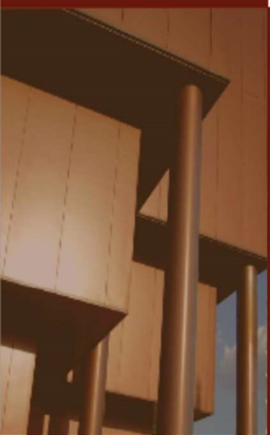
Trabajo Fin de Máster
Máster Universitario en Ingeniería Aeronáutica

Análisis del problema de rotación de aviones para una compañía mediana

Formato de Publicación de la Escuela Técnica Superior de Ingeniería de Sevilla

Autor: Tomás Sánchez Sánchez

Tutor: José María del Castillo Granados



Dep. Ingeniería y Ciencia de los materiales y del Transporte
Escuela Técnica Superior de Ingeniería

Sevilla, 2018

Trabajo Fin de Máster
Máster Universitario en Ingeniería Aeronáutica
Anáisis del problema de rotación de aviones para una compañía mediana

Formato de Publicación de la Escuela Técnica Superior de Ingeniería de Sevilla

Autor:

Tomás Sánchez Sánchez

Tutor:

José María del Castillo Granados

Profesor titular

Dep. de Ingeniería y Ciencia de los materiales y del Transporte

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2018

Trabajo Fin de Máster: Formato de Publicación de la Escuela Técnica Superior de Ingeniería de Sevilla

Autor: Tomás Sánchez Sánchez

Tutor: José María del Castillo

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2018

El Secretario del Tribunal

Agradecimientos

Agradecerselo a mis padres y a mi hermano que han estado ahí siempre que lo he necesitado. Este es el principio de una nueva etapa de mi vida, conseguido con esfuerzo y trabajo.

Agradecer también a José María del Castillo la oportunidad de profundizar en el mundo del transporte aéreo, así como la ayuda recibida en su realización.

Gracias por todo.

Resumen

El objetivo del presente trabajo es mostrar un algoritmo de resolución del método 'Aircraft Routing'. Es decir, la finalidad es obtener las rotaciones que realizarán los aviones de una determinada compañía para poder realizar todos los vuelos programados sin repetir ninguno, optimizando las funciones objetivo que sean requeridas. Además, se mostrarán ejemplos con empresas y horarios reales. Finalmente, se realizarán retrasos y adelantos en los horarios de los vuelos para observar el efecto que estos tendrían en las rotaciones obtenidas, y las posibles mejoras que estos acarrearían.

Abstract

The main goal of this project is to show an algorithm to solve the 'Aircraft Routing' problem, which means that there will be calculated the possible rosters of a company to make all the flights without repeating any of them. The rosters will be optimized following the objective function required. Furthermore, some examples will be hold with real companies and slots. Finally, arrivals and departures will suffer advancements and delays to show the effect on the rosters, and the imporvements that the changes will carry on them.

Índice

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

Notación

A	Compañía de estudio
B	Nº de aeropuertos
M	Nomenclatura de un aeropuerto de salida
N	Nomenclatura de un aeropuerto de destino
E	Nomenclatura de un aeropuerto de destino,2
C	Nº de vuelos entre aeropuertos
i	Nodo de salida
j	Nodo de llegada
l	Segundo nodo de llegada
k	Día de estudio
S	Matriz de salidas
LL	Matriz de llegadas
C_{Mmax}	Nº de asignaciones de un aeropuerto
α	Vector de salidas-llegada perteneciente a un nodo con más de una asignación
T	Tiempo de estancia en tierra
LLR	Matriz de rotación de llegada formado por dos nodos
SR	Matriz de rotación de salida formado por dos nodos
LLF	Matriz de rotación completa de llegada
SF	Matriz de rotación completa de salida
RF	Matriz de rotación completa
a	Nodos de asignación principal
b	Nodos de asignación secundaria
LC	Vector de posicionamiento de cambio de asignaciones
CC	Vector de valor de los nodos cambiados de asignación
β	Nodo de salida fijo
γ	Nodo de llegada fijo
F	Posición en el vector de rotación del último nodo de la rotación
F	Función objetivo
RT	Matriz de rotaciones con duración concreta
k_R	Días de duración de la rotación de estudio
X	Variable de decisión, si se realiza o no una rotación en concreto
r	Nº de rotaciones total
RI	Matriz de restricciones de desigualdad
P	Nº de aviones disponible
RD	Matriz de restricciones de igualdad
W	Peso de la función objetivo

1 INTRODUCCIÓN

En la industria del transporte aéreo, cobra especial importancia la mejora de los procesos, de cara a optimizar el movimiento de aeronaves por el espacio aéreo así como dar un servicio lo más eficiente posible al cliente. Por ello, a lo largo de los años se han desarrollados diversos métodos que permiten encontrar la solución que mejor se adapta a las necesidades de cada empresa, según los intereses de cada una. En el presente trabajo, se va a desarrollar el método de Aircraft Routing, se desglosará paso a paso mostrando como implementarlo en un software, en este caso *Matlab*. Además, se aportarán casos reales para poder entender con mayor facilidad el proceso seguido y el alcance del método. El método de Aircraft Routing se puede subdividir en los apartados expuestos a continuación, a partir de los cuales se desgranarán las diversas ecuaciones matemáticas usadas para obtener un resultado satisfactorio.

1. Obtención de datos.

Para poder realizar el problema primero será necesario obtener datos reales de las salidas y llegadas de diversos vuelos correspondientes a una aerolínea en concreto. Siendo estrictos esto no pertenece al método de Aircraft Routing, sino que una vez se tienen la información de los vuelos, entonces puede empezarse a operar con dichos datos.

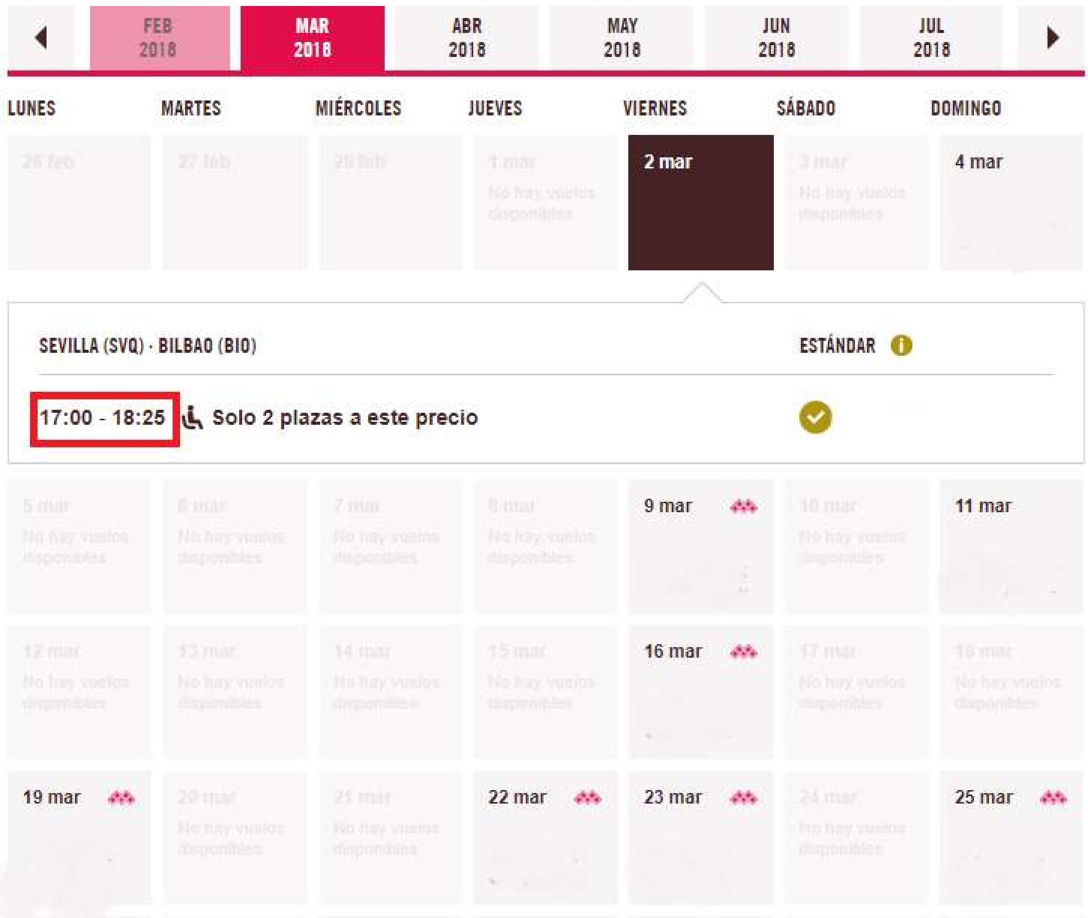


Figura 1: Obtención de datos, referencia [1]

2. Cálculo de rotaciones.

Una vez conocidos los vuelos, se deberá ver si la concatenación de estos es posible. En caso afirmativo, se buscarán todas las combinaciones viables. Como resultado se tendrán todas las trayectorias reales posibles que puede realizar una aeronave con los horarios de la compañía de estudio. En la **Figura 2** se muestra un ejemplo de conexión en un sistema de aeropuertos (A, B, C, D, E, …), donde habrá que tener en cuenta los horarios de los vuelos.

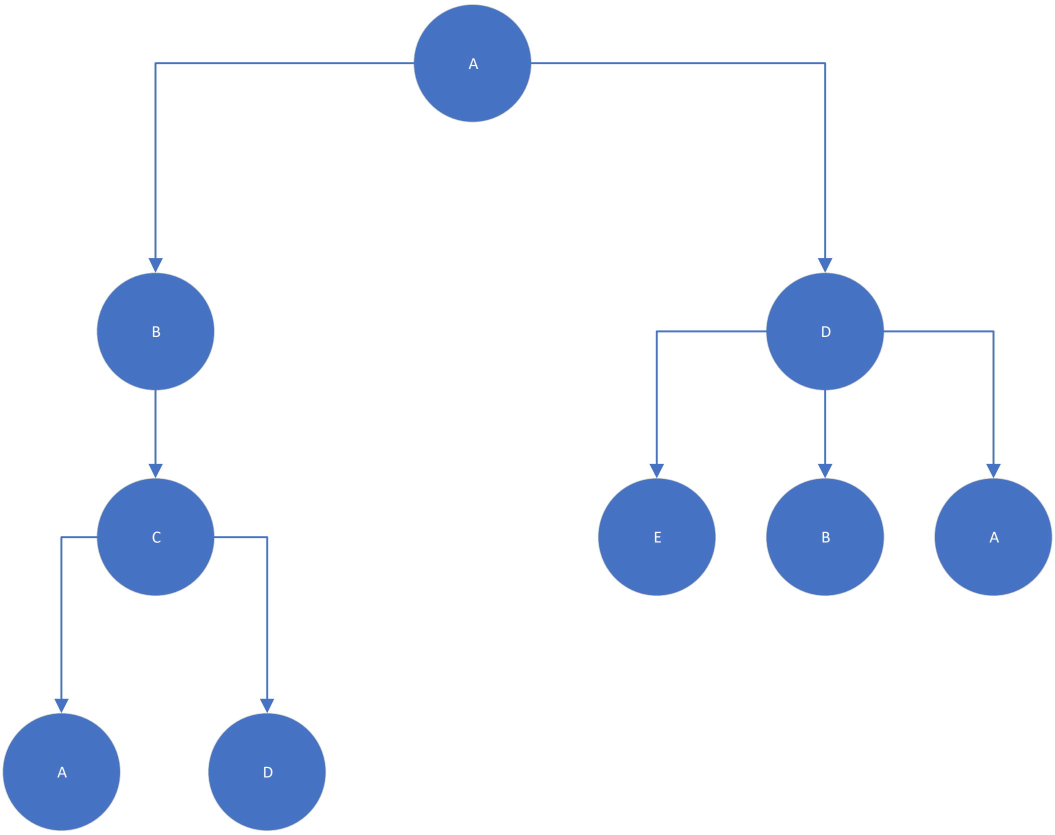


Figura 2: Diagrama de conexión entre aeropuertos

3. Selección de rotaciones según criterios impuestos por la empresa.

Cada compañía 'repite' las rotaciones cada cierto número de días (1,2, 3…). Esto quiere decir que en el caso de que las rotaciones se repitan cada 2 días, por ejemplo, de todas las rotaciones calculadas en el apartado anterior, sólo se mantendrán aquellas cuyo destino final en la noche del día 2 sea el mismo que el origen de la mañana del día 1. El concepto 'repite' no es del todo exacto, pues la compañía puede, y de hecho es una práctica habitual, cambiar los horarios y los vuelos según el día y las necesidades. Lo que quiere decir es que una rotación de un día, la aeronave iniciará el día en un aeropuerto A y terminará el día, pasará la noche, en el mismo aeropuerto A. Para rotaciones de dos días, como ya se ha mencionado, la noche intermendia la pasará en un aeropuerto distinto del original. Las rotaciones de tres días, suelen ser combinación de las rotaciones de uno y dos días concatenadas.

4. Agrupación de rotaciones.

Una vez se ha hecho una criba acerca de las combinaciones que satisfacen los requisitos de la empresa de estudio es necesario ver qué rotaciones, combinándolas entre sí permiten completar el mapa de vuelos, es decir, que conjunto de rotaciones permiten realizar todos los vuelos sin que se repita ninguno. Se obtendrán varias posibles combinaciones y todas son soluciones al problema.

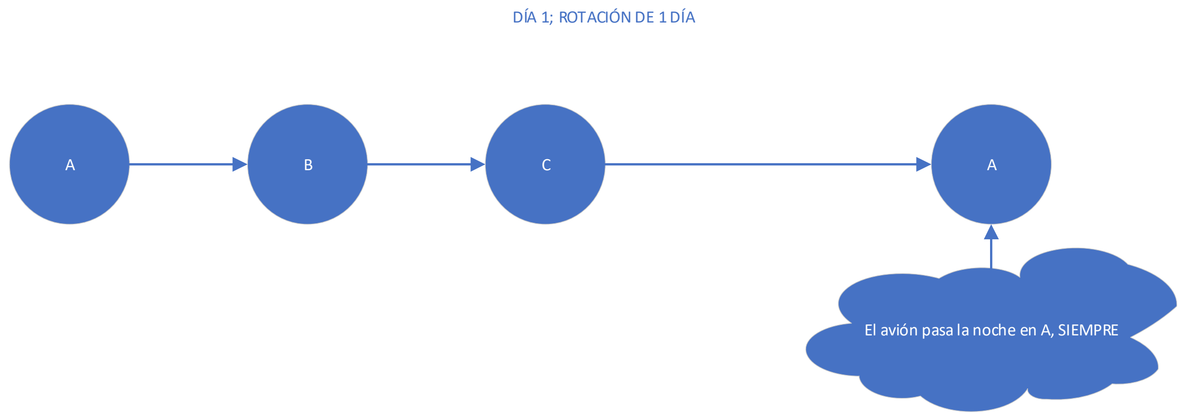


Figura 3:Rotación de 1 día

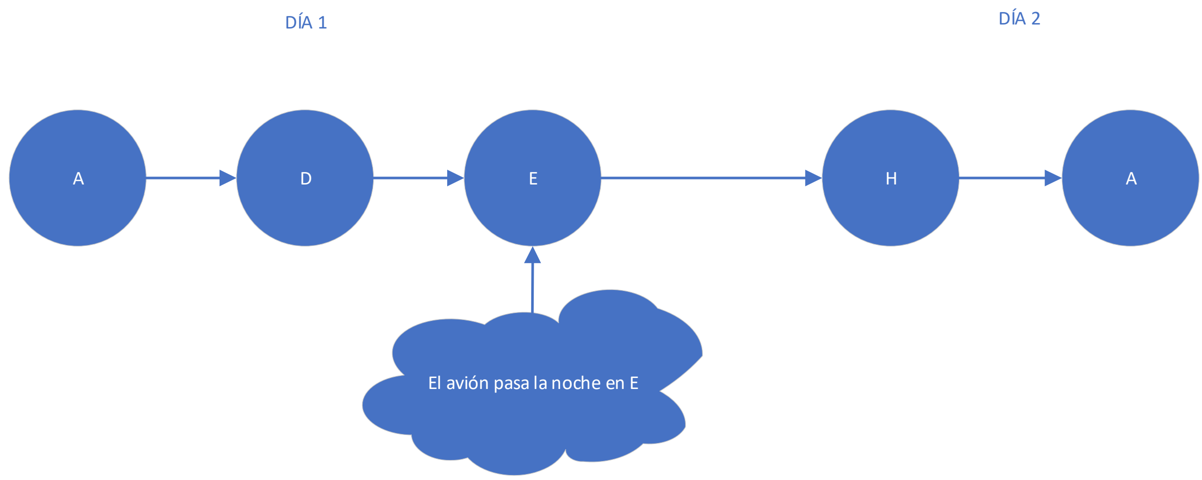


Figura 4:Rotación de 2 días

5. Optimizar el resultado.

Este apartado puede tener varias soluciones. Todo depende de lo que se quiera optimizar. Un objetivo puede ser que la empresa tenga ciertos aeropuertos base en los que prefiera que sus empleados pasen la noche, en lugar de en otros que sólo son escalas. Pues se asignarán diversas ponderaciones a cada rotación para ver de todas las combinaciones de rotaciones posibles, cuál mejora la solución al problema. Puede darse el caso de que halla más de un objetivo en cuestión. En dicho caso se obtendrá el máximo absoluto de la solución, o lo que es lo mismo, una solución que maximice (o minimice, según corresponda) el conjunto y no sólo una variable del problema.

Tanto el apartado 4 como el 5 se harán de forma conjunta mediante el uso de métodos de programación lineal.

6. Proposición de mejora.

Por último, se tiene un algoritmo que permite variar los horarios de los vuelos un determinado periodo de tiempo (se entiende que relativamente corto) y observar que cambios se producen en las posibles combinaciones de los vuelos. O sea, aconsejar posibles cambios que mejoren las posibilidades existentes de cara al futuro.

2 ESTADO DEL ARTE

Como punto de partida se ha realizado un estudio de los procedimientos usados a día de hoy en este campo, así como los algoritmos más utilizados. Esto permitirá tener una visión global del problema al que nos enfrentamos en este trabajo y poder abordarlo con mayor criterio. Sin embargo, la intención no es emular otro algoritmo ya existente sino adquirir los conocimientos suficientes como para poder realizar un algoritmo propio de resolución del método de Aircraft Routing.

La optimización de las rutas seguidas por las aeronaves de una compañía es un problema a la par importante y complejo, de manera que existen multitud de artículos versando sobre este tema. El objetivo de todos ellos es, con distintos métodos, obtener aquellas soluciones que sean óptimas en el menor tiempo posible de computación. Es importante entender que cuando una aeronave se queda en tierra durante más tiempo del necesario para repostar y dar entrada y salida a los pasajeros, es debido a que en el mismo se están realizando tareas de mantenimiento. En ([2], [3]) , explica como este mantenimiento debe estar de acuerdo con la normativa de la FAA donde se distinguen comprobaciones de la aeronave mayores y menores, según la exhaustividad de las mismas.

Aquellas que sean mayores requieren de mayor dedicación y obligan a la aeronave a permanecer en tierra más tiempo. El primero de estos chequeos suele hacer alrededor de una vez a la semana (66 horas de vuelo). A la hora de imponer horarios de vuelo habrá que tener en cuenta la restricción del mantenimiento, que obligará a mantener en tierra aviones, lo que supone pérdida de beneficios a la compañía, pues el objetivo es que un avión en concreto esté en vuelo el mayor tiempo posible. Para determinar el método de Aircraft Routing, al partir de unos horarios ya definidos, se supone que la compañía ha realizado un estudio previo para poder cumplir con dichos requisitos.

Sin embargo, al igual que se plantea en ([2], [3]), pueden existir tareas de mantenimiento no programadas. Por ejemplo, un impacto de pájaro que obliga a la aerolínea a mantener durante unos minutos extra a la aeronave. O incluso, sin necesidad de que sea un mantenimiento, puede haber retrasos debido a que los slots no se estén cumpliendo en un aeropuerto en concreto. Esto producirá un cambio en los vuelos en cadena, y en mayor medida si se trata de 'Hubs'. Para ello, habrá que dimiensionar el sistema para que, aún habiendo ciertos retrasos, estos no se propaguen de manera incontrolada. Esto se consigue aumentando el tiempo T que tarda un avión entre su aterrizaje y el posterior despegue. Aquellas ventanas de despegue donde dicho tiempo T sea menor, podrán verse afectadas en mayor medida, aunque pueden disiparse si en una rotación en concreto existen tiempos mayores en otros vuelos posteriores que permitan acumular dicho retraso y no impactar en el desarrollo normal de la aerolínea. Este tiempo T contempla la bajada de pasajeros,la subida de los mismos así como las tareas de mantenimiento prevuelo que hay que cumplir para asegurar que todos los sistemas están a punto y no hay ningún defecto que impida la operatividad de la aeronave en cuestión.

En [3] se muestra como en la función objetivo, a minimizar, se pueden usar parámetros a priori desconocidos que penalizan los retrasos. En el algoritmo de este trabajo se ha realizado un enfoque parecido jugando con el valor T. Cuanto mayor es dicho parámetro T globalmente en cada rotación, menor será la probabilidad de retrasos en los vuelos indicados.

Como se indica en [4], en torno a un 30% de los vuelos que requieren un mayor tiempo de mantienimiento por motivos meteorológicos y de colas, se vieron retrasados en el tiempo, y hasta un 10% por más de un cuarto de hora, lo que quiere decir que controlar dicho tiempo T no es sencillo y fijarlo requiere de un estudio intensivo del sistema para comprobar como se puede propagar el retraso, sobre todo si los pasajeros necesitan hacer trasbordo en algún lugar. De ahí que si se puede conseguir que las rotaciones contemplen vuelos que permitan realizar todos sin necesidad de realizar trasbordos, o al menos, realizando el menor número de ellos, mejor.

Para evitar estos retrasos, la mejor manera es añadir 'buffers' a aquellos vuelos que se conocen por experiencia que suelen sufrir más veces retrasos, así se aportará mayor estabilidad al sistema y sobre todo, se mejorará el servicio al cliente pues el principal objetivo es atraer a pasajeros que quieran volar en la compañía en concreto no solo una vez, sino cada vez que hagan uso del transporte aéreo.

Además, aunque el método de Aircraft Routing no lo tenga en cuenta, también es un factor relevante el hecho de que los aviones deberán cambiar las rutas según la climatología, solamente predecible al corto plazo, de forma que no se puede saber de antemano la ruta exacta que realizará la aeronave. Para ello, según [5], existen métodos estocásticos que permiten conocer con algo más de antelación las rutas reales nodo a nodo que seguirá la aeronave. Sin embargo, este enfoque, aunque necesario para llevar a cabo en su totalidad el método de Aircraft Routing, se escapa del contenido aquí presente pues es necesario tener datos medioambientales en tiempo real para poder realizar la propagación, y esto no afecta a la ruta en sí, sino a la trayectoria. Si es cierto que algunos vuelos se pueden ver retrasados o incluso suspendidos por el clima, lo que afectaría a todos los vuelos que realiza la aronave.

En cuanto al método resolutivo, es decir, el método usado para la optimización de rotaciones, nos podemos encontrar con variedad de ejemplos, con el objetivo todos ellos de reducir el tiempo de ejecución obteniendo aquel resultado que mejore las prestaciones y que cumpla con todos los requisitos impuestos ya sea por la compañía o por la normativa. En particular, uno de los métodos más usados es el PSO (Particle Swarm Optimization) [4]. Por otro lado, la metodología usada en este proyecto se basa en el uso de flujos estáticos, donde se ve la viabilidad o no de el roster en función de los valores matriciales de llegadas y salidas. Sin embargo, existen otros algoritmos que incluyen lo que se denomina como flujo dinámico [6], que implica que a medida que pasa el tiempo el estado en que se encuentra el espacio aéreo cambia, y puede interesar modificar determinadas combinaciones puntuales, para ir adaptando la capacidad tanto de la compañía aérea como del espacio aéreo y así adaptar el sistema.

Como se puede ver, no es un problema trivial y existen multitud de métodos para resolverlo, se han mencionado dos. Esto se debe a la importancia capital que supone para las aerolíneas optimizar las rutas seguidas por sus aeronaves. El coste de operación de las aeronaves asciende a un 25% del total de una compañía [7], lo que significa que interesa que todo ese tiempo de operación sea en vuelo y no en tierra de forma que se aproveche al máximo la utilización de la aeronave. Sin embargo, como se ha mencionado anteriormente se añadirán buffers para evitar retrasos aunque eso conlleve en muchos casos pérdida de eficiencia en la utilización, reduciendo el riesgo de congestión en una ruta determinada transmitiendo el retraso y generando cuellos de botella.

El mayor problema reside en aquellos aeropuertos que son ‘Hub’ donde aparecen los mayores retrasos pues habrá trasbordos y una mayor afluencia de vuelos. De ahí que existan estrategias como la descentralización de las rutas [8], repartiendo así la congestión y reduciendo el riesgo en determinados aeropuertos. A priori se podría pensar que es una estrategia arriesgada, pero si que es cierto que evita que un aeropuerto sólo pueda afectar a un gran número de vuelos para una misma compañía.

En definitiva, existen variedad de métodos e hipótesis a la hora de resolver un problema tan importante y complejo como es obtener las rtuas óptimas de las aeronaves. El ojetivo de este trabajo no es más que aportar dos ejemplos de empresas como resultado de la aplicación del método de Aircraft Routing usando un algoritmo propio y explicando el desarrollo conceptual del mismo para así entender como se ha llegado a las soluciones reales con un tiempo de computación bajo.

3 Empresas de estudio

Aunque el desarrollo matemático es válido para cualquier empresa y se realizará la explicación de forma genérica, es conveniente el uso de una empresa real para poder mostrar los resultados y aportar datos feacientes de que el método funciona, así como evitar posibles malentendidos derivado de las abstracciones matemáticas usadas en el proceso.

3.1. Canaryfly

Canaryfly es una empresa de aviación que opera vuelos regulares únicamente en las islas Canarias. La sede social de esta compañía se encuentra en Gran Canaria, en la base aérea de Gando, aeropuerto de Gran Canaria. La posición sectorial en el año 2016 era la 16º. Esta clasificación permite ver como Canaryfly es una compañía regional, donde sus condiciones geográficas de operación afectarán al entendimiento de los resultados obtenidos.

Posición Sector	Evolución Posiciones	Nombre de la empresa	Facturación (€)	Provincia
11	2 ↓	CANARIAS AIRLINES COMPAÑIA DE AVIACION, SOCIEDAD LIMITADA	53.527.214	Tenerife
12	2 ↓	PRIVILEGE STYLE SA	49.841.000	Baleares
13	2 ↓	GESTAIR SA	49.200.646	Madrid
14	7 ↑	AERONOVA SL	44.449.123	Baleares
15	1 ↓	HABOCK AVIATION SL.	grande	Madrid
16	1 ↑	**CANARY FLY SL**	**grande**	**Palmas (las)**

Figura 5: Posición sectorial de Canaryfly, 2017, [8]

Tiene 6 destinos posibles (El Hierro, Tenerife Norte, Fuerteventura, Gran Canaria, La Palma y Lanzarote). Comenzó a operar en el 2008 y posee tres aeronaves ATR72-500. Para poder tratar de forma numérica dichos aeropuertos, se les asignarán una serie de valores principales (1, 2, …,6 respectivamente), a los que habrá que añadir más asignaciones debido a la aparición de más de un vuelo diario entre el par origen-destino.

Tabla 1: Asignación numérica de los aeropuertos correspondientes a Canaryfly

Aeropuerto	Valor numérico
El Hierro	1
Tenerife Norte	2 7 8 20
Fuerteventura	3 9 10
Gran Canaria	4 11 12 13

La Palma	5 14 15 16
Lanzarote	6 17 18 19

3.2. Volotea

Volotea es una empresa de aviación de gran tamaño que opera vuelos regulares entre 80 destinos a lo largo de toda Europa. Esta compañía opera principalmente en verano, viendo reducido el número de vuelos, y también de pasajeros, en las temporadas de invierno. Este dato permitirá comprender los resultados obtenidos durante el proceso. Volotea consta de 28 aviones A319 y B717, aunque para el estudio se supondrá que sólo opera con un único modelo de avión. Esta suposición no dista tanto de la realidad pues Volotea espera operar sólo con aviones de la gama AIRBUS en los próximos años. Las bases de la compañía se sitúan en Nantes, Burdeos y Toulouse. La sede social de la compañía se sitúa en Barcelona.

