Optimization and Control of Airport and Air Traffic Flow

Arthur Richards

with a lot of work by Gillian Clare

arthur.richards@bristol.ac.uk
Background
Background
Overview

• Two pieces of work
  – Routing and timing of aircraft taxiing
  – Robust air traffic flow management

• Both exploit ideas from OR and control
  – Integer optimization
  – Cell network modelling
  – Receding horizons / model predictive control
• A **receding horizon, iterative** algorithm **combining taxiing** and **runway** operations in one optimizer, in a **continuous time** environment
A MILP Approach

- Airport layout modelled as a graph
  - *Discrete* decisions of node sequence
  - *Continuous* decisions of passage times

- Posed and solved as *Mixed-Integer Linear Programming (MILP)*
  - Good global solvers available (e.g. CPLEX, Gurobi)
  - Some heritage in airport optimization

- Still highly complex – *NP-hard*
Basic Problem Formulation

• **Key Decision Variables:**

\[ X(a, n, m, k) \text{ binary } = 1 \text{ if and only if aircraft } a \text{ is routed from node } n \text{ to node } m \text{ during planning period } k \]

\[ T(a, k) \text{ the time at which aircraft } a \text{ starts its } k\text{th planning period} \]
Basic Problem Formulation

**Objective:**

\[ J = w_0 t_{end} + w_1 \sum_{a=1}^{N_a} (T(a, N_k) - T(a,1)) \]

\[ + w_3 \sum_{a=1}^{N_a} \sum_{n=1}^{N_n} \sum_{m=1}^{N_n} \left( \sum_{k=2}^{(N_k-1)} L(n, m) X(a, n, m, k) \right) \]

where:
- \( J \) is the objective function.
- \( w_0, w_1, w_3 \) are weights.
- \( t_{end} \) is the end time.
- \( T(a, N_k) \) is the taxi time for agent \( a \).
- \( L(n, m) \) is the distance between \( n \) and \( m \).
- \( X(a, n, m, k) \) is a function indicating travel between \( a \), \( n \), \( m \), and \( k \).
Basic Problem Formulation

**Objective:**

\[ J = w \cdot t_{end} \]

\[ + w_1 \sum_{a=1}^{N_a} (T(a, N_k) - T(a, 1)) \]

\[ + w_2 \sum_{a=1}^{N_a} \sum_{n=1}^{N_n} \sum_{m=1}^{N_n} (N_k - 1) L(n, m) X(a, n, m, k) \]

Last time on runway
Basic Problem Formulation

**Objective:**

\[ J = w_0 t_{end} \]

\[ + w_1 \sum_{a=1}^{N_a} (T(a, N_k) - T(a, 1)) \]  
\[ \text{Taxi time for } a \]

\[ + w_3 \sum_{a=1}^{N_a} \sum_{n=1}^{N_n} \sum_{m=1}^{N_n} \sum_{k=2}^{(N_k-1)} L(n, m) X(a, n, m, k) \]  
\[ \text{Distance travelled by } a \]

Last time on runway

Last movement
Basic Problem Formulation

**Objective:**

\[ J = w_0 t_{\text{end}} + w_1 \sum_{a=1}^{N_a} (T(a, N_k) - T(a, 1)) \]

Taxi time for \( a \)

\[ + w_3 \sum_{a=1}^{N_a} \sum_{n=1}^{N_n} \sum_{m=1}^{N_n} (N_k - 1) \sum_{k=2}^{N_k} L(n, m) X(a, n, m, k) \]

Distance travelled by \( a \)

Distance between nodes \( n \) and \( m \)

Last time on runway

Last movement
Basic Problem Formulation

**Constraints:**

- Initial Conditions
- Routing
- Taxi Timing
- Taxi Conflict
- Runway Timing — wake vortex / SID separation

A.

B.
Scaling Up the Problem

- Initial formulation handles up to 6 aircraft
  - Identified potential savings and coupling
    - Keith, Richards & Sharma, AIAA GNC 2008

- To handle larger cases:
  - Constraint Iteration
  - Receding Horizon / Rolling Window
    - Clare et al, GNC 2009
  - …and include arrivals
Constraint Iteration: Basic Idea

(Iterative MILP ~ Earl and D’Andrea (2005))

Start

Remove taxi conflict constraints from the problem formulation

Solve

Any conflicts? N

End

Add in constraints to prevent detected conflicts
Constraint Iteration: Results

- Solve Time (s)
- Number of Aircraft in Problem

- Iterative
- Non-Iterative

- 1 Hour
- 1 Min
- 1 Sec
Receding Horizon

- Plan in detail only up to a **planning** horizon
- Execute up to execution horizon before **re-planning**
Receding Horizon

- Plan in detail only up to a *planning* horizon
- Execute up to execution horizon before *re-planning*
Receding Horizon

