DYNAMIC SECTORIZATION IN ACTION 2
Introduction
Introduction

- Air traffic controllers are persons who operate the air traffic control system to expedite and maintain a safe and orderly flow of air traffic. They typically accepts traffic from, and ultimately passes traffic to, the control of a Terminal Control Center or of another Center.
- Sector are geographic subdivisions in the airspace. Every sector belongs to a center.
Motivation

• Humans in charge: Air Traffic Controllers
• How to improve their job?
• Objective: minimize the workload of the Air Traffic Controllers
• Problem: the complexity of NAS
• Approach chosen in this project: dynamic boundaries
Problem Statement

• The main purpose is here to move the boundaries of the sectors in order to decrease the air traffic controller workload. By focusing only on en-route sectors, the choice of the Air Traffic Controller workload metric which was made is the peak count in a sector.

• How to resectorize the national air space (NAS) so that the maximum peak count of aircrafts per polygon is as small as possible.

• Analyzing flight plan is implicated directly into capacity distribution of the system
The primary difficulty in modeling the ATC workload is to select the most relevant factors amongst the many potential variables affecting it: the number of aircraft (which contains peak counts), the presence of conflicts (number of intersections), the climbing and descending flights (traffic mix with arrivals, departures and overflights), the sector itself (geometry, size), the coordination and of course the traffic flow structure.
Fixed vs Dynamic Boundaries

- Fixed boundaries: start with an initial sectorization setting and then change the center to which sectors belong.

- Dynamic boundaries: change the geometry of the sectors
Modeling

Flow Model, Grid Model, Hybrid Model
Big Picture

ASDI

Flow Model  Occupancy Grid

Weighted graph

Spectral Bisection

Geometric Sector Construction
This model is able to give a graph representing the main flows for the NAS by aggregating the routes at a sector level.
Flow Model

ARTCC level

sector level (ZOA33)

Large Capacity Cell Transmission Model.
Charles Robelin & Dengfeng Sun

~ 650 nm

~ 180 nm

link level

cell

1 minute flight time (~8-9 nm)
Occupancy Grid

Compute cumulative or peak number of aircrafts for every cell. Model developed by S. Martinez
Occupancy Grid
Occupancy Grid
Use any Clustering Algorithm to influence all the cells by a node. Region growing or K-mean
Add the count in each cell affected by the corresponding nodes. Now we have weighted nodes.
Spectral Bisection

It is based on the Laplacian Matrix
- Keep highest connectivity
- Cut the least edges.

Repeat the same but with constraints are: number of sectors (actual number) or max peak count (less than limit)
Sector construction

Use Influence area
Sector construction

Use Voronoi diagrams
Occupancy Grid
Building ASDI Raw Data

ASDI format, script language, PScript, Route finder, ICAO/IATA
### BERLIN - BER

<table>
<thead>
<tr>
<th>Flight</th>
<th>Arrival</th>
<th>Departure</th>
<th>Time 1</th>
<th>Operator 1</th>
<th>Date 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234567</td>
<td>06:00</td>
<td>07:35</td>
<td>01:35</td>
<td>KL1818</td>
<td>16/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>10:35</td>
<td>12:05</td>
<td>01:30</td>
<td>KL1822</td>
<td>16/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>12:10</td>
<td>13:45</td>
<td>01:35</td>
<td>KL1824</td>
<td>16/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>17:25</td>
<td>18:55</td>
<td>01:30</td>
<td>KL1830</td>
<td>16/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>20:00</td>
<td>21:30</td>
<td>01:30</td>
<td>KL1834</td>
<td>19/02-23/03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flight</th>
<th>Arrival</th>
<th>Departure</th>
<th>Time 1</th>
<th>Operator 1</th>
<th>Date 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234567</td>
<td>07:10</td>
<td>08:55</td>
<td>01:45</td>
<td>AF1135</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>09:55</td>
<td>11:40</td>
<td>01:45</td>
<td>AF1435</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>12:55</td>
<td>14:40</td>
<td>01:45</td>
<td>AF1735</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>15:55</td>
<td>17:40</td>
<td>01:45</td>
<td>AF2035</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>18:55</td>
<td>20:40</td>
<td>01:45</td>
<td>AF2335</td>
<td>17/01-24/03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flight</th>
<th>Arrival</th>
<th>Departure</th>
<th>Time 1</th>
<th>Operator 1</th>
<th>Date 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234567</td>
<td>08:55</td>
<td>09:55</td>
<td>01:00</td>
<td>AF6301(CK)</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>19:10</td>
<td>20:10</td>
<td>01:00</td>
<td>AF6303(CK)</td>
<td>17/01-23/03</td>
</tr>
</tbody>
</table>