Posición Sector	Evolución Posiciones	Nombre de la empresa	Facturación (€)
1	0	IBERIA LINEAS AEREAS DE ESPAÑA SOCIEDAD ANONIMA OPERADORA	4.090.000.000
2	0	VUELING AIRLINES, SA	2.027.258.000
3	0	AIR EUROPA LINEAS AEREAS SA	1.790.966.000
4	0	AIR NOSTRUM LINEAS AEREAS DEL MEDITERRANEO SA	425.462.000
5	0	COMPAÑIA OPERADORA DE CORTO Y MEDIO RADIO IBERIA EXPRESS SA.	409.660.000
6	0	**VOLOTEA SA**	**250.529.000**

Figura 6: Posición nacional Volotea, 2017, [7]

El estudio se realizará para el mes de noviembre de 2017, de manera que habrá destinos sin cobertura durante dicho periodo. Esos aeropuertos no se tendrán en cuenta, y se verá reducida la asignación.

Tabla 2: Asignación numérica de los aeropuertos correspondientes a Volotea

Aeropuerto	Valor numérico	Aeropuerto	Valor numérico
Sevilla	1	Santorini	26
Asturias	2	Praga	27
Bilbao	3	Ajaccio	28
Santander	4	Bastia	29

Fuerteventura	5	Brest	30
Alicante	6	Caen	31
Gran Canaria	7	Estrasburgo	32
Lanzarote	8	Lille	33
Madrid	9	Marsella	34
Mallorca	10	Montpellier	35
Málaga	11	Paupirineos	36
Tenerife	12	Perpiñan	37
Valencia	13	Toulouse	38
Alguer	14	Trieste	39
Ancona	15	Turín	40
Bari	16	Venecia	41 49
Cagliari	17	Verona	42 43
Catania	18 19	Burdeos	44
Génova	20 21	Nantes	45
Nápoles	22 23	Palermo	46 47
Viena	24	Niza	48
Atenas	25		

4 OBTENCIÓN DE DATOS

Lo relevante no es conocer los horarios de salidas y llegadas de la compañía (es importante, pero una vez conocidos no suponen un cuello de botella), sino el tratamiento de estos para poder operar con ellos de forma adecuada, fácil y lo que es aún más importante, rápidamente. La combinatoria, base matemática en la que se sustenta el método de 'Aircraft Routing', exige control sobre el tiempo de ejecución de los programas, pues cuanto mayor sea el número de vuelos diarios que realiza una empresa, mayores son las posibilidades de combinación y, con ello, aumenta el tiempo que las herramientas usadas para el tratamiento de datos, en este caso *Matlab,* tarda en tratar los datos para sacar un resultado. Por ello, en los apartados siguientes se mostrará la manera correcta de almacenar los horarios de vuelos para poder ejecutar los programas asociados.

4.1. Unidad horaria

Los clientes y todos los compradores potenciales de billetes para volar con una compañía seleccionan su vuelo pensando qué día les será posible volar. Todo el mundo suele pensar en un horario de 24 horas, y por ello, las compañías también lo hacen. En el proceso seguido en el presente trabajo se ha usado como unidad horaria las 24 horas de un día. Indicar que se podría haber usado otra distinta en el proceso, pero si se hubiese escogido 48 horas, dos días, no se podrían calcular las rotaciones de un día de duración, sino que como mínimo se podrían obtener aquellas con dos días de duración.

4.2. Definición de nodos

Sea una empresa A, que opera en un número de aeropuertos B. Si B es pequeño se podría pensar que se trata de una empresa pequeña, fácil de manejar. Este concepto es importante, pues erróneamente B no es indicativo a priori del tamaño ni de la complejidad de nuestro estudio. Todo avión que despega de un sitio M obligatoriamente llega a otro N, siendo $\mathbf{M \neq N}$. Esto quiere decir que cada vuelo genera dos nodos, uno de salida (MN_1) y otro de llegada (NM_1). En el caso de que haya más de un vuelo en la unidad horaria definida, C vuelos entre dos aeropuertos M y N, existirán **2C** nodos (MN_1, MN_2, …, MN_C, NM_1, NM_2, …, NM_C) asociados a M y N. Esto puede complicarse aún más, cuanto mayor sea el número de nodos, recalcando que C es totalmente independiente de B, y es C (número de vuelos) el que determina el tamaño de los datos a evaluar y no B (número de aeropuertos). El concepto de aeropuerto y nodo se puede visualizar en **Figura 7**.

4.3. Tratamiento matricial

Para poder tratar matemáticamente un aeropuerto, es necesario asignarle un valor numérico a cada uno. Sin embargo, después de conocer que un aeropuerto puede llegar a tener más de un nodo, lo más probable es que sea necesario asignarle más de un valor numérico, como ya se indicó anteriormente para Volotea y Canaryfly. La cuestión es cual es la forma más eficiente de hacer frente a todos los nodos que se pueden generar con B aeropuertos. Para poder comprender mejor el proceso, supongamos por un momento que sólo existe un vuelo entre M y N en cada unidad horaria. En ese caso, un vuelo quedará definido por su hora de llegada a N ($\mathbf{LL_{MN}}$) y la de salida de M ($\mathbf{S_{MN}}$). Si hacemos lo mismo para los B aeropuertos, se obtendrán dos matrices, una de salidas(S) y otra de llegadas(LL) donde las filas indicarán el aeropuerto de salida y las columnas las de llegada, siendo matrices cuadradas en todos los casos. Es decir, la posición (i, j) en la matriz de salidas se corresponderá con el horario de salida desde el aeropuerto i con destino j, y en la matriz de llegadas, estará indicado la llegada desde i con destino j. A su vez la posición (j, i), tendrá como origen j y destino i.

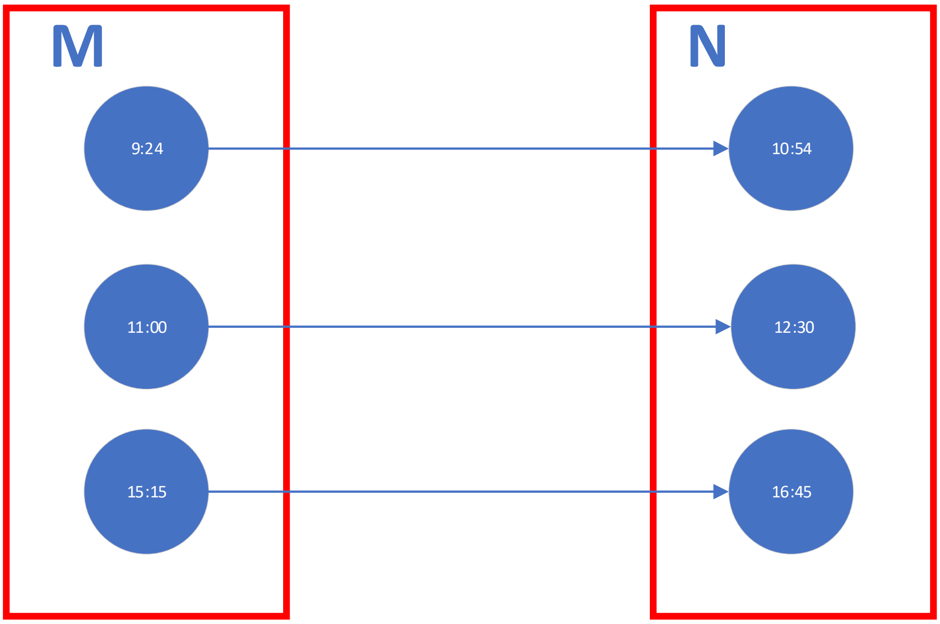


Figura 7: Ejemplo de sistema con B = 2 y C = 6

Si representamos un esquema general de la forma que adquirirán las matrices, se obtendrá el resultado mostrado en la ecuación **(3-1).**

$$\mathbf{S} = \begin{bmatrix} \mathbf{S_{11}} & \dots & \mathbf{S_{1B}} \\ \dots & \mathbf{S_{ij}} & \dots \\ \mathbf{S_{B1}} & \dots & \mathbf{S_{BB}} \end{bmatrix}, \quad \mathbf{LL} = \begin{bmatrix} \mathbf{LL_{11}} & \dots & \mathbf{LL_{1B}} \\ \dots & \mathbf{LL_{ij}} & \dots \\ \mathbf{LL_{B1}} & \dots & \mathbf{LL_{BB}} \end{bmatrix} \tag{4–1}$$

Cuando no existe ningún vuelo entre los aeropuertos (i, j), al lugar de la matriz definido por la pareja se le dará un valor nulo (0). De antemano se puede inferir que cuando i = j, el valor será nulo pues no existen vuelos de M a M, un aeropuerto no puede operar consigo mismo. Demos el siguiente paso, recordemos que puede existir más de una salida y llegada entre M y N, o lo que es lo mismo, $\mathbf{LL_{MN}}$ y $\mathbf{S_{MN}}$, tal y como la hemos definido, no es única, sino que poseería más de un valor. Por ello, se debe retomar la definición de nodo. Por simplicidad se usarán sólo los nodos de salida, pero se puede realizar el mismo desarrollo con los de llegada. Si se estudian los nodos obtenidos para un sólo aeropuerto M, se tienen la siguiente lista (M_{1-1}, M_{1-2}, …, M_{1-C_1}; M_{i1}, M_{i2}, …, M_{i-C_i}; M_{B1}, M_{B2}, …, M_{B-C_B}). Como se puede observar el número de nodos total (C_M) de salida vendrá dada por la expresión **(3-2)**. Donde C_i representa el número de vuelos que habrá en una unidad horaria entre dos aeropuertos M y N.

$$C_M = \sum_{i=1}^{B} C_i \tag{4–2}$$

Si se observan las matrices de llegadas y salidas, cada fila de esta contendrá **B-1** nodos como máximo (no B pues un aeropuerto no puede operar consigo mismo como ya se expuso anteriormente), con la salvedad de que por cada fila sólo puede haber un nodo asociado al par llegada-salida (**M, N**). Esto quiere decir que el número de filas necesarias para poder expresar los C_M nodos es igual al máximo de $\mathbf{C_i}$. El número de valores asignados a un aeropuerto M será entonces el dado por la expresión, Max (C_1, C_2, …, C_i, …, C_B), así para cada aeropuerto. Sea $C_{M\text{-}max}$ al número de valores asignados al aeropuerto M, donde M=1, 2, …, B, entonces el tamaño real de las matrices de llegada y salida será el mostrado en (3-3), recordemos que son matrices cuadradas para cada unidad horaria k. Al ser una expresión matricial, se generarán nodos que realmente no existen, no todas las combinaciones (i, j) tienen vuelos, y por ello esos huecos se rellenarán con ceros (0). Se trata de matrices tridimensionales si lo que se requiere es el estudio de más de una unidad horaria.

$$LL = [\sum_{i=1}^{B} C_{i-max}, \sum_{i=1}^{B} C_{i-max}, D] \qquad (4\text{–}3)$$

$$S = [\sum_{i=1}^{B} C_{i-max}, \sum_{i=1}^{B} C_{i-max}, D] \qquad (4\text{–}4)$$

En esta definición hay que matizar algunos detalles importantes. Supongamos un aeropuerto M que tiene asignados un conjunto de valores debido a que existe más de un vuelo diaria entre el mismo y otro aeropuerto del sistema. Aunque las salidas asociadas a cada fila sean distintas, pues ya se ha dicho que el objetivo de esta formulación es la repetición de vuelos entre un mismo origen y destino en la unidad horaria seleccionada, las llegadas deben ser las mismas. En definitiva, aunque sean valores distintos, todos están asociados al mismo aeropuerto, de manera que las llegadas a un nodo determinado deben ser las mismas. Si los valores de llegadas fueran distintos significaría que no se están contemplando las mismas llegadas para un mismo aeropuerto, y eso es una incongruencia. Esto implica que habrá cierta repetitividad en los datos, necesaria para que la formulación sea correcta. La formulación final se muestra en **(3-5)** donde α_1 y α_2 son vectores que indican la igualdad en las columnas correspondientes a un mismo aeropuerto. Se recalca que, aunque las columnas sean iguales, las filas deben ser distintas, pues ese es el objetivo final de la asignación múltiple de valores a un aeropuerto M. Cabe destacar que las matrices LL y S son tridimensionales, pues lo que se muestra aquí no es más que los horarios de vuelos para una unidad horaria determinada. La tercera dimensión contempla las unidades horarias que el usuario estime oportuno, es decir, la variable D designa el número de unidades horarias empleadas en el estudio.

$$S = \begin{bmatrix} S_{11}=0 & 1i & \dots & S_{iC_{(i-1)-max}} = S_{1i} & \dots = S_{1i} & S_{iC_{(i)-max}} = S_{1i} & \dots & S_{1B} \\ \dots & \alpha_1 & \dots & \alpha_1 & \alpha_1 & \alpha_1 & \dots & \dots \\ S_{j1} & S_{ji} & \dots & S_{jC_{(i-1)-max}} = S_{ji} & \dots = S_{ji} & S_{jC_{(i)-max}} = S_{ji} & \dots & \dots \\ \dots & \alpha_2 & \dots & \alpha_2 & \alpha_2 & \alpha_2 & \dots & \dots \\ S_{B1} & S_{Bi} & \dots & S_{iC_{(i-1)-max}} = S_{Bi} & \dots = S_{Bi} & S_{iC_{(i)-max}} = S_{Bi} & \dots & S_{BB}=0 \end{bmatrix} \qquad (4\text{–}5)$$

4.4. Caso práctico: Canaryfly y Volotea

Una vez se ha definido la forma de ordenar los horarios de vuelos y designar a los aeropuertos de operación, se mostrará un caso real para su aplicación. Es en la pagina principal de las compañías donde se debe realizar la recopilación de datos, mostrándolos en *Excel* siguiendo las indicaciones dadas en apartados anteriores y por ello se pretende dar ejemplo por medio de Canaryfly para conocer en mayor medida el procedimiento. Ya se definió la empresa Canaryfly en apartados anteriores, **(2.1)**. Los datos de vuelos se obtuvieron de la web de la empresa. Para organizar dichos datos de forma fácil, se realizará por medio de *Excel*, como ya se ha mencionado. El estudio se va a realizar para el mes de octubre de 2017 donde Canaryfly oferta los mismos horarios de vuelos para cada día del mes. De esta forma, las matrices S y LL serán más fáciles de montar pues la dimensión D es idéntica para cada unidad horaria, donde se ha escogido como unidad horaria 24 horas por simplicidad. Se recomienda ver los anexos donde se muestra cómo quedaría la matriz en *Excel*. Sin embargo, hay un matiz, y es que *Matlab* opera mucho mejor con decimales que en horas. Para ello Excel posee formulación a partir de la cual se puede pasar los minutos a decimal. Para ello, y con el objetivo de evitar confusiones, se muestra paso a paso como se ha hecho uso del archivo.

1. En un archivo Excel en blanco, se crean dos hojas distintas por cada unidad horaria que se pretenda estudiar. El nombre asignado a cada hoja es indiferente, por simplicidad se le asigna el del día de estudio y si está en decimal o en minutos.

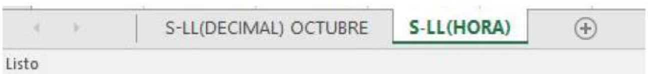


Figura 8: Nombres en hojas de Excel

2. En la hoja donde queramos indicar los horarios en minutos, se escriben las matrices LL y S en minutos

tal y como se indica en los anexos.

3. En la hoja donde queramos indicar los horarios en decimales, se deberá tener especial cuidado en hacer coincidir las celdas de las matrices S y LL con las mismas matrices en minutos. Si no se hace así no ocurrirá nada, pero la posibilidad de inducirnos a error es muy elevada. Conviene ser metódicos para evitar posibles fallos. Una vez creado el marco de las matrices hay que rellenarlas con los datos correspondientes, en este caso sólo hay que transformar los datos en minutos a decimal. Si lo hemos hecho correctamente, los horarios de salida comenzarían en la celda B2. En dicha celda hay que escribir la formula indicada en **(3-6)**. Es importante saber que 'S-LL(HORA)' cambia según el nombre que se le asigne a la hoja del apartado 2, es decir, la hoja con los horarios en minutos.

$$=' S-LL(HORA)'!B2*24 \tag{4–6}$$

4. Una vez están los datos en decimal, habrá que exportarlos a *Matlab*. Para ello, los datos serán guardados como una matriz numérica, pues se trata de horarios de vuelos. Además, se generará un Script para poder importarlos cuantas veces sea necesario en el caso de que se quiera modificar el Excel en algún momento durante el proceso.

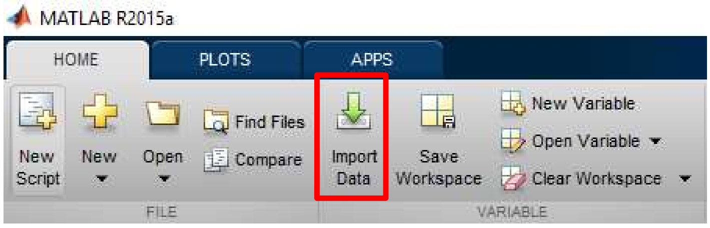


Figura 9: Acción de importar datos en Matlab

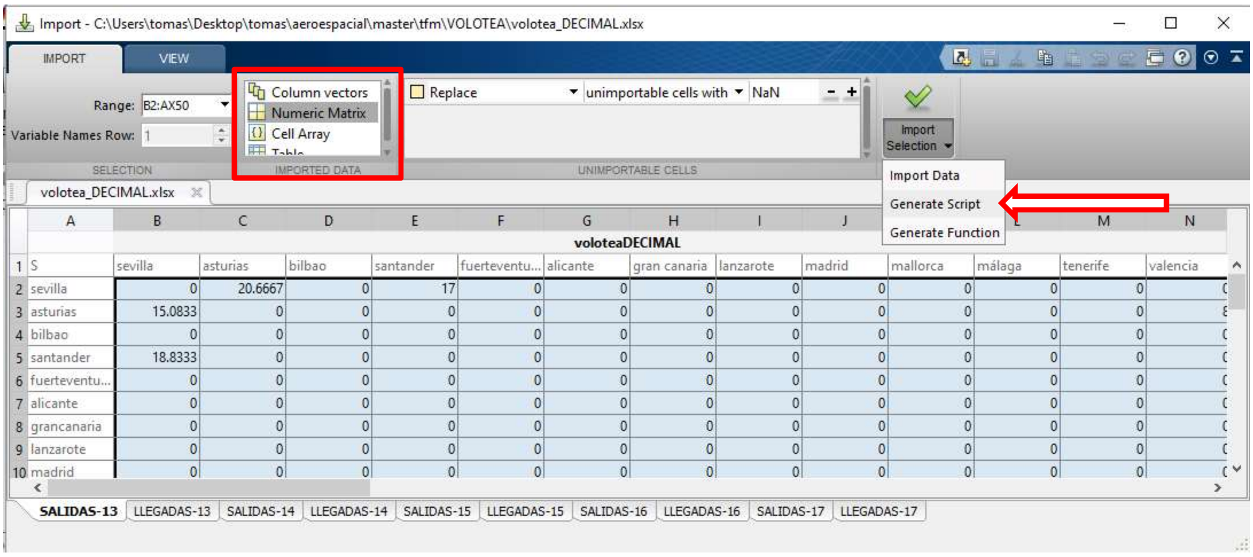


Figura 10: Selección de datos para importarlos a Matlab.

Matlab automáticamente generará las líneas de código correspondientes. En la **Figura 11** se muestra como sería el código para importar los datos correspondientes a los horarios de llegadas para el día 15 de noviembre de 2017 para la compañía Volotea. Se resalta en rojo el lugar donde se indica las filas y columnas afectadas. Se podrá cambiar dichos valores según añadamos más filas y columnas al archivo de datos. Los puntos suspensivos indicarían la ruta del archivo en el ordenador, se ha eliminado para mayor claridad en la imagen.

```
clearvars raw;
%% Import the data
[~, ~, raw] = xlsread('C:\Users\...\volotea_DECIMAL.xlsx','LLEGADAS-15','B2:AX50');

%% Create output variable
LLEGADAS15 = reshape([raw{:}],size(raw));
```

Figura 11: Código ejecutable de datos de importación

Una vez se ha realizado para todas las llegadas y salidas, estas se unirán en una única matriz LL y S con la tercera dimensión correspondiente a las unidades horarias de estudio.

5 CÁLCULO DE ROTACIONES

Una vez se tienen los datos de llegadas y salidas podemos comenzar a operar con ellos. El primer objetivo del proceso es conocer las combinaciones que se pueden realizar entre vuelos sin que halla interferencias en los horarios. Como datos de entrada, sólo se requieren los horarios de salidas y llegadas obtenidos en el apartado anterior, (apartado 2, anexos), así como el tiempo que debe transcurrir desde que una aeronave aterriza hasta que despega. Esta última variable (T), cobra especial importancia pues es un dato que imponer y que dependerá de cada compañía. T puede variar según la empresa y según el tamaño de aeronave, y su valor puede determinar un cambio en las combinaciones posibles, de manera que es un valor para tener en cuenta.

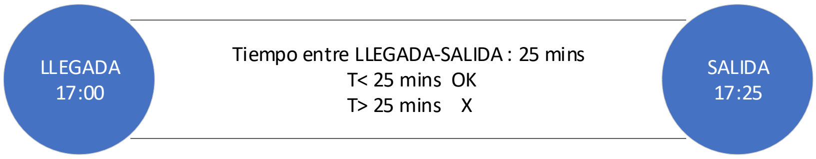


Figura 12: Importancia de T en las rotaciones, ejemplo

5.1. Emparejamiento de vuelos

En primera instancia, sólo visualizando las matrices de llegadas y salidas ya se conocen los pares origen-destino posibles en el problema. Aquellas celdas que posean un valor distinto a cero son una pareja posible. Sin embargo, hay que ir algo más allá. A continuación, lo que se buscarán son tríos de vuelos concatenados de manera que se puedan dividir en dos parejas, una de origen y otra de destino. Imaginemos que la combinación (3-7-15) es posible, donde 3,7,15 son los valores asignados a diversos aeropuertos de nuestro estudio. En tal caso tendremos la pareja de orígenes (3,7) y la pareja de destinos (7,15). Ahora bien, cómo se obtienen todas las parejas posibles mediante un algoritmo matemático.

Sean dos aeropuertos M y N, donde entre M y N existe al menos un vuelo con su horario de salida y de llegada definido por la celda (i, j, k) tanto en S como en LL, donde k sólo refleja la unidad horaria en la que nos encontramos (si se trata de 24 h y k posee un valor de 3 significa que estamos en el tercer día de estudio, recordemos que las matrices LL y S eran tridimensionales). Habrá que buscar todas aquellas salidas que una vez haya aterrizado el vuelo entre M y N, tengan un horario de salida que supere el tiempo T desde la llegada del vuelo. Para poder evaluar todas las salidas posibles desde N se hará uso de la variable **l** que barrerá toda la columna correspondiente al aeropuerto N en cuestión (en el caso de que el aeropuerto tenga asignado más de un valor no ocurre nada pues no debemos olvidar que todas las columnas correspondientes a ese aeropuerto poseen los mismos horarios). Si este estudio no lo realizamos para la combinación M y N, sino para al sistema completo nos queda la expresión indicada en **(4-1)**. Para programarlo no hay más que realizar cuatro bucles concatenados empezando por k, i, j, y finalmente l. Es importante saber que se debe cumplir que LL(i, j, k) $\neq$0, pues si fuera nulo podría ocurrir (de hecho ocurre) que la desigualdad se cumpla, pero es un error pensar que pueda ser una combinación posible pues si el valor de la celda de llegada es nulo significa que no existe tal llegada.

$$LL(i,j,k) + T \leq S(j,l,k), \qquad i,j,l = 1,2,\ldots,B; k = 1,2,\ldots,D \tag{5–1}$$

Si la desigualdad **(4-1)** se cumple, significa que la pareja (i, j) serán dos aeropuertos de origen y (j, l) serán dos aeropuertos de llegada.

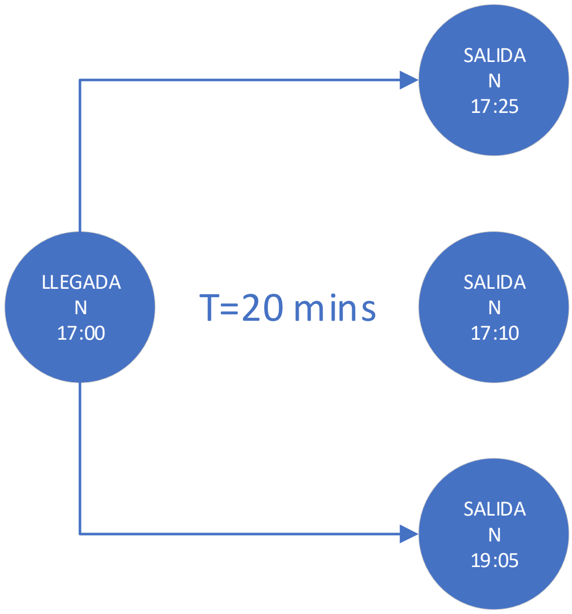


Figura 13: Ejemplo de dos salidas posibles frente a 3 totales

Para poder almacenar toda la información relativa a rotaciones, es decir, las parejas origen destino posibles, habrá que usar variables al efecto, que se llamarán **LLR** y **SR**. Estas variables se inicializarán como conjuntos vacíos e irán aumentando de tamaño conforme se encuentren pares (i, j) y (j, l) que cumplan con los criterios de unión.

$$LLR = \begin{bmatrix} i_1, j_1 \\ i_2, j_2 \\ \dots \\ \dots \end{bmatrix} \tag{5–2}$$

$$SR \quad = \begin{bmatrix} j_1, l_1 \\ j_2, l_2 \\ \dots \\ \dots \end{bmatrix} \tag{5–3}$$

El caso mostrado en la **Figura 14** enseña como existirían para el par de valores i=5 y j=1, cuatro pares de rotaciones salida-llegada, o lo que es lo mismo, cuatro rutas formadas por tres aeropuertos.

$$SR \quad = \begin{bmatrix} 5, 1 \\ 5, 1 \\ 5, 1 \\ 5, 1 \end{bmatrix} \tag{5–4}$$

$$LLR = \begin{bmatrix} 1, 3 \\ 1, 4 \\ 1, 6 \\ 1, 7 \end{bmatrix} \tag{5–5}$$

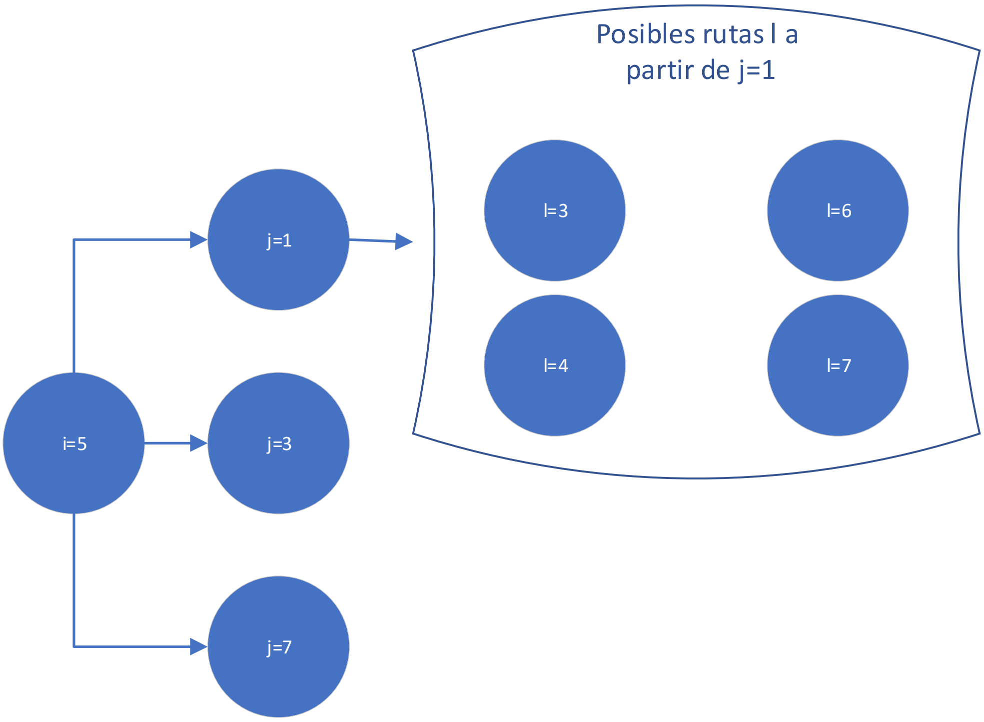


Figura 14: Ejemplo de combinaciones entre aeropuertos

5.2. Agrupación de emparejamientos

Una vez agrupadas todas las opciones posibles habrá que combinarlas entre sí para obtener las secuencias más largas viables. Las iteraciones se realizarán para cada unidad horaria de forma independiente, esto quiere decir que la primera operación será un bucle en k, donde k irá desde 1 hasta D. Para poder llevar a cabo el algoritmo se hará uso de otras variables como LLF y SF. Inicialmente estas variables se les dará el valor del conjunto vacío, y se irán actualizando a medida que avancemos en el bucle.

