

Inteligencia Artificial

Estado del Arte: Problema *Aircraft Landing Scheduling*

Victor Gonzalez Rodriguez
victor.gonzalezro@alumnos.usm.cl

13 de mayo de 2013

Evaluación

Resumen (5 %):	_____
Introducción (5 %):	_____
Definición del Problema (10 %):	_____
Estado del Arte (35 %):	_____
Modelo Matemático (20 %):	_____
Conclusiones (20 %):	_____
Bibliografía (5 %):	_____
Nota Final (100 %):	_____

Resumen

En este informe, consideramos el problema de Aircraft Landing Scheduling, donde se modela el aterrizaje de aviones en distintas pistas de un aeropuerto. Este problema nos ayuda a decidir el tiempo y lugar donde un avión debería aterrizar dentro de una ventana predeterminada de tiempo, respetando la separación de tiempo entre aterrizajes y la secuencia respectiva de aviones que deben aterrizar después. Analizaremos el estado del arte de los métodos actualmente publicados, de manera que se genere una idea general de las implementaciones disponibles para resolver este problema, y analizar así cuales son los mejores acercamientos y sus evoluciones. Finalmente, esto se presentará mediante un modelo matemático, formulando sus respectivas restricciones y variables, para luego comentar y concluir el estado del arte al presente año.

Keywords: Aircraft Landing Scheduling problem, estado del arte, modelo matemático, calendarización de aterrizajes, Aircraft Arrival and Departure Sequencing problem.

1. Introducción

El Aircraft Landing Scheduling Problem (ALSP), o *Problema de Calendarización de Aterrizaje de Aeronaves* en español, es un problema que ha ganado mucha importancia a nivel mundial, especialmente los últimos años debido a la gran congestión de aviones que están circulando actualmente en el espacio aéreo mundial, y la necesidad de mantener el flujo de aeronaves lo más fluido posible sin perder de vista la seguridad de los pasajeros que son parte esencial de todo este sistema. Ejemplo de esto es la investigación impulsada gubernamentalmente en el aeropuerto de Heathrow hace un par de años atrás [2].

El ALSP es catalogado como un problema muy complejo, el cual en la clasificación estándar, se lo cataloga como NP-complejo¹, y debido a esto mismo, las implementaciones disponibles en las publicaciones a nivel mundial, pueden llegar a ser muy complejas y difíciles de entender. Debido a esta situación, hemos de entregar una definición del problema que sólo involucre los factores más comunes y/o significativos para modelar el ALSP, y a partir de esto, se presentará un modelo matemático tan sencillo como la definición del problema lo permita.

Estudiaremos además, las distintas implementaciones disponibles actualmente de manera superficial, de modo que se pueda comparar a grandes rasgos cual es la tendencia mundial o histórica para resolver este tipo de problemas, situando al lector en la situación actual de la modelación de estos problemas.

A partir de lo presentado previamente en este documento, se comentará, justificará y comentará lo expuesto en la sección llamada *Estado del Arte y Modelo Matemático*.

Finalmente, es de esperar que la información entregada en este documento, sea de ayuda para tener una buena noción de lo que está ocurriendo actualmente respecto al ALSP, y cuales serían los mejores acercamientos para resolver este tipo de problemas en la actualidad.

2. Definición del Problema

Debido a la creciente demanda de vuelos al rededor del mundo durante las últimas décadas, junto con el impulso de la tecnología y el explosivo aumento de pasajeros, ha puesto a los aeropuertos manejando cientos viajes en un día. Todo esto sin dejar de lado la seguridad, los tiempos de despegue/aterrizaje y la optimización del uso de los recursos de infraestructura.

Para darnos una idea gráfica del tráfico actual en el mundo, disponemos de la siguiente captura del día viernes 10 de mayo de 2013 a las 19:00 horas aproximadamente.

El ALSP modela la situación de aeropuertos, aeródromos, bases militares y portaaviones, los cuales desde este momento los llamaremos simplemente como *aeropuertos*, donde se quiere aterrizar, los cuales se llamarán a partir de ahora *aterrizajes*, englobando también a los despegues, ya que en la práctica, no existe mucha diferencia entre ambos; despegar o aterrizar es simplemente hacer uso de una de las pistas del aeropuerto en una ventana de tiempo determinado.

¹Tan complejo como el más difícil de los problemas polinomiales no-deterministas.