### BEYROUTH - BEY

<table>
<thead>
<tr>
<th>Flight</th>
<th>Arrival</th>
<th>Departure</th>
<th>Time 1</th>
<th>Operator 1</th>
<th>Date 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234567</td>
<td>02:05</td>
<td>05:45</td>
<td>04:40</td>
<td>AF563(ME)</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>08:05</td>
<td>11:45</td>
<td>04:40</td>
<td>AF569(ME)</td>
<td>17/01-24/03</td>
</tr>
<tr>
<td>1234567</td>
<td>15:55</td>
<td>19:35</td>
<td>04:40</td>
<td>AF565</td>
<td>17/01-24/03</td>
</tr>
</tbody>
</table>
ASDI Data

- ASDI is an acronym for Aircraft Situation Display to Industry. The ASDI data stream is a service made available through the U.S. Department of Transportation's Volpe Transportation Center. The ASDI stream consists of data elements which show the position and flight plans of all aircraft in U.S. and optionally, UK airspace.
### ASDI Data (text file)

<table>
<thead>
<tr>
<th>Entry</th>
<th>Day</th>
<th>Hour</th>
<th>Min</th>
<th>Sec</th>
<th>Latitude</th>
<th>Longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>0A1F09235440KSDFTZ</td>
<td>09</td>
<td>23</td>
<td>54</td>
<td>40</td>
<td>3908N</td>
<td>08609W</td>
</tr>
</tbody>
</table>

09 : day
23 : hour
54 : min
40 : sec

3908N : latitude 39 degre 08 min
08609W : longitude 086 degre 09 min
088 : FL
<asdiMessage sourceFacility="KZKC" sourceTimeStamp="2007-01-31T18:59:50.0Z">
  <trackInformation>
    <nxcm:aircraftId>N154NS</nxcm:aircraftId>
    <nxcm:computerId>
      <nxce:idNumber>254</nxce:idNumber>
    </nxcm:computerId>
    <nxcm:speed>493</nxcm:speed>
    <nxcm:reportedAltitude>
      <nxce:assignedAltitude>
        <nxce:simpleAltitude>244C</nxce:simpleAltitude>
      </nxce:assignedAltitude>
    </nxcm:reportedAltitude>
    <nxcm:position>
      <nxce:latitude>
        <nxce:latitudeDMS degrees="38" minutes="19" direction="NORTH"/>
      </nxce:latitude>
      <nxce:longitude>
        <nxce:longitudeDMS degrees="088" minutes="52" direction="WEST"/>
      </nxce:longitude>
    </nxcm:position>
  </trackInformation>
</asdiMessage>
Scripting languages (commonly called scripting programming languages or script languages) are computer programming languages that are typically interpreted and can be typed directly from a keyboard. Thus, scripts are often distinguished from programs, because programs are converted permanently into binary executable files (i.e., zeros and ones) before they are run. Scripts remain in their original form and are interpreted command-by-command each time they are run. Scripts were created to shorten the traditional edit-compile-link-run process.
Script Language: PScript

Data retrieval
s(n1:n2), s(n), s(n1:), s('val')
i(n1:n2), i(n), i(n:), i('val')
ls(in1,in2,in3,...) where in:=s,i,or
*, *(n)

Logical
or(in1,in2,...) where in:=s,i,ls
if(in) where in:=s,i,ls,or,not
not(in) where in:=s,i,ls,or
Loops
hloop(n), hloop(n1:)
vloop(n), vloop(n1:)

