***Facultad***
***de***
***Ciencias***

# EMPAREJAMIENTOS ESTABLES

(STABLE MATCHINGS)

Trabajo de fin de Grado
para acceder al

## GRADO EN MATEMÁTICAS

Autora: Paula Santamaría Manteca

Director: Daniel Sadornil Renedo

Octubre - 2020

# Agradecimientos

En primer lugar, le doy las gracias a mi tutor del trabajo, Daniel Sadornil, por haberme dado la oportunidad de realizar un trabajo tan acorde a la temática que yo tenía en mente y que finalmente, al menos en mi opinión, ha resultado ser muy interesante. También es de agradecer cómo se ha volcado y preocupado por mí, y todo el tiempo que le ha dedicado a guiarme y ayudarme con el trabajo en este año tan complicado para todos. Gracias también a todos los profesores que me han dado clase, porque cada uno de ellos forma parte ya de este camino.

Por otra parte le doy las gracias a mi familia, en especial a mis padres y a mi hermano, por el apoyo que me han dado durante estos cuatro años de carrera, y por hacer posible todo cuanto he conseguido hasta hoy.

Por último pero no menos importante, gracias a mis amigos por estar siempre ahí y confiar en mí incluso cuando yo misma no lo hacía. Y gracias en concreto a Marcos, a quien le dedico este trabajo, por la paciencia que ha tenido conmigo todos estos años, y porque sin su ayuda y apoyo no habría podido llegar hasta aquí. Gracias.

# Resumen

En este trabajo se pretende mostrar al lector diferentes tipos de problemas de asignación, cuyas aplicaciones se siguen utilizando a día de hoy en la vida real. Se comienza introduciendo una serie de nociones y resultados importantes de la Teoría de Emparejamientos y su relación con la Teoría de Grafos, en particular bipartitos. A continuación, se trata el primero de los problemas de asignación bipartitos del trabajo, el Problema del Matrimonio Estable, donde se estudiará el concepto de la estabilidad en un emparejamiento, así como el Algoritmo de Gale-Shapley que lo resuelve. Este primer problema da lugar a una extensión bipartita que también se tratará en este texto, el Problema de los Hospitales y Residentes, también conocido como el Problema de la Asignación de Estudiantes a Universidades. Por último, se desarrolla una extensión no bipartita, el Problema de los Compañeros de Habitación, en el cual se presentará el Algoritmo de Irving que ofrece una solución al problema en caso de que esta exista.

**Palabras clave:** Problemas de asignación, emparejamientos, emparejamientos estables.

# Abstract

This paper aims to show the reader different types of assignment problems whose applications are still used actually in real life. It begins by introducing several important notions and results of the Matching Theory and its relationship with the Graph Theory, in particular bipartite. Next, we discuss the first of the bipartite assignment problems of the work, the Stable Marriage Problem, where the concept of stability in a match will be studied, as well as the Gale-Shapley Algorithm that solves it. This first problem gives rise to an bipartite extension that will also be discussed in this text, the Hospitals and Residents Problem, also known as the Assigning Students to Universities Problem. Finally, a non bipartite extension is developed, the Rommates Problem, in which the Irving Algorithm will be presented, which offers a solution to the problem if it exists.

**Key words:** Assigmnet problems, matchings, stable matchings.

# Índice


**Introducción** **1**

**1. Algunos conceptos de grafos y emparejamientos** **3**
- 1.1. Grafos . . . . . 3
- 1.2. Emparejamientos . . . . . 5

**2. El Problema de los Matrimonios Estables** **12**
- 2.1. El Algoritmo de Gale-Shapley . . . . . 13
- 2.2. Resultados y demostraciones . . . . . 17
- 2.3. Variantes del problema . . . . . 20
  - 2.3.1. Listas Incompletas o Inaceptables . . . . . 20
  - 2.3.2. Listas con Indiferencias . . . . . 26

**3. Hospitales y Residentes ó Admisión en Universidades** **30**
- 3.1. Algoritmo y resultados . . . . . 31
- 3.2. Variantes del problema . . . . . 34

**4. Compañeros de habitación** **35**
- 4.1. Primera Etapa del Algoritmo de Irving . . . . . 36
- 4.2. Segunda Etapa del Algoritmo de Irving . . . . . 40

**5. Otros problemas de emparejamientos** **48**

**Bibliografía** **50**

# Introducción

En una empresa deben contratar una serie de personas para realizar unos trabajos, el solicitante $A$ está capacitado para los trabajos 1, 2 y 4; el $B$ para los trabajos 3, 4, 5 y 6; el $C$ para 1 y 5; el $D$ para 1, 3, 4 y 8; el $E$ para 1, 2, 4, 6 y 8; el $F$ para 4 y 6; y el $G$ para los trabajos 3, 5 y 7. ¿Será posible encontrar trabajadores de forma que se realicen todos los trabajos? Si este no fuera el caso, ¿cuál es el máximo número de trabajos que se pueden hacer y quién lo realizaría?

Responder a esta pregunta y otras similares es lo que busca la Teoría de Emparejamientos, que se basa en los problemas de asignación, cuyo principal objetivo es encontrar una manera de emparejar diversos agentes, de tal manera que dicho emparejamiento sea óptimo en cada situación. Estos problemas de asignación son muy utilizados en la vida real, debido a sus múltiples aplicaciones en ámbitos como la Economía, la Educación o en el marco laboral.

El problema base de la asignación puede modelarse en términos de grafos y emparejamientos en ellos. Por ello, lo primero que se trata en esta memoria es presentar ese escenario en términos de grafos, que además permite demostrar ciertas propiedades básicas de los emparejamientos en general. Se ha procurado que el trabajo sea autocontenido, introduciendo los resultados básicos para que el lector pueda seguirlo fácilmente. Un tipo particular de emparejamientos son los estables, que tienen sus orígenes en Estados Unidos, a mediados del siglo XIX, cuando los estudiantes de medicina tenían que ser elegidos como residentes en los diferentes hospitales para realizar las prácticas, y era por tanto necesario organizar a cada residente en un hospital, siguiendo en la medida de lo posible las preferencias de cada una de las partes.

Uno de los problemas más conocidos es el problema de los Matrimonios Estables, que en concreto, es un problema de asignación bipartito que parte de dos conjuntos de personas que tienen el mismo cardinal, hombres y mujeres, donde cada persona de cada grupo tiene una lista de preferencias en la que ordena a los miembros del sexo opuesto. El problema consiste por tanto en encontrar un emparejamiento estable entre ambos grupos de personas.

La estabilidad de este problema de asignación es de suma importancia, ya que no nos conformamos con que un hombre y una mujer estén emparejadas con otra persona si ambos preferirían estar juntos antes que con sus parejas actuales.

A pesar de que este problema ha sido muy estudiado a lo largo de las últimas décadas, los primeros en encontrar una solución para el mismo fueron D.Gale y G.S.Sapley, quienes en 1962 plantearon el llamado Algoritmo de Gale-Shapley. Con este algoritmo se demostró que, si el número de hombres y mujeres es el mismo, siempre existe un emparejamiento estable entre ellos.

El interés de su estudio radica en la infinidad de aplicaciones y problemas de la vida real que se pueden resolver gracias a este algoritmo y a variantes del mismo u otros algoritmos del estilo adaptados para una gran variedad de problemas de asignación estable.

Una de las extensiones más cercanas a este problema es el de los Residentes y Hospitales, también conocido como el Problema de la Admisión de estudiantes a Universidades. Este problema es también bipartito, pero trata de asignar a cada estudiante una universidad, siendo el número de universidades menor que el número de estudiantes.

Además, en este problema cada universidad tiene un límite de estudiantes permitidos en la misma que no se puede superar. El algoritmo que resuelve este problema es una adaptación del Algoritmo de Gale-Shapley.

Una de las extensiones no bipartitas más conocidas del Problema del Matrimonio Estable es el Problema de los Compañeros de Habitación. En este caso, el conjunto de partida en lugar de poderse dividir en dos conjuntos disjuntos, es uno único. Se trata de un conjunto de personas a las que queremos organizar en habitaciones dobles (para dos personas). No fue hasta el año 1985 cuando Robert W.Irving propuso un algoritmo que resolvía dicho problema, el Algoritmo de Irving.

No en todos estos problemas existe siempre una solución estable, incluso se pueden describir diversos tipos de estabilidad según el contexto. Veremos que para este Problema de los Compañeros de Habitación no tiene por qué existir un emparejamiento estable, sin embargo el Algoritmo de Irving nos ofrece uno en el caso de que este exista.

A lo largo de este trabajo de fin de grado iremos tratando estos problemas, estudiando cuándo podemos encontrar estas soluciones y cómo hallarlas en cada caso.

# Capítulo 1

# Algunos conceptos de grafos y emparejamientos

La Teoría de Grafos surgió en el siglo XVIII con el problema de los puentes de Königsberg, que consistía en hallar un camino que cruzara los siete puentes del río Pregel de la ciudad de Königsberg (actuakmente Kaliningrado) pasando una sola vez por cada uno de dichos puentes.

En este capítulo, vamos a introducir una serie de conceptos y propiedades básicas sobre grafos y emparejamientos ([6], [14]), y vamos a ver teoremas conocidos que son la base de muchos otros resultados en este campo.

## 1.1. Grafos

En esta primera sección vamos a definir aquellos conceptos sobre grafos que vamos a utilizar posteriormente.

**Definición 1.1** *Un grafo (finito) $G$ es un par ordenado $G = (V, E)$, donde $V = V(G)$ es un conjunto (finito) cuyos elementos se llaman vértices o nodos, y $E = E(G)$ es un conjunto (finito) cuyos elementos se llaman aristas. Dichas aristas asocian vértices de la forma $e = (x, y)$, con $x, y \in V$, donde $x$ e $y$ se denominan extremos de $e \in E$, y $(x, y) = (y, x)$ son pares no ordenados, por lo que se trata de grafos no dirigidos. Se dice que una arista $e$ es un lazo si $e = (x, x)$ con $x \in V(G)$, es decir, si los dos extremos de la arista son el mismo vértice.*

Dos vértices se dice que son adyacentes si existe una arista que les une. En tal caso, se dice que ambos vértices son incidentes en dicha arista. Se dice que un vértice es aislado si no incide en ninguna arista del grafo. Dos aristas se dice que son adyacentes si tienen un extremo en común.

Además, diremos que un grafo es simple si no hay lazos y dos vértices están unidos a lo sumo por una única arista.

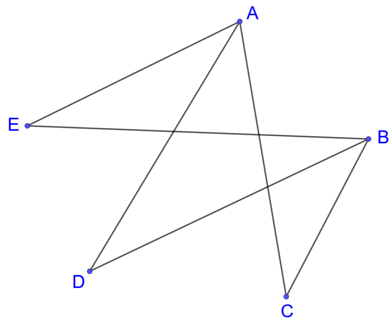



Figura 1.1: Grafo simple

Dados dos grafos, $G_1 = (V_1, E_1)$ y $G_2 = (V_2, E_2)$, se dice que $G_2$ es subgrafo de $G_1$ si $V_2 \subset V_1$, y $A_2 \subset A_1$.

**Definición 1.2** *Sea $G$ un grafo y $X \subset V(G)$ un conjunto de vértices de $G$. Se define el conjunto de vecinos de $X$, y se denota por $N(X)$, al conjunto de vértices distintos del grafo que son adyacentes a algún vértice de $X$.*

**Definición 1.3** *Se denomina grado del vértice $v$, y se denota por $d(v)$, al número de aristas incidentes en $v$, entendiendo que un lazo aporta 2 al grado. Un grafo se dice regular si todos los vértices tienen el mimo grado. Decimos por tanto que un grafo es $k$-regular si todos sus vértices tienen grado $k$.*

Se llama camino a una secuencia de vértices y aristas dentro de un grafo $G$ tal que exista una arista entre cada vértice y el siguiente: $v_1 a_1 v_2 a_2 v_3 ... v_r a_r v_{r+1}$. Se dice que dos vértices están conectados si existe un camino que una ambos vértices. Se dice que un camino es cerrado si termina en el mismo vértice en el que comienza. Un ciclo es un camino cerrado donde no se repite ningún vértice (a excepción del primero, ya que es también el último).

Además, se entiende por grafo conexo a aquel grafo $G = (V, E)$ en el que existe un camino entre cualquier par de vértices distintos del mismo. Dado un grafo $G = (V, E)$ no conexo, se denomina componente conexo a un subgrafo de $G$ conexo maximal (que no está contenido en ningún otro subgrafo de $G$).

**Definición 1.4** *Un grafo $G$ es bipartito si existe una partición de $V$, $V = V_1 \oplus V_2$ (i.e. $V_1 \cap V_2 = \emptyset$ y $V_1 \cup V_2 = V$) tal que todas las aristas de $G$ unen un vértice de $V_1$ con un vértice de $V_2$.*

Vamos a definir a continuación dos conceptos esenciales para la demostración de algunos de los teoremas que veremos en la siguiente sección.

**Definición 1.5** *Sea un grafo $G = (V, E)$. Un conjunto $K \subset V$ se dice que es un recubrimiento por vértices de $G$ si toda arista $e \in E$ es incidente en al menos un vértice $v$ de $K$. Un recubrimiento por vértices no es único, y el tamaño del más pequeño de ellos, se conoce como el número de recubrimiento por vértices de $G$, y se denota por $\tau(G)$.*

*Dado un grafo $G = (V, E)$, un subconjunto de vértices $S \subset V$ se llama conjunto independiente si no hay dos vértices en $S$ que sean adyacentes, es decir, si $V \backslash S$ es un recubrimiento por vértices. El número de independencia de $G$ es el tamaño máximo de un conjunto independiente y se denota por $\alpha(G)$.*

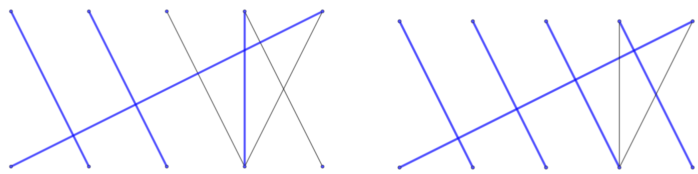
**Definición 1.6** *Sea un grafo $G = (V, E)$. Un conjunto $F \subset E$ se dice que es un recubrimiento por aristas de $G$ si todo vértice $v \in V$, es incidente en al menos una arista $e$ de $F$. Un recubrimiento por aristas no es único, y el tamaño del más pequeño de ellos se conoce como el número de recubrimiento por aristas de $G$ y se denota por $\rho(G)$.*

## 1.2. Emparejamientos

En esta sección veremos algunos resultados importantes sobre emparejamientos así como las demostraciones de los mismos.

**Definición 1.7** *Un emparejamiento en un grafo $G$ es un conjunto de aristas $M$ de $G$ sin vértices comunes, es decir, un subgrafo donde todos los vértices tienen grado menor o igual que 1.*

Además, llamamos emparejamiento máximo a aquel que contiene el número máximo posible de aristas, esto es, que no hay emparejamientos con más aristas. Un emparejamiento se dice que es maximal si no está contenido en otro emparejamiento. Un emparejamiento es perfecto si todos los vértices del grafo forman parte de él.



(a) Emparejamiento maximal

(b) Emparejamiento máximo y perfecto

Figura 1.2

En un grafo bipartito $G = V_1 \cup V_2$ con $|V_1| < |V_2|$ no puede haber emparejamientos perfectos. Además, en un grafo bipartito, se entienden por emparejamientos completos a aquellos en los que todos los vértices de $V_1$ forman parte de los mismos.

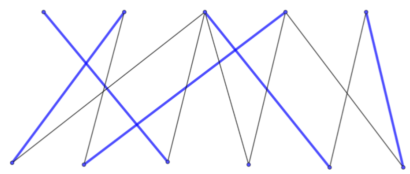
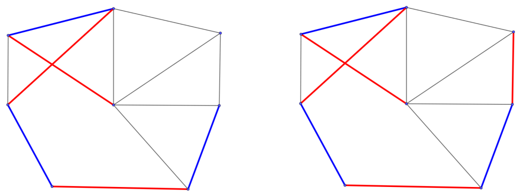


Figura 1.3: Emparejamiento completo

Un vértice se dice saturado por un emparejamiento cuando es extremo de alguna arista del emparejamiento. De no ser así, se denomina vértice libre.

Vamos a demostrar a continuación dos teoremas clásicos relacionados con los emparejamientos: el Teorema de Berge y el Teorema de Hall (ver [7]). Para ello, definiremos algunos conceptos más.

**Definición 1.8** *Sea un grafo $G = (V, E)$ y un emparejamiento $M$ de $G$. Se llama camino alternado a un camino en el cual sus aristas alternativamente pertenecen y no pertenecen a $M$. Se llama camino aumentador para un emparejamiento $M$ a un camino alternado que comienza y termina con aristas no pertenecientes a $M$.*



(a) Camino alternado (b) Camino aumentador

Figura 1.4: Caminos generados a partir del emparejamiento M de aristas azules.

A continuación, mostramos un algoritmo que genera un emparejamiento mediante caminos aumentadores. El Teorema de Berge, que veremos posteriormente, permite demostrar que este algoritmo construye un emparejamiento máximo.

Sea un grafo simple $G = (V, E)$:

- Paso 1: Partimos de un emparejamiento $M$ cualquiera de $G$.
- Paso 2: Buscamos un camino aumentador para $M$.
  - Si existe, vamos al paso 3.
  - Si no existe, vamos al paso 4.
- Paso 3: Generamos el emparejamiento $M'$ a partir del camino aumentador de $M$ (que tendrá por tanto una arista más que $M$), y volvemos al paso 2 tomando $M = M'$.
- Paso 4: $M$ es un emparejamiento máximo.

Vamos a ejemplificar el funcionamiento de este algoritmo con el siguiente ejemplo.

**Ejemplo 1.9** *Sea un grafo simple $G = (V, E)$.*

- *Paso 1: Elegimos un emparejamiento $M$ de partida cualquiera:*

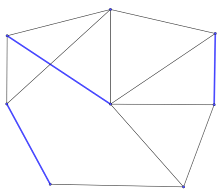


Figura 1.5

- *Paso 2: Encontramos el siguiente camino aumentador:*

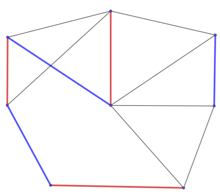


Figura 1.6

- *Paso 3: Generamos el emparejamiento $M'$ y volvemos al Paso 2, donde, al tener $|G| = |M'|$, no existen más caminos aumentadores.*

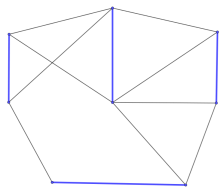


Figura 1.7

- *Paso 4: Hemos obtenido un emparejamiento máximo.*

Veamos ahora el Teorema de Berge, el cual proporciona una condición necesaria y suficiente para que un emparejamiento sea máximo.

**Teorema 1.10 (Berge)** *Sea $M$ un emparejamiento de un grafo $G = (V, E)$. $M$ es máximo si y sólo si $G$ no contiene caminos aumentadores para $M$.*

**Dem:** Sea $G$ un grafo y supongamos primero que $M$ es un emparejamiento máximo. Por reducción al absurdo, supongamos que P es el siguiente camino aumentador de $M$: $v_1 a_1 v_2 a_2 ..., v_{k-1} a_{k-1} v_k$.

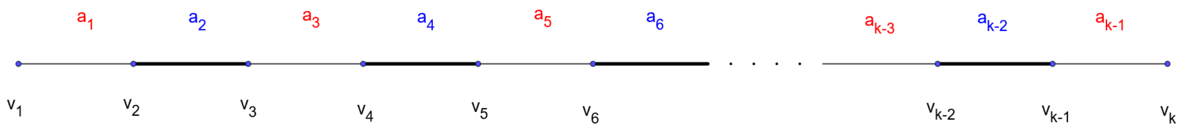



Figura 1.8: Camino aumentador de $M$

Por definición de camino aumentador, $v_1$ y $v_k$ son vértices libres, luego $k$ es par, y por la naturaleza alternativa de dichos caminos, las aristas $a_2, a_4, ..., a_{k-2}$ pertenecen a $M$, y en cambio $a_1, a_3, ..., a_{k-1}$ no pertenecen a $M$. Si ahora definimos el conjunto de aristas $M_1 = (M \setminus \{a_2, ..., a_{k-2}\}) \cup \{a_1, ..., a_{k-1}\}$, entonces $M_1$ es un emparejamiento con una arista más que $M$, lo cual contradice nuestra hipótesis de partida en la que asumíamos que $M$ era máximo. Por tanto, $G$ no puede contener caminos aumentadores.

