

Quantum Annealing for Air Traffic Management

Tobias Stollenwerk¹

Collaborators²

Bryan O'Gorman, Salvatore Mandrà, Davide Venturelli, Eleanor G. Riffel

Domain Experts³

Olga Rodionova, Hok K. Ng, Banavar Sridhar

¹German Aerospace Center (DLR)

²NASA QuAIL

³NASA Aviation Systems Division



Knowledge for Tomorrow



Content

- Optimizing flight trajectories
- Classical preprocessing
- Mapping to Quadratic Unconstrained Binary Optimization (QUBO)
- Quantum annealing experiments



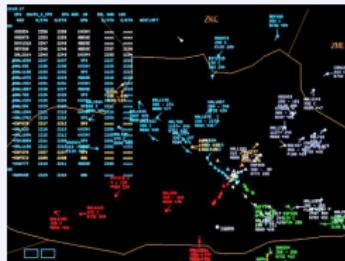
Motivation

Today



- Manual conflict avoidance
- Computer support for humans

Future

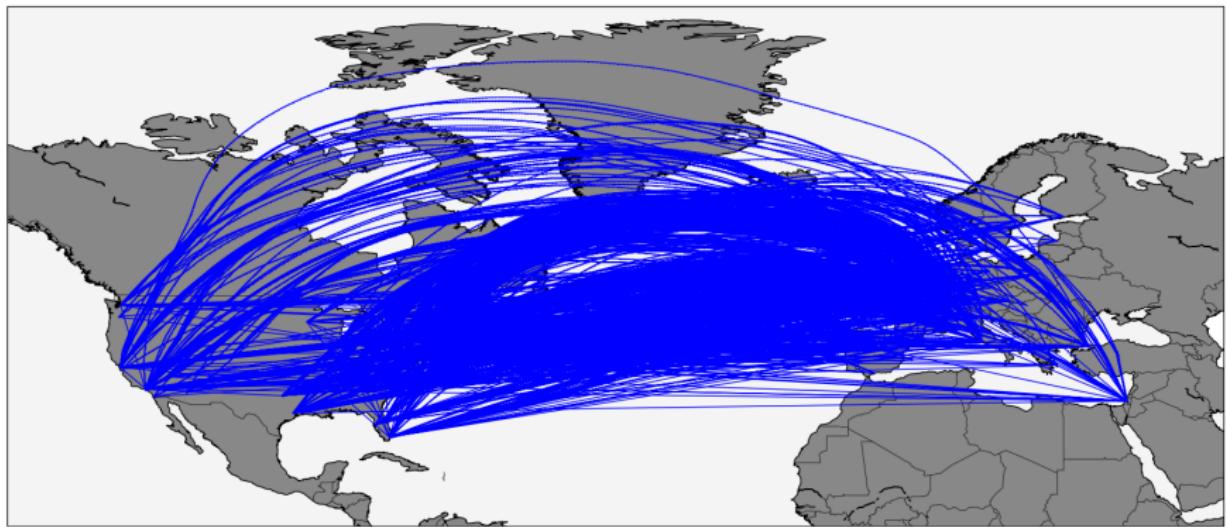


- Increase in computer support for humans
- Precomputation of trajectories: Scheduling



