

# **Simulation-Based Study to Quantify Data-Communication Benefits in Congested Airport Terminal Area**

Gabriele Enea

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Master of Science  
In  
Civil and Environmental Engineering

Dr. Antonio A. Trani

Dr. Hojong Baik

Dr. Amedeo R. Odoni

Dr. Dusan Teodorovic

January 25, 2008

Blacksburg, VA

Keywords: Air Traffic Management, Air Traffic Control, RAMS, CPDLC, NextGen.

# **Simulation-Based Study to Quantify Data-Communication Benefits in Congested Airport Terminal Area**

Gabriele Enea

## **ABSTRACT**

The scope of this study was to evaluate the impact of the air traffic controller-to-pilot communication standard known as CPDLC or Data-Communication on the future air traffic operations. The impact was evaluated from the double prospective of airport delays and air traffic controllers workload. RAMS simulation software is used to perform all the runs and from its output data the values of terminal area delays and controllers workload are extrapolated. The New York Metroplex terminal area was used as the case study. Because of its complexity, where three major airports (i.e. JFK, Newark, and La Guardia) interact and constraint each other, this area was particularly interesting to be studied and the data analyzed gave a valuable insight on the possible future impact of Data-Communication in congested terminal areas. The results of the study, based on some previous man-in-the-loop simulations performed by the FAA in the nineties, showed that significant potential benefits could be obtained with the complete implementation of such technologies in the workload experienced by air traffic controllers. Moreover some small but not negligible benefits were obtained in the total delays accrued by each airport studied. On the other hand, the simulations of the future demand predicted by the FAA demonstrated that without a significant increment in capacity or limitation on the traffic growth intolerable delays would be recorded across the NAS in the future. For the complexity of the simulation model calibration and for the very time-consuming run time not all the scenarios described in the methodology were tested, demonstrating the weakness of RAMS as a ground simulation model.

## ACKNOWLEDGEMENTS

I would like to thank Dr. Trani for being a continuous guiding light throughout my research and for helping me in the difficult moments I went through not only to write this thesis. He is a great researcher and an example of warm human being that I would like to be myself in life. I would also like to thank the members of my committee Dr. Hojong Baik, Dr. Amedeo Odoni and Dr. Dusan Teodorovic for being part of this research effort. I would like to thank all my colleagues at the ATSL Lab: Nola, Anand, Jeffrey, Sohn (for the great PDARS analysis), Senanu, Yue-Thing, Yue Xu, Howard. A special thanks goes to Nick for teaching me Matlab and how to make a tie node.

I would like to thank my family, Mom, Dad and Daniele, for supporting me in every possible way during this year and half that I spent at Virginia Tech, they are a big part of this thesis also if they don't suspect it.

I would like to thank my American family in New Jersey, for making me feel home here in US, thanks Monica, Ava, Patrick, Ninni and Sherry.

I would like to thank my grandparents for protecting me from wherever they are.

I would like to thank my Italian crew in Blacksburg, the best friends I could have possible found so far from home: Davide (Lecco), Davide (Latina), Carmine, Giovanni (il vero bomber di Lipari), Attilio (Forza Mantova!), Maurizio, Edo, Letizia (Scalda), Andrea, and Giovanni (Forli).

I would like to thank my friend back home, the best supporters of my entire life, thanks Guido, Fabio, Ale, Fre, Claudio, Dario.

I would like also to thank all my soccer buddies for all the great time playing and cursing together.

I would like to thank two special friends I made in the beginning of this experience and that I'm sure will last forever: Carlos and Amir, thanks for being there when I needed you.

I would also like to thank all the GLC guys for being my family far from home: Andreas, Suzy, Kunmi, Rob, Dana, Seyran, Daniel, Adam, Sher, Guni, Paola, and anyone else I'm forgetting.

I would like also to thank Franziska for giving me the best and the worst moments of this entire experience, thanks because I learned a lot from both.

I would like to thank Silvia, for being such a special person in the end of this period; I really hope I'll see you again!

In the end I would like to thank somebody who I don't talk to anymore but who is part of this thesis as much as my parents, grazie Vale!

# Table of Contents

<b>ACKNOWLEDGEMENTS</b> .....	<b>iii</b>
<b>1. INTRODUCTION</b> .....	<b>1</b>
1.1. Recent Trends .....	1
1.2. Scope of NextGen .....	1
1.3. Airspace and Terminal Area Capacity .....	3
1.4. Scope of This Study.....	5
<b>2. LITERATURE REVIEW</b> .....	<b>7</b>
2.1. Introduction .....	7
2.2. Simulation and Airports.....	7
2.3. Controller Pilot Data-Link Studies .....	25
2.4. Controller Workload Studies .....	34
<b>3. METHODOLOGY</b> .....	<b>44</b>
3.1. Model Choice.....	44
3.2. RAMS Model.....	44
3.3. Building the Simulation Scenarios .....	46
3.4. Drawing the Airports .....	47
3.5. Traffic Demand Data.....	53
3.6. Building the sectors .....	58
3.7. Workload Tasks Data.....	62
3.8. Analyzing the New York Air Traffic Control Area .....	63
3.9. Balancing the Operations in JFK .....	66
3.10. Capacity Analysis.....	70
3.11. Aircraft Analysis .....	74
<b>4. RESULTS</b> .....	<b>80</b>
4.1. Capacity Analysis .....	80
4.2. Arrivals Only Analysis .....	83
4.2.1. La Guardia Airport.....	83
4.2.2. Newark Liberty Airport.....	84

4.2.3. John Fitzgerald Kennedy Airport .....	85
4.3. Departures Only Analysis .....	86
4.3.1. La Guardia Airport .....	86
4.3.2. Newark Liberty Airport .....	87
4.3.3. John Fitzgerald Kennedy Airport .....	88
4.4. Mixed Operations Analysis .....	89
4.4.1. La Guardia Airport .....	89
4.4.2. Newark Liberty Airport .....	90
4.4.3. John Fitzgerald Kennedy Airport .....	91
4.5. Practical Capacity Envelope Diagrams .....	92
4.5.1. La Guardia Airport .....	93
4.5.2. Newark Liberty Airport .....	95
4.5.3. John Fitzgerald Kennedy Airport .....	97
4.6. Delay Analysis .....	98
4.6.1. La Guardia Airport .....	98
4.6.2. Newark Liberty Airport .....	101
4.6.3. John Fitzgerald Kennedy Airport .....	104
4.7. Workload Analysis .....	107
<b>5. RESULTS VALIDATION .....</b>	<b>113</b>
5.1. La Guardia Airport .....	114
5.2. Newark Liberty Airport .....	118
5.3. John Fitzgerald Kennedy Airport .....	121
<b>6. CONCLUSIONS AND RECOMMENDATIONS .....</b>	<b>127</b>
6.1. Summary of Results .....	127
6.2. Recommendations .....	128
<b>REFERENCES .....</b>	<b>129</b>
<b>APPENDIX A ACRONYMS .....</b>	<b>134</b>
<b>APPENDIX B FLOWCHART OF THE ANALYSIS .....</b>	<b>135</b>
<b>APPENDIX C 1 MATLAB FILE TO CREATE RAMS DEMAND FILES ...</b>	<b>136</b>

**APPENDIX C 2 MATLAB FILE TO CREATE DETAILED DELAY FILES**  
.....145

**APPENDIX C 3 MATLAB FILE TO ANALYZE THE DELAY FILES.....158**

**APPENDIX C 4 MATLAB FILE TO COUNT THE HOURLY OPERATIONS**  
.....159

## List of Figures

Figure 1.1 Scope and Framework of NextGen.....	2
Figure 1.2 Flowchart of the Study.....	6
Figure 2.1 A Sample Flight Path Envelope (Blumstein 1957, used with permission of the publisher).....	8
Figure 2.2 A Sample Representation of the Model (Blumstein 1959, used with permission of the publisher).....	9
Figure 2.3 Future Path Proposed in the Study (Dunlay 2001, used with permission of the author).....	15
Figure 2.4 Activity Based Airport Simulation Model (Martinez 2001, used with permission of the author).....	16
Figure 2.5 Sample Simulation Results (Martinez 2001, used with permission of the author).....	17
Figure 2.6 Logical Block Graph of the Simulation Model (Gangel 2004, used with permission of the author).....	18
Figure 2.7 Existing and Prospective Airport Layout (Janic 2004, used with permission of the author).....	20
Figure 2.8 The Integrated System of Models (Stamatopoulos 2004, used with permission of the author).....	21
Figure 2.9 The Model for a Generic Airport (Andreatta 2005, used with permission of the author).....	24
Figure 2.10 Flow Chart of the Study Process (Majumdar 2001, used with permission of the author).....	37
Figure 2.11 Flow Chart of the Study (Majumdar 2001, used with permission of the author).....	39
Figure 2.12 Results in a 3d Curve Representation (Majumdar 2001, used with permission of the author).....	40



Figure 2.13 Factors Influencing Controllers Workload (Majumdar 2002, used with permission of the author). ..... 41

Figure 3.1 The Control Elements in RAMS (Majumdar 2001, used with permission of the author). ..... 45

Figure 3.2 Input and Output Structure in RAMS..... 46

Figure 3.3 JFK Airport Plate Superimposed in RAMS..... 48

Figure 3.4 LGA Gate Attribute Window..... 49

Figure 3.5 Runway Attributes Window. EWR 22L Occupancy Characteristics. .... 50

Figure 3.6 Performance Time (ROT) Plus Locking Time Representation. (ISA-Software 2004) ..... 51