Se parten de los resultados obtenidos para SR y LLR. A priori sabemos que las columnas tienen un tamaño de 2, recordemos que eran emparejamientos, pero el objetivo es ir agrandando dichos emparejamientos de forma que se obtengan todas las rotaciones al completo. Para ello habrá que iterar, se indica el algoritmo seguido a continuación.

Para poder iterar es necesario conocer el tamaño tanto de LLF y SF en cada iteración, que denominaremos (I,J). J indicará el tamaño de las rotaciones mientras que I muestra el número de rotaciones en el sistema.

Ahora comienza la búsqueda de parejas que encajen entre sí. Para ello hacemos uso de dos índices (m, m_2) que irán aumentando de valor desde 1 hasta I, es decir, hasta el tamaño de la matriz donde se han guardado los emparejamientos posibles. El objetivo es ver que parejas se pueden combinar para formar conjuntos más grandes. La variable m indica el emparejamiento que se está estudiando y la variable m_2 indica la pareja con la que se está comparando. Si $m = m_2$ entonces no se debe realizar comparativa alguna pues se trata de la misma rotación. Para visualizar mejor el proceso supongamos que tenemos dos pares de salidas y llegadas para una unidad horaria k en concreto. Donde β_x indica un valor asignado a un aeropuerto de salida, y γ_x es idéntico, pero para aeropuertos de llegada.

$$SR(m) \ = [\, \beta_{(m1)}, \beta_{(m2)}] \tag{5–6}$$

$$SR(m_2) \ = [\, \beta_{(m_2 1)}, \beta_{(m_2 2)}] \tag{5–7}$$

$$LLR(m) = [\, \gamma_{(m1)}, \gamma_{(m2)}] \tag{5–8}$$

$$LLR(m_2) \quad = [\,\gamma_{(m_2 1)}, \gamma_{(m_2 2)}] \tag{5–9}$$

En este caso se tiene que para que ambas rotaciones se pueden concatenar, se debe cumplir la ecuación (4-10). Es decir, cuando tenemos dos rotaciones formadas por tres aeropuertos, el aeropuerto final debe coincidir con el inicial de la otra rotación de comparación.

$$\beta_{(m2)} = \beta_{(m_2 1)} \tag{5–10}$$

$$\gamma_{(m2)} = \gamma_{(m_2 1)} \tag{5–11}$$

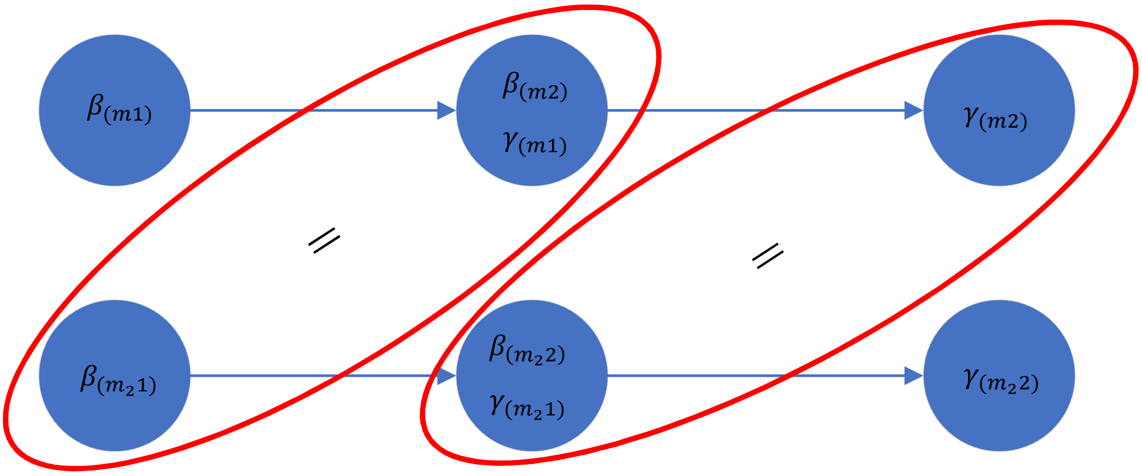


Figura 15: Algoritmo de emparejamientos

De esta forma se obtendría una rotación de 4 aeropuertos, o lo que es lo mismo, SF y LLF serán tríos de aeropuertos.

$$SF(\boldsymbol{end} + 1, :) = [\beta_{(m1)}, \beta_{(m2)}, \beta_{(m_2 2)}] \tag{5–12}$$

$$LLF(\boldsymbol{end} + 1, :) = [\gamma_{(m1)}, \gamma_{(m2)}, \gamma_{(m_2 2)}] \tag{5–13}$$

La expresión '**end** +1' implica que SF y LLF tendrán más rotaciones, y que cuando se encuentra una nueva esta se añadirá a las ya existentes, end indica la última fila. Los dos puntos indica que se trata de una columna entera y no un elemento en concreto.

Además, se debe tener en cuenta que, a pesar de haber encontrado una nueva combinación más larga, no hay que descartar la combinación LLR(m) que estábamos estudiando pues, aunque sea una combinación más corta, para los intereses de una compañía no se puede saber a priori que es más beneficioso. Es decir, habrá que guardar en la variable LLF dicho valor. Al tratarse de una combinación más corta, y almacenarla en una matriz de mayor tamaño, a la combinación se le añadirá un cero al final.

$$SF(\boldsymbol{end} + 1, :) = [\beta_{(m1)}, \beta_{(m2)}, 0] \tag{5–14}$$

$$LLF(\boldsymbol{end} + 1, :) = [\gamma_{(m1)}, \gamma_{(m2)}, 0] \tag{5–15}$$

La rotación m_2 no se incluye porque es la rotación de comparación, no de estudio, m barrerá todas las rotaciones existentes en LLR y SR.

En el caso de que una vez realizado el barrido en m_2 no haya ninguna combinación posible, no se hará acción alguna pues ya se ha almacenado la combinación m según **(4-14, 4-15)**, es decir, esa combinación no puede concatenarse con ninguna otra. Este proceso se realizará para cada combinación, es decir, se habrá realizado un barrido completo a m. Sin embargo, una vez terminado de realizar dicho bucle, hay que saber si todavía pueden realizarse más uniones entre combinaciones, pues solo tenemos hasta 4 aeropuertos, pero puede haber más. Para ello se hará uso de un bucle *'while'*, por medio del cual *'mientras'* haya alguna combinación cuyo último valor sea distinto de cero, 0, significará que puede seguir combinándose. No debe olvidarse que, por medio del algoritmo realizado, siempre que una rotación no tiene combinaciones posibles se le añade un cero al final. Esto quiere decir que al final se debe acabar con un sistema de **r** rotaciones posibles o filas, y **t+1** columnas, siendo t el tamaño de la combinación más larga. Haciendo referencia a la ecuación **(4-12, 4-13)**, se puede ver como por cada vuelta que se de en el bucle 'while' el tamaño de las rotaciones máximo que se encontrará será de 3,4,6,10, ... Esto se debe a que siempre se comparan los dos últimos elementos de LLF y SF para m y los dos primeros para m_2, y son el resto de los elementos los que difieren entre rotaciones concatenadas. Destacar que se usaran variables auxiliares para LLF y SF pues al inicio de cada bucle estas se reinician para almacenar las nuevas rotaciones. Aquellas rotaciones con un cero al final se seguirán rellenando de ceros para evitar problemas en las dimensiones de las matrices.

Finalmente, teniendo en cuenta que se realiza un estudio separado para cada unidad horaria, habrá que cuadrar las matrices obtenidas cada día de manera que se rellenarán de ceros las filas y columnas necesarias para que todos tengan las mismas medidas y así obtener una matriz única y evitar el uso de k matrices diferenciadas. El hecho de que halla filas y columnas nulas de más no es inconveniente, y se usa sencillamente para evitar errores en la lectura del código porque las dimensiones matriciales no sean consistentes. Es decir se obtendrá una matriz final LLF(r, t+1, k) y SF(r, t+1, k) tridimensional.

Tanto LLF como SF se unirán en una única matriz RF donde se almacenarán las rutas al completo y no sólo los nodos de salida y llegada. Esto no es más que sumar a SF el último valor de LLF que no sea nulo, y así se tendrá un conocimiento completo del total de las rotaciones posibles.

5.3. Eliminación de combinaciones repetidas

Debido a la numeración seguida habrá ciertas combinaciones que salgan repetidas. Cuando un determinado aeropuerto tiene más de un valor asignado, esto se debe a que el número de salidas que tiene dicho aeropuerto no se pueden representar por medio de un único valor y por ello se asignaba más de uno al mismo aeropuerto. Sin embargo, como ya se vió, cuando el aeropuerto es el de destino, el tener más de un valor asignado lo único que aporta es la repetición de datos pues los horarios de llegada a un aeropuerto deben ser los mismos para todos sus valores asignados. En el caso de que el aeropuerto en cuestión sea un nodo intermedio, es decir, tenga un nodo anterior y otro posterior no es ningún inconveniente pues ese era el objetivo de la asignación múltiple de un aeropuerto, cada valor está asociado a un vuelo en concreto. Sin embargo, cuando el aeropuerto en cuestión es el último nodo de la rotación, es un nodo de sólo destino (en esa unidad horaria, en la siguiente será de sólo origen). Al ser un nodo de sólo destino, si el aeropuerto en cuestión tiene varios valores asignados, aparecerán combinaciones repetidas. Para ver mejor el por qué se expone un ejemplo.

Supongamos un aeropuerto M que tiene asignados los valores 6,17,18, por ejemplo. Imaginemos una rotación posible del sistema,(4-16).Las tres combinaciones son posibles siguiendo el algoritmo descrito, sin embargo, son totalmente redundantes pues los valores 17 y 18 como último nodo no aportan información que permita generar una rotación nueva. Por ello, deberán ser eliminados, reduciendo considerablemente en algunos casos el número de rotaciones totales. Para evitar posibles errores en el tamaño de las matrices resultantes, las filas de rotaciones que deban ser eliminadas se rellenarán de ceros.

$$[4\ \ 15\ \ 28\ \ 6], [4\ \ 15\ \ 28\ \ 17], [4\ \ 15\ \ 28\ \ 18] \tag{5–16}$$

5.4. Aplicación a casos reales

Los datos de entrada para poder realizar este apartado 4 son el tiempo entre llegadas y salidas para una aeronave (T), así como los horarios de los vuelos.

5.4.1. Canaryfly

La empresa Canaryfly obliga a sus aviones a estar 20 minutos en tierra como mínimo, es decir, T será igual a 0.33, en minutos. Las matrices LL y S se obtuvieron en apartados anteriores. Con ello, y siguiendo el algoritmo desarrollado en este capítulo, se obtuvieron los siguientes resultados.

Existen 18526 rotaciones posibles, teniendo en cuenta las rotaciones cuyo nodo final es repetido debido a la asignación múltiple en aeropuertos(es decir, realmente se tienen unas 3000 combinaciones). El elevado número de rotaciones para solo 38 vuelos diarios se debe a las condiciones de operación de la compañía. Canaryfly opera únicamente en las islas Canarias, cercanas entre sí de manera que los vuelos son relativamente cortos (inferiores a una hora en muchos casos). Además, la flota de aviones es de tamaño medio de manera que el tiempo de espera en tierra por parte de la aeronave es pequeño, lo que posibilita que existan muchas combinaciones posibles entre todos los aeropuertos.

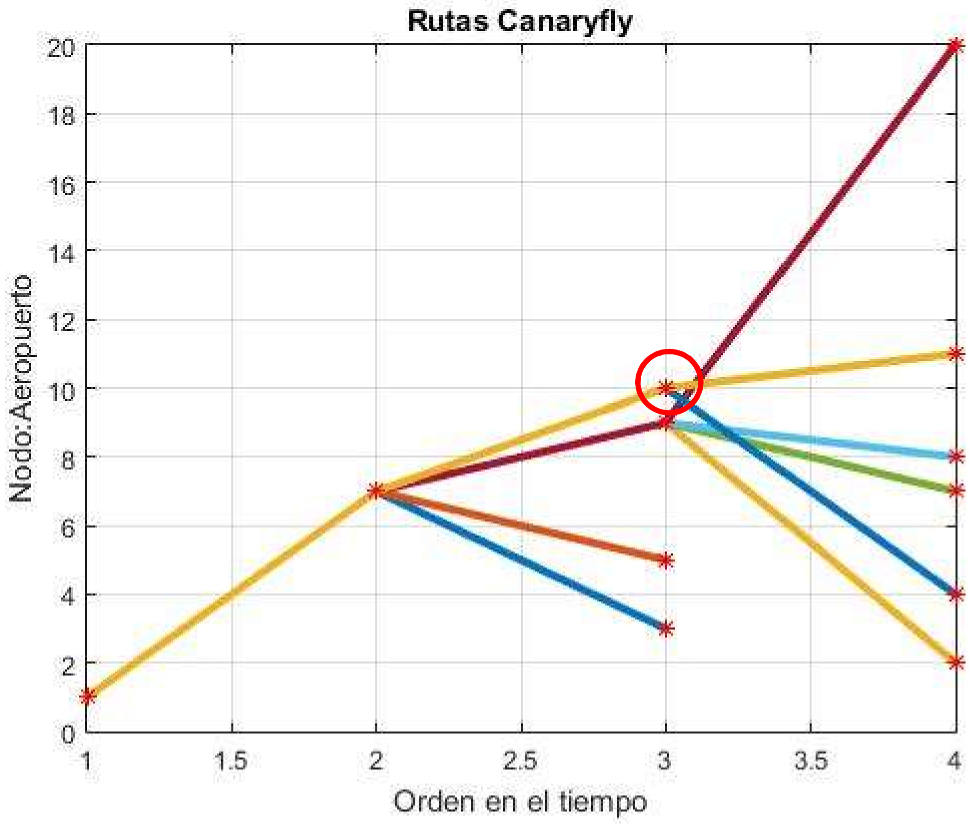


Figura 16: Ejemplo 1 de rutas Canaryfly

El eje de ordenadas indica el orden de los vuelos, pero no la escala temporal en la que se encuentran. Como se puede ver existen rutas que se bifurcan, algunas están marcadas con círculos rojos. Esto se debe a que una ruta (i, j) posee más de un vuelo l con el que concatenarse. El proceso es un diagrama de árbol en el cuál hay rutas más largas que otras (esto quiere decir que hay más vuelos no que en el tiempo sean más largas, no tiene por qué pues un vuelo puede requerir muchas horas), y donde existen rutas que pueden combinarse más fácilmente con otras.

Existirán vuelos que sean limitantes pues no se puedan realizar excesivas combinaciones con los mismos. Esto querrá decir que las rutas que contemplen dichos vuelos deban realizarse con una mayor probabilidad que el resto pues todos los vuelos se deben llevar a cabo.

Como se puede observar, se comprueba como dichas rutas no tienen un origen y destino impuesto (rotaciones de un día, dos días). Esto se debe a que de momento se han calculado todas las rutas posibles. Además, aunque no se aprecie en los diagramas pues están superpuestos, si cogemos la rotación [1 7 10 11], de la **Figura 16**, también estará almacenada la rotación [1 7], como la [1 7 10], es decir, no tenemos por qué escoger la más larga pues los intereses pueden ser varios y almacenar todas las posibilidades permitirá obtener los mejores resultados, aunque el número de combinaciones aumente.

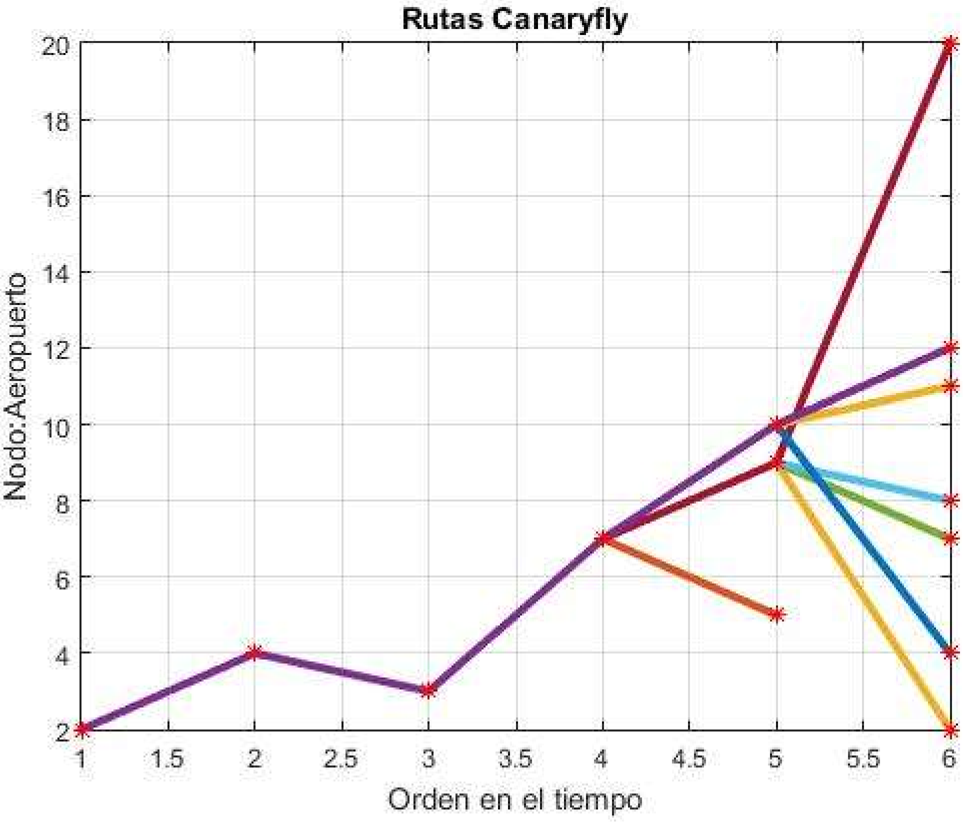


Figura 17: Ejemplo 2 rutas Canaryfly

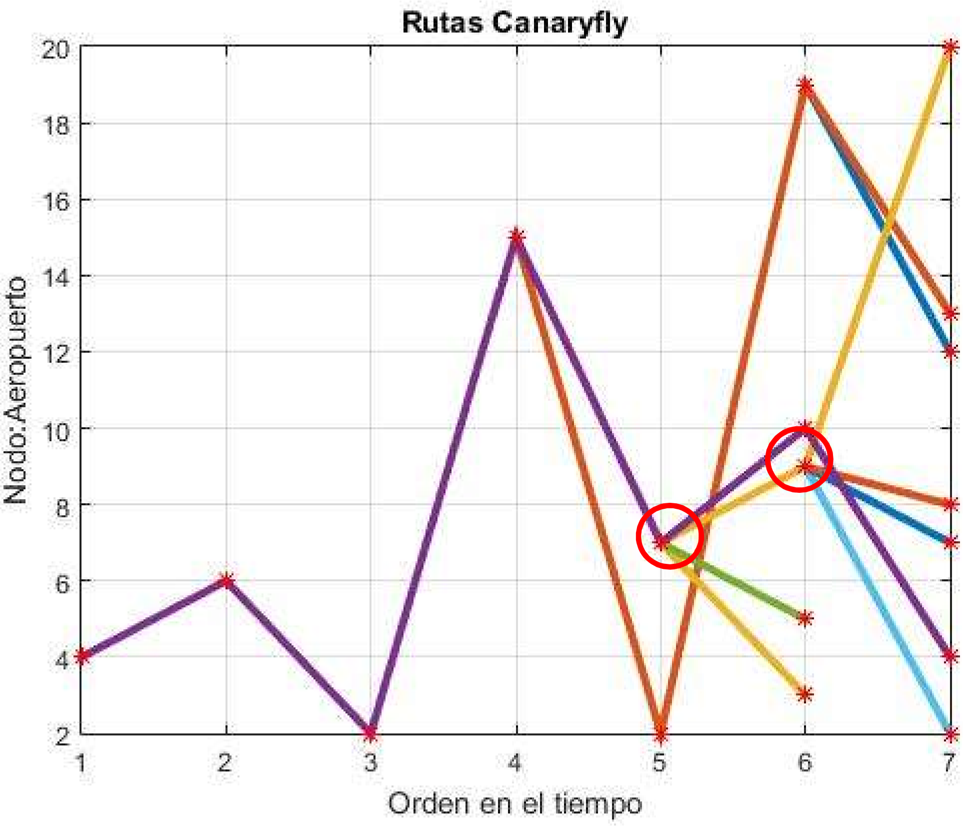


Figura 18: Ejemplo 3 rutas Canaryfly

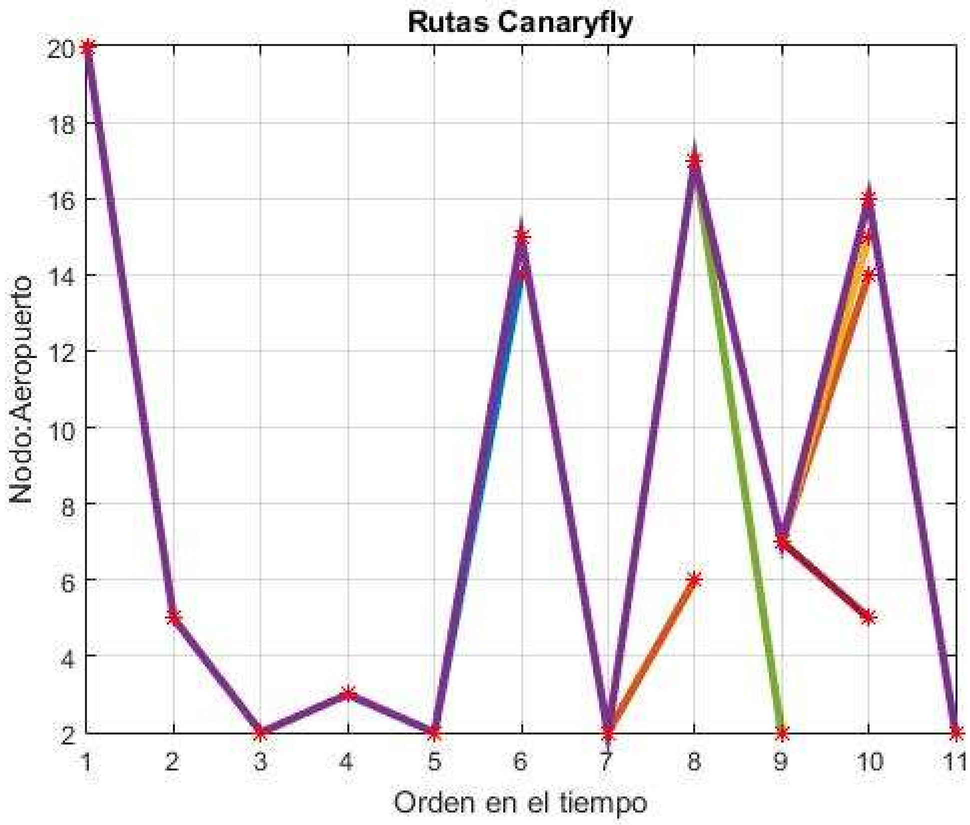


Figura 19: Ejemplo 4 rutas Canaryfly

5.4.2. Volotea

La empresa Volotea obliga a sus aviones a estar 25 minutos en tierra como mínimo (para un tiempo mayor no existe una solución final posible pues se ha supuesto T constante, realmente la mayor parte de sus vuelos tienen una mayor estancia en tierra), es decir, T será igual a 0.42, algo superior que para Canaryfly. Usando la matriz de salidas y llegadas obtenida en apartados anteriores y haciendo uso del algoritmo de este capítulo se obtienen los siguientes resultados. El estudio se realizará para los días 13, 14, 15, 16 de noviembre, pues los horarios según el día varían.

Se han obtenido 1799 rotaciones como máximo en un día, en concreto en el día 13 lunes. Esto se debe a que los vuelos entre semana suelen descender. Volotea es una empresa dedicada al turismo, de manera que la mayoría de los vuelos son en verano, donde hay mayor oferta de destinos y los fines de semana. El día 13 se realizan un total de 70 vuelos, el doble que en Canaryfly, y, sin embargo, se obtienen 3 veces menos combinaciones. Esto se debe a que Volotea opera en toda europa y los vuelos suelen ser más largos en el tiempo. En general, las rotaciones suelen ser cortas, no más de 10 vuelos, y esto es debido a que es una compañía con una gran flota de aeronaves con la que operar todos los vuelos sin necesidad de cargar mucho a una aeronave en concreto.

El día 14 y 15 posee menos de 300 rotaciones posibles, lo que implica que hay un menor número de vuelos a realizar, y que además estos vuelos están limitados en las combinaciones, es decir, habrá rutas determinadas que tendrán una alta probabilidad de ser solución al sistema pues serán las únicas, o casi las únicas, que realicen determinados vuelos del sistema.