Wind-Optimal Trajectories

- 984 transatlantic flights on a single day

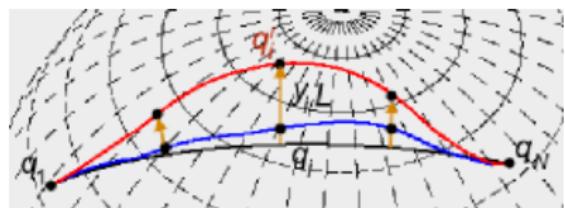


Data from O. Rodionova

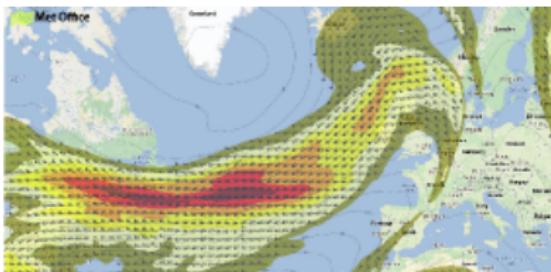


Classical Optimization Based on Wind-Optimal Trajectories

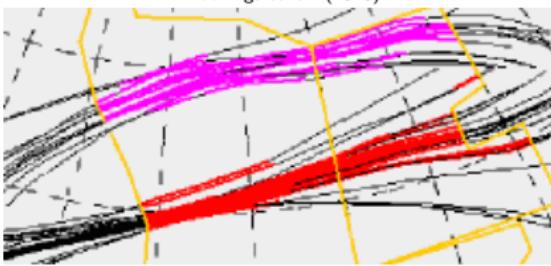
- Jetstream lead to narrow *highways*
- Clusters of conflicting flights
- Classical algorithm:
Continuous deviation of trajectories



Rodionova et. al. (2016)



Woollings et. al. (2010)



Rodionova et. al. (2016)

Optimization Problem Formulation

Variables

- Departure delays d_i for each flight i



- Maneuver of flight i to avoid conflict k introduce delay d_{ik}



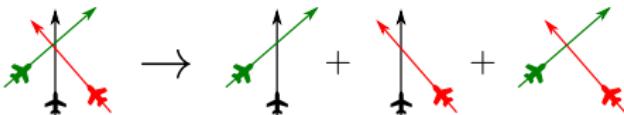
Cost function contribution

$$\text{Total delay: } C = \sum_i d_i + \sum_{ik} d_{ik}$$

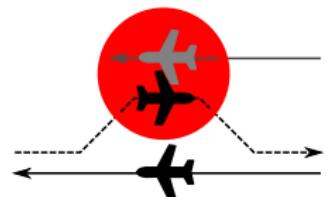


Optimization Problem Formulation - Simplifications

- Only pairwise conflicts

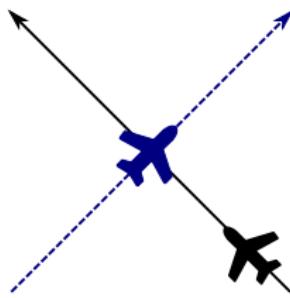


- Conflict avoiding maneuvers impact **only** on delay.

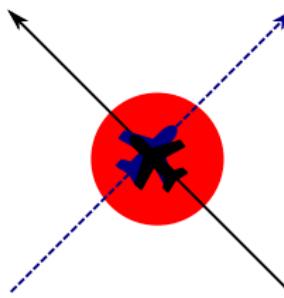


Conflict Avoidance - Arrival Times

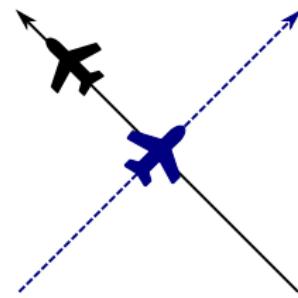
- Difference of arrival times at the conflict between flights i and j ,
 $\Delta_k = T_{ik} - T_{jk}$



$$\Delta_k \ll 0$$



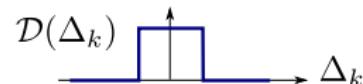
$$\Delta_k \approx 0$$



$$\Delta_k \gg 0$$

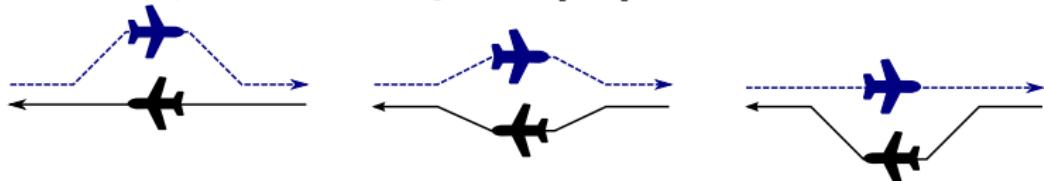
- Delay resulting from conflict avoidance is function of $\Delta_k = T_{ik} - T_{jk}$:

$$d_{ik} = \mathcal{D}_{ik}(\Delta_k)$$



Conflict Avoidance - Maneuver Parameter

- Maneuver parameter a_k , e.g. $a_k \in [0, 1]$



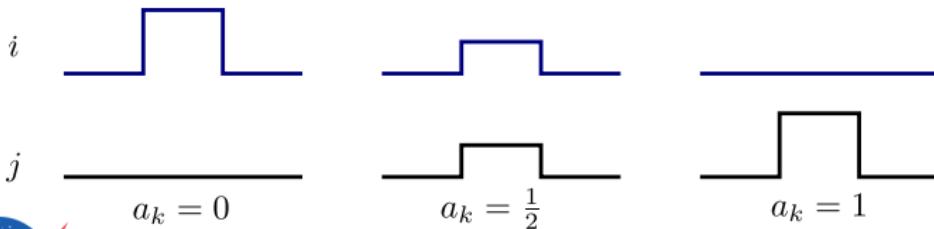
$$a_k = 0$$

$$a_k = \frac{1}{2}$$

$$a_k = 1$$

- Delay resulting from conflict avoidance depends on maneuver:

$$d_{ik} = \mathcal{D}_{ik}(\Delta_k, a_k)$$



Optimization Problem Formulation

- Arrival time of flight i at conflict k is delayed by preceding conflicts

$$T_{ik} = t_{ik} + d_i + \sum_{p < k} d_{ip} \quad t_{ik}: \text{Wind-optimal arrival time}$$

- Optimization problem

$$\underset{d_i, d_{ik}, a_k}{\text{minimize}} \quad \sum_i d_i + \sum_{ik} \mathcal{D}_{ik}(\Delta_k, a_k)$$

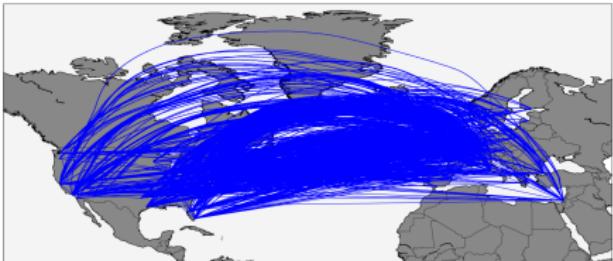
subject to

$$\begin{aligned} \Delta_k &= t_{ik} + d_i + \sum_{p < k} d_{ip} - t_{jk} - d_j - \sum_{q < k} d_{jq} \\ d_{ik} &= \mathcal{D}_{ik}(\Delta_k, a_k) \end{aligned}$$



Precalculating Conflicts

Given the trajectories of all flights i



⇒ How to calculate the *potential* conflicts k ?

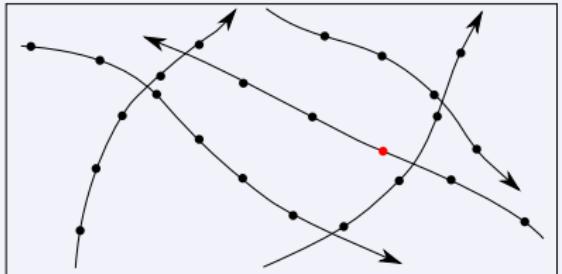


Spatial Conflict Detection

- Spatial conflict, if trajectory points are close (3 nautical miles) to each other.