Figure 3.7 JFK Arrivals from PDARS Data in Googleearth®. .... 52

Figure 3.8 Detailed Image of JFK Arrival Paths in Googleearth®..... 53

Figure 3.9 ATO-P Forecast for JFK Up to Year 2025. .... 56

Figure 3.10 ATO-P Forecast for EWR Up to Year 2025. .... 57

Figure 3.11 ATO-P Forecast for LGA Up to Year 2025..... 58

Figure 3.12 The Sectors Structure Included in the Simulation..... 61

Figure 3.13 Detail of The Core Sectors in the Simulation. .... 62

Figure 3.14 Capacity Diagram Derived from PDARS Data..... 69

Figure 3.15 Example of Capacity Diagram Calculation..... 71

Figure 3.16 Mixed Operation Demands Tested. .... 73

Figure 3.17 LGA Aircraft Mix..... 74

Figure 3.18 LGA Aircraft Mix by Weight Class..... 75

Figure 3.19 JFK Aircraft Mix..... 76

Figure 3.20 JFK Aircraft Mix by Weight Class..... 77

Figure 3.21 EWR Aircraft Mix..... 78

Figure 3.22 EWR Aircraft Mix by Weight Class..... 79

Figure 4.1 Relationship Between Delay-Related and Ultimate Capacity..... 80

Figure 4.2 LGA “Arrivals Only” Results..... 83

Figure 4.3 EWR “Arrivals Only” Results..... 84

Figure 4.4 JFK “Arrivals Only” Results..... 85

Figure 4.5 LGA “Departures Only” Results. .... 86

Figure 4.6 EWR “Departures Only” Results. ....	87
Figure 4.7 EWR “Departures Only” Results, Area of Interest.....	87
Figure 4.8 JFK “Departures Only” Results. ....	88
Figure 4.9 JFK “Departures Only” Results, Area of Interest.....	88
Figure 4.10 LGA Mixed Operation Results. ....	89
Figure 4.11 LGA Mixed Operation results, Area of Interest. ....	89
Figure 4.12 EWR Mixed Operation Results. ....	90
Figure 4.13 EWR Mixed Operation Results, Area of Interest.....	90
Figure 4.14 JFK Mixed Operation Results. ....	91
Figure 4.15 JFK Mixed Operations Results. ....	92
Figure 4.16 LGA Capacity Envelope Diagram. ....	93
Figure 4.17 EWR Capacity Envelope Diagram. ....	95
Figure 4.18 JFK Capacity Envelope Diagram. ....	97
Figure 4.19 LGA Total Delays Growth, Baseline Scenario. ....	98
Figure 4.20 LGA Average Delay per Operation Baseline Scenario 2006 Demand. ....	100
Figure 4.21 EWR Total Delays Growth, Baseline Scenario. ....	101
Figure 4.22 Queue Accumulation in EWR 2025 Scenario. ....	102
Figure 4.23 EWR Average Delay per Operation Baseline Scenario 2006 Demand. ....	103
Figure 4.24 JFK Total Delays Growth, Baseline Scenario. ....	104
Figure 4.25 JFK Average Delay per Operation Baseline Scenario 2006 Demand.....	106
Figure 4.26 Total Occupation Time.....	109
Figure 4.27 Percentage of Occupation Time. ....	110
Figure 4.28 Occupation Time in Minutes per Hour.....	111
Figure 5.1 LGA Delays Comparison with ASPM.....	114
Figure 5.2 LGA Departure Separations from PDARS Data Analysis. ....	115
Figure 5.3 LGA Departure Separations from RAMS. ....	116
Figure 5.4 LGA Arrival Separations from PDARS Data Analysis. ....	117
Figure 5.5 LGA Arrival Separations from RAMS.....	118
Figure 5.6 EWR Delays Comparison with ASPM.....	118
Figure 5.7 EWR Departure Separations from PDARS Data Analysis. ....	119
Figure 5.8 EWR Departure Separations from RAMS Output. ....	120

Figure 5.9 EWR Arrival Separations from PDARS Data Analysis..... 120  
Figure 5.10 EWR Arrival Separations from RAMS Output. .... 121  
Figure 5.11 JFK Delays Comparison with ASPM..... 122  
Figure 5.12 JFK 31R Arrival Separations from PDARS Data..... 123  
Figure 5.13 JFK 31R Arrival Separations from RAMS Output. .... 124  
Figure 5.14 JFK 31L Arrival Separations from PDARS Data. .... 125  
Figure 5.15 JFK 31L Arrival Separations from RAMS Output. .... 126

## List of Tables

Table 3-1 JFK Frequency List. ....	59
Table 3-2 LGA Frequency List. ....	59
Table 3-3 EWR Frequency List. ....	60
Table 3-4 Scenario 1 Operations. ....	64
Table 3-5 Scenario 2 Operations. ....	64
Table 3-6 Total Coverage With Two Scenarios.....	65
Table 3-7 Scenario 3 Operations. ....	65
Table 3-8 Total Coverage With Three Scenarios.....	66
Table 3-9 Runs Performed for All the Scenarios and Equipment Levels. ....	66
Table 3-10 JFK Original Demand.....	67
Table 3-11 Unbalanced Demand by Hour.....	68
Table 3-12 Demand After the Balancing Process. ....	70
Table 3-13 La Guardia Airport Demand Levels Tested.....	72
Table 3-14 Newark Airport Demand Levels Tested. ....	72
Table 3-15 JFK Airport Demand Levels Tested. ....	73
Table 4-1 Wake Vortex Separation Matrix Baseline Scenario. ....	81
Table 4-2 Wake Vortex Separation Matrix Data-Communication Scenarios.....	81
Table 4-3 Simulated Scenarios.....	82
Table 6-1 Capacity Multipliers Obtained. ....	127

# 1. INTRODUCTION

## 1.1. *Recent Trends*

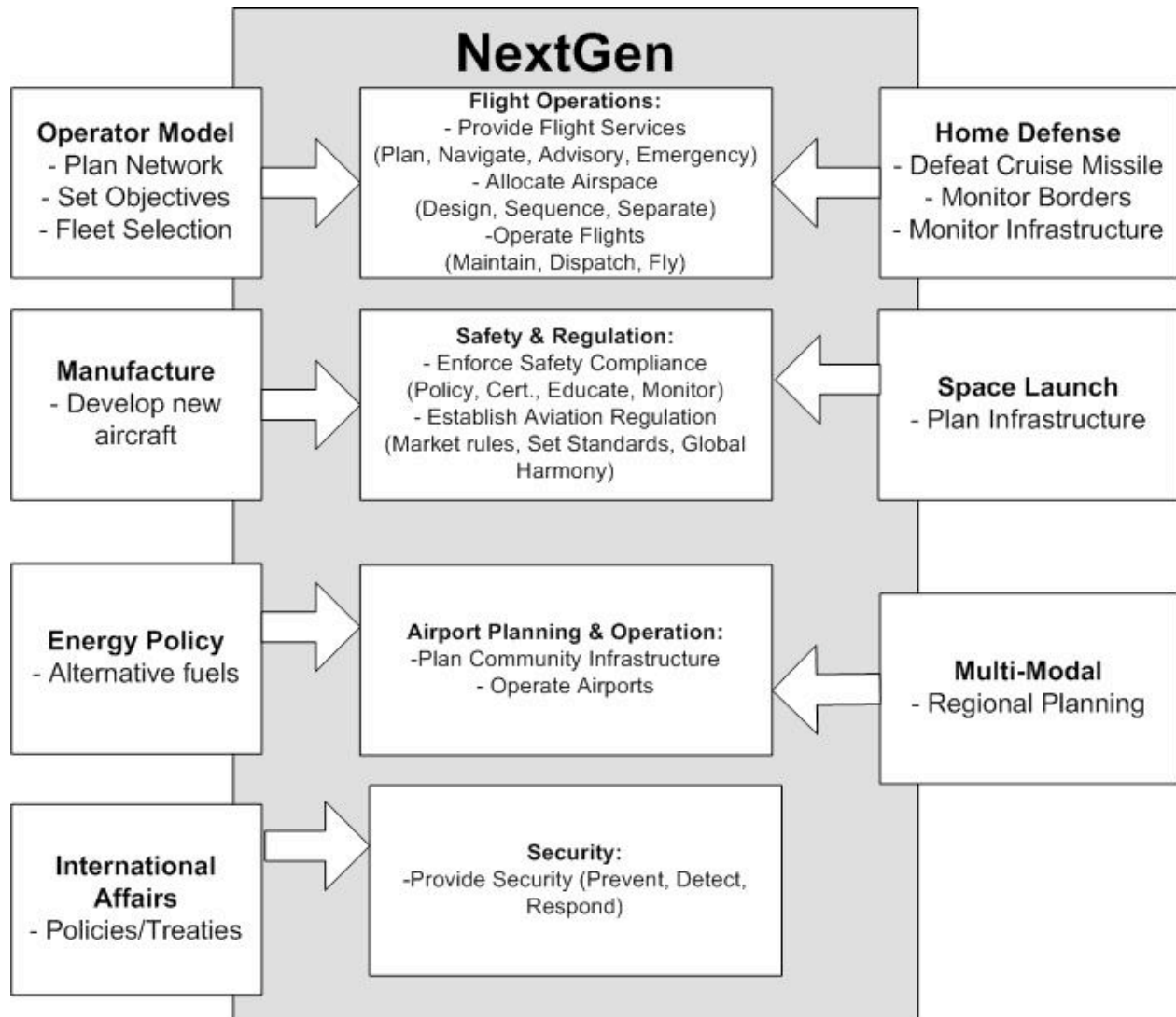
The continuous growth of air traffic is rapidly bringing the National Airspace System (NAS) to congestion levels that are strongly affecting Level Of Service (LOS), delays and safety. These issues will affect the future air transportation demand and the costs of providing commercial air services in the future. “If the FAA didn’t even try to modernize the ATC system (an improbable scenario) it would be spending the same amount over the decade due to the cost of maintaining an aging system”(Hughes 2006). “Given the importance of the air transport industry in the economy, and the lead time necessary to implement any change, courage is needed to invest in the future: a vision is needed!” (Garot 2003).

The Next Generation Air Transportation System (NextGen) is being developed to address the need to grow and to accommodate up to three times the number of operations in today’s system. The Joint Planning & Development Office (JPDO) and other Federal agencies together with FAA are working to move forward the system; the horizon of this is set to be the year 2025. The building of NextGen will not occur in a “big bang” fashion. Rather the system will evolve because the current system must continue operating for years to come. The transition is expected to take two decades. The NextGen System is a system-wide transformation, its core elements must be treated as an integrated set of activities including how outside influences could produce new challenges (Anderegg 2006). In addition to increasing capacity, such transformation must also meet a number of other system performance targets in areas such as noise, emissions, safety and security.

## 1.2. *Scope of NextGen*

The scope and framework of NextGen that can summarize the necessities of the new system are presented in Figure 1.1. In this project we focus on the Operator Model and specifically in the Flight operations block. Inside this block we find all the issues related

to the airspace allocation and management that will play a key role on the capacity changes needed to address the future air transportation demand.