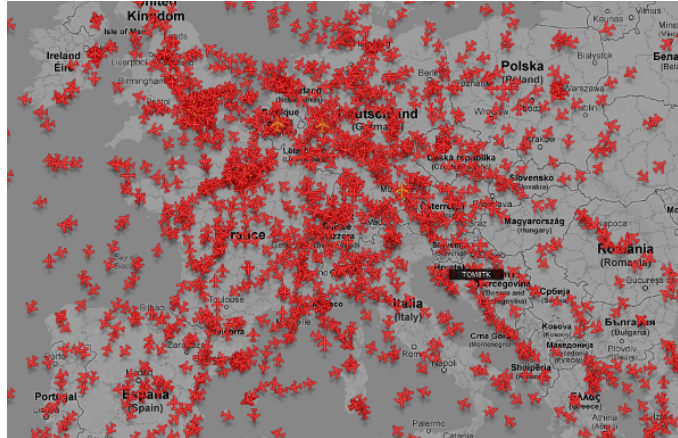


Figura 1: fuente: <http://www.planefinder.net/>

En la instancia que analizaremos de este problema, abordaremos la situación de un aeropuerto con n pistas y con un listado estático de aterrizajes, donde se sabe explícitamente y sin cambios, cuales serán las franjas de tiempo donde deben aterrizar los aviones entre sus muchas otras variables necesarias para describir el aterrizaje, el cual es la entrada del modelo matemático. El problema por el cual surge este problema, es que los vuelos están sujetos a retrasos, ya sea debido a variables atmosféricas, fallas técnicas de los aviones, daños durante el vuelo, etc. Todos estos problemas son propensos a retrasar el aterrizaje de un avión, afectando además a las demás naves. Es por esto que es necesario calendarizar de manera óptima los vuelos, de modo que se disponga de los tiempos óptimos para el aterrizaje considerando todos los eventos externos que pueden retrasar un vuelo, sin afectar el posible alto flujo de aterrizajes que se deben producir.

Para entrar en detalle, definiremos los aspectos más relevantes que pueden complicar la calendarización de los aterrizajes, los cuales obligarán al orden tomar en cuenta no solo el tiempo de llegada y las pistas disponibles:

- **Carga de la nave:** Aviones con muchos pasajeros, llevan mucho más equipaje, por lo que el tiempo de carga/descarga y los chequeos de seguridad afectan directamente al personal del aeropuerto, el cual es limitado y debe tener bajos tiempos de respuesta para efectuar los trabajos respectivos.
- **Velocidad de vuelo:** Durante el vuelo, los aviones tienen una velocidad crucero, la cual permite estimar con antelación sus aterrizajes. Si se altera esa velocidad crucero, implica un costo tanto por el uso de combustible, como para definir un aterrizaje.
- **Combustible:** La cantidad de combustible define el tiempo máximo de vuelo de un avión, lo cual es relevante para definir la ventana de tiempo de espera, antes que un avión aterrice mientras espera un cupo.
- **Turbulencia de despegue:** Los aviones más grandes, ya sean comerciales o militares, generan mucha turbulencia al momento de despegue o aterrizaje, por

lo que se debe disponer de una ventana de tiempo entre aterrizajes para estos aviones, de manera que no afecten la aerodinámica de los que vienen después.

3. Estado del Arte

3.1. Métodos Actuales

3.1.1. Algoritmos Genéticos

Este tipo de algoritmos por un buen tiempo han estado en la punta de lanza para resolver problemas de optimización o calendarización. Por lo general, estos algoritmos son los más usados cuando se trata de resultados liberados en tiempo real, ya que tienen un muy baja incertidumbre respecto al largo de las operaciones [1].

Los algoritmos genéticos se pueden resumir mediante el siguiente pseudo-código:

```
Inicializar población inicial
Repetir hasta terminar:
    Evaluar conveniencia (ej: costos, tiempo de viaje)
    Reducir la población
    Seleccionar pares entre las mejores opciones (fitness)
    Repoblar usando los pares
        Aplicar operador de cruce
        Aplicar operador de mutación
    Verificar criterio de término
        Limites de tiempo, minima conveniencia satisfecha, etc.
    Iterar si no se ha terminado
```

De acuerdo a esto, ajustando el ALSP a este tipo de algoritmos nos damos cuenta, que contamos con una población de calendarizaciones potenciales y está siempre disponible, y mientras más se ejecute el algoritmo, aparecerán mejores soluciones, esto nos dice que en cualquier momento encontraremos nuestro óptimo.

Sin embargo, existe una variante de este algoritmo que interesantemente se asemeja a lo que nosotros necesitamos, la cual se llama *seed modifier*, el cual hace que en cada mutación del algoritmo, no se parta sin aviones aterrizados (sin semillas), lo cual se puede lograr del siguiente modo:

```
Población inicial: población restante del último problema
Repetir hasta terminar:
    Evaluar conveniencia (ej: costos, tiempo de viaje)
    Reducir la población: eliminar aviones que aterrizaron
    Seleccionar pares entre las mejores opciones
    Repoblar usando los pares: aviones que llegaron después
        Aplicar operador de cruce
        Aplicar operador de mutación
    Verificar criterio de término
        Limites de tiempo, minima conveniencia satisfecha, etc.
    Iterar si no se ha terminado
```

Haciendo la analogía entre este algoritmo y ALSP, los genes representan el tiempo de aterrizaje y la pista de aterrizaje que les corresponde, donde se tiene una lista estática que puede ser consultada en cualquier momento para ejecutar las funciones de fitness o para repoblar el algoritmo, de manera que este algoritmo puede consultar por el tiempo de vuelo máximo de los aviones, la carga, cantidad de pasajeros, etc.