Brute force algorithm

- Check distance between nearly all trajectory points

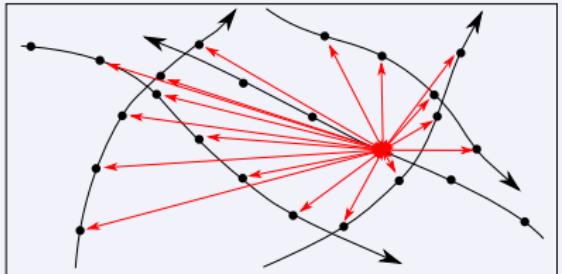


Spatial Conflict Detection

- Spatial conflict, if trajectory points are close (3 nautical miles) to each other.

Brute force algorithm

- Check distance between nearly all trajectory points

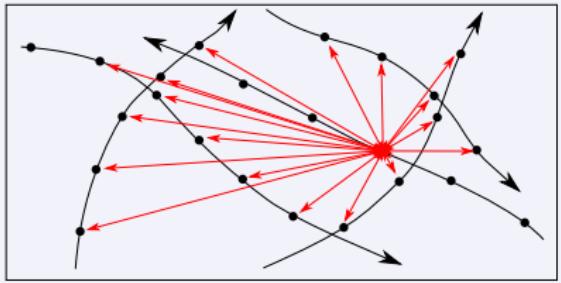


Spatial Conflict Detection

- Spatial conflict, if trajectory points are close (3 nautical miles) to each other.

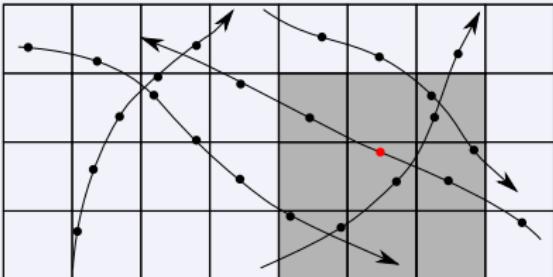
Brute force algorithm

- Check distance between nearly all trajectory points



Coarse grid algorithm

- Map trajectory points to coarse grid

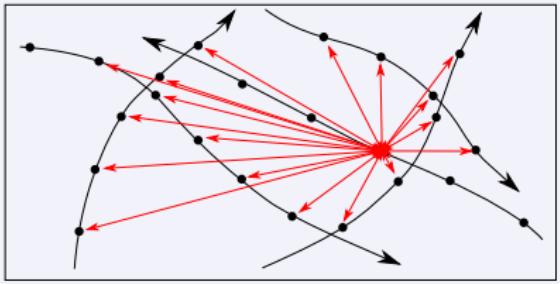


Spatial Conflict Detection

- Spatial conflict, if trajectory points are close (3 nautical miles) to each other.