**Figure 1.1 Scope and Framework of NextGen.**

The eight basic capabilities necessary to make NextGen work are (Hughes 2006):

- Network-enabled information access
- Performance-based services
- Assimilation of weather into decision-making
- Layered, adaptive security

- Aircraft 4D trajectory-based operations
- Broad-area precision navigation
- Equivalent visual operations
- Super density operations

The first of these capabilities is a basic requirement to manage twice the air traffic in 2025. The combination of these capabilities will enable air traffic controllers to manage higher density traffic requiring smaller aircraft separations. Moreover, some of the actual duties will be transferred to the cockpit. In the first step only aircraft equipped with high levels technology such as ADS-B (Automatic Dependent Surveillance-Broadcast) and CPDLC (Controller Pilot Data Link Communication) will be allowed to fly with reduced separations but at some point these will be required capabilities for every aircraft in the system to achieve the desired capacity goal.

### ***1.3. Airspace and Terminal Area Capacity***

Separation criteria strongly influence airspace capacity, which is divided into sectors in order to distribute the traffic control into smaller blocks where a limited number of flights can be processed and managed by ATC operators. The capacity of each airspace sector influences the capacity of the system. Many simulation studies have been conducted to quantify the capabilities of an improved ATC/ATM system (FAA 1995; FAA 1996; Smith 2004).

Airspace capacity is generally defined as the number of aircraft handled by a sector in a specific time interval. The controller workload is a critical aspect for the estimation of airspace capacity because it has central functions in traffic prediction, aircraft separation safety distance assurance and pilot instructions. Studies suggest that the upper limit of aircraft per hour by an air traffic controller is 48 man-minutes per hour for a radar controller and 66 man-minutes for a combined radar/manual controller team in several sectors evaluated (Tofukuji 1993). In order to augment these capacity limits two types of innovations can be introduced:

- A new route structure, and
- A new concept in ATC to alleviate controller workload.

The new route structure can be introduced through the application of the so-called Free Flight concept. In this concept, aircraft, with increased level of on board technology, could fly more direct routes exploiting the flight deck navigation equipment. Advances in flight deck technologies such as ADS-B and Datalink are key to reduce aircraft to aircraft separations and thus increase capacity.

In NextGen new concepts to reduce ATC controller workload will involve the application of new procedures for communications between pilots and controller using CPDLC protocols. This type of controller and pilot information exchange via data link reduces the radio broadcast communications and can reduce the time to process each flight. Entering a new airspace sector, traveling through and exiting the sector requires a number of tasks performed by air traffic controllers. Currently, such tasks are performed using voice commands. In a mature data link communication environment some of the tasks could be automated and via codified messages reducing pilot and controller workload and increasing communication security and avoiding misunderstandings.

These technological improvements must be fully tested and evaluated before their introduction in the field. No single model has the scope and the flexibility to perform the full range of simulations required to evaluate all NextGen strategies and their potential impact within a transformed NAS. For these reasons in order to predict future impacts of NextGen deployment a series of different technology scenarios with different models should be ran and evaluated under various assumptions. According to some predictions NextGen will be able to accommodate a 30% increase in airport capacity and 200% increase in airspace sector capacities beyond current capacity (Borener 2006). An increase in capacity will impact the environment as well as safety and security. While these issues are important they are not in the scope of this study.



#### **1.4. Scope of This Study**

The scope of this study is to partially answer to some of the questions above evaluating the impact and the benefit of the new communication technologies known as CPDLC or Data-Communication. Throughout this research effort we will use both notations indifferently meaning the same improvement package studied by FAA in the nineties (FAA 1995; FAA 1996). The benefits evaluated are from the double prospective of delays and air traffic controllers workload. A fast time simulation model, RAMS, was used to test the future scenarios and potential improvement.

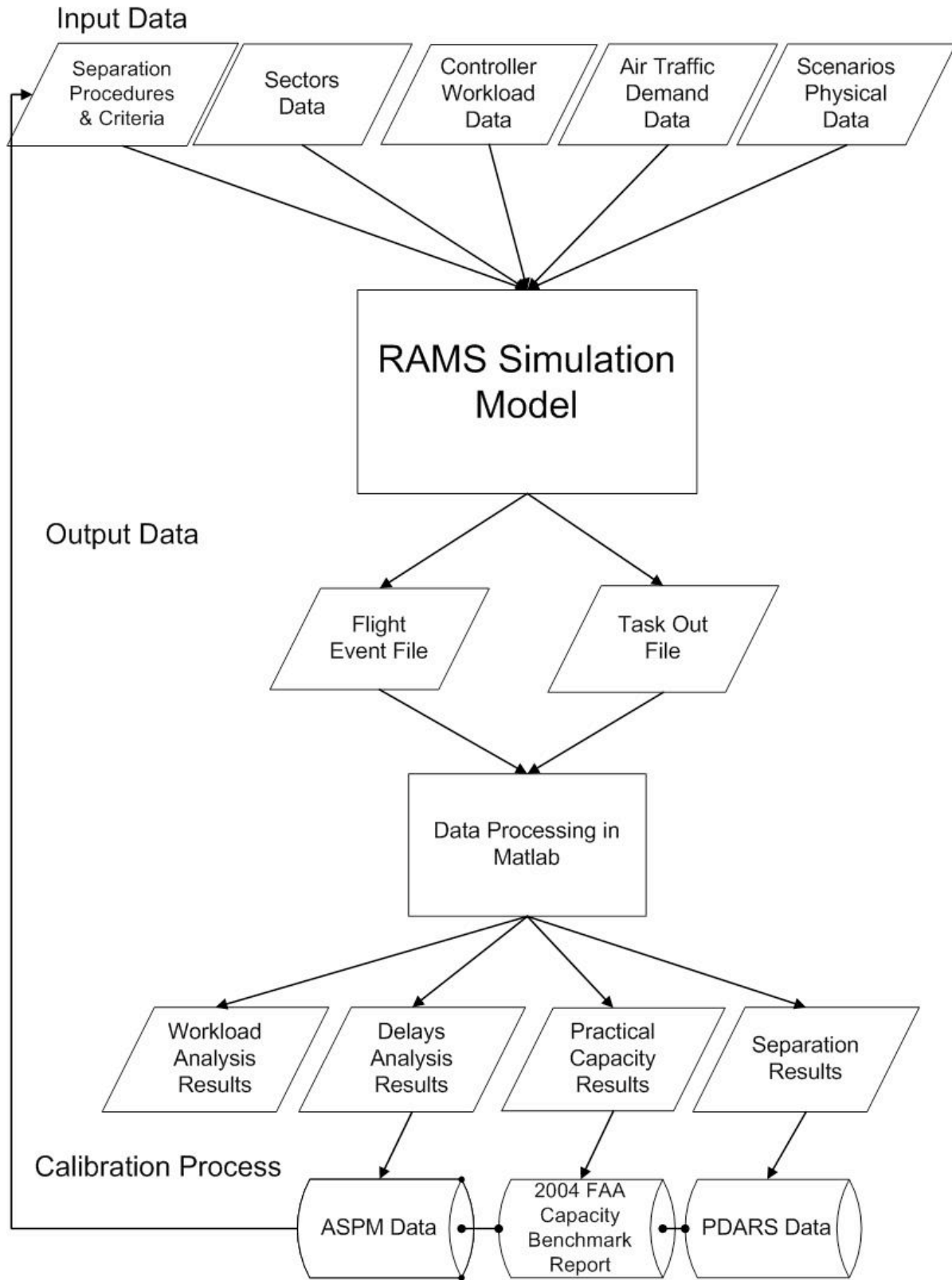


Figure 1.2 Flowchart of the Study.

## **2. LITERATURE REVIEW**

### **2.1. *Introduction***

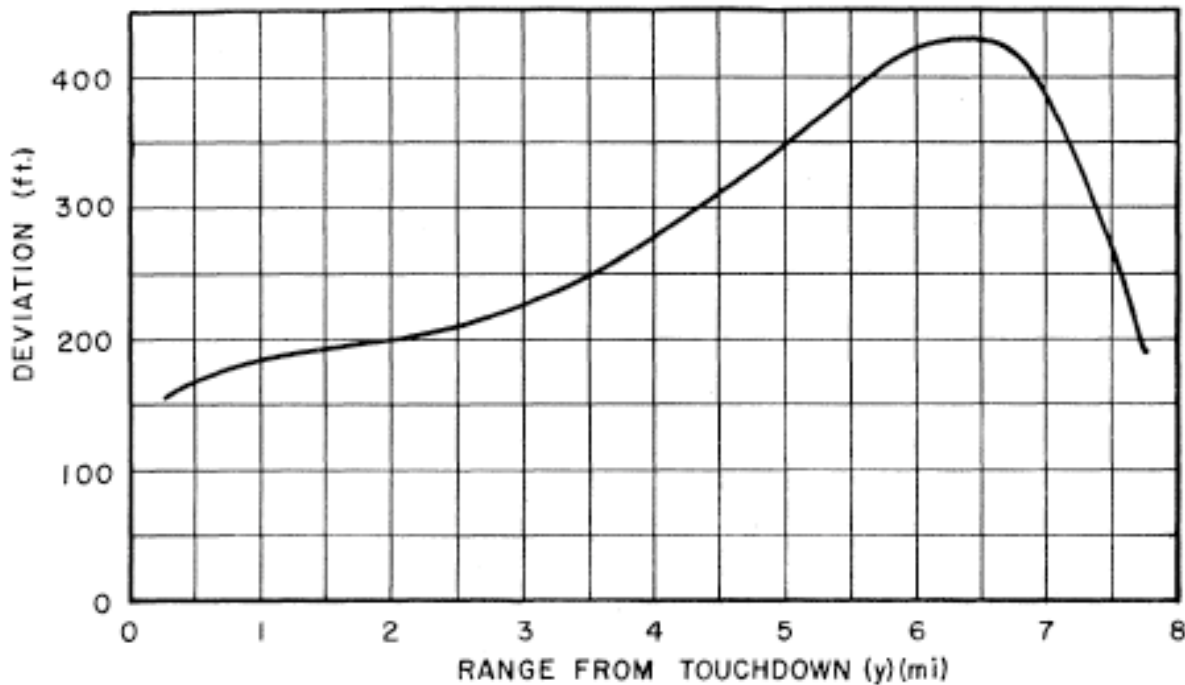
Simulation is the process of designing a model of a real system and conducting experiments on this model for the purpose of understanding the behavior of the system under various concepts of operation and strategies. The power of simulation is the ability to model the dynamics of a real system and to analyze the results (Huang 2004). Simulation studies of aviation systems have been used for decades to study impacts of technology before a system is fielded. Most of the studies focus on the prediction of delays and workload.