Other
end
break
jump(n)
Route Finder

- Using the Aeronautical Information Management tools developed by the company ASA srl (Italy founded in 1991), RouteFinder offers a powerful set of tools aimed at effective flight planning (on PC flight simulation environment only!)
- They have set up a custom PHP interface script that we can query via HTTP
### Route Finder

<table>
<thead>
<tr>
<th>ID</th>
<th>DIST</th>
<th>Coords</th>
<th>Name/Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>N49°00'46.00&quot; E002°33'00.00&quot;</td>
<td>CHARLES DE GAULLE</td>
</tr>
<tr>
<td>NURMO</td>
<td>49</td>
<td>N49°49'34.00&quot; E002°45'19.00&quot;</td>
<td>NURMO</td>
</tr>
<tr>
<td>DIPER</td>
<td>41</td>
<td>N50°20'45.00&quot; E002°03'41.00&quot;</td>
<td>DIPER</td>
</tr>
<tr>
<td>VESAN</td>
<td>2</td>
<td>N50°22'19.00&quot; E002°01'35.00&quot;</td>
<td>VESAN</td>
</tr>
<tr>
<td>RATUK</td>
<td>23</td>
<td>N50°39'25.00&quot; E001°38'11.00&quot;</td>
<td>RATUK</td>
</tr>
<tr>
<td>SOVAT</td>
<td>10</td>
<td>N50°46'46.00&quot; E001°28'00.00&quot;</td>
<td>SOVAT</td>
</tr>
<tr>
<td>SANDY</td>
<td>23</td>
<td>N51°03'51.00&quot; E001°04'03.00&quot;</td>
<td>SANDY</td>
</tr>
<tr>
<td>EGLL</td>
<td>63</td>
<td>N51°28'39.00&quot; W000°27'41.00&quot;</td>
<td>HEATHROW</td>
</tr>
</tbody>
</table>

Charles De Gaulle to Heathrow Airport
ICAO vs IATA Codes

• The ICAO airport code or location indicator is a four-letter alphanumeric code designating each airport around the world. These codes are defined by the International Civil Aviation Organization. The ICAO codes are used by air traffic control and airline operations such as flight planning.

• An IATA airport code, also known an IATA location identifier, IATA station code or simply a location identifier, is a three-letter code designating many airports around the world, defined by the International Air Transport Association (IATA). The characters prominently displayed on baggage tags attached at airport check-in desks are an example of a way these codes are used.
Airline Tables → IATA O/D → ICAO O/D → Routes → ASDI
Airline Companies Timetable

All airline company issue every 7 months a detailed timetable of all its air flights: Origin, Desitination, Local time, operating days, Departure time, arrival time, duration, validity period.
<table>
<thead>
<tr>
<th>Jours</th>
<th>Dép.</th>
<th>Arr.</th>
<th>Durée</th>
<th>N° Vol</th>
<th>Via</th>
<th>Validité</th>
</tr>
</thead>
</table>

**PARIS**

→ Aberdeen (ABZ) → PAR

- **1234567** 10.00 2 11.05 02:05 AF5556 (YS)
- **1234567** 15.25 2 16.30 02:05 AF5558 (YS)

**PARIS**

→ Hartford (CT) (HFD) → PAR

- **1234567** 11.10 2 21.40 15.30 AF8998 (CL)/AF9076 (DL) ATL 12/03/24/03
- **1234567** 2 19.20 2 15.40 14.20 AF174 12/03/24/03

**PARIS**

→ San Francisco (CA) (SFO) → PAR

- **1234567** 10.15 2 12.30 11:15 AFD84 09/04/27/10
- **1234567** 10.15 2 12.55 11:40 AFD84 25/03/08/04

---

**Exemple fictif**

*imaginary example / Fiktives Beispiel*
Chunk of the code that parse the AF file
Conclusion
• Validate the Dynamic sectorization algorithm over Europe and compare the results with the actual sectors.