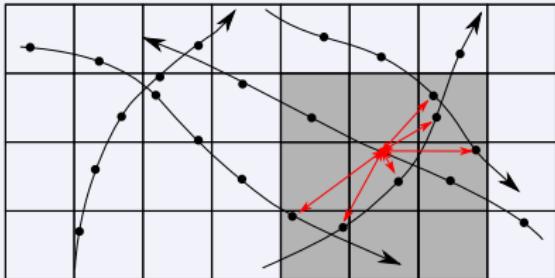
Brute force algorithm

- Check distance between nearly all trajectory points



Coarse grid algorithm

- Map trajectory points to coarse grid

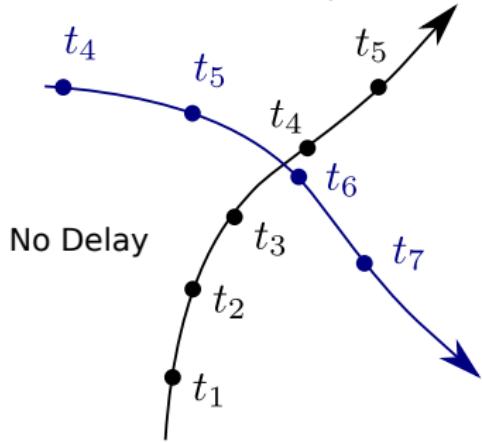


- Check distance only with neighboring cells

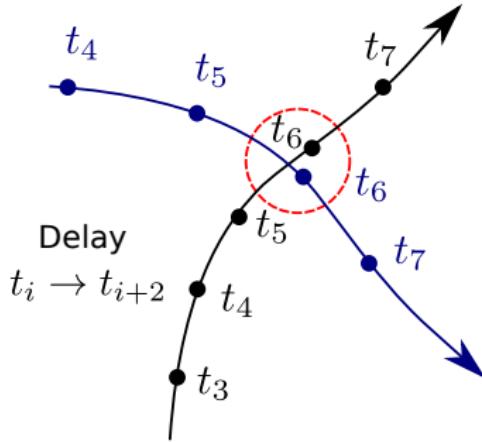


Potential Conflicts

- Potential conflict: Spatial conflict which can become real conflict



Spatial Conflict



Real Conflict

- First step: Potential conflict, if difference in wind-optimal arrival times $t_{ik} - t_{jk} < 2$ hours.



Potential Conflicts

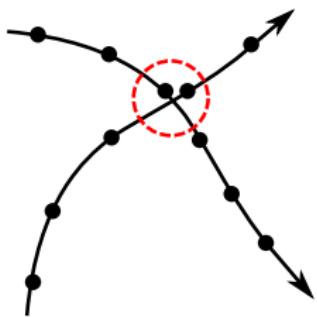
- How to reduce the huge number of potential conflicts: $N_{\text{spatial}} = 43251?$



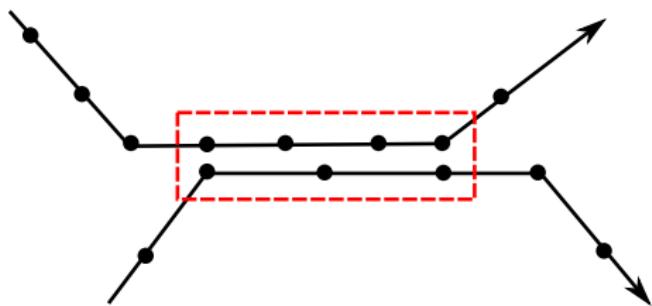
Potential Conflicts - Classification

Reduce the vast number of potential conflicts by categorizing:

- Point Conflict: Isolated in time $N_{\text{point}} = 3573$
- Parallel conflict: Point conflicts consecutive in time $N_{\text{parallel}} = 2510$
- Reduction of 86%