Recíprocamente, supongamos que $G$ no contiene caminos aumentadores de un emparejamiento $M$ de $G$ y supongamos, por reducción al absurdo, que $M$ no es máximo, luego existe un emparejamiento $M'$ con un número de aristas mayor que el de $M$ ($|M'| > |M|$). Definimos un subgrafo $H$ de $G$ de la siguiente manera: sea $V(H) = V(G)$ y sea $E(H)$ el conjunto de aristas que aparece en exactamente uno y sólo uno de los emparejamientos $M$ y $M'$. Por otro lado, como cada uno de los vértices de $G$ está en a lo sumo una arista de $M$ y a lo sumo en una arista de $M'$, el grado (en $H$) de cada vértice de $H$ es como mucho 2. Esto implica que cada componente conexa de $H$ es un vértice libre, un camino, o un ciclo (porque estos son los únicos grafos tales que todos sus vértices tienen grado menor o igual que 2). Si una componente es un ciclo, entonces tiene que ser un ciclo con un número par de aristas, ya que al ser $M$ y $M'$ emparejamientos (es decir, conjuntos de aristas de $G$ sin vértices en común), las aristas del ciclo en $H$ tendrían que ir alternando entre aristas de $M$ y aristas de $M'$. Además, como $|M'| > |M|$, los vértices libres no aportan ninguna arista a dicho subgrafo, y los ciclos aportan un número par de aristas, entonces debe haber al menos una componente conexa de $H$ que sea un camino que empieza y termina en aristas de $M'$. Este camino sería por tanto un camino aumentador de $M$ (pues todo camino en una componente conexa de $H$ alterna aristas de $M$ y $M'$), lo que contradice la hipótesis de partida de que $M$ no contenía caminos aumentadores. Por tanto, no existe un $M'$ con cardinal mayor que el de $M$, así que $M$ tiene que ser máximo. ■

Ahora vamos a ver el Teorema de Hall para grafos bipartitos, que es de suma importancia en la Teoría de emparejamientos, como ya hemos dicho en este apartado:

**Teorema 1.11 (Hall)** *Sea $G = (X \cup Y, E)$ un grafo bipartito con $|X| \leq |Y|$. $G$ posee algún emparejamiento completo si y sólo si para todo subconjunto $S \subset X$ se cumple que $|N(S)| \geq |S|$.*

**Dem:** Supongamos que $G$ posee algún emparejamiento completo $M$, y sea $S$ un subconjunto cualquiera de $X$. Como $S \subset X$, entonces también existe al menos un emparejamiento completo $M'$ entre $S$ e $Y$, que estaría contenido en $M$. Por tanto, como $M$ es un emparejamiento completo de $X$, y $S$ está contenido en $X$, cada vértice de $S$ está emparejado por $M$ con algún vértice de $Y$. Entonces, al menos habrá $|S|$ vecinos a los vértices de $S$, es decir, $|N(S)| \geq |S|$ (para todo subconjunto $S \subset X$).

Recíprocamente, supongamos ahora que $|N(S)| \geq |S|$ para todo $S \subset X$, y sea $M$ un emparejamiento máximo de $G$. Supongamos que $u \in X$ no está saturado por $M$, y definimos entonces un conjunto $A$ formado por el conjunto de vértices de $G$ que pueden ser conectados con $u$ a través de un camino alternado. Sea $S = A \cap X$, y sea $T = A \cap Y$. Notemos que como $M$ es máximo, por el Teorema de Berge, $G$ no contiene caminos aumentadores. Por otro lado, por definición de $T$, todos los vértices de $T$ pueden ser conectados con $u$ por caminos alternados que empiezan por tanto con una arista que no pertenece a $M$ (puesto que $u$ no está saturado por $M$). Y al ser además $G$ bipartito, con $S \subset X$ y $T \subset Y$, entonces todos los vértices de $T$ han de estar emparejados por $M$ con un vértice de $S \backslash \{u\}$ (ya que de lo contrario, el camino alternado que les uniría con $u$ sería un camino aumentador, el cual no puede existir). De igual manera, por definición de $S$, todos los vértices de $S$ distintos de $u$ pueden ser conectados con $u$ a través de un camino alternado. Y como $u \in S$ pero no está saturado por $M$, para que dichos caminos sean alternados, deben llegar a los vértices de $S$ por aristas pertenecientes a $M$. De aquí deducimos que todo vértice de $S$, a excepción de $u$, está emparejado por $M$ con un vértice de $T$. Por tanto, por definición de los conjuntos $A$, $S$ y $T$, se llega a establecer una biyección entre el conjunto $T$ y $S \backslash \{u\}$, luego $|T| = |S| - 1$ (puesto que $u \in S$). Veamos ahora que $N(S) \subseteq T$:

Sea $b \in N(S)$, entonces existe $a \in S$ tal que la arista $(a, b) \in G$. Como $a \in S$, existe un camino alternado que une $a$ con $u$. Sea $c \in T$ tal que la última arista de dicho camino es $(a, c) \in M$ (porque si no dicho camino sería aumentador y hemos dicho que eso no puede ser). Pueden darse dos situaciones:

- Si $b = c$, entonces $(a, b) \in M$ y $b \in T$.
- Si $b \neq c$, entonces $(a, b) \notin M$ (ya que $a$ ya está emparejado con $c$ por $M$), y por tanto, $b \in T$ porque podría ser conectado con $u$ a través del camino alternado que une $a$ con $u$ añadiéndole la arista $(a, b) \notin M$.

Entonces tenemos que $|N(S)| \leq |T| = |S| - 1 < |S|$, lo cual es una contradicción con nuestra hipótesis de partida ($|N(S)| \geq |S|$). Por tanto, concluimos que ese vértice $u$ no saturado por $M$ no puede existir, luego $M$ es un emparejamiento completo. ■

De los teoremas demostrados anteriormente se deducen los siguientes resultados.

**Corolario 1.12** *Sea $G = (X \cup Y, E)$ un grafo bipartito con $|X| \leq |Y|$. Si existe un número natural $k$ tal que para todo $x \in X$ y para todo $y \in Y$, $d(x) \geq k$, y $d(y) \leq k$, entonces $G$ tiene un emparejamiento completo de $X$ en $Y$.*

**Dem:** Sea $A$ un subconjunto de $X$. Como por hipótesis $d(x) \geq k$ para todo vértice $x \in X$, y $A \subseteq X$, entonces hay al menos $k|A|$ aristas con un extremo en $A$. El otro extremo de estas aristas está en $N(A)$. Como además por hipótesis $d(y) \leq k$ para todo $y \in Y$, entonces también se cumple que todo $y \in N(A)$ es incidente en como mucho $k$ aristas (ya que al ser $G$ bipartito $N(A) \subseteq Y$). Llegamos así a que el número de vértices de $N(A)$ es al menos $k|A|/k = |A|$. Aplicando entonces el Teorema de Hall, $G$ tiene un emparejamiento completo de $X$ en $Y$. ■

**Corolario 1.13** *Si $G = (X \cup Y, E)$ es un grafo bipartito $k$-regular ($k \geq 1$), entonces $G$ tiene un emparejamiento perfecto.*

**Dem:** Si $G$ es $k$-regular, como $G$ es bipartito, entonces $|X| = |Y|$. Por ello, todo emparejamiento perfecto saturará $X$.
Sea $A \subseteq X$. El conjunto $A$ es incidente en $k|A|$, que pertenecen al conjunto de las $k|N(A)|$ aristas incidentes en $|N(A)|$. Por tanto, $k|A| \leq k|N(A)|$. Aplicando entonces el Teorema de Hall, llegamos a que $G$ tiene un emparejamiento completo que satura $X$, luego es perfecto. ■

Vamos a demostrar también en esta sección una serie de resultados interesantes acerca de los emparejamientos, como son el Teorema de la Deficiencia, el de König o el de Gallai. Para cada uno de ellos será necesario introducir algunos conceptos previos.

**Definición 1.14** *Se llama deficiencia de un grafo bipartito $G = (X \cup Y, E)$ al número entero $def(G) = \text{máx}\{|A| - |N(A)| \text{ tal que } A \subset X\}$. La deficiencia de un grafo siempre va a ser mayor o igual que cero (puesto que se puede considerar $A$ como el conjunto vacío).*

Dado un grafo $G$, denotamos al tamaño máximo de un emparejamiento de $G$ como el número $\mu(G)$. A partir de estas notaciones, podemos demostrar el siguiente teorema:

**Teorema 1.15 (Deficiencia)** *En un grafo bipartito $G = (X \cup Y, E)$ se tiene lo siguiente $\mu(G) = |X| - def(G)$.*

**Dem:** Para cualquier emparejamiento $M$, vamos a llamar $m$ al número de vértices de $X$ saturados por dicho emparejamiento, y $n$ al número de vértices de $X$ no saturados por el mismo, luego vamos a tener que $|X| = m + n$. Además, como existe algún conjunto $A$ de vértices en $X$ tal que $def(G) = |A| - |N(A)|$, tenemos que $|N(A)| = |A| - def(G)$. Es decir, que en cualquier emparejamiento hay al menos $def(G)$ vértices de $A$, y por tanto de $X$, que no pueden ser saturados por el emparejamiento ($n \geq def(G)$). Por lo tanto, el número máximo de vértices en $X$ que pueden ser saturados por un emparejamiento es

$$\mu(G) = m = |X| - n \leq |X| - def(G) \tag{1.1}$$

Veamos que existe un emparejamiento que satura exactamente $|X| - def(G)$ vértices de $X$:

- Caso I: Si $def(G) = 0$, se tiene que $|A| - |N(A)| \leq 0$ para todo subconjunto $A \subseteq X$. Por el Teorema de Hall, entonces existe un emparejamiento completo, y por tanto el número máximo de vértices en $X$ que pueden ser saturados por un emparejamiento es $\mu(G) = |X| = |X| - def(G)$.

- Caso II: Si $def(G) = d > 0$, entonces existe al menos un conjunto $A \subseteq X$ tal que $|A| > |N(A)|$. Construimos ahora un grafo $G'$ añadiendo $d$ vértices nuevos al conjunto $Y$ del grafo $G$, de forma que todos esos $d$ vértices estén unidos con todos los vértices de $X$ en $G'$.

  Veamos que en este nuevo grafo $G'$ se cumple el teorema de Hall, es decir, que $|A| \leq |N(A)|$ en $G'$ para todo $A \subseteq X$.

  Por reducción al absurdo, supongamos que existe un $A_0 \subseteq X$ tal que $|A_0| > |N(A_0)|$ en $G'$. Esto significa que $|A_0| - |N_{G'}(A_0)| > 0$. Como los $d$ vértices nuevos están en $N(A_0)$, entonces $|N_G(A_0)| = |N_{G'}(A_0)| - d$. Así, a lo que se llegaría sería a que $|A_0| - |N_G(A_0)| = |A_0| - |N_{G'}(A_0)| + d > d$. Esto por tanto implicaría que $def_G(A_0) > d = def(G)$, lo cual por definición de deficiencia no puede ser.

  Por lo tanto, el conjunto $A_0$ no puede existir en $G'$ y se cumple que $|A| \leq |N(A)|$ en $G'$ para todo $A \subseteq X$, luego, por el Teorema de Hall, $G'$ tiene un emparejamiento completo, $M$. Este emparejamiento $M$ empareja, como mucho, d vértices de $X$ con los vértices añadidos a $Y$, luego si restringimos el emparejamiento a $G$, $n \leq d$. Por consiguiente, este emparejamiento, en $G$ satura al menos $m = |X| - n \geq |X| - d$ vértices de $X$, que junto con la desigualdad (1.1), nos lleva a que dicho emparejamiento (el cual siempre existe por definición), es máximo y por tanto, $m = \mu(G) = |X| - d$.

■

A continuación vamos a demostrar, como ya adelantamos, el Teorema de König, el cual relaciona el tamaño máximo de un emparejamiento en un grafo bipartito con el tamaño mínimo de un recubrimiento por vértices de dicho grafo.

**Teorema 1.16 (König)** *Sea un grafo bipartito $G$. Entonces $\mu(G) = \tau(G)$.*

**Dem:** Sea $M$ un emparejamiento máximo de $G$ que tiene $\mu(G)$ aristas. Es claro que se necesita un vértice para cubrir cada arista del emparejamiento. Por tanto, necesitamos al menos $\mu(G)$ vértices para cubrir todas las aristas de $G$, luego $\tau(G) \geq \mu(G)$.

Ahora, nos basta con probar que se pueden cubrir todas las aristas de $G$ con $\mu(G)$ vértices. Por el Teorema de Deficiencia, $\mu(G) = |X| - def(G)$. Además, por la definición de deficiencia de un grafo, sabemos que existe un conjunto $A \subseteq X$ que cumple que $|A| - |N(A)| = def(A) = def(G) = |X| - \mu(G)$, luego, $\mu(G) = |X| - |A| + |N(A)|$. Sea el conjunto $C = (X \backslash A) \cup N(A)$. Este conjunto, por su definición, es un recubrimiento por vértices, ya que toda arista es incidente a vértices que o bien no están en $A$ (y estaría cubierta por $X \backslash A$), o bien están en $A$ (y están cubiertos por $N(A)$). Así, $\tau(G) \leq |C| = |X| - |A| + |N(A)| = \mu(G)$.

Por tanto, de las dos desigualdades, llegamos a que, efectivamente, $\mu(G) = \tau(G)$. ■

El último de los teoremas que habíamos mencionado era el Teorema de Gallai [8].
Este teorema relaciona el cardinal del conjunto de vértices de un grafo sin vértices aislados, por una parte con el tamaño mínimo de un recubrimiento por vértices y el tamaño máximo de un conjunto independiente, y por otra parte con el tamaño mínimo de un recubrimiento por aristas y el tamaño máximo de un emparejamiento.

**Teorema 1.17 (Gallai)** *Sea un grafo $G = (V, E)$ que no tiene vértices aislados. Entonces $\alpha(G) + \tau(G) = |V| = \rho(G) + \mu(G)$.*

**Dem:** Para la primera igualdad, como por definición un conjunto $S$ independiente cumple que $V \setminus S$ es un recubrimiento por vértices, entonces si tomamos un $S$ de cardinal máximo $\alpha(G)$, el cardinal de su complementario, es decir $|V \setminus S|$, sería $\tau(G)$, luego tenemos que $\alpha(G) + \tau(G) = |V|$.

Para la segunda igualdad razonaremos de la siguiente manera. Sea $M$ un emparejamiento de cardinal máximo $\mu(G)$. Para cada uno de los $|V| - 2|M|$ vértices $v \in V$ de $G$ no saturados por $M$, añadimos una arista a $M$ que incida sobre $v$ (sin importar cuál sea el otro extremo de estas aristas). Obtenemos así un recubrimiento por aristas de tamaño $|M| + (|V| - 2|M|) = |V| - |M| = |V| - \mu(G)$. Luego $\rho(G) \leq |V| - \mu(G)$.

Ahora, sea $F$ un recubrimiento mínimo por aristas, es decir, de tamaño $\rho(G)$. Para cada uno de los $v \in V$, eliminamos de $F$ $d_F(v) - 1$ aristas distintas cualesquiera incidentes en $v$ (entendiendo como $d_F(v)$ el grado del vértice $v$ siempre respecto del recubrimiento de aristas $F$ de partida). Notemos que aquellos vértices con $d_F(v) = 1$ supondrán eliminar cero aristas de $F$. Notemos también que para aquellos vértices con $d_F(v) > 1$ siempre se van a poder eliminar $d_F(v) - 1$ aristas del nuevo $F$ (es decir, no eliminadas anteriormente), ya que de no ser así, nuestro $F$ no sería un recubrimiento mínimo por aristas.
Obtendríamos así un emparejamiento de cardinal, por lo menos, $|F| - \sum_{v \in V}(d_F(v) - 1) = |F| - (2|F| - |V|) = |V| - |F| = |V| - \rho(G)$. Por tanto, $\mu(G) \geq |V| - \rho(G)$.

De las dos desigualdades llegamos entonces a que $|V| = \rho(G) + \mu(G)$. ■

**Corolario 1.18** *Sea un grafo bipartito $G$ que no tiene vértices aislados. Entonces se tiene que $\alpha(G) = \rho(G)$.*

**Dem:** La demostración es trivial a partir de los Teoremas de König y Gallai, ya que el primero de ellos nos lleva a que $\mu(G) = \tau(G)$, y el segundo (al no haber vértices aislados por hipótesis), concluye que $\alpha(G) + \tau(G) = \rho(G) + \mu(G)$, luego $\alpha(G) = \rho(G)$. ■

# Capítulo 2

# El Problema de los Matrimonios Estables

En este capítulo vamos a tratar el Problema de los Matrimonios Estables, para lo cual tenemos que introducir algunas notaciones específicas que vamos a usar a lo largo del capítulo.

Este problema se puede definir en términos de grafos, puesto que partimos de dos conjuntos disjuntos del mismo cardinal, $H = \{h_1, h_2, ..., h_n\}$ y $M = \{m_1, m_2, ..., m_n\}$, que conformarán el conjunto de vértices de nuestro grafo bipartito ($V = H \cup M$, con $|H| = |M|$). Además, cada vértice de $H$ y cada vértices de $M$ tiene su propia lista de preferencias en la que ordena a todos los vértice del conjunto al que no pertenece. Es decir, que cada vértice de $H$ tiene una ordenación en sus aristas incidentes en cada vértice de $M$, y viceversa. Resaltar que dicha ordenación no implica que la arista $h_i, m_j$ se encuentre en la lista de preferencias del vértice $h_i$ en la misma posición que en la lista de preferencias del vértice $m_j$.

Como observamos, en esta situación todos los vértices de una de las partes tienen una arista que incide sobre cada vértice de la otra parte del grafo, luego tenemos un grafo bipartito $G = (V = H \cup M, E)$ $n$-regular, más concretamente sería completo $K_{n,n}$. Por tanto, por el Corolario 1.12, existe un emparejamiento completo en $G$. A cada posible pareja del emparejamiento la denotaremos con el par no ordenado $(h_i, m_j)$.

En esta situación se puede llegar a tener un emparejamiento en el que ninguna pareja que no pertenezca al emparejamiento prefiera estar junta antes que con sus respectivas parejas del emparejamiento (con las preferencias de cada vértice).

El objetivo de este capítulo es poder encontrar un emparejamiento con las características expuestas en el párrafo anterior, y para ello, estudiaremos el Algoritmo de Gale-Shapley del que hablamos en la introducción, el cual nos proporcionará lo que buscamos.

Para tratar el Problema de los Matrimonios hacen falta dos conjuntos bien diferenciados, disjuntos entre sí, por lo que vamos a denotar a $H$ como un conjunto de hombres y a $M$ como un conjunto de mujeres, sin intención de excluir o discriminar los matrimonios entre personas del mismo sexo. Esto es simplemente para acotar el problema en cierta manera y evitar así confusión entre conjuntos.

Nos apoyaremos durante este capítulo esencialmente en [15], [9], [12], y [17].

## 2.1. El Algoritmo de Gale-Shapley

En 1962, Gale y Shapley proporcionaron un algoritmo que determina un emparejamiento en un grafo bipartito completo con preferencias en los vértices que además es estable.

Antes de introducir dicho algoritmo, definiremos en términos de grafos el concepto de estabilidad tan presente en este algoritmo, y a continuación, mostraremos un ejemplo de un emparejamiento estable y de otro que no lo es.

**Definición 2.1** *Sea un grafo bipartito compuesto por los dos conjuntos de hombres y mujeres $G = (H \cup M, E)$ y un emparejamiento $P$ del grafo. Se entiende por pareja bloqueante a aquellos dos vértices $h_i \in H$ y $m_j \in M$ tales que si $(h_i, m_i), (h_j, m_j)$ son parejas establecidas en $P$, entonces la arista $(h_i, m_j)$ está antes que la arista $(h_i, m_i)$ en la lista de preferencias de $h_i$, y la arista $(h_i, m_j)$ está antes que la arista $(h_j, m_j)$ en la lista de preferencias de $m_j$.*

En términos de matrimonios, el concepto de pareja bloqueante se puede formular de la siguiente manera. Si hay dos matrimonios $(h_i, m_i)$ y $(h_j, m_j)$ tales que $h_i$ prefiere estar antes con $m_j$ que con $m_i$, y $m_j$ prefiere estar antes con $h_i$ que con $h_j$, entonces $(h_i, m_j)$ es una pareja bloqueante, ya que tanto $h_i$ como $m_j$ se divorciarán de sus parejas para estar juntos.

En esta definición se basa el concepto clave de este capítulo, que es el de la estabilidad de un emparejamiento.

**Definición 2.2** *Se dice que un emparejamiento es estable si y sólo si no existen parejas bloqueantes. Además, se dice que un par $(h_i, m_j)$ es una pareja válida si existe un emparejamiento estable $M$ en el cual ambos vértices están juntos.*

Vamos a mostrar a continuación un ejemplo para asimilar mejor estos conceptos que hemos introducido.

**Ejemplo 2.3** *Sean los conjuntos $H = \{h_1, h_2, h_3\}$ y $M = \{m_1, m_2, m_3\}$ de hombres y mujeres respectivamente. Contamos también con la lista de preferencias de los hombres sobre las mujeres y de las mujeres sobre los hombres, mostradas a continuación:*

| | | | |
|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_3$ |
| $h_2$: | $m_1$ | $m_3$ | $m_2$ |
| $h_3$: | $m_2$ | $m_1$ | $m_3$ |

| | | | |
|---|---|---|---|
| $m_1$: | $h_1$ | $h_3$ | $h_2$ |
| $m_2$: | $h_3$ | $h_1$ | $h_2$ |
| $m_3$: | $h_3$ | $h_2$ | $h_1$ |

*En esta situación, el emparejamiento $(h_1, m_1), (h_2, m_3), (h_3, m_2)$ sería estable, ya que no hay parejas bloqueantes que prefieran estar juntas antes que con sus respectivas parejas. Así, los pares $(h_1, m_1), (h_2, m_3), (h_3, m_2)$ son todos ellos parejas válidas porque existe un emparejamiento estable, en este caso el formado por ellas tres juntas, para el cual son parejas.*

*Sin embargo, el emparejamiento $(h_1, m_2), (h_2, m_1), (h_3, m_3)$ no sería estable, ya que el par $(h_1, m_1)$ sería una pareja bloqueante, puesto que $h_1$ prefiere estar con $m_1$ antes que con $m_2$, y $m_1$ prefiere estar con $h_1$ antes que con $h_2$.*

Partiendo de los supuestos expuestos al comienzo del capítulo, los pasos del Algoritmo de Gale-Shapley serían los siguientes:

Al principio, todos los hombres y las mujeres están desemparejados.