The study presented here also use delays and workload to evaluate the performance of an airport terminal area system. The following sections provide information on past efforts to model the terminal area and airport operations. The review includes DATA-LINK studies to understand the possible impact of such technology on TMA operations and on controller workload.

### **2.2. *Simulation and Airports***

One of the first studies found in the literature regarding simulation and airfield capacity is the one performed by Blumstein (Blumstein 1957). This study applied a Monte Carlo Simulation approach to the Ground Control and Approach System to control aircraft under minimum visibility conditions. Through the Monte Carlo simulation method the aircraft paths were generated and stepped through five-second increments. Random numbers were assigned to any of the following events, e.g. initial positioning errors, radar errors, pilot's error, etc., in order to state which events would have occurred. Sample paths generated in this way were analyzed to yield the envelope of airplane positions and the probability of successful landings. By generating a large sample of these paths, detailed information about the system distributions were obtained. An

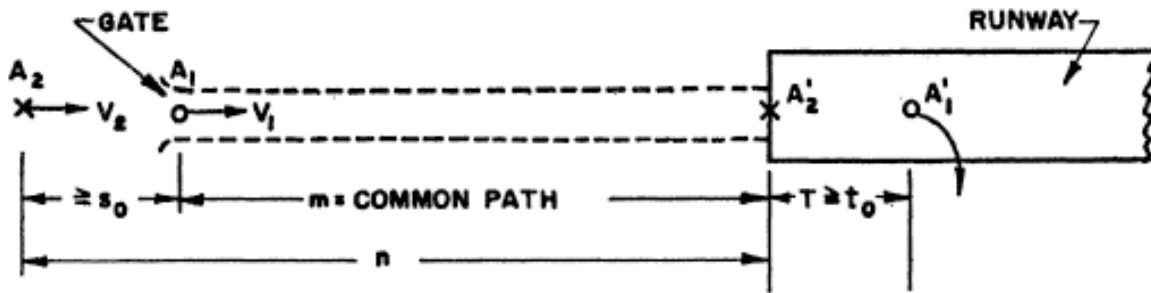
envelope showing the deviations along the approach path (Figure 2.1) encompassing two third of the airplanes was derived. This study focused on evaluating the probability of success of an instrument approach system. The results obtained and the poor quality of the input data was not enough to allow recommendations to traffic control authorities at that time.



**Figure 2.1 A Sample Flight Path Envelope (Blumstein 1957, used with permission of the publisher).**

Another study, that could be probably considered a milestone to asses a single runway capacity was developed by Blumstein (Blumstein 1959). In this study a simple model was built to describe the operations on a single runway airfield. The model incorporate the basic logic that is still used inside modern analytical tools developed for more complex runway configurations (Swedish 1981). After building time-space model of a stream of approaching airplanes on a runway, the basic simulation obtained results in terms of capacity that are very realistic for the period analyzed and for the aircraft performance used at that time. The most important parameters of the model are: aircraft mix, the physical dimension of  $m$  and  $s$ , the common length approach distance and the separation between aircraft, and the minimum landing separation  $t$  (Figure 2.2). Changing these parameters in the model, different results were obtained for the landing

capacity of LaGuardia and Idlewild (today JFK) airports, (i.e. around 30-40 landings per hour). Using this model capacity benefit derived from smaller aircraft separations could be easily obtained. Also the benefits of reducing runway occupancy time (ROT) could be studied.



**Figure 2.2 A Sample Representation of the Model (Blumstein 1959, used with permission of the publisher).**

A complete review of the models available in the period was presented by Odoni (Odoni 1971) in which all the simulation models were divided and classified by surface traffic movements, runway utilization, terminal areas, en route traffic and safety models. In a number of studies the runway system is usually the bottleneck of the airport operations. This category of models was divided into sub-groups considering the technique applied. The sub-groups were runway capacity models and queuing models. One of the most important findings of the report was that the concept of “capacity” is not strictly connected with the runway as a physical entity as one could expect. Because of the scarcity of CPU resources at that time, airport terminal area simulation models were almost impossible build. The models extensions to multiple runways scenarios were discussed but poorly applicable for the reason above. The NAFEC/FAA facilities at Atlantic City were capable to perform realistic real-time simulations as mentioned in (Slattery 1970) were a capacity analysis was performed in the New York Area.

A model based on Blumstein (Blumstein 1959) was expanded and developed for the Terminal Area as a whole in Janic et al. (Janic 1982). In this study the concept of “ultimate” or “saturation” capacity was applied to the runway approach path in a 3d representation fashion. The Standard Terminal Area Arrivals (STARs) were used as the

actual trajectories flown by aircraft and the minimization of separations between different categories of airplanes, e. g. Light, Medium and Heavy, were taken as variables in the model. These variables were used to perform a sensitivity analysis on the capacity of the terminal airspace of an airport. Three different sets of separations minima were applied to the model and some experiments were run to evaluate the impact of each of them into the final evaluation parameter i.e. the capacity of the system. The results presented showed once more that capacity strongly depends on in-trail aircraft separation. The difference in capacity varied from 20 to 50 %. The utilization of entry gate with respect to the runway in use was found to influence the capacity with a range of variation from 15 to 40%. The model could be used for different purposes such as (i) computation of Terminal Airspace capacity, (ii) optimization of entry gate separations in order to maximize the flow and (iii) the heuristic optimization of terminal approach trajectories.

A paper on modeling issues for a large network of Airports in the national system was presented by Odoni (Odoni 1991). The principal problems taken into account were:

- the problem size and the large amounts of data required;
- the dynamic essence of Demand and Capacity that is difficult to model;
- the large combinatorial number of possible network states and the consequent statistical issues to sample them;
- The level of detail of the model that could be too high or too small;
- The uncertainties on predicting demand for the future scenarios especially from the airports connections point of view; and
- The needs for robustness and portability of the built model.

Only two Simulation models were present: AIRNET and NASPAC. The first was developed for FAA by ATAC Corporation in 1989 and focuses on 63 Airports in the US. This model requires a level of detail that is appropriate for a policy analysis. For example, the model is not concerned with the En-Route segment of the flights using constant travel times on each Origin-Destination pair. NASPAC (National Airspace System Performance Analysis Capability) was developed by MITRE Corporation for the FAA in 1989. The model covers a network of 58 major airports and incorporates also the

En-Route segment of the flights. The model has some microscopic and some macroscopic features. The model needs significant computational resources to run. Both models incorporate large datasets with information about airport characteristics, flight schedules and ATC. The future schedules are produced using simple heuristics to project the actual operations into the future. The models were far to be completely efficient at the time of the study.

An important aspect of the landing operations is the uncertainty related to the rolling time of the Aircraft and consequently on the Runway Occupancy Time (ROT). A study on this problem and the consequent representation through the application of a microcomputer model was presented as a possible tool to predict and simulate the Random behavior of pilot landing on a runway (Trani 1993). The Runway Exit Design Interactive Model (REDIM) has kinematics equations to optimally locate and design high-speed runway exits at airports. The aircraft landing dynamic is used to implement an algorithm to minimize the weighted ROT of an aircraft mix selected by the user. The program is a design and planning tool for analysts and airport engineers. The outputs of the program recommend optimal locations and geometric design for high-speed exits under various airport scenarios, including meteorological conditions, different aircraft mix and so on. Since the releasing of this study REDIM has been used in many airport design projects.

A study on the sequencing problem (Venkatakrishnan 1993) in which three different algorithms were presented to create a decision support tool for Traffic Controllers to better sequence the aircraft in the final approach phase. A case study on Boston Logan Airport was presented based on a large set of data that allowed the authors to extensively validate and calibrate their model. The solutions were presented to real Air Traffic Controllers and the most interesting result was that they were not concerned on the efficiency of the model in terms of delay savings or improved capacity but on the possible augmented workload that they would have to face in an already stressful environment. The possible re-sequencing operations would necessitate an extra effort to control and track the aircraft that they were not willing to perform. The different models

demonstrated an overall savings in terms of delay and operating costs. In terms of Landings Time Interval (LTI) the model showed a benefit up to 30% in the most congested period of the day. The simple basic assumption of the model was that an average arrival rate at a runway is equal to the inverse of the LTI. One important factor affecting the LTI is the number of runways in use. Another important factor affecting LTI is sequencing of aircraft in different weight categories. The models presented need further development to be applied in a real-time air traffic control system.

Another study (Terrab 1993) address the problem of Air Traffic Flow Management (ATFM) from a ground holding viewpoint. A ground hold is the decision to hold an aircraft on the ground when its flight plan is expected to arrive at an airport with a limited capacity due to weather and during high demand conditions. This procedure holds airplanes on the ground rather than in the air. The study presented a case of many flights arriving at an airport from different origins and scheduled to land in a congested peak hour condition. A deterministic and a stochastic model were evaluated. Under certain conditions the former, revealed to be more efficient. The total delay costs were calculated and they were used as a decision support tool to help controllers and traffic managers in their strategic policies. In order to make this tool feasible a precise prediction of weather conditions and of hourly airport capacity would be necessary.

Some Optimization techniques were applied on the Arrival/Departure balance in order to maximize capacity and minimize total delay. In a study by Gilbo (Gilbo 1993) a linear programming technique was applied to solve the capacity problem of an airport to optimally allocate arrivals and departures in a fixed time window. The two operations were considered interdependent and the airport capacities were used as decision variables. Important prerequisites for the efficiency of this model were a set of accurate and realistic capacity curves and their correct application to balance arrivals and departures. The model developed is at an early segment of development to be used as a decision-making support tool. A parameter called  $\alpha$  representing the Arrival/Departure ratio was identified as key for the strategies to be applied in order to maximize the flow.