Point Conflict



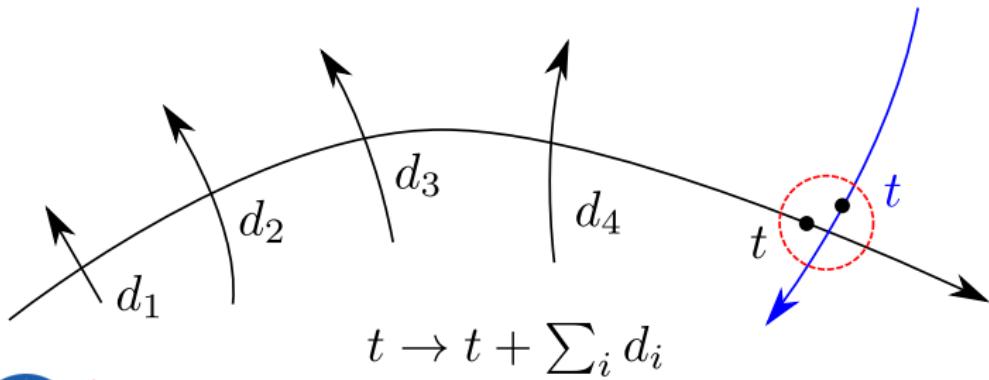
Parallel Conflict



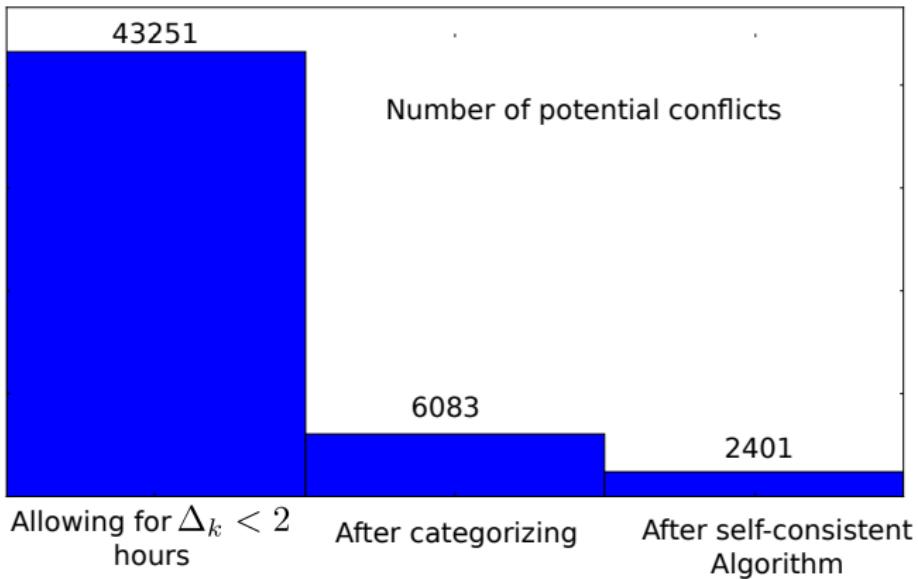
Potential Conflicts - Reduction

Self-consistent algorithm:

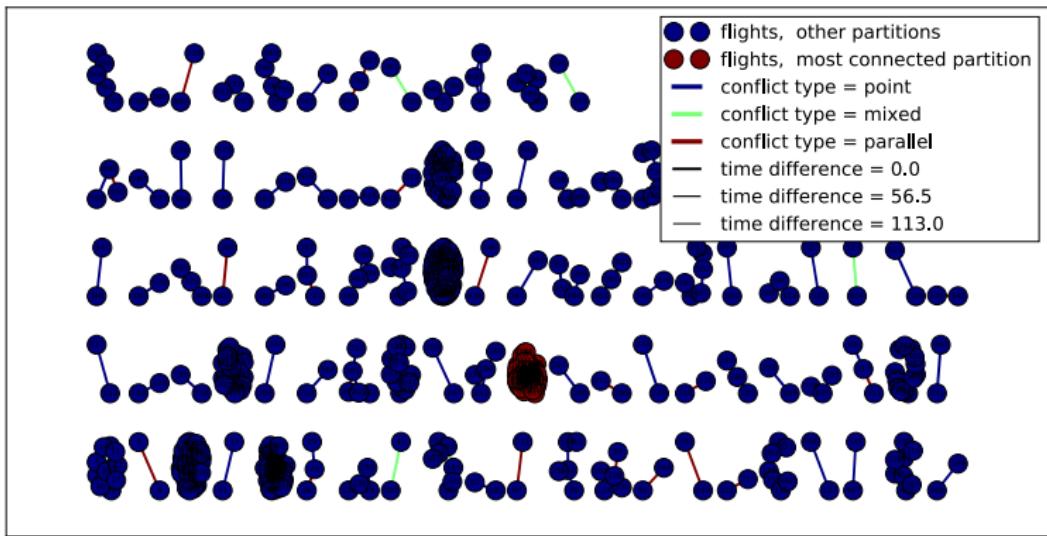
- For each flight, order conflicts in time
- For each potential conflict, calculate the maximal delay of both flights
- Remove potential conflicts which can not become real conflicts
- Repeat the above steps until convergence (N_{spatial} invariant)