- Paso 1: Cada $h_i$ no emparejado le propone ser su pareja al vértice $m_j$ que más prefiere según su ordenación (siempre que ese vértice no le haya rechazado ya).
- Paso 2: Cada vértice $m_j$ que haya recibido alguna propuesta en el paso anterior:
  - Si no estaba emparejado, se empareja con el vértice $h_i$ de los que tiene propuesta que más prefiera según su ordenación, y deja a los demás como vértices libres del emparejamiento.
  - Si ya estaba emparejado, se empareja con el vértice $h_i$ que más prefiere de entre todas sus propuestas (incluido su pareja actual), y deja a los demás como vértices libres del emparejamiento.
- Paso 3: Se repiten los pasos 1 y 2 hasta que todos los vértices $h_i$ estén emparejados con un vértice $m_j$.

**Observación 2.4** *Notemos que con los pasos anteriores del algoritmo:*

- *Cada vértice $h_i \in H$ se va intentando emparejar con los $m_j \in M$ en orden decreciente de preferencia.*
- *Este algoritmo siempre lleva al mismo emparejamiento, independientemente del orden de las proposiciones por parte del conjunto de hombres libres en cada iteración.*
- *Una vez que un vértice $m_j \in M$ es emparejado, nunca vuelve a estar desemparejado. Esto supone que el cardinal del conjunto de parejas formadas en cada iteración es siempre igual o superior al de la iteración anterior.*
- *Si un vértice $m_j$ está emparejado en una etapa, en las sucesivas etapas $m_j$ se emparejará con un vértice $h_i$ mejor o igual en su lista de preferencias. Es decir, que los emparejamientos de las mujeres van siempre a mejor respecto de sus listas de preferencias, mientras que los de los hombres van a peor.*

En los pasos definidos anteriormente, las elecciones iniciales las realizan los hombres $h_i \in H$, pero esto mismo se podría hacer a la inversa, siendo las mujeres $m_i \in M$ las proponentes. En ambos casos, se cumple la siguiente propiedad del algoritmo.

**Teorema 2.5** *El Algoritmo de Gale-Shapley acaba en un número finito de iteraciones.*

**Dem:** Si hay más de una iteración, entonces en cada iteración distinta de la última (en la cual todos los $h_i$ están emparejados puesto que ninguno habrá sido rechazado), al menos un $h_i \in H$ es rechazado por un $m_j \in M$, a la cual no volverá a hacer una proposición.
Entonces si cada conjunto $H$ y $M$ está formado por $n$ vértices, el algoritmo a lo sumo terminará en $n^2$ iteraciones, ya que cada $h_i \in H$ podría llegar a hacer $n$ proposiciones diferentes como mucho (en [13] se demuestra que en concreto el número de pasos máximo es $n^2 - n + 1$). ■

Para entender mejor el funcionamiento del algoritmo, mostraremos un ejemplo en el que partimos de un conjunto de 4 hombres y otro de 4 mujeres, cada uno con sus respectivas listas de preferencias, ordenadas de más preferido a menos, en dos tablas (preferencias de los hombres y preferencias de las mujeres).

**Ejemplo 2.6** *Sean los conjuntos $H = \{h_1, h_2, h_3, h_4\}$ y $M = \{m_1, m_2, m_3, m_4\}$, de hombres y mujeres respectivamente, y las listas de preferencias de ambos conjuntos mostradas a continuación:*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_3$ | $m_4$ |
| $h_2$: | $m_1$ | $m_2$ | $m_4$ | $m_3$ |
| $h_3$: | $m_1$ | $m_3$ | $m_2$ | $m_4$ |
| $h_4$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_1$ | $h_2$ | $h_3$ | $h_4$ |
| $m_2$: | $h_3$ | $h_2$ | $h_1$ | $h_4$ |
| $m_3$: | $h_2$ | $h_4$ | $h_1$ | $h_3$ |
| $m_4$: | $h_4$ | $h_3$ | $h_2$ | $h_1$ |

*En base a estas dos tablas, aplicamos entonces el Algoritmo de Gale-Shapley:*

- *Paso 1: Cada hombre $h_1, h_2, h_3$ y $h_4$, le propone matrimonio a su mujer preferida, en este caso todos a $m_1$.*
- *Paso 2: Cada mujer que haya recibido alguna propuesta acepta la del hombre que más prefiere y rechaza el resto, de forma que las parejas formadas hasta el momento serían:*

| | |
|---|---|
| $h_1$: | $m_1$ |
| $h_2$: | $\cancel{m_1}$ |
| $h_3$: | $\cancel{m_1}$ |
| $h_4$: | $\cancel{m_1}$ |

- *Paso 3: Como todavía quedan hombres sin emparejar, se vuelven a repetir los pasos 1 y 2.*
- *Paso 1: Ahora, cada hombre rechazado, $h_2, h_3$ y $h_4$, que ahora están sin emparejar, elige a la siguiente mujer de su lista, $m_2, m_3$ y $m_3$ respectivamente.*
- *Paso 2: Cada mujer que ha recibido propuesta vuelve a elegir su mejor opción de entre las que tiene (incluida su pareja actual en caso de que la tuviera), y rechaza a los demás:*

| | | |
|---|---|---|
| $h_1$: | $m_1$ | $m_1$ |
| $h_2$: | $\cancel{m_1}$ | $m_2$ |
| $h_3$: | $\cancel{m_1}$ | $\cancel{m_3}$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ |

- *Paso 3: Como todavía quedan hombres sin emparejar, se vuelven a repetir los pasos 1 y 2.*
- *Paso 1: De nuevo, cada hombre rechazado, que en este caso sólo es $h_3$, elige a la siguiente mujer de su lista, $m_2$.*

- *Paso 2: Como antes, cada mujer que ha recibido nueva propuesta, en este caso $m_2$, elige su mejor opción de entre las que tiene y rechaza a las demás:*

| | | | |
|---|---|---|---|
| $h_1$: | $m_1$ | $m_1$ | $m_1$ |
| $h_2$: | $\cancel{m_1}$ | $m_2$ | $\cancel{m_2}$ |
| $h_3$: | $\cancel{m_1}$ | $\cancel{m_3}$ | $m_2$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ | $m_3$ |

- *Paso 3: Como todavía quedan hombres sin emparejar, se vuelven a repetir los pasos 1 y 2.*

- *Paso 1: El único hombre rechazado ahora, $h_2$, elige a la siguiente mujer de su lista, $m_4$.*

- *Paso 2: Ahora, $m_4$ acepta por tanto la única propuesta que le ha llegado.*

- *Paso 3: Ya no hay hombres rechazados, luego hemos llegado a un emparejamiento en el que todos los hombres tienen pareja, y sería el siguiente:*

$$(h_1, m_1), (h_2, m_4), (h_3, m_2), (h_4, m_3)$$

*Como podemos observar, el emparejamiento que hemos encontrado entre ambos conjuntos es un emparejamiento estable, ya que ningún par $(h_i, m_j)$ distinto de los que conforman el emparejamiento prefieren estar juntos que con sus actuales parejas.*

*Si aplicásemos el algoritmo partiendo de que el conjunto de proponentes fuese el de las mujeres, llegaríamos al siguiente emparejamiento:*

- *Paso 1: Cada mujer $m_1, m_2, m_3$ y $m_4$, le propone matrimonio a su hombre preferido.*

- *Paso 2: Cada hombre que haya recibido alguna propuesta acepta la de la mujer que más prefiere y rechaza al resto, de forma que las parejas formadas serían:*

| | |
|---|---|
| $m_1$: | $h_1$ |
| $m_2$: | $h_3$ |
| $m_3$: | $h_2$ |
| $m_4$: | $h_4$ |

- *Paso 3: Como todas las mujeres están ya emparejadas, el algoritmo finaliza, dando lugar al emparejamiento:*

$$(m_1, h_1), (m_2, h_3), (m_3, h_2), (m_4, h_4)$$

*Podemos ver que este emparejamiento es distinto al generado antes, pero también es estable, ya que todas las mujeres están con sus hombres preferidos, luego no hay parejas bloqueantes.*

## 2.2. Resultados y demostraciones

Vamos a ver ahora una serie de resultados acerca del emparejamiento formado por el Algoritmo de Gale-Shapley y que demuestran su estabilidad. Empezaremos por definir ciertos conceptos ligados al término de emparejamiento en este contexto de grafos bipartitos $G = (H \cup M, E)$ con listas de preferencias, los cuales manejaremos en dichos resultados.

Un emparejamiento $P$ se dice que es óptimo para $H$ (también se puede adaptar para definirlo respecto al conjunto $M$, cambiando los $h_i$ por $m_i$ y recíprocamente), si se cumple que:

- $P$ es un emparejamiento estable.
- Todo $h_i \in H$ está emparejado con el $m_j \in M$ que más prefiere de entre todas las parejas válidas de $h_i$. O lo que es lo mismo, por la definición de pareja válida, en ningún otro emparejamiento estable $h_i$ prefiere antes a otro $m_k$ distinto del $m_j$ con el que está emparejado $P$.

Un emparejamiento $P$ se dice que es pésimo para $M$ (también se puede definir respecto al conjunto $H$, cambiando los $m_i$ por $h_i$ y recíprocamente), si se cumple que:

- $P$ es un emparejamiento estable.
- Todo $m_i \in M$ prefiere a cualquier otra pareja válida posible de $m_i$ antes que con la que está en $P$.

Una vez vista la formulación del algoritmo, nos surgen numerosas preguntas como: ¿existe siempre un emparejamiento perfecto? ¿Y estable? ¿Hay alguno que sea óptimo?

A continuación se demuestra que el Algoritmo de Gale-Shapley produce tanto un emparejamiento perfecto (aquel en el que todos los vértices están emparejados), como uno estable (aquel en el que no existen parejas bloqueantes o divorcios entre las parejas para obtener una pareja mejor).

**Teorema 2.7** *El emparejamiento generado por el Algoritmo de Gale-Shapley es un emparejamiento perfecto.*

**Dem:** Supongamos por reducción al absurdo que el algoritmo termina y existe un vértice $h_i \in H$ que no está emparejado con ningún vértice de $M$. Esto significaría que $h_i$ ha sido rechazado por todos los vértices de $M$. Para que un vértice $m_j \in M$ rechace a $h_i$, es necesario que tenga otro vértice de $H$ distinto de $h_i$ con el que emparejarse, lo cual implicaría que todo vértice de $M$ está emparejado. Pero entonces, por definición de emparejamiento y porque $|H| = |M|$, todo vértice de $H$ estaría también emparejado, lo cual se contradice con nuestra hipótesis de partida. Por tanto, al terminar el algoritmo, no puede haber ningún vértice sin emparejar, o lo que es lo mismo, el emparejamiento resultante es perfecto. ■

**Teorema 2.8** *El emparejamiento generado por el Algoritmo de Gale-Shapley es un emparejamiento estable.*

**Dem:** Sean $(h_{i_1}, m_{j_1})$ y $(h_{i_2}, m_{j_2})$ dos parejas del emparejamiento obtenido por el algoritmo. Supongamos por reducción al absurdo que $(h_{i_1}, m_{j_2})$ es una pareja bloqueante, es decir, que $h_{i_1}$ prefiere estar con $m_{j_2}$ antes que con $m_{j_1}$, y $m_{j_2}$ prefiere estar con $h_{i_1}$ antes que con $h_{i_2}$.

Entonces $h_{i_1}$ le habrá propuesto a $m_{j_2}$ antes que a $m_{j_1}$, y $m_{j_2}$ le habrá tenido que rechazar (para que este posteriormente haya podido proponer a su actual pareja $m_{j_1}$). Luego si $m_{j_2}$ ha rechazado a $h_{i_1}$ y finalmente ha quedado emparejado con $h_{i_2}$, eso significa que $m_{j_2}$ prefiere estar con $h_{i_2}$ antes que con $h_{i_1}$, lo cual se contradice con la hipótesis.

Por tanto, no puede existir ninguna pareja bloqueante, y entonces el emparejamiento resultante del algoritmo es estable. ∎

Se acaba de ver que el algoritmo produce un emparejamiento estable, pero dada una lista de preferencias, este no tiene por qué ser único, en el Ejemplo 2.6 se han visto dos emparejamientos distintos que son estables, dependiendo de qué conjunto elegía primero. En los distintos pasos del algoritmo, las parejas que se van formando no tienen por qué ser las definitivas, algunas peticiones se rechazan y otras no.

El siguiente resultado muestra que las parejas que sí podrían aparecer en algún emparejamiento estable no se rechazan.

**Teorema 2.9** *Al aplicar el Algoritmo de Gale-Shapley no se rechaza ninguna pareja válida.*

**Dem:** Sea $(h_{i_1}, m_{j_1})$ una pareja del emparejamiento generado por el algoritmo. Supongamos por reducción al absurdo que $m_{j_1}$ durante el algoritmo rechaza a $h_{i_2}$, siendo $(h_{i_2}, m_{j_1})$ la primera pareja válida rechazada, en la etapa $N_k$. En esta situación, tenemos que $m_{j_1}$ prefiere a $h_{i_1}$ antes que a $h_{i_2}$. Como $(h_{i_2}, m_{j_1})$ era una pareja válida, entonces existe un emparejamiento estable P en el que ambos vértices están emparejados.

Ahora, sea $m_{j_2}$ la pareja de $h_{i_1}$ en $P$, luego el par $(h_{i_1}, m_{j_2})$ sería también una pareja válida. Como partíamos de que $(h_{i_2}, m_{j_1})$ era la primera pareja válida rechazada, entonces necesariamente $h_{i_1}$ prefiere a $m_{j_1}$ antes que a $m_{j_2}$. Esto se debe a que de no ser así, $h_{i_1}$ habría hecho una proposición a $m_{j_2}$ en una etapa $N_l$ anterior a la etapa $N_k$ (en la que se rechazó por primera vez a una pareja válida). Como el algoritmo nos llevaba a que $h_{i_1}$ quedaba emparejado con $m_{j_1}$, entonces $m_{j_2}$ habría rechazado la proposición de $h_{i_1}$, pero hemos dicho que el par $(h_{i_1}, m_{j_2})$ era una pareja válida y no es la primera rechazada $(h_{i_1}, m_{j_2})$.

Esto nos lleva a una contradicción con el hecho de que $P$ era un emparejamiento estable, ya que el par $(h_{i_1}, m_{j_1})$ es una pareja bloqueante porque $h_{i_1}$ prefiere a $m_{j_1}$ antes que a su pareja $m_{j_2}$ en $P$, y $m_{j_1}$ prefiere a $h_{i_1}$ antes que a su pareja $h_{i_2}$ en $P$. Luego no es posible que el algoritmo rechace una pareja válida. ∎

En el Ejemplo 2.6 cada uno de los cuatro hombres se ha emparejado con la mujer que se encontraba en la posición 1, 3, 3 y 2 de sus listas de preferencias respectivamente, pero puede darse el caso de que alguno de los hombres se empareje con la última mujer de su lista. No obstante, como se muestra a continuación, este hecho sólo puede ocurrir para un único hombre.

**Teorema 2.10** *El Algoritmo de Gale-Shapley no permite que más de un hombre quede emparejado con la última opción de su lista de preferencias.*

**Dem:** Supongamos por reducción al absurdo que existen dos vértices $h_{i_1}$, $h_{i_2} \in H$ que quedan emparejados con su última opción. Supongamos también que $h_{i_1}$ quedó emparejado definitivamente en la etapa $N_k$ y que $h_{i_2}$ lo hizo en la etapa $N_l$, siendo $N_k \leq N_l$.

Sin embargo, en la etapa $N_k$ el vértice $h_{i_1}$, al quedar emparejado con su última opción, ha sido rechazado ya por las $n-1$ opciones restantes. Por el tercer punto de la Observación 2.4, esto significa que ya había $n-1$ parejas formadas, incluida, por tanto, la pareja de $h_{i_2}$, y que todas ellas serían ya las definitivas (puesto que $h_{i_1}$ se acaba de emparejar con su última opción y ya no quedarían vértices sin emparejar). Luego tendríamos que $N_l < N_k$, lo cual se contradice con la hipótesis de la que partíamos. ∎

Como hemos mostrado en la sección anterior, el Algoritmo de Gale-Shapley parte de una serie de propuestas lanzadas por uno de los conjuntos, por ejemplo el conjunto H de los hombres.
Además, como hemos visto en el Ejemplo 2.6, es un algoritmo no simétrico, ya que el conjunto de hombres no juega el mismo papel que el conjunto de mujeres. Los dos siguientes resultados hacen referencia a los efectos que esto supone sobre ambos conjuntos.

**Teorema 2.11** *El emparejamiento generado por el Algoritmo de Gale-Shapley es un emparejamiento óptimo para el conjunto $H$.*

**Dem:** Sea $(h_{i_1}, m_{j_1})$ una pareja resultante de aplicar el algoritmo. Supongamos por reducción al absurdo que $h_{i_1}$ prefiere a $m_{j_2}$ antes que a $m_{j_1}$ y que existe un emparejamiento estable $P$ del cual forma parte el par $(h_{i_1}, m_{j_2})$. En esta situación, durante el desarrollo del algoritmo, $m_{j_2}$ ha rechazado a $h_{i_1}$, lo cual es una contradicción con el Teorema 2.9, puesto que $(h_{i_1}, m_{j_2})$ era una pareja válida por hipótesis. Por tanto, no existe ninguna pareja válida para ningún vértice $h_i \in H$ con la cual dicho vértice estuviera más satisfecho que con la generada por el algoritmo. ∎

**Teorema 2.12** *El emparejamiento generado por el Algoritmo de Gale-Shapley es un emparejamiento pésimo para el conjunto $M$.*

**Dem:** Sea $P$ el emparejamiento resultante de aplicar el algoritmo y sea el par $(h_{i_1}, m_{j_1})$ perteneciente a dicho emparejamiento. Sea $P'$ otro emparejamiento estable tal que tiene por parejas $(h_{i_2}, m_{j_1})$ y $(h_{i_1}, m_{j_2})$, con $h_{i_2}$ distinto de $h_{i_1}$. Veamos que $m_{j_1}$ tiene en su lista de preferencias antes a $h_{i_2}$ que a $h_{i_1}$. Supongamos lo contrario, es decir, que $m_{j_1}$ prefiere antes a $h_{i_1}$ que a $h_{h_2}$. Por el Teorema 2.11, $P$ es un emparejamiento óptimo para $H$, luego $h_{i_1}$ prefiere a $m_{j_1}$ antes que a $m_{j_2}$. Así, llegaríamos a que el par $(h_{i_1}, m_{j_1})$ es una pareja bloqueante de $P'$, puesto que $h_{i_1}$ prefiere a $m_{j_1}$ antes que a $m_{j_2}$, y $m_{j_1}$ prefiere a $h_{i_1}$ antes que a $h_{i_2}$. Así, $P'$ no sería estable, lo cual contradice la hipótesis de sí lo era. Por tanto, siempre se tiene que en cualquier otro emparejamiento estable, $m_{j_1}$ tiene como pareja a otra anterior en su lista de preferencias. ∎

Podemos comprobar que estos dos últimos teoremas se cumplen para los dos emparejamientos obtenidos en el Ejemplo 2.6, es decir, para el generado por el algoritmo cuando los hombres eran los proponentes, $P = \{(h_1, m_1), (h_2, m_4), (h_3, m_2), (h_4, m_3)\}$, y para el generado cuando lo eran las mujeres, $P' = \{(m_1, h_1), (m_2, h_3), (m_3, h_2), (m_4, h_4)\}$.

En el caso de $P$, vemos que es óptimo para los hombres comparándolo con el otro emparejamiento estable $P'$, puesto que todos los hombres prefieren a su pareja en $P$ que a la que tienen en $P'$.

Por otra parte, $P$ es pésimo para las mujeres respecto a $P'$ porque por ejemplo vemos que $(m_3, h_4) \in P$, pero $(m_3, h_2)$ es una pareja válida (al pertenecer a un emparejamiento estable como es $P'$), y $m_3$ según su lista prefiere a $h_2$ antes que a $h_4$.

En el caso de $P'$, vemos que es óptimo para las mujeres en comparación con $P$, ya que todas las mujeres prefieren a su pareja en $P'$ que a la que tienen en $P$.

Por otro lado, $P'$ es pésimo para los hombres respecto a $P$, puesto que por ejemplo $(h_2, m_3) \in P'$ pero $(h_2, m_4)$ es una pareja válida (al pertenecer al emparejamiento estable $P$), y $h_2$ prefiere a $m_4$ antes que a $m_3$.