The development of the model just presented was described in another study (Gilbo 1997) where to the capacity of the runways, was added also the capacity of the arrival fixes as decision variables. The problem of unbalanced demand at the near arrivals fixes is one of the bottlenecks at congested airports. The same parameter  $\alpha$  was used in order to optimally allocate the percent arrival/departure ratio to maximize the airport flow. Different results with different  $\alpha$  values were obtained advising ATC controllers different arrival/departure strategies in congested periods of demand.

A comprehensive report by Odoni et al. (Odoni 1997) describes the capabilities of aviation models. This report summarizes the deficiencies of aviation models available at that time of writing. The report covers different aspects of the air transportation operations e. g. safety, capacity, noise, etc. The objective of the study was to assess the strengths and weaknesses of existing fast-time models and tools for the study of ATM systems and concepts. The report identifies the requirements for future development of additional modeling capabilities. The tools evaluated were fast-time analytical and simulation models: Real-time models were neglected from the study involving man-in-the-loop simulation. Some of the models reviewed were: LMI Runway Capacity Model, FAA Airfield Capacity Model, DELAYS, AND, SIMMOD, RAMS, TAAM, ASCENT, RATSG, MIDAS, ACIM, INM and NOISIM. The report found that, depending on the category of model analyzed, the level of sophistication varied significantly. A ranking was proposed as follows:

1. Capacity and delay models,
2. Conflict generation, detection and resolution models,
3. Human factors and automation models,
4. Cost/benefit models, and
5. Models of strategies and behavior of airlines (airline operations centers) and of other users vis-a-vis ATM.

From the proposed ranking, the most mature models fall in the capacity domain. In fact, as this literary review prove, this branch of air traffic modeling began fifty years ago and have reached an adequate level of maturity and reliability. Different models cover

operational aspects with different level of detail, depending on the needs and capability of the modelers. Nevertheless there are some deficiencies in aviation capacity models the following areas:

- lack of essential features (e.g., Stochasticity);
- lack of flexibility;
- lack of mutual compatibility;
- poor User Graphic Interface (GUI); and
- costly and time-consuming training and experiment preparation.

At the 79<sup>th</sup> Transportation Research Board Annual Meeting a workshop on Airport-Airspace Simulations for Capacity Evaluation was held (Rakas 2001). FAA, EUROCONTROL, and Consulting Firms presented studies about the state-of-the-art simulation applied to airports and airspace. The models presented included new development efforts on SIMMOD, the Airport Machine, DPAT, the LMI Capacity Model, TAAM, and RAMS. Most of the airport capacity models were represented. Dunlay (Dunlay 2001) presented an airspace redesign for the New York Area. The goals of the study were:

- Develop precision approach, departure, and missed approach procedures
- Reduce aircraft delays and flying times with simultaneous operations
- Define requirements for subsequent airspace restructuring
- Provide incentives for technological implementations

The simulation was performed with SIMMOD to redesign existing procedures and to evaluate the impact of these in future operations. Different levels of technological implementation evaluated in this study included: procedures that exploit modern flight management systems, the use of Global Positioning System (GPS) and differential GPS. The procedures described in the paper would act as system enablers that would produce substantial benefits in terms of delay reductions (millions of dollars). The simulation modeling faced many challenges similar to our effort. These challenges are:

- Fleet equipage retrofits,
- Accommodation of partially equipped fleet,
- "Flyability" testing and blunder analysis,

- Development of precision curved and segmented instrument procedures,
- Restructured airspace,
- Changing in Air Traffic Control and procedural requirements and clearance criteria,
- Evaluation of noise impact,
- Accommodation of foreign operators,
- Redefining role of human controller, and
- Pilot and controller acceptance.

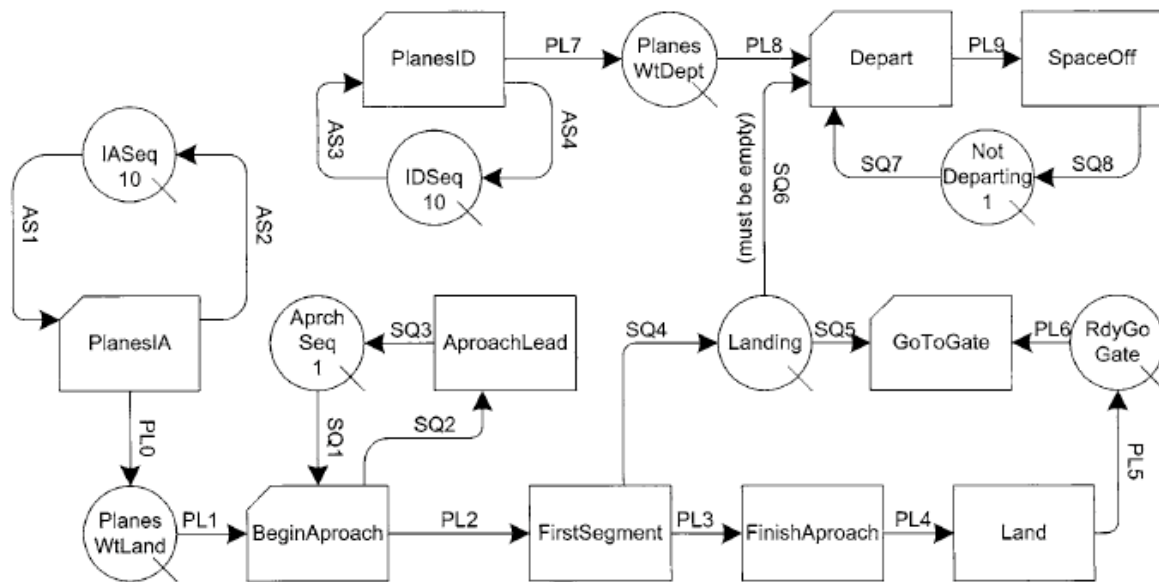
The future operations path proposed in the study is shown in Figure 2.3.



**Figure 2.3 Future Path Proposed in the Study (Dunlay 2001, used with permission of the author).**

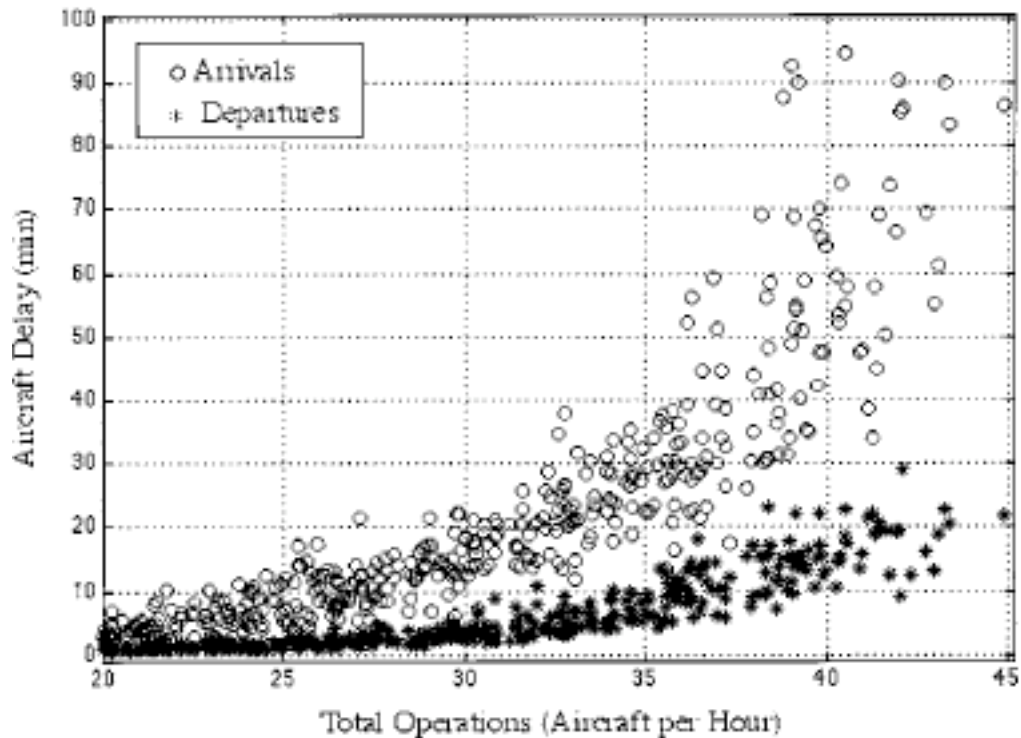
An interesting simulation approach to Airside capacity is presented by Martinez et Al. (Martinez 2001), in this work a general Activity Based Discrete Event Simulation tool (STROBOSCOPE) was applied to the study of the capacity of an Airfield. Resources are the fundamental entities in the model; they represent materials, parts, machines,

equipment, or anything else that can perform a task. For the aviation application, resources are the runways. The model considers entities such as aircraft, signals and arrival schedulers. The modeling framework is presented in figure 2.4 where the main logic of the model is presented without going into details. The diagram includes nodes of various kinds connected with links that indicate how resources flow from node to node.



**Figure 2.4 Activity Based Airport Simulation Model (Martinez 2001, used with permission of the author).**

Ten runs of the model were performed representing ten different days, the output of the model was the average number of Arrivals and Departures per hour, the maximum waiting time, and the average waiting time. The average arrivals waiting time was around 12 seconds, almost five times longer than the departure waiting times. The saturation capacity results were consistent with those calculated with the FAA Airfield Capacity Model for the average hourly capacity with the same aircraft mix. In addition, to the steady state model the one presented was more dynamic predicting aircraft delay under a time varying demand function. Sample results of the simulation are presented in figure 2.5.