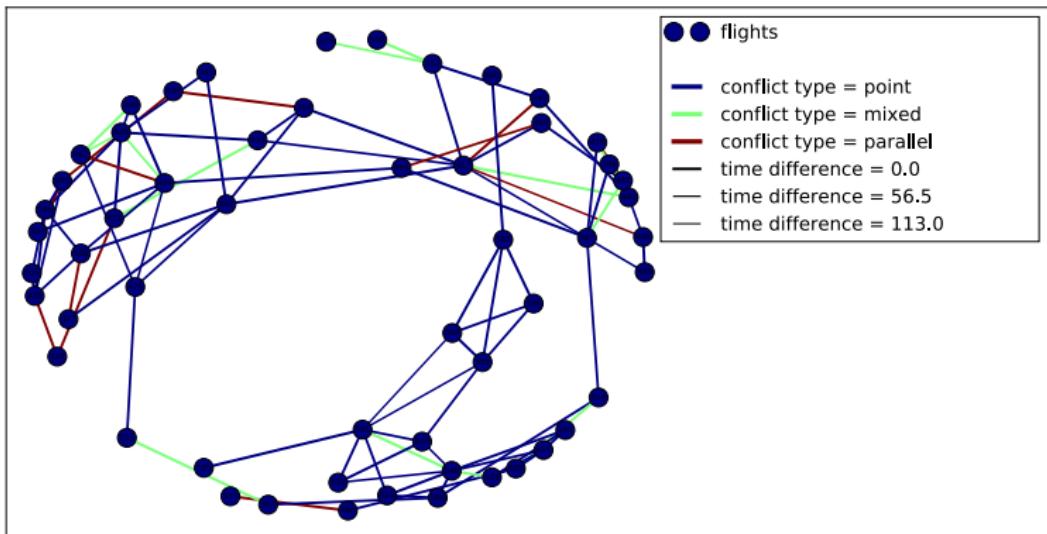
Potential Conflicts - Reduction



Potential Conflicts - Disconnected Subsets



Potential Conflicts - Disconnected Subsets



Potential Conflicts - Disconnected Subsets

Use disconnected subsets

- As starting point for hybrid classical-quantum approaches
- For solving small problems on the D-Wave quantum annealer



Mapping to PUBO

Optimization problem

$$\underset{d_i, d_{ik}, a_k}{\text{minimize}} \quad \sum_i d_i + \sum_{ik} \mathcal{D}_{ik}(\Delta_k, a_k)$$

subject to $\Delta_k = t_{ik} + d_i + \sum_{p < k} d_{ip} - t_{jk} - d_j - \sum_{q < k} d_{jq}$

$$d_{ik} = \mathcal{D}_{ik}(\Delta_k, a_k)$$

- Restrict all variables to integers
- Express integers as binary variables. E.g.

$$d_i = \sum_{\alpha} \alpha d_{i\alpha} \quad \text{with } d_{i\alpha} \in \{0, 1\}$$



Mapping to PUBO

Polynomial Unconstrained Binary Optimization (PUBO)

$$\begin{aligned} & \underset{d_{i\alpha}, d_{ik\beta}, \Delta_{k\delta}, a_{k\gamma}}{\text{minimize}} \sum_{i\alpha} \alpha d_{i\alpha} + \sum_{ik\beta} \beta d_{ik\beta} \\ & + \sum_k \left(\sum_{\delta} \delta \Delta_{k\delta} - t_{ik} - \sum_{\alpha} \alpha d_{i\alpha} - \sum_{\substack{p < k \\ \beta}} \beta d_{ip\beta} + t_{jk} + \sum_{\alpha} \alpha d_{j\alpha} + \sum_{\substack{q < k \\ \beta}} \beta d_{jq\beta} \right)^2 \\ & + \sum_{ik} \left(\sum_{\beta} \beta d_{ik\beta} - \sum_{\delta\gamma} \mathcal{D}_{ik}(\delta, \gamma) \Delta_{k\delta} a_{k\gamma} \right)^2 + \sum_i \left(\sum_{\alpha} d_{i\alpha} - 1 \right)^2 + \dots \end{aligned}$$



