eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
ZTF-FCT
Zientzia eta Teknologia Fakultatea
Facultad de Ciencia y Tecnología

Algoritmo de Gale-Shapley. Variaciones y alternativas

Trabajo Fin de Grado
Grado en Matemáticas

Juan Pablo Mínguez

Trabajo dirigido por
Luis Martínez
Carlos Gorria

Leioa, 1 de septiembre de 2014

Índice general

Introducción

En esta memoria se trata el problema de encontrar un algoritmo que construya un emparejamiento entre dos grupos, entendiendo por emparejamiento la asignacion a cada individuo, de cada grupo, otro individuo. La situación inicial de la que parte el problema es la siguiente:

- Dos grupos, los proponentes y los propuestos, que están formados por n individuos cada uno, siendo n la dimensión del problema.
    - El grupo de los proponentes es el encargado de hacer las propuestas a la hora de construir el emparejamiento.
    - El grupo de los propuestos es el encargado de recibir y gestionar las propuestas a la hora de construir el emparejamiento.
- Cada individuo de cada grupo ordena en una lista, de manera decreciente, a individuos del otro grupo atendiendo a su preferencia a la hora de ser emparejado, a esta lista la llamaremos lista de preferencia del individuo, considerando el quedarse solo la opción menos preferida de entre las aceptables.

El objetivo del problema es crear un emparejamiento en el que cada pareja sea satisfactoria para los individuos que la crean en base a las preferencias de cada uno.

Hace más de medio siglo, en 1962, Gale y Shapley [1] empezaron a estudiar matemáticamente el problema de emparejar dos grupos, aunque en Estados Unidos ya existía el National Resident Matching Program (NRMP). Este programa había sido implementado en 1951 para la asignación de residentes a los hospitales según sus preferencias.

Un emparejamiento entre dos grupos, los proponentes y los propuestos, se dice que es estable cuando, de acuerdo con las lista de preferencia de cada individuo de cada grupo sobre los individuos del otro grupo, no existan un proponente y un propuesto no emparejados que se prefieran entre sí antes que a su pareja asignada.

El problema de encontrar un emparejamiento con dichas caracteristicas, y lo solucionaron los citados autores con el algoritmo que en esta memoria llamamaremos Gale-Shapley I. Para la ejecución de este algoritmo se parte de dos grupos, uno de n proponentes y otro de n propuestos, donde cada individuo de cada grupo tiene unas preferencias sobre los individuos del otro grupo. La descripción del algoritmo es la siguiente:

- Cada proponente, mientras esté libre, sin ningún orden preestablecido entre ellos, va proponiendo a los propuestos siguiendo en orden decreciente su lista de preferencias.
- Cada propuesto, si está libre, acepta la propuesta, y, si no lo está compara a los dos proponentes y acepta el preferido según su lista de preferencias.
- Este proceso se continúa hasta que no queden proponentes sin emparejar.

.

Este algoritmo y sus variaciones son usados hoy en día en muchas áreas, como por ejemplo en la asignación de estudiantes de medicina a hospitales, alumnos a escuelas, donantes de órganos a receptores, etc.

Esta última aplicación, fue desarrollada por Alvin Roth (Catedrático de economía en Hardvard).

En todo el mundo la demanda de órganos es mucho mayor que la donación, por ello es importante la optimización de la viabilidad de los trasplantes. Las donaciones provienen de cadáveres o de personas vivas. El problema que puede darse en la donación por parte de personas vivas es que el donante lo sea sólo para un receptor predeterminado, por ejemplo un padre P_1 dispuesto a donar un órgano a su hijo h_1 y otro padre P_2 dispuesto a donar a su hijo h_2. Entonces dos personas están dispuestas a donar un órgano a su hijo y no están dispuestas a donarlo a otra persona, pero podría darse el caso de que el órgano de P_1 sea incompatible con h_1, o que según criterios médicos a h_1 le convenga más el órgano de P_2 y a h_2 el de P_1. Entonces cada padre estaría dispuesto a donar su órgano a una persona que no sea su hijo siempre que su hijo reciba un órgano.

El problema se complica cuando el número de parejas donante-receptor es grande ya que si P_i dona su órgano a h_j entonces P_j no tiene que ser necesariamente el donante de h_i. Las preferencias en este caso podrían ser ciertos criterios médicos en los que se basa la viabilidad del trasplante.

Otra aplicación más reciente y de mucha transcendencia, en el área de la economía, es la desarrollada por Roth y Shapley, sobre la teoría de asignaciones estables y la práctica en el diseño de mercados, que les valió el premio Nobel de Economía en el año 2012.

En el primer capítulo de esta memoria se formaliza lo expuesto tanto por Gale y Shapley como por Harris, Hirst y Mossinghoff [2]. En la primera sección se contempla el problema con listas de preferencias estrictas y completas, o sea que cada individuo de cada grupo está en la lista de preferencias de todos los individuos del otro grupo y además estas preferencias no tienen indiferencias. En una segunda sección, se trata el problema cuando algún individuo de algún grupo tiene como inaceptable a algún individuo del otro grupo y las preferencias de los aceptables son estrictas. Y en una tercera sección se estudia el caso de que las listas de preferencias son completas pero pueden tener indiferencias.

En el segundo capítulo se dan dos alternativas al agoritmo de Gale-Shapley cuando las listas de preferencia son completas y estrictas. Estas alternativas tratan de primar el interés colectivo frente a los intereses individuales de los que parte el algoritmo de Gale-Shapley, además de buscar un emparejamiento más equilibrado, ya que el emparejamiento del algoritmo de Gale-Shapley es óptimo para los proponentes y pésimo para los propuestos. Las dos alternativas que se van a estudiar son las propuestas por Fuku, Namatame y Kaizouji [3] en las que se renuncia al concepto de la estabilidad del emparejamiento.

En el tercer cápitulo se cuantifica la calidad de los emparejamientos definiendo y midiendo la satisfacción de los individuos, la de cada grupo y la del conjunto de los dos grupos en los emparejamientos deducidos, mediante los algoritmos vistos, en los cápitulos 1 y 2, y se comparan mediante un sofware desarrollado en este trabajo (Apéndice B).

Notación

- $b \overset{x}{\succ} a$: x prefiere a b antes que a a.
- $b \overset{x}{\prec} a$: x prefiere a a antes que a b.
- $b \overset{x}{=} a$: x tiene indiferencia entre a y b.
- $b \overset{x}{\succsim} a$: x prefiere a b antes que a a o tiene indiferencia entre ellos.
- $b \overset{x}{\precsim} a$: x prefiere a a antes que a b o tiene indiferencia entre ellos.
- $\overset{x}{\succ}: a_1\ a_2 \dots a_n$: $a_i \overset{x}{\succ} a_{i+1} \quad \forall i \in 1, 2, \dots, n-1$
- $x : a_1 \succ a_2 \succ \cdots \succ a_i = a_{i+1} \succ \cdots \succ a_n$: $a_j \overset{x}{\succ} a_{j+1} \quad \forall j \neq i \in 1, 2, \dots, n-1$ y $a_i \overset{x}{=} a_{i+1}$
- $\mu_i(h) = w$: tras el i-ésimo paso del algoritmo h está emparejado con w (no siendo necesariamente pareja definitiva).
- $\mu(h) = w$: la pareja definitiva de h es w.
- $\mu_i(h) = h$: tras el i-ésimo paso del algoritmo h está sin pareja.
- $\mu(h) = h$: h queda sin emparejar.
- $\mathrm{M}_i \overset{h,w}{\succ} \mathrm{M}_j$: $\mathrm{M}_i(h) \overset{h}{\succ} \mathrm{M}_j(h)$ y $\mathrm{M}_i(w) \overset{w}{\succ} \mathrm{M}_j(w)$, siendo M_i, M_j emparejamientos
- p_x^a : posición que ocupa a en la lista de preferencia de x (siendo $p_x^a = 1$ cuando a es el más deseado por x).
- Nota: Por regla general denotaremos como μ al emparejamiento generado por los algoritmos Gale-Shapley y por M cualquier otro emparejamiento.

Capítulo 1

Algoritmo de Gale-Shapley y sus variantes

En este cápitulo tratamos de formalizar tanto lo que publicaron Gale y Shapley [1] como lo que escribieron Harris, Hirst y Mossinghoff [2].

Estudiaremos el algoritmo introducido por Gale y Shapley como método de construcción de un emparejamiento estable, algunas de sus propiedades y algunas de sus variantes.

1.1. Algoritmo de Gale-Shapley

En esta sección presentamos un algoritmo que llamamos Gale-Shapley I el cual proporciona un método para dar un emparejamiento estable (definido a continuación) cuando el número de proponentes y el número de propuestos es el mismo y además las listas de preferencia de cada individuo son completas y estrictas, o sea que cada individuo de cada grupo está en la lista de preferencias de todos los individuos del otro grupo y además estas preferencias no tienen indiferencias.

A continuación se definen unos conceptos previos.

Definición 1. Sean H y W los conjuntos de los proponentes y de los propuestos, respectivamente. Definimos un *emparejamiento* como una aplicación $\mathrm{M} : H \cup W \to H \cup W$ que verifica las siguientes condiciones:

- M es biyectiva.

- Si $h \in H$ entonces $\mathrm{M}(h) \in \{h\} \cup W$.
- Si $w \in W$ entonces $\mathrm{M}(w) \in \{w\} \cup H$.
- $\mathrm{M}(h) = w$ si y sólo si $\mathrm{M}(w) = h$.

Definición 2. Sean H, W los conjuntos de los proponentes y de los propuestos, respectivamente. Se dice que el par h, w, con $h \in H$ y $w \in W$, es *bloqueante* del emparejamiento M si se verifican las dos condiciones siguientes:

- $\mathrm{M}(h) \neq w$.
- $w \overset{h}{\succ} \mathrm{M}(h)$ y $h \overset{w}{\succ} \mathrm{M}(w)$.

Definición 3. Decimos que un emparejamiento M es *estable* si no existe en él un par bloqueante.

Ejemplo 1. Sean el conjunto de los proponentes $H = \{a, b\}$ y el de los propuestos $W = \{y, z\}$ y una relación de preferencias:

$$\begin{cases} \overset{a}{\succ}: y\ z \\ \overset{b}{\succ}: z\ y \end{cases} \quad \text{y} \quad \begin{cases} \overset{y}{\succ}: a\ b \\ \overset{z}{\succ}: a\ b \end{cases}$$

El emparejamiento $(a, z), (b, y)$ no es estable ya que el par a, y es un par bloqueante ($y \overset{a}{\succ} \mu(a) = z \quad$ y $\quad a \overset{y}{\succ} \mu(y) = b$).

En 1962 Gale y Shapley [1] probaron que siempre existe al menos un emparejamiento estable entre n proponentes y n propuestos con listas de preferencias completas y estrictas, y describieron un algoritmo que denominaremos Gale-Shapley I para encontrar uno de estos emparejamientos estables.

Este algoritmo necesita como entrada un conjunto de n proponentes y unas preferencias de los propuestos por cada proponente además de un conjunto de n propuestos y unas preferencias de los proponentes por cada propuesto. A continuación definimos el algoritmo:

```
Inicialmente todos los proponentes estan sin emparejar

MIENTRAS (exista un proponente sin emparejar)
   -propone a los que no haya propuesto segun su orden de preferencia.
```

```
   SI (el propuesto esta libre)
      -acepta la propuesta.
   SI NO

      SI (el propuesto prefiere a su actual pareja)
         -rechaza la propuesta.
      SI NO
         -acepta la propuesta.
      FIN SI

   FIN SI

FIN MIENTRAS
```

Nos referiremos a este algoritmo como Gale-Shapley I

Observación 1. Siendo W el conjunto de los propuestos:
$\forall w \in W$ y $\forall i < j$ se tiene que $\mu_i(w) \overset{w}{\precsim} \mu_j(w) \overset{w}{\precsim} \mu(w)$ teniendo en cuenta que $w \overset{w}{\prec} h \;\; \forall h \in H$.
Además, si existe i tal que $\mu_i(w) = w$ con $w \in W$, entonces $\forall j < i$ se tiene que $\mu_j(w) = w$.

Ejemplo 2. Sean el conjunto de los proponentes $H = \{a, b, c, d\}$ y el de los propuestos $W = \{w, x, y, z\}$ y sus respectivas listas de preferencias

$$\begin{cases} \overset{a}{\succ}: y\ x\ z\ w \\ \overset{b}{\succ}: z\ w\ x\ y \\ \overset{c}{\succ}: z\ w\ x\ y \\ \overset{d}{\succ}: z\ y\ w\ x \end{cases} \quad \text{y} \quad \begin{cases} \overset{w}{\succ}: d\ c\ b\ a \\ \overset{x}{\succ}: a\ c\ d\ b \\ \overset{y}{\succ}: b\ c\ d\ a \\ \overset{z}{\succ}: c\ a\ b\ d \end{cases}$$

El algoritmo tendría la siguiente ejecución:

Paso 1 a propone a y,
y acepta la propuesta ya que está libre,
con lo que $\mu_1(a) = y$ y $\mu_1(y) = a$
además de
$\mu_1(b) = b, \mu_1(c) = c, \mu_1(d) = d, \mu_1(w) = w, \mu_1(x) = x, \mu_1(z) = z.$

Paso 2 b propone a z,
z acepta la propuesta ya que está libre,
con lo que $\mu_2(b) = z$ y $\mu_2(z) = b$
además de
$\mu_2(a) = y$, $\mu_2(c) = c$, $\mu_2(d) = d$, $\mu_2(w) = w$, $\mu_2(x) = x$, $\mu_2(y) = a$.

Paso 3 c propone a z,
z acepta la propuesta ya que $c \overset{z}{\succ} b = \mu_2(z)$,
con lo que $\mu_3(c) = z$ y $\mu_3(z) = c$
además de
$\mu_3(a) = y$, $\mu_3(b) = b$, $\mu_3(d) = d$, $\mu_3(w) = w$, $\mu_3(x) = x$, $\mu_3(y) = a$.

Paso 4 d propone a z,
z rechaza la propuesta ya que $\mu_3(z) = c \overset{z}{\succ} d$,
d propone a y,
y acepta la propuesta ya que $d \overset{y}{\succ} a = \mu_3(y)$,
con lo que $\mu_4(d) = y$ y $\mu_4(y) = d$
además de
$\mu_4(a) = a$, $\mu_4(b) = b$, $\mu_4(c) = z$, $\mu_4(w) = w$, $\mu_4(x) = x$, $\mu_4(z) = c$.