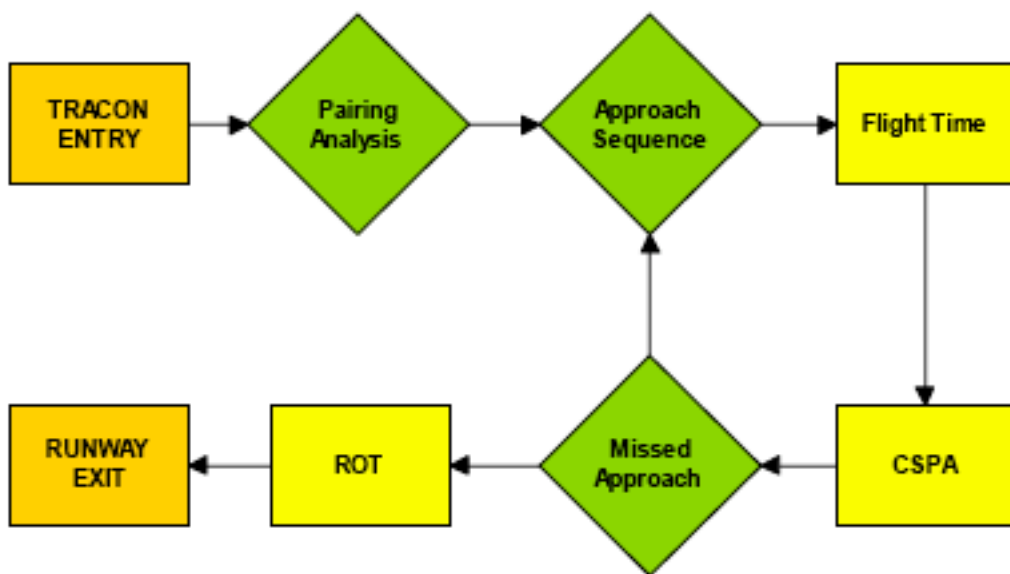
**Figure 2.5 Sample Simulation Results (Martinez 2001, used with permission of the author).**

The model presented in this paper was relatively fast to run differing from other simulation models that usually require long execution times and have a steep learning curve.

A similar approach to the one used by us was utilized by Hoffman (Hoffman 2001) to analyze the maximum throughput capacity at La Guardia Airport. TAAM was used to simulate the airport operations and to evaluate the impact of growing demand in the future. The goal of the study was to identify the desired number of operations per day that the airport can support at a tolerable level of delay. The metrics used in this study were *throughput* and *delay*. Another metric calculated was controller workload. The delays calculated by the model were slightly higher than in reality. Fifteen demand scenarios were simulated up to the unrestricted demand with 1697 operations per day. The results of the simulation runs proved that as soon as the number of arrivals and departures increased from the baseline demand the level, i.e. 1266 flights per day, the delays increase also dramatically. From the results of the study it could be stated that an

optimum level of demand for LGA is around 1260 operations per day, this level of demand was reached already in September 2000.

A simple application of simulation on the Terminal Area (TMA) concept was performed by Gangel et Al. (Gangel 2004) in order to evaluate the impact of a Closely Spaced Parallel Approach (CSPA) in the Air Traffic System. The model assumes a complete implementation of all-weather navigation systems such as ILS Cat III B for all aircraft. A time-based simulation and a statistical analysis model were used to evaluate the effectiveness of the CSPA concept. The conceptual model is presented in figure.2.6. The model starts at the TRACON entry nodes and simulates the flights until the touchdown on the runways.



**Figure 2.6 Logical Block Graph of the Simulation Model (Gangel 2004, used with permission of the author).**

The simulation was performed in a discrete event simulation package called ARENA whereas MATLAB was used for the statistical analysis. Two Airports were selected for the simulation Atlanta Hartsfield International Airport (ATL) and San Francisco International Airport (SFO) under bad weather conditions. These airports have high delays and substantial capacity constraints. The study concludes that CSPA operations would not result in additional workload to the TRACON controllers. It is claimed that after

a period of familiarization with the new procedures, these could eventually double the capacity of the major hubs in the United States. However an increased number of operations could shift the constraints of the system from the physical infrastructure to the workload of TMA controllers.

In another study, Janic (Janic 2004) evaluates the future impact of runway capacity improvements from a more economical point of view. The comparison between future demand and capacity is made by extrapolation of historical data. The case study of London Heathrow, one of the busiest hub in Europe, was presented to underline the risk of late actions that in the future would be economically costly from a delay and Level of Service viewpoint. Two sets of future demands were created. The first one is an “optimistic” or business-as-usual and the second set so-called “pessimistic”. The first set was based on the assumption that the economy would keep growing following past trends. The second scenario was based on reductions of future demand growth. In order to study these future demand sets, four different capacity scenarios were created based on four different development plans for Heathrow Airport. It must be noted that London Heathrow Airport operates at reduced capacity for 7 hours of a typical day due to Noise Abatement Procedures. The airport operates segregated operations to two parallel runways that, due to spacing, could support mixed operations.

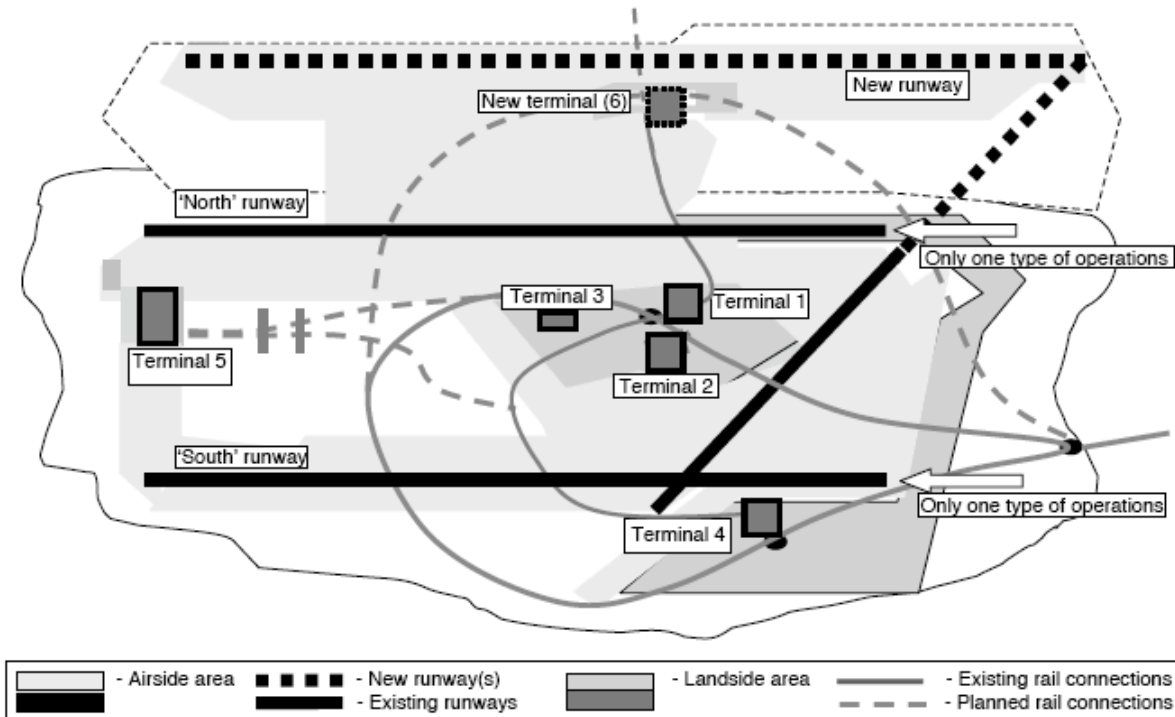
- Solution 0, do-nothing.
- Solution 1, extend airport daily operating time
- Solution 2, changing runway operating mode and extend airport daily operating times
- Solution 3, building a new runway and extend airport daily operating times

Moreover two passenger terminal capacity scenarios were added to the future options:

- Scenario 1, building new passenger terminal within the existing airport area
- Scenario 2, building a passenger terminal and new runway

These scenarios were mixed together and in six different combinations studied. As expected the do-nothing scenario would eventually result in severe delays. The pessimistic demand could be easily faced with the augmented capacity measures whereas the optimistic demand, from a passenger terminal capacity viewpoint, could not

be completely satisfied with the solution proposed. Figure 2.8 shows the existing and prospective airside and landside solutions.



**Figure 2.7 Existing and Prospective Airport Layout (Janic 2004, used with permission of the author).**

A paper on airport decision support system (Stamatopoulos 2004) investigates the feasibility of creating models for the estimation of the landside capacity of an airport and the delays associated with its operations. The final goal was to develop a decision support tool for airport planning from a strategic viewpoint with limited input requirements and in a relative small time. The starting point of the study is to fill a gap in the airport landside modeling at the mesoscopic level of detail. A model called MACAD (MANTEA Airfield Capacity And Delays model) was proposed in order to address this requirement. The model main attributes are its computational speed, flexibility and the relative ease to execute it.



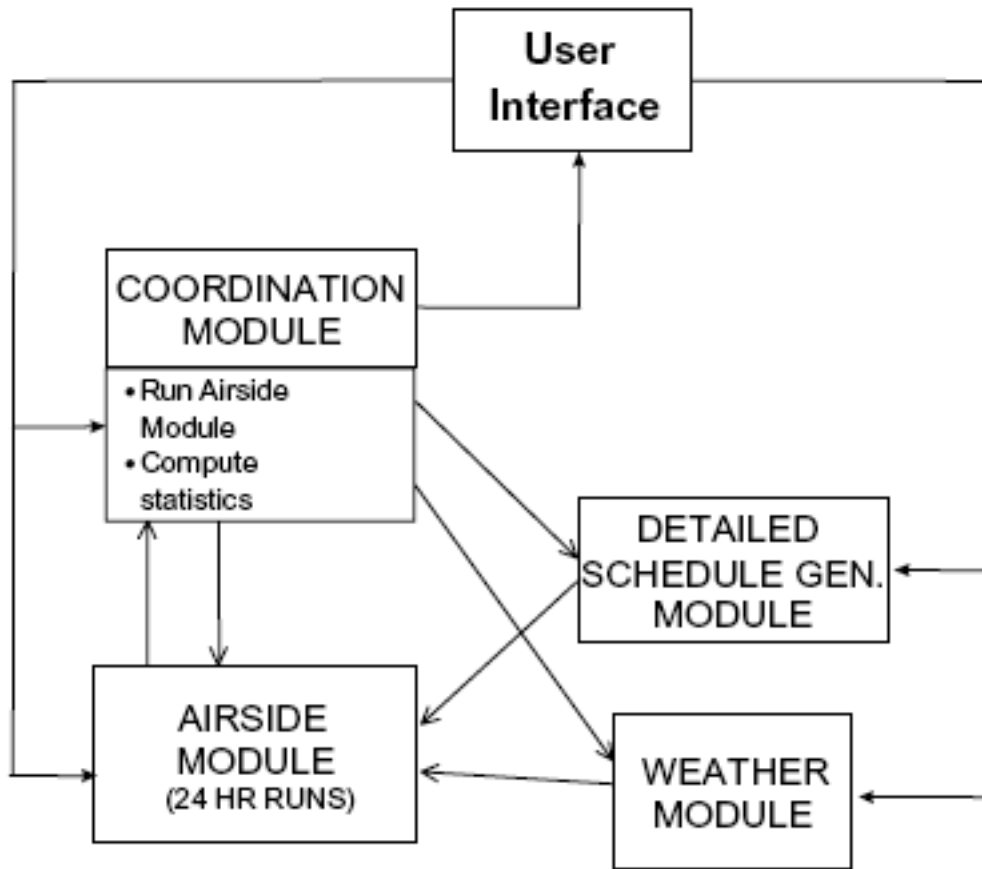


Figure 2.8 The Integrated System of Models (Stamatopoulos 2004, used with permission of the author).