- **Plan sections:**
  - *Detailed plan* by MILP timing and routing
  - *Approximate plan* using shortest path
  - *Anticipated runway time* subject to scheduling constraints
Receding Horizon

• Idea linked to *rolling window* of aircraft
  – All aircraft currently moving
  – All aircraft scheduled for arrival or push back within the next execution horizon (---)

• Also include *foresight* aircraft departures
  – Scheduled for push back within next 400s (- - - -)
  – No detail plan: shortest path & runway timing only
  – *Reduces “shuffling”* in departure order
Receding Horizon: Computation Results

![Diagram showing solve time (s) vs. number of aircraft for different time horizons (1 sec, 1 min, 1 hour). The graph includes data points for RH - Single Horizon, RH - Total Problem, and Iterative methods.]
RH: Large Scale Result

*126 node 240 aircraft*

Heathrow problem

55% Total Taxi Time saving over FCFS

Runway time created
Large Scale Result:
Taxi Times Comparison to FCFS

FCFS Taxi Time (s)

1:1

10 mins
5 mins
Large Scale Result:
Take-off Times Comparison to FCFS

17 minutes of available runway time

Earlier with RH

FCFS = RH

Later with RH
Receding Horizon: Computation

- Total simulation running time, including optimization: 5 hours

- *Close to real time operation*
ONBOARD

• A SESAR WP-E research project looking at uncertainty handling in Air Traffic Management
  – Led by GMV, who developed airline operations optimizer
  – Bristol developed tactical flow optimizer
Tactical Flow

- Primarily concerned with two sources of uncertainty:
  - Weather
  - Unscheduled demand
  - Some forecasting
Demand Capacity Balance

- Preferred option is to increase capacity but not always an option.

- Assign delays to limit aggregate flow rates to available capacity.
Handling Uncertainty

• Goal: optimize flow management to minimize delays subject to capacity limits
  – Don’t know exactly what’s going to happen

• Options:
  – Design delays for nominal conditions
  – Design delays to suit all possible conditions
  – Design delay policy to respond to conditions
Handling Uncertainty

• Goal: optimize flow management to minimize delays subject to capacity limits
  – Don’t know exactly what’s going to happen

• Options:
  Probably won’t work for nominal conditions
  – Design delays to suit all possible conditions
  – Design delay policy to respond to conditions
Handling Uncertainty

• Goal: optimize flow management to minimize delays subject to capacity limits
  – Don’t know exactly what’s going to happen

• Options:
  – Design delays for nominal conditions
    Probably won’t work or even infeasible
  – Design delay policy to respond to conditions
Handling Uncertainty

• Goal: optimize flow management to minimize delays subject to capacity limits
  – Don’t know exactly what’s going to happen

• Options:
  – Design delays for nominal conditions
  – Design delays to suit all possible conditions
  – Design delay policy to respond to conditions

  **Probably won’t work**

  Conservative or even infeasible

  – Design delay policy to respond to conditions

  Sounds like feedback robust MPC: can we exploit those ideas?
Aggregated Flow

Paths are grouped by destination and split into a series of cells which each represent a sector in the shared flight path. Control actions represented as binaries.

\[ u^i(k) = \text{no. aircraft held back at cell } i \text{ in time period } k \]
\[ u^{i,j}(k) = \text{no. aircraft moving, cell } i \rightarrow j \text{ in time period } k \]
Flow Optimization

Objective:
Minimize weighted sum of **Airborne Delay + Ground Delay**

\[
\min \sum_{k \in T} \left( \sum_{s \in S} \sum_{i \in B(s)} c_a u^{i}(k) + \sum_{a \in A} \sum_{i \in B(a)} c_g u^{i}(k) \right)
\]

Capacity Constraints:

\[
\sum_{i \in B(s)} \left( u^{i}(k) + \sum_{j \in L_i} u^{i,j}(k) \right) \leq C_s(k)
\]

\[\forall s, k \in T : k > 1\]
Uncertainty 1: Scenarios

Each branch point represented by binary variable $W_n$

Each scenario $e$ has associated capacity reduction $q$ and demand variation $f$

\[
\sum_{i \in B(s)} \left( u^i(k) + \sum_{j \in L_i} u^{i,j}(k) \right) \leq c_s(k) - q(e, s, k) \\
\forall s, k \in T : k > 1
\]

\[
\overline{f}^i(e, k) = f^i(k) + f_d^i(e, k)
\]
Scenario Feedback

**Re-Formulated Control Variables:**

\[ u^i(k) = v^i(k) + \sum_{n: tw(n) < k} M^i_n(k) W_n(c) \]

\[ u^{i,j}(k) = v^{i,j}(k) + \sum_{n: tw(n) < k} N^{i,j}_n(k) W_n(c) \]