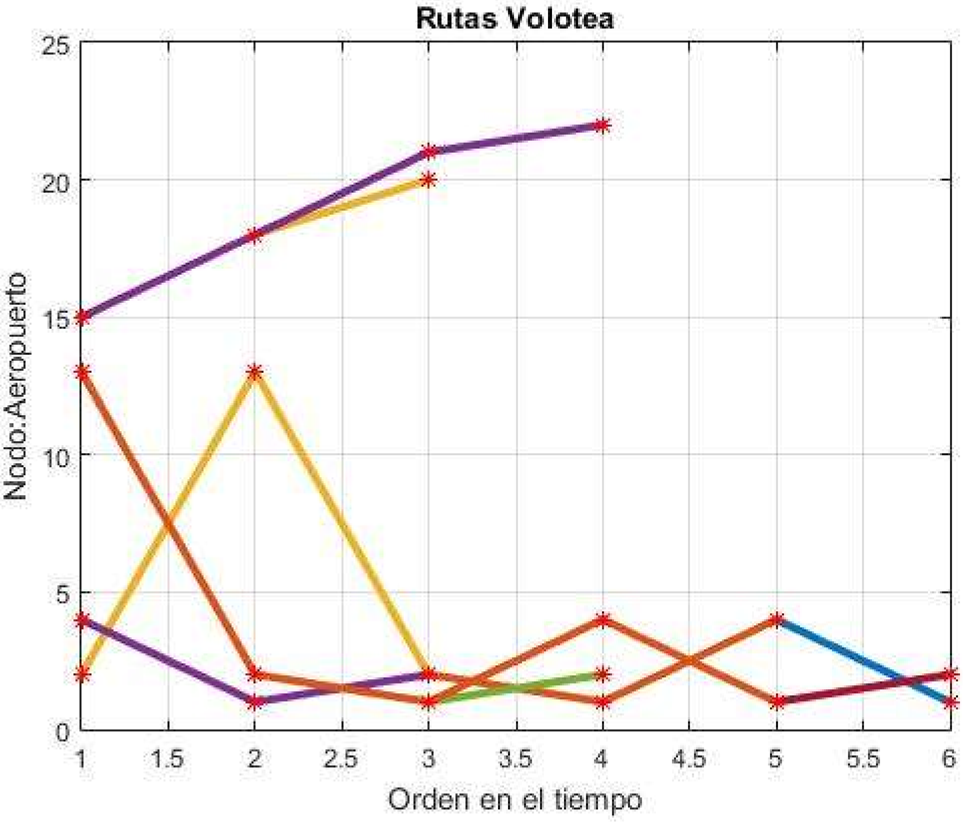


Figura 20: Ejemplo 1 rutas Volotea, día 13

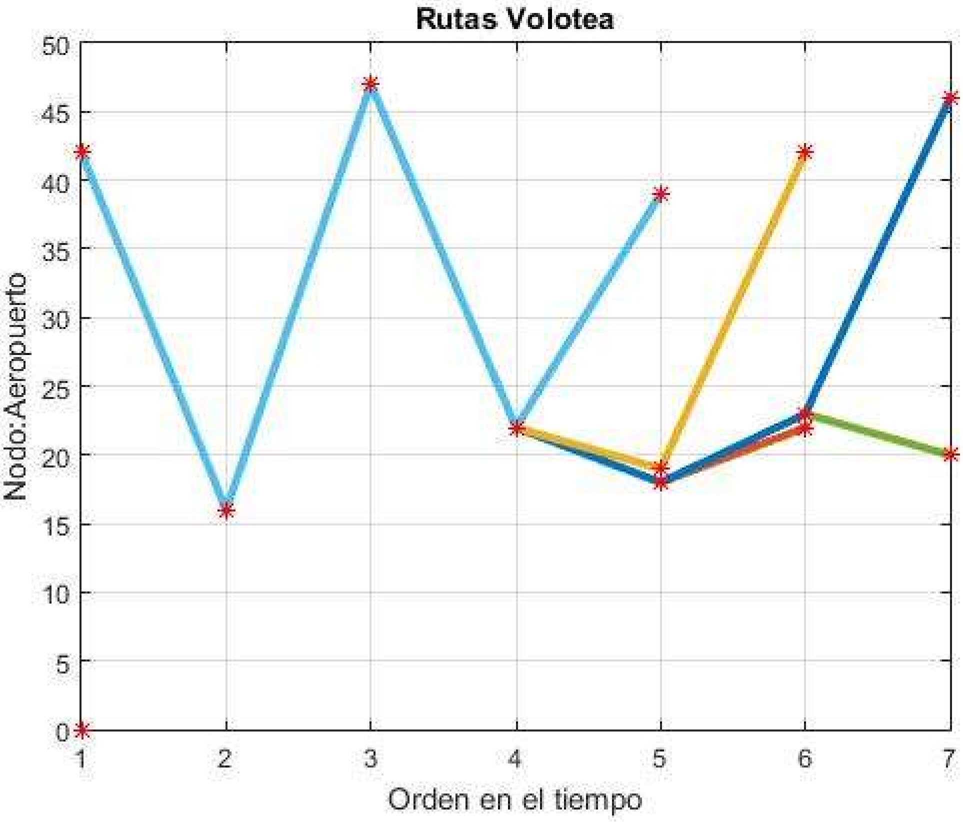


Figura 21: Ejemplo 2 rutas Volotea, día 13

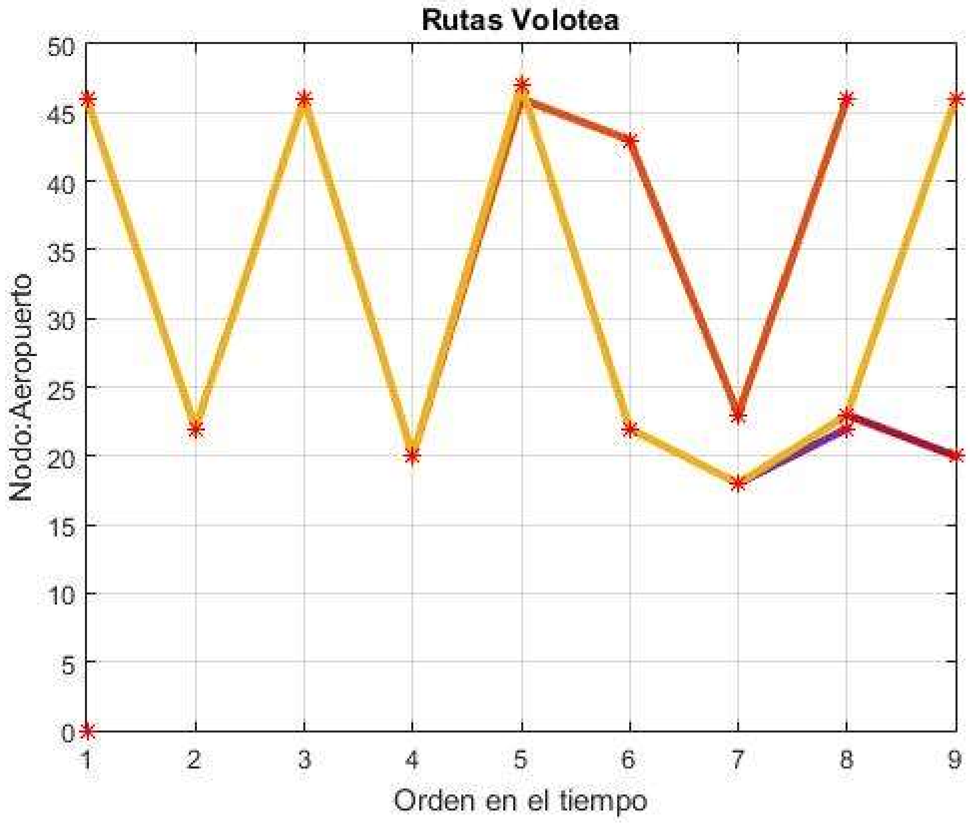


Figura 22: Ejemplo 3 rutas Volotea, día 13

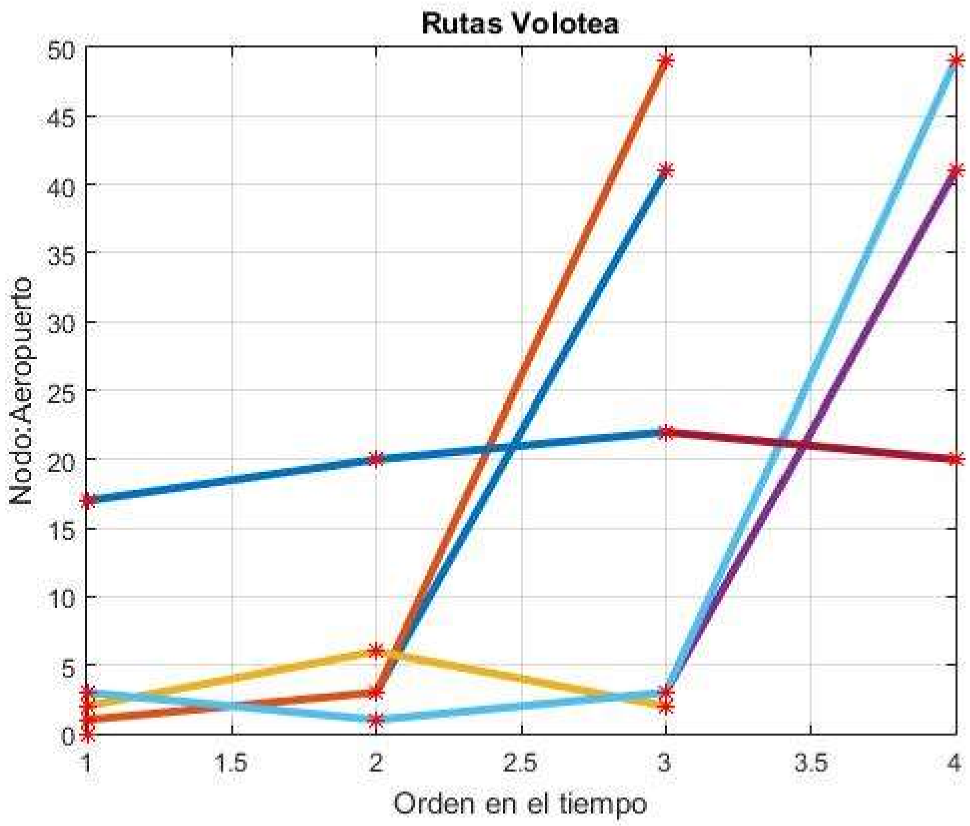


Figura 23: Ejemplo1 rutas Volotea, día 14

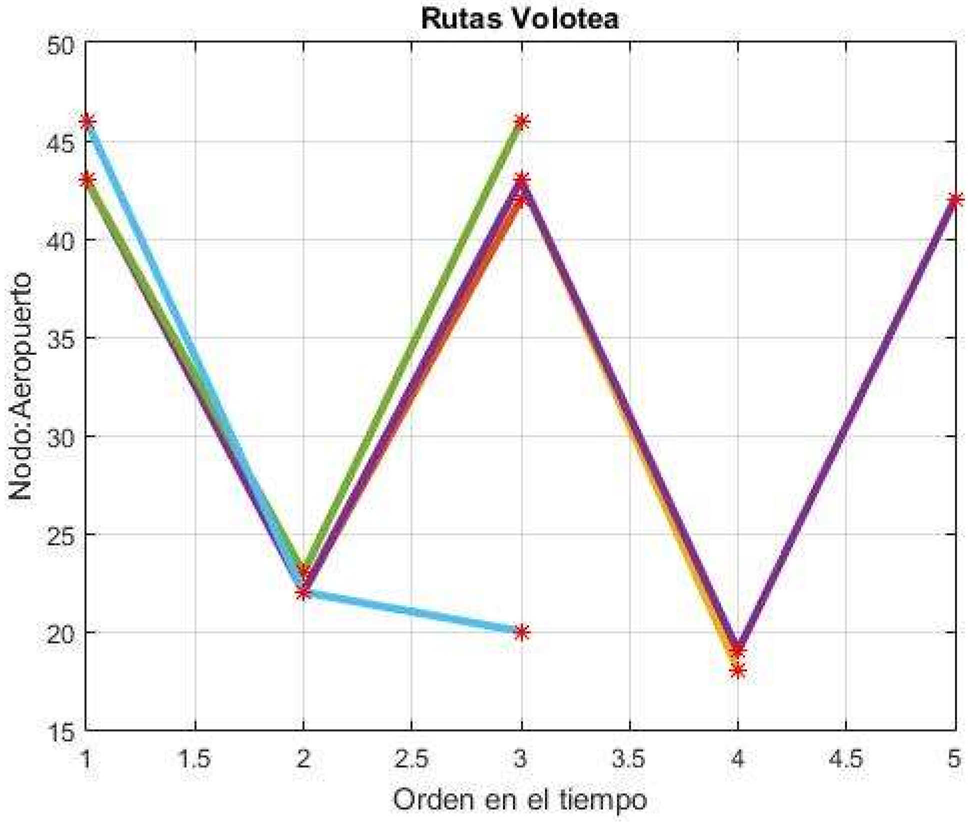


Figura 24: Ejemplo 2 rutas volotea, día 14

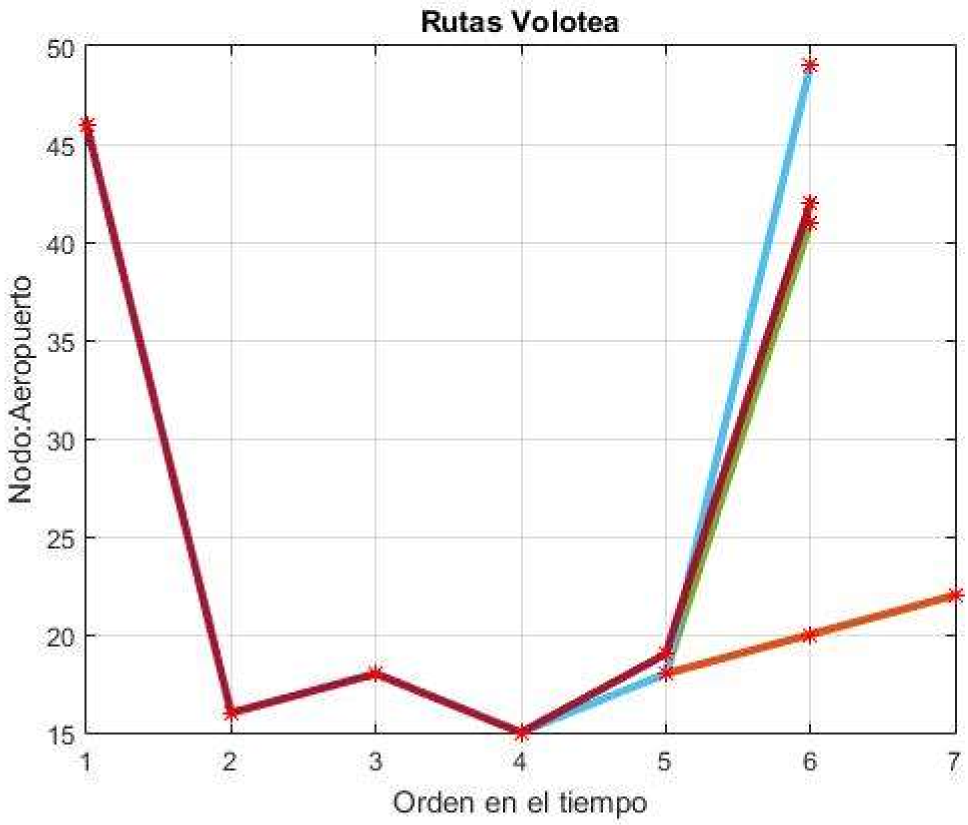


Figura 25: Ejemplo 1 rutas Volotea, día 15

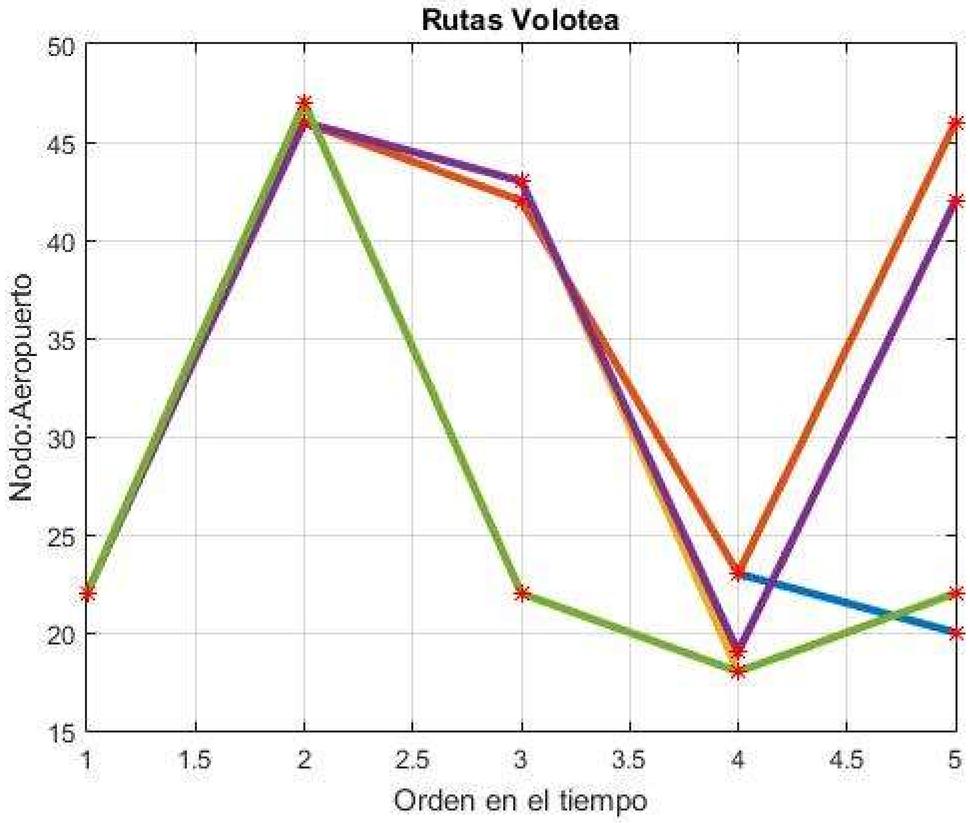


Figura 26: Ejemplo 1 rutas Volotea, día 16

6 DURACIÓN DE ROTACIONES

Una vez se han obtenido todas las rotaciones posibles para cada unidad horaria de estudio, es necesario escoger aquellas combinaciones que permitan resolver el problema según criterios impuestos por la compañía. Toda empresa suele operar de forma que los vuelos se repiten cada cierto tiempo, o lo que es lo mismo, partiendo de un aeropuerto de origen un día por la mañana y acabar tras un periodo de tiempo (un día, dos días, tres días, …) como destino final el mismo aeropuerto al final del ciclo.

6.1. Ciclo real y equivalente

Hay que recordar que el estudio se realiza para cada unidad horaria por separado, sin embargo, una unidad horaria puede representar desde un día hasta varios días, en este caso representerán un día. De esta forma, la duración mínima de las rotaciones será una unidad horaria, según el tramo real que esta represente. Esto quiere decir que sólo se podrán obtener rotaciones proporcionales a la unidad horaria. Es decir, si la unidad horaria es de 48 horas, no se podrán conseguir rotaciones de 3 días, pues representan 72 horas que es 3/2 de la unidad horaria establecida.

6.2. Cambio de asignación

Cuando un aeropuerto tiene más de un vuelo diario hacia un mismo destino, se le asigna más de un valor para poder definir todos los vuelos de manera inequívoca. Sin embargo, el hecho de que dos rotaciones tengan como valor inicial y final números distintos no quiere decir que se traten de aeropuertos distintos. Supongamos que el aeropuerto M tiene asignados los valores 6,17,18. En ese caso, todas las rotaciones que empiecen por 6,17,18, independientemente de los vuelos que contemplen, tienen el mismo origen. Para poder operar de forma congruente y que no haya problemas habrá que transformar dicha numeración.

Como se ha indicado, los únicos nodos que interesan en este momento son el primero y el último en cada rotación. Sin embargo, el último nodo ya fue estudiado en apartados anteriores para evitar la repetición de rotaciones y no deberá ser reasignado. Todos los primeros nodos de todas las rotaciones deben ser estudiados para que, en el caso de un aeropuerto con más de una asignación posible, estos valores se transformen todos en un mismo valor, en el ejemplo anterior sería 6. Pero claro está, se deberán guardar los valores originales pues estos dan información acerca del nodo (recordemos que el valor 17 o 18 nos indicaban que se trataba de la fila 17 o 18 de la matriz de salidas y llegadas). A continuación, se muestra matemáticamente como se resuelve este paso.

En primera instancia se introducen los valores de aquellos aeropuertos que tienen más de una asignación. Para ello introducimos una variable llamada **a** y otra **b**. Estos vectores deben tener el mismo tamaño y lo que indican es que para cada elemento en **b** correspondiente a un aeropuerto, hay otro elemento en **a** en la misma posición que indica el valor de asignación principal de dicho aeropuerto. Para Canaryfly **(5-3, 5-4)** y Volotea **(5-1, 5-2)** se tendrán los siguientes vectores. Los valores de Canaryfly son más repetitivos debido a la alta afluencia de vuelos entre los mismos aeropuertos que lleva a asignar múltiples valores a un solo aeropuerto.

$$a_{Volotea} = [\,18, 20, 22, 42, 46] \tag{6–1}$$

$$b_{Volotea} = [\,19, 21, 23, 43, 47] \tag{6–2}$$

$$a_{Canaryfly} = [2, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 6,6,6] \tag{6–3}$$

$$b_{Canaryfly} = [7, 8, 20, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] \tag{6–4}$$

A continuación, se realiza un barrido para cada unidad horaria desde 1 hasta I, siendo I el número total de rotaciones. Si el elemento número uno de la rotación coincide con cualquiera de los valores en b, cuyos componenetes se leen con el subíndice j, deberá ser sustituido por el valor correspondiente de a.

$$\text{Si} \quad RF(i,1,k) = b(j) \;\; i = 1, \dots, I; k = 1, \dots, D \tag{6–5}$$

$$\text{Entonces} \quad RF(i,1,k) = a(j) \tag{6–6}$$

Para no perder la información de los valores de b habrá que guardar que rotaciones han sufrido cambios para revertir los valores al final del proceso. Para ello se hace uso de una nueva variable llamada **LC** y **CC**, donde se guardarán las posiciones donde se han realizado los cambios y el número eliminado y añadido, respectivamente. Estas variables se irán actualizando de manera que todas las rotaciones que sufran cambios queden guardadas para posteriormente devolverlas a su estado inicial.

$$LC(end + 1) = [i, k] \tag{6–7}$$

$$CC(end + 1) = [b(j), a(j)] \tag{6–8}$$

Una vez realizados los cambios se puede proceder a realizar el estudio.

6.3. Rotaciones de 1 día

Hay compañías, normalmente de tamaño reducido, que pueden y realizan los vuelos de manera que el aeropuerto de origen al inicio de un día es el mismo que el aeropuerto de destino al final de dicho día. En sí las rotaciones pueden ser distintas cada día, pero el lugar de origen y destino es el mismo al final del día. En este caso bastará con realizar un estudio por separado para cada unidad horaria, y escoger sólo las rotaciones que cumplen con el criterio indicado. De esta forma las posibilidades reales son inferiores al montante total de rotaciones posibles. Si se realiza dicho proceso de forma matemática, para ello habrá que evaluar las rotaciones.

Habrá que tener en cuenta que, para evitar problemas de evaluación en *Matlab*, si una rotación tiene un tamaño inferior a la rotación mayor del sistema, entonces se habrá rellenado de ceros. Es decir, el último nodo de la rotación no tiene por qué coincidir con el último elemento de la fila. Puede ocurrir que la rotación sea la indicada en la ecuación **(5-10)**, donde los ceros no son más que para evitar problemas de dimensiones y no aportan información. Cabe destacar que el símbolo ':' indica que se está evaluando la fila al completo. En este caso lo que interesa es el primer y el quinto nodo de la rotación pues como ya se ha dicho y se reitera, los ceros no aportan información. Para ello habrá que evaluar cada fila y escoger el valor **F** tal que el nodo correspondiente a dicha posición sea distinto a cero, pero el nodo correspondiente a F+1 sea nulo, ecuación **(5-9)**.

$$F / \quad RF(i, F, k) \neq 0 \quad \& \quad RF(i, F+1, k) = 0\, ; i = 1, \dots, I; k = 1, \dots, D \tag{6–9}$$

$$RF(i, :, k) = [2, 4, 3, 11, 7, 9, 2, 0, 0, 0] \rightarrow F = 7 \tag{6–10}$$

Una vez se ha calculado F se comprueba que se cumplen los requisitos para rotaciones de un día.

$$RF(i,1,k) = RF(i,F,k); i = 1, \dots, I; k = 1, \dots, D \tag{6–11}$$

En caso afirmativo, se guardará la rotación y se seguirá en el bucle i hasta obtener todas las rotaciones que cumplen con el criterio. Una vez halla concluido el bucle, habrá que devolver los valores de inicio a sus asignaciones reales, es decir, habrá que hacer uso de las variables LC y CC.

6.4. Casos prácticos, rotaciones de 1 día

6.4.1. Canaryfly

Canaryfly cumple la condición de ser una empresa de tamaño pequeño y además todos los días tiene los mismos horarios, de forma que la solución más probable es el uso de rotaciones de un día. Usando como dato de partida los resultados obtenidos en **RF** escogemos aquellos cuyo aeropuerto de origen y de destino sea el mismo, siendo el resultado de 4273 rotaciones. Puede parecer un resultado elevado, pero las condiciones geográficas de operación de Canaryfly llevan a este fenómeno. Canaryfly opera todos sus vuelos, llegando hasta los 38, en horarios que suelen bajar de 1 hora de duración. Además, sólo opera en las Canarias, islas muy próximas entre sí lo que permite poder realizar diversas combinaciones.

Para determinar cuales son las rutas más limitantes y las que poseen numerosas combinaciones, se representa un diagrama con las veces que aparece cada vuelo en las rotaciones. El mínimo numero de veces que aparece una rotación es 84 veces, y se trata de la combinación de 12 a 2, de Gran Canaria a Tenerife Norte. 84 veces es un número muy elevado, y evidencia el hecho de que Canaryfly tiene unas condiciones que, aunque sea una compañía pequeña, el número de rotaciones es muy grande.

En la **Figura 27** se evidencia como existen vuelos que se llevan a cabo un menor número de veces. Aquellas que aparecen nulas no son vuelos, se corresponden con los valores nulos de la matriz, es decir, que no existen vuelos entre dichos nodos. Lo que representa esta gráfica es el número de veces que se repite un determinado vuelo entre todas las rotaciones posibles. El eje 'asignación de nodos' hace referencia a los nodos de salida de un determinado vuelo (filas de la matriz S, LL), mientras que el eje 'Aeropuertos' hace referencia a los nodos de llegada (columnas de la matriz S, LL). Como se puede ver, el eje 'Aeropuertos' es más corto y porque se han eliminado aquellos nodos de llegada redundantes. Recordemos que cuando a un aeropuerto se le asigna más de un número, este sólo aporta información en la salida, pero no en la llegada. De esta forma en la posición (15,4) hay un valor de 123, lo que quiere decir que el vuelo de 15 hacia 4 se repite 123 veces en todas las rotaciones estudiadas.

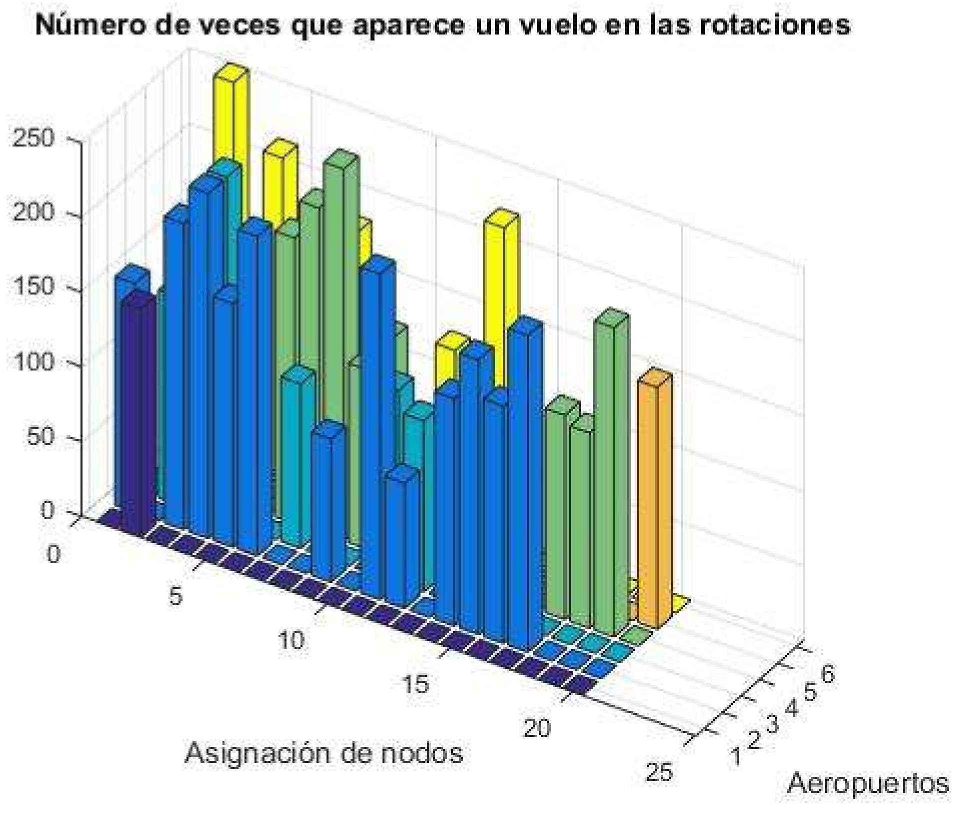


Figura 27: Aparición de vuelos en todas las rotaciones, Canaryfly

6.4.2. Volotea

Aunque existen soluciones al caso de rotaciones de 1 día, y en conjunto no pueden resolver el problema al completo será de interés realizar el estudio en conjunto con rotaciones de dos días. Esto se debe a que Volotea es una empresa de considerable tamaño donde los vuelos no son diarios, sino que varían según convenga y es posible dar una solución para rotaciones diarias y no diarias en conjunto.

Por ello, los resultados obtenidos se presentan a continuación. Se realiza el estudio para cuatro días de la semana (de lunes a jueves), por lo cual se obtendrá una matriz tridimensional (4 días) donde el máximo número de combinaciones es de 173, y se dan el lunes, donde el número de vuelos es considerablemente superior al del resto de días laborales (esxceptuando el viernes). Al igual que se hizo para Canaryfly, observaremos cuales son las rutas limitantes, partiendo de la base de que existirán rutas que no se alcanzarán a realizar por medio de estas rotaciones.

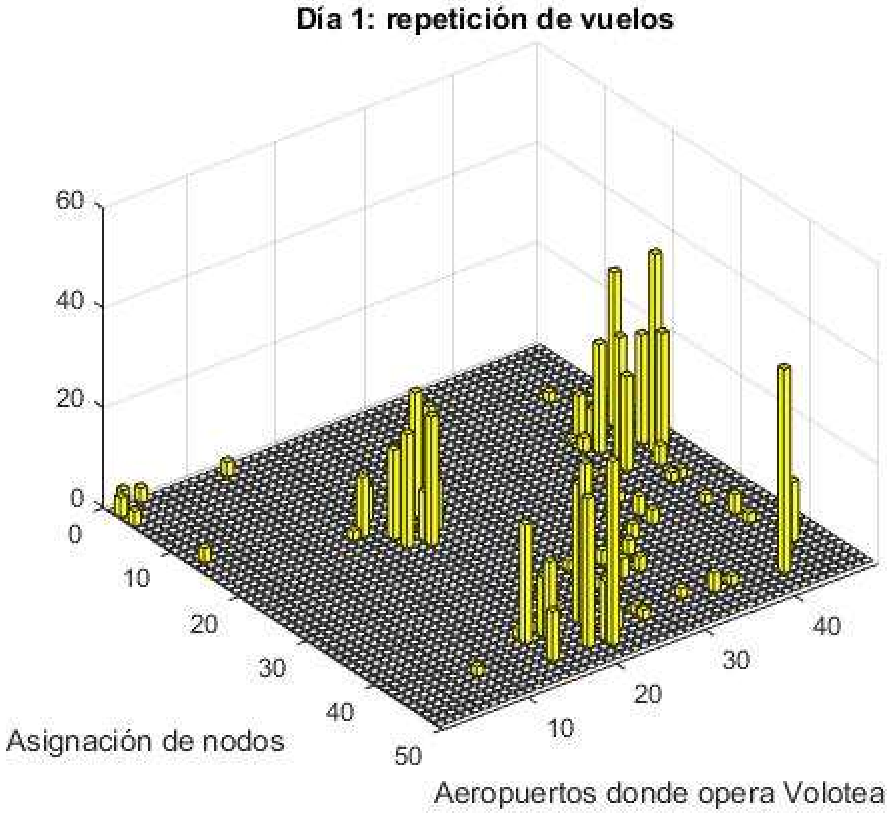


Figura 28: Aparición de vuelos en rotaciones, día 1 (13 noviembre), Volotea

Para el caso de Volotea, ocurre exáctamente igual. La diferencia estriba en que la asignación de valores para los aeropuertos prácticamente es una relación uno a uno, de manera que ambos ejes son similares. Al final, lo que quieren representar estas gráficas son el número de veces que se repite un determinado vuelo entre todas las rotaciones encontradas. Aquellos puntos de la matriz donde no hay nada, no tiene que implicar que no se realice el vuelo. Se puede tratar de combinaciones inexistentes, ceros en las matrices de salida y llegada.

Estas gráficas son útilies pues permiten ver que vuelos serán más limitantes pues existen menos rotaciones que las realizan, o dicho de otra manera, que vuelos fijan más el sistema y obligan a realizar determinadas combinaciones. Por otro lado, aquellas columnas de maor tamaño, son los vuelos que se realizan un mayor número de veces, es decir, exiten multitud de rotaciones que tienen en cuenta dicho vuelo.

Al final la gráfica lo que quiere mostrar es que aquellas columnas de mayor tamaño limitan menos el problema pero lo dejan más abierto, con mayor número de posibilidades mientras que las columnas más chicas son aquellas que restringen el problema y hacen que sea más sencillo buscar la solución correcta.

Para el caso de Canaryfly, como se pudo ver, todos los vuelos se realizaban en multitud de rotaciones, lo que quiere decir que ninguna rotación en concreto fijaba el sistema y quedaba todo abierto, con la posibilidad de obtener optimizaciones en el sistema mayores que para Volotea, que tiene fijado gran parte de sus vuelos a muy pocas rotaciones posibles. Esto se debe, a que Volotea opera en un espacio mayor, lo que impide poder combinar con todos y cada uno de sus aeropuertos.

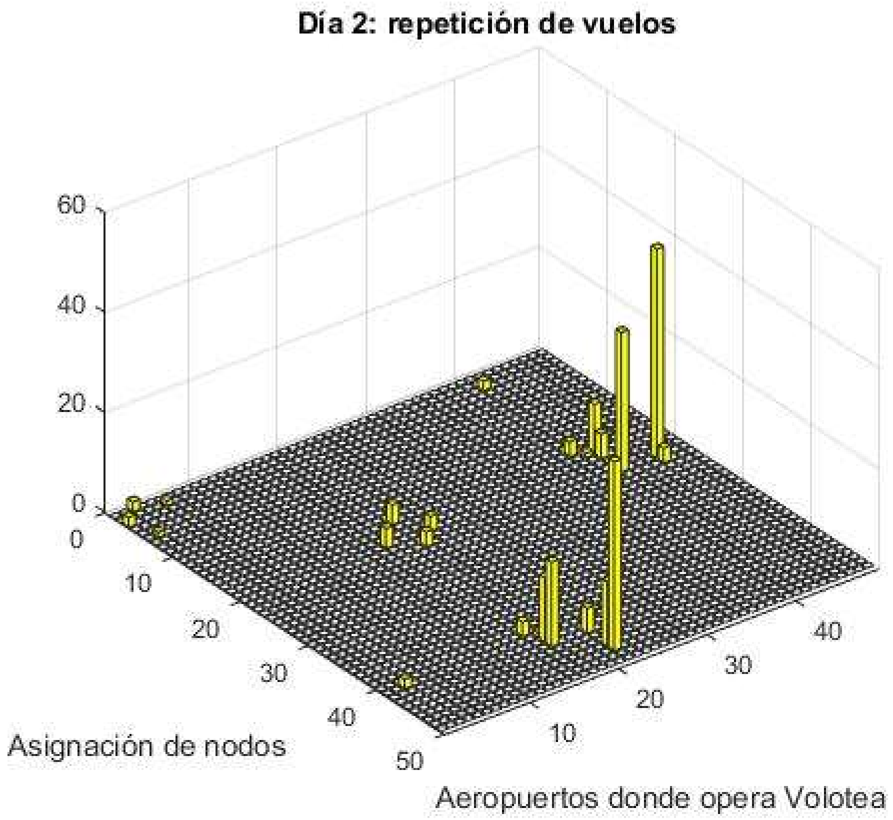


Figura 29: Aparición de vuelos en rotaciones día 2 (14 noviembre), Volotea

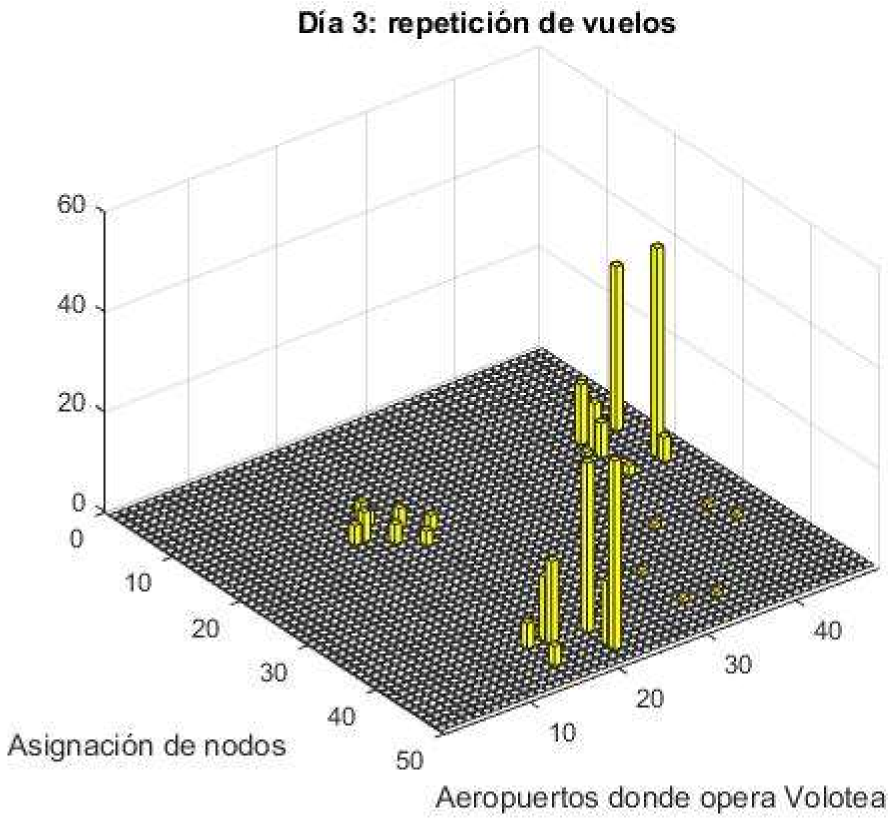


Figura 30: Aparición de vuelos en rotaciones, día 3 (15 noviembre), Volotea

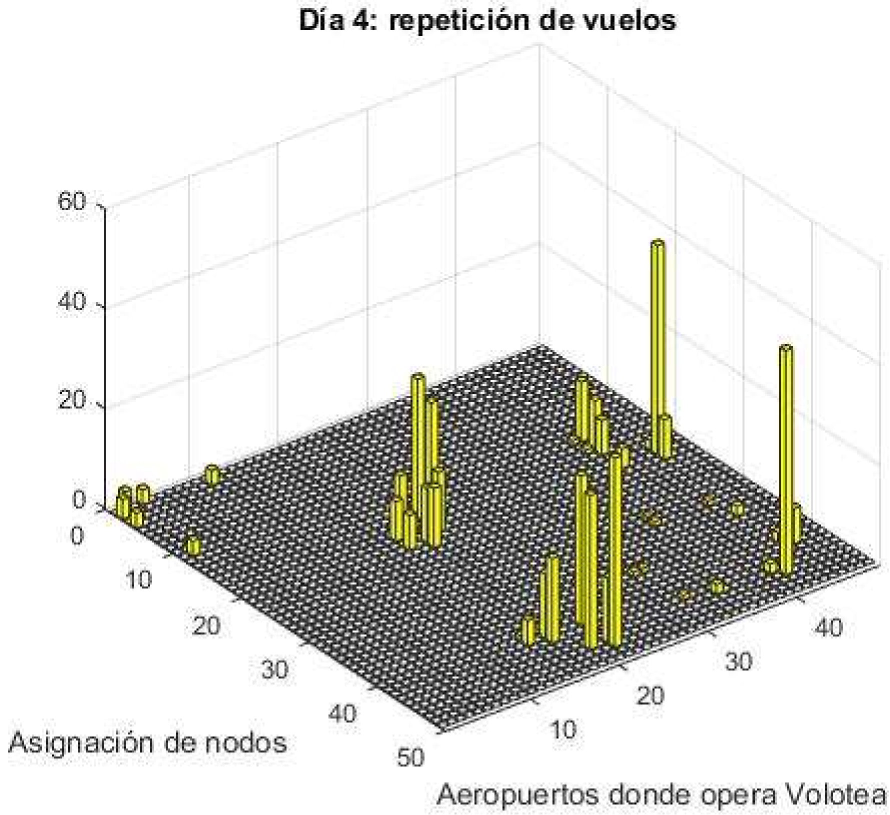


Figura 31: Aparición de vuelos en rotaciones día 4 (16 noviembre), Volotea

Todos los vuelos se pueden llevar a cabo por medio de rotaciones diarias. Sin embargo, es interesante comparar que solución sería óptima. Si usar rotaciones de 1 día o 2 días, o una combinación de ambas. Este estudio se puede realizar y aportará información extra que permita alcanzar la solución que mejor se adapte al sistema. Como se verá más adelante, los días 13 y 14 las rotaciones de 1 día serán las óptimas mientras que los días 15 y 16 una combinación de todas será la mejor opción. Bien es cierto que se podría resolver con rotaciones de un día solamente, pero no aportaría información extra del estudio hecho con Canaryfly, y de esta forma se alcanzan nuevas soluciones y conclusiones.

6.5. Rotaciones de 2 días

En este caso habrá que estudiar los días por parejas. De esta manera habrá que concatenar las rotaciones de dos días de forma que el aeropuerto de origen al inicio del día 1 sea el mismo que el aeropuerto de destino al final del día 2. La aeronave deberá pasar noche en algún aeropuerto del sistema, y en principio no se imponen restricciones al respecto. A continuación, se pretende explicar matemáticamente el algoritmo a partir del cual se lleva a cabo dicho procedimiento.

Se supone que tenemos I rotaciones para el día 1 de estudio (día k) y J para el día 2 (día k+1), y que I≠J, ya que todos los días tengan las mismas rotaciones es poco probable. Habrá que ver si se pueden combinar entre sí dichas combinaciones. Para ello se realizará un barrido en i desde la rotación primera hasta la I del día 1. Cada i corresponde a una rotación del día 1. Para poder comparar con las rotaciones del día 2 se hará un segundo barrido en j desde 1 hasta J. Para que ambas rotaciones se puedan combinar se debe cumplir lo indicado en la ecuación **(5-12, 5-13, 5-14)**. Donde F indica el último valor de la rotación de estudio, es decir, si una rotación está compuesta por cuatro nodos F tendrá un valor de 4. Es importante entender esto pues hay que recordar que **RF** tiene las dimensiones de la rotación más grande para evitar errores dimensionales y por ello F no tiene por qué ser el número de columnas total de esta matriz pues puede resultar que en alguna fila dicho valor sea un cero, o lo que es lo mismo, no sea un nodo y no estemos realizando el estudio correctamente. Los ceros se escribían para cuadrar las matrices, pero no aportaban información adicional. La tercera ecuación **(5-14)** indica que, si el aeropuerto de origen y de destino en el día uno son el mismo, no valdrá como opción pues eso sería dar uso de las rotaciones de un día, obtenidas en la sección anterior. F posee sobíndice pues dependerá de la rotación de estudio y del día.

$$RF(i,1,k) = RF(j, F_{j,k+1}, k+1); i = 1, \dots, I; j = 1, \dots, J; k = 1, \dots, D \tag{6–12}$$

$$RF(i, F_{i,k}, k) = RF(j, 1, k+1) \tag{6–13}$$

$$RF(i,1,k) \neq RF(i, F_{i,k}, k) \tag{6–14}$$

En el caso afirmativo de que se cumplan dichos requisitos, habrá que guardar dichas rotaciones. Al tratarse de rotaciones de dos días, habrá que agrupar las rotaciones de forma coherente. Para ello, se hará uso de una nueva variable **RT**, donde se guardarán las rotaciones que cumplen el criterio de búsqueda. La rotación i correspondiente al día k y la rotación j correspondiente al día k+1 están concatenadas, de forma que generan una única rotación de dos días. Para indicar que forman una rotación deberán ser guardadas en la misma fila indicando que se trata de dos días distintos, para ello se añadirá un valor nulo entre ambas cadenas.

$$RT(end+1) = [RF(i, 1{:}F_{i,k}, k), 0, RF(j, 1{:}F_{j,k+1}, k+1)\,] \tag{6–15}$$

Al igual que en RF después será necesario añadir ceros al final de cada fila para que la matriz RT no tenga problemas de dimensiones.

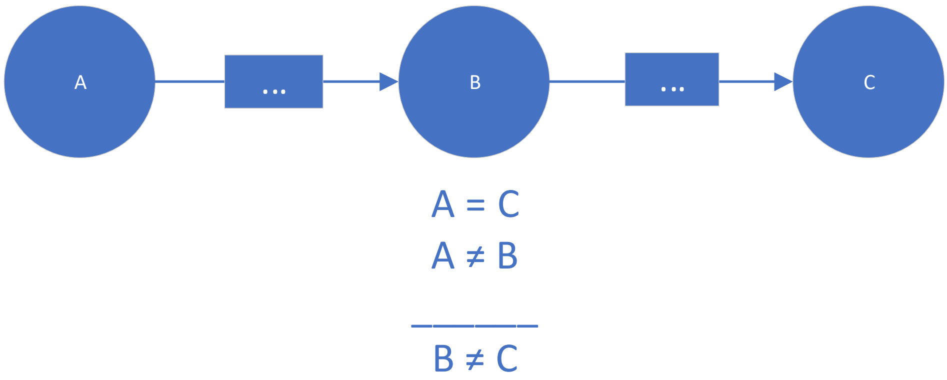


Figura 32: Rotaciones de 2 días.

6.6. Caso práctico, rotaciones de 2 días

6.6.1. Canaryfly

Se llevó a cabo el estudio para Canaryfly y se obtuvieron una cantidad de combinaciones posibles excesivamente elevada. Esto se debe a las condiciones expuestas anteriormente, y es que, al operar en un espacio tan reducido, esto permite aumentar las combinaciones entre vuelos. Al ser todos los horarios idénticos diariamente, nos lleva a concluir que Canaryfly opera con rotaciones de 1 día, realizar el estudio para 2 días llevaría un tiempo elevado y no aportaría información relevante, pues se existe la posibilidad de reducir el periodo de las rotaciones para que la tripulación duerma siempre en el mismo sitio.

6.6.2. Volotea

Para el caso de Volotea se llegó a la conclusión de que para obtener soluciones era necesario combinar rotaciones de 1 y 2 días, pues con sólo rotaciones de 2 días no se llegaba a una solución que englobara todos los vuelos. En apartados anteriores ya se indicó los resultados obtenidos para rotaciones de 1 día, se presentan los resultados para rotaciones de 2 días.

Si se suman las soluciones obtenidas para 1 día y 2 días el problema queda completamente definido, si bien algunos vuelos sólo son realizados por una rotación en concreto, lo que llevará a que dichas rotaciones deben formar parte de la solución. Cabe destacar que, con las rotaciones de dos días, se alcanza a realizar un menor número de vuelos el lunes 13 que, con rotaciones de 1 día, esto se debe a la escasa cantidad de vuelos que se llevan a cabo el martes 14, sin ningún parecido con los realizados el lunes. Esto imposibilita llegar a una solución viable con rotaciones de dos días, pero en conjunto con rotaciones de un día puede ocurrir que las soluciones sean óptimas, en lugar de sólo usar un tipo de rotaciones. El tipo de rotación puede depender de la aeronave en concreto y no tiene por qué ser todo el sistema con el mismo tipo de rotaciones, sino que se da una mayor flexibilidad para adaptar los vuelos a la realidad y así ajustar los resultados lo más posible.

Destacar que el día 15 de noviembre sí se realizan todos los vuelos con las rotaciones de dos días, pero como el día 16 no se llegan a realizar todos los vuelos, también harán falta rotaciones de un día para completar el sistema.

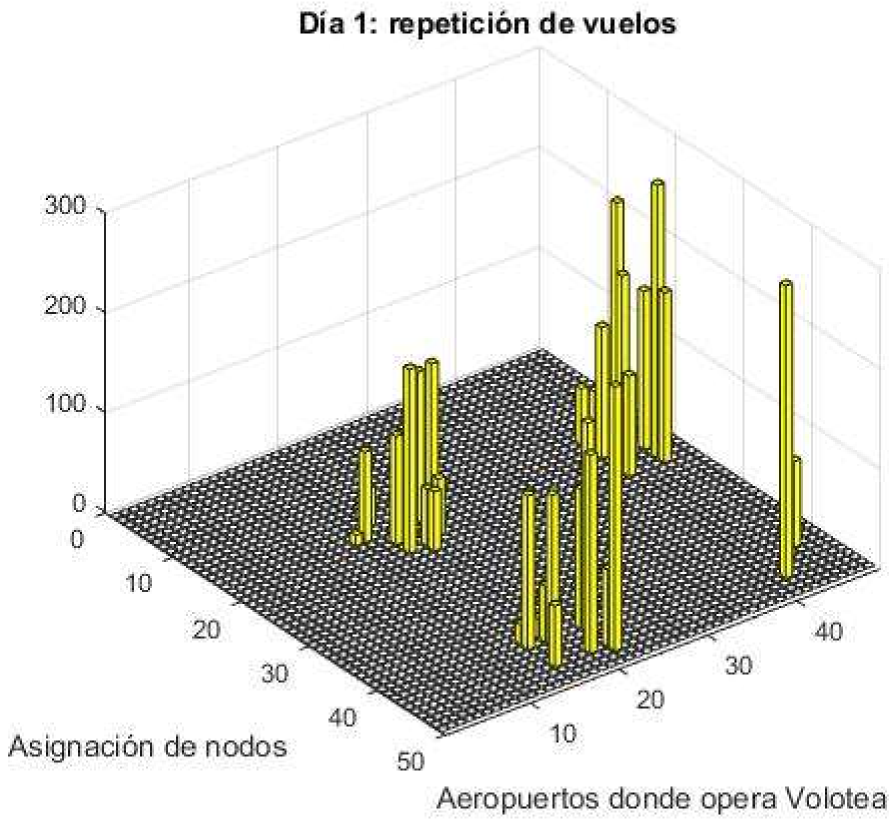


Figura 33: Aparición de vuelos en rotaciones de 2 días, día 1(13 noviembre), Volotea

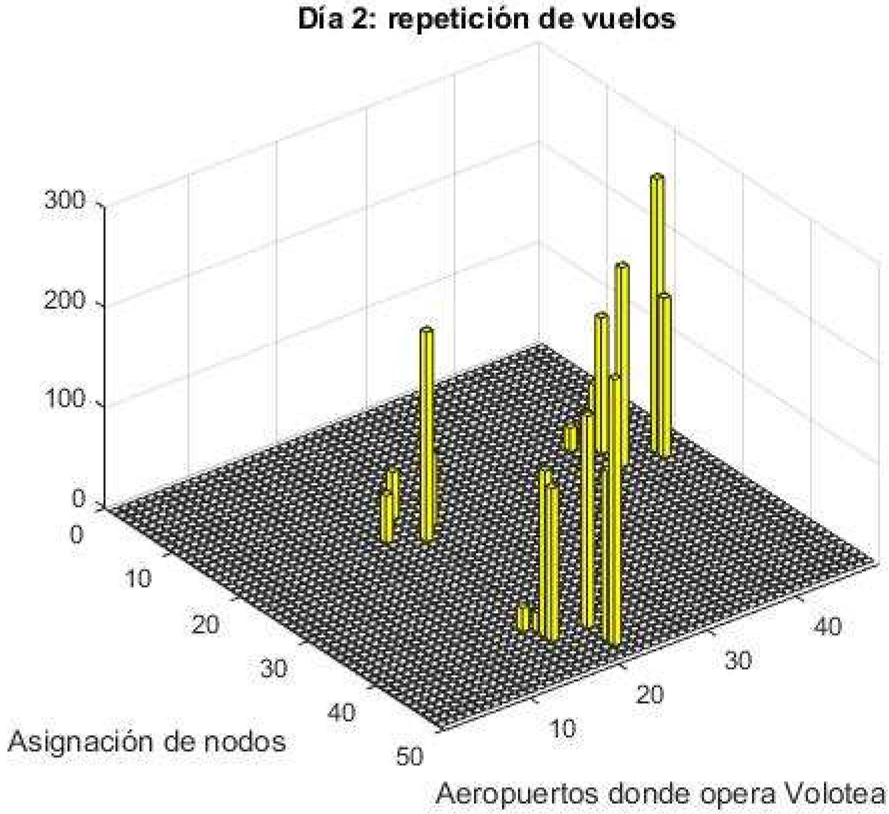


Figura 34: Aparición de vuelos en rotaciones de 2 días, día 2(14 noviembre), Volotea

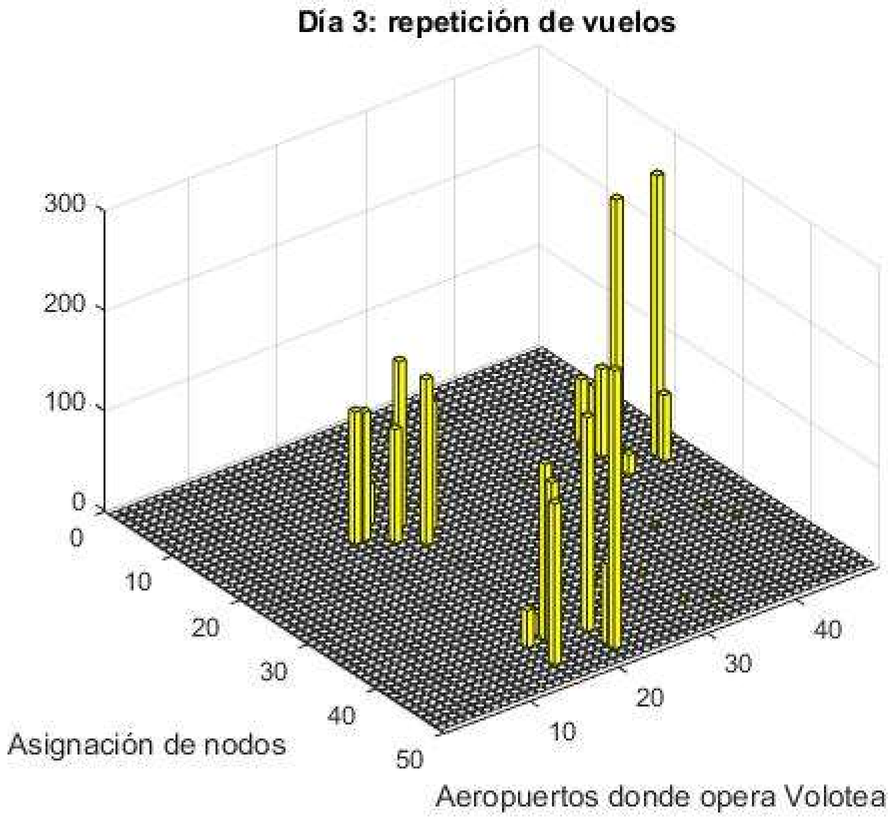


Figura 35: Aparición de vuelos en rotaciones de 2 días, día 3(15 noviembre), Volotea

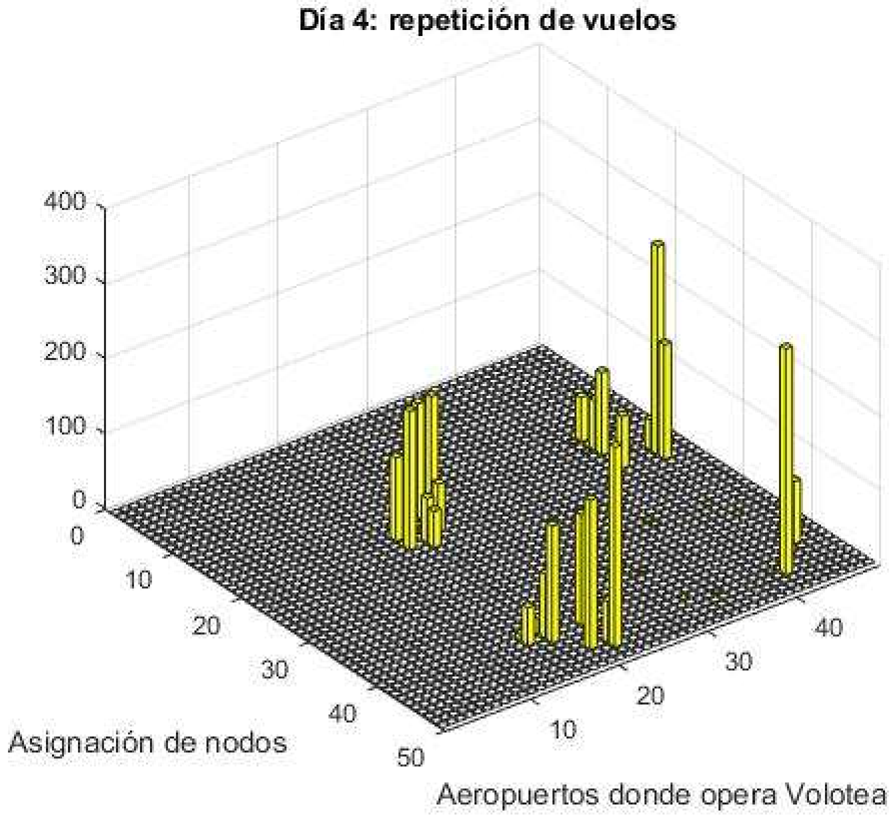


Figura 36: Aparición de vuelos en rotaciones de 2 días, día 4(16 noviembre), Volotea

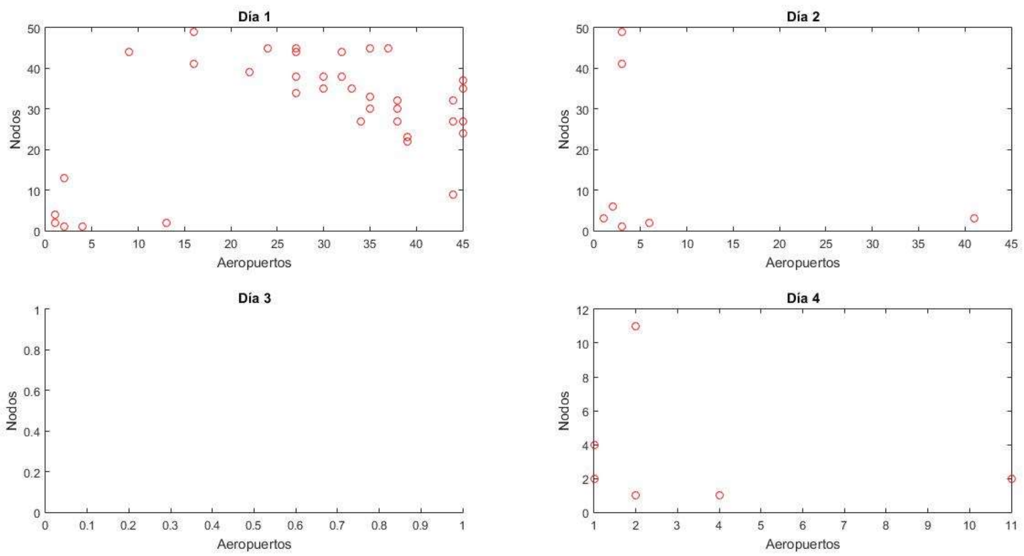


Figura 37: Vuelos no realizados por ninguna rotación en combinaciones de 2 días, Volotea

7 COMBINACIÓN DE ROTACIONES

Una vez se han obtenido todas aquellas rotaciones que cumplen con el ciclo propuesto (ciclos de 1 día, 2 días, ...), es necesario obtener la combinación de dichas rotaciones que permite realizar todos los vuelos sin repetir ninguno.

. Será necesario hacer uso de la programación lineal para poder obtener una solución real. Como es de esperar, el problema tendrá varias soluciones, así que habrá que imponer un criterio de optimización. Existen varios, aquellas rotaciones que permitan usar el menor número de aviones, aquellas combinaciones que hagan que todos los vuelos tiendan al mismo número de horas, aquella combinación que permita disminuir el tiempo de espera en tierra, o aumentarlo para poder realizar el mantenimiento, ... Se pueden imponer muchos criterios dependiendo de los intereses del usuario, y en este trabajo se incluirán algunos de ellos.

7.1. Método de programación lineal, SIMPLEX

El método usado para la resolución del problema se denomina comúnmente **SIMPLEX**. Se trata de un método de programación lineal, en nuestro caso, entera. En concreto, se hará uso de **MILP** (‘mixed integer linear programming’) por medio de la herramienta ***‘intlinprog’*** de *Matlab*. A continuación, se mostrará como es el proceso de resolución de este, y como se puede controlar de forma externa el método y la tolerancia de la solución obtenida.

El sistema se basa en definir una función objetivo **F**, a maximizar o minimizar, sujeta a una serie de restricciones (**R**). La función objetivo puede cambiar, y de hecho se resolverá el problema para distintos objetivos.

7.1.1 Restricciones

Las restricciones expresan que un vuelo solo puede ser realizado por una rotación, y que todos los vuelos deben aparecer en el total de rotaciones seleccionadas. Para obtener ecuaciones que expresen estas restricciones habrá que tratar **RT**.

Cada rotación realiza una serie de vuelos que están definidos por el origen i y el destino j, relacionados con la matriz de valores que se obtuvo en el apartado 3. El primer paso consiste en definir una matriz de 0 y 1 que indica los vuelos que realiza una determinada rotación r ($\mathbf{M_{r,\,kR}}$). En el caso de que la rotación sea de dos días, la matriz será tridimensional, pues habrá que diferenciar los vuelos según el día k que se realizan. Para ello M se inicializará como una matriz de ceros y k tan grande como días tenga la rotación. Si la rotación es de un día, M será bidimensional. Después se rellenará con valores unidad donde corresponda según los aeropuertos i, j de los que esté compuesta la rotación. k_R indica los días por los que está compuesta la rotación, y, por ende, la matriz M de vuelos. En el caso de que sea bidimensional k no aporta información, es redundante, pues k y k_R indican el día de estudio.

$$M_{r,k_R}(i,j,k) = 1 \tag{7–1}$$

Cada rotación tendrá asignada una variable $\mathbf{x_{r,kR}}$ que puede adoptar el valor 1 si la rotación se realiza, o 0 si la rotación no se lleva a cabo. Es decir, x es una variable de decisión. Si las restricciones mustran que todos los vuelos se deben realizar y sólo se pueden hacer una vez, eso se indicará por medio de **(6-2).** La ecuación se realizará para todos los vuelos (i, j, k) a realizar, es decir, que se tendrán tantas igualdades como vuelos tenga el sistema. $\mathbf{x_{r,kR}}$ es desconocida y es la variable que habrá que hallar por medio de la programación lineal.

$$\sum_{\forall r} M_{r,k_R}(i,j,k)\,x_{r,k_R} = 1 \tag{7–2}$$

En *Matlab* al final lo que se tendrá es una matriz (**RI**) donde cada fila representará el valor de M de la rotación para cada vuelo en concreto. (i - j) se indica pues RI expresa las restricciones para un vuelo (i, j, k) en concreto, no es un subíndice, simplemente la primera dimensión de RI será tan grande como vuelos halla en un determinado día. Al realizarse matricialmente, habrá días donde halla menos vuelos y sea necesario rellenar la matriz de ceros para evitar problemas dimensionales. Esas filas se eliminarán una vez halla que hacer el estudio, pero permiten guardar la información de forma más compacta.

$$RI(i-j,r,k) = M_{r,k_R}(i,j,k) \tag{7–3}$$

Una vez se han calculado las restricciones de igualdad, es necesario calcular las de desigualdad. Esta restricción es una única, e implica que el sistema nunca puede usar más aviones que los que dispone la compañía, pues sino la solución no se podrá llevar a la práctica. Sea **P** el número de aviones que posee una compañía A.

$$\sum_{\forall \mathrm{r}} x_{r,k} \leq P \tag{7–4}$$

La última restricción se guardará en una matriz (**RD**), según la ecuación **(6-5)**.

$$RD(r,k) = 1 \tag{7–5}$$

De esta forma, el sistema expresado de manera matricial quedará como se indica en **(6-6, 6-7)**.

$$RI_k\ \bar{x}_k = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \dots \end{bmatrix} \quad ; \quad \forall \mathrm{k} \tag{7–6}$$

$$RD_k\ \bar{x}_k \leq P \quad ; \quad \forall \mathrm{k} \tag{7–7}$$

7.1.2 Función objetivo

El primer objetivo que puede ser de interés es usar el menor número de aviones posibles. Puede ser que, a la compañía, a pesar de tener una flota determinada, por temas de mantenimiento le sea de especial interés poder realizar las rutas con menos aviones de los que posee. Usar menos aviones es lo mismo que decir que se realizan menos rotaciones pues cada rotación la realiza un avión en concreto.

1. Menor número de aviones posible.

   Para ello habrá que minimizar la ecuación **(6-8)** para cada día. En el caso de que sean rotaciones de dos días, habrá que realizar la minimización cada R días, siendo R la duración de la rotación.

   $$Min(\sum_{\forall \mathrm{r}} x_{r,k}) \tag{7–8}$$

   La función objetivo (**F**) será en este caso una matriz unidad tal que se cumpla la expresión mostrada en **(6-9, 6-10).**

   $$Min(F_k^T\ \bar{x}_k) \tag{7–9}$$

   $$F_k = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \dots \end{bmatrix} \tag{7–10}$$

   Donde F_k representa las rotaciones de un día en concreto, pero el estudio se debe realizar día a día, o en caso de rotaciones de más de un día, cada R días.

2. Pasar el mayor número de noches posibles en las bases de la compañía.

   Para alcanzar este objetivo primero es necesario definir cuales son las bases de la compañía. Para el caso de Canaryfly es Gran Canaria, mientras que para Volotea son Nantes, Toulouse y Burdeos. Habrá que ir rotación por rotación estudiando cuantas noches la pasan en dichos lugares.

   Supongamos una rotación de un día. Si dicha rotación pasase noche en una de las bases se le asignaría un valor unidad, en caso contrario tendría un valor nulo. Para rotaciones de dos días, hay más de una noche, de manera que se le podrá asignar a la rotación un valor de 2, 1 ó 0 según las noches que pase en una base. Así sucesivamente, para el caso de 3 noches, etc.

   Al estar buscando un máximo, pero ir a resolverlo por medio de un mínimo, se tiene la expresión **(6-11).**

$$Max\left(F_k^T\ \bar{x}_k\right) = Min(-F_k^T\ \bar{x}_k) \tag{7–11}$$

3. Que los aviones realicen el mismo número de horas de vuelo, o lo más parecidas posibles.

   Para conocer el número de horas de vuelo realizadas en una rotación, habrá que ir vuelo a vuelo calculando la duración de este y obteniendo el total.

$$F_k^T(r) = \sum_{j=1}^{J} (LL(i,j,k) - S(i,j,k)) - \mu_k \tag{7–12}$$

   Como el objetivo es que todos los vuelos esten equilibrados en sus horas, habrá que obtener la media de horas de vuelo totales del sistema que dependerá del número de aviones que operen. Donde P' será el número de aviones de la combinación de las rotaciones, siempre inferior a P. P' es desconocida a priori, habrá que suponer un valor y comprobar que la solución encontrada coincide con el número de aviones propuesto.

$$\mu_k = \sum_{i=1}^{C_M} \sum_{j=1}^{B} (LL(i,j,k) - S(i,j,k))\ /P' \tag{7–13}$$

4. Obtener un valor óptimo para todas las funciones anteriores.

   Puede que hallamos obtenido un óptimo para un objetivo en concreto, pero dicha solución es mala para el resto de los objetivos. Por ello, por medio de la ponderación, habrá que obtener cual es la solución que, aunque no optimice un objetivo en concreto, si lo haga de forma global. El factor de ponderación se denominará w. Cuando se quiera que todas las funciones tengan la misma importancia se le dará un valor de 1/3 a la ponderación.

$$F_k^T = w_1 F_{k\ 1}^T + w_2 F_{k\ 2}^T + w_3 F_{k\ 3}^T \tag{7–14}$$

$$\sum_{1}^{3} w_i = 1 \tag{7–15}$$

   En el caso de que se quiera dar más importancia a alguna de las funciones objetivo frente al resto, pero seguir teniendo en cuenta todos los factores, se irá variando w y aumentará el valor para aquella F que tenga mayor relevancia en el estudio.

7.2. Casos prácticos

7.2.1. Canaryfly

Hagamos un estudio del primer objetivo, usar el menor número de aviones posibles teniendo en cuenta que Canaryfly cuenta con una flota de 5 aviones. Para ello, sabemos que la función objetivo será la descrita en **(6-10)**. Si se realiza el estudio se obtiene que el mínimo número de aviones con el que se pueden realizar todas las rotaciones sin repetir ninguna es 4 aviones. Las rotaciones que permiten resolver el sistema son las siguientes.

Tabla 3: Combinación de rotaciones con el mínimo número de aviones, Canaryfly

Avión	Rotaciones											
Avión 1	2→	4→	3→	11→	2→	1→	7→	9→	2			
Avión 2	4→	6→	11→	17→	11→	9→	12→	18→	13→	19→	4	
Avión 3	4→	7→	6→	7→	12→	8→	12→	10→	4			
Avión 4	20→	5→	8→	14→	2→	3→	2→	15→	2→	17→	7→	16→	2

Si la función objetivo es minimizar el número de noches que pasa fuera de Gran Canaria, la base de la compañía aparece un resultado curioso, y es que no existe una única solución, sino que, si se ejecuta varias veces, el programa tiende a una solución distinta pero igual de óptima. A continuación, se mostrarán tres resultados distintos obtenidos. Esto propicia que el método converja rápidamente, pues debido a las condiciones de las rotaciones es relativamente fácil encontrar soluciones que satisfagan el problema, aunque el punto inicial de estudio del algortimo varíe.

Tabla 4: Combinación de rotaciones con el menor número de noches fuera de Gran Canaria, solución 1, Canaryfly

Avión	Rotaciones									
Avión 1	2→	4→	3→	11→	17→	12→	2			
Avión 2	4→	6→	11→	8→	12→	10→	4			
Avión 3	4→	7→	11→	9→	12→	18→	13→	19→	4		
Avión 4	7→	6→	2→	15→	2→	17→	7→	16→	2		
Avión 5	20→	5→	8→	14→	2→	3→	2→	1→	7→	9→	2

Tabla 5: Combinación de rotaciones con el menor número de noches fuera de Gran Canaria, solución 2, Canaryfly

Avión	Rotaciones									
Avión 1	2→	4→	7→	6→	2→	15→	2→	17→	7→	16→	2
Avión 2	4→	3→	11→	9→	12→	18→	13→	19→	4		

Avión 3	4→	6→	11→	17→	12→	8→	12→	10→	4
Avión 4	7→	11→	2→	1→	7→	9→	2		
Avión 5	20→	5→	8→	14→	2→	3→	2		

Si se escoge la función en la que todos los vuelos realicen el mismo número de horas aproximadamente, se obtiene que para 4 aviones el número de horas media es de 6.125 horas de vuelo, y la desviación total de los cuatro aviones del sistema en la solución óptima es de 3 horas.

Tabla 6: Combinación de rotaciones con la media de horas de vuelo más parecida, Canaryfly

Avión	Rotaciones												
Avión 1	2→	4→	3→	11→	2→	1→	7→	9→	2				
Avión 2	4→	6→	11→	17→	12→	8→	12→	10→	4				
Avión 3	4→	7→	6→	7→	11→	9→	12→	18→	13→	19→	4		
Avión 4	20→	5→	8→	14→	2→	3→	2→	15→	2→	17→	7→	16→	2

Si se realiza un estudio para que se cumplan todos los criterios con una importancia de 1/3, se obtiene la solución indicada en la tabla **(7)**. Se puede comprobar que no solo es el óptimo global, sino que es también la más óptima para todos los objetivos por separado (compartiendo óptimo con otras para número de aviones y noches en Gran Canaria)

Tabla 7: Solución óptima para los tres objetivos propuestos

Avión	Rotaciones												
Avión 1	2→	4→	3→	11→	2→	1→	7→	9→	2				
Avión 2	4→	6→	11→	17→	12→	8→	12→	10→	4				
Avión 3	4→	7→	6→	7→	11→	9→	12→	18→	13→	19→	4		
Avión 4	20→	5→	8→	14→	2→	3→	2→	15→	2→	17→	7→	16→	2

El hecho de que exista más de una solución óptima para cada objetivo por separado, no en el global que, si es único, se debe a que al haber gran cantidad de vuelos en una zona geográficamente reducida y tiempos de vuelo inferiores a una hora puede haber una cantidad de combinaciones elevada, pero sin grandes diferencias entre las rotaciones, salvo intercambio de vuelos entre aviones. Aún así, la solución mostrada en la tabla **(7)** y la **Figura 38** es la combianción de rotaciones óptima y que resuelve de mejor manera el problema de estudio.

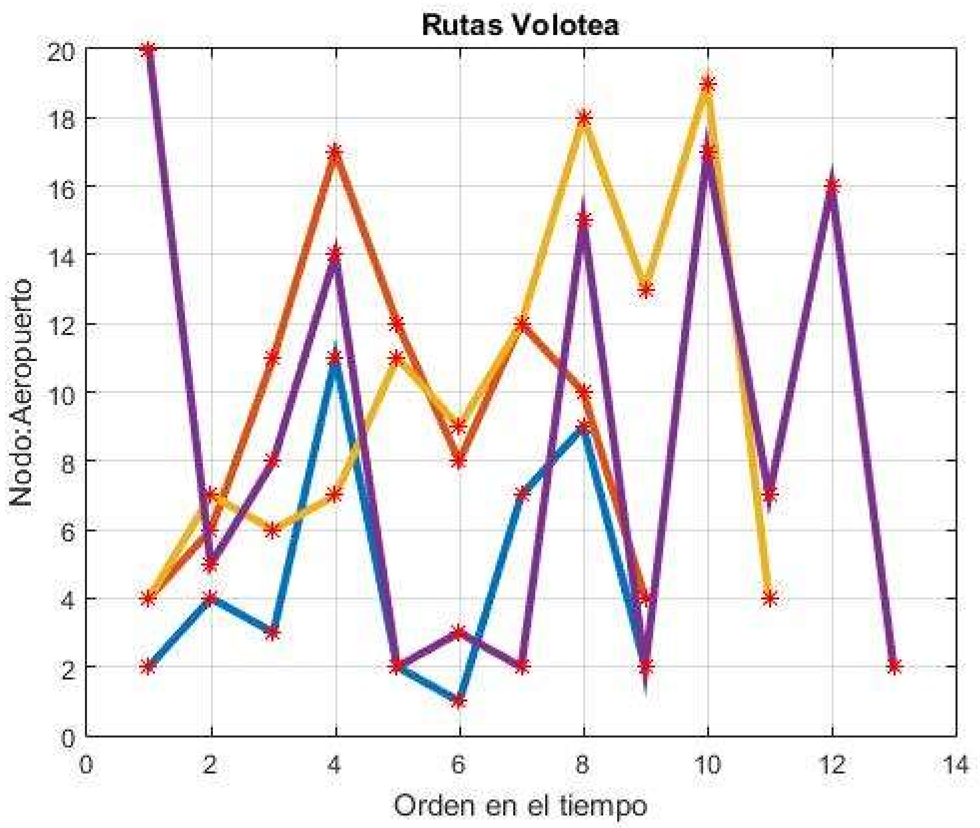


Figura 38: Solución mostrada en la Tabla 7

Si representamos el resultado de manera más visual, misma información que la Figura 38, donde las imágenes superiores representan los aviones 1 y 2 de izquierda a derecha y las inferiores los aviones 3 y 4. Los números indican el número de vuelo en la rotación en concreto. Si existen varios valores, como por ejemplo (1,2,3,4) significa que el primer, segundo, tercer y cuarto vuelo son entre esos dos destinos en concreto.

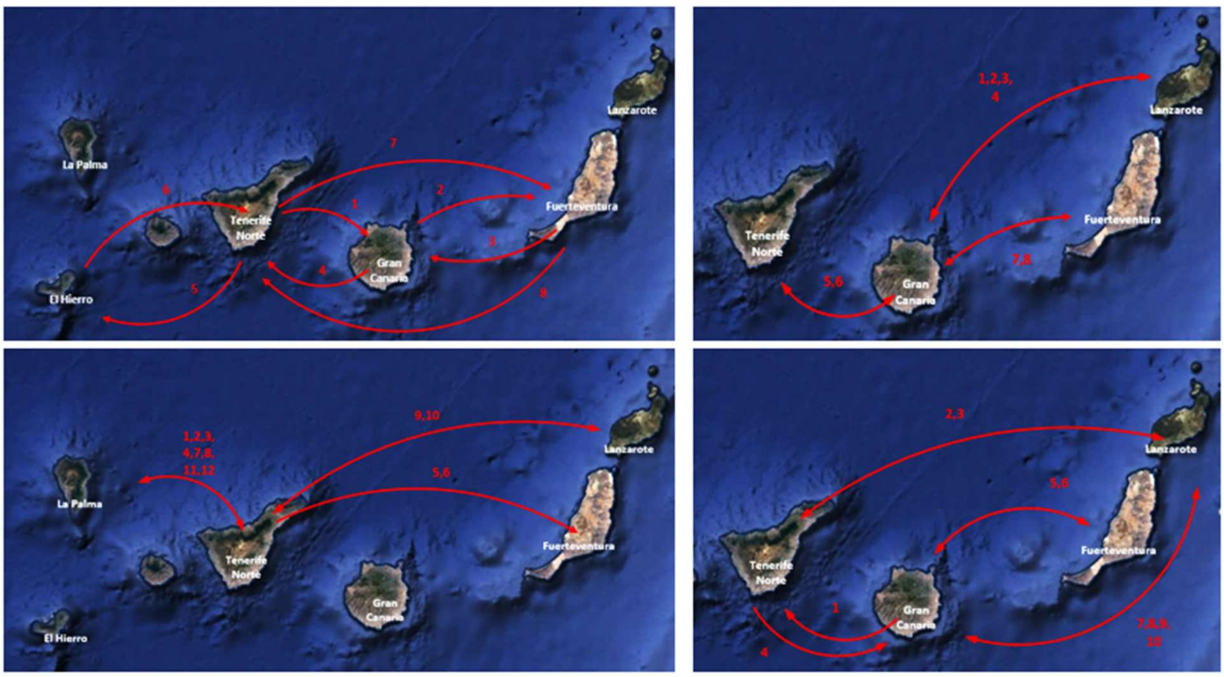


Figura 39: Rotaciones Canaryfly sobre el mapa de Gran Canarias

Como se puede ver la mayoría de aviones operan entre un conjunto de islas determinado realizando ida y vuelta sin parar. Esto hace que el sistema sea eficiente pues cada aeronave está especializada en una ruta en concreto, y también la tripulación. Sin embargo, a la hora de realizar trasbordo esto hace que los pasajeros deban abandonar el vuelo y coger otro avión si el objetivo es realizar el enlace con una isla determinada.

7.2.2. Volotea

Para Volotea hay que recordar que, aunque existían soluciones para rotaciones de 1 día, pero no para rotaciones de 2 días, si se combinaban entre sí, se podía obtener una solución al sistema al completo. Al igual que para Canaryfly, se hará el estudio para las distintas funciones objetivo que han sido propuestas.

Si se obtienen las soluciones relativas a usar el menor número de aviones posibles, se obtienen que para los días 13 y 14 con 19 aviones se podrían realizar los vuelos mientras que para los días 15 y 16 se podrían realizar todos los vuelos con 16 aviones. Si la compañía tiene 27 aviones en su flota es debido a que es una empresa que opera principalmente en temporadas turísticas y los vuelos invernales se ven reducidos sustancialmente, siendo el momento de realizar el mantenimiento de la mayoría de sus aeronaves.

Tabla 8: Combinaciones para días 13,14, menor número de aviones posible, Volotea

Aviones	Rotaciones											
Avión 1	46→	22→	46→	20→	47→	22→	19→	**42**→	18→	43→	23→	46
Avión 2	22→	42→	16→	43→	23→	**20**→	17→	40→	17→	20→	22	
Avión 3	20→	22→	20→	18→	21→	**22**→	47→	22→	20			
Avión 4	18→	15→	18→	23→	**46**→	22→	43→	18				
Avión 5	41→	18→	42→	46→	43→	**18**→	41					
Avión 6	42→	18→	**41**→	19→	42							
Avión 7	2→	13→	2→	1→	4→	1→	**2**					
Avión 8	38→	32→	38→	30→	35→	30→	**38**					
Avión 9	44→	32→	44→	9→	**44**							
Avión 10	45→	27→	38→	27→	**45**							
Avión 11	45→	35→	33→	35→	**45**							
Avión 12	46→	16→	18→	16→	**46**							
Avión 13	45→	37→	45→	24→	**45**							
Avión 14	44→	27→	34→	27→	**44**							
Avión 15	41→	16→	**41**									
Avión 16	39→	22→	**39**									
Avión 17	42→	23→	**42**									
Avión 18	**41**→	3→	1→	3→	41							

Avión 19	**2**→	6→	2

Tabla 9: Combinaciones para días 15,16, menor número de aviones posible, Volotea

Aviones	Rotaciones									
Avión 1	20→	18→	15→	18→	20→	**22**→	20→	18→	21→	23→	20
Avión 2	42→	16→	43→	**18**→	42→	46→	42→	22→	42		
Avión 3	46→	22→	47→	22→	**20**→	22→	46→	21→	46		
Avión 4	18→	43→	23→	**42**→	16→	43→	18				
Avión 5	42→	18→	16→	**46**	**47**→	22→	19→	42			
Avión 6	46→	16→	18→	**41**→	18→	23→	46				
Avión 7	41→	19→	**42**→	18→	41						
Avión 8	32→	**38**→	31→	38→	32						
Avión 9	23→	**46**→	22								
Avión 10	38→	**32**→	38								
Avión 11	44→	32→	**44**								
Avión 12	45→	35→	**45**								
Avión 13	**2**→	11→	2→	1→	4→	1→	2				
Avión 14	**45**→	41→	45→	35→	45						
Avión 15	**41**→	16→	41								
Avión 16	**44**→	32→	44								

Habrá que comprobar si existe conexión entre las rotaciones de los días 13, 14 y los días 15, 16. En este caso no es totalmente necesario pues la compañía posee más aviones, pero si es recomendable para que exista continuidad en los vuelos. Se puede ver como los vuelos están concatenados, salvo para el vuelo 16 (día 13) y el vuelo 8 (días 15 y 16). Es decir, para el total de los cuatro días se necesitarán en total 20 aviones como mínimo, de un total de 27 que es la flota. Como ya se ha mencionado, esto se debe a que en invierno el número de vuelos disminuye considerablemente con respecto a las campañas turísticas.

Tabla 10: Combinación entre rotaciones de días distintos F_1, Volotea

Días 13, 14	Días 15, 16
Avión 3	Avión 1

Avión 6, Avión 17	Avión 2
Avión 1, Avión 12	Avión 3
Avión 4	Avión 4
Avión 6, Avión 17	Avión 5
Avión 1, Avión 12	Avión 6
Avion 5, Avión 18	Avión 7
-	Avión 8
Avión 2	Avión 9
Avión 8	Avión 10
Avión 9, Avión 14	Avión 11
Avión 10, Avión 11, Avión 13	Avión 12
Avión 7, Avión 19	Avión 13
Avión 10, Avión 11, Avión 13	Avión 14
Avión 5, Avión 15, Avión 18	Avión 15
Avión 9, Avión 14	Avión 16
Avión 16	-

Si se realiza un estudio para obtener la solución con el menor número de noches posibles fuera de los aeropuertos base, se obtiene que es necesario usar los 28 aviones de la flota. Esto era predecible, pues cuantos más aviones, más pequeñas son las rotaciones y permiten controlar mejor los aeropuertos de origen y destino.

Tabla 11: Combinación para los días 13, 14, menor número de noches fuera de los aeropuertos base, Volotea

Aviones	Rotaciones						
Avión 1	2→	13→	2→	1→	4→	1→	**2**
Avión 2	46→	22→	47→	22→	18→	23→	**46**
Avión 3	20→	22→	20→	18→	**20**		
Avión 4	38→	30→	35→	30→	**38**		
Avión 5	41→	18→	15→	18→	**41**		
Avión 6	42→	16→	43→	19→	**42**		
Avión 7	42→	22→	43→	19→	**42**		
Avión 8	44→	27→	34→	27→	**44**		

Avión 9	44→	32→	44→	9→	**44**		
Avión 10	45→	27→	38→	27→	**45**		
Avión 11	45→	35→	33→	35→	**45**		
Avión 12	45→	37→	45→	24→	**45**		
Avión 13	46→	16→	18→	16→	**46**		
Avión 14	21→	23→	**20**				
Avión 15	38→	32→	**38**				
Avión 16	39→	22→	**39**				
Avión 17	41→	16→	**41**				
Avión 18	42→	22→	**42**				
Avión 19	42→	46→	**42**				
Avión 20	46→	20→	**46**				
Avión 21	42→	18→	43→	22→	43→	19→	42
Avión 22	20→	17→	40→	17→	20		
Avión 23	41→	3→	1→	3→	41		
Avión 24	**2**→	6→	2				
Avión 25	**20**→	22→	20				
Avión 26	**41**→	18→	41				
Avión 27	**46**→	22→	46				
Avión 28	**47**→	23→	46				

Tabla 12: Combinación para los días 15, 16, menor número de noches fuera de los aeropuertos base, Volotea

Aviones	Rotaciones									
Avión 1	42→	18→	16→	47→	23→	**46**→	22→	47→	22→	19→	42
Avión 2	46→	16→	**18**→	23→	46						
Avión 3	18→	**42**	**43**→	18							
Avión 4	20→	18→	15→	18→	**20**						
Avión 5	20→	22→	**20**								

Avión 6	32→	38→	**32**				
Avión 7	41→	18→	**41**				
Avión 8	42→	16→	**42**				
Avión 9	43→	19→	**42**				
Avión 10	43→	23→	**42**				
Avión 11	44→	32→	**44**				
Avión 12	45→	35→	**45**				
Avión 13	46→	22→	**46**				
Avión 14	**20**→	22→	20→	18→	21→	23→	20
Avión 15	**2**→	11→	2→	1→	2		
Avión 16	**1**→	4→	1				
Avión 17	**32**→	38→	32				
Avión 18	**38**→	31→	38				
Avión 19	**41**→	16→	41				
Avión 20	**41**→	18→	41				
Avión 21	**42**→	16→	42				
Avión 22	**42**→	18→	42				
Avión 23	**42**→	22→	42				
Avión 24	**42**→	46→	42				
Avión 25	**44**→	32→	44				
Avión 26	**45**→	35→	45				
Avión 27	**45**→	41→	45				
Avión 28	**46**→	21→	46				

Si realizamos la comparativa de las rotaciones que se concatenan en los distintos días, se obtienen los resultados mostrados en **(Tabla 13)**.

A pesar de que para cada jornada con 28 es suficiente, al unir las dos combinaciones se torna imposible realizar el problema. Es decir, habrá que buscar soluciones con menor número de rotaciones, es decir, aviones, pues sino se resolverá el problema a trozos, pero no de manera general. Es por ello, que se resolverá el problema para un óptimo intermedio, con w de valor ½ y dos funciones objetivo. F_1 será usar el mínimo número de aviones posible y F_2 será pasar fuera de los aeropuertos base el menor número de noches posible.

Tabla 13: Combinación entre rotaciones de días distintos F_2, Volotea

Días 13, 14	Días 15, 16
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 1
Avión 2, Avión 13, Avión 20, Avión 27, Avión 28	Avión 2
-	Avión 3
Avión 3, Avión 14, Avión 22	Avión 4
Avión 3, Avión 14, Avión 22	Avión 5
-	Avión 6
Avion 5, Avión 17, Avión 23, Avión 26	Avión 7
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 8
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 9
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 10
Avión 8, Avión 9	Avión 11
Avión 10, Avión 11, Avión 12	Avión 12
Avión 2, Avión 13, Avión 20, Avión 27, Avión 28	Avión 13
Avión 3, Avión 14, Avión 22	Avión 14
Avión 1, Avión 24	Avión 15
-	Avión 16
-	Avión 17
Avión 4, Avión 15	Avión 18
Avion 5, Avión 17, Avión 23, Avión 26	Avión 19
Avion 5, Avión 17, Avión 23, Avión 26	Avión 20
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 21
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 22
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 23
Avión 6, Avión 7, Avión 18, Avión 19, Avión 21	Avión 24
Avión 8, Avión 9	Avión 25
Avión 10, Avión 11, Avión 12	Avión 26

Avión 10, Avión 11, Avión 12	Avión 27
Avión 2, Avión 13, Avión 20, Avión 27, Avión 28	Avión 28

Tabla 14: Combinación para los días 13, 14, combinación F_1 y F_2, Volotea

Aviones	Rotaciones								
Avión 1	46→	22→	46→	20→	47→	22→	18→	23→	**46**
Avión 2	2→	13→	2→	1→	4→	1→	**2**		
Avión 3	20→	22→	20→	18→	21→	23→	**20**		
Avión 4	38→	32→	38→	30→	35→	30→	**38**		
Avión 5	42→	18→	42→	46→	43→	19→	**42**		
Avión 6	42→	22→	42→	16→	43→	23→	**42**		
Avión 7	41→	18→	15→	18→	**41**				
Avión 8	44→	27→	34→	27→	**44**				
Avión 9	44→	32→	44→	9→	**44**				
Avión 10	45→	27→	38→	27→	**45**				
Avión 11	45→	35→	33→	35→	**45**				
Avión 12	45→	37→	45→	24→	**45**				
Avión 13	46→	16→	18→	16→	**46**				
Avión 14	39→	32→	**39**						
Avión 15	41→	16→	**41**						
Avión 16	**20**→	17→	40→	17→	20→	22→	20		
Avión 17	**42**→	18→	43→	22→	43→	19→	42		
Avión 18	**41**→	3→	1→	3→	41				
Avión 19	**46**→	22→	47→	23→	46				
Avión 20	**2**→	6→	2						
Avión 21	**41**→	18→	41						

Tabla 15: Combinación para los días 15, 16, combinación F_1 y F_2, Volotea

Aviones	Rotaciones										
Avión 1	42→	18→	16→	47→	23→	**46**→	22→	47→	22→	19→	42
Avión 2	46→	16→	18→	20→	22→	**20**→	22→	20→	18→	23→	46
Avión 3	20→	18→	15→	19→	**42**→	18→	21→	23→	20		
Avión 4	42→	16→	43→	**18**→	42→	46→	42→	22→	42		
Avión 5	18→	43→	23→	**42**→	16→	43→	18				
Avión 6	32→	38→	**32**								
Avión 7	41→	18→	**41**								
Avión 8	44→	32→	**44**								
Avión 9	45→	35→	**45**								
Avión 10	46→	22→	**46**								
Avión 11	**2**→	11→	2→	1→	4→	1→	**2**				
Avión 12	**45**→	41→	45→	35→	45						
Avión 13	**32**→	38→	32								
Avión 14	**38**→	31→	38								
Avión 15	**41**→	16→	41								
Avión 16	**41**→	18→	41								
Avión 17	**44**→	32→	44								
Avión 18	**46**→	21→	46								

Si se realiza de nuevo una comparativa de las rotaciones que se pueden concatenar se observan las combinaciones de aviones entre los distintos días, lo que permitirá tener conocimiento del número de aviones total necesarios para el estudio.

Tabla 16: Combinación entre rotaciones de días distintos F_2 y F_1, Volotea

Días 13, 14	Días 15, 16
Avión 5, Avión 6, Avión 17	Avión 1
Avión 1, Avión 13, Avión 19	Avión 2

Avión 3, Avión 16	Avión 3
Avión 5, Avión 6, Avión 17	Avión 4
-	Avión 5
-	Avión 6
Avion 7, Avión 15, Avión 18, Avión 21	Avión 7
Avión 8, Avión 9	Avión 8
Avión 10, Avión 11, Avión 12	Avión 9
Avión 1, Avión 13, Avión 19	Avión 10
Avión 20	Avión 11
Avión 10, Avión 11, Avión 12	Avión 12
-	Avión 13
Avión 4	Avión 14
Avion 7, Avión 15, Avión 18, Avión 21	Avión 15
Avion 7, Avión 15, Avión 18, Avión 21	Avión 16
Avión 8, Avión 9	Avión 17
Avión 1, Avión 13, Avión 19	Avión 18

En total en los 4 días de estudio y como solución óptima, se obtuvo que son necesarios 24 aviones de 28 totales para poder realizar el total de vuelos. A continuación se muestran sobre el mapa real algunos de los vuelos para entender no sólo numéricamente hacia donde va sino el lugar exacto. Como se aprecia, la mayoría de los vuelos opera en una región determinada no muy amplia. En línea discontínua se muestran los vuelos realizados en el segundo día de la rotación, en el caso de que esta dure más de un día.

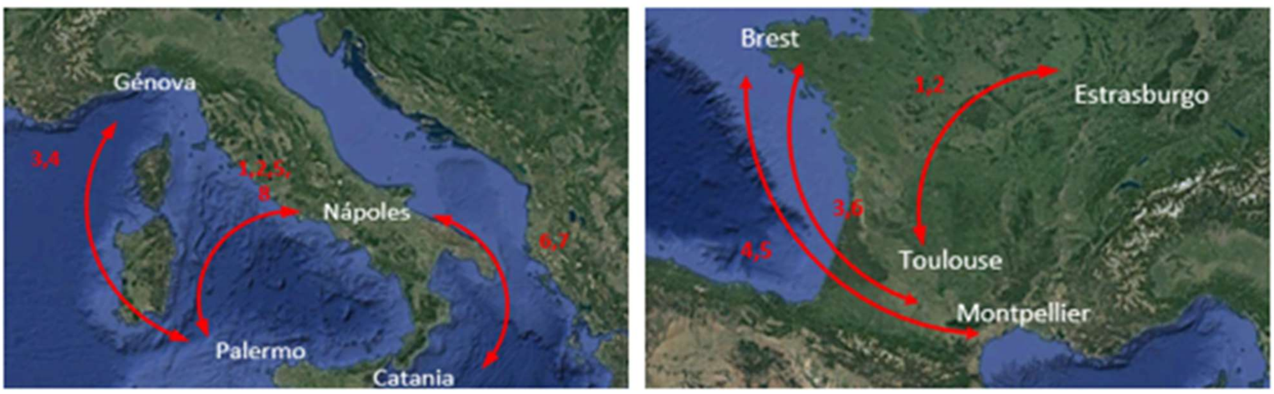


Figura 40: Avión 1 y 4, Volotea sobre el mapa real, día 13

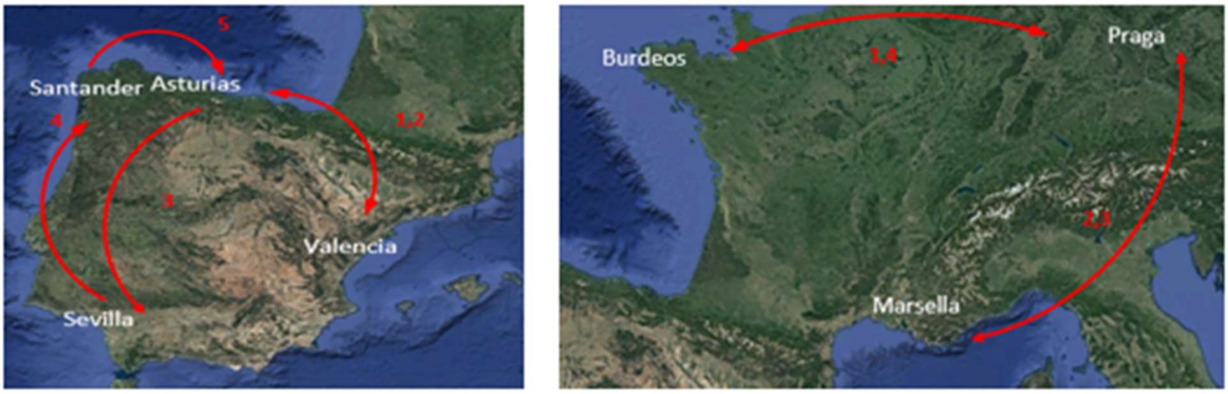


Figura 41: Avión 2 y 8, Volotea sobre el mapa real, día 13

Figura 42: Avión 1, Volotea sobre el mapa real, día 15-16

8 CAMBIO DE HORARIOS

A continuación, se pretende mostrar las combinaciones nuevas que aparecerían en el caso de que los vuelos se vieran retrasados o adelantados. Para ello, se desarrolla el algoritmo seguido para su obtención. Si se recuerda el método usado para encontrar tríos de vuelos, o lo que es lo mismo los pares salida-llegada, entre aeropuertos. El objetivo aquí será variar el horario vuelo a vuelo y observar los cambios sufridos en dichos pares con respecto a los pares originales obtenidos al inicio del trabajo con los horarios reales.

Sin embargo, el código se puede reutilizar. Si recordamos las ecuaciones usadas, **(7-1)**, los horarios se ajustaban con el tiempo T que un avión debe permanecer en tierra antes de volver a despegar.

$$LL(i,j,k) + T \leq S(j,l,k), \qquad i,j,l = 1,2,\ldots,B; k = 1,2,\ldots,D \tag{8–1}$$

Este parámetro T lo fijamos, pero que ocurriría si lo reducimos. Si T fuese inferior al valor real, significaría que el vuelo puede permanecer en tierra menos tiempo. Sin embargo, también hay otra manera de plantearlo, y es que, si se reduce el tiempo T, es lo mismo que decir que el tiempo de estancia en tierra es el mismo pero que el horario de llegada se ha adelantado. Llamemos T' al tiempo que se reduce el valor T.

$$LL(i,j,k) + (T - T') \leq S(j,l,k) \tag{8–2}$$

$$(LL(i,j,k) - T') + T \leq S(j,l,k) \tag{8–3}$$

$$LL(i,j,k) + T \leq S(j,l,k) + T' \tag{8–4}$$

Las tres expresiones enunciadas son idénticas, salvo por el concepto que representan. En la ecuación **(7-3)** se indica que el valor de llegada LL(i, j, k) se ha adelantado un valor T' mientras que en la ecuación **(7-4)** se ha atrasado el valor de la salida del vuelo S(j, l, k). Es decir, sin más que variar el valor T podemos obtener como afectaría el atrasar o adelantar un vuelo determinado.

El estudio se va a realizar para los tríos de vuelos, no para rotaciones al completo pues sólo interesa saber que aperturas de vuelo se producirían y no que combinaciones ficticias se obtendrían de dicho cambio. Supongamos que existe un trío que no aparecía con los horarios originales.

$$M \rightarrow N \rightarrow E \;/\; T' \neq 0 \tag{8–5}$$

En este caso las rotaciones que se formen con el trío (M, N, E) todos los vuelos anteriores a M deberán estar adelantados T' o todos los vuelos que vayan después de E deberán estar retrasados T', donde T' es siempre positivo.

8.1. Casos prácticos

8.1.1 Canaryfly

Si probamos a reducer T en diez minutos, es decir, le asignamos a T' un valor de 10 minutos (10/60 en el programa), se obtienen sólo una nueva posible combinacion a realizar. Como se puede observar el vuelo de 15 a 2 es la única ventana de salida nueva en el sistema. Si se analiza el avión que realizaba dicho vuelo en el resultado óptimo obtenido en la **(Tabla 7)**.

Tabla 17: Resultado óptimo del vuelo 9→12

Avión 4	20→	5→	8→	14→	2→	3→	2→	15→	2→	17→	7→	16→ 2

La salida de 15→ 2 es a las 15:20 mientras que la llegada de 2→ 15 es a las 14: 50. Si se adelanta el vuelo diez minutos, para cumplir la restricción de T > 20 min es necesario retrasar el vuelo 11→ 9 sólo 5 minutos. El vuelo 7→ 11 llega a las 14:25 de forma que no habría que cambiar más horarios en la rotación obtenida y se conseguirían tres aperturas más en los vuelos desde 12.

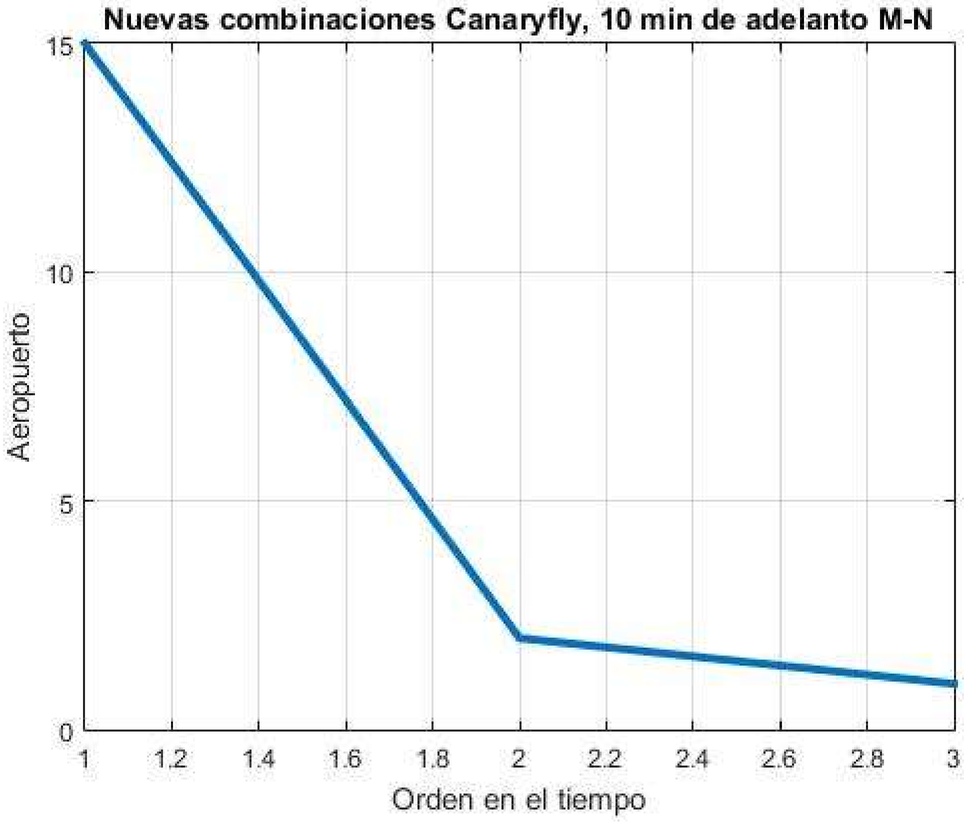


Figura 43: Combinaciones nuevas con T' de valor 10 minutos, Canaryfly.

Si realizamos los cambios indicados y calculamos todas las rotaciones de 1 día de duración y optimizamos el sistema para poder compararlo con el original. Se obtienen en total 1186 rotaciones al día de 1 día de duración. Si se procede a su optimización tanto para F_1 como para F_2., al igual que para el caso anterior se necesitan 4 aviones para realizar los vuelos. De hecho, se obtienen los mismos resultados, lo que quiere decir que el sistema está lo suficientemente ajustado como para que los cambios en los horarios no lleven a una mejora de las rotaciones. Esto quiere decir que Canaryfly opera de forma eficiente, y no necesita cambios en los horarios actuales.

8.1.2 Volotea

Se realiza el mismo procedimiento que para Canaryfly, se procede a reducir T en diez minutos, para ver cuales son las nuevas ventanas de salidas que se generan en el problema. En este caso aparecen cinco nuevas rotaciones que pueden realizarse. Habrá que observar cuando se realizan los vuelos iniciales para poder acometer cambios en los horarios para que no halla incoherencias en los tiempos de espera en tierra.

Tabla 18: Rotaciones óptimas que contemplan los vuelos con nuevas combinaciones

Avión 10	45→	27→	38→	27→	**<u>45</u>**				
Avión 1	46→	22→	46→	20→	47→	22→	18→	23→	**<u>46</u>**

Avión 3	20→	22→	20→	18→	21→	23→	<u>**20**</u>

Todos los vuelos nuevos aparecen en el día 13, el resto de los días no tienen combinaciones nuevas, es decir, están ajustados al máximo. Hay una combinación que es 22→ 21 que no aparece en ningún vuelo. Esto se debe a que el aeropuerto 21 también es el 20 en la asignación dada, como aeropuerto de llegada no aportaba información, pero sí como salida.

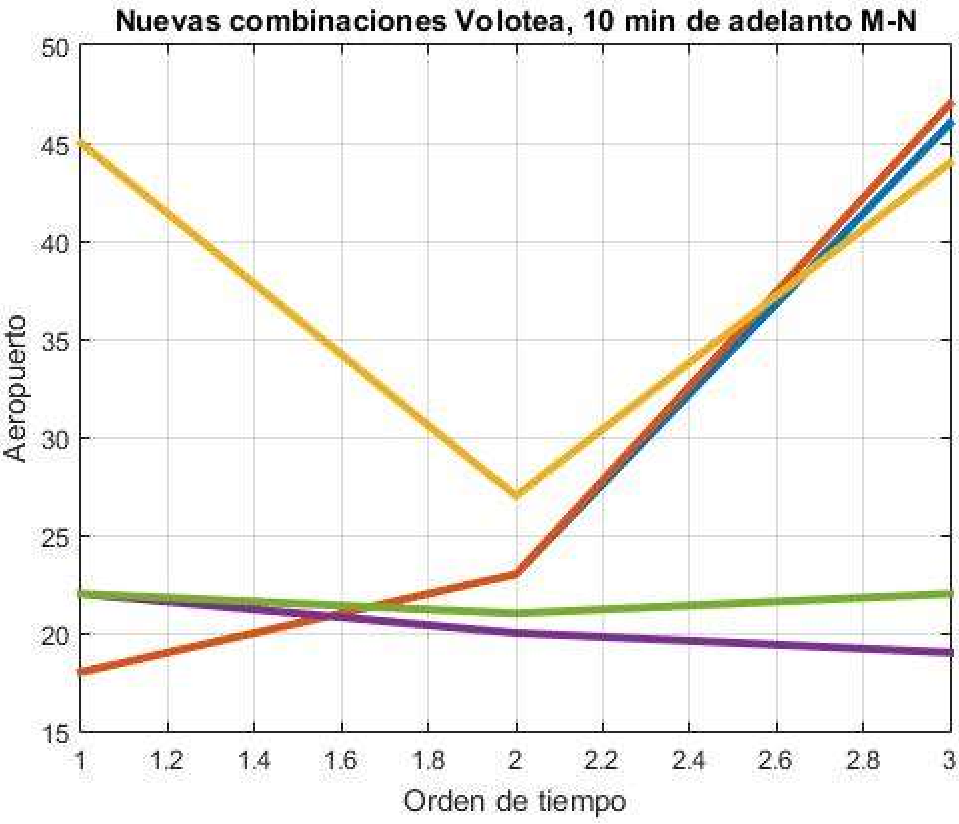


Figura 44: Combinaciones nuevas con T' igual a 10 minutos, día 13, Volotea

En el caso del vuelo **45→ 27** sólo sería necesario cambiar el horario de dicho vuelo pues es el primero de la rotación. En el vuelo **18→ 23**, hay que estudiar el vuelo anterior 22→ 18, donde el horario de salida del primero es a las 18:15 y la llegada del vuelo 22→ 18 es a las 17:50. Es decir el vuelo 22→18 deberá ser adelantado al menos 10 minutos para cumplir con los límites actuales,17:40. Habrá que seguir el estudio para comprobar que no se imcuple con el vuelo 47→ 22, que llega a las 16:20 a su destino, cuando el vuelo siguiente salía a las 16:45, a las 16:35 ahora. Habrá que adelantar a las 16:10. El vuelo 20→47 llega a las 14:15, mientras que el vuelo 47→ 22 sale a las 15:20. Es decir, el vuelo 20: 47 no es necesario variar su horario.

Si estudiamos el vuelo **22→ 20** se tiene que su salida es a las 20:45 mientras que el vuelo 20→ 22 llega a las 8:20, es decir, no es necesario cambiar horario alguno a parte del adelanto del vuelo seleccionado para nuevas rotaciones.

Si realizamos los cambios pertinentes en los vuelos se podrían abrir dichas ventanas de despegue, que no quiere decir que interese en términos de optimización.

9 CONCLUSIONES

Destacar que el tiempo de ejecución de los programas es proporcional al número de combinaciones posibles y no al número de vuelos de una compañía en concreto. Si existen pocas posibilidades de concatenar vuelos, el tiempo de ejecución será bajo. Los algoritmos ejecutados en global tardan unos 3 minutos en realizar todos los cálculos, desde importar los valores de los horarios hasta alcanzar la solución óptima al problema. Aún así, en el caso de que se quisieran estudiar sistemas más grandes y un número de días más elevado, del orden de meses, el sistema podría comenzar a tardar demasiado. Por ello, aunque no se ha realizado de forma implícita en el trabajo, se realizaron simulaciones paralelizando Matlab, con la capacidad de 8 núcleos del ordenador (utilizando 4 de ellos), y el tiempo de ejecución se reducía drásticamente. Es evidente, que este tipo de problemas donde la combinatoria puede llegar a ser un inconveniente, es fundamental realizar algoritmos donde las ecuaciones puedan ser realizadas de manera paralela y no secuencial.

Como empresa Volotea es mayor, sin embargo, Canaryfly posee un mayor número de rotaciones posibles. Conocer la situación geográfica de operación, así como el tipo de vuelos que realiza y los aviones que opera es importante de cara a comprender los resultados obtenidos. Volotea, al ser aviones de mayor tamaño, requiere tiempos mayores de estancia en tierra, lo que limita las combinaciones. Además, los aeropuertos están alejados entre sí de forma que las posibles rotaciones están limitadas. Canaryfly por su lado, prácticamente todos los vuelos se pueden combinar entre sí, lo que da un sistema de ecuaciones grande en comparación con la operatividad de la compañía.

El algoritmo generado de simulación del método de 'Aircraft Routing' permite de forma rápida y eficiente obtener no sólo aquellas combinaciones solución del problema, sino aquellas que permiten optimizar dicha solución según criterio de la compañía. Además, es fácilmente extrapolable a otras compañías diferentes, no sólo las de estudio, puediendo añadir variaciones en los vuelos para obtener posibles mejoras de horarios que permitan tener mayores ventanas de vuelo entre aeropuertos. Este método es ampliamente usado por las compañías aéreas pues gracias a este estudio se puede ahorrar costes tanto de combustible, optimizando las distancias, como para los pasajeros pues permite optimizar vuelos en el caso de que haya paradas intermedias, evitando que sea necesario cambiar de avión. El algoritmo generado, aunque se ha comparado con algoritmos existentes, se ha realizado desde cero, partiendo del concepto de 'Aircraft Routing', y realizando la formulación en base a los datos de partida y resultados objetivo.

El gran problema que abordan este tipo de algoritmos es la gran cantidad de información con la que deben operar, pues existen muchas combinaciones posibles, y la combinatoria entre vuelos es elevada. El principal objetivo de este procedimiento es reducir al máximo el tiempo de ejecución. Para ello, ha sido necesario partir de un código base que hiciese el objetivo marcado para el mismo, e ir puliendo partes del código, reduciendo al máximo las combinaciones, sin saltarse ninguna posible solución. El gran inconveniente estriba en que muchas combinaciones calculadas, aunque eran viables, no aportaban información a la hora de resolver el problema en conjunto. Por ejemplo, si se realizaban 40 vuelos y sólo había 4 aviones de flota, si la rotación más larga eran 12 vuelos, calcular rotaciones de menos de 8 vuelos es innecesario pues nunca serán solución del sistema. Aunque son rotaciones posibles, la compañía nunca las llevará a cabo con la flota actual pues no podrían satisfacer la demanda completa de vuelos operables.

No hay que olvidar que el algortimo consta de cinco partes esenciales diferenciadas unas de otras, que permiten compartimentar el problema y optimizar cada paso de forma independiente. Cada compañía de estudio posee un cuello de botella en el proceso. En el caso de Volotea, era la gran cantidad de vuelos, apartado 1 y 2 del proceso. Sin embargo, luego se obtenía que a pesar de ser una compañía de envergadura, no había tantas combinaciones como se esperaba pues los vuelos son de larga duración y la combinatoria es reducida, es decir, hay poco margen de maniobra en caso de que algún avión quede inoperativo. De ahí que se puedean realizar todos los vuelos con menos aviones de los que cuenta la flota, pues aparte de tener que realizar mantenimientos de larga duración, que reduce la flota real en algunos aviones, deben existir aviones en condiciones operativas en caso de que sea necesaria la sustitución de alguna aeronave.