The MACAD model comprises five modules:

- The *Co-Ordination* module, coordinates the sequence of other modules,
- The *Airside* module, computes the capacity, principal component of the system,
- The *Weather* module, combines the weather conditions with the runway capacity,
- The *Detailed Schedule Generation* module, generates detailed schedules when not available, and
- The User Interface, a Windows-based user-friendly interface.

The model proposed was based on the single-runway model developed at LMI, but the methodology has been extended to two-runway configurations. Moreover some important differences exist between the two models especially in the way they estimate delays. It is reported that the logic of the MACAD is simple. In order to compute delays MACAD uses an analytical queuing model based on DELAYS, a tool to numerically

compute delays, without the use of discrete-event simulation. MACAD was evaluated and validated at Rome's Fiumicino Airport, one of the busiest in Europe, giving results in terms of delay very close to the observed values.

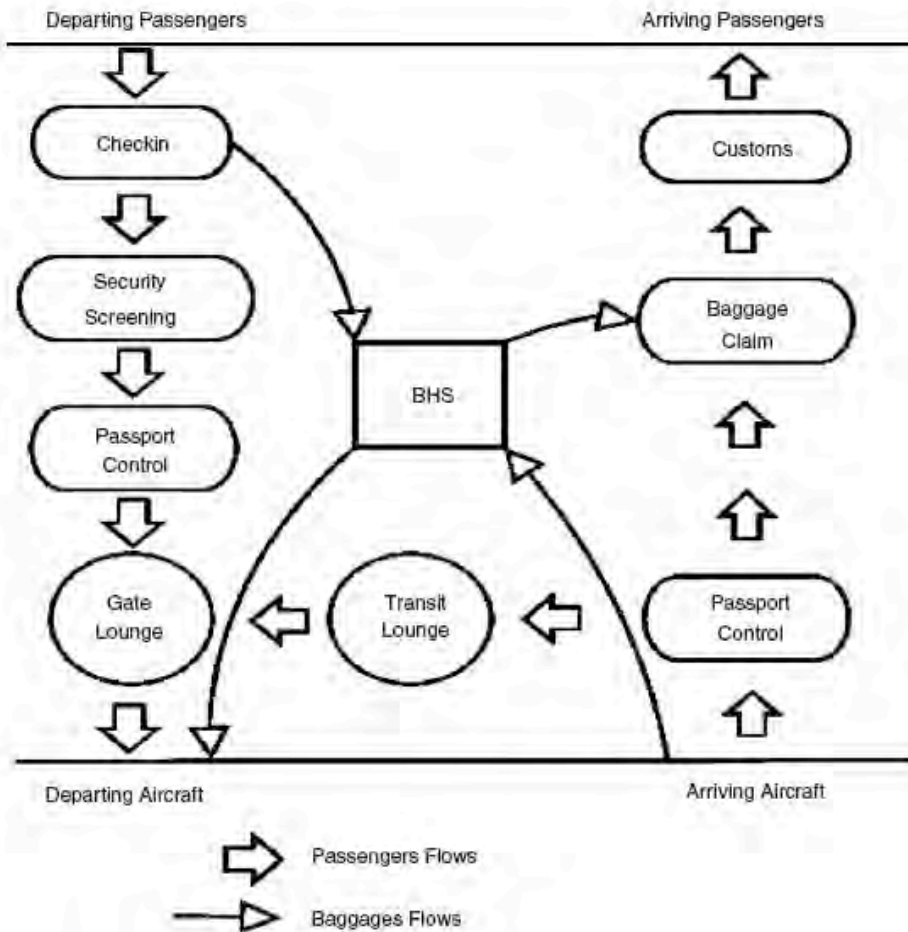
A recent study (Swedish 2005) describes a new simulation model developed by MITRE Corporation called Airport Capacity Analysis Through Simulation (ACATS). The model is easy to use, fast and can model future ATC separations. At the core of ACATS there is a simulation engine that is common for all the Airports driven by data representing ATC rules, runway layout, and demand characteristics. The outputs of the model consist of a graphical animation of the scenario, statistics about the throughput capacity, and some analysis charts. The graphical user interface (GUI), figure 2.10, is particularly simple especially if compared with other Airports simulation models like RAMS, TAAM or SIMMOD. The main data required by the model are:

- Runway Layout,
- Wake Vortex Separation Requirement,
- Fleet Mix,
- Aircraft Performance,
- Arrival-Departure Ratio,
- ATC Separation Rules, and
- Simulation Control Attributes.

The ACATS algorithm is relatively simple, in fact it repeatedly performs the same functions, i.e. generating a set of aircraft waiting for selection, next it selects the next aircraft to arrive or depart on a first-come, first-served basis. Then it calculates the aircraft trajectory checking the interactions with the other operations. The process is then repeated for other flights as they are added to the simulation. The ACATS model could compute both capacity and delays. When this model was applied to complex airport configurations it gave more accurate results than a steady state model such as the Enhanced Airfield Capacity Model (EACM) (Swedish 1981).

Another aspect of airport capacity that has been studied through simulation is the landside of the Airport. Because this topic is outside the scope of our analysis we provide only one case study cited in the literature that is considered interesting.

In this study (Andreatta 2005) the performance of the landside of Athens International Airport (AIA) landside was tested with a simulation tool called the Simple Landside Aggregate Model (SLAM) during a particular period during the 2004 Olympic Games. The unusual demand generated by such an event could bring any well-designed facility to its saturation level and produce unacceptable Level of Service (LOS). Due to its computational speed and the relative ease of use SLAM can be considered a strategic model of landside operations. The objective of the model is not to provide a detailed analysis of the facility but to roughly estimate the throughput of the system in terms of passengers per hour and the delays suffered by departing flights. The airport landside was modeled taking into account only functional components, i.e., elements providing services directly related to passengers boarding and un-boarding an airplane including the baggage system that is often the cause of delays for traveling passengers.



**Figure 2.9 The Model for a Generic Airport (Andreatta 2005, used with permission of the author).**

SLAM was tested combined with MACAD (Stamatopoulos 2004) under three following scenarios:

- Baseline Scenario, a regular peak day (529 movements and 60,205 passengers)
- Hubbing Scenario, (562 movements and 63,610 passengers)
- Olympic Games Scenario, (654 movements and 73,188 passengers)

Based on MACAD statistics the Airside of Athens Airport performed well with low delays during the most congested periods. The SLAM analysis predicted some delays between the check-in and the gates that at AIA between 10 and 25 minutes. For this reason, almost all the flights were delayed at least of one minute. Moreover in the last scenario studied, the simulation predicted delays due to the baggage handling system. Overall the model evaluated the LOS of Athens Airport as being excellent under all three scenarios investigated.

### **2.3. *Controller Pilot Data-Link Studies***

A detailed study on the New York Metropolitan Area Air Traffic Capacity was conducted by the FAA (Faison 1970) to evaluate potential problems of one of the most congested areas in the country. In this study, the underlying conclusion was that also at that time the most limiting factor to constrain growth was controller workload and not inadequate airspace or runway system capacity. The evaluation was carried out using simulation, ten different scenarios were evaluated to match capacity and demand in both VFR and IFR conditions. The methodology applied in the study was the “Airport Capacity Criteria used in preparing the National Airspace Plan” of the FAA. Workload was one of the metrics evaluated in the study. A precursor of workload was the communication time per hour per aircraft. Inputs to calculate Workload were created using theoretical and empirical probability theory, queuing theory and aircraft dynamics. It was found that none of the en-route sectors studied approached the “physical capacity” at the demand level applied in the study. With the implementation of future en-route system tools the complexity was reduced consequently adding more capacity to the system. A new runway at EWR was also added increasing the Terminal Area capacity. It must be noted that of the proposed new runways only runway 22R/4L at Newark and the lengthening of runway 22L/4R at JFK were actually implemented, causing the results of the study to be optimistic.

A man-in-the-loop simulation study (FAA 1995) using two-way Data Link ATC communications was conducted by the Federal Aviation Administration (FAA). This study addressed two operational en-route ATC problems that have been attributed to limitations of the communications capability of controllers using existing voice radio system. Air traffic controllers, ATC supervisors and professional pilots participated in these high fidelity simulation tests in which a combined Data Link and voice radio communication system was used to control traffic in the en-route airspace. Baseline test scenarios were built to precisely duplicate air traffic sample days taken from two

airspace sectors within the Atlanta ARTCC. Two separate experiments were carried out to address two separate operational en-route ATC problems: one modeling an arrival sector and the other set to model a departure sector. In both cases, delay data experienced by the traffic on the sample day was collected using the System Analysis Recording (SAR) tapes. In both experiments additional test runs were done with increased traffic levels representative of future demands. For all cases a comparison was made between voice only radio communication and Data Link and voice radio communications. The aircraft entering a departure airspace sector are separated by a miles-in-trail (MIT) restriction which usually contribute to ground delays. The first experiment was designed to determine whether the MIT restrictions could be relaxed with the introduction of Data Link communications and tested the ability of Two-way Data Link ATC communications to reduce airport departure delays that are caused by capacity problems in a high altitude en route departure sector. The set of experiments conducted were based on sample of air traffic for sector 32 of the Atlanta ARTCC. The baseline scenario was run using the routine 20 MIT restrictions for departures entering the sector using the same air traffic as those on a sample day SAR tape. Further runs were made after decreasing the MIT restriction to 15, 10 and finally without any restriction (minimum 5 mile separation). A final test run was made with no MIT restriction but with a 10 percent increase in traffic demand to simulate future demand. Aircraft arrival times and positions at the sector boundary for each of the above scenarios were obtained using SIMMOD. SIMMOD was primarily used to calculate benefits like ground delay and was also used to assess the impact on total delays of increasing the departure traffic demand. Some of the findings of the experiment were that the controllers and ATC supervisors were able to handle the air traffic safely and effectively at each reduced MIT restriction and also with a 10 percent increase in air traffic using a combination of Data Link and voice radio communication. No aircraft separation violations were detected hence indicating that safety of ATC operations was in no way compromised. Total delays for all departing aircraft decreased from 1,795 minutes with 20 MIT restriction imposed at Atlanta to 687 minutes when the use of Data Link permitted the elimination of restrictions. Comparable savings were also obtained with the 10 percent increase in traffic.