**Re-Formulated Objectives:**

\[ \min_{\epsilon_1} \sum_{k \in T} \left( \sum_{s \in S} \sum_{i \in B(s)} c_{td} v^i(k) + \sum_{a \in A} \sum_{i \in B(a)} c_g v^i(k) \right) \]

\[ + \epsilon_2 \sum_{w \in W} \sum_{k \in T} \left( \sum_{s \in S} \sum_{i \in B(s)} c_{td} u^i(k) + \sum_{a \in A} \sum_{i \in B(a)} c_g u^i(k) \right) \]

- **Delay Cost of nominal (disturbance-free) plan**
- **Delay Cost of disturbance recovery plans**

New decision variables representing feedback.
Uncertainty 2: Virtual Aircraft

- Instead of scenarios, model a second set of flows *beyond our control*
  - $\tilde{u}^{i,j}(k) =$ number of virtual flights moving $i$ to $j$ in time $k$

- Can represent both unscheduled demand and weather
  - We can express our scenarios in terms of $\tilde{u}$

- New feedback
  
  $u^{i,j}(k) = v^{i,j}(k) + \sum_{n,m,p<k} \tilde{N}^{i,j,m,n}(p,k)\tilde{u}^{m,n}(p)$

- Why two models?
  - W more efficient for independent scenarios (e.g. weather)
  - Virtual a/c better for linear dependence (e.g. unscheduled demand)
Initial Test Case

- 30 flights
- 5 airports
- Flights between 06:00h and 16:00h
- 5-aircraft capacity limit for airspace sectors
- 5-minute time windows
Initial Test Case

Capacity Reductions

- 4 storms, one subject to some speed uncertainty

- Storms reduce capacities to 1 aircraft per 5-minute time window.
Benefits of Feedback

- **Initial** plan violates capacity
- **Nominal** ($c_1$) doesn’t always fix it
- **Robust** never violates capacity but excessively delays $c_1$-$c_3$
- **Feedback** never violates capacity and allows $c_1$-$c_3$ to work better
  - Cost of higher computation
Larger Example

- 205 flights over 18 hours
  - 224 periods of 5 minutes each
  - 21 sectors around London
  - 9 weather scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>No. Sector Breaches</th>
<th>Total Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOC Ideal Plan</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>c1</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>c2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>c3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Solve Time: n/a</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c4</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>c5</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>c6</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>c7</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>c8</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>c9</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Nominal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c1</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c2</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c3</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Solve Time: 9.3 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c4</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>c5</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>c6</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>c7</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c8</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c9</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Robust</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c1</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c2</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c3</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>Solve Time: 65.4 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c4</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c5</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c6</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c7</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c8</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c9</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>Disturbance Feedback</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c1</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c2</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>c3</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Solve Time: 326.4 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c4</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c5</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c6</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c7</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>c8</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>c9</td>
<td>0</td>
<td>15</td>
</tr>
</tbody>
</table>
Unscheduled Demand

- 50 flights with 4 or 5 uncertain entry times
  - 16 ($2^4$) or 32 ($2^5$) scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Solve Time (s):</th>
<th>Scenario Tree</th>
<th>Virtual Aircraft</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>Objective:</td>
<td>293</td>
<td>173</td>
</tr>
<tr>
<td></td>
<td>No Variables:</td>
<td>1140.8</td>
<td>1142.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>57,292</td>
<td>17,855</td>
</tr>
<tr>
<td>32</td>
<td>Solve Time (s):</td>
<td>1485</td>
<td>504</td>
</tr>
<tr>
<td></td>
<td>Objective:</td>
<td>1886.4</td>
<td>1889.6</td>
</tr>
<tr>
<td></td>
<td>No Variables:</td>
<td>108,976</td>
<td>21,074</td>
</tr>
</tbody>
</table>

- Key to improving computation: carefully identify and remove redundant variables before the optimization
Where to next?

- Richer airport problem?
  - De-icing; inspection; gates
  - Robustness
  - Hybrid solver

- Deeper investigations
  - Decomposition for robustness
  - Better terminal constraints
More Information

• Clare, G.L.; Richards, A.G., "Optimization of Taxiway Routing and Runway Scheduling," *Intelligent Transportation Systems, IEEE Transactions on*, vol.12, no.4
  – 10.1109/TITS.2011.2131650

• Clare, G.; Richards, A., "Disturbance feedback for handling uncertainty in Air Traffic Flow Management," 2013 European Control Conference (ECC)
More Information

• ONBOARD: http://www.onboard-sesar.eu/

• SUPEROPT (another SESAR opt project)
  – wikis.bris.ac.uk/display/agc/SUPEROPT

• Bristol’s new CDT in autonomous systems:
  – http://farscope.bris.ac.uk

• Arthur’s website
  – http://seis.bris.ac.uk/~aeagr