Paso 5 a propone a x,
x acepta la propuesta ya que está libre,
con lo que $\mu_5(a) = x$ y $\mu_5(x) = a$
además de
$\mu_5(b) = b$, $\mu_5(c) = z$, $\mu_5(d) = y$, $\mu_5(w) = w$, $\mu_5(y) = d$, $\mu_5(z) = c$.

Paso 6 b propone a w,
w acepta la propuesta ya que está libre,
con lo que $\mu_6(b) = w$ y $\mu_6(w) = b$
además de
$\mu_6(a) = x$, $\mu_6(c) = z$, $\mu_6(d) = y$, $\mu_6(x) = a$, $\mu_6(y) = d$, $\mu_6(z) = c$.

Así pues el emparejamiento que nos da el algoritmo es $\{(a, x), (b, w), (c, z), (d, y)\}$

Veamos que verifica lo dicho en la Observación 1

- $w = \mu_1(w) = \mu_2(w) = \mu_3(w) = \mu_4(w) = \mu_5(w) \overset{w}{\prec} b = \mu_6(w) = \mu(w)$.
- $x = \mu_1(x) = \mu_2(x) = \mu_3(x) = \mu_4(x) \overset{x}{\prec} a = \mu_5(x) = \mu_6(x) = \mu(x)$.
- $a = \mu_1(y) = \mu_2(y) = \mu_3(y) \overset{y}{\prec} d = \mu_4(y) = \mu_5(y) = \mu_6(y) = \mu(y)$.

- $z = \mu_1(z) \overset{z}{\prec} b = \mu_2(z) \overset{z}{\prec} c = \mu_3(z) = \mu_4(z) = \mu_5(z) = \mu_6(z) = \mu(z)$.

Teorema 1.1.1. *El algoritmo Gale-Shapley I descrito anteriormente termina en un número finito de pasos.*

Demostración. Si tenemos n proponentes y n propuestos, como cada proponente a lo sumo propone a n propuestos diferentes, el algoritmo terminará a lo sumo en n^2 pasos.

□

Definición 4. Se dice que un emparejamiento M es *perfecto* si se verifican las dos condiciones siguientes:

- $\mathrm{M}(h) \in W \quad \forall h \in H$.
- $\mathrm{M}(w) \in H \quad \forall w \in W$.

Teorema 1.1.2. *El algoritmo Gale-Shapley I produce un emparejamiento perfecto.*

Demostración. Supongamos por reducción al absurdo que existe h tal que $\mu(h) = h$ con lo que existirá w tal que $\mu(w) = w$ lo que es absurdo, ya que por la construcción del emparejamiento existe i tal que h propone a w en el paso i-ésimo y w habría aceptado la propuesta de h ya que por la Observación 1 si $\mu(w) = w$ entonces $\mu_j(w) = w \quad \forall j$, en particular $\mu_i(w) = w$.

Por lo que el emparejamiento es perfecto. □

Teorema 1.1.3. *El algoritmo Gale-Shapley I produce un emparejamiento estable.*

Demostración. Si μ es el emparejamiento generado por el algoritmo Gale-Shapley I, supongamos por reducción al absurdo que μ no es estable con lo que existe un proponente $h \in H$ y un propuesto $w \in W$ tal que h, w es un par bloqueante de μ con lo que $\mu(h) \neq w$ y $w \overset{h}{\succ} \mu(h)$, $h \overset{w}{\succ} \mu(w)$.

Dado que los proponentes realizan sus proposiciones en orden decreciente según su lista de preferencias y como $w \overset{h}{\succ} \mu(h)$, tenemos que existe i tal que en la etapa i-ésima h ha propuesto a w, y dado que

$\mu(h) \neq w$ existe $j \geq i$ tal que en la etapa j-ésima w ha rechazado a h en favor de h' con $h' \in H - \{h\}$ por lo que $h' \overset{w}{\succ} h$ y $\mu_j(w) = h'$.

Por la Observación 1 $\mu(w) \overset{w}{\succsim} \mu_j(w) = h'$ lo que es absurdo ya que tenemos que $\mu(w) \overset{w}{\succsim} \mu_j(w) = h' \overset{w}{\succ} h$ habiendo supuesto que $h \overset{w}{\succ} \mu(w)$.

Por lo que el emparejamiento generado por el algoritmo es estable.

□

Definición 5. Decimos que el emparejamiento M es *óptimo* para el conjunto A si cumple las dos condiciones siguientes:

- M es estable.
- $\forall a \in A$ tenemos que $\mathrm{M}(a) \overset{a}{\succsim} \mathrm{M}'(a) \quad \forall \mathrm{M}'$ emparejamiento estable.

Definición 6. Decimos que el emparejamiento M es *pésimo* para el conjunto A si cumple las dos condiciones siguientes:

- M es estable.
- $\forall a \in A$ tenemos que $\mathrm{M}(a) \overset{a}{\precsim} \mathrm{M}'(a) \quad \forall \mathrm{M}'$ emparejamiento estable.

Definición 7. Diremos que (h, w) es una *pareja válida* si existe M emparejamiento estable tal que $\mathrm{M}(h) = w$.

Teorema 1.1.4. *En el proceso definido por Gale-Shapley I no se rechaza ninguna pareja válida.*

Demostración. Supongamos por reducción al absurdo que el primer rechazo de una pareja válida es el de w a h en favor de h' en la etapa i-ésima, por lo que $h' \overset{w}{\succ} h$ y por la definición de pareja válida existe M emparejamiento estable tal que $\mathrm{M}(h) = w$.

Sea w' tal que $\mathrm{M}(h') = w'$. Dado que el primer rechazo de una pareja válida es el de w a h en favor de h', tenemos que $w \overset{h'}{\succ} w'$, ya que si no existiría $j < i$ tal que en la etapa j-ésima w' ha rechazado a h' lo que no puede ser ya que el primer rechazo de una pareja válida es de w a h en favor de h' en la etapa i-ésima . Esto entra en contradicción con que M sea estable ya que tenemos que $h' \overset{w}{\succ} \mathrm{M}(w) = h$ y $w \overset{h'}{\succ} \mathrm{M}(h') = w'$, por lo que el par h', w es un par bloqueante de M.

□

Teorema 1.1.5. *El algoritmo Gale-Shapley I nos proporciona un emparejamiento óptimo para los proponentes y pésimo para los propuestos*

Demostración. Sean H y W los conjuntos de los proponentes y los propuestos respectivamente y sea μ el emparejamiento resultante de aplicar el algoritmo Gale-Shapley I. Supongamos por reducción al absurdo que existe M emparejamiento estable y existe $h \in H$ tal que $M(h) \overset{h}{\succ} \mu(h)$ entonces en la construcción del emparejamiento μ, mediante el algoritmo Gale-Shapley I, h rechaza a $\mathrm{M}(h)$ en algun momento ya que h propone en orden decreciente según su lista de preferencia, lo que entra en contradicción con el Teorema 1.1.4 ya que $h, \mathrm{M}(h)$ es una pareja valida.

Veamos ahora que es pésimo para los propuestos. Sea $w \in W$ tal que $\mu(w) = h$ y sea M' cualquier emparejamiento estable tal que $\mathrm{M}'(w) = h'$ y $\mathrm{M}'(h) = w'$.
Supongamos por reducción al absurdo que $\mu(w) \overset{w}{\succ} \mathrm{M}'(w)$. Entonces $h \overset{w}{\succ} h'$ y, como μ es óptimo para los proponentes, $\mu(h) \overset{h}{\succ} \mathrm{M}'(h)$ con lo que $w \overset{h}{\succ} w'$ y tendriamos que el par h, w sería bloqueante de M' ya que $h \overset{w}{\succ} \mathrm{M}'(w) = h'$ y $w \overset{h}{\succ} \mathrm{M}'(h) = w'$, lo que es absurdo ya que hemos supuesto que M' es estable.

Por lo que el emparejamiento generado por el algoritmo es óptimo para los proponentes y pésimo para los propuestos. □

Teorema 1.1.6. *El emparejamiento generado por el algoritmo Gale-Shapley I es independiente del orden de las proposiciones.*

Demostración. Sean μ y μ' dos emparejamientos generados por la ejecución del algoritmo en distinto orden, y sea el conjunto H el de los proponentes. Por el Teorema 1.1.5 (μ y μ' son óptimos para los proponentes) tenemos que:

- $\forall h \in H \quad \mu(h) \overset{h}{\succsim} \mathrm{M}(h) \quad \forall \mathrm{M}$ emparejamiento estable
  en particular $\mu(h) \overset{h}{\succsim} \mu'(h)\ \forall h \in H$.
- $\forall h \in H \quad \mu'(h) \overset{h}{\succsim} \mathrm{M}(h) \quad \forall \mathrm{M}$ emparejamiento estable
  en particular $\mu'(h) \overset{h}{\succsim} \mu(h)\ \forall h \in H$.

Por lo que $\mu(h) = \mu'(h) \quad \forall h \in H$ □

Teorema 1.1.7. *El emparejamiento obtenido mediante el algoritmo Gale-Shapley I no permite que más de un proponente quede emparejado con su última opción*

Demostración. Supongamos por reducción al absurdo que existen dos proponentes m_i y m_j que están emparejados con su última opción, es decir, $\mu(m_i) = w_i$ con $w_k \overset{m_i}{\succ} w_i$, $\forall w_k \in W - \{w_i\}$ y $\mu(m_j) = w_j$ con $w_k \overset{m_j}{\succ} w_j$, $\forall w_k \in W - \{w_j\}$.

Sin pérdida de generalidad supongamos que el emparejamiento definitivo de m_i ha sido previo al de m_j, es decir, $\mu(m_i) = \mu_{N_i}(m_i)$ y $\mu(m_j) = \mu_{N_j}(m_j)$ donde $N_i < N_j$.

Dado que en la etapa N_i el proponente m_i se ha emparejado con su última opción, $\mu(m_i) = \mu_{N_i}(m_i) = w_i$, después de proponer en etapas previas de manera descendente según la relación de orden $\overset{m_i}{\succ}$ a todos los elementos anteriores a w_i del conjunto de propuestos $w_k \in W - \{w_i\}$. Esto significa que todos ellos ya están emparejados en dicho instante, es decir, que $\mu(w_k) \neq w_k \, \forall w_k \in W - \{w_i\}$. Consecuentemente en la etapa N_i todos los proponentes quedan también emparejados $\mu(m_k) \neq m_k$, $\forall m_k \in H$ y el algoritmo finaliza automáticamente. Se deduce que m_j debiera estar emparejado anteriormente, $N_j < N_i$, lo cual lleva a una contradicción

□

1.2. Inaceptables

En la primera sección se trató el problema con listas de preferencia completas y estrictas, en esta segunda sección se estudia el caso en que las listas de preferencia sean estrictas pero no completas, o sea que cada individuo puede tener inaceptables en el otro grupo a los que omite en su lista de preferencias.

Definición 8. Sean H, W los conjuntos de los proponentes y de los propuestos, respectivamente. Se dice que el par h, w, con $h \in H$ y $w \in W$, es *bloqueante* del emparejamiento M si se verifica alguna de las dos condiciones siguientes:

- $\mathrm{M}(h) = w$ y h inaceptable para w ó w inaceptable para h.
- $\mathrm{M}(h) \neq w$ y además $h \overset{w}{\succ} \mathrm{M}(w)$ y $w \overset{h}{\succ} \mathrm{M}(h)$.

Definición 9. Decimos que un emparejamiento M es *estable* si no existe en él un par bloqueante.

Observación 2. Dados los conjuntos, de proponentes y de los propuestos H y W respectivamente, y una relación de preferencias, no siempre existe un emparejamiento perfecto tal y como lo definimos en el apartado anterior (Definición 4) (como mostramos en el siguiente ejemplo).

Ejemplo 3. Sea $H = \{a, b\}$ y $W = \{x, y\}$ y sus relaciones de preferencia $\begin{cases} \overset{a}{\succ}: x\ y \\ \overset{b}{\succ}: x\ y \end{cases}$ y $\begin{cases} \overset{x}{\succ}: a\ b \\ \overset{y}{\succ}: a \end{cases}$.

Hay dos posibles emparejamientos perfectos que son $\mathrm{M}_1 = \{(a, x), (b, y)\}$ y $\mathrm{M}_2 = \{(a, y), (b, x)\}$.

- M_1 no es estable ya que b es inaceptable para y.
- M_2 no es estable ya que el par a, x bloquea el emparejamiento M_2 ($x \overset{a}{\succ} \mathrm{M}_2(a) = y$ y $a \overset{x}{\succ} \mathrm{M}_2(x) = b$).

Teorema 1.2.1. *Sean H, W los conjuntos de los proponentes y los propuestos respectivamente, $h \in H$, $w \in W$ y M_1, M_2 emparejamientos estables tales que $M_1(h) = w$ y $M_2(h) \neq w$. Entonces no puede darse el caso que $M_1 \overset{h,w}{\succ} M_2$.*

Demostración. Sean h_1, w_1 tal que $\mathrm{M}_2(h) = w_1$, $\mathrm{M}_2(w) = h_1$. Supongamos por reducción al absurdo que $\mathrm{M}_1 \overset{h,w}{\succ} \mathrm{M}_2$, entonces tendríamos que $w \overset{h}{\succ} \mathrm{M}_2(h) = w_1$ y $h \overset{w}{\succ} \mathrm{M}_2(w) = h_1$ con lo que el par h, w sería un par bloqueante de M_2 lo que es absurdo ya que M_2 es estable. □

Teorema 1.2.2. *Sean H, W los conjuntos de los proponentes y los propuestos respectivamente, $h \in H$, $w \in W$ y M_1, M_2 emparejamientos estables tales que $M_1(h) = w$ y $M_2(h) \neq w$. Entonces no puede darse el caso que $M_2 \overset{h,w}{\succ} M_1$.*

Demostración. Sean h_1, w_1 tal que $\mathrm{M}_2(w) = h_1$ y $\mathrm{M}_2(h) = w_1$. Supongamos por reducción al absurdo que $\mathrm{M}_2 \overset{h,w}{\succ} \mathrm{M}_1$, entonces tendríamos que:

- Como $\mathrm{M}_2(h) = w_1$ y $\mathrm{M}_2(w) = h_1$ y además hemos supuesto que $\mathrm{M}_2 \overset{h,w}{\succ} \mathrm{M}_1$ tenemos que por el Teorema 1.2.1 (con M_2 y M_1 como emparejamientos estables y con el par h, w_1 primero, y luego con el par h_1, w) que $\mathrm{M}_1 \overset{h_1,w_1}{\succ} \mathrm{M}_2$, además $\mathrm{M}_1(h_1) \neq h_1$, $\mathrm{M}_1(w_1) \neq w_1$ ya que estar solo es la opción menos preferida y además $\mathrm{M}_1(h_1) \neq w_1$ por el Teorema 1.2.1.

- Sean h_2, w_2 tal que $\mathrm{M}_1(h_1) = w_2$ y $\mathrm{M}_1(w_1) = h_2$. Entonces $\mathrm{M}_2 \overset{h_2,w_2}{\succ} \mathrm{M}_1$, además $\mathrm{M}_2(h_2) \neq h_2$, $\mathrm{M}_2(w_2) \neq w_2$ y además $\mathrm{M}_2(h_2) \neq w_2$.

- Sean h_3, w_3 tal que $\mathrm{M}_2(h_2) = w_3$ y $\mathrm{M}_2(w_2) = h_3$. Entonces $\mathrm{M}_1 \overset{h_3,w_3}{\succ} \mathrm{M}_2$, además $\mathrm{M}_1(h_3) \neq h_3$, $\mathrm{M}_1(w_3) \neq w_3$ y además $\mathrm{M}_1(h_3) \neq w_3$.

  $\vdots$

y así tendríamos un proceso infinito, lo que es absurdo ya que el número de proponentes y propuestos es finito.

□

Observación 3. En estos dos últimos teoremas hemos visto que siendo H los proponentes y W los propuestos, si tenemos $h \in H$, $w \in W$ y $\mathrm{M}_1, \mathrm{M}_2$ emparejamientos estables tal que $\mathrm{M}_1(h) = w$ y $\mathrm{M}_2(h) \neq w$, se tiene que $\mathrm{M}_i(h) \overset{h}{\succ} \mathrm{M}_j(h)$ y $\mathrm{M}_j(w) \overset{w}{\succ} \mathrm{M}_i(w)$ donde o bien $i = 1, j = 2$ ó $i = 2, j = 1$.

Veamos ahora que para todos los emparejamientos estables el subconjunto de proponentes y propuestos que se quedan sin emparejar es el mismo.

Teorema 1.2.3. *Sean los conjuntos H y W de proponentes y propuestos respectivamente y sus respectivas relaciones de preferencias con posibles inaceptados. Entonces si existe $a \in H \cup W$ y existe M emparejamiento estable tal que $M(a) = a$ entonces $M'(a) = a$ $\forall M'$ emparejamiento estable.*

Demostración. Sean M_1 y M_2 emparejamientos estables.

Supongamos por reducción al absurdo que $\mathrm{M}_1(a) = b$ y $\mathrm{M}_2(a) = a$, entonces tendríamos que:

- $M_1(a) \overset{a}{\succ} M_2(a)$ y por la Observación 3 $M_2(b) \overset{b}{\succ} M_1(b)$.
- Sea a_1 tal que $M_2(b) = a_1$. Entonces $M_1(a_1) \overset{a_1}{\succ} M_2(a_1)$.
- Sea b_1 tal que $M_1(a_1) = b_1$. Entonces $M_2(b_1) \overset{b_1}{\succ} M_1(b_1)$.
- Sea a_2 tal que $M_2(b_1) = a_2$. Entonces $M_1(a_2) \overset{a_2}{\succ} M_2(a_2)$.
  $\vdots$
- Por la manera en que mostramos las parejas y como el número de proponentes y propuestos es finito habrá un punto en el que se de alguna de las siguientes dos situaciones:
  - Sea a_i tal que $M_1(a_i) = a_i$ y $M_2(a_i) = b_{i-1}$ y tendríamos que $M_2 \overset{a_i,b_{i-1}}{\succ} M_1$, lo que es absurdo por el Teorema 1.2.1.
  - Sea b_i tal que $M_2(b_i) = b_i$ y $M_1(b_i) = a_i$ y tendríamos que $M_1 \overset{a_i,b_i}{\succ} M_2$, lo que es absurdo por el Teorema 1.2.1.

$\square$

Nos queda ver que con estas condiciones en las listas de preferencias siempre es posible encontrar un emparejamiento estable, para ello vamos a dar una modificacón del algoritmo Gale-Shapley I y demostraremos que el emparejamiento resultante es estable.

El algoritmo viene dado por:

```
Inicialmente todos los proponentes estan sin emparejar

MIENTRAS (exista un proponente sin emparejar y
          tenga propuestos aceptables que no haya propuesto)

   -propone a los que no haya propuesto y sean aceptables segun
   su orden de preferencia.

   SI (el propuesto esta libre)

      SI (el proponente es aceptable para el propuesto)
```

```
      -acepta la propuesta.
   SI NO
      -rechaza la propuesta.
SI NO

   SI (el propuesto prefiere a su actual pareja
       o el proponente es inaceptable)
      -rechaza la propuesta.
   SI NO
      -acepta la propuesta.
   FIN SI

FIN SI

FIN MIENTRAS
```

Nos referiremos a este algoritmo como Gale-Shapley II

Observación 4. Siendo W el conjunto de los propuestos:
$\forall w \in W$ y $\forall i < j$ se tiene que $\mu_i(w) \overset{w}{\precsim} \mu_j(w) \overset{w}{\precsim} \mu(w)$ teniendo en cuenta que $w \overset{w}{\prec} h \ \ \forall h$ aceptable para w.
Además, si existe i tal que $\mu_i(w) = w$ con $w \in W$, entonces $\forall j < i$ $\mu_j(w) = w$.

Ejemplo 4. Sean el conjunto de proponentes $H = \{a, b, c, d\}$ y el de propuestos $W = \{w, x, y, z\}$ y sus respectivas listas de preferencias

$$\begin{cases} \overset{a}{\succ}: x \ y \\ \overset{b}{\succ}: w \ x \\ \overset{c}{\succ}: z \ w \\ \overset{d}{\succ}: x \ z \end{cases} \quad \text{y} \quad \begin{cases} \overset{w}{\succ}: a \ b \\ \overset{x}{\succ}: a \ d \ c \\ \overset{y}{\succ}: b \ d \\ \overset{z}{\succ}: a \ c \end{cases}$$

El algoritmo tendría la siguiente ejecución:

Paso 1 a propone a x,
x acepta la propuesta ya que está libre y a es aceptable para x, con lo que $\mu_1(a) = x$ y $\mu_1(x) = a$
además de
$\mu_1(b) = b$, $\mu_1(c) = c$, $\mu_1(d) = d$, $\mu_1(w) = w$, $\mu_1(y) = y$, $\mu_1(z) = z$.

Paso 2 b propone a w,
w acepta la propuesta ya que está libre y b es aceptable para w,
con lo que $\mu_2(b) = w$ y $\mu_2(w) = b$
además de
$\mu_2(a) = x$, $\mu_2(c) = c$, $\mu_2(d) = d$, $\mu_2(x) = a$, $\mu_2(y) = y$, $\mu_2(z) = z$.

Paso 3 c propone a z,
z acepta la propuesta ya que está libre y c es aceptable para z,
con lo que $\mu_3(c) = z$ y $\mu_3(z) = c$
además de
$\mu_3(a) = x$, $\mu_3(b) = w$, $\mu_3(d) = d$, $\mu_3(w) = b$, $\mu_3(x) = a$, $\mu_3(y) = y$.

Paso 4 d propone a x,
x rechaza la propuesta ya que $\mu_3(x) = a \overset{x}{\succ} d$,
d propone a z,
z rechaza la propuesta ya que z tiene a d como inaceptable.

Con lo que el emparejamiento que nos da el algoritmo es $(a, x), (b, w), (c, z), d, y$

Veamos que verifica la Observación 4:

- $w = \mu_1(w) \overset{w}{\prec} b = \mu_2(w) = \mu_3(w) = \mu_4(w)$
- $a = \mu_1(x) = \mu_2(x) = \mu_3(x) = \mu_4(x)$
- $y = \mu_1(y) = \mu_2(y) = \mu_3(y) = \mu_4(y)$
- $z = \mu_1(z) = \mu_2(z) \overset{z}{\prec} c = \mu_3(z) = \mu_4(z)$

Teorema 1.2.4. *El emparejamiento resultante del algoritmo Gale-Shapley II es estable*

Demostración. Si μ es el emparejamiento generado por el algoritmo Gale-Shapley II, supongamos por reducción al absurdo que μ no es estable con lo que existe un proponente $h \in H$ y un propuesto $w \in W$ tal que h, w es un par bloqueante de μ. Entonces se daría alguno de los tres casos siguientes:

- $\mu(h) = w$ con w inaceptable para h. Es absurdo ya que h no lo habría propuesto.

- $\mu(h) = w$ con h inaceptable para w. Es absurdo ya que w no habría aceptado la propuesta.

- $\mu(h) \neq w$ y $w \overset{h}{\succ} \mu(h)$, $h \overset{w}{\succ} \mu(w)$. Dado que los proponentes realizan sus proposiciones en orden decreciente según su lista de preferencias y como $w \overset{h}{\succ} \mu(h)$, tenemos que existe i tal que en la etapa i-ésima h ha propuesto a w, y dado que $\mu(h) \neq w$ existe $j \geq i$ tal que en la etapa j-ésima w ha rechazado a h en favor de h' con $h' \in H - \{h\}$ por lo que $h' \overset{w}{\succ} h$ y $\mu_j(w) = h'$. Por la Observación 4 $\mu(w) \overset{w}{\succsim} \mu_j(w) = h'$ lo que es absurdo ya que tenemos que $\mu(w) \overset{w}{\succsim} \mu_j(w) = h' \overset{w}{\succ} h$ habiendo supuesto que $h \overset{w}{\succ} \mu(w)$.

En consecuencia el emparejamiento generado por el algoritmo Gale-Shapley II es estable.

□

1.3. Indiferencias

Hasta ahora hemos vistos dos casos:

- Listas de preferencia completas y estrictas.

- Listas de preferencia incompletas y estrictas.

En esta sección vamos a tratar la variante en la cual las listas de preferencia son completas pero puede haber indiferencia entre ciertos individuos.

En esta nueva variante podemos definir varios tipos de emparejamientos estables.

Definición 10. Un emparejamiento M es *débilmente estable* si no existe $h \in H$ y $w \in W$ tal que:

- $\mathrm{M}(h) \neq w$

- $w \overset{h}{\succ} \mathrm{M}(h)$ *y* $h \overset{w}{\succ} \mathrm{M}(w)$

Definición 11. Un emparejamiento M es *fuertemente estable* si no existe $h \in H$ y $w \in W$ tal que:

- $\mathrm{M}(h) \neq w$
- y se da una de las dos siguientes condiciones:
  - $w \overset{h}{\succsim} \mathrm{M}(h)$ y $h \overset{w}{\succ} \mathrm{M}(w)$
  - $w \overset{h}{\succ} \mathrm{M}(h)$ y $h \overset{w}{\succsim} \mathrm{M}(w)$

Definición 12. Un emparejamiento M es *super-estable* si no existe $h \in H$ y $w \in W$ tal que:

- $\mathrm{M}(h) \neq w$
- $w \overset{h}{\succsim} \mathrm{M}(h)$ y $h \overset{w}{\succsim} \mathrm{M}(w)$

Observación 5. Podemos ver claramente las siguientes implicaciones

- Si M es super-estable entonces M es fuertemente estable.
- Si M es fuertemente estable entonces M es débilmente estable.

Veamos con un ejemplo que no siempre existe un emparejamiento fuertemente estable con lo que tampoco existirá siempre un emparejamiento super-estable.

Ejemplo 5. Sean $H = \{a, b\}$ y $W = \{x, y\}$ y sus listas de preferencia:
$$\begin{cases} a : x = y \\ b : x \succ y \end{cases} \text{ y } \begin{cases} x : a \succ b \\ y : a \succ b \end{cases}.$$

Los dos posibles emparejamientos fuertemente estables son:

- $\mathrm{M}_1 = \{(a, x), (b, y)\}$, pero M_1 no es fuertemente estable ya que:
  - $M_1(a) \neq y$.
  - $y \overset{a}{\succsim} x$ y $a \overset{y}{\succ} b$.
- $\mathrm{M}_2 = \{(a, y), (b, x)\}$, pero M_2 no es fuertemente estable ya que:
  - $M_2(a) \neq x$.
  - $x \overset{a}{\succsim} y$ y $a \overset{x}{\succ} b$.

Veamos, con una pequeña modificación del algoritmo Gale-Shapley I, que siempre existe un emparejamiento débilmente estable

```
Inicialmente todos los proponentes estan sin emparejar

MIENTRAS (exista un proponente sin emparejar)
   -propone a los que no haya propuesto segun su orden de preferencia.

   SI (el propuesto esta libre)
      -acepta la propuesta.
   SI NO

      SI (el propuesto prefiere debilmente a su actual pareja)
         -rechaza la propuesta.
      SI NO
         -acepta la propuesta.
      FIN SI

   FIN SI

FIN MIENTRAS
```

Nos referiremos a este algoritmo como Gale-Shapley III

Observación 6. Siendo W el conjunto de los propuestos se ve que: $\forall w \in W$ y $\forall i < j$ se tiene que $\mu_i(w) \overset{w}{\precsim} \mu_j(w) \overset{w}{\precsim} \mu(w)$ teniendo en cuenta que $w \overset{w}{\prec} h \;\; \forall h \in H$.
Además, si existe i tal que $\mu_i(w) = w$ con $w \in W$, entonces $\forall j < i$ tenemos que $\mu_j(w) = w$.

Veamos con un ejemplo que en este caso sí que importa el orden de las proposiciones.

Ejemplo 6. Sean $H = \{a, b\}$ y $W = \{x, y\}$ y sus listas de preferencia: $\begin{cases} a : x \succ y \\ b : x \succ y \end{cases}$ y $\begin{cases} x : a = b \\ y : a \succ b \end{cases}$.

Si el que empieza a proponer es a tenemos la siguiente ejecución:

Paso 1 a propone a x,
x acepta la propuesta ya que está libre,

con lo que $\mu_1(a) = x \quad \text{y} \quad \mu_1(x) = a$
además de
$\mu_1(b) = b$, $\mu_1(y) = y$.

Paso 2 b propone a x,
x rechaza la propuesta ya que $\mu_1(x) = a \overset{x}{\succsim} b$,
b propone a y,
y acepta la propuesta ya que está libre,
con lo que $\mu_2(b) = y \quad \text{y} \quad \mu_2(y) = b$
además de
$\mu_2(a) = x$, $\mu_2(x) = a$

Por lo que tenemos que el emparejamiento es $\mu = \{(a, x), (b, y)\}$

Si el que empieza a proponer es b tenemos la siguiente ejecución:

Paso 1 b propone a x,
x acepta la propuesta ya que está libre,
con lo que $\mu'_1(b) = x \quad \text{y} \quad \mu'_1(x) = b$
además de
$\mu'_1(a) = a$, $\mu'_1(y) = y$.

Paso 2 a propone a x,
x rechaza la propuesta ya que $\mu'_1(x) = b \overset{x}{\succsim} a$,
a propone a y,
y acepta la propuesta ya que está libre,
con lo que $\mu'_2(a) = y \quad \text{y} \quad \mu'_2(y) = a$
además de
$\mu'_2(b) = x$, $\mu'_2(x) = b$

Por lo que tenemos que el emparejamiento es $\mu' = \{(a, y), (b, x)\}$

Teorema 1.3.1. *El algoritmo Gale-Shapley III produce un emparejamiento débilmente estable.*

Demostración. Si μ es el emparejamiento generado por el algoritmo Gale-Shapley III, supongamos por reducción al absurdo que μ no es débilmente estable con lo que existe un proponente $h \in H$ y un propuesto $w \in W$ tal que $\mu(h) \neq w$ y $w \overset{h}{\succ} \mu(h)$, $h \overset{w}{\succ} \mu(w)$.

Dado que los proponentes realizan sus proposiciones en orden decreciente según su lista de preferencias y como $w \overset{h}{\succ} \mu(h)$, tenemos que existe i tal que en la etapa i-ésima h ha propuesto a w, y dado que $\mu(h) \neq w$ existe $j \geq i$ tal que en la etapa j-ésima w ha rechazado a h en favor de h' con $h' \in H - \{h\}$ por lo que $h' \overset{w}{\succ} h$ y $\mu_j(w) = h'$. Ahora bien, por la Observación 6 $\mu(w) \overset{w}{\succsim} \mu_j(w) = h'$ lo que es absurdo ya que tenemos que $\mu(w) \overset{w}{\succsim} \mu_j(w) = h' \overset{w}{\succ} h$ habiendo supuesto que $h \overset{w}{\succ} \mu(w)$.

Por lo que el emparejamiento generado por el algoritmo es débilmente estable.

$\square$

Podemos observar que no siempre existe un emparejamiento óptimo, para los proponentes, entre los débilmente estables. Efectivamente en el Ejemplo 6 tenemos como únicos emparejamientos débilmente estables $\mu = ((a, x), (b, y))$ y $\mu' = ((a, y), (b, x))$ y vemos que $\mu \overset{a}{\succ} \mu'$ y que $\mu' \overset{b}{\succ} \mu$ con lo que ni μ ni μ' son óptimos para el conjunto de los proponentes $H = \{a, b\}$.

Capítulo 2

Alternativas

En este segundo cápitulo vamos a proponer dos algoritmos alternativos al propuesto por Gale y Shapley. Estos dos algoritmos son los propuestos por Fuku, Namatame y Kaizouji [3]. El objetivo de buscar nuevos algoritmos es dar un emparejamiento más equilibrado entre proponentes y propuestos, además de buscar un emparejamiento que satisfaga a los dos grupos. Para ello se deja a un lado el concepto individualista de emparejamiento estable.

2.1. Primer Algoritmo Alternativo

En este algoritmo los individuos de los dos grupos a emparejar, a priori, no se consideran ni proponentes ni propuestos, sino que se ordenan según el deseo que generen en el otro grupo.

Siendo n la dimensión de cada grupo, a cada individuo le asignaremos un vector $(q_1, q_2, \ldots, q_n)$ donde q_i es el número de listas de preferencia en las que el individuo está posicionado en la i-ésima posición y le atribuiremos la puntuación $\sum_1^n (q_i \cdot (n - i + 1))$. Según esa puntuación se ordenan todos los individuos, de ambos grupos, de manera decreciente y en este orden irán haciendo sus proposiciones.

En caso de empate en la puntuación se compararán los q_i de manera creciente hasta deshacer el empate. Por ejemplo, si los individuos m y m' con vectores $(q_{m1}, q_{m2}, \cdots, q_{mn})$ y $(q_{m'1}, q_{m'2}, \cdots, q_{m'n})$ tienen la misma puntuación, si $q_{mi} = q_{m'i} \quad \forall i < j$ y $q_{mj} > q_{m'j}$, entonces m hará proposiciones antes que m'. Si $q_{mi} = q_{m'i}$ $\forall i \in \{1, 2, \cdots, n\}$ el orden entre ellos se decidirá aleatoriamente.

Partiendo de ese orden, cada individuo, según su lista de preferencia, hace sucesivas proposiciones hasta que es emparejado. Las proposiciones serán rechazadas si el propuesto está emparejado ya, y serán aceptadas en caso de que el propuesto esté sin pareja. A continuación mostramos el algoritmo,

```
Se ordenan todos los individuos en funcion del deseo
generado en el otro grupo

En base a ese orden se haran las proposiciones

MIENTRAS (exista un individuo sin pareja)
   -El individuo propone segun su orden de preferencias

   -SI (el propuesto esta libre)
      -Acepta
      -El siguiente proponente, segun el orden fijado,
      hara sus propuestas.
   -SI NO
      -Rechaza la propuesta
      -El proponente propone a su siguiente opcion

   FIN SI

FIN MIENTRAS
```

Nos referiremos a este algoritmo como Alternativo I

Veamos con un ejemplo la ejecución del mismo.

Ejemplo 7. Sean los conjuntos $H = \{a, b, c, d\}$ y $W = \{w, x, y, z\}$ y sus respectivas listas de preferencia

$$\left\{\begin{array}{l} \overset{a}{\succ}: y\ z\ x\ w \\ \overset{b}{\succ}: w\ y\ x\ z \\ \overset{c}{\succ}: x\ w\ y\ z \\ \overset{d}{\succ}: x\ w\ y\ z \end{array}\right. \quad \text{y} \quad \left\{\begin{array}{l} \overset{w}{\succ}: d\ a\ b\ c \\ \overset{x}{\succ}: d\ a\ c\ b \\ \overset{y}{\succ}: a\ b\ c\ d \\ \overset{z}{\succ}: a\ b\ d\ c \end{array}\right.$$

Calculamos la puntuación de cada individuo:

- a tiene como vector asignado $(2, 2, 0, 0)$ lo que le da una puntuación de $2 \cdot 4 + 2 \cdot 3 + 0 \cdot 2 + 0 \cdot 1 = 14$.

- b tiene como vector asignado $(0,2,1,1)$ lo que le da una puntuación de $0\cdot 4+2\cdot 3+1\cdot 2+1\cdot 1=9$.
- c tiene como vector asignado $(0,0,2,2)$ lo que le da una puntuación de $0\cdot 4+0\cdot 3+2\cdot 2+2\cdot 1=6$.
- d tiene como vector asignado $(2,0,1,1)$ lo que le da una puntuación de $2\cdot 4+0\cdot 3+1\cdot 2+1\cdot 1=11$.
- w tiene como vector asignado $(1,2,0,1)$ lo que le da una puntuación de $1\cdot 4+2\cdot 3+0\cdot 2+1\cdot 1=11$.
- x tiene como vector asignado $(2,0,2,0)$ lo que le da una puntuación de $2\cdot 4+0\cdot 3+2\cdot 2+0\cdot 1=12$.
- y tiene como vector asignado $(1,1,2,0)$ lo que le da una puntuación de $1\cdot 4+1\cdot 3+2\cdot 2+0\cdot 1=11$.
- z tiene como vector asignado $(0,1,0,3)$ lo que le da una puntuación de $0\cdot 4+1\cdot 3+0\cdot 2+3\cdot 1=6$.

Así pues el orden es $a,\ x,\ d,\ w,\ y,\ b,\ z,\ c$. En este ejemplo w e y tienen ambos una puntuación de 11 y unos vectores asociados $(1,2,0,1)$ y $(1,1,2,0)$ respectivamente, y $p_{w1}=p_{y1}=1$, pero $p_{w2}=2>p_{y2}=1$ por lo que w propone antes que y.

Las proposiciones serán las siguientes:

- a propone a y,
  y acepta ya que está libre.
- x propone a d,
  d acepta ya que está libre.
- d no propone ya que está emparejado con x
- w propone a d,
  d rechaza ya que está emparejado con x.
  w propone a a,
  a rechaza ya que está emparejado con y.
  w propone a b,
  b acepta ya que está libre.
- y no propone ya que está emparejado con a

- b no propone ya que está emparejado con w
- z propone a a,
  a rechaza ya que está emparejado con y.
  z propone a b,
  b rechaza ya que está emparejado con w.
  z propone a d,
  d rechaza ya que está emparejado con x.
  z propone a c,
  c acepta ya que está libre.
- c no propone ya que está emparejado con z

Por lo que el emparejamiento resultante es $((a, y), (b, w), (c, z), (d, x))$.

2.2. Segundo Algoritmo Alternativo

En el segundo algoritmo propuesto en este tema sí hay un grupo de proponentes y otro de propuestos.

El orden en el que los proponentes proponen viene dado por el deseo generado por cada proponente en el grupo de los propuestos. Como en el algoritmo Alternativo I, a cada proponente le asociamos un vector $(q_1, q_2, \ldots, q_n)$ donde q_i es el número de listas de preferencia de los propuestos en las que el proponente está posicionado en la i-ésima posición y a cada proponente le asignaremos la puntuación $\sum_1^n (q_i \cdot (n - i + 1))$, desempatando en caso de empate como en el algoritmo Aternativo I. Según esa puntuación se ordenan todos los proponentes de manera decreciente. Inicialmente, los propuestos fijan un nivel de aceptación α_0 en sus listas de preferencia, por debajo del cual rechazarán la propuesta, por ejemplo si teniendo un nivel de aceptación de $\alpha = 3$, x es propuesto por a y $1 \leq p_x^a \leq 3$ entonces x aceptará la propuesta, y si $p_x^a > 3$ la rechazará, siendo p_x^a la posición que ocupa el individuo a en la lista de preferencia de x.

Una vez fijado el orden de los proponentes y el nivel inicial de aceptación, cada proponente, en el orden establecido, propone según su preferencia. Si el propuesto está emparejado rechazará la propuesta, y si el propuesto no está emparejado, y el proponente está situado dentro del nivel de aceptación en la lista del propuesto, el propuesto aceptará la proposicion, si no la rechazará dando paso a la proposicion del siguiente proponente.

Cuando hayan propuesto todos los proponentes se hace otra ronda de proposiciones entre los proponentes que están libres, en el mismo orden, y los propuestos aumentan en una unidad el nivel de aceptación, y así rondas sucesivas hasta que todos estén emparejados.

Nótese que cuando el nivel de aceptación sea el cardinal de cada grupo (n) el propuesto que está libre no rechazará ninguna propuesta, por lo que el algoritmo termina en un número finito de pasos y con un emparejamiento perfecto, entendiendo por emparejamiento perfecto como en la Definicin 4.

```
Se ordenan los proponentes en funcion del deseo
generado en el otro grupo

Se fija un nivel inicial de aceptacion

MIENTRAS (exista un proponente sin pareja)
   -El proponente propone segun su orden de preferencias

   SI (el propuesto esta emparejado)
      -Rechaza la propuesta
      -El proponente propone a su siguiente opcion

   SI NO

      SI (el proponente esta en la region de aceptacion)
         -Acepta
      SI NO
         -Rechaza
         -El siguiente proponente, segun el orden fijado,
          hara sus propuestas
      FIN SI

   FIN SI

   -El nivel de aceptacion se incrementa en una unidad

FIN MIENTRAS
```

Nos referiremos a este algoritmo como Alternativo II

En el ejemplo que sigue veremos la ejecución del algoritmo y lo haremos con dos niveles iniciales de aceptación diferentes.

Ejemplo 8. Sean el conjunto de proponentes $H = \{a, b, c, d\}$ y el de propuestos $W = \{w, x, y, z\}$ y sus respectivas listas de preferencia

$$\begin{cases} \overset{a}{\succ}: y\ w\ x\ z \\ \overset{b}{\succ}: z\ w\ y\ x \\ \overset{c}{\succ}: y\ x\ z\ w \\ \overset{d}{\succ}: z\ w\ y\ x \end{cases} \quad \text{y} \quad \begin{cases} \overset{w}{\succ}: b\ c\ d\ a \\ \overset{x}{\succ}: b\ c\ a\ d \\ \overset{y}{\succ}: b\ c\ d\ a \\ \overset{z}{\succ}: b\ a\ c\ d \end{cases}$$

Calculamos la puntuación de cada proponente:

- a tiene como vector asignado $(0, 1, 1, 2)$ lo que le da una puntuación de $0 \cdot 4 + 1 \cdot 3 + 1 \cdot 2 + 2 \cdot 1 = 7$.
- b tiene como vector asignado $(4, 0, 0, 0)$ lo que le da una puntuación de $4 \cdot 4 + 0 \cdot 3 + 0 \cdot 2 + 0 \cdot 1 = 16$.
- c tiene como vector asignado $(0, 3, 1, 0)$ lo que le da una puntuación de $0 \cdot 4 + 3 \cdot 3 + 1 \cdot 2 + 0 \cdot 1 = 11$.
- d tiene como vector asignado $(0, 0, 2, 2)$ lo que le da una puntuación de $0 \cdot 4 + 0 \cdot 3 + 2 \cdot 2 + 2 \cdot 1 = 6$.

Así pues el orden es b, c, a, d. Veamos ahora cómo se construye el emparejamiento, siguiendo el algoritmo descrito y fijado un nivel inicial de aceptación de 1.

Las proposiciones serán las siguientes:

- Empezamos con un nivel de aceptación de 1.
  - b propone a z,
    z acepta ya que está libre y $p_z^b = 1$ (b ocupa la 1^a posición en la lista de z).
  - c propone a y,
    y rechaza ya que $p_y^c = 2$.

- a propone a y,
  y rechaza ya que $p_y^a = 4$.
- d propone a z,
  z rechaza ya que está emparejado con b.
- d propone a w,
  w rechaza ya que $p_w^d = 3$.

- El nivel de aceptación se incrementa en una unidad quedando en 2.

  Teniendo la pareja (b, z) hecha, continuamos con las propuestas:

  - c propone a y,
    y acepta ya que está libre y $p_y^c = 2$.
  - a propone a y,
    y rechaza ya que está emparejado con c.
  - a propone a w,
    w rechaza ya que $p_w^a = 4$.
  - d propone a z,
    z rechaza ya que está emparejado con b.
  - d propone a w,
    w rechaza ya que $p_w^d = 3$.

- El nivel de aceptación se incrementa en una unidad quedando en 3.

  Teniendo las parejas $(b, z), (c, y)$ hechas, continuamos con las propuestas:

  - a propone a y,
    y rechaza ya que está emparejado con c.
  - a propone a w,
    w rechaza ya que $p_w^a = 4$.
  - d propone a z,
    z rechaza ya que está emparejado con b.
  - d propone a w,
    w acepta ya que está libre y $p_w^d = 3$.

- El nivel de aceptación se incrementa en una unidad quedando en 4.

Teniendo las parejas $(b, z), (c, y), (d, w)$ hechas, continuamos con las propuestas:

- a propone a y,
  y rechaza ya que está emparejado con c.
- a propone a w,
  w rechaza ya que está emparejado con d.
- a propone a x,
  x acepta ya que está libre y $p_x^a = 3$.

Así que el emparejamiento final queda $\{(a, x), (b, z), (c, y), (d, w)\}$.

Si hacemos el mismo ejemplo pero con un nivel de aceptación inicial de 4, como las listas de preferencia son las mismas, el orden en el que los propuestos hacen sus propuestas no cambia y el emparejamiento tendría la siguiente ejecución:

- Empezamos con un nivel de aceptación de 4.
  - b propone a z,
    z acepta ya que está libre y $p_z^b = 1$.
  - c propone a y,
    y acepta ya que está libre y $p_y^c = 2$.
  - a propone a y,
    y rechaza ya que está emparejado con c.
  - a propone a w,
    w acepta ya que está libre y $p_w^a = 4$.
  - d propone a z,
    z rechaza ya que está emparejado con b.
  - d propone a w,
    w rechaza ya que está emparejado con a.
  - d propone a y,
    y rechaza ya que está emparejado con c.
  - d propone a x,
    x acepta ya que está libre y $p_x^d = 4$.

Por lo que el emparejamiento final queda $\{(a, w), (b, z), (c, y), (d, x)\}$, distinto de cuando el nivel inicial de aceptación era $\alpha_0 = 1$.

Capítulo 3

Comparación de los 3 algoritmos

En los cápitulos anteriores hemos visto tres algoritmos para generar emparejamientos cuando se tienen listas de preferencias completas y estrictas, que son:

- Gale-Shapley I
- Alternativo I
- Alternativo II.

Recordamos que los tres algoritmos generan un emparejamiento perfecto, o sea que a cada individuo de cada grupo se le asigna un individuo del otro grupo.

En este cápitulo cuantificaremos la satisfación conseguida por el emparejamiento resultante de cada algoritmo, con el objetivo de compararlos.

3.1. Introducción

Introducimos los conceptos que van a cuantificar la satisfacción de los emparejamientos.

- La satisfacción de un individuo a en el emparejamiento M viene dada por
$$P_{\mathrm{M}}(a) = n - p_a^{\mathrm{M}(a)} + 1.$$
Recordamos que n es el tamaño de cada grupo y que p_a^x denota la posición que ocupa x en la lista de preferencia de a (siendo $p_a^x = 1$ cuando x es el más deseado por a).

  Se observa que $1 \leq P_{\mathrm{M}}(a) \leq n$. Será máxima cuando a esté emparejado con su primera opción, o sea $p_a^{\mathrm{M}(a)} = 1$, y será mínima cuando $p_a^{\mathrm{M}(a)} = n$.

- Si H y W son los dos grupos a emparejar y M un emparejamiento cualquiera, cuantificamos la satisfacción de sendos grupos mediante
$$S_H = \sum_{h \in H} P_{\mathrm{M}}(h) \quad \text{y} \quad S_W = \sum_{w \in W} P_{\mathrm{M}}(w).$$

- Con estos conceptos previos cuantificamos la satisfación del conjunto de los dos grupos, en el emparejamiento M, de dos formas diferentes:
  - $S_{suma} = \sum_{h \in H}(P_{\mathrm{M}}(h) + P_{\mathrm{M}}(\mathrm{M}(h)))$.
  - $S_{producto} = \sum_{h \in H}(P_{\mathrm{M}}(h) \cdot P_{\mathrm{M}}(\mathrm{M}(h)))$.

Observación 7. Si M es un emparejamiento perfecto, que es lo que consideramos, podemos observar que:

- $\sum_{h \in H}(P_{\mathrm{M}}(h) + P_{\mathrm{M}}(\mathrm{M}(h))) = \sum_{w \in W}(P_{\mathrm{M}}(w) + P_{\mathrm{M}}(\mathrm{M}(w)))$.
- $\sum_{h \in H}(P_{\mathrm{M}}(h) \cdot P_{\mathrm{M}}(\mathrm{M}(h))) = \sum_{w \in W}(P_{\mathrm{M}}(w) \cdot P_{\mathrm{M}}(\mathrm{M}(w)))$.

Veamos con un ejemplo la aplicación de estos conceptos.

Ejemplo 9. Sean los conjuntos $H = \{a, b\}$ y $W = \{x, y\}$, sus respectivas listas de preferencia $\begin{cases} \overset{a}{\succ}: y\ x \\ \overset{b}{\succ}: x\ y \end{cases}$ y $\begin{cases} \overset{x}{\succ}: a\ b \\ \overset{y}{\succ}: a\ b \end{cases}$ y el emparejamiento $\mathrm{M} = \{(a, y), (b, x)\}$.

Tenemos que:

- la satisfacción de cada individuo es $P_{\mathrm{M}}(a) = 2 - 1 + 1 = 2$, $P_{\mathrm{M}}(b) = 2$, $P_{\mathrm{M}}(x) = 1$, $P_{\mathrm{M}}(y) = 2$.
- La satisfacción de cada grupo es $S_H = P_{\mathrm{M}}(a)+P_{\mathrm{M}}(b) = 2+2 = 4$ y $S_W = P_{\mathrm{M}}(x) + P_{\mathrm{M}}(y) = 1 + 2 = 3$
- $S_{suma} = (\ P_{\mathrm{M}}(a)+P_{\mathrm{M}}(y)\ )+(\ P_{\mathrm{M}}(b)+P_{\mathrm{M}}(x)\ ) = 2+2+2+1 = 7$
- $S_{producto} = (\ P_{\mathrm{M}}(a) \cdot P_{\mathrm{M}}(y)\ )+(\ P_{\mathrm{M}}(b) \cdot P_{\mathrm{M}}(x)\ ) = 2 \cdot 2+2 \cdot 1 = 6$

En resumen, los parámetros que tenemos para la comparación son los siguientes:

- Dos puntuaciones para cada emparejamiento
  - S_{suma}
  - $S_{producto}$
- la satisfacción del conjunto de los proponentes S_H.
- la satisfacción del conjunto de los propuestos S_W.

Observación 8. Como los cuatro parámetros que hemos definido para medir la satisfacción de los emparejamientos crecen cuando el tamaño de cada grupo crece y con el objetivo de que las gráficas sean legibles, hacemos una asignación ponderarando los parametros (S_{suma}, $S_{producto}$, S_H, S_W) con respecto al tamaño de cada grupo, como sigue:

- $S_{suma} := S_{suma} \cdot (\frac{1000}{2 \cdot n^2})$
- $S_H := S_H \cdot (\frac{1000}{2 \cdot n^2})$
- $S_W := S_W \cdot (\frac{1000}{2 \cdot n^2})$
- $S_{producto} := S_{producto} \cdot (\frac{1000}{n^3})$

Diremos que una lista de preferencias ha sido generada de manera aleatoria y uniforme si para cada i, el i-ésimo elemento de la lista ha sido generado mediante una variable aleatoria discreta con función de densidad uniforme entre los posibles elementos.

Como a la hora de generar aleatoriamente una lista de preferencia, de un individuo, se genera en orden, o sea el elemento que ocupa la i-ésima posición de la lista es el i-ésimo generado, tenemos que cada individuo, que todavia no está posicionado en la lista, tiene una probabilidad $\frac{1}{n-i+1}$.

Diremos que una lista de preferencias ha sido generada de manera aleatoria y no uniforme si para cada i, el i-ésimo elemento de la lista ha sido generado mediante una variable aleatoria discreta con función de densidad no uniforme, en la que un elemento fijo tiene el doble de probabilidad que los demás, mientras no este posicionado en la lista de preferencia, o sea $\frac{2}{n-i+2}$ frente a $\frac{1}{n-i+2}$ del resto de los individuos.

Para hacer la comparación realizamos series, de 1000 ejemplos de relaciones de preferencia en las que cada lista de preferencia es generada de manera aleatoria y uniformemente y 1000 ejemplos de relaciones de preferencia en las que cada lista de preferncia es generada de manera aleatoria y no uniformemente, en cada una de las siguientes condiciones:

- Tres tamaños distintos de los grupos, $n = 30$, $n = 100$, $n = 200$.
- Para cada tamaño n, tres niveles de aceptación diferentes, 1, $\frac{n}{2}$, n.

En cada ejemplo hallamos el emparejamiento generado por los distintos algoritmos, Gale-Shapley I, Alternativo I y Alternativo II.

De cada emparejamiento calculamos los cuatro parámetros que cuantifican la satisfacción, S_{suma}, $S_{producto}$, S_H y S_W.

De cada serie, de 1000 ejemplos, en cada una de las condiciones, calculamos el promedio y la desviación típica de S_{suma}, $S_{producto}$, S_H y S_W, como mostramos en el Apéndice 1. Y a partir de esos promedios se lleva a cabo la comparación.

Todas las series se han hecho con el software diseñado en este trabajo.

3.2. Comparación

Lo primero que hacemos es ver si la satisfación generada por cada emparejamiento (medida por $S_{suma},\ S_{producto},\ S_H,\ S_W$), varia dependiendo de si la generación de la relación de preferencia es de manera aleatoria y uniforme o no uniforme, para ello realizamos, con el SPSS, contrastes $Mann-Whitney \begin{cases} H_0 : \mu_1 = \mu_2 \\ H_1 : \mu_1 \neq \mu_2 \end{cases}$ donde μ_1 es el promedio del parametro que se este contrastando, con relación de preferencias generadas de manera aleatoria y distribución uniforme, y μ_2 es el promedio del parametro que se este contrastando, con relación de preferencias generadas de manera aleatoria y distribución no uniforme. Todos los contrastes realizados los mostramos en el Apendice 1.

Los contrastes los realizamos en cada una de las condiciones contempladas en este capitulo (tamaño de cada grupo, tipo de emparejamiento y de nivel inicial de aceptación). Por ejemplo vease en el Apendice 1 la Figura A.1 en el que se hacen los contrastes de los cuatro parametros $S_{suma},\ S_{producto},\ S_H,\ S_W$ en el emparejamiento generado por el algoritmo Gale-Shapley con un tamaño de cada grupo $n = 30$.

Nótese que los diferentes niveles de aceptación solo influyen en la ejecución del algoritmo Alternativo II.

Aunque vemos, en la Figura A.8, que se rechaza la hipotesis nula en el parámetro S_H, consideraremos que el generar de manera uniforme o no uniforme no influye en la satisfación de cada emparejamiento.

En lo que mostramos a continuación no se hace diferencia entre las listas de preferencia generadas aleatoriamente con distribución uniforme y no uniforme.

En la Figura 3.1 vemos que en el emparejamiento generado por el algoritmo Alternativo II y para los tres tamaños de grupos considerados $n = 30,\ 100,\ 200$, el nivel incial de aceptación $\alpha_0 = 1$ es el que mayor satisfacción genera según los parametros $S_{suma},\ S_{producto},\ S_W$, y sin embargo para el parámetro S_H se obtiene una puntuación ligeramente mayor con $\alpha_0 = n$.

Efectivamente en el algoritmo *Alternativo II*, si $\alpha_0 = n$, los propuestos no rechazan ninguna proposición mientras estén libres. En consecuencia la satisfacción de los proponentes y por tanto S_H son mayores que si $\alpha_0 = 1$ ó $\alpha_0 = \frac{n}{2}$.

Como la motivación de buscar nuevos algoritmos que generasen un emparejamiento era la de buscar un emparejamiento que, entre otras cosas, fuese mas equilibrado en cuanto a la satisfacción de los propuestos y proponentes seguiremos analizando los resultados con $\alpha_0 = 1$.

En la Figura 3.2, habiendo fijado $\alpha_0 = 1$, mostramos los promedios de S_{suma}, $S_{producto}$, S_H y S_W para cada emparejamiento en cada uno de los tamaños contemplados ($n = 30,\ 100,\ 200$). Se ve claramente que el emparejamiento más equilibrado es el generado por el algoritmo *Alternativo I*, pero queda muy por debajo con respecto a los otros dos emparejamientos en la satisfacción total de los grupos S_{suma}, $S_{producto}$.

Si nos centramos en los parametros S_{suma} y $S_{producto}$ vemos que el emparejamiento que más satisfacción genera, según los parámetros que hemos definido, es el que viene dado por el algoritmo de Gale-Shapley I.

3.3. Conclusión

Como se dijo en el Capítulo 2, la motivación de buscar algoritmos alternativos al propuesto por Gale y Shapley, es la de generar un emparejamiento que satisfaga a los dos grupos por igual o sea que no sea óptimo para uno de los grupos y pésimo para el otro, además de tener una satisfacción global elevada.

Habiendo hecho la comparacón de los tres algoritmos, no vemos que se cumpla el objetivo que nos llevo a plantear las alternativas, aun así el problema está abierto ya que además de poder generar otros algoritmos, se podría cuantificar la satisfacción del emparejamiento de manera diferente según qué se quiera primar en cada caso.

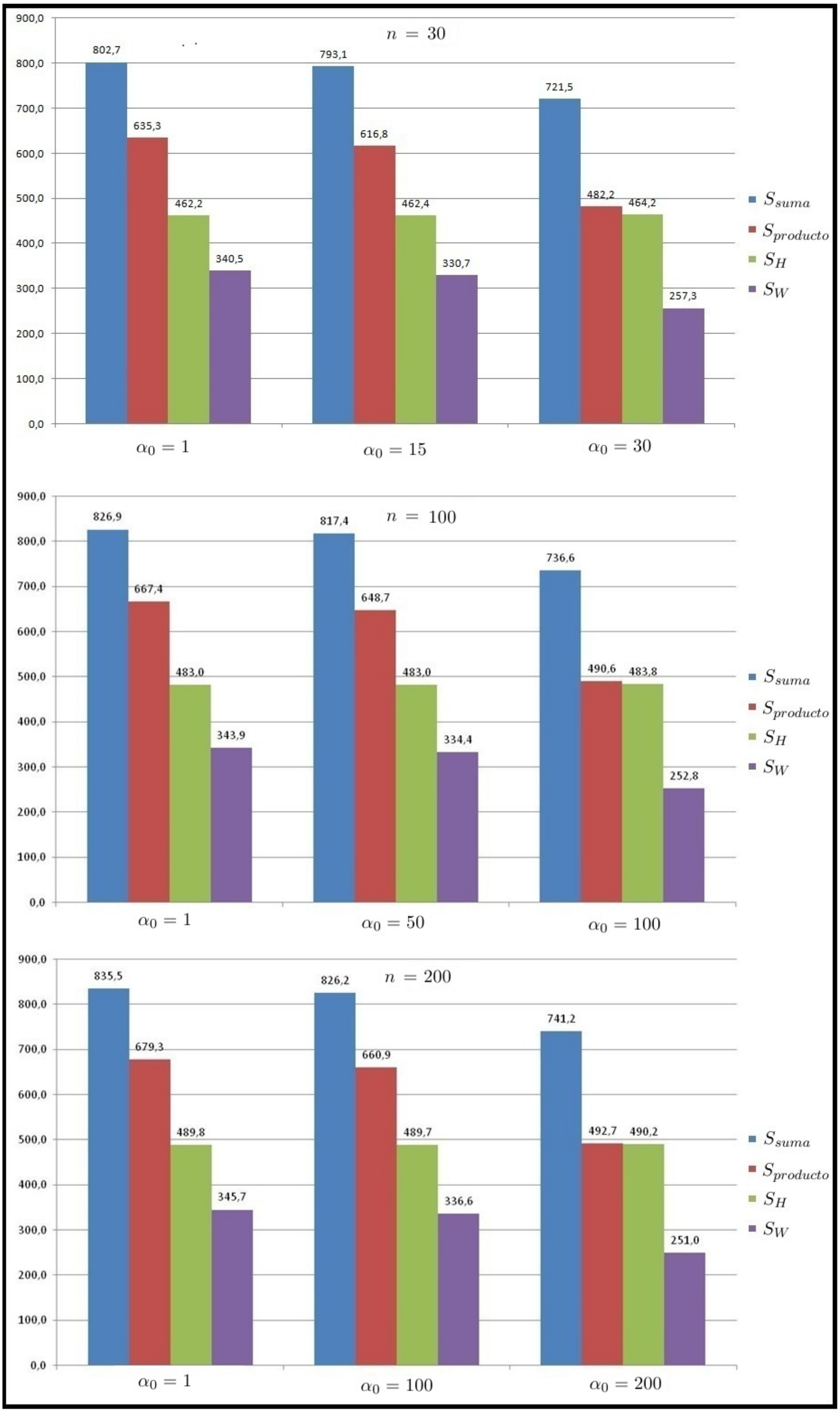


Figura 3.1: Promedio de S_{suma}, $S_{producto}$, S_H, S_W en el emparejamiento *Alternativo II* para cada n y para cada α_0 contemplados.

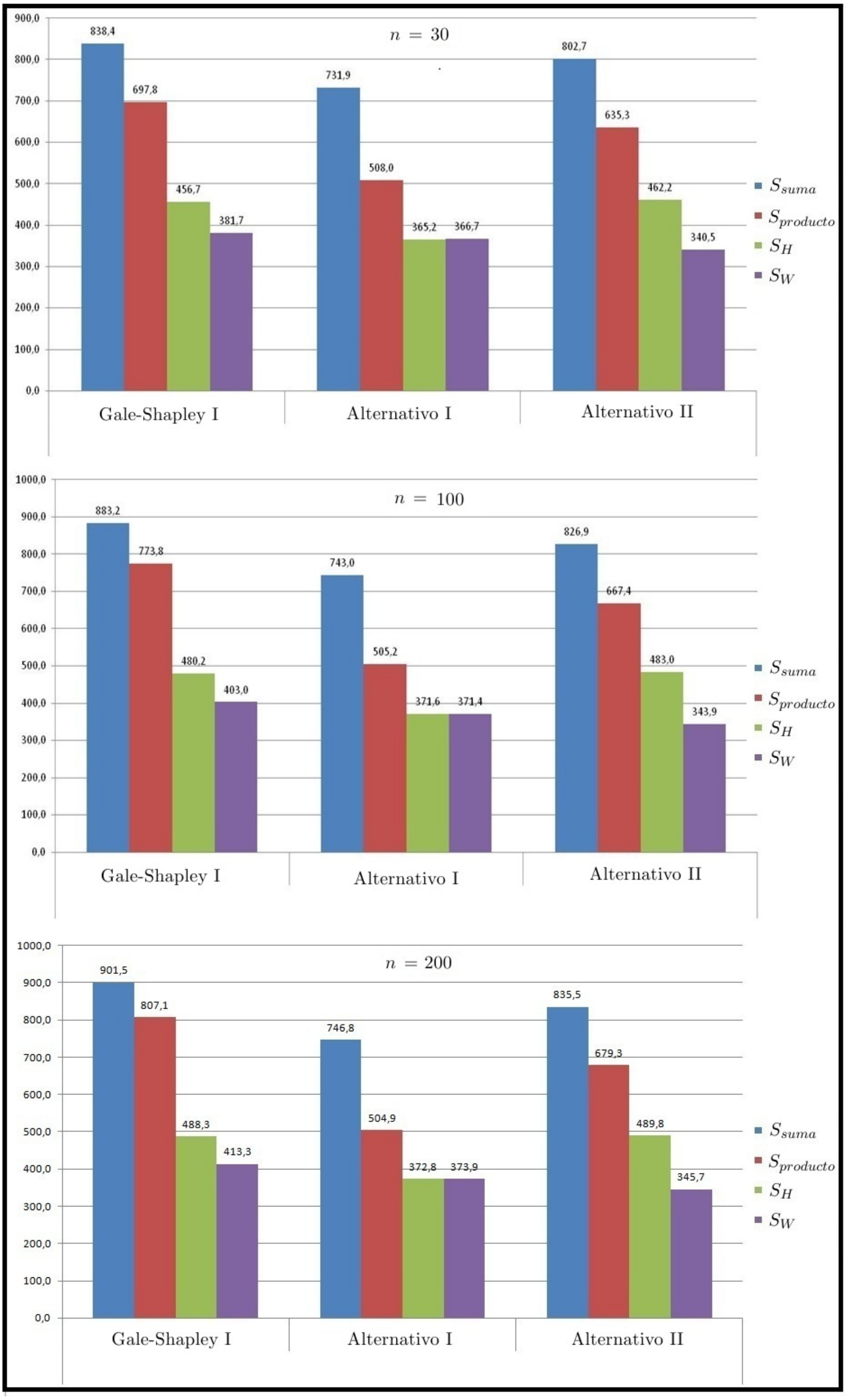


Figura 3.2: Promedio de S_{suma}, $S_{producto}$, S_H, S_W para cada n y para cada emparejamiento con $\alpha_0 = 1$.

Apéndice A

Tabla resumen y contrastes de hipótesis

En este apéndice mostramos, en primer lugar, una tabla-resumen de las series simuladas aleatoriamente, en la que se dan los promedios de los diferentes parámetros, que miden la satisfacción del emparejamiento, y sus desviaciones típicas en cada una de las condiciones contempladas en el Capítulo 3.

Y en segundo lugar se muestran los resultados, obtenidos usando SPSS, de hacer los contrastes de hipotesis Mann-Whitney para saber si hay diferencia, o no, entre generar las relaciones de preferencia de manera aleatoria y uniforme y de manera aleatoria y no uniforme. Cada figura muestra cuatro contrastes (uno por cada parámetro) en unas condiciones determinadas, por ejemplo la Figura A.1 muestra los contrastes de los cuatro parámetros obtenidos de evaluar la satisfacción del emparejamiento generado por el algoritmo de Gale-Shapley I, cuando la serie generada corresponde a un $n = 30$.

Iteraciones	Aleatorio	Tamaño grupo	Nivel inicial de aceptación α_0	Emparejamiento	Parametro	Media	Desviacion	% en Media ± Desv.
1000	uniforme	30,0	1,0	Gale-Shapley	S_{suma}	839,0	20,4	68,8
1000	uniforme	30,0	1,0	Gale-Shapley	$S_{producto}$	698,6	41,1	69,4
1000	uniforme	30,0	1,0	Gale-Shapley	S_H	456,5	42,4	86,4
1000	uniforme	30,0	1,0	Gale-Shapley	S_W	382,5	60,4	71,4
1000	uniforme	30,0	1,0	Alternativo1	S_{suma}	730,5	28,9	69,1
1000	uniforme	30,0	1,0	Alternativo1	$S_{producto}$	504,8	50,8	69,5
1000	uniforme	30,0	1,0	Alternativo1	S_H	364,3	46,1	66,7
1000	uniforme	30,0	1,0	Alternativo1	S_W	366,1	45,5	66,0
1000	uniforme	30,0	1,0	Alternativo2	S_{suma}	802,8	24,2	68,3
1000	uniforme	30,0	1,0	Alternativo2	$S_{producto}$	635,3	44,6	68,7
1000	uniforme	30,0	1,0	Alternativo2	S_H	462,2	19,7	66,2
1000	uniforme	30,0	1,0	Alternativo2	S_W	340,6	51,2	68,5
1000	NO uniforme	30,0	1,0	Gale-Shapley	S_{suma}	837,8	20,7	69,8
1000	NO uniforme	30,0	1,0	Gale-Shapley	$S_{producto}$	696,9	41,6	72,6
1000	NO uniforme	30,0	1,0	Gale-Shapley	S_H	456,9	40,2	85,7
1000	NO uniforme	30,0	1,0	Gale-Shapley	S_W	381,0	59,0	72,4
1000	NO uniforme	30,0	1,0	Alternativo1	S_{suma}	733,3	27,4	68,2
1000	NO uniforme	30,0	1,0	Alternativo1	$S_{producto}$	511,2	48,1	68,6
1000	NO uniforme	30,0	1,0	Alternativo1	S_H	366,0	44,3	68,0
1000	NO uniforme	30,0	1,0	Alternativo1	S_W	367,3	44,8	67,8
1000	NO uniforme	30,0	1,0	Alternativo2	S_{suma}	802,6	23,3	69,2
1000	NO uniforme	30,0	1,0	Alternativo2	$S_{producto}$	635,3	43,7	69,5
1000	NO uniforme	30,0	1,0	Alternativo2	S_H	462,2	19,6	68,3
1000	NO uniforme	30,0	1,0	Alternativo2	S_W	340,4	50,0	68,7
1000	uniforme	30,0	15,0	Gale-Shapley	S_{suma}	839,6	20,9	69,8
1000	uniforme	30,0	15,0	Gale-Shapley	$S_{producto}$	699,7	42,0	71,2
1000	uniforme	30,0	15,0	Gale-Shapley	S_H	456,5	40,9	84,8
1000	uniforme	30,0	15,0	Gale-Shapley	S_W	383,1	61,2	70,7
1000	uniforme	30,0	15,0	Alternativo1	S_{suma}	732,8	28,5	67,6
1000	uniforme	30,0	15,0	Alternativo1	$S_{producto}$	509,4	50,3	67,0
1000	uniforme	30,0	15,0	Alternativo1	S_H	366,3	46,9	67,3
1000	uniforme	30,0	15,0	Alternativo1	S_W	366,5	47,3	69,4
1000	uniforme	30,0	15,0	Alternativo2	S_{suma}	793,5	23,0	68,9
1000	uniforme	30,0	15,0	Alternativo2	$S_{producto}$	616,9	42,6	67,9
1000	uniforme	30,0	15,0	Alternativo2	S_H	463,0	19,7	68,5
1000	uniforme	30,0	15,0	Alternativo2	S_W	330,5	49,5	69,0
1000	NO uniforme	30,0	15,0	Gale-Shapley	S_{suma}	839,0	19,9	69,2
1000	NO uniforme	30,0	15,0	Gale-Shapley	$S_{producto}$	699,4	40,3	70,0
1000	NO uniforme	30,0	15,0	Gale-Shapley	S_H	455,5	41,5	84,1
1000	NO uniforme	30,0	15,0	Gale-Shapley	S_W	383,5	59,4	70,5
1000	NO uniforme	30,0	15,0	Alternativo1	S_{suma}	732,0	27,7	67,9
1000	NO uniforme	30,0	15,0	Alternativo1	$S_{producto}$	509,5	48,1	67,3
1000	NO uniforme	30,0	15,0	Alternativo1	S_H	365,7	44,9	67,1
1000	NO uniforme	30,0	15,0	Alternativo1	S_W	366,3	44,2	70,0
1000	NO uniforme	30,0	15,0	Alternativo2	S_{suma}	792,7	23,0	67,4
1000	NO uniforme	30,0	15,0	Alternativo2	$S_{producto}$	616,8	42,1	66,8
1000	NO uniforme	30,0	15,0	Alternativo2	S_H	461,9	19,4	65,4
1000	NO uniforme	30,0	15,0	Alternativo2	S_W	330,8	48,8	66,7
1000	uniforme	30,0	30,0	Gale-Shapley	S_{suma}	840,4	20,4	68,2
1000	uniforme	30,0	30,0	Gale-Shapley	$S_{producto}$	700,9	41,1	69,9
1000	uniforme	30,0	30,0	Gale-Shapley	S_H	457,1	39,7	87,1
1000	uniforme	30,0	30,0	Gale-Shapley	S_W	383,3	58,9	69,2
1000	uniforme	30,0	30,0	Alternativo1	S_{suma}	730,2	28,2	69,6
1000	uniforme	30,0	30,0	Alternativo1	$S_{producto}$	504,8	49,6	68,8
1000	uniforme	30,0	30,0	Alternativo1	S_H	364,7	43,7	69,8
1000	uniforme	30,0	30,0	Alternativo1	S_W	365,5	43,8	69,7

Iteraciones	Aleatorio	Tamaño grupo	Nivel inicial de aceptación α_0	Emparejamiento	Parametro	Media	Desviacion	% en Media ± Desv.
1000	uniforme	30,0	30,0	Alternativo2	S_{suma}	721,5	26,9	67,8
1000	uniforme	30,0	30,0	Alternativo2	$S_{producto}$	481,7	48,4	67,7
1000	uniforme	30,0	30,0	Alternativo2	S_H	464,5	17,2	67,1
1000	uniforme	30,0	30,0	Alternativo2	S_W	256,9	50,4	68,1
1000	NO uniforme	30,0	30,0	Gale-Shapley	S_{suma}	839,0	20,1	69,6
1000	NO uniforme	30,0	30,0	Gale-Shapley	$S_{producto}$	698,9	40,6	70,4
1000	NO uniforme	30,0	30,0	Gale-Shapley	S_H	455,5	41,5	85,8
1000	NO uniforme	30,0	30,0	Gale-Shapley	S_W	383,5	60,4	71,1
1000	NO uniforme	30,0	30,0	Alternativo1	S_{suma}	731,2	27,5	65,8
1000	NO uniforme	30,0	30,0	Alternativo1	$S_{producto}$	507,4	48,0	67,7
1000	NO uniforme	30,0	30,0	Alternativo1	S_H	365,2	46,4	68,8
1000	NO uniforme	30,0	30,0	Alternativo1	S_W	366,0	46,5	67,5
1000	NO uniforme	30,0	30,0	Alternativo2	S_{suma}	721,5	27,2	68,3
1000	NO uniforme	30,0	30,0	Alternativo2	$S_{producto}$	482,7	48,3	68,5
1000	NO uniforme	30,0	30,0	Alternativo2	S_H	464,0	17,8	66,8
1000	NO uniforme	30,0	30,0	Alternativo2	S_W	257,6	50,6	69,2
1000	uniforme	100,0	1,0	Gale-Shapley	S_{suma}	883,0	15,9	67,1
1000	uniforme	100,0	1,0	Gale-Shapley	$S_{producto}$	773,5	32,9	68,4
1000	uniforme	100,0	1,0	Gale-Shapley	S_H	480,2	32,5	97,8
1000	uniforme	100,0	1,0	Gale-Shapley	S_W	402,8	41,9	76,4
1000	uniforme	100,0	1,0	Alternativo1	S_{suma}	742,2	14,6	69,1
1000	uniforme	100,0	1,0	Alternativo1	$S_{producto}$	503,6	30,9	76,9
1000	uniforme	100,0	1,0	Alternativo1	S_H	371,1	24,4	68,1
1000	uniforme	100,0	1,0	Alternativo1	S_W	371,1	25,0	70,0
1000	uniforme	100,0	1,0	Alternativo2	S_{suma}	826,9	13,4	67,0
1000	uniforme	100,0	1,0	Alternativo2	$S_{producto}$	667,3	26,2	66,9
1000	uniforme	100,0	1,0	Alternativo2	S_H	483,1	6,5	68,0
1000	uniforme	100,0	1,0	Alternativo2	S_W	343,8	28,3	66,3
1000	NO uniforme	100,0	1,0	Gale-Shapley	S_{suma}	883,3	15,6	66,8
1000	NO uniforme	100,0	1,0	Gale-Shapley	$S_{producto}$	774,1	32,2	68,6
1000	NO uniforme	100,0	1,0	Gale-Shapley	S_H	480,1	32,3	98,3
1000	NO uniforme	100,0	1,0	Gale-Shapley	S_W	403,2	40,8	75,7
1000	NO uniforme	100,0	1,0	Alternativo1	S_{suma}	743,8	15,2	68,0
1000	NO uniforme	100,0	1,0	Alternativo1	$S_{producto}$	506,9	31,5	67,8
1000	NO uniforme	100,0	1,0	Alternativo1	S_H	372,0	25,5	68,2
1000	NO uniforme	100,0	1,0	Alternativo1	S_W	371,7	25,0	69,1
1000	NO uniforme	100,0	1,0	Alternativo2	S_{suma}	826,9	13,4	68,5
1000	NO uniforme	100,0	1,0	Alternativo2	$S_{producto}$	667,5	26,3	67,9
1000	NO uniforme	100,0	1,0	Alternativo2	S_H	482,9	6,2	67,1
1000	NO uniforme	100,0	1,0	Alternativo2	S_W	344,0	28,4	68,1
1000	uniforme	100,0	50,0	Gale-Shapley	S_{suma}	883,5	15,4	67,0
1000	uniforme	100,0	50,0	Gale-Shapley	$S_{producto}$	774,4	31,8	68,5
1000	uniforme	100,0	50,0	Gale-Shapley	S_H	480,4	29,4	98,6
1000	uniforme	100,0	50,0	Gale-Shapley	S_W	403,1	40,2	76,1
1000	uniforme	100,0	50,0	Alternativo1	S_{suma}	742,2	14,7	67,3
1000	uniforme	100,0	50,0	Alternativo1	$S_{producto}$	503,7	30,1	68,1
1000	uniforme	100,0	50,0	Alternativo1	S_H	370,2	25,5	66,8
1000	uniforme	100,0	50,0	Alternativo1	S_W	372,0	25,4	67,1
1000	uniforme	100,0	50,0	Alternativo2	S_{suma}	817,9	12,9	68,5
1000	uniforme	100,0	50,0	Alternativo2	$S_{producto}$	649,6	25,0	68,1
1000	uniforme	100,0	50,0	Alternativo2	S_H	483,1	6,3	69,0
1000	uniforme	100,0	50,0	Alternativo2	S_W	334,8	26,9	67,6
1000	NO uniforme	100,0	50,0	Gale-Shapley	S_{suma}	883,7	15,0	68,5
1000	NO uniforme	100,0	50,0	Gale-Shapley	$S_{producto}$	774,9	31,0	69,8
1000	NO uniforme	100,0	50,0	Gale-Shapley	S_H	480,0	29,4	97,8
1000	NO uniforme	100,0	50,0	Gale-Shapley	S_W	403,7	39,5	75,8

Iteraciones	Aleatorio	Tamaño grupo	Nivel inicial de aceptación α_0	Emparejamiento	Parametro	Media	Desviacion	% en Media ± Desv.
1000	NO uniforme	100,0	50,0	Alternativo1	S_{suma}	742,6	14,4	70,6
1000	NO uniforme	100,0	50,0	Alternativo1	$S_{producto}$	504,7	30,8	69,4
1000	NO uniforme	100,0	50,0	Alternativo1	S_H	371,5	25,2	67,3
1000	NO uniforme	100,0	50,0	Alternativo1	S_W	371,1	25,7	68,7
1000	NO uniforme	100,0	50,0	Alternativo2	S_{suma}	816,9	12,2	68,6
1000	NO uniforme	100,0	50,0	Alternativo2	$S_{producto}$	647,7	23,6	67,9
1000	NO uniforme	100,0	50,0	Alternativo2	S_H	482,9	6,6	65,7
1000	NO uniforme	100,0	50,0	Alternativo2	S_W	334,1	25,6	68,2
1000	uniforme	100,0	100,0	Gale-Shapley	S_{suma}	883,5	15,2	67,2
1000	uniforme	100,0	100,0	Gale-Shapley	$S_{producto}$	774,4	31,5	69,0
1000	uniforme	100,0	100,0	Gale-Shapley	S_H	480,4	26,7	98,1
1000	uniforme	100,0	100,0	Gale-Shapley	S_W	403,1	40,1	74,2
1000	uniforme	100,0	100,0	Alternativo1	S_{suma}	742,9	14,4	69,2
1000	uniforme	100,0	100,0	Alternativo1	$S_{producto}$	504,8	29,9	69,1
1000	uniforme	100,0	100,0	Alternativo1	S_H	371,2	25,2	69,2
1000	uniforme	100,0	100,0	Alternativo1	S_W	371,6	24,9	71,0
1000	uniforme	100,0	100,0	Alternativo2	S_{suma}	736,7	14,4	66,3
1000	uniforme	100,0	100,0	Alternativo2	$S_{producto}$	490,8	27,5	66,3
1000	uniforme	100,0	100,0	Alternativo2	S_H	483,8	5,6	68,1
1000	uniforme	100,0	100,0	Alternativo2	S_W	252,9	28,2	66,2
1000	NO uniforme	100,0	100,0	Gale-Shapley	S_{suma}	884,1	15,4	67,0
1000	NO uniforme	100,0	100,0	Gale-Shapley	$S_{producto}$	775,7	31,7	68,0
1000	NO uniforme	100,0	100,0	Gale-Shapley	S_H	479,9	26,5	97,9
1000	NO uniforme	100,0	100,0	Gale-Shapley	S_W	404,1	40,2	73,8
1000	NO uniforme	100,0	100,0	Alternativo1	S_{suma}	742,9	14,9	68,5
1000	NO uniforme	100,0	100,0	Alternativo1	$S_{producto}$	505,1	32,1	68,4
1000	NO uniforme	100,0	100,0	Alternativo1	S_H	371,3	24,2	68,0
1000	NO uniforme	100,0	100,0	Alternativo1	S_W	371,6	25,2	68,1
1000	NO uniforme	100,0	100,0	Alternativo2	S_{suma}	736,5	14,9	66,5
1000	NO uniforme	100,0	100,0	Alternativo2	$S_{producto}$	490,4	28,6	66,2
1000	NO uniforme	100,0	100,0	Alternativo2	S_H	483,9	5,7	68,0
1000	NO uniforme	100,0	100,0	Alternativo2	S_W	252,6	29,4	66,2
1000	uniforme	200,0	1,0	Gale-Shapley	S_{suma}	901,8	13,3	66,5
1000	uniforme	200,0	1,0	Gale-Shapley	$S_{producto}$	807,5	26,6	67,0
1000	uniforme	200,0	1,0	Gale-Shapley	S_H	488,3	5,6	71,5
1000	uniforme	200,0	1,0	Gale-Shapley	S_W	413,5	31,5	65,0
1000	uniforme	200,0	1,0	Alternativo1	S_{suma}	746,8	11,2	69,0
1000	uniforme	200,0	1,0	Alternativo1	$S_{producto}$	505,2	22,0	68,0
1000	uniforme	200,0	1,0	Alternativo1	S_H	372,9	18,5	69,0
1000	uniforme	200,0	1,0	Alternativo1	S_W	373,9	17,1	67,5
1000	uniforme	200,0	1,0	Alternativo2	S_{suma}	835,8	10,9	69,5
1000	uniforme	200,0	1,0	Alternativo2	$S_{producto}$	679,8	21,5	70,0
1000	uniforme	200,0	1,0	Alternativo2	S_H	489,8	3,4	65,5
1000	uniforme	200,0	1,0	Alternativo2	S_W	346,0	22,5	69,0
1000	NO uniforme	200,0	1,0	Gale-Shapley	S_{suma}	901,3	12,6	68,5
1000	NO uniforme	200,0	1,0	Gale-Shapley	$S_{producto}$	806,6	25,3	68,5
1000	NO uniforme	200,0	1,0	Gale-Shapley	S_H	488,2	5,3	72,5
1000	NO uniforme	200,0	1,0	Gale-Shapley	S_W	413,1	30,0	69,5
1000	NO uniforme	200,0	1,0	Alternativo1	S_{suma}	746,7	9,7	65,5
1000	NO uniforme	200,0	1,0	Alternativo1	$S_{producto}$	504,7	18,9	68,0
1000	NO uniforme	200,0	1,0	Alternativo1	S_H	372,7	17,1	68,5
1000	NO uniforme	200,0	1,0	Alternativo1	S_W	374,0	17,1	65,5
1000	NO uniforme	200,0	1,0	Alternativo2	S_{suma}	835,3	10,1	73,0
1000	NO uniforme	200,0	1,0	Alternativo2	$S_{producto}$	678,7	19,7	70,5
1000	NO uniforme	200,0	1,0	Alternativo2	S_H	489,8	3,3	68,0
1000	NO uniforme	200,0	1,0	Alternativo2	S_W	345,4	20,7	72,5

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of suma is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,128	Retain the null hypothesis.
Sprod.	The distribution of producto is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,240	Retain the null hypothesis.
S_H	The distribution of proponente is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,556	Retain the null hypothesis.
S_W	The distribution of propuesto is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,267	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.1:
Emparejamiento: Gale-Shapley
$n = 30$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00046 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,827	Retain the null hypothesis.
Sprod.	The distribution of VAR00047 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,767	Retain the null hypothesis.
S_H	The distribution of VAR00048 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,307	Retain the null hypothesis.
S_W	The distribution of VAR00049 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,649	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.2:
Emparejamiento: Gale-Shapley
$n = 100$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00090 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,708	Retain the null hypothesis.
Sprod.	The distribution of VAR00091 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,741	Retain the null hypothesis.
S_H	The distribution of VAR00092 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,586	Retain the null hypothesis.
S_W	The distribution of VAR00093 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,822	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.3:
Emparejamiento: Gale-Shapley
$n = 200$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00021 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,465	Retain the null hypothesis.
Sprod.	The distribution of VAR00022 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,915	Retain the null hypothesis.
S_H	The distribution of VAR00023 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,621	Retain the null hypothesis.
S_W	The distribution of VAR00024 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,990	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.4:
Emparejamiento: Alternativo I
$n = 30$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00081 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,977	Retain the null hypothesis.
Sprod.	The distribution of VAR00082 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,863	Retain the null hypothesis.
S_H	The distribution of VAR00083 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,931	Retain the null hypothesis.
S_W	The distribution of VAR00084 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,944	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.5:
Emparejamiento: Alternativo I
$n = 100$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00095 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,707	Retain the null hypothesis.
Sprod.	The distribution of VAR00096 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,794	Retain the null hypothesis.
S_H	The distribution of VAR00097 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,588	Retain the null hypothesis.
S_W	The distribution of VAR00098 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,975	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.6:
Emparejamiento: Alternativo I
$n = 200$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00011 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,662	Retain the null hypothesis.
Sprod.	The distribution of VAR00012 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,969	Retain the null hypothesis.
S_H	The distribution of VAR00013 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,887	Retain the null hypothesis.
S_W	The distribution of VAR00014 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,709	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.7:
Emparejamiento: Alternativo II
$n = 30$
$\alpha_0 = 1$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00026 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,384	Retain the null hypothesis.
Sprod.	The distribution of VAR00027 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,899	Retain the null hypothesis.
S_H	The distribution of VAR00028 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,010	Reject the null hypothesis.
S_W	The distribution of VAR00029 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,841	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.8:
Emparejamiento: Alternativo II
$n = 30$
$\alpha_0 = 15$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00041 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,849	Retain the null hypothesis.
Sprod.	The distribution of VAR00042 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,678	Retain the null hypothesis.
S_H	The distribution of VAR00043 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,214	Retain the null hypothesis.
S_W	The distribution of VAR00044 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,553	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.9:
Emparejamiento: Alternativo II
$n = 30$
$\alpha_0 = 30$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00056 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,929	Retain the null hypothesis.
Sprod.	The distribution of VAR00057 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,851	Retain the null hypothesis.
S_H	The distribution of VAR00058 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,083	Retain the null hypothesis.
S_W	The distribution of VAR00059 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,753	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.10:
Emparejamiento: Alternativo II
$n = 100$
$\alpha_0 = 1$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00071 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,204	Retain the null hypothesis.
Sprod.	The distribution of VAR00072 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,178	Retain the null hypothesis.
S_H	The distribution of VAR00073 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,151	Retain the null hypothesis.
S_W	The distribution of VAR00074 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,286	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.11:
Emparejamiento: Alternativo II
$n = 100$
$\alpha_0 = 50$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00086 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,705	Retain the null hypothesis.
Sprod.	The distribution of VAR00087 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,746	Retain the null hypothesis.
S_H	The distribution of VAR00088 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,603	Retain the null hypothesis.
S_W	The distribution of VAR00089 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,581	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.12:
Emparejamiento: Alternativo II
$n = 100$
$\alpha_0 = 100$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00100 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,521	Retain the null hypothesis.
Sprod.	The distribution of VAR00101 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,498	Retain the null hypothesis.
S_H	The distribution of VAR00102 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,727	Retain the null hypothesis.
S_W	The distribution of VAR00103 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,562	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.13:
Emparejamiento: Alternativo II
$n = 200$
$\alpha_0 = 1$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00115 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,309	Retain the null hypothesis.
Sprod.	The distribution of VAR00116 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,331	Retain the null hypothesis.
S_H	The distribution of VAR00117 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,306	Retain the null hypothesis.
S_W	The distribution of VAR00118 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,401	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.14:
Emparejamiento: Alternativo II
$n = 200$
$\alpha_0 = 100$

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
Ssuma	The distribution of VAR00130 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,880	Retain the null hypothesis.
Sprod.	The distribution of VAR00131 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,785	Retain the null hypothesis.
S_H	The distribution of VAR00132 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,492	Retain the null hypothesis.
S_W	The distribution of VAR00133 is the same across categories of Uniformidad.	Independent-Samples Mann-Whitney U Test	,648	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figura A.15:
Emparejamiento: Alternativo II
$n = 200$
$\alpha_0 = 200$

Apéndice B

Software

En este apéndice daremos unas directrices a la hora de usar el sofware diseñado en este trabajo.

Como el software se ha diseado en un libro excel mediante programación en Visual Basic, tenemos que para el correcto funcionamiento del software el usuario deberá tener las macros activadas en el excel, en caso que no estén activadas, el siguiente enlace da unas instrucciones de cómo hacerlo

$$https://www.youtube.com/watch?v=zoXM1jY7ozg$$

El libro excel consta de una serie de pestañas (Figura B.1) que se explican a continuación.

Figura B.1: Las pestañas de las que se compone el libro excel

(i) En la pestaña *Simulación* (Figura B.2), dada una dimensión (tamaño de cada grupo), un nivel de aceptación α_0 (para el algoritmo Alternativo II) y una relación de preferencia, hallamos mediante los botones ejecutables:

El emparejamiento generado por cada algoritmo, su correspondiente puntuación y todos los emparejamientos estables bajo esa relación de preferencia (Figura B.2).

La utilidad de los botones ejecutables es la siguiente:

- Aleatorio: genera de manera aleatoria y uniforme (Capitulo 3) una relación preferencia según la dimensión dada.
- Aleatorio2: genera de manera aleatoria y no uniforme (Capitulo 3) unas relación preferencia según la dimensión dada.

  Observación 9. Siendo $(h1, h2, \cdots, hn)$ y $(w1, w2, \cdots, wn)$ los proponentes y propuestos respectivamente. Las listas de preferencia se muestran numéricamente, en la Figura B.2 se puede ver que, por ejemplo, la lista de preferencias de $h1$ se muestra como $h1$ 4 3 6 5 8 2 7 1 lo que significa que $h1$ prefiere a $w4$ antes que a $w3$, a $w3$ antes que a $w6$ y así sucesivamente.

  Observación 10. Las listas de preferencia pueden ser determinadas por el usuario completando la tabla como se muestra en el ejemplo de la Figura B.2.

- Borrar: limpia de datos la pantalla.
- Estables: Muestra los emparejamientos estables (máxima dimensión n=9).

  Observación 11. En los emparejamientos estables sólo se muestran los propuestos, ya que los proponentes están en orden, por ejemplo en la figura B.2 uno de los emparejamientos estables se muestra como 4 5 7 8 1 6 3 2 (puesto en columna) y el emparejamiento estable sería $\{(h1, w4), (h2, w5), (h3, w7), (h4, w8), (h5, w1), (h6, w6), (h7, w3), (h8, w2)\}$.

- Gale-Shapley: Muestra el emparejamiento resultante de ejecutar el algortmo de Gale-Shapley I y las correspondientes puntuaciones en este orden S_{suma}, $S_{producto}$, S_H, S_W.
- Alternativo I: Muestra el emparejamiento resultante de ejecutar el algortmo de Alternativo I y las correspondientes puntuaciones en este orden S_{suma}, $S_{producto}$, S_H, S_W.
- Alternativo II: Muestra el emparejamiento resultante de ejecutar el algortmo de Alternativo II y las correspondientes puntuaciones en este orden S_{suma}, $S_{producto}$, S_H, S_W.

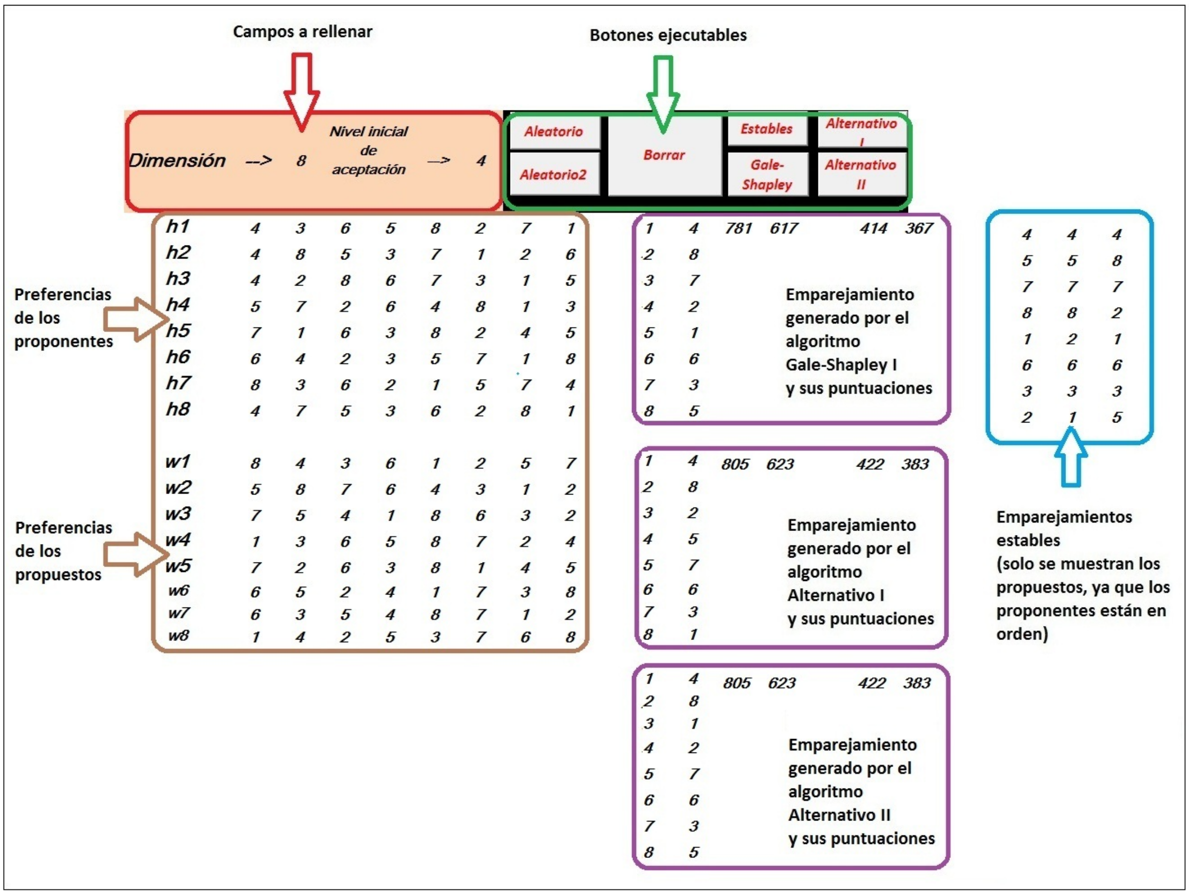


Figura B.2: Pestaña *simulación*

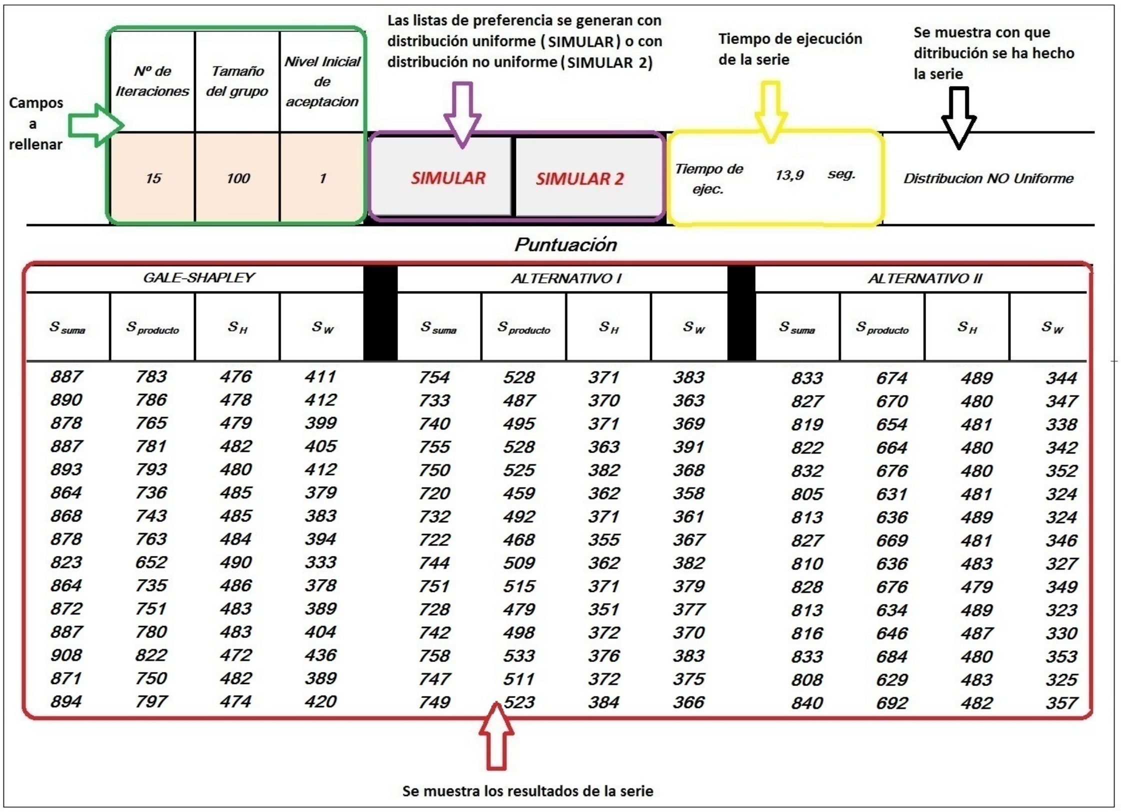
(ii) En la pestaña *Series* (Figura B.3), dados el número de iteraciones (el número de simulaciones que contiene la serie), el tamaño de cada grupo y el nivel inicial de aceptación, calculamos mediante los botones ejecutables:

Los cuatro parámetros que tenemos para cuantificar la satisfacción de cada uno de los tres emparejamientos generados por los distintos algoritmos con unas preferencias aleatorias, y se realizará el número de iteraciones dadas (una por cada fila).

Los botones ejecutables tienen la siguiente función:

- El botón *Simular* realiza la serie simulada generando las listas de preferencia con distribución uniforme (cápitulo 3).
- El botón *Simular* 2 realiza la serie simulada generando las

listas de preferencia con distribución no uniforme (cápitulo 3).



GALE-SHAPLEY				ALTERNATIVO I				ALTERNATIVO II			
S_{suma}	$S_{producto}$	S_H	S_W	S_{suma}	$S_{producto}$	S_H	S_W	S_{suma}	$S_{producto}$	S_H	S_W
887	783	476	411	754	528	371	383	833	674	489	344
890	786	478	412	733	487	370	363	827	670	480	347
878	765	479	399	740	495	371	369	819	654	481	338
887	781	482	405	755	528	363	391	822	664	480	342
893	793	480	412	750	525	382	368	832	676	480	352
864	736	485	379	720	459	362	358	805	631	481	324
868	743	485	383	732	492	371	361	813	636	489	324
878	763	484	394	722	468	355	367	827	669	481	346
823	652	490	333	744	509	362	382	810	636	483	327
864	735	486	378	751	515	371	379	828	676	479	349
872	751	483	389	728	479	351	377	813	634	489	323
887	780	483	404	742	498	372	370	816	646	487	330
908	822	472	436	758	533	376	383	833	684	480	353
871	750	482	389	747	511	372	375	808	629	483	325
894	797	474	420	749	523	384	366	840	692	482	357

Figura B.3: Pestaña *Series*

(iii) En la pestaña *comprobación tendencia* se hace una comprobación de la diferencia entre generar listas de preferencia aleatorias y uniformes y no uniformes.

(iv) En la pestaña *Series variadas* se tiene los datos de las series de 1000 simulaciones con las que se han hecho los cálculos en el cápitulo 3.

(v) En las pestañas *Resultados*, *Tabla dinámica* y *Gráficos* son parte del tratamiento de los datos con los que se ha hecho la comparación del capítulo 3.

Bibliografía

[1] Gale D., Shapley Ll., **College Admissions and the Stability of Marriage**, The American Mathematical Monthly, vol 69, 1962.

[2] Harris J.M., Hirst J.L., Mossinghoff M.J., **Combinatorics and Graph Theory**, Springer, (2008) 248-264.

[3] Fuku T., Namatame A., Kaizouji T., **Collective Efficiency in Two-Sided Maching**, Japan (2002).