In some situations, it is seen that in an en route sector, the voice communication channel gets saturated causing delay to airplanes passing through the sector and hence the arrival of these planes to an airport gets delayed. Due to voice frequency congestion, controllers are often unable to adequately perform and meet restrictions minima for airport arrivals. The second experiment in the study tested the ability of the Data Link to improve air traffic throughput in an en-route sector where saturation is responsible for inefficient processing of aircraft arriving at an airport. The scenario for this experiment was based on sample of air traffic derived from sector 9 in area 5 of the Atlanta ARTCC. The test runs were made using the arrival and over flight traffic demand recorded on sample day morning peak period for sector 9 at the Atlanta ARTCC. Four more runs were made, each time increasing the traffic demand by about 10 percent. In this case SIMMOD was used as the analytical tool to estimate the throughput and efficiency for the baseline case as well as for the increased traffic demand using only voice radio communications and was compared against the data produced by the live simulation with Data Link.

Some of the conclusions of the FAA Data Link study for the en route scenario were:

- Data Link enabled the elimination of spacing restrictions that are enforced to prevent saturation in a departure sector. This resulted in a 62 percent reduction in ground delays for departing aircraft,
- Average flight times and distances were reduced by approximately 20 percent in both departure and arrival sectors,
- Data Link reduced frequency congestion making the voice radio more available for time critical clearance delivery, and
- The benefits estimated at the NAS level annually were approximated to be \$337 million with the introduction of two-way Data Link ATC communications.

A second man-in-the-loop study (FAA 1996) was conducted to identify and quantify some of the benefits of CPDLC in terminal airspace. The study examined operational ATC performance within the Newark area of the New York Terminal Radar Approach Control (TRACON) and, for the subject of our research effort it was particularly interesting. Three experiments were conducted separately and all were performed using

a case study methodology. Test scenarios were built to duplicate air traffic data sample periods taken from the Newark area.

In experiment 1 the operational baseline data was compared to data from live simulations of ATC operations using a combined voice and Data Link communication system. For experiment 2 the number of flights were increased by 10 from the original 55 flights scenario to increase the demand on ATC resources. Experiment 3 examined the impact of Data Link on the Newark area satellite arrival position (MUGZY). The experiment was conducted on complex airspace that surround EWR, with crossing traffic from LGA, JFK and TEB and tested whether the addition of Data Link communications would improve the processing of aircraft arrivals. Results of experiment 1 showed that the average flight arrived at the airspace boundary 1.98 minutes sooner and the mean flight distance and times in the terminal airspace were reduced by an average of 6 miles and 3 minutes, respectively. No holding of aircraft was required that had occurred on the operational baseline day. The productivity of the sector improved as the average number of flights handled by controllers during the test period increased by two and the average number of arrivals increased by four. In experiment 2 entry sector times were reduced by an average of 1.36 minutes and the mean flight distance and times reduced by 1.7 miles and 0.7 minutes, respectively. Holding was required but to a much lesser extent as compared to the operational baseline day. Experiment 3 failed to yield any significant benefits of CPDLC. Small improvements in sector performance were obtained in some test runs while, as others showed no benefit. Some of the conclusions of the FAA Data Link study for the terminal area are:

- CPDLC would provide significant benefits when implemented in terminal ATC environments by increasing the arrival rate, reducing flight delays, improve airspace productivity, enhance safety and reduce controller workload,
- It would reduce the number of voice messages sent by controllers by an average of 45 to 66 percent, and
- Reduction in annual operational costs to NAS users would be around \$152 million.

A study by Rodgers (Rodgers 1997) developed a program called NASSIM, National Airspace System Simulation. This study evaluated the benefits of Data-Link in an en-



route environment and to assess that benefits connected to different equipage rates on different FAA facilities. A discrete event simulation model was built to simulate transmission of communication messages through a voice channel and Data-Link. The model was used to analyze the impacts of two different Data Link message sets and four different Data Link equipage rates (25%, 50%, 75% and 90%) against a baseline scenario of only voice radio communications. Two message sets were used, A and B with different level of details. The model was applied at three different en-route ARTCC facilities including Atlanta, Denver and Los Angeles. The study demonstrated that with this equipment improvement the communication utilization and voice channel occupancy would decrease significantly (e.g. 91% in some Denver sectors). Since the Atlanta ARTCC is busier than the one in Denver and the benefits there were greater, the study concluded that the more congested the airspace is, the more benefits could be gained with Data-Link.

A series of studies were conducted by the FAA to improve the Human Computer Interface (HCI) air traffic procedures and to train for the Controller Pilot Data-Link Communication (CPDLC). The first (Darby 1999) was performed to evaluate the Display System Replacement (DSR) HCI and to obtain controller opinions on the design of route assignment and downlink services needed to complete phase IA of the CPDLC Program. Eight Air Traffic Controllers were involved in the study taking part on a high-fidelity simulation to begin practice with the DSR and HCI. Controllers were also active part in the route assignment services. Some of the design improvements proposed were:

- A better interaction between controllers should be enabled by the Data-Link settings HCI.
- Some change in the symbols of the Full Data Block (FDB) and of an ongoing transfer in order to avoid confusion.
- The location of two Data-Link keyboard keys should be changed to improve accessibility.

The second study (Darby 2000) was conducted to obtain a final review of the CPDLC I

HCI and to review the HCI functionality for services that would be eventually added to phase IA. Procedures for CPDLC I failures and other atypical events were also evaluated. Six en-route Controllers were recruited from the Miami ARTCC in order to evaluate the system and its procedures. The controllers found the HCI and the functionality for CPDLC I acceptable.

Some of the Controllers' suggestions for the CPDLC IA were:

- To add some functionality allowing the controller to view the menu text and status list in a fully unfiltered form
- To implement the status list and menu text lists as DSR views.

Another CPDLC study (Ferra 2000) was conducted by the FAA to evaluate the functionality of the CPDLC test bed facilities like the DSR, the Host Computer System (HCS), the Target Generation Facility (TGF), and the Engineer Research Simulator (ERS). Seven test cases simulating realistic air traffic scenarios like 'Mistuned Frequency' were implemented to evaluate the impact of selected user errors and system induced CPDLC events. Feedback from controllers, pilots and expert observers provided the effectiveness and performance of the procedures and of the facilities. The study claimed that the exchange of data between TGF and HCS/DSR was reliable and effective demonstrating the adequate performance of the test facilities. The human factor aspect of the procedures was also evaluated and tested.

The third study by the FAA (Darby 2001) on CPDLC I was conducted to assess the readiness and operational acceptability of the FAA system and the ARINC sub-network. It was simulated the ATN and VDL-2 sub-network using a production HCS software release and prototype DLAP release. Another phase of the study would update the DLAP to a production release and simulate the ATN and VDL-2 sub-network in a complete end-to-end environment. The objectives of the study were to resolve five Critical Operational Issues (COI):

- Can CPDLC be used without degradation of the ATC operations?
- Does CPDLC maintain the current level of efficiency and accuracy in the controller-pilot communications?

- Are the CPDLC time performance adequate for the requested communications?
- Does CPDLC support effectively air traffic operations?
- Is sufficient training provided to the users (controllers and pilots) to adequately use the system?

Eight ATC controllers from the Miami ARTCC took part in the experiment. A full-scale simulation was run and they evaluated the training program, HCI and CPDLC procedures. Some of the conclusions were:

- CPDLC did not increase controller workload and did not degrade the ATC operations,
- Controllers claimed that CPDLC maintained at least the current level of precision and accuracy in the ATC communications,
- The HCI was found to effectively support ATC however there were some concerns that need to be solved, and
- Controller performance indicated that training was sufficient to allow controllers to effectively operate the system.

A piloted simulation (Waller 1989) was performed to determine the benefits of using digital data link for the transmission of ATC instructions. Instructions like altitude, airspeed, heading, radio frequency and route assignment data were selected for the study. Aircraft data link communications were integrated with flight management functions of the aircraft. The conclusion of the study was that a reduced time (25 percent) to process ATC data that arrived in the flight deck was achieved.

A study by Massimini (Massimini 2000) proposed a method to use Total Airspace and Airport Modeler (TAAM) to measure the voice channel occupancy in a sector manipulating the simulator output. This method could be applied to evaluate the impact of CPDLC in the future scenarios. Every event reported by TAAM in a sector has an associated a communication time and voice channel occupancy time. The next step would be to compare the results obtained by TAAM with the one obtained with the human-in-the-loop simulation performed by the FAA. Some calibration would

be necessary if the results obtained by TAAM were less in terms of communications. After determining the exact nature of the sector that will simulate exactly the operations with CPDLC, a new run could be made to evaluate the system benefits such as reduction of delays and distance traveled.

A study by Ryan (Ryan 1992) suggested that with a voice message system that would rely 90 percent on data link and only 10 percent of voice communications, airlines and FAA could achieve significant operational benefits.

Riley (Riley 1992) conducted a study where the Function Allocation Issues and Trade-offs (FAIT) methodology developed at Honeywell Inc. was applied to identify potential human factor issues and requirement areas associated with air to ground Data Link. Based on the FAIT analysis, 48 significant human factor issues and 119 requirement areas were identified. The human factor issue areas were:

- Sources and effects of delays
- Pilot and controller situational awareness
- Crew resource management
- Pilot and controller workload
- Some of the human factors requirements areas were:
  - Human-computer interface design
  - Crew Coordination
  - Sector workload

A recent human in the loop study (Smith 2004) was performed at NASA Ames Research Center. A series of air ground simulations were performed to evaluate the possible impact on workload of the concept of air-ground trