mayor prioridad
        y se le reduce en una unidad su demanda. Se vuelven a ordenar los pares
        agente-demanda y se procede de la misma manera hasta asignar completamente
        el estado.'),
      h5('Regla descendente orientada a demandas:'),
      h6('En este caso, considerando el mismo estándar de comparación, se asigna
        el déficit unidad a unidad. Para empezar, se asigna a cada agente su
        demanda. Después, al agente cuyo par tenga mayor prioridad, se le asigna
        una unidad de déficit (i.e. se le quita una unidad de la asignación) y se
        le reduce en una unidad su demanda. Se vuelven a ordenar los pares
        agente-demanda y se repite el proceso hasta asignar completamente el
        déficit.'),
      h4('Asignación:'),
      verbatimTextOutput("oid2")
    )
  ),
  tabPanel("Regla U.Vigo y Propuesta",
    sidebarPanel(
      numericInput('E3','Estado',"7",min=0),
      textInput('S','Derecho mínimo de cada grupo',"1,1,1,1"),
      fileInput("dat", "Archivo .csv con los datos.",
        accept = c("text/csv","text/comma-separated-values,text/plain",".csv")),
      selectInput('regla3',label='Regla de reparto',choices = c("UVigo"="UVigo",
        "Propuesta"="Propuesta")),
         tags$hr()
    ),
    mainPanel(
      h3('Regla utilizada por la Universidade de Vigo y regla propuesta'),
      h5('Regla utilizada por la Universidade de Vigo:'),
      h6('En este caso, cada agente va a tener una puntuación para optar a cada
        una de las unidades del estado. Las puntuaciones menores tendrán preferencia
        sobre puntuaciones mayores.'),
```

```
        h6('Se reparte el estado en dos etapas: En la primera, se reparte entre los
          agentes de cada grupo su derecho mínimo teniendo en cuenta que los agentes
          con menor puntuación tienen prioridad. En la segunda etapa, se reparte el
          estado restante entre todos los agentes, independientemente del grupo al
          que pertenezcan, teniendo en cuenta las puntuaciones.'),
        h5('Regla propuesta:'),
        h6('Al igual que antes, cada agente tiene una puntuación para optar a cada
          una de las unidades del estado.'),
        h6('Se considera un nuevo poblema en el que los agentes son los grupos y sus
          puntuaciones son las puntuaciones agregadas de los agentes que los forman.
          Teniendo en cuenta estas puntuaciones agregadas y los derechos mínimos de
          cada grupo, se asigna a cada uno de ellos lo que le corresponde del
          estado. Después, se reparten las asignaciones de cada grupo entre los
          agentes que lo forman teniendo en cuenta sus puntuaciones.'),
        h6('Para ambas reglas hay que subir un archivo .csv con los datos
          necesarios. En la primera columna estarán los agentes, en la segunda el
          grupo al que pertenecen, en la tercera su demanda y en las siguientes las
          puntuaciones para optar a cada una de las unidades del estado. A
          continuación se muestra un ejemplo.'),
          img(src="ejemplo.png",align="down",width="500px"),
        h4('Asignación:'),
        h6('En caso de tener un problema con más de diez agentes, se mostrarán sólo
          los agentes que que reciben alguna unidad del estado.'),
        verbatimTextOutput("oid3")
      )
    )
  )
))

s <- shinyServer(function(input, output) {
  output$oid1 <- renderPrint({
    E1<-as.numeric(input$E1)
    c1 <- as.numeric(unlist(strsplit(input$vec1,",")))
    p1 <- as.numeric(unlist(strsplit(input$p1,",")))
    if(input$regla1=="DCEA"){
      DCEA=DCEA(E1,c1,p1)
      print(DCEA)} else {
      DCEL=DCEL(E1,c1,p1)
      print(DCEL)
    }
  })
  output$oid2 <- renderPrint({
    E2<-as.numeric(input$E2)
    c2 <- as.numeric(unlist(strsplit(input$vec2,",")))
    p2 <- as.numeric(unlist(strsplit(input$p2,",")))
    if(input$regla2=="RAD"){
      RAD=RAD(E2,c2,p2)
      print(RAD)} else {
      RDD=RDD(E2,c2,p2)
      print(RDD)
    }
  })
```

```
  output$oid3 <- renderPrint({
    inFile <- input$dat
    if (is.null(inFile))
      return(NULL)
    dat=read.csv2(inFile$datapath, header = T,dec=",")
    dat=as.data.frame(dat)
    E3=as.numeric(input$E3)
    S<-as.numeric(unlist(strsplit(input$S,",")))
    if(input$regla3=="UVigo"){
      UVigo=UV(E3,S,dat)
      print(UVigo)} else {
      RP=RP(E3,S,dat)
      print(RP)
    }
  })
})

#Se ejecuta la aplicación
shinyApp(ui = u, server = s)
```

# Apéndice B

# Resolución de áreas de conocimiento seleccionadas en la Universidade de Vigo.

En este apéndice se incluye el documento pdf en el que se recoge la relación de las áreas de conocimiento con las puntuaciones obtenidas, datos con los que se crea el archivo .csv necesario para poder utilizar la regla aplicada por la Universidade de Vigo y la regla propuesta. Además, se recoge la resolución definitiva de áreas de conocimiento seleccionadas, que se compara con los resultados obtenidos aplicando ambas reglas de reparto.

**ACORDO DA COAP DO 18 DE SETEMBRO DE 2018 SOBRE A SELECCIÓN DE ÁREAS DE COÑECEMENTO PARA PARTICIPAR NA CONVOCATORIA DE PRAZAS DE PROFESORADO TITULAR DE UNIVERSIDADE NO ANO 2017**

No Consello de Goberno que tivo lugar o 21 de marzo de 2018, aprobáronse os criterios para a selección das áreas de coñecemento na convocatoria de prazas de profesorado titular de universidade no ano 2017, delegando na Comisión de Organización Académica e Profesorado (COAP), a selección destas áreas de coñecemento e a creación das prazas correspondentes.

En aplicación deste acordo do Consello de Goberno e unha vez finalizado o prazo de alegacións establecido no acordo provisional da COAP da sesión do 30 de xullo de 2017 sobre a selección de áreas, a COAP na súa sesión do 18 de setembro de 2017, e unha vez resoltas as alegacións presentadas, tomou o seguinte **ACORDO:**

1) Aprobar a convocatoria e a creación de sete prazas de Profesorado Titular de Universidade nas áreas de coñecemento relacionadas no **anexo I.**
2) Publicar as puntuacións definitivas, unha vez aplicados os criterios aprobados polo Consello de Goberno de 21 de marzo de 2018, para participar na selección de áreas de coñecemento **(anexo II)**.

Contra esta resolución, que esgota a vía administrativa, poderase interpoñer recurso perante a xurisdición contencioso administrativa, no prazo de dous meses contados dende o día seguinte á súa publicación (ou dende o día seguinte ao recibimento da súa notificación), de conformidade co disposto na Lei 29/1998, do 13 de xullo, reguladora da xurisdición contencioso administrativa.

As persoas interesadas poderán optar por interpoñer contra esta resolución recurso de reposición, no prazo dun mes contados a partir do dia seguinte á publicación da presente resolución (ou dende o día seguinte ao recibo desta notificación), perante o mesmo órgano que a ditou. Neste caso non se poderá interpoñer un recurso contencioso-administrativo ata que sexa resolto expresamente ou se producira a desestimación presunta do recurso de reposición interposto, segundo o previsto no artigo 123.2 da lei 39/2015, de 1 de outubro, do Procedemento Administrativo Común das Administracións Públicas.

Vigo, 18 de setembro de 2018

O presidente da COAP

Manuel Ramos Cabrer

## ANEXO I.

**Relación definitiva, por orde alfabético, de áreas de coñecemento seleccionadas, segundo os criterios aprobados en Consello de Goberno (21/03/2018)**

| ÁREA | DPTO. | CAMPUS |
|---|---|---|
| COMUNICACIÓN AUDIOVISUAL E PUBLICIDADE (105) | COMUNICACIÓN AUDIOVISUAL E PUBLICIDADE (X14) | PONTEVEDRA |
| DEREITO DO TRABALLO E DA SEGURIDADE SOCIAL (140) | DEREITO PÚBLICO ESPECIAL (X13) | VIGO |
| EDUCACIÓN FÍSICA E DEPORTIVA (245) | DIDÁCTICAS ESPECIAIS (X05) | PONTEVEDRA |
| ENXEÑARÍA MECÁNICA (545) | ENXEÑARÍA MECÁNICA, MÁQUINAS E MOTORES TÉRMICOS E FLUÍDOS (T03) | VIGO |
| ESTATÍSTICA E INVESTIGACIÓN OPERATIVA (265) | ESTATÍSTICA E INVESTIGACIÓN OPERATIVA (C05) | OURENSE |
| ORGANIZACIÓN DE EMPRESAS (650) | ORGANIZACIÓN DE EMPRESAS E MÁRKETING (X09) | VIGO |
| TRADUCIÓN E INTERPRETACIÓN (814) | TRADUCIÓN E LINGÜÍSTICA (H12) | VIGO |

UNIVERSIDADE DE VIGO 1990
REITORÍA

**ANEXO II**

**Relación de áreas de coñecemento coas puntuacións obtidas, segundo os criterios aprobados en Consello de Goberno (21/03/2018)**

| ÁREA/DPTO. | H | C | F | Q | CAMPUS |
|---|---|---|---|---|---|
| A0105/X14 | 6428 | 64,28 | 9 | 0,14 | PONTEVEDRA |
| A0140/X13 | 2220,15 | 22,2015 | 3 | 0,14 | VIGO |
| A0245/X05 | 4740,75 | 47,4075 | 7 | 0,15 | PONTEVEDRA |
| A0650/X09 | 12364,2 | 123,642 | 20 | 0,16 | VIGO |
| A0650/X09 | 12364,2 | 123,642 | 21 | 0,17 | |
| A0245/X05 | 4740,75 | 47,4075 | 8 | 0,17 | |
| A0650/X09 | 12364,2 | 123,642 | 22 | 0,18 | |
| A0140/X13 | 2220,15 | 22,2015 | 4 | 0,18 | |
| A0545/T03 | 2218 | 22,18 | 4 | 0,18 | VIGO |
| A0245/X05 | 4740,75 | 47,4075 | 9 | 0,19 | |
| A0135/X03 | 16174,6 | 161,746 | 33 | 0,20 | |
| A0160/X03 | 16174,6 | 161,746 | 33 | 0,20 | |
| A0170/X03 | 16174,6 | 161,746 | 33 | 0,20 | |
| A0175/X03 | 16174,6 | 161,746 | 33 | 0,20 | |
| A0381/X02 | 16174,6 | 161,746 | 33 | 0,20 | |
| A0095/X09 | 3009,5 | 30,095 | 6 | 0,20 | |
| A0095/X09 | 3009,5 | 30,095 | 7 | 0,23 | |
| A0545/T03 | 2218 | 22,18 | 5 | 0,23 | |
| A0150/X13 | 1704,3 | 17,043 | 4 | 0,23 | |
| A0814/H12 | 7136 | 71,36 | 18 | 0,25 | VIGO |
| A0185/H09 | 3086,5 | 30,865 | 8 | 0,26 | |
| A0814/H12 | 7136 | 71,36 | 19 | 0,27 | |
| A0095/X09 | 3009,5 | 30,095 | 8 | 0,27 | |
| A0814/H12 | 7136 | 71,36 | 20 | 0,28 | |
| A0185/H09 | 3086,5 | 30,865 | 9 | 0,29 | |
| A0150/X13 | 1704,3 | 17,043 | 5 | 0,29 | |
| A0065/T05 | 2343,5 | 23,435 | 7 | 0,30 | |
| A0230/X07 | 8311,75 | 83,1175 | 27 | 0,32 | |
| A0567/H10 | 2176,5 | 21,765 | 7 | 0,32 | |
| A0570/T15 | 6052,3 | 60,523 | 20 | 0,33 | |
| A0570/T15 | 6052,3 | 60,523 | 21 | 0,35 | |
| A0465/H05 | 8972,17 | 89,7217 | 32 | 0,36 | |
| A0570/T15 | 6052,3 | 60,523 | 22 | 0,36 | |
| A0570/T15 | 6052,3 | 60,523 | 23 | 0,38 | |
| A0345/H04 | 5276,1 | 52,761 | 20 | 0,38 | |
| A0785/T11 | 4776,5 | 47,765 | 18 | 0,38 | |
| A0295/T06 | 1310,75 | 13,1075 | 5 | 0,38 | |
| A0570/T15 | 6052,3 | 60,523 | 24 | 0,40 | |
| A0570/T15 | 6052,3 | 60,523 | 25 | 0,41 | |
| A0690/H08 | 2456,67 | 24,5667 | 10 | 0,41 | |

UNIVERSIDADE DE VIGO 1990 REITORÍA

| ÁREA/DPTO. | H | C | F | Q | CAMPUS |
|---|---|---|---|---|---|
| A0555/T04 | 4322,29 | 43,2229 | 18 | 0,42 | |
| A0570/T15 | 6052,3 | 60,523 | 26 | 0,43 | |
| A0555/T04 | 4322,29 | 43,2229 | 19 | 0,44 | |
| A0265/C05 | 4080,3 | 40,803 | 18 | 0,44 | OURENSE* |
| A0265/C05 | 4080,3 | 40,803 | 19 | 0,47 | |
| A0225/X06 | 4925,5 | 49,255 | 30 | 0,61 | |
| A0560/T13 | 3370 | 33,7 | 21 | 0,62 | |
| A0560/T13 | 3370 | 33,7 | 22 | 0,65 | |
| A0560/T13 | 3370 | 33,7 | 23 | 0,68 | |
| A0566/C03 | 2666,1 | 26,661 | 18 | 0,68 | |
| A0560/T13 | 3370 | 33,7 | 24 | 0,71 | |
| A0560/T13 | 3370 | 33,7 | 25 | 0,74 | |
| A0800/T14 | 4472 | 44,72 | 36 | 0,81 | |
| A0800/T14 | 4472 | 44,72 | 37 | 0,83 | |
| A0765/C12 | 1267 | 12,67 | 11 | 0,87 | |
| A0640/C07 | 1030,17 | 10,3017 | 9 | 0,87 | |
| A0220/C04 | 1314 | 13,14 | 12 | 0,91 | |
| A0412/C02 | 548 | 5,48 | 5 | 0,91 | |
| A0640/C07 | 1030,17 | 10,3017 | 10 | 0,97 | |

*Para asignación da praza ao campus, utilizáronse os seguintes valores de Q, calculados segundo os criterios que aparecen na convocatoria: Campus Vigo: Q= 0.50; Campus Ourense: Q= 0.27

UNIVERSIDADE DE VIGO 1990 REITORÍA

# Bibliografía


[1] Aumann R, Maschler M (1985) Game theoretic analysis of a bankruptcy problem from the Talmud. Journal of Economic Theory 36, pp 195-213.

[2] Cano-Berlanga S, Giménez-Gómez JM, Vilella C (2015) Enjoying cooperative games: The R package GameTheory.

[3] Chen S (2015) Systematic favorability in claims problems with indivisibilities. Springer-Verlag Berlin Heidelberg.

[4] Chun Y (1988) The proportional solution for rights problem. Mathematical Social Sciences 15, pp 231-246.

[5] Chun Y (2006) Secured lower bound, composition up, and minimal rights first for bankruptcy problems. Journal of Mathematical Economics 44, pp 925-932.

[6] Curiel I, Maschler M, Tijs SH (1987) Bankruptcy Games. Zeitschrift für Operations Research 31, pp 143-159.

[7] Dagan N (1996) New characterizations of old bankruptcy rules. Social Choice and Welfare 13, pp 51-59

[8] Herrero C, Martínez R (2004) Egalitarian rules in claims problems with indivisible goods.

[9] Herrero C, Martínez R (2008a) Balanced allocation methods for claims problems with indivisibilities. Springer-Verlag.

[10] Herrero C, Martínez R (2008b) Up methods in the allocation of indivisibilities when preferences are single-peaked. Sociedad de Estadística e Investigación Operativa.

[11] Herrero C, Villar A (2001a) The three musketeers: four classicalsolutions to bankruptcy problems. Mathematical Social Sciences 42, pp 307-328.

[12] Herrero C, Villar A (2001a) Sustainability in bankruptcy problems. University of Alicante.

[13] Hwang Y (2015). Two characterizations of the randon arrival rule. Econometric Theory Bulletin 3(1), pp 43-52.

[14] Moreno-Ternero JD, Villar A (2004) The talmud rule and the securement of agents awards. Mathematical Social Sciences 47(2), pp 245-257.

[15] Moulin H (2000) Priority rules and other asymetric rationing methods. Econometrica, Vol.68, No.3, pp 643-684.

[16] Moulin H, Stong R (2002) Fair Queuing and Other Probabilistic Allocation Methods. Mathematics of Operations Research 27(1), pp 1-30.

[17] Thomson W (2003) Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey. Mathematical Social Sciences 45, pp 249-297.

[18] Yeh CH (2001) Sustainability, claims monotonicity, and the constrained equal award rule. Mimeo.

[19] Young P (1988) Distributive justice in taxation. Journal of Economic Theory 43, pp 321-335.

[20] Young P (1994) Equity: theory and practice. Princeton University Press.