En resumen, si los hombres son el conjunto de proponentes, el Algoritmo de Gale-Shapley genera un emparejamiento estable que es el mejor para los hombres y el peor para las mujeres, y viceversa. Es por tanto algo normal tratar de buscar nuevos algoritmos con el objetivo de conseguir un mejor equilibrio entre proponentes y propuestos, es decir, que el resultado sea casi tan óptimo para los hombres como para las mujeres. Para ello, sin embargo, es necesario quitarle importancia a la estabilidad de los emparejamientos, llegando incluso a dejar de lado este concepto. Tomoko Fuku, Akira Namatame y Taisei Kaizouji propusieron unos algoritmos que resolvían estas ideas, y se tratan con más detalle en [4].

## 2.3. Variantes del problema

En el problema de los matrimonios estables cada hombre y cada mujer tenía una lista de preferencias que incluía y ordenaba de forma estricta a todas las personas del sexo opuesto.

En esta sección expondremos algunas variantes cercanas que existen del problema de los matrimonios estables. Estas versiones a veces se adaptan mejor a los objetivos o aplicaciones de la vida real cuando el número de elementos de cada conjunto disjunto es demasiado elevado, o cuando las listas de preferencias presentan algunas variaciones.

En concreto, vamos a ver el problema de los Matrimonios Estables con Listas Incompletas y con Listas con Indiferencias, en donde las listas de preferencias no están necesariamente completas u ordenadas estrictamente. En esta situación veremos que se pueden dar emparejamientos estables de diferentes tamaños. Estas variantes están estudiadas en [7], [12] y [11].

En primer lugar, vamos a estudiar la variante de las listas incompletas también conocida como la variante de las listas inaceptables. Este nombre se debe a que en las listas de preferencias de ciertas personas existen personas del sexo opuesto que son inaceptables (no aparecen en su lista), es decir, que son personas con las que por alguna razón no se quieren emparejar bajo ningún concepto y que por tanto las excluyen de su lista.

### 2.3.1. Listas Incompletas o Inaceptables

En el problema de las Listas de preferencias Incompletas partimos de los mismos conjuntos disjuntos que en el Problema de los Matrimonios Estables, hombres y mujeres

($H = \{h_1, h_2, ..., h_n\}$ y $M = \{m_1, m_2, ..., m_n\}$), que forman el conjunto de vértices del grafo bipartito ($V = H \cup M$, con $|H| = |M|$).

Sin embargo, la gran diferencia está en que la lista de preferencias de cada hombre y cada mujer puede estar incompleta, es decir, puede no incluir a determinadas personas del sexo opuesto con las que no quiera emparejarse de ninguna manera.

Aquellas mujeres $m_j \in M$ excluidas de la lista de preferencias de un hombre $h_i \in H$ serán no aceptables (o inaceptables) por $h_i$, y el par $(h_i, m_j)$ será una pareja inaceptable.

De forma análoga ocurre con aquellos hombres $h_i \in H$ que estén excluidos de la lista de preferencias de una mujer $m_j \in M$, diremos que son inaceptables por $m_j$, y el par $(h_i, m_j)$ será una pareja inaceptable.

Por esto, ningún emparejamiento puede contener una pareja inaceptable. Podemos notar que en esta situación no todos los vértices de una de las partes del grafo tienen una arista que incide sobre cada vértice de la otra parte, ya que algunos vértices puede que, por lo expuesto antes, no se puedan emparejar con ciertos vértices de la parte opuesta. Es por esto que el emparejamiento final no tiene por qué ser perfecto puesto que algunos vértices podrían quedar desemparejados, por ejemplo en el caso de que un hombre no tenga a ninguna mujer en su lista, de forma que se quedaría soltero.

Sin embargo, el hecho de que un posible emparejamiento pueda no ser perfecto no implica que este no pueda ser estable, si no que supone una modificación en la definición de lo que se entiende como emparejamiento estable para este tipo de problemas. Ahora un emparejamiento estable va a seguir siendo aquel en el cual no existen parejas bloqueantes, pero es en la definición de pareja bloqueante donde aparece un nuevo matiz.

En este contexto, un par $(h_i, m_j)$ se dice que es una pareja bloqueante del emparejamiento $P$ si no pertenece a $P$ y se cumple alguna de las siguientes condiciones (donde se contempla también la posibilidad de que alguno de los dos vértices haya quedado soltero cuando la pareja $(h_i, m_j)$ podía producirse):

- $h_i$ y $m_j$ prefieren estar juntos antes que con sus parejas en $P$.
- $h_i$ y $m_j$ están solteros en $P$ pero son una pareja aceptable entre sí.
- $h_i$ está soltero en $P$ pero $m_j$ está en su lista y $m_j$ prefiere antes a $h_i$ que a su pareja actual (o viceversa).

Las preguntas que nos surgen ahora serían: ¿Existe siempre un emparejamiento estable para esta variante del problema? ¿Existe algún algoritmo para llegar a un emparejamiento estable? ¿Cada emparejamiento estable empareja siempre el mismo subconjunto de hombres y mujeres?

Para poder responder a estas preguntas demostraremos primero un lema que utilizaremos posteriormente. Notemos además que un vértice del grafo prefiere un emparejamiento $P$ a otro $P'$ si dicho vértice prefiere a la pareja con la que está en $P$ que a la que está emparejado en $P'$.

**Lema 2.13** *Sea el grafo bipartito $G = (H \cup M, E)$ asociado a este tipo de problemas con listas de preferencias incompletas donde $|H| = |M| = n$, y sean $P$ y $P'$ dos emparejamientos estables de $G$. Si $(h_i, m_j) \in P$ y $(h_i, m_j) \notin P'$ entonces uno de los dos vértices prefiere $P$ antes que $P'$ y el otro prefiere $P'$ antes que $P$.*

**Dem:** Supongamos que $(h_{i_0}, m_{j_0}) \in P$ y $(h_{i_0}, m_{j_0}) \notin P'$. Entonces $h_{i_0}$ y $m_{j_0}$ no pueden preferir ambos a $P$ antes que a $P'$ puesto que entonces $P'$ no sería estable. Supongamos por tanto que ambos prefieren al emparejamiento $P'$ antes que a $P$, y supongamos que $(h_{i_0}, m_{j_1})$ y $(h_{i_1}, m_{j_0})$ están en $P'$. Entonces $h_{i_0}$ y $m_{j_1}$ no pueden preferir ambos $P'$ porque $P$ es estable, luego $m_{j_1}$ tiene que preferir a $P$. De la misma manera, $h_{i_1}$ tiene que preferir a $P$. Por tanto, $h_{i_1}$ y $m_{j_1}$ no pueden estar emparejados en $P$ ya que si no $P'$ no sería estable, así que supongamos que sus parejas en $P$ son $m_{j_2}$ y $h_{i_2}$, respectivamente. Por el mismo razonamiento de antes, $h_{i_2}$ y $m_{j_2}$ tienen que preferir a $P'$ pero $(h_{i_2}, m_{j_2})$ no puede pertenecer a $P'$, luego denotamos a sus parejas en $P'$ como $m_{j_3}$ y $h_{i_3}$, las cuales tienen que preferir a $P$ y por ello no pueden estar emparejados en $P$. Continuando este proceso indefinidamente obtenemos una secuencia infinita $h_{i_0}, m_{j_0}, h_{i_2}, m_{j_2}, h_{i_4}, m_{j_4}, \ldots$ de diferentes hombres y mujeres que prefieren a $P'$ antes que a $P$, y otra secuencia infinita $h_{i_1}, m_{j_1}, h_{i_3}, m_{j_3}, \ldots$ que prefieren a $P$ antes que a $P'$. Esto entra en contradicción con el hecho de que el conjunto de hombres y mujeres es finito. ■

Apoyándonos en este resultado que acabamos de probar vamos a demostrar que en esta variante con listas incompletas todo emparejamiento estable deja solteros o sin emparejar al mismo conjunto de hombres y mujeres.

**Teorema 2.14** *Sea $G = (H \cup M, E)$ en las condiciones del lema anterior. Entonces existen $H_0 \subset H$ y $M_0 \subset M$ tales que todo emparejamiento estable de $G$ deja sin emparejar los vértices de $H_0$ y $M_0$.*

**Dem:** Sean $P$ y $P'$ dos emparejamientos estables diferentes y sea $h_{i_0}$ un vértice de $H$ que está emparejado en $P$ pero no en $P'$. Sea $m_{j_0}$ la pareja de $h_{i_0}$ en $P$. Como $h_{i_0}$ claramente prefiere a $P$ antes que a $P'$ (ya que en uno está emparejado y en el otro ha quedado sin pareja), por el Lema 2.13 $m_{j_0}$ tiene que preferir a $P'$ antes que a $P$. Sea $h_{i_1}$ la pareja de $m_{j_0}$ en $P'$, entonces $h_{i_1}$ prefiere a $P$, luego su pareja en $P$, $m_{j_1}$, prefiere a $P'$ antes que a $P$. Continuando este proceso indefinidamente obtenemos una secuencia infinita de parejas $(h_{i_0}, m_{j_0}), (h_{i_1}, m_{j_1}), (h_{i_2}, m_{j_2}), \ldots$ que pertenecen a $P$, y otra secuencia $(h_{i_1}, m_{j_0}), (h_{i_2}, m_{j_1}), (h_{i_3}, m_{j_2}), \ldots$ que pertenecen a $P'$. Esto no puede ser ya que $H$ y $M$ son conjuntos finitos, luego si un vértice está emparejado en un emparejamiento estable también lo está en cualquier otro emparejamiento estable y viceversa. ■

Ahora vamos a ver cómo es posible encontrar un emparejamiento estable para este problema con listas incompletas haciendo dos modificaciones sobre el Algoritmo de Gale-Shapley.

En primer lugar, el algoritmo terminará cuando todos los proponentes estén emparejados, o bien cuando a ningún proponente no emparejado le queden más parejas aceptables a las que pedir matrimonio.
Y en segundo lugar, aquella persona que haya recibido alguna propuesta rechazará toda propuesta que provenga de un inaceptable para dicha persona.

Teniendo en cuenta estas dos modificaciones del Algoritmo de Gale-Shapley, la demostración de que dicho algoritmo siempre va a producir un emparejamiento estable (que en este caso puede ser completo o no serlo) es idéntica a la que probamos en la sección anterior.

A continuación, mostraremos un ejemplo que ilustra las principales diferencias entre el Problema de los Matrimonios Estables original y esta variante.

**Ejemplo 2.15** *Partimos de los conjuntos* $H = \{h_1, h_2, h_3, h_4\}$ *y* $M = \{m_1, m_2, m_3, m_4\}$ *de hombres y mujeres y de las siguientes listas de preferencias:*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_3$ | $m_4$ |
| $h_2$: | $m_1$ | $m_2$ | $m_3$ | |
| $h_3$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ |
| $h_4$: | $m_1$ | $m_3$ | $m_4$ | |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_3$ | $h_2$ | $h_4$ | |
| $m_2$: | $h_1$ | $h_4$ | $h_3$ | $h_2$ |
| $m_3$: | $h_4$ | $h_3$ | $h_1$ | $h_2$ |
| $m_4$: | $h_2$ | $h_1$ | | |

*Al ver estas listas de preferencias podemos notar que las listas de algunas personas están incompletas puesto que no contienen a los 4 miembros del sexo opuesto, lo cual implica que estas personas no quieren emparejarse con las personas excluidas de sus listas.*

*Esto genera una serie de parejas inaceptables que son por una parte las no aceptadas por los hombres:* $(h_2, m_4)$, $(h_4, m_3)$, *y por otra parte las no aceptadas por las mujeres:* $(h_1, m_1)$, $(h_3, m_4)$ *y* $(h_4, m_4)$.

*Además, en este ejemplo ningún emparejamiento estable va a poder ser perfecto puesto que el vértice* $m_4$ *nunca podrá ser emparejado. En consecuencia, otro vértice* $h_i \in H$ *tampoco, luego mínimo dos vértices quedarían libres de cualquier emparejamiento estable.*

*Esto se debe a que* $m_4$ *sólo acepta emparejarse con* $h_2$ *o con* $h_1$ *pero:*

- $(h_2, m_4)$ *es una pareja inaceptable.*
- *Si* $m_4$ *estuviera emparejado con* $h_1$ $(h_1, m_2)$ *sería una pareja bloqueante ya que* $h_1$ *prefiere a* $m_2$ *antes que a* $m_4$ *y* $m_2$ *prefiere a* $h_1$ *antes que a cualquier otro* $h_i$ *(donde se incluiría su actual pareja que no es* $h_1$*). Luego al existir una pareja bloqueante el emparejamiento no sería estable.*

*Si aplicamos el algoritmo con las dos modificaciones mencionadas antes quedaría de la siguiente manera.*

- *Etapa 1:*

| | |
|---|---|
| $h_1$: | $m_1$ |
| $h_2$: | $m_1$ |
| $h_3$: | $m_1$ |
| $h_4$: | $m_1$ |

$\Rightarrow$

| | |
|---|---|
| $h_1$: | $\cancel{m_1}$ |
| $h_2$: | $\cancel{m_1}$ |
| $h_3$: | $m_1$ |
| $h_4$: | $\cancel{m_1}$ |

- *Etapa 2:*

| | | |
|---|---|---|
| $h_1$: | $\cancel{m_1}$ | $m_2$ |
| $h_2$: | $\cancel{m_1}$ | $m_2$ |
| $h_3$: | $m_1$ | $m_1$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ |

$\Rightarrow$

| | | |
|---|---|---|
| $h_1$: | $\cancel{m_1}$ | $m_2$ |
| $h_2$: | $\cancel{m_1}$ | $\cancel{m_2}$ |
| $h_3$: | $m_1$ | $m_1$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ |

- *Etapa 3:*

| | | | |
|---|---|---|---|
| $h_1$: | $\cancel{m_1}$ | $m_2$ | $m_2$ |
| $h_2$: | $\cancel{m_1}$ | $\cancel{m_2}$ | $m_3$ |
| $h_3$: | $m_1$ | $m_1$ | $m_1$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ | $m_3$ |

$\Rightarrow$

| | | | |
|---|---|---|---|
| $h_1$: | $\cancel{m_1}$ | $m_2$ | $m_2$ |
| $h_2$: | $\cancel{m_1}$ | $\cancel{m_2}$ | $\cancel{m_3}$ |
| $h_3$: | $m_1$ | $m_1$ | $m_1$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ | $m_3$ |

- *Etapa 4: Como el único hombre que sigue sin pareja es $h_2$ y este ya ha propuesto a todas las mujeres de su lista habiendo sido rechazado por todas ellas, el algoritmo termina dando lugar al emparejamiento:*

$$P = \{(h_1, m_2), (h_3, m_1), (h_4, m_3)\}$$

*Podemos ver que $h_2$ queda soltero, al igual que $m_4$, como habíamos anticipado antes, y como no forman una pareja bloqueante puesto que son inaceptables entonces $P$ es un emparejamiento estable (no completo). De esta forma, en este ejemplo los conjuntos $H_0$ y $M_0$ del Teorema 2.14 son $H_0 = \{h_2\}$ y $M_0 = \{m_4\}$.*

Acabamos de ver por tanto cómo se adapta el Algoritmo de Gale-Shapley a esta variante. Sin embargo, se pueden alterar los datos de tal forma que podríamos aplicar el Algoritmo de Gale-Shapley directamente sin hacer ninguna modificación. Esto no siempre nos llevará a un emparejamiento estable pero más adelante veremos qué ventajas tiene este método.

Se trataría de añadir un hombre ficticio, $\tilde{h}$, que marque el límite entre las parejas aceptables y las inaceptables en las listas de cada mujer, y de la misma manera se añadiría una mujer ficticia en las listas de cada hombres, $\tilde{m}$. El ficticio $\tilde{h}$ se añadiría al final de la lista de cada mujer y a continuación, se añadirían los inaceptables para cada mujer en un orden arbitrario. Por tanto, cada mujer preferiría emparejarse con $\tilde{h}$ antes que con cualquier inaceptable de su lista original. De igual forma se procedería con el ficticio $\tilde{m}$ en las listas de los hombres. Por último, el ficticio $\tilde{h}$ preferirá antes a cualquier otra mujer que a $\tilde{m}$, y $\tilde{m}$ preferirá antes a cualquier hombre antes que a $\tilde{h}$ (siendo irrelevante el orden en el que se añadan a los hombres y mujeres reales). A este nuevo problema con una persona más de cada sexo nos vamos a referir como problema extendido del original, con su nuevo grafo asociado $G'$.

Aplicando este procedimiento al Ejemplo 2.15 podríamos ampliar las listas de preferencias como mostramos a continuación.

**Ejemplo 2.16** *Añadiendo los ficticios $\tilde{h}$ y $\tilde{m}$ a cada lista del sexo opuesto e incluyendo las propias listas de preferencias arbitrarias de estos dos ficticios podríamos llegar a las siguientes tablas:*

| | | | | | |
|---|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $\tilde{\mathbf{m}}$ |
| $h_2$: | $m_1$ | $m_2$ | $m_3$ | $\tilde{\mathbf{m}}$ | $m_4$ |
| $h_3$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ | $\tilde{\mathbf{m}}$ |
| $h_4$: | $m_1$ | $m_3$ | $m_4$ | $\tilde{\mathbf{m}}$ | $m_2$ |
| $\tilde{\mathbf{h}}$: | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $\tilde{\mathbf{m}}$ |

| | | | | | |
|---|---|---|---|---|---|
| $m_1$: | $h_3$ | $h_2$ | $h_4$ | $\tilde{\mathbf{h}}$ | $h_1$ |
| $m_2$: | $h_1$ | $h_4$ | $h_3$ | $h_2$ | $\tilde{\mathbf{h}}$ |
| $m_3$: | $h_4$ | $h_3$ | $h_1$ | $h_2$ | $\tilde{\mathbf{h}}$ |
| $m_4$: | $h_2$ | $h_1$ | $\tilde{\mathbf{h}}$ | $h_3$ | $h_4$ |
| $\tilde{\mathbf{m}}$: | $h_1$ | $h_2$ | $h_3$ | $h_4$ | $\tilde{\mathbf{h}}$ |

*Como podemos ver, hemos transformado el ejemplo original en un problema de matrimonios clásico con listas completas y con una persona más por cada sexo (5 hombres y 5 mujeres). Y si aplicamos el Algoritmo de Gale-Shapley a este nuevo problema llegaríamos a lo siguiente:*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $\cancel{m_1}$ | $m_2$ | $m_2$ | $m_2$ |
| $h_2$: | $\cancel{m_1}$ | $\cancel{m_2}$ | $\cancel{m_3}$ | $\tilde{m}$ |
| $h_3$: | $m_1$ | $m_1$ | $m_1$ | $m_1$ |
| $h_4$: | $\cancel{m_1}$ | $m_3$ | $m_3$ | $m_3$ |
| $\tilde{h}$: | $\cancel{m_1}$ | $\cancel{m_2}$ | $\cancel{m_3}$ | $m_4$ |

*Con el algoritmo llegamos al siguiente emparejamiento en el cual $\tilde{h}$ y $\tilde{m}$ no son pareja:*

$$P' = \{(h_1, m_2), (h_2, \tilde{m}), (h_3, m_1), (h_4, m_3), (\tilde{h}, m_4)\}$$

*Ahora eliminamos los ficticios y en este caso, el emparejamiento $P'$ restringido al problema original queda igual que el emparejamiento $P$ que hallamos en el ejemplo anterior, que ya dijimos que era estable (pero no completo), y en donde efectivamente, $H_0 = h_2$ y $M_0 = m_4$ quedaban solteros.*

Sin embargo, el hecho de añadir los ficticios $\tilde{h}$ y $\tilde{m}$ para aplicar Gale-Shapley directamente, aunque siempre nos va a proporcionar un emparejamiento estable para el problema extendido (ya que se trata de un problema de matrimonios clásico), no tiene por qué garantizarnos que sea ni completo ni estable para el problema restringido a las listas incompletas. En el siguiente ejemplo veremos un caso en el que ocurre esto y en el que además, cada uno de los dos métodos vistos nos lleva a emparejamientos distintos para un mismo problema de Listas Incompletas.

**Ejemplo 2.17** *Sean las siguientes listas de preferencias de 4 hombres y 4 mujeres:*

| | | | |
|---|---|---|---|
| $h_1$: | $m_4$ | $m_1$ | |
| $h_2$: | $m_1$ | $m_4$ | |
| $h_3$: | $m_3$ | $m_2$ | $m_1$ |
| $h_4$: | $m_1$ | $m_2$ | $m_4$ |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_1$ | $h_3$ | $h_4$ | $h_2$ |
| $m_2$: | $h_2$ | $h_3$ | | |
| $m_3$: | $h_4$ | $h_1$ | $h_2$ | |
| $m_4$: | $h_3$ | $h_2$ | | |

*Si aplicamos el Algoritmo de Gale-Shapley adaptado al problema de Listas Incompletas (sin añadir los ficticios) llegaríamos al siguiente emparejamiento estable no completo: $P_{L.I.} = \{(h_1, m_1), (h_2, m_4), (h_3, m_2)\}$, que deja solteros a $h_4$ y a $m_3$.*

*Si ahora añadimos los ficticios como hemos hecho antes y aplicamos directamente el Algoritmo de Gale-Shapley a las nuevas listas completas llegamos al siguiente emparejamiento $P'_{L.C.} = \{(h_1, m_1), (h_2, m_4), (h_3, m_3), (h_4, \tilde{m}), (\tilde{h}, m_2)\}$, el cual restringido al problema original con Listas Incompletas quedaría $P'_{L.I.} = \{(h_1, m_1), (h_2, m_4)\}$ puesto que se eliminan los ficticios y además, la pareja $(h_3, m_3)$ no es aceptable luego hay que romperla también.*

*Como vemos, $P_{L.I.} \neq P'_{L.I.}$ y además, a pesar de que $P'_{L.C.}$ lógicamente es estable (por haber aplicado Gale-Shapley a un problema de matrimonios clásicos con listas de preferencias completas), $P'_{L.I.}$ no lo es puesto que el par $(h_3, m_2)$ es una pareja bloqueante (porque no pertenece a $P'_{L.I.}$ y son aceptables mutuamente).*

Por tanto, hemos visto que el hecho de aplicar Gale-Shapley al problema extendido no nos garantiza que el emparejamiento resultante restringido al problema original sea estable. Vamos a probar entonces bajo qué condiciones un problema con listas incompletas tiene un emparejamiento estable completo, es decir, un emparejamiento estable en el que no quedan personas solteras. En el siguiente resultado se muestra la utilidad de añadir estos ficticios que es básicamente el poder determinar si existe un emparejamiento estable completo aplicando el Algoritmo de Gale-Shapley al problema extendido.

**Teorema 2.18** *Sea el grafo bipartito $G = (H \cup M, E)$ asociado a un problema de este tipo con listas incompletas, donde $|H| = |M| = n$. Sea $G'$ el grafo con $n+1$ hombres y $n+1$ mujeres correspondiente al proceso de añadir los dos ficticios $\tilde{h}$ y $\tilde{m}$ tal y como hemos detallado antes. Entonces, el grafo $G$ del problema original tiene un emparejamiento estable completo si y solo si el grafo $G'$ del problema ampliado tiene un emparejamiento estable donde $\tilde{h}$ y $\tilde{m}$ están emparejados.*

**Dem:** Supongamos primero que el grafo $G$ del problema original tiene un emparejamiento estable completo $P$. Entonces cada mujer en el problema ampliado prefiere a su pareja en $P$ antes que a $\tilde{h}$ porque al ser $P$ un emparejamiento estable completo, toda mujer está emparejada con una pareja aceptable (las cuales van antes que $\tilde{h}$ en las listas de preferencias ampliadas). De igual forma, cada hombre en el problema ampliado prefiere a su pareja en $P$ antes que a $\tilde{m}$. Añadiendo ahora la pareja $(\tilde{h},\, \tilde{m})$ a $P$ obtendríamos un emparejamiento estable para el problema ampliado.

Supongamos ahora que el problema ampliado tiene un emparejamiento estable $P$ que contiene a la pareja $(\tilde{h},\, \tilde{m})$ y sea $P' = P \backslash \{(\tilde{h}, \tilde{m})\}$. Supongamos que $(h_i, m_j) \in P'$. Si $h_i$ es inaceptable para $m_j$ entonces $m_j$ preferiría a $\tilde{h}$ antes que a $h_i$, pero como sabemos que $\tilde{h}$ prefiere a $m_j$ antes que a $\tilde{m}$ esto supondría que $P$ no es estable, lo cual entra en contradicción con la hipótesis. Por tanto, $h_i$ tiene que ser aceptable por $m_j$. De igual manera llegamos a que $m_j$ tiene que ser aceptable por $h_i$. Entonces llegamos a que $P'$ es un emparejamiento completo del problema original y que también es estable, puesto que $P$ lo es. ∎

### 2.3.2. Listas con Indiferencias

En el problema de las Listas de preferencias con Indiferencias partimos de los mismos conjuntos disjuntos de hombres $H = \{h_1, h_2, ..., h_n\}$ y mujeres $M = \{m_1, m_2, ..., m_n\}$, de forma que tenemos el conjunto de vértices del grafo bipartito ($V = H \cup M$, con $|H| = |M|$).

La diferencia en este caso radica en que las listas de preferencias no tienen por qué estar estrictamente ordenadas, pueden existir indiferencias entre algunas personas del sexo opuesto. Es decir, puede que algún $h_i$ no tenga ninguna preferencia sobre $m_{j_1}$ y $m_{j_2}$ o lo que es lo mismo, que le de igual emparejarse con $m_{j_1}$ o con $m_{j_2}$. Y lo mismo puede ocurrir en las listas de preferencias de las mujeres. Asumimos a lo largo de este apartado que las listas de preferencias vuelven a ser completas como en el problema de matrimonios original.

Al igual que en la sección anterior, nos surgen algunas preguntas esenciales sobre esta variante como son: ¿Existe siempre un emparejamiento estable para este problema? ¿Existe algún algoritmo para llegar a un emparejamiento estable? ¿Cada emparejamiento estable es perfecto?

En esta situación, se pueden diferenciar tres tipos de emparejamientos estables: súper-estables, fuertes y débiles. Todos ellos cumplen el hecho de que no existen parejas bloqueantes, sin embargo, cada tipo de estabilidad tiene asociada una definición ligeramente diferente de pareja bloqueante (cada una menos restrictiva que la anterior):

- En un emparejamiento súper-estable una pareja bloqueante será aquel par $(h_i, m_j)$ que no pertenezca al emparejamiento pero que $h_i$ prefiere a $m_j$ antes que a su actual pareja y $m_j$ prefiere a $h_i$ antes que a su actual pareja. Este tipo de estabilidad es el mismo del que hemos estado hablando a lo largo de todo el capítulo.
- En un emparejamiento estable de tipo fuerte una pareja bloqueante será aquel par $(h_i, m_j)$ que no pertenezca al emparejamiento y que cumpla que, $h_i$ prefiere a $m_j$ antes que a su pareja actual y a $m_j$ le resulta indiferente emparejarse con $h_i$ o con su pareja actual (o viceversa).
- En un emparejamiento estable de tipo débil una pareja bloqueante será aquel par $(h_i, m_j)$ que no pertenezca al emparejamiento y que cumpla que, a $h_i$ le es indiferente emparejarse con $m_j$ o con su pareja actual y a $m_j$ le es indiferente emparejarse con $h_i$ o con su pareja actual.

Si partimos de un problema de este tipo con listas de preferencia con indiferencias podemos llegar fácilmente a un problema de matrimonios clásico con listas de preferencias estrictamente ordenadas simplemente rompiendo cada indiferencia de las listas arbitrariamente y manteniendo, claro está, las preferencias estrictamente ordenadas. A este nuevo problema derivado del que partíamos con indiferencias le denominaremos problema estrictamente ordenado, en el cual sabemos que existe un emparejamiento estable y perfecto. Notar que romper las indiferencias de forma arbitraria da lugar como veremos más adelante a diferentes emparejamientos estables, según surjan de diferentes listas de preferencias estrictamente ordenadas.

Es trivial que si un emparejamiento es estable para el problema estrictamente ordenado también lo es para el problema original con indiferencias puesto que las preferencias estrictas iniciales se van a seguir cumpliendo por la manera en que hemos definido la transformación de los problemas.

En el siguiente teorema aclararemos esto y también veremos que todo emparejamiento estable del problema original con indiferencias puede obtenerse de esa forma.

**Teorema 2.19** *Sea el grafo bipartito $G = (H \cup M, E)$ asociado a un problema con listas de preferencias con indiferencias donde $|H| = |M| = n$. Entonces existe un emparejamiento súper-estable para dicho problema, y todo emparejamiento súper-estable del mismo es un emparejamiento estable de algún problema estrictamente ordenado derivado del problema original.*

**Dem:** Para la primera parte de la demostración, sea $P$ un emparejamiento estable de un problema estrictamente ordenado derivado del original (el cual existe porque ya hemos dicho que un problema derivado es un Problema de Matrimonios clásico). Si $h_i$ y $m_j$ no están emparejados en $P$, y en el problema original ambos vértices se prefieren mutuamente de forma estricta antes que a sus respectivas parejas en $P$, entonces también se preferirán mutuamente de manera estricta en el problema estrictamente ordenado. Esto contradice que $P$ sea estable para el problema estrictamente ordenado. Por tanto, $P$ es súper-estable para el problema original.
Para la segunda parte, sea $P'$ un emparejamiento súper-estable del problema original. Queremos construir por tanto un problema estrictamente ordenado del original para el cual $P'$ también sea estable. Procederemos de la siguiente manera. Si $(h_i, m_j) \in P'$ y $h_i$ siente indiferencia entre $m_j$ y $m_k$ en el problema original, entonces en el problema estrictamente ordenado pondremos que $h_i$ prefiere a $m_j$ antes que a $m_k$. Igualmente, si $m_j$ siente indiferencia entre $h_i$ y $h_l$ en el problema original entonces en el problema estrictamente ordenado pondremos que $m_j$ prefiere a $h_i$ antes que a $h_l$. Cualquier otra indiferencia del problema original que no se vea involucrada en $P'$ la romperemos de forma arbitraria para completar nuestro nuevo problema estrictamente ordenado derivado. Supongamos ahora que $h_0$ y $m_0$ no están emparejados en $P'$ pero se prefieren mutuamente en el problema estrictamente ordenado antes que a sus parejas en $P'$. Como $P'$ es súper-estable para el problema original entonces o bien $h_0$ siente indiferencia entre su pareja en $P'$ y $m_0$, o bien $m_0$ siente indiferencia entre su pareja en $P'$ y $h_0$, o ambos. Llegamos en cualquiera de los casos a una contradicción debido a la construcción del problema estrictamente ordenado. Por tanto, según este problema derivado que hemos generado no puede haber parejas bloqueantes en $P'$ y $P'$ es un emparejamiento estable (según la definición de estabilidad de un Problema de Matrimonios clásico como lo es un problema derivado). ■

Una vez probado el teorema anterior podemos notar que para esta variante del Problema del Matrimonio con Listas con Indiferencias, el Algoritmo de Gale-Shapley no necesita de ninguna modificación una vez determinado un problema estrictamente ordenado. El algoritmo puede generar diferentes emparejamientos estables para distintos problemas estrictamente ordenados derivados del original (incluso si se toma el mismo conjunto de proponentes).

A continuación, vamos a ejemplificar todas estas ideas sobre este tipo de problemas con indiferencias en las listas de preferencia para así clarificar también las distintas versiones de estabilidad de las que hemos hablado.

**Ejemplo 2.20** *Sean los conjuntos* $H = \{h_1, h_2, h_3, h_4\}$ *y* $M = \{m_1, m_2, m_3, m_4\}$ *de hombres y mujeres y sean también las siguientes listas de preferencias con indiferencias:*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$<br>$m_4$ | $m_3$ | |
| $h_2$: | $m_2$<br>$m_4$ | $m_1$ | $m_3$ | |
| $h_3$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ |
| $h_4$: | $m_2$ | $m_4$ | $m_1$<br>$m_3$ | |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_3$ | $h_4$ | $h_1$ | $h_2$ |
| $m_2$: | $h_2$<br>$h_4$ | $h_1$ | $h_3$ | |
| $m_3$: | $h_1$<br>$h_2$ | $h_4$ | $h_3$ | |
| $m_4$: | $h_2$ | $h_1$ | $h_3$ | $h_4$ |

*Como vemos en las tablas de preferencias anteriores hay algunas personas que sienten indiferencia sobre otras (las que aparecen una sobre otra en la misma posición).*

*En el caso de los hombres esto ocurre con* $h_1$ *a quien le resulta indiferente emparejarse con* $m_2$ *o con* $m_4$*; con* $h_2$*, que le da igual emparejarse con* $m_2$ *o con* $m_4$*; y con* $h_4$*, que le da lo mismo emparejarse con* $m_1$ *o con* $m_3$*. En el caso de las mujeres a* $m_2$ *le es indiferente emparejarse con* $h_2$ *o con* $h_4$*, y a* $m_3$ *le da igual emparejarse con* $h_1$ *o con* $h_2$*.*

*En base a este ejemplo vamos a trabajar el concepto de pareja bloqueante para los diferentes tipos de estabilidades de un emparejamiento. Para ello, partiremos de dos emparejamientos diferentes.*

*El primero de los emparejamientos que vamos a analizar es:*

$$P = \{(h_1, m_1), (h_2, m_4), (h_3, m_3), (h_4, m_2)\}$$

*En este emparejamiento podemos comprobar que los siguientes pares son parejas bloqueantes para un determinado tipo de estabilidad:*

- $(h_3, m_1)$ *es una pareja bloqueante para la súper-estabilidad de* $P$ *ya que* $h_3$ *prefiere a* $m_1$ *antes que a su pareja* $m_3$ *y* $m_1$ *prefiere a* $h_3$ *antes que a su pareja* $h_1$*.*
- $(h_2, m_2)$ *es una pareja bloqueante para la estabilidad débil de* $P$ *ya que* $h_2$ *siente indiferencia entre* $m_2$ *y su pareja* $m_4$ *y a* $m_2$ *le resulta indiferente emparejarse con* $h_2$ *o con su pareja* $h_4$*.*

*Sin embargo, para la estabilidad de tipo fuerte de* $P$ *no existen parejas bloqueantes, de hecho, ningún par* $(h_i, m_j) \notin P$ *distinto de* $(h_3, m_1)$ *y de* $(h_2, m_2)$ *es bloqueante para ningún tipo de estabilidad. Esto se debe a que en todos ellos* $h_i$ *prefiere estrictamente a su pareja en* $P$ *antes que a* $m_j$ *o viceversa, luego no suponen ningún bloqueo.*

*El segundo de los emparejamientos que vamos a analizar es:*

$$P' = \{(h_1, m_4), (h_2, m_2), (h_3, m_3), (h_4, m_1)\}$$

*En este emparejamiento podemos comprobar que los siguientes pares son parejas bloqueantes para un determinado tipo de estabilidad:*

- $(h_2, m_4)$ *es una pareja bloqueante para la estabilidad fuerte de* $P'$ *ya que* $m_4$ *prefiere a* $h_2$ *antes que a su pareja* $h_1$ *y* $h_2$ *siente indiferencia entre* $m_4$ *y su pareja* $m_2$.
- $(h_4, m_2)$ *es una pareja bloqueante para la estabilidad fuerte de* $P'$ *ya que* $h_4$ *prefiere a* $m_2$ *antes que a su pareja* $m_1$ *y* $m_2$ *siente indiferencia entre* $h_4$ *y su pareja* $h_2$.
- $(h_4, m_3)$ *es una pareja bloqueante para la estabilidad fuerte de* $P'$ *ya que* $m_3$ *prefiere a* $h_4$ *antes que a su pareja* $h_3$ *y* $h_4$ *siente indiferencia entre* $m_3$ *y su pareja* $m_1$.
- $(h_3, m_1)$ *es una pareja bloqueante para la súper-estabilidad de* $P'$ *ya que* $h_3$ *prefiere a* $m_1$ *antes que a su pareja* $m_3$ *y* $m_1$ *prefiere a* $h_3$ *antes que a su pareja* $h_4$.

*Sin embargo, para la estabilidad de tipo débil de* $P'$ *no existen parejas bloqueantes, de hecho, ningún par* $(h_i, m_j) \notin P'$*, distinto de los cuatro pares analizados, es bloqueante para ningún tipo de estabilidad porque en todos ellos* $h_i$ *prefiere estrictamente a su pareja en* $P$ *antes que a* $m_j$ *o viceversa, luego no suponen ningún bloqueo.*

*Por último, vamos a comprobar que efectivamente, como ya habíamos adelantado antes del ejemplo, si aplicamos el Algoritmo de Gale-Shapley a diferentes problemas estrictamente ordenados derivados del original del ejemplo no siempre nos genera los mismos emparejamientos estables.*

*(1)*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_4$ | $m_3$ |
| $h_2$: | $m_2$ | $m_4$ | $m_1$ | $m_3$ |
| $h_3$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ |
| $h_4$: | $m_2$ | $m_4$ | $m_3$ | $m_1$ |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_3$ | $h_4$ | $h_1$ | $h_2$ |
| $m_2$: | $h_2$ | $h_4$ | $h_1$ | $h_3$ |
| $m_3$: | $h_1$ | $h_2$ | $h_4$ | $h_3$ |
| $m_4$: | $h_2$ | $h_1$ | $h_3$ | $h_4$ |

*(2)*

| | | | | |
|---|---|---|---|---|
| $h_1$: | $m_1$ | $m_2$ | $m_4$ | $m_3$ |
| $h_2$: | $m_4$ | $m_2$ | $m_1$ | $m_3$ |
| $h_3$: | $m_1$ | $m_3$ | $m_4$ | $m_2$ |
| $h_4$: | $m_2$ | $m_4$ | $m_3$ | $m_1$ |

| | | | | |
|---|---|---|---|---|
| $m_1$: | $h_3$ | $h_4$ | $h_1$ | $h_2$ |
| $m_2$: | $h_2$ | $h_4$ | $h_1$ | $h_3$ |
| $m_3$: | $h_1$ | $h_2$ | $h_4$ | $h_3$ |
| $m_4$: | $h_2$ | $h_1$ | $h_3$ | $h_4$ |

*En el primer problema de matrimonios clásico el algoritmo nos genera el emparejamiento:*

$$P_1 = \{(h_1, m_4), (h_2, m_2), (h_3, m_1), (h_4, m_3)\}$$

*Los únicos pares* $(h_i, m_j) \notin P_1$ *que son parejas bloqueantes de* $P_1$ *para algún tipo de estabilidad son* $(h_2, m_4)$ *y* $(h_4, m_2)$*, que lo son para la estabilidad fuerte. En todos los demás pares* $h_i$ *prefiere estrictamente a su pareja en* $P$ *antes que a* $m_j$ *o viceversa. Por tanto,* $P_1$ *es un emparejamiento tanto súper-estable como estable de tipo débil ya que no tiene ninguna pareja bloqueante para estos dos tipos de estabilidad.*

*Sin embargo, en el segundo problema de matrimonios clásico, que se diferencia del primero únicamente en que* $h_2$ *prefiere antes a* $m_4$ *que a* $m_2$*, el algoritmo genera un emparejamiento distinto al anterior:*

$$P_2 = \{(h_1, m_3), (h_2, m_4), (h_3, m_1), (h_4, m_2)\}$$

*En este caso el único par* $(h_i, m_j) \notin P_2$ *que es pareja bloqueante de* $P_1$ *para algún tipo de estabilidad es* $(h_2, m_2)$*, que lo son para la estabilidad débil. Por tanto,* $P_2$ *es un emparejamiento tanto súper-estable como estable de tipo fuerte ya que no tiene ninguna pareja bloqueante para estos dos tipos de estabilidad.*

# Capítulo 3

# Hospitales y Residentes ó Admisión en Universidades

Una de las aplicaciones más prácticas del Problema del Matrimonio es el Problema de los Hospitales y Residentes, también conocido como el Problema de la Admisión en Universidades.

En este problema de asignación se parte de dos conjuntos disjuntos, al igual que en el capítulo anterior, pero en este caso se puede entender el conjunto $E = \{e_1, e_2, ..., e_n\}$ como estudiantes y el conjunto $U = \{u_1, u_2, ..., u_m\}$ como universidades, donde el conjunto de estudiantes es mayor que el de universidades disponibles. Ambos conjuntos conforman el conjunto de vértices de nuestro grafo bipartito, $V = E \cup U$ donde $|E| \geq |U|$ $(n \geq m)$.

Cada estudiante tiene su propia lista de preferencias estricta sobre las universidades a las que le gustaría ir, pudiendo esta ser no completa, mientras que las universidades por su parte tienen estrictamente ordenados a todos los estudiantes según sus calificaciones de acceso, no pudiendo estos tener exactamente la misma nota.

Además las universidades, al igual que los estudiantes, van a poder presentar cada una diferentes listas de preferencias de los estudiantes a aceptar dado que dichas preferencias pueden depender no solo de la nota de los estudiantes, sino también de los requisitos de acceso que imponga cada universidad. Sin ir más lejos, en el sistema de acceso a los grados universitarios en España los diferentes grados no suelen tener la misma ordenación de alumnos. Esto se debe a que cada alumno tiene su nota pero al ser una nota puntuada sobre 14 en vez de sobre 10, la ponderación de una asignatura no tiene por que ser la misma para un grado que para otro y por tanto la nota final de acceso de un alumno puede variar de un grado a otro.

Por otro lado, la principal diferencia de este problema con el del capítulo anterior, además del cardinal de los conjuntos de vértices, es que cada universidad puede aceptar a varios estudiantes a la vez. Para controlar la capacidad de las universidades cada universidad determina una cuota o límite de plazas que pueden ser ocupadas por los estudiantes, las cuales designaremos con $\{q_1, q_2, ..., q_m\} \subset \mathbb{N}$.

De esta forma en términos de grafos, un par $(e_i, u_j)$ es una pareja aceptable si $u_j$ aparece en la lista de preferencias de $e_i$. Además, se definiría como emparejamiento $P$ del grafo $G$ a un conjunto de aristas que unen parejas aceptables, donde varias de estas aristas pueden incidir en un mismo vértice $u_j \in U$ siempre y cuando no excedan la cuota $q_j$ de cada universidad.
En este capítulo nos basaremos principalmente [5], [12], [16], y [1].

## 3.1. Algoritmo y resultados

En esta sección vamos a exponer un algoritmo propuesto en 1984 por Alvin Elliot Roth, que es una adaptación del de Gale-Shapley para este tipo de problema con el objetivo de encontrar un emparejamiento estable. También mostraremos un ejemplo aplicando el mismo.

La estabilidad de un emparejamiento $P$ en este problema parte de la definición de pareja bloqueante para $P$, la cual es un par $(e_i, u_j)$ que no pertenece a $P$ y que cumple las tres condiciones siguientes:

- $u_j$ está en la lista de preferencias de $e_i$.
- $u_j$ o no ha cubierto el cupo máximo de estudiantes o prefiere a $e_i$ antes que a otro estudiante de $E$ al que ya ha aceptado dicha universidad.
- $e_i$ o no ha sido aceptado por ninguna universidad o prefiere a $u_j$ antes que a la universidad en la que ha sido aceptado.

De esta forma un emparejamiento estable será aquel para el cual no exista ninguna pareja bloqueante.

El algoritmo propio para este problema es muy similar al del problema del matrimonio con listas incompletas pero teniendo en cuenta el concepto de cuota de las universidades y el hecho de que las únicas listas que pueden ser incompletas son las de los estudiantes. Los pasos a seguir seguirían los siguientes pasos:

- Paso 1: Cada estudiante que no tenga una universidad asignada manda una solicitud a la mejor universidad según su lista ordenada de preferencias (siempre que esa universidad no le haya rechazado ya).
- Paso 2: Cada universidad ordena, según sus preferencias, a todos aquellos estudiantes de los que haya recibido solicitud o hubiera aceptado anteriormente. Acepta a todos ellos de forma ordenada y si se alcanza el cupo de estudiantes que pueden ser aceptados en la universidad, se rechaza a los menos deseados.
- Paso 3: Se repiten los pasos 1 y 2 hasta que todos los estudiantes han sido admitidos en una universidad o han sido rechazados por todas ellas.

Este algoritmo vuelve a generar como en el capítulo anterior un emparejamiento estable (siguiendo la definición de estabilidad que hemos dado al principio de la sección), y es lo que vamos a demostrar el siguiente teorema.

**Teorema 3.1** *El emparejamiento generado por el algoritmo recién detallado es un emparejamiento estable.*

**Dem:** Sea $P$ el emparejamiento obtenido por el algoritmo. Supongamos por reducción al absurdo que $(e_{i_1}, u_{j_1})$ es una pareja bloqueante, es decir, que el estudiante $e_{i_1}$ no ha sido aceptado en la universidad $u_{j_1}$ (la cual está en su lista de preferencias) y se cumplen una de las siguientes condiciones:

(1) $u_{j_1}$ no ha cubierto el cupo máximo de estudiantes, y $e_{i_1}$ no ha sido aceptado por ninguna universidad.

(2) $u_{j_1}$ no ha cubierto el cupo máximo de estudiantes, y $e_{i_1}$ prefiere a $u_{j_1}$ antes que a la universidad en la que ha sido aceptado en $P$, $u_{j_2}$.

(3) $u_{j_1}$ prefiere a $e_{i_1}$ antes que a un estudiante admitido en $P$, $e_{i_2}$, y $e_{i_1}$ no ha sido aceptado por ninguna universidad.

(4) $u_{j_1}$ prefiere a $e_{i_1}$ antes que a un estudiante admitido en $P$, $e_{i_2}$, y $e_{i_1}$ prefiere a $u_{j_1}$ antes que a la universidad en la que ha sido admitido en $P$, $u_{j_2}$.

La opción (1) no puede darse puesto que si $u_{j_1}$ no ha cubierto todas sus plazas el algoritmo no ha podido rechazar a $e_{i_1}$, luego este no debería haberse quedado sin plaza en $u_{j_1}$.
De igual forma la opción (2) no puede ocurrir porque si $e_{i_1}$ prefiere a $u_{j_1}$ antes que a la universidad que le ha aceptado en $P$, $u_{j_2}$, al aplicar el algoritmo $e_{i_1}$ le habría pedido solicitud primero a $u_{j_1}$ y este al no haber cubierto todas sus plazas disponibles habría aceptado dicha solicitud.
Por su parte, la opción (3) tampoco puede darse ya que si $e_{i_1}$ no ha sido aceptado por ninguna universidad eso quiere decir que al aplicar el algoritmo $u_{j_1}$ ha preferido admitir a $e_{i_2}$ antes que a $e_{i_1}$, lo cual contradice la hipótesis de (3).
En cuanto a la opción (4), si $e_{i_1}$ prefiere a $u_{j_1}$ antes que a la universidad en la que ha sido admitido en $P$, $u_{j_2}$, al aplicar el algoritmo $e_{i_1}$ le habría pedido solicitud primero a $u_{j_1}$ y esta universidad hemos dicho que finalmente le ha rechazado, por lo que en alguna etapa del algoritmo prefirió al estudiante $e_{i_2}$ al que admitió antes que a $e_{i_1}$, lo cual contradice la hipótesis de (4). Por lo que esta opción tampoco puede ocurrir.

Por lo tanto, no puede existir ninguna pareja bloqueante y entonces el emparejamiento resultante del algoritmo es estable. ∎

A continuación mostramos un ejemplo en el que se aplica el algoritmo descrito antes.

**Ejemplo 3.2** *Sean los conjuntos de estudiantes y universidades* $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ *y* $U = \{u_1, u_2, u_3\}$ *respectivamente. Además, las tres universidades tiene un cupo máximo de estudiantes admitidos. En el caso de* $u_1$ *sería* $q_1 = 3$*, en el caso de* $u_2$ *sería* $q_2 = 2$*, y para* $u_3$ *sería* $q_3 = 2$*. Y cada estudiante y cada universidad tiene su lista de preferencias que serían las siguientes.*

| | | | |
|---|---|---|---|
| $e_1$: | $u_1$ | $u_2$ | $u_3$ |
| $e_2$: | $u_1$ | $u_2$ | |
| $e_3$: | $u_2$ | $u_1$ | $u_3$ |
| $e_4$: | $u_1$ | $u_3$ | $u_2$ |
| $e_5$: | $u_1$ | $u_2$ | |
| $e_6$: | $u_1$ | $u_2$ | $u_3$ |

| | | | | | | |
|---|---|---|---|---|---|---|
| $u_1$: | $e_3$ | $e_2$ | $e_1$ | $e_5$ | $e_6$ | $e_4$ |
| $u_2$: | $e_6$ | $e_2$ | $e_1$ | $e_4$ | $e_5$ | $e_3$ |
| $u_3$: | $e_1$ | $e_4$ | $e_5$ | $e_6$ | $e_2$ | $e_4$ |

*Aplicamos entonces el algoritmo expuesto anteriormente:*

- *Paso 1: Cada estudiante* $e_1, e_2, e_3, e_4, e_5$ *y* $e_6$ *solicita entrar en su universidad preferida según su lista, en este caso todos a* $u_1$ *menos* $e_3$ *que se lo solicita a* $u_2$*.*

- *Paso 2: Cada universidad que haya recibido alguna solicitud acepta a tantos estudiantes pueda, según su lista de preferencias, hasta alcanzar su cupo máximo, punto a partir del cual rechazará el resto de solicitudes menos deseadas. De esta forma,* $u_1$ *rechazaría a* $e_j$ *ya que* $q_1 = 3$*:*

| | |
|---|---|
| $e_1$: | $u_1$ |
| $e_2$: | $u_1$ |
| $e_3$: | $u_2$ |
| $e_4$: | $u_1$ |
| $e_5$: | $\cancel{u_1}$ |
| $e_6$: | $\cancel{u_1}$ |

- *Paso 3: Como todavía quedan estudiantes sin universidad asignada se vuelven a repetir los pasos 1 y 2.*

- *Paso 1: Los estudiantes sin universidad asignada,* $e_5$ *y* $e_6$*, solicitan entrar en su siguiente mejor universidad según su lista, en este caso ambos a* $u_2$*.*

- *Paso 2: Ahora como* $u_2$ *prefiere a* $e_5$ *y a* $e_6$ *antes que a* $e_3$*, acepta a los dos primeros y rechaza a este último ya que* $q_2 = 2$*:*

| | | |
|---|---|---|
| $e_1$: | $u_1$ | $u_1$ |
| $e_2$: | $u_1$ | $u_1$ |
| $e_3$: | $u_2$ | $\cancel{u_2}$ |
| $e_4$: | $u_1$ | $u_1$ |
| $e_5$: | $\cancel{u_1}$ | $u_2$ |
| $e_6$: | $\cancel{u_1}$ | $u_2$ |

- *Paso 3: Como* $e_3$ *no tiene universidad asignada se vuelven a repetir los pasos 1 y 2.*

- *Paso 1:* $e_3$ *solicita entrar en su siguiente mejor universidad según su lista,* $u_1$*.*

- *Paso 2: Ahora como* $u_1$ *prefiere a* $e_3$ *antes que a cualquier otro estudiante, acepta a* $e_3$ *y rechaza* $e_4$ *ya que es el menos deseado por* $u_1$ *y* $q_1 = 3$*:*

| | | | |
|---|---|---|---|
| $e_1$: | $u_1$ | $u_1$ | $u_1$ |
| $e_2$: | $u_1$ | $u_1$ | $u_1$ |
| $e_3$: | $u_2$ | $\cancel{u_2}$ | $u_1$ |
| $e_4$: | $u_1$ | $u_1$ | $\cancel{u_1}$ |
| $e_5$: | $\cancel{u_1}$ | $u_2$ | $u_2$ |
| $e_6$: | $\cancel{u_1}$ | $u_2$ | $u_2$ |

- *Paso 3: Como* $e_4$ *no tiene universidad asignada se vuelven a repetir los pasos 1 y 2.*

- *Paso 1:* $e_4$ *solicita entrar en su siguiente mejor universidad según su lista,* $u_3$*.*

- *Paso 2: Como* $u_3$ *sigue teniendo plazas disponibles (*$q_3 = 2$*) acepta a* $e_4$*.*

- *Paso 3: Como todos los estudiantes ya tienen una universidad asignada el algoritmo finaliza, dando lugar al siguiente emparejamiento de admisión en las 3 universidades:* $P = \{(e_1, u_1),\ (e_2, u_1),\ (e_3, u_1),\ (e_4, u_3),\ (e_5, u_2)$ *y* $(e_6, u_2)\}$

*Como podemos ver,* $u_1$ *y* $u_2$ *han cubierto todas sus plazas, 3 y 2 respectivamente, mientras que* $u_3$ *se ha quedado con una plaza libre.*

El problema de la asignación de estudiantes a universidades se puede plantear también de otras maneras. Hasta ahora cada universidad $u_i$ tenía una cuota $q_i \geq 1$ por la cual no admitía más de $q_i$ estudiantes. Por tanto, el número de plazas totales a ocupar por los alumnos era $\sum q_i$. De esta forma si este sumatorio es mayor que el número de estudiantes algunas universidades quedarían con plazas libres mientras que si es inferior al número de estudiantes serían algunos de estos los que se quedarían sin plazas disponibles.

En [2] se enfoca el problema desde un punto de vista mucho más similar al del problema del matrimonio. Se trata de introducir universidades ficticias adicionales. En concreto, el nuevo planteamiento consiste en clonar las universidades, es decir, hacer $q_i$ copias de la universidad $u_i$ donde cada copia tendría una capacidad únicamente de un estudiante. De esta forma los emparejamientos que se produzcan emparejarán cada vértice de un conjunto con uno solo del otro conjunto.

Por otra parte, cada copia de cada universidad tendrá la misma lista de preferencia que la de la universidad de la que es copia y cada estudiante ordenará arbitrariamente las copias de cada universidad, respetando los órdenes entre universidades (es decir, que si $e_i$ prefiere a $u_k$ antes que a $u_l$ entonces todas las copias de $u_k$ irán por delante de todas las de $u_l$ en su lista de preferencias).

En estas condiciones nos encontramos ante un problema idéntico al del problema del matrimonio con listas incompletas a excepción de que el cardinal de ambos conjuntos puede no ser el mismo. Por tanto todo los resultados de esa sección son aplicables a esta.

## 3.2. Variantes del problema

El problema de la Admisión en Universidades en la vida real tiene diferentes versiones en función de las características o exigencias impuestas por las universidades. Vamos a exponer a continuación algunas de las variantes que nos podemos encontrar de este problema en diferentes casos, las cuales se recogen en [1].

- Listas incompletas por parte de las universidades: Cuando las universidades tienen requisitos de ingreso como un determinado nivel de algún idioma, puede haber ciertos estudiantes que no las cumplan y por tanto no puedan ser aceptados por esas universidades por muy buenas calificaciones de acceso que hayan obtenido.

- Indiferencias por parte de las universidades: Esta situación tratada en la última sección del capítulo anterior, puede deberse en este caso a igualdades en las notas de acceso de dos o más alumnos, de forma que a la universidad le da lo mismo aceptar a uno que a otro. Esto sin embargo, se suele resolver desempatando con otros criterios.

- Cuotas mínimas: El problema de la Admisión en Universidades cuenta con la cuota máxima de capacidad pero también puede darse que algunas universidades impongan una cuota mínima de forma que por debajo de ese cupo mínimo de estudiantes la universidad cerraría por razones de consistencia o rentabilidad de la propia universidad y por tanto no aceptaría a menos estudiantes que el número de la cuota.

- Cuotas comunes: Otra de las características que pueden variar respecto del problema original es el hecho de que ciertos campos de universidades pueden tener cuotas o convenios financiados estatalmente o de forma privada, de manera que entre varias universidades no se pueda superar un cierto número de estudiantes admitidos.

# Capítulo 4

# Compañeros de habitación

Tanto en el problema del matrimonio como en el de la admisión de estudiantes a universidades considerábamos dos conjuntos distinguidos (disjuntos) entre los cuales buscábamos un emparejamiento estable. El problema de los compañeros de habitación es una extensión no bipartita del problema del matrimonio.

En este problema se parte de un único conjunto de personas $C = \{c_1, c_2, ..., c_{2n}\}$ (de cardinal par) que pueden ser tanto hombres como mujeres, es decir, no hay distinción entre sexos en este caso. En este sentido, esta variante se podría interpretar como un problema del matrimonio en el cual se contempla también la posibilidad de que se produzcan parejas homosexuales, es decir, que se emparejen dos personas del mismo sexo.

Cada persona va a tener una lista de preferencias en la que ordena estrictamente a las $2n-1$ personas restantes de $C$. Como hemos dicho que solo va a haber un conjunto de personas entonces habrá una única tabla de listas de preferencias.

Así tendremos un grafo completo $K_{2n} = G = (C, E)$ donde $|C| = 2n$ y $E$ es el conjunto de aristas, que por tanto une cada vértice con los demás.

Este problema surge al tratar de resolver un problema de asignación de pares de estudiantes en habitaciones de una universidad. El objetivo por tanto va a consistir en asignar a cada persona un compañero para una habitación doble de manera que se generen $n$ parejas de forma estable.

La principal diferencia con respecto al problema del matrimonio es que en este caso es posible que no exista un emparejamiento estable para determinados conjuntos de personas con sus preferencias. Esta variante fue introducida por Gale-Shapley, pero no fue hasta 1985 cuando Robert W.Irving propuso un algoritmo, el Algoritmo de Irving, que resuelve este problema en caso de que exista solución.

Además de la aplicación directa de asignar personas en habitaciones dobles este problema también se puede aplicar en los campamentos para emparejar a los niños en cabañas de 2 personas, formar las parejas en los torneos de ajedrez, o incluso en los intercambios de riñón entre paciente y donante (ver [12]).

A lo largo de este capítulo nos basaremos principalmente en [10].

## 4.1. Primera Etapa del Algoritmo de Irving

En esta sección vamos a ver cuáles son las definiciones de pareja bloqueante y emparejamiento estable en este nuevo contexto, así como a probar la eficacia de la primera etapa del Algoritmo de Irving y un ejemplo de su aplicación.

Para el problema de los Compañeros de habitación tanto la definición de pareja bloqueante como la de estabilidad de un emparejamiento son las mismas que para el problema del matrimonio. Se corresponden por tanto con la Definición 2.1 y la Definición 2.2 (del segundo capítulo) respectivamente, adaptadas a un grafo no bipartito, cuyos vértices $c_i$ en este caso pertenecen a un mismo conjunto $C$.

Como ya dijimos al comienzo del capítulo este tipo de problemas no tiene por qué tener una solución estable. Es por ello que dado un problema de compañeros de habitación podemos encontrarnos con dos casos:

- Si el problema tiene una solución o emparejamiento estable, diremos que es resoluble.
- En caso de no ser así, diremos que es irresoluble.

En el siguiente ejemplo planteamos un problema de tipo irresoluble.

**Ejemplo 4.1** *Sea $C = \{c_1, c_2, c_3, c_4\}$ un conjunto de personas que pretenden organizarse en 2 habitaciones de 2 personas cada una. En la siguiente tabla se reflejan las preferencias que tiene cada una de ellos sobre las demás.*

| | | | |
|---|---|---|---|
| $c_1$: | $c_3$ | $c_2$ | $c_4$ |
| $c_2$: | $c_4$ | $c_3$ | $c_1$ |
| $c_3$: | $c_2$ | $c_4$ | $c_1$ |
| $c_4$: | $c_3$ | $c_2$ | $c_1$ |

*En este problema al ser de cardinal $|C| = 4$ solo existen tres emparejamientos que emparejen a todos los vértices: $P_1 = \{(c_1, c_2), (c_3, c_4)\}$, $P_2 = \{(c_1, c_3), (c_2, c_4)\}$ y por último $P_3 = \{(c_1, c_4), (c_2, c_3)\}$. Sin embargo, podemos comprobar que efectivamente ninguno de estos emparejamientos son estables ya que en todos ellos existe una pareja bloqueante:*

- *En $P_1$ tanto el par $(c_2, c_3)$ es una pareja bloqueante porque $c_2$ prefiere a cualquier otro compañero antes que a $c_1$ y $c_3$ prefiere compartir habitación con $c_2$ antes que con $c_4$.*
- *En $P_2$ el par $(c_3, c_4)$ es una pareja bloqueante ya que $c_3$ prefiere a cualquier otro compañero antes que a $c_1$ y $c_4$ prefiere compartir habitación con $c_3$ antes que con $c_2$.*
- *En $P_3$ el par $(c_2, c_4)$ es una pareja bloqueante puesto que $c_4$ prefiere a cualquier otro compañero antes que a $c_1$ y $c_2$ prefiere compartir habitación con $c_4$ antes que con $c_3$.*

*Por tanto, al no tener ningún emparejamiento estable, se trata de un problema irresoluble.*

Sin embargo, hay situaciones en las que sí que existen emparejamientos estables. El Algoritmo de Irving determina si existe un emparejamiento estable dado un conjunto de personas con sus correspondientes listas de preferencias, y encuentra uno en caso de que sí exista. Este algoritmo se divide en dos etapas. La primera de ellas se asemeja al Algoritmo de Gale-Shapley, al que se le añade una primera fase de reducción en las listas de preferencias de cada vértice. Detallamos a continuación los pasos de esta etapa del algoritmo.

- Etapa 1: Al principio ninguna persona tiene propuestas.
  - Paso 1: Cada $c_i$ al que no le hayan aceptado alguna propuesta le propone compartir habitación al vértice que más prefiere según su ordenación (siempre que ese vértice no le haya rechazado ya).
  - Paso 2: Cada vértice que haya recibido alguna propuesta en el paso anterior:
    - Si al llegar aquí no había aceptado ninguna propuesta, acepta la del vértice que más prefiere según su lista de todos los que le hayan propuesto, y rechaza a todas las demás propuestas.
    - Si ya había aceptado una propuesta, acepta la del vértice que más prefiere de entre todos los que le han propuesto (incluido el vértice cuya propuesta había aceptado provisionalmente), y rechaza a todas las demás propuestas.
  - Paso 3: Se repiten los pasos 1 y 2 a no ser que se de una de las dos situaciones siguientes (en cuyo caso esta etapa 1 termina):
    - A todos los vértices les han aceptado una propuesta, en cuyo caso se pasaría a la Fase de reducción I (detallada en el posterior Corolario 4.5).
    - Algún vértice ha sido rechazado por todos los demás, en cuyo caso no existe ningún emparejamiento estable para el problema.

A continuación, vamos a demostrar una serie de resultados sobre el funcionamiento de esta etapa del algoritmo, y en todos ellos partiremos de un grafo finito $G = (C, E)$. Además vamos a presentar y justificar el desarrollo de la Fase de reducción I con la que va a finalizar esta primera etapa del algoritmo en el caso de que ningún vértice haya sido rechazado por todos los demás. Esta fase de reducción va a consistir en reducir de cada una de las listas de preferencias ciertas personas que van a resultar irrelevantes para la Etapa 2 del algoritmo y por tanto simplificaremos antes de pasar a ella.

**Lema 4.2** *Si $c_j$ rechaza a $c_i$ en el desarrollo de la etapa 1 del algoritmo entonces $c_i$ y $c_j$ no van a poder ser pareja en ningún emparejamiento estable.*

**Dem:** Supongamos que el primero de todos los rechazos se produce por parte de $c_j$ a $c_i$ y supongamos, por reducción al absurdo, que $P$ es un emparejamiento estable en el que $c_i$ y $c_j$ son compañeros. Si $c_j$ rechaza a $c_i$ es porque $c_j$, o acaba de recibir o ya tenía una propuesta mejor de otra persona, que denotamos por $c_s$. Pero si $c_j$ prefiere a $c_s$ antes que a $c_i$ entonces por la estabilidad de $P$, como $c_i$ y $c_j$ son compañeros, la pareja de $c_s$ no es $c_j$, y $c_s$ debe preferir a su compañero en $P$ denotado por $c_t$, antes que a $c_j$. Por tanto, antes de que $c_s$ pueda hacerle una propuesta a $c_j$, tiene que haber sido rechazado por $c_t$, lo cual se contradice con la hipótesis de partida. ■

Este lema da lugar a los dos corolarios siguientes.

**Corolario 4.3** *Si en algún momento de la secuencia de proposiciones $c_i$ le hace una propuesta a $c_j$ entonces en cualquier emparejamiento estable:*

*(i) $c_i$ no puede tener un compañero mejor que $c_j$ según lista de preferencias.*

*(ii) $c_j$ no puede tener un compañero peor que $c_i$ según su lista de preferencias.*

**Dem:** Si $c_i$ le hace una propuesta a $c_j$, significa que $c_i$ ha sido rechazado por todo vértice mejor que $c_j$ según su lista, y por el Lema 4.2, ningún vértice que rechaza a otro puede ser compañero del rechazado en ningún emparejamiento estable. Luego hemos probado (i). Probaremos (ii) por reducción al absurdo. Supongamos que $c_s$ es compañero de $c_j$ en un emparejamiento estable $P$, y que $c_j$ prefiere a $c_i$ antes que $c_s$. Entonces como por (i), $c_i$ prefiere a $c_j$ antes que a su compañero en $P$, el emparejamiento $P$ no sería estable. ■

**Corolario 4.4** *Si algún vértice ha sido rechazado por todos los demás al terminar la etapa 1 del algoritmo, entonces no existe ningún emparejamiento estable para el problema.*

**Dem:** Por el lema 4.2, el vértice rechazado por todos los demás, no puede tener compañero en ningún un emparejamiento estable, luego no existe tal emparejamiento estable. ■

Como ya hemos visto, esta primera etapa del algoritmo ya identifica una situación en la que no existe un emparejamiento estable, y es la que se daría al aplicar esta etapa al ejemplo 4.1, que como ya vimos, no podía tener un emparejamiento estable. A continuación se muestran los diferentes pasos de la primera etapa del Algoritmo de Irving para ese ejemplo, con las peticiones de los vértices y los rechazos en cada paso.

$$
\begin{array}{c||c}
c_1: & \cancel{c_3} \\ \hline
c_2: & c_4 \\ \hline
c_3: & c_2 \\ \hline
c_4: & c_3
\end{array}
\quad \Rightarrow \quad
\begin{array}{c||cc}
c_1: & \cancel{c_3} & \cancel{c_2} \\ \hline
c_2: & c_4 & c_4 \\ \hline
c_3: & c_2 & c_2 \\ \hline
c_4: & c_3 & c_3
\end{array}
\quad \Rightarrow \quad
\begin{array}{c||ccc}
c_1: & \cancel{c_3} & \cancel{c_2} & \cancel{c_4} \\ \hline
c_2: & c_4 & c_4 & c_4 \\ \hline
c_3: & c_2 & c_2 & c_2 \\ \hline
c_4: & c_3 & c_3 & c_3
\end{array}
$$

Como vemos, $c_1$ es rechazado por todos los demás, luego es un problema irresoluble.

Una vez probados los dos resultados anteriores, ya podemos introducir la Fase de Reducción I.

**Corolario 4.5 (Fase de Reducción I)** *Si todas las personas tienen una propuesta al terminar la primera etapa del algoritmo, entonces la lista de preferencias de un $c_i$, cuya propuesta es de $c_j$, puede reducirse eliminando de la misma a las siguientes personas:*

*(i) Todo $c_s$ tal que $c_i$ prefiere a $c_j$ antes que a $c_s$ (los vértices que están por detrás de $c_j$ en la lista de $c_i$).*

*(ii) Todo $c_s$ tal que su pareja provisional, $c_t$, sea anterior a $c_i$ en su lista de preferencias (incluyendo aquellas que rechazaron a $c_i$).*

*En las listas de preferencias reducidas resultantes:*

*(iii) $c_i$ está el primero en la lista de $c_j$, y $c_j$ está el último en la lista de $c_i$.*

*(iv) En general, un $c_s$ está en la lista de un $c_t$ si y solo si $c_t$ está en la de $c_s$ (diremos que son simétricos en la listas).*

**Dem:** Los apartados (i) y (ii) se deducen directamente del Corolario 4.3 (ii), de manera que aquellos vértices que se eliminan de las listas de preferencias, no van a influir en un emparejamiento estable. Del apartado (iii), $c_j$ está el último en la lista de $c_i$ ya que, por el apartado (i), se eliminan todos los vértices que estén por detrás de $c_j$; $c_i$ por su parte, está el primero en la lista de $c_j$ porque por el apartado (ii), se eliminan también aquellos vértices que rechazaron, en este caso, a $c_j$. Y el (iv) es trivial, puesto que si un vértice no está en la lista de otro, esa pareja no se va a dar en el algoritmo. ∎

**Lema 4.6** *Si las listas de preferencias reducidas de cada persona tienen una única persona, entonces determinan el emparejamiento estable que genera el algoritmo.*

**Dem:** El hecho de que las listas que cumplen la hipótesis de este lema determinan un emparejamiento $P$, es una consecuencia del Corolario 4.5 (iv). Ahora, supongamos que $c_i$ prefiere a $c_j$ antes que a la única persona que tiene en su lista. Entonces $c_i$ fue rechazado por $c_j$, porque habría recibido una propuesta mejor. Pero la propuesta con la que finalmente se ha quedado $c_j$ en $P$ es la de única persona que hay en su lista, y por eso $c_j$ claramente prefiere a esa persona antes que a $c_i$. Por lo tanto, no puede haber inestabilidad en el emparejamiento determinado por las listas de preferencias reducidas, es decir, que $P$ es estable. ∎

Para comprender mejor el funcionamiento de esta primera etapa algoritmo y la reducción de sus listas de preferencias, mostraremos a continuación un ejemplo:

**Ejemplo 4.7** *Sea* $C = \{c_1, c_2, c_3, c_4, c_5, c_6\}$ *un conjunto de personas que quieren ocupar tres habitaciones de 2 personas cada una. Las siguientes listas de preferencias de estas personas son las siguientes:*

| | | | | | |
|---|---|---|---|---|---|
| $c_1$: | $c_3$ | $c_4$ | $c_2$ | $c_6$ | $c_5$ |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ | $c_3$ |
| $c_3$: | $c_2$ | $c_4$ | $c_5$ | $c_1$ | $c_6$ |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $c_6$ | $c_1$ |
| $c_5$: | $c_3$ | $c_1$ | $c_2$ | $c_4$ | $c_6$ |
| $c_6$: | $c_5$ | $c_1$ | $c_3$ | $c_4$ | $c_2$ |

*Para organizar las habitaciones aplicamos primero al problema la secuencia de propuestas, llegando así a que a cada una de las 6 personas le han aceptado las siguientes:*

| | | |
|---|---|---|
| $c_1$: | $\cancel{c_3}$ | $c_4$ |
| $c_2$: | $c_6$ | $c_6$ |
| $c_3$: | $c_2$ | $c_2$ |
| $c_4$: | $c_5$ | $c_5$ |
| $c_5$: | $c_3$ | $c_3$ |
| $c_6$: | $\cancel{c_5}$ | $c_1$ |

*Como no hay ninguna persona que haya sido rechazada por todas las demás, podemos pasar a la Fase de Reducción I, en la cual eliminamos ciertas personas de algunas de las listas de preferencias de partida y a los simétricos seguidamente (por el Corolario 4.5 (iv)).*

*En primer lugar vamos a eliminar a las personas rechazadas en la fase de las propuestas (luego son parejas que no van a poder darse). Por tanto eliminamos $c_3$ de la lista de $c_1$ y al simétrico, y quitamos también a $c_5$ de la lista de $c_6$ y a su simétrico.*

*En segundo lugar, por el Corolario 4.5 (i), de la lista de $c_1$ eliminaríamos a $c_5$, y de la lista de $c_3$ eliminaríamos a $c_1$ (que ya le habíamos quitado) y a $c_6$ al simétrico. Con lo que ya hemos quitado, el Corolario 4.5 (ii) ya no cambiaría nada más y por tanto daríamos por terminada esta fase, acabando así la Etapa 1 del algoritmo.*

*De esta forma, las listas de preferencias reducidas quedarían de la siguiente manera:*

| | | | | | |
|---|---|---|---|---|---|
| $c_1$: | $\cancel{c_3}$ | $c_4$ | $c_2$ | $c_6$ | $\cancel{c_5}$ |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ | $c_3$ |
| $c_3$: | $c_2$ | $c_4$ | $c_5$ | $\cancel{c_1}$ | $\cancel{c_6}$ |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $c_6$ | $c_1$ |
| $c_5$: | $c_3$ | $\cancel{c_1}$ | $c_2$ | $c_4$ | $\cancel{c_6}$ |
| $c_6$: | $\cancel{c_5}$ | $c_1$ | $\cancel{c_3}$ | $c_4$ | $c_2$ |

Con este ejemplo se ve claro que, tras esta primera etapa, el algoritmo no devuelve de primeras un emparejamiento realmente, pues $c_4$ recibe una propuesta de $c_1$ y $c_1$ a su vez la recibe de $c_6$. Por tanto, será necesaria una segunda etapa para poder finalmente emparejar los elementos, y de una forma estable (siempre que esto sea posible).

## 4.2. Segunda Etapa del Algoritmo de Irving

Una vez vista la Etapa 1 del Algoritmo de Irving, falta por estudiar la segunda etapa que consiste en seguir reduciendo de forma iterativa las listas ya reducidas, cuando estas siguen teniendo alguna con más de una persona en ellas. Cuando hablemos de listas reducidas en esta sección nos referiremos a la siguiente definición.

**Definición 4.8** *Diremos que un conjunto de listas de preferencias está reducido si cumple las siguientes condiciones:*

- *Han pasado por la Fase de Reducción I (Corolario 4.5).*
- *Han pasado por cero o más reducciones de la Etapa 2 (que describiremos más abajo y llamaremos Fase de Reducción II).*

La Etapa 2 se basa en generar secuencias cíclicas $a_1 ... a_r$ de personas tales que:

- Para $i = 1, ..., r-1$, la segunda persona de la actual lista de preferencias reducida de $a_i$ es la primera persona de la de $a_{i+1}$; denotaremos a esa persona como $b_{i+1}$.
- La segunda persona de la actual lista de preferencias reducida de $a_r$ es la primera de la de $a_1$; denotaremos a esa persona como $b_1$.

Como $b_i$ es el primero en la lista de $a_i$, esto nos indica que $b_i$ tiene una propuesta (que ha aceptado) de $a_i$, por el Corolario 4.5.

**Definición 4.9 (Ciclos "todo o nada")** *A una secuencia cíclica con las características recién expuestas, se le llama ciclo "todo o nada", relativo a las actuales listas de preferencias reducidas, que permiten reducir dichas listas aún más, y se generan de la siguiente manera. Basta tomar una persona arbitraria $p_i$ cuya lista de preferencias reducida contenga a más de una persona, y generar la secuencia:*

- $q_i$ = *segunda persona de la actual lista de preferencias reducida de* $p_i$.
- $p_{i+1}$ = *última persona de la actual lista de preferencias reducida de* $q_i$ *(luego* $q_i$ *es la primera persona en la lista reducida de* $p_{i+1}$*).*
- *Continuar generando la secuencia hasta que se encuentre un ciclo en la sucesión de* $p$ *(se repita un* $p_s$*), y entonces definiremos:*

$$a_i = p_{s+i-1}(i = 1, 2, ...)$$

*donde* $p_s$ *es la primera persona de la secuencia de* $p$ *que se repite. Denotaremos* $p_1, p_2, ..., p_{s-1}$ *como la "cola" del ciclo.*

Para ejemplificar la creación de un ciclo "todo o nada", vamos a construir uno a partir del Ejemplo 4.7.

**Ejemplo 4.10** *Partimos entonces de las listas reducidas obtenidas al aplicar la Fase de Reducción I en dicho Ejemplo 4.7, así como de las propuestas aceptadas hasta el momento en el mismo:*

| | | | | | |
|---|---|---|---|---|---|
| $c_1$: | $c_4$ | $c_2$ | $c_6$ | | |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ | $c_3$ |
| $c_3$: | $c_2$ | $c_4$ | $c_5$ | | |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $c_6$ | $c_1$ |
| $c_5$: | $c_3$ | $c_2$ | $c_4$ | | |
| $c_6$: | $c_1$ | $c_4$ | $c_2$ | | |

| | |
|---|---|
| $c_1$: | $c_4$ |
| $c_2$: | $c_6$ |
| $c_3$: | $c_2$ |
| $c_4$: | $c_5$ |
| $c_5$: | $c_3$ |
| $c_6$: | $c_1$ |

*Vamos a generar un ciclo "todo o nada"partiendo del vértice* $p_1 = c_5$*, que tiene más de una persona en su lista de preferencias reducida. Ahora vamos creando una secuencia como en la Definición 4.9:*

$$\begin{array}{ll} q_1 = c_2 & p_2 = c_3 \\ q_2 = c_4 & p_3 = c_1 \\ q_3 = c_2 & p_4 = c_3 \end{array}$$

*Como ya hemos repetido un* $p_i$*, que es* $p_2 = p_4$*, entonces* $s = 2$*, y así podemos definir:*

$$\begin{array}{ll} a_1 = p_2 = c_3 & b_1 = c_2 \\ a_2 = p_3 = c_1 & b_2 = c_4 \\ a_3 = p_4 = c_3 & b_3 = c_2 \end{array}$$

*Llegamos así al ciclo "todo o nada"* $c_3c_1c_3$*, que tiene una cola de longitud 1:* $c_5$.

La Etapa 2 del Algoritmo de Irving, parte de las propuestas aceptadas en la Etapa 1 y de las listas reducidas generadas en la Fase de Reducción I, y se va desarrollando iterativamente. Cada iteración consta de los siguientes pasos:

- Paso 1 (Fase de Reducción II): Generar un ciclo "todo o nada"$a_1...a_r$, con sus correspondientes $b_1...b_r$, y forzar a cada $b_i$ $(1 \leq i \leq r)$ a rechazar la propuesta que tenía de $a_i$, de manera que se fuerza a cada $a_i$ a hacerle una propuesta a $b_{i+1}$ (módulo $r$), que es la segunda persona de su actual lista reducida. De esta forma se reducirían las listas reducidas, eliminando a $a_i$ de la lista reducida de $b_i$ y al simétrico.

- Paso 2: Aplicar la Fase de Reducción I a las nuevas listas reducidas en la Fase de Reducción II del paso anterior, teniendo en cuenta también los cambios del paso anterior en las propuestas aceptadas.
- Paso 3: Volver a repetir los pasos 1 y 2 (nueva iteración), a no ser que se de una de las dos situaciones siguientes:
    - Alguna lista reducida ha quedado vacía, en cuyo caso el problema no tiene solución estable.
    - Todas las listas reducidas contienen a una única persona, en cuyo caso determinan el emparejamiento estable generado por el algoritmo.

**Observación 4.11** *Respecto a esta segunda etapa del algoritmo, notemos que:*

*(1) Cuando una persona tiene una sola opción en su lista reducida, quiere decir que aceptó una propuesta de esa otra persona, y que a la vez se propuso a dicha persona, quien también lo aceptó a él. Podríamos entonces eliminarlos del problema a ambos, ya que forman una pareja estable: el otro es simultáneamente lo mejor y lo peor que puede conseguir cada uno, por el Lema 4.6.*

*(2) En cada iteración del paso 2 de la Etapa 2 del algoritmo, se puede aplicar en cada lista de preferencias reducida la Fase de Reducción I (Corolario 4.5), puesto que como $a_i$ no logra un socio mejor que $b_{i+1}$, entonces, para interés de la estabilidad, $b_{i+1}$ no puede conformarse con un socio peor que $a_i$ (luego se pueden ir eliminando vértices de esa forma).*

Para ver cómo funciona esta Etapa 2 del Algoritmo de Irving en su conjunto, vamos a continuar con el Ejemplo 4.7, aplicándole una iteración de esta segunda etapa.

**Ejemplo 4.12** *Partimos entonces de las listas reducidas obtenidas al aplicar la Fase de Reducción I en el Ejemplo 4.7, junto con las propuestas aceptadas hasta el momento en dicho ejemplo:*

| | | | | | |
|---|---|---|---|---|---|
| $c_1$: | $c_4$ | $c_2$ | $c_6$ | | |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ | $c_3$ |
| $c_3$: | $c_2$ | $c_4$ | $c_5$ | | |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $c_6$ | $c_1$ |
| $c_5$: | $c_3$ | $c_2$ | $c_4$ | | |
| $c_6$: | $c_1$ | $c_4$ | $c_2$ | | |

| | |
|---|---|
| $c_1$: | $c_4$ |
| $c_2$: | $c_6$ |
| $c_3$: | $c_2$ |
| $c_4$: | $c_5$ |
| $c_5$: | $c_3$ |
| $c_6$: | $c_1$ |

*Vamos a aplicar la Etapa 2 del Algoritmo, iterando ciclos "todo o nada"para seguir reduciendo el conjunto de listas de preferencias.*

- *Paso 1: Tomamos $p_1 = c_1$, que tiene más de una persona en su lista de preferencias reducida. Ahora vamos creando una secuencia como en la Definición 4.9:*

$$q_1 = c_2 \qquad p_2 = c_3$$
$$q_2 = c_4 \qquad p_3 = c_1$$

*Como ya hemos repetido un $p_i$, que es $p_1 = p_3$, entonces $s = 1$, y así podemos definir:*

$$\begin{array}{ll} a_1 = p_1 = c_1 & \quad b_1 = c_4 \\ a_2 = p_2 = c_3 & \quad b_2 = c_2 \\ a_3 = p_3 = c_1 & \quad b_3 = c_4 \end{array}$$

*Ahora que ya tenemos el ciclo "todo o nada" $c_1c_3c_1$, forzamos por tanto a $c_4$ a rechazar la propuesta de $c_1$ y a $c_2$ a rechazar la de $c_3$, de manera que $c_1$ le hace una propuesta a $c_2$, y $c_3$ a $c_4$. Entonces las nuevas propuestas aceptadas, así como las listas reducidas según la Fase de Reducción II, quedarían de la siguiente forma:*

| | | | | | |
|---|---|---|---|---|---|
| $c_1$: | $\cancel{c_4}$ | $c_2$ | $c_6$ | | |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ | $\cancel{c_3}$ |
| $c_3$: | $\cancel{c_2}$ | $c_4$ | $c_5$ | | |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $c_6$ | $\cancel{c_1}$ |
| $c_5$: | $c_3$ | $c_2$ | $c_4$ | | |
| $c_6$: | $c_1$ | $c_4$ | $c_2$ | | |

| | | |
|---|---|---|
| $c_1$: | $\cancel{c_4}$ | $c_2$ |
| $c_2$: | $c_6$ | $c_6$ |
| $c_3$: | $\cancel{c_2}$ | $c_4$ |
| $c_4$: | $c_5$ | $c_5$ |
| $c_5$: | $c_3$ | $c_3$ |
| $c_6$: | $c_1$ | $c_1$ |

- *Paso 2: Aplicamos ahora la Fase de Reducción I, por la cual según el Corolario 4.5 (i), de todas las listas reducidas actuales sólo habría que eliminar a $c_6$ de la lista de $c_4$ y al simétrico. Nos quedaría por tanto el siguiente conjunto de listas de preferencias reducidas:*

| | | | | |
|---|---|---|---|---|
| $c_1$: | $c_2$ | $c_6$ | | |
| $c_2$: | $c_6$ | $c_5$ | $c_4$ | $c_1$ |
| $c_3$: | $c_4$ | $c_5$ | | |
| $c_4$: | $c_5$ | $c_2$ | $c_3$ | $\cancel{c_6}$ |
| $c_5$: | $c_3$ | $c_2$ | $c_4$ | |
| $c_6$: | $c_1$ | $\cancel{c_4}$ | $c_2$ | |

- *Paso 3: Como no todas las listas reducidas contienen una única persona, y ninguna de ellas está vacía, habría que volver a repetir los pasos 1 y 2.*

El siguiente Lema trata ciertas propiedades en relación con los elementos de los ciclos "todo o nada".

**Lema 4.13** *Sea $a_1...a_r$ un ciclo "todo o nada" relativo a un conjunto de listas de preferencias reducidas, y sea $b_i$ la primera persona de la lista reducida de $a_i$ $(1 \leq i \leq r)$. Entonces:*

*(i) En cualquier emparejamiento estable contenido en esas listas reducidas, tanto $a_i$ como $b_i$ o están emparejados para todo valor de $i$, o no lo están para ningún valor de $i$.*

*(ii) Si hay un emparejamiento estable en el cual $a_i$ y $b_i$ están emparejados para todo $1 \leq i \leq r$, entonces hay otro emparejamiento estable en el que no lo están.*

**Dem:** (i) Consideremos los subíndices módulo $r$, y supongamos que para algún $i$ fijado, $a_i$ y $b_i$ son compañeros en un emparejamiento estable contenido en las listas reducidas. Sabemos que $a_i$ es la última persona en la lista reducida de $b_i$, y que $b_i$ es la segunda persona en la de $a_{i-1}$ (luego $a_{i-1}$ está en la lista reducida de $b_i$). Por tanto, $b_i$ prefiere a $a_{i-1}$ antes que a $a_i$. Entonces por la estabilidad del emparejamiento, $a_{i-1}$ debe ser compañero de una persona a la cual prefiere antes que a $b_i$, y la única persona de su lista reducida que está por delante es $b_{i-1}$ (la primera). Repitiendo sucesivamente este razonamiento llegamos a que $a_i$ y $b_i$ son compañeros en el emparejamiento estable para todos los valores de $i$.