3.1.2. Algoritmo evolutivo diferencial de búsqueda de vecinos

Los algoritmos evolutivos tienen la característica de cumplir bien bajo la demanda de problemas de optimización de múltiples objetivos. Esto resulta especialmente útil cuando consideramos que por lo general, las restricciones impuestas por los modelos tienden a tener conflictos entre sí.

En este caso, el algoritmo busca optimizar 2 objetivos:

1. Costo asociado al aterrizaje.
2. Tiempo estimado de llegada (ETA: Estimated Time Arrival).

Primero se define una secuencia de aterrizaje utilizando algún método de ordenamiento como PEPS², o técnicas más complejas como NAR³. Luego de esto, se aplica el algoritmo evolutivo multiobjetivo llamado MONSDE⁴ [7], el cual es el encargado de determinar finalmente la secuencia de aterrizaje para todos los aviones.

3.1.3. Algoritmo evolutivo de manejo de restricciones

Este algoritmo lidia de mejor manera respecto a las restricciones que nos entrega el ALSP. Tiene un parecido significativo respecto al Algoritmo Genético, pero su función es resolver los conflictos de las restricciones respecto a la función objetivo del problema. Su implementación es algo similar a esto:

Generación de Población Inicial

Repetir hasta terminar:

Evaluar funcion objetivo

Generar nueva población:

Mutación/recombinación/selección

Evaluar término

Criterio de Tiempo, costo, etc. alcanzado

Iterar si no ha terminado

Este algoritmo puede encontrar una región relativamente desable dentro del espacio de búsqueda, de manera que entregue una calendarización para el problema. Todo esto depende de que tan bien se defina el manejo de las restricciones, ya que estas son el cuello de botella del algoritmo evolutivo. [8]

²Primero en entrar, primero en salir, o FCFS (First Come, First Served) por sus siglas en inglés.

³Non-dominated Average Ranking, el cual consiste en un algoritmo indirecto y eficiente por lo rápido, y por lo conveniente de su sistema de ranking que provee.

⁴Muti-Objective Neighborhood Search Differential Evolution

3.1.4. Método híbrido de resolución basado en formulación de Job Shop

Se plantea una formulación del problema de aircraft landing scheduling estático con varias pistas de aterrizaje que contiene menor número de restricciones, lo que puede reducir el tiempo de computo), y luego se modela el mismo problema como si fuera un job shop scheduling, para mostrar la relación que existe entre ambas formulaciones.

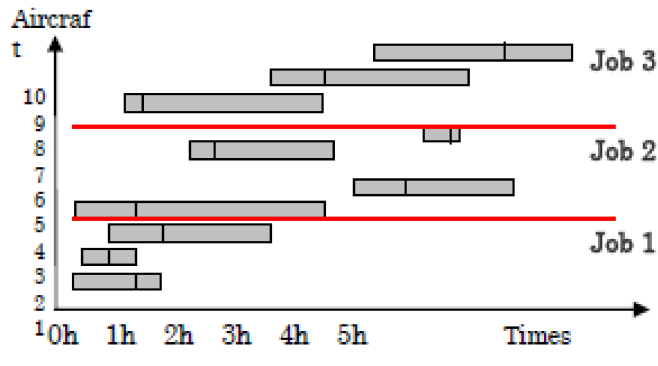


Figura 2: Los aviones se dividen en subconjuntos, llamados Jobs

El gran beneficio de este método, es que tiene mucha investigación de respaldo, por lo que sirve para entregar información valiosa a nuestro ALSP.

3.2. Tendencia Actual

Para el ALSP no basta con minimizar los costos, se deben tomar en cuenta todos los factores que pueden influir en el retraso o adelanto de los distintos aterrizajes. Debido a esto los métodos que entregan solo una solución (monosolución), no sirven, ya que se deben considerar múltiples alternativas; es aquí donde destacan los algoritmos que utilizan el análisis multiobjetivo gracias a la calidad de sus soluciones.

Pero eso no es todo. Debido a que los algoritmos evolutivos y genéticos tienden a tener tiempos de cómputo más elevados, no siempre resultan como mejor alternativa, dado la necesidad de saber los datos de manera más rápida posible.

Debido a esto, se puede considerar cualquier algoritmo, ya que no existe un algoritmo *mágico*, que sea la mejor opción. Es probable que un algoritmo que sea óptimo en un aeropuerto, no lo sea en otro, debido a las diferencias de tráfico, cantidad de pistas, etc.