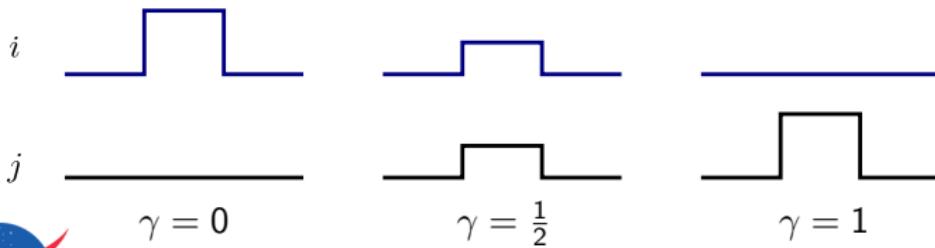
Mapping to PUBO

- Binary variable for difference in arrival times $\Delta_k = \sum_{\delta} \delta \Delta_{k\delta}$
- Conflict avoiding term

$$\sum_{ik} \left(\sum_{\beta} \beta d_{ik\beta} - \sum_{\delta\gamma} \mathcal{D}_{ik}(\delta, \gamma) \Delta_{k\delta} a_{k\gamma} \right)^2$$

where

$$\mathcal{D}_{ik}(\delta, \gamma) = \begin{cases} D & \text{if } |\delta| < 3 \text{ minutes} \\ 0 & \text{else} \end{cases}$$



Reduction to QUBO

- Reduce quartic and cubic terms to quadratic terms by degree reduction
- Drawback: More variables



Simplified Model: Only Departure Delays

- Simplification: No maneuver delays d_{ik}
- Penalize conflicts with

$$\mathcal{P}_k(\Delta_k) = \begin{cases} P & \text{if } |\Delta_k| < 3 \text{ minutes} \\ 0 & \text{else} \end{cases}$$

where $P = \text{const.}$

Optimization problem

$$\underset{d_i}{\text{minimize}} \sum_i d_i + \sum_k \mathcal{P}_k(\Delta_k)$$



Simplified Model - Mapping to QUBO

Difference of conflict arrival times becomes

$$\Delta_k = t_{ik} - t_{jk} + d_i - d_j$$

QUBO

$$\begin{aligned} & \underset{d_{i\alpha}, \Delta_{k\delta}}{\text{minimize}} \quad p_1 \sum_{i\alpha} \alpha d_{i\alpha} + p_2 \sum_k \sum_{\substack{\delta \\ |\delta| < 3\text{min}}} \Delta_{k\delta} \\ & + p_3 \sum_k \left(\sum_{\delta} \delta \Delta_{k\delta} - t_{ik} - \sum_{\alpha} \alpha d_{i\alpha} + t_{jk} + \sum_{\alpha} \alpha d_{j\alpha} \right)^2 \\ & + p_4 \sum_i \left(\sum_{\alpha} d_{i\alpha} - 1 \right)^2 + p_5 \sum_k \left(\sum_{\delta} \Delta_{k\delta} - 1 \right)^2 \end{aligned}$$



Simplified Model - Small Problem Instances

Generate random instances with

- Number of flights: 5, 10, 20, 50
- Number of random conflicts: 5, 10, 20, 50
- Random conflict arrival times
- Departure delays: 0, 3, ..., 18 minutes
- Different sets of penalty weights p_i

Preliminary quantum annealing results (Work in progress)

- So far, no valid solutions
- Possible reason: Precision issue



Summary and Outlook

Summary

- Air traffic management problems can be mapped to PUBO
- Significant classical preprocessing for real data

Next steps

- Map PUBO to QUBO
- Investigate alternative approaches, like Job Shop Scheduling
- Quantum annealing runs