(ii) Sean $A = \{a_1, ..., a_r\}$ y $B = \{b_1, ..., b_r\}$. Se pueden dar dos situaciones: $A \cap B \neq \emptyset$ o $A \cap B = \emptyset$. Veamos que cuando $A \cap B \neq \emptyset$, no puede existir un emparejamiento estable donde $a_i$ esté con $b_i$ (primera opción en su lista reducida) para todo $1 \leq i \leq r$. Esto se debe a que si $A \cap B \neq \emptyset$, entonces existe al menos un $a_i = b_j$, donde $b_j (= a_i)$ es la primera persona en la lista reducida de $a_j$, y $b_i$ la primera en la de $a_i (= b_j)$, lo cual lleva a que $a_j = b_i$. Pero entonces, al ser $a_i = b_j$ y $a_j = b_i$, se llega a que tanto para $a_i$ como para $a_j$, la primera y la última persona de sus respectivas listas reducidas coinciden, por lo que sus listas contendrían a una única persona, y entonces ni $a_i$ ni $a_j$ podrían formar parte del ciclo "todo o nada". Por el apartado (i) por tanto, $a_i$ y $b_i$ no pueden ser compañeros para ningún $1 \leq i \leq r$ en ningún emparejamiento estable.

Tenemos que suponer entonces que $A \cap B = \emptyset$. Sea $M$ un emparejamiento estable contenido en las listas de preferencias reducidas, en el cual $a_i$ y $b_i$ son compañeros para todo $1 \leq i \leq r$. Sea $M'$ un emparejamiento en el cual cada $a_i$ está emparejado con $b_{i+1}$, y cada persona que no está en $A \cup B$ tiene el mismo compañero que en $M$. Queremos ver por tanto que $M'$ es estable.
Cada persona de $B$ consigue en $M'$ un compañero mejor que en $M$ (ya que el de $M$ es su peor opción), las personas de $A$ se quedan en $M'$ con un compañero peor que el que tenían en $M$ (ya que en $M$ estaban emparejados con sus mejores opciones), y las personas que no están ni en $A$ ni en $B$ se quedan igual. Por tanto, cualquier inestabilidad en $M'$ que no estuviera en $M$, tiene que venir de las personas de $A$. Supongamos entonces que un $a_i$ prefiere a $c_j$ más que a $b_{i+1}$ (su compañero en $M'$), luego sólo habría tres casos posibles a contemplar:

- $a_i$ y $c_j$ son compañeros en $M$, es decir $c_j = b_i$, y por tanto $c_j$ hemos dicho que prefiere a su pareja en $M'$ (que sería $a_{i-1}$) antes que a $a_i$ (la última opción de su lista).

- $a_i$ también prefiere a $c_j$ antes que a $b_i$, en cuyo caso $c_j$ no puede estar en la actual lista reducida de $a_i$, porque de ser así, $c_j$ precedería a la actual primera persona de la lista reducida de $a_i$, que es $b_i$. Por ello, $c_j$ ha tenido antes que rechazar a $a_i$, o bien ha sido forzado a rechazarlo. En el primer caso, $c_j$ habría tenido que recibir una propuesta de una persona mejor que $a_i$ según su lista, mientras que en el segundo caso, el rechazo obligado habría ofrecido a $c_j$ una propuesta de nuevo mejor que $a_i$ según su lista. En ambos casos, $c_j$ prefiere a su compañero en $M'$ antes que a $a_i$.

- $a_i$ prefiere a $b_i$ antes que a $c_j$, en cuyo caso $c_j$ se encuentra entre $b_i$ y $b_{i+1}$ en la lista original de $a_i$, pero de nuevo no está en la actual lista reducida de $a_i$. Esto sería el resultado de que $c_j$ obtenga una propuesta de alguien mejor que $a_i$ según su lista, de modo que, al igual que en el caso anterior, $c_j$ debe preferir a su compañero en $M'$ antes que a $a_i$.

Por tanto, hemos visto que no existe ninguna inestabilidad en $M'$, luego $M'$ es estable. ■

El significado esencial de un conjunto reducido de listas de preferencias es una consecuencia del lema anterior, y es el siguiente. Si el problema original admite un emparejamiento estable, entonces hay un emparejamiento estable en el que cada persona es emparejada con alguien de su lista de preferencias reducida. Decimos entonces que ese emparejamiento está contenido en las listas reducidas. Además, el lema que acabamos de demostrar, tiene dos consecuencias inmediatas recogidas en los dos siguientes corolarios.

**Corolario 4.14** *Si el problema original admite un emparejamiento estable, entonces hay un emparejamiento estable contenido en cualquier conjunto de listas reducidas.*

**Corolario 4.15** *Si una o más listas de un conjunto de listas de preferencias reducidas están vacías, entonces el problema original no admite ningún emparejamiento estable.*

El resultado que vamos a ver a continuación es una extensión del Lema 4.6, en donde se complementa la situación en la que el Algoritmo de Irving llega a una solución estable.

**Lema 4.16** *Si en un conjunto de listas de preferencias reducidas, toda lista contiene a una única persona, entonces dichas listas determinan el emparejamiento estable generado por el algoritmo.*

**Dem:** El hecho de que las listas determinen un emparejamiento es una consecuencia del Corolario 4.5. Supongamos ahora que $c_i$ prefiere a $c_j$ antes que a la única persona de su lista reducida. Entonces, igual que hicimos en Lema 4.13 (ii), se demuestra que $c_j$ prefiere a la única persona de su lista reducida antes que a $c_i$. Por tanto, tal emparejamiento no puede ser inestable, luego se trata de un emparejamiento estable. ■

Vamos a terminar ahora el Ejemplo 4.12 aplicando más iteraciones de la segunda etapa del Algoritmo de Irving, para llegar finalmente a una solución estable.

**Ejemplo 4.17** *Retomando el Ejemplo 4.12, en el paso 3 de la primera iteración llegamos a que hay que hacer una nueva iteración, luego volvemos a repetir los pasos 1 y 2 de esta etapa del algoritmo.*

- *Paso 1: Tomamos $p_1 = c_1$, que sigue teniendo más de una persona en su lista de preferencias reducida, y creamos de igual manera la secuencia:*

$$\begin{array}{ll} q_1 = c_6 & p_2 = c_2 \\ q_2 = c_5 & p_3 = c_4 \\ q_3 = c_2 & p_4 = c_1 \end{array}$$

  *Como ya hemos repetido un $p_i$, que es $p_1 = p_4$, entonces $s = 1$, y así podemos definir:*

$$\begin{array}{ll} a_1 = p_1 = c_1 & b_1 = c_2 \\ a_2 = p_2 = c_2 & b_2 = c_6 \\ a_3 = p_3 = c_4 & b_3 = c_5 \\ a_4 = p_4 = c_1 & b_4 = c_2 \end{array}$$

*Ahora que ya tenemos el ciclo "todo o nada"* $c_1c_2c_4c_1$*, forzamos por tanto a* $c_2$ *a rechazar la propuesta de* $c_1$*, a* $c_6$ *a rechazar la de* $c_2$*, y a* $c_5$ *a rechazar la de* $c_4$*, de manera que* $c_1$ *le hace una propuesta a* $c_6$*,* $c_2$ *a* $c_5$*, y* $c_4$ *a* $c_2$*. Entonces las nuevas propuestas aceptadas, así como las listas reducidas según la Fase de Reducción II, quedarían de la siguiente forma:*

| | | | | |
|---|---|---|---|---|
| $c_1$: | $\cancel{c_2}$ | $c_6$ | | |
| $c_2$: | $\cancel{c_6}$ | $c_5$ | $c_4$ | $\cancel{c_1}$ |
| $c_3$: | $c_4$ | $c_5$ | | |
| $c_4$: | $\cancel{c_5}$ | $c_2$ | $c_3$ | |
| $c_5$: | $c_3$ | $c_2$ | $\cancel{c_4}$ | |
| $c_6$: | $c_1$ | $\cancel{c_2}$ | | |

| | | |
|---|---|---|
| $c_1$: | $\cancel{c_2}$ | $c_6$ |
| $c_2$: | $\cancel{c_6}$ | $c_5$ |
| $c_3$: | $c_4$ | $c_4$ |
| $c_4$: | $\cancel{c_5}$ | $c_2$ |
| $c_5$: | $c_3$ | $c_3$ |
| $c_6$: | $c_1$ | $c_1$ |

- *Paso 2: En esta iteración, la Fase de Reducción I no cambia nada en las listas reducidas actuales.*

- *Paso 3: Como no todas las listas reducidas contienen una única persona (aunque ya vemos que las de* $c_1$ *y* $c_6$ *sí, que por la Observación 4.11 (ii) ya son compañeros definitivos del emparejamiento estable), y ninguna de ellas está vacía, volvemos a repetir los pasos 1 y 2.*

- *Paso 1: Tomamos* $p_1 = c_2$*, que tiene más de una persona en su lista de preferencias reducida y creamos la secuencia:*

$$\begin{array}{ll} q_1 = c_4 & \quad p_2 = c_3 \\ q_2 = c_5 & \quad p_3 = c_2 \end{array}$$

*Como ya hemos repetido un* $p_i$*, en concreto* $p_1 = p_3$ $(s = 1)$*, entonces tenemos que:*

$$\begin{array}{ll} a_1 = p_1 = c_2 & \quad b_1 = c_5 \\ a_2 = p_2 = c_3 & \quad b_2 = c_4 \\ a_3 = p_3 = c_2 & \quad b_3 = c_5 \end{array}$$

*Ahora que ya tenemos el ciclo "todo o nada"* $c_2c_3c_2$*, forzamos por tanto a* $c_5$ *a rechazar la propuesta de* $c_2$ *y a* $c_4$ *a rechazar la de* $c_3$*, de manera que* $c_2$ *le hace una propuesta a* $c_4$*, y* $c_3$ *a* $c_5$*. Entonces las nuevas propuestas aceptadas, así como las listas reducidas según la Fase de Reducción II, quedarían de la siguiente forma:*

| | | |
|---|---|---|
| $c_1$: | $c_6$ | |
| $c_2$: | $\cancel{c_5}$ | $c_4$ |
| $c_3$: | $\cancel{c_4}$ | $c_5$ |
| $c_4$: | $c_2$ | $\cancel{c_3}$ |
| $c_5$: | $c_3$ | $\cancel{c_2}$ |
| $c_6$: | $c_1$ | |

| | | |
|---|---|---|
| $c_1$: | $c_6$ | $c_6$ |
| $c_2$: | $\cancel{c_5}$ | $c_4$ |
| $c_3$: | $\cancel{c_4}$ | $c_5$ |
| $c_4$: | $c_2$ | $c_2$ |
| $c_5$: | $c_3$ | $c_3$ |
| $c_6$: | $c_1$ | $c_1$ |

- *Paso 2: En esta iteración, la Fase de Reducción I tampoco cambia nada en las listas reducidas actuales.*

- *Paso 3: Como hemos llegado a un conjunto de listas de preferencias reducidas en el que todas ellas contienen a una única persona, dichas listas determinan el emparejamiento estable al que llegamos finalmente al aplicar el Algoritmo de Irving completo.*

*Esta solución estable obtenida, organizaría al conjunto de 6 personas de nuestro ejemplo* ($C = \{c_1, c_2, c_3, c_4, c_5, c_6\}$) *en habitaciones dobles de la siguiente manera:*

| | |
|---|---|
| *Habitación 1:* | $c_1$ $c_6$ |
| *Habitación 2:* | $c_2$ $c_4$ |
| *Habitación 3:* | $c_3$ $c_5$ |

En este problema se pueden contemplar también las variantes que se trataron en el capítulo 2, en este caso serían el problema de los compañeros de habitación con listas de preferencias incompletas y en el que las listas presentan indiferencias. Vamos a limitarnos a definir ambas variantes (ver [3]).

La primera generalización de este problema es aquella en la que las listas de preferencias de cada persona pueden estar incompletas, es decir, que puede que alguna persona no quiera compartir habitación con ciertas personas, y por tanto no las incluya en su lista de preferencias. Diremos que $c_j$ es aceptable para $c_i$ si $c_j$ aparece en la lista de preferencias de $c_i$, diremos que es inaceptable en caso contrario. En este contexto, un emparejamiento $M$ debe satisfacer la propiedad de que si $(c_i, c_j) \in M$, entonces $c_i$ y $c_j$ son aceptables entre sí. En cuanto a la definición de estabilidad, tendríamos que: un par $(a_i, a_j)$ de personas mutuamente aceptables es una pareja bloqueante para un emparejamiento $M$ si no pertenece a $M$, y ninguno de los dos tiene compañero en $M$ o prefieren compartir habitación juntos antes que con sus respectivas parejas en $M$. Así, podemos suponer sin pérdida de generalidad en la estabilidad de un emparejamiento, que $c_i$ es aceptable para $c_j$ si y solo si $c_j$ es aceptable para $c_i$. Notemos que esta definición asume que una persona prefiere compartir habitación con cualquier persona aceptable según su lista de preferencias antes que quedarse sin compañero.

Otra de las generalizaciones naturales de este problema surge cuando algunas personas sienten indiferencia a la hora de compartir habitación entre determinadas personas. En esta situación, como ocurría al final del segundo capítulo, se pueden definir tres tipos de estabilidad: súper-estable, fuerte y débil. Los tres tipos se definen a partir del concepto de pareja bloqueante para cada una de las situaciones, y se caracterizan de la misma manera en que lo hicimos en la sección de Listas con Indiferencias del capítulo 2.

# Capítulo 5

# Otros problemas de emparejamientos

A lo largo de la memoria hemos tratado principalmente tres de los problemas de asignación más conocidos, cada uno de ellos con su correspondiente definición de estabilidad. Sin embargo, estos tres problemas de asignación no son ni mucho menos los únicos que han sido estudiados y analizados. Hay muchas más situaciones de la vida real donde se requiere establecer emparejamientos entre diversos agentes y en donde se aplican procedimientos y algoritmos similares a los que hemos visto ([16], [12]).

Una aplicación importante y a gran escala del Problema del Matrimonio Estable consiste en asignar usuarios a servidores en un gran servicio de Internet distribuido. Miles de millones de usuarios acceden a páginas web, videos y otros servicios en Internet, lo que requiere que cada usuario sea emparejado con uno de (potencialmente) cientos de miles de servidores en todo el mundo que ofrecen ese servicio.
Cada usuario prefiere servidores que estén lo suficientemente próximos como para garantizar un tiempo de respuesta más rápido, luego para mejorar el rendimiento, será mejor asignar servidores a clientes que estén más cerca geográficamente. Además cada servidor prefiere servir a los usuarios que le suponen un coste menor. Una diferencia con el Problema del Matrimonio Estable estándar es que además hay que tener en consideración los costes que deben satisfacerse se asignan en forma de pesos en el grafo a cada agente del problema.

Podemos considerar también una extensión "de muchos a muchos" del matrimonio estable de modo que tanto hombres como mujeres tengan cuota. Uno puede ver fácilmente que la técnica de añadir copias idénticas (utilizada para asemejar el problema de la Admisión de estudiantes a universidades al de los Matrimonios Estables) ya no se puede aplicar aquí ya que si lo hacemos, el emparejamiento estable resultante puede crear múltiples copias del mismo par.

Hay algunos problemas de emparejamiento en los que solo un conjunto (por ejemplo los hombres) tiene listas de preferencias sobre el otro. Son problemas con listas de preferencias unilaterales. Además, un emparejamiento máximo de rango (o emparejamiento codicioso) es aquel que empareja el número máximo de hombres con sus primeras opciones en sus respectivas listas y, sujeto a esta condición, el número máximo de hombres con sus segundas opciones, y así sucesivamente. Irving proporcionó algoritmos para este problema de encontrar un emparejamiento codicioso. Para dos emparejamientos $P$ y $P'$, si el número de hombres que prefieren $P$ a $P'$ (en términos del rango de su pareja) es mayor que el de hombres que prefieren $P$ a $P'$ decimos que $P$ es más popular que $P'$. Un emparejamiento $P$ es popular si no hay emparejamiento más popular que $P$.

De manera similar al problema de la asignación de estudiantes de medicina a hospitales tenemos también el problema de asignar marineros a los barcos en la Marina de los EE.UU. La Marina es responsable de organizar estas asignaciones y debe hacerlo de manera que el coste de realizar esas asignaciones se minimice o se mantenga dentro de los límites del presupuesto con el que cuentan. El coste de reeducar y capacitar a los marineros de manera adecuada para sus nuevas funciones también es un parámetro de coste.

Los marineros por su parte reciben nuevas asignaciones cada pocos años y deben entregar una lista de preferencias donde indiquen los destinos que prefieren ordenadamente, puesto que de la asignación va a depender su conformidad y la de los comandantes de los barcos (en el caso óptimo del barco, el agrado de los marineros es mínimo, lo que resultaría en una caída de la moral a bordo, y viceversa en el caso óptimo para el marinero).

Otra aplicación real e indispensable a día de hoy es el Problema del Intercambio de Riñón. Actualmente hay largas listas de espera para los trasplantes de riñón.Además se pueden producir incompatibilidades sanguíneas o inmunológicas entre el donante y el paciente. Es por esto que es lógico que se pretenda realizar la mejor asignación posible de las parejas basándose en criterios de compatibilidad y priorización (que hagan que la asignación resulte eficaz y eficiente) con la ayuda de un algoritmo.

En definitiva, los problemas de asignación estable están presentes en nuestro día a día y buscan emparejar agentes de forma estable con el fin de relacionar determinados conjuntos sin que dichas relaciones se rompan.

# Bibliografía


[1] P.Biró, T.Fleiner, R.W.Irving, D.F.Manlove. *The College Admissions problem with lower and common quotas.* Theoretical Computer Science, Vol.411: 3136-3153. 2010.

[2] L.E.Dubins, D.A.Freedman. *Machiavelli and the Gale-Shapley Algorithm.* The American Mathematical Monthly, Vol.88, Nº 7: 485-494. 1981.

[3] T.Fleiner, R.W.Irving, D.F.Manlove. *Efficient algorithms for generalized Stable Marriage and Roommates problems.* Theoretical Computer Science, Vol.381: 162-176. 2007.

[4] T.Fuku, A.Namatame, T.Kaizouji. *Collective Efficiency in Two-Sided Matching.* En: Beckmann M. et al. (eds) Artificial Economics. Lecture Notes in Economics and Mathematical Systems, Vol.564: 115-126 2006.

[5] D.Gale, L.S.Shapley. *College Admissions and the Stability of Marriage.* The American Mathematical Monthly, Vol.69, Nº1: 9-15. 1962.

[6] J.Gross, J.Yellen. *Discrete Mathematics and Its Applications.* CRC Press. 2003.

[7] J.M.Harris, J.L.Hirst, M.J.Mossinghoff. *Combinatorics and Graph Theory.* Springer. 2008.

[8] F.Havet. Combinatorial Optimization, Lecture notes. International Masters in Computer Science, Sophia-Antipolis (s.f.). Disponible en `http://www-sop.inria.fr/members/Frederic.Havet/` (Fecha de recuperación: 17-09-2020).

[9] J.Hidakatsu. *Structure of the Stable Marriage and Stable Roommate Problems and Applications.* Trabajo Fin de Master, University of Michigan. 2014. Disponible en `https://scholarcommons.sc.edu/cgi/viewcontent.cgi?article=4769&context=etd` (Fecha de recuperación: 09-10-2020).

[10] R.W.Irving. *An Efficient Algorithm for the "Stable Roommates"Problem.* Journal of Algorithms, Vol.6, 577-595. 1985.

[11] R.W.Irving, K.Iwama, D.F.Manlove, S.Miyazaki, Y.Morita. *Hard variants of stable marriage.* Theoretical Computer Science, Vol.276: 261-279. 2002.

[12] K.Iwama, S.Miyazaki. *A Survey of the Stable Marriage Problem and its Variants.* Kyoto University. Informatics Education and Research for Knowledge-Circulating Society. 131-136. 2008.

[13] D.E.Knuth. *Stable Marriage and Its Relation to Other Combinatorial Problems.* CRM. 1997.

[14] L.Lovász, M.D.Plummer. *Matching Theory.* North-Holand. 1986.

[15] J.P.Mínguez. *Algoritmo de Gale-Shapley. Variaciones y alternativas.* Trabajo Fin de Grado, Euskal Herriko Unibertsitatea. 2014. Disponible en `https://addi.ehu.es/handle/10810/14861` (Fecha de recuperación: 09-10-2020).

[16] O.Muiño. *El problema de asignación.* Trabajo Fin de Master, Master en Técnicas Estadísticas. Universidad de Vigo. 2016. Disponible en `http://eio.usc.es/pub/mte/descargas/ProyectosFinMaster/Proyecto_1368.pdf` (Fecha de recuperación: 09-10-2020).

[17] J.Ortega. Estabilidad en emparejamientos. Trabajo Fin de Grado, Universidad Politécnica de Madrid. 2017. Disponible en `http://www.dma.fi.upm.es/personal/gregorio/grafos/emparejamiento_estable/documents/Memoria.pdf` (Fecha de recuperación: 09-10-2020).