4. Modelo Matemático

En la sección previa se han expuesto modelos capaces de resolver problemas de ALSP estático e incluso dinámico. Pero para nuestro caso consideraremos solo los ALSP estáticos para n pistas.

Para esto, utilizaremos el método del Job Shop⁵. [3]

⁵JSSP: Job Shop Scheduling Problem

4.1. Variables

N: número de aviones esperando aterrizar.

R: número de pistas de aterrizaje disponibles.

e_i : el tiempo más temprano de aterrizaje del avión i.

l_i : el tiempo más tarde de aterrizaje del avión i.

ta_i : el tiempo ideal de aterrizaje del avión i.

Pb_i : costo en una unidad de tiempo para avión i adelantado.

Pa_i : costo en una unidad de tiempo para avión i atrasado.

S_{ij} : ventana de tiempo entre avión i y j, si i aterriza antes de j en la misma pista.

s_{ij} : ventana de tiempo entre avión i y j, si i aterriza antes de j en distintas pistas.

4.2. Variables de decisión

t_i : el tiempo calendarizado para el avión i.

er_i : tiempo de adelanto de avión i.

x_{ij} : 1 si avión i aterriza antes de j, -1 al revés.

y_{ir} : 1 si avión i aterriza en pista r, 0 si no aterriza en pista r.

z_{ij} : Si avión i y j aterrizan en la misma pista, 0 si no lo hacen.

4.3. Restricciones

- $e_i \leq t_i \leq l_i$ el tiempo calendarizado debe estar en la ventana de aterrizaje $[e_i, l_i]$
- $X_{ij} + X_{ji} = 0$ i y j no pueden aterrizar juntos
- $x_{ij} * t_j \geq x_{ij} t_{ij} + S_{ij} * z_{ij} + s_{ij}(1 - z_{ij})$ se debe respetar la separación entre las variables
- $\sum_{r=1}^R y_{ir} = 1$ un avión solo puede aterrizar en una pista

4.4. Función Objetivo

Minimizar el costo de las desviaciones del tiempo real de aterrizaje y el tiempo estimado de aterrizaje.

$$Min(\sum_{i=1}^N Pb_i + tr_i * Pa_i)$$

5. Conclusiones

Dada la gran cantidad de implementación y variaciones de los mismos métodos, existe una amplia gama de alternativas para resolver ALSP, pero por lo general estas son costosas computacionalmente en distintos ámbitos, ya que algoritmos evolutivos entregan soluciones muy interesantes comparadas con las otras, pero recurren en un alto gasto de recursos como la memoria y tiempo de cómputo, y por otro lado, algoritmos que entregan excelentes soluciones óptimas tienen tiempos aún más malos de cómputo. Sin embargo, cada implementación debería ser específica para cada caso, ya que imposible saber a priori como se comportará un algoritmo con ciertos parámetros, por lo que no existe *una gallina de los huevos de oro* que permita el mejor equilibrio.

Dado la corta vida que tiene este tipo de problemas, es importante pensar que la paralelización de estos algoritmos sería una buena opción para acelerar los tiempos de cómputo, en especial en lo de tipo genético, donde se necesita evaluar y comparar distintas opciones no necesariamente en tiempos sincrónicos.

6. Bibliografía

A continuación la bibliografía utilizada para redactar este documento.

1. An Anytime Algorithm for Scheduling of Aircraft Landing Times Using Genetic Algorithms, Vic Ciesielski, Paul Scerri. Agosto 1998.
2. Scheduling aircraft landings at London Heathrow using a population heuristic, J. E. Beasley, J. Sonander, P. Havelock. Julio 2000.
3. Hybrid method for aircraft landing scheduling based on a Job Shop formulation, G. Bencheikh, J. Boukachour, A. El Hilali Alaoui , and F. El Khoukhi, Agosto 2009
4. Departure Runway Scheduling at London Heathrow Airport Extended Abstract, Jason A. D. Atkin, Edmund K. Burke, John S. Greenwood, Dale Reeson. Octubre 2004..
5. Algorithms of Scheduling Aircraft Landing Problem, Min Wen, Noviembre 2005.
6. Decentralized aircraft landing scheduling at single runway non-controlled airports, Yuanyuan Ding. Mayo 2007.
7. A Multi-objective Evolutionary Approach to Aircraft Landing Scheduling Problems, Ke Tang, Zai Wang, Xianbin Cao, Jun Zhang. 2008
8. Constraint handling based multiobjective evolutionary algorithm for aircraft landing scheduling, Yuanping Guo, Xianbin Cao, Jun Zhang. Junio 2008.
9. Scheduling Aircraft Landings to Balance Workload of Ground Sta, Nils Boysen, Malte Fliedner. Agosto 2008.