Para Canaryfly, se vió como a pesar de no tener una gran flota, si había un número de vuelos diarios considerable. Esto se debía a la proximidad de operación. El principal inconveniente era no ya el calculo de las rotaciones, sino la combinación entre ellas, pues las posibilidades eran muchas y variadas. Había rotaciones que variaban en un vuelo exclusivamente, pudiendo obtener más de una solución óptima plausible. A pesar de ello, si se buscaba la solución óptima para varios objetivos, había una combinación que resaltaba por encima de todas.

Figura 45: Diagrama de flujo, etapas del algoritmo

En definitiva, el algoritmo permite visualizar todos y cada uno de los resultados al final de cada fase, así como consultar los tiempos entre despegues y aterrizajes para así poder comprobar como de limitante es una combinación. Es decir, cada una de las fases muestra resultados que pueden ser útiles para entender el proceso seguido así como las limitaciones que se pueden encontrar en fases futuras debido a resultados previos que impiden determinadas condiciones, es decir, puede ser que se quiera pasar noche en un determinado lugar optimizando el número de aviones, y no se obtengan soluciones para la flota real. Eso se puede ver ya en fases tempranas del proceso cuando no existen muchas combinaciones que hagan noche en dicho lugar.

Bien es cierto que se ha hecho un estudio para cuatro días de vuelos. Pero esto no es un inconveniente, pues el algoritmo es escalable. Se ha diseñado para poder paralelizar tareas. Cada fase descrita en la **Figura 41**, debe ser lineal con la anterior, es decir, debe haberse terminado el proceso precedente para empezar con la siguiente. Sin embargo, dentro de cada fase, suponiendo que es una caja negra, se puede dividir el estudio y calcular todo usando toda la potencia disponible, en este caso hasta 3 núcleos. Esto agiliza los resultados y podría aumentar la velocidad en caso de disponer de mayor capacidad. Esta paralelización se puede realizar gracias a que en el proceso 3 del diagrama de flujo, cuando se parte de los pares de vuelos concatenados, se comienza con un vuelo en concreto comparado con el resto y así para cada una de las combinaciones. Bien pues cada par de vuelos comparado original del procedimiento se puede paralelizar y obtener resultados de forma independiente.

Al final del proceso se deberán unir todos esos resultados en una única matriz con la que poder operar en apartados sucesivos. Lo que quiere decir que se puede ahorrar tiempo de procesado mediante estos procedimientos. En cualquier caso, para rotaciones de dos días, aunque mejora el rendimiento del algoritmo no es necesario pues el tiempo de ejecución no resulta problemático a la hora de sacar resultados.

Por último, recalcar que los programas realizados para llevar a cabo los algoritmos se han ido perfeccionando a lo largo del trabajo, mejorando su tiempo de ejecución y puliendo líneas de código innecesarias. Operar con Matlab matrices de n dimensiones tiene sus ventajas e inconvenientes. Matlab, al ser un programa de resolución numérica realiza cálculos muy rápido permitiendo así una ejecución veloz de los resultados. Sin embargo, los vectores de los que debe estar formada la matriz deben ser de igual tamaño o habrá problemas de guardado en las variables. Esto se podría evitar por medio del uso de 'arrays' pero el tiempo de realización de cálculos aumentaría pues serían tratados los números como caracteres. De ahí que el empleo de lenguajes de programación como Python puedan ser de ayuda para mejorar los tiempos de ejecución si el estudio se realiza para compañías de gran tamaño y periodos de tiempo elevado. Aún así, el uso eficiente que Matlab realiza de los cálculos numéricos hacen que sea un gran motor de procesado para tareas como ‘Aircraft Routing’ ya que con el simple hecho de asociar valores enteros a los aeropuertos, se puede obtener resultados óptimos evitando el uso de ‘strings’, que suelen ralentizar pues hay que comparar letra a letra si se trata del mismo aeropuerto o no.

En definitiva, el método de ‘Aircraft Routing’ permite ahorrar costes de cara a las aerolíneas por medio de la potencia de cálculo usando software específico para obtener las rotaciones más convenientes según los intereses. Además permite ver que nuevos vuelos pueden crearse viendo aquellas ventanas de despegue más libres y con mayor combinatoria. Ese estudio resulta beneficioso pues si se quieren abrir nuevas rutas, dichos vuelos deberán pertenencer a una rotación en concreto, y para ello saber que rotaciones permitirán con mayor facilidad incorporar dicho vuelo, o al menos, aportar mayor número de rotaciones que puedan realizar el vuelo en cuestión para así, a la hora de optimizar, tener un resultado deseable y no empeorar con respecto a las combinaciones posteriores a los nuevos vuelos.

Para poder comparar las ventajas del algoritmo de forma cuantitativa, frente a resolver el problema manualmente, se presenta un histograma de la empresa Volotea con todas las rotaciones y los valores de la función objetivo para todas ellas. Cuanto mayor es el valor de la función objetivo, mejor es el sistema. Si se realizase manualmente una rotación, podría obtenerse cualquiera de las que el algoritmo ha obtenido. La probabilidad de obtener una rotación que maximice el sistema (valor de la función objetivo de 2) es de 7.3%. Sin embargo, la probabilidad de obtener una rotación de valor negativo de función objetivo es de un 54.7%, más de la mitad de las veces. Lo que quiere decir que la probabilidad de encontrar una rotación ‘mala’ es 8 veces superior a encontrar una rotación óptima para los intereses de la compañía. Esto sin tener en cuenta el ahorro de tiempo pues el algoritmo minimiza el proceso operacional a pesar de calcular todas las opciones posibles. Hay que tener en cuenta que el número de aviones hace que la función tome valores negativos o nulos pues habrá que minimizarlo, mientras que los valores positivos son de la función objetivo de pasar noche en determinados aeropuertos, que habrá que maximizar.

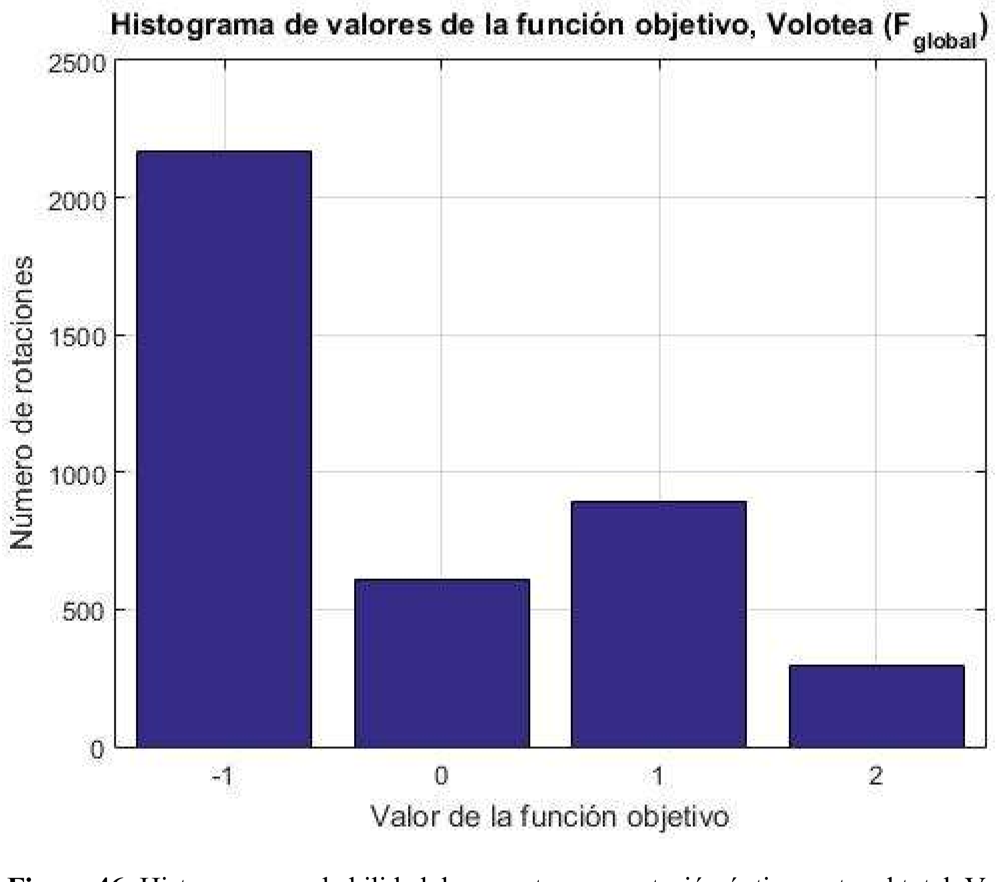


Figura 46: Histograma, probabilidad de encontrar una rotación óptima entre el total, Volotea

De esta forma concluimos que la ventaja operacional del algoritmo es elevada pues permite obtener siempre la opción correcta entre el total en un periodo corto de tiempo. Por ello, este tipo de problema se ha hecho tan popular en las compañías aéreas, intentando optimizar lo máximo posible para así reducir gastos.

Función objetivo 1	**Función Objetivo 2**	**Función Objetivo 1+2**	Volotea
19	24	20	**Número de noches en base**
20	32	24	**Número de aviones**

Para el caso de Canaryfly se llegó a la conclusión que había una solución que era óptima para todos los resultados, siendo esta la misma. Calculemos como de buena es esta solución frente al total. La figura 47 refleja la desviación frente a la media de horas de vuelo que debe realizar cada avión en el caso de que fuesen 4 aviones los utilizados.

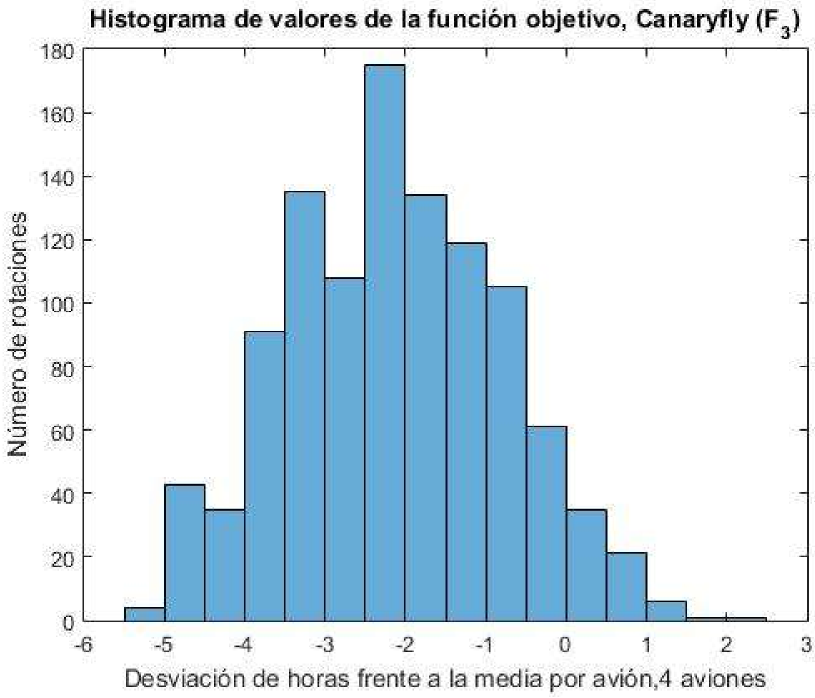


Figura 47: Histograma, probabilidad de encontrar una rotación óptima entre el total, Canaryfly,F_3

Función objetivo 1	**Función Objetivo 2**	**Función Objetivo 3**	**Función Objetivo 1+2+3**	Canaryfly
4	4	4	4	**Número de noches en base**
4	4	4	4	**Número de aviones**

3.75	3.75	3.75	3.75	**Desviación de horas media**

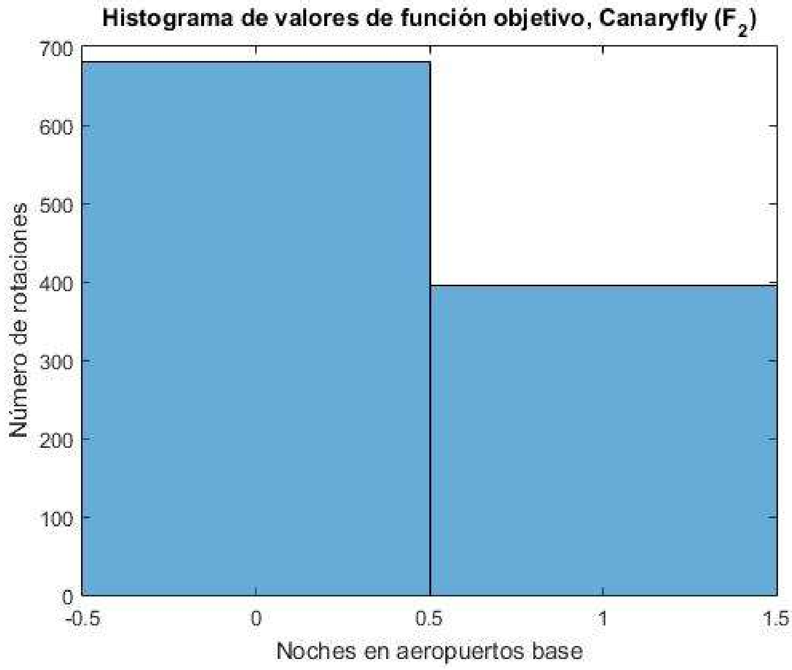


Figura 48: Histograma, probabilidad de encontrar una rotación óptima entre el total, Canaryfly, F_2

REFERENCIAS

[1] Canaryfly, «Página oficial del la compañía aérea 'Canaryfly',» 2018. [En línea]. Available: https://www.canaryfly.es/.

[2] C. Jeenanunta, B. Kasemsontitum y T. Noichawee, «A multi-commodity flow approach for aircraft routing and maintenance problem,» *2011 IEEE International Conference on Quality and Reliability,* 14-17 Sept. 2011.

[3] M. Sarhani, O. Ezzinbi, A. El Afia y Y. Benadada, «Particle swarm optimization with a mutation operator for solving the preventive aircraft maintenance routing problem,» *2016 3rd International Conference on Logistics Operations Management (GOL),* 23-25 May 2016.

[4] M. Ben Ahmed, F. Zeghal Mansour y M. Haouari, «A PSO Approach for Robust Aircraft Routing,» *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM),* 6-9 Dec. 2015.

[5] P. A. Vela, A. E. Vela y G. Ogunmakin, «Topologically based decision support tools for aircraft routing,» *29th Digital Avionics Systems Conference.*

[6] K. Roy y C. J. Tomlin, «Solving the aircraft routing problem using network flow algorithms,» *Proceedings of the 2007 American Control Conference,* New York City, USA, July 11-13, 2007.

[7] M. De-yi y Z. Zong-xian, «The integrated model of airline fleet assignment and aircraft routing based on flight cycle,» *2010 International Conference on Management Science & Engineering,* November 24-26, 2010.

[8] S. Baromand, M. Nekouie y K. Navaie, «Optimal Distributed Fuzzy Control Strategy for Aircraft Routing and,» *IV International Congress on Ultra Modern Telecommunications and Control Systems 2012,* 3-5 Oct. 2012.

[9] Volotea, «Página oficial de la compañía aérea 'Volotea',» 2018. [En línea]. Available: http://www.volotea.com/es/.

[10] M. Barzagan, Airline operations and scheduling, Ashgate, 2004.

[11] J. M. d. C. Granados y L. M. R. Pérez, Aplicación de la programación lineal al transporte aéreo, Complementos de Transporte aéreo, 1º de MUIA.

[12] Mathworks, «Página de soporte de productos de Matlab,» [En línea]. Available: https://es.mathworks.com/help/optim/ug/linprog.html.

[13] Mathworks, «Página de soporte de productos de Matlab,» [En línea]. Available: https://es.mathworks.com/help/optim/ug/intlinprog.html.

[14] Economista. [En línea]. Available: http://ranking-empresas.eleconomista.es/VOLOTEA.html.

[15] Economista. [En línea]. Available: http://ranking-empresas.eleconomista.es/CANARY-FLY.html.

ÍNDICE DE CONCEPTOS

A319: Avión comercial de pasajeros basado en el diseño del A320, de corto o medio alcance. Su fabricante es Airbus.

Asignación: Valor numérico asociado a un aeropuerto. No tiene por qué ser único, permite definir los nodos de forma unívoca.

ATR72-500: Avión comercial de pasajeros propulsado por dos motores turbohélices. Suele ser usado como avión de línea regional, la empresa fabricante es ATR.

B717: Avión comercial de pasajeros de corto y medio alcance. Su fabricante es Boing.

Ciclo de tiempo: Toda rotación debe tener un principio y un final. En este caso puede empezar y terminar el mismo día, donde el aeropuerto inicial y final de la rotación debe ser el mismo. Puede ocurrir que sea de dos días la duración de la rotación, y ocurriría lo mismo con el aeropuerto final e inicial. El *ciclo de tiempo* de la rotación puede ser tan amplio como la compañía operadora estime oportuno.

Combinación: Conjunto de rotaciones que permiten resolver un sistema con una función objetivo y unas restricciones concretas.

Flota: Conjunto de aviones de los que dispone una compañía aérea.

Función objetivo: Para optimizar un sistema habrá que declarar un objetivo prioritario que se pretenda cumplir en la realidad, y después expresarlo de forma matemática por medio de una función a maximizar o minimizar.

Nodo: Todo vuelo tiene un aeropuerto de salida y otro de llegada, cada aeropuerto es un nodo del vuelo. Existen dos nodos por vuelo.

Pesos: Cuando una función objetivo no es única, se debe ponderar según la importancia de esta con respecto al resto.

Restricciones: Condiciones que debe cumplir un sistema de ecuaciones para adecuarse a unas pautas dadas por el caso real. Pueden ser ecuaciones de igualdad o desigualdad, en el caso de que no se pueda superar un determinado valor.

Rotación: Combinación de vuelos realizados por una aeronave en concreto.

Ruta: Rotación.

SIMPLEX: Método de programación lineal que permite resolver problemas con dichas condiciones de manera rápida por medio de algoritmos hechos al efecto. En este trabajo se ha usado para la resolución de problemas de programación lineal binaria.

String: Conjunto de caracteres que forman una palabra. En computación, para reconocer la palabra hay que guardarlo letra a letra, mientras que los números son elementos únicos con los que es más rápido operar.

Tiempo de espera: Todo avión debe permanecer en tierra un tiempo mínimo antes de despegar, habrá que bajar a los pasajeros, el equipaje, realizar repostaje y tareas de puesta a punto. Además, habrá que esperar que la pista este disponible según slots. No todos los vuelos tienen los mismos tiempos de espera, pero si se puede suponer un tiempo mínimo general por compañía.

Anexos

LL	1	2	3	4	5	6
1	0	17:50	0	0:00	0	0
2	16:40	0	12:30	8:30	14:50	17:10
3	0	13:50	0	12:30	0	0
4	0:00	9:50	11:10	0	0	8:10
5	0	9:20	0	0	0	0
6	0	12:35	0	9:25	0	0
7	0:00	0	19:10	14:25	19:45	11:10
8	0	0	0	19:50	10:15	0
9	0	20:30	0	17:55	0	0
10	0	0	0	22:25	0	
11	0	15:30	16:55	0	0	13:40
12	0	18:40	21:20	0	0	19:20
13	0	0	0	0	0	21:15
14	0	11:10	0	0	0	0
15	0	15:50	0	0	0	0
16	0	20:45	0	0	0	0
17	0	18:25	0	14:50	0	0
18	0	0	0	20:30	0	0
19	0	0	0	22:55	0	0
20	0	0	0	0	8:20	0

Figura 49: Horarios de llegada, Canaryfly

S	1	2	3	4	5	6
1	0	17:10	0	0:00	0	0
2	16:00	0	11:40	8:00	14:20	16:15
3	0	12:55	0	11:50	0	0
4	0:00	9:20	10:25	0	0	7:25
5	0	8:50	0	0	0	0
6	0	11:40	0	8:40	0	0
7	0:00	0	18:20	13:55	19:20	10:20
8	0	0	0	19:20	9:45	0
9	0	19:40		17:20	0	0
10	0	0	0	21:45	0	0
11	0	15:00	16:15	0	0	12:55
12	0	18:10	20:40	0	0	18:40
13	0	0	0	0	0	20:55
14	0	10:40	0	0	0	0
15	0	15:20	0	0	0	0
16	0	20:15	0	0	0	0
17	0	17:35	0	14:05	0	0
18	0	0	0	19:50	0	0
19	0	0	0	22:15	0	0
20	0	0	0	0	7:50	0

Figura 50: horarios de salida, Canaryfly

Para el caso de Volotea, al ser matrices de gran tamaño se indican los valores de llegadas y salidas para los cuatro días en tablas.

Salida	**Llegada**	**Horario de salida**	**Horario de llegada**
<u>Día 13</u>			
Sevilla	Asturias	20:40	22:10
Asturias	Sevilla	15:05	16:35
Santander	Sevilla	18:50	20:15
Sevilla	Santander	17:00	18:25
Asturias	Valencia	8:00	9:25
Valencia	Asturias	9:50	11:20
Madrid	Burdeos	14:45	16:10
Ancona	Catania	11:10	12:35
Bari	Catania	9:20	10:35
Bari	Venecia	9:20	10:45
Bari	Verona	12:30	14:00
Bari	Palermo	13:05	14:20
Catania	Ancona	9:10	10:45
Catania	Bari	11:25	12:40
Catania	Genova	16:40	18:35
Catania	Nápoles	18:15	19:30
Catania	Venecia	13:00	14:50
Catania	Verona	9:20	11:10
Catania	Verona	20:55	22:45
Catania	Palermo	13:00	14:50
Genova	Catania	14:35	16:15
Genova	Nápoles	7:00	8:20
Génova	Nápoles	19:00	20:20
Génova	Palermo	12:45	14:15

Salida	Llegada	Horario de salida	Horario de llegada
Nápoles	Catania	16:45	17:50
Nápoles	Génova	12:30	13:50
Nápoles	Génova	20:45	22:05
Nápoles	Triestre	17:05	18:25
Nápoles	Verona	9:00	10:20
Nápoles	Verona	19:45	21:05
Nápoles	Palermo	9:25	10:25
Nápoles	Palermo	19:55	20:55
Viena	Nantes	19:50	22:30
Praga	Marsella	10:15	12:20
Praga	Toulouse	13:45	16:00
Praga	Burdeos	15:10	17:30
Praga	Nantes	19:00	21:15
Brest	Montpellier	13:55	15:30
Brest	Toulouse	18:00	19:30
Estrasburgo	Toulouse	8:55	10:25
Lille	Montpellier	11:00	12:35
Montpellier	Brest	15:55	17:35
Montpellier	Lille	9:00	10:35
Montpellier	Nantes	13:00	14:25
Perpiñan	Nantes	12:20	13:40
Triestre	Nápoles	10:45	12:05
Venecia	Bari	7:15	8:35
Venecia	Catania	7:00	8:45
Verona	Bari	10:45	12:05
Verona	Catania	18:50	20:30
Verona	Catania	7:15	8:55

Salida	Llegada	Horario de salida	Horario de llegada
Verona	Nápoles	7:00	8:20
Verona	Palermo	14:00	19:20
Burdeos	Madrid	12:55	14:20
Nantes	Viena	17:00	19:05
Nantes	Praga	11:15	13:20
Nantes	Montpellier	7:15	8:35
Nantes	Perpiñan	10:35	11:55
Palermo	Bari	7:30	8:40
Palermo	Génova	10:50	12:20
Palermo	Nápoles	8:00	9:00
Palermo	Nápoles	15:20	16:20
Palermo	Verona	16:00	17:35
Burdeos	Praga		9:50
Día 14			
Sevilla	Bilbao	17:00	18:25
Asturias	Alicante	17:00	18:35
Alicante	Asturias	19:00	20:40
Bilbao	Sevilla	15:10	16:35
Bilbao	Venecia	18:50	21:00
Cagliari	Génova	16:40	18:00
Cagliari	Turín	12:55	14:25
Catania	Venecia	18:10	20:00
Catania	Verona	9:20	11:10
Catania	Verona	20:45	22:45
Nápoles	Génova	20:45	22:05
Génova	Cagliari	11:15	12:30

Salida	Llegada	Horario de salida	Horario de llegada
Génova	Nápoles	19:00	20:20
Nápoles	Verona	17:05	18:25
Nápoles	Palermo	9:25	10:25
Nápoles	Palermo	19:55	20:55
Turín	Cagliari	14:50	16:15
Venecia	Bilbao	12:20	14:45
Venecia	Catania	16:00	17:45
Verona	Catania	7:15	8:55
Verona	Catania	18:50	20:30
Verona	Nápoles	15:20	16:40
Palermo	Nápoles	8:00	9:00
Palermo	Nápoles	18:30	19:30
Día 15			
Ancona	Catania	14:10	15:35
Bari	Catania	9:20	10:35
Bari	Verona	17:00	18:25
Bari	Palermo	13:05	14:20
Catania	Ancona	12:10	13:45
Catania	Bari	11:25	12:40
Catania	Génova	16:00	17:55
Catania	Venecia	18:10	20:00
Catania	Verona	9:20	11:10
Catania	Verona	20:55	22:45
Génova	Catania	10:00	11:40
Nápoles	Verona	19:45	21:05
Génova	Nápoles	19:00	20:20

Salida	Llegada	Horario de salida	Horario de llegada
Nápoles	Génova	20:45	22:05
Nápoles	Palermo	9:25	10:25
Nápoles	Palermo	19:55	20:55
Estrasburgo	Toulouse	17:30	19:00
Estrasburgo	Burdeos	19:25	20:55
Montpellier	Nantes	18:50	20:15
Toulouse	Estrasburgo	19:25	20:55
Venecia	Catania	16:00	17:45
Verona	Bari	15:00	16:20
Verona	Catania	7:15	8:55
Verona	Catania	18:50	20:30
Verona	Nápoles	18:00	19:20
Burdeos	Estrasburgo	17:10	18:45
Nantes	Montpellier	17:05	18:25
Palermo	Bari	7:30	8:40
Palermo	Nápoles	8:00	9:00
Palermo	Nápoles	18:30	19:30
Día 16			
Sevilla	Asturias	20:40	22:10
Sevilla	Santander	17:00	18:25
Asturias	Sevilla	15:05	16:35
Asturias	Málaga	10:30	12:05
Santander	Sevilla	18:50	20:15
Málaga	Asturias	12:30	14:15
Catania	Nápoles	18:15	19:30
Bari	Venecia	17:00	18:20
Bari	Verona	17:00	18:25

Salida	**Llegada**	**Horario de salida**	**Horario de llegada**
Catania	Génova	16:40	18:35
Catania	Venecia	18:10	20:00
Catania	Verona	9:20	11:10
Catania	Verona	20:55	22:45
Génova	Catania	14:35	16:15
Génova	Nápoles	7:00	8:20
Génova	Nápoles	19:00	20:20
Génova	Palermo	15:05	16:35
Nápoles	Catania	16:45	17:50
Nápoles	Génova	9:00	10:20
Nápoles	Génova	20:45	22:05
Nápoles	Verona	19:45	21:05
Nápoles	Palermo	9:25	10:25
Nápoles	Palermo	19:55	20:55
Caen	Toulouse	17:00	18:35
Estrasburgo	Toulouse	17:30	19:00
Estrasburgo	Burdeos	19:25	20:55
Montpellier	Nantes	18:50	20:15
Toulouse	Caen	15:10	16:35
Toulouse	Estrasburgo	19:25	20:55
Venecia	Bari	15:00	16:20
Venecia	Catania	16:00	17:45
Venecia	Nantes	14:25	16:40
Verona	Palermo	14:00	15:35
Verona	Bari	15:00	16:20
Verona	Catania	7:15	8:55
Verona	Catania	18:50	20:30

Salida	**Llegada**	**Horario de salida**	**Horario de llegada**
Verona	Nápoles	18:00	19:20
Burdeos	Estrasburgo	17:10	18:45
Nantes	Montpellier	17:05	18:25
Nantes	Venecia	12:00	14:00
Palermo	Génova	13:10	14:40
Palermo	Nápoles	8:00	9:00
Palermo	Nápoles	15:20	16:20
Palermo	Verona	16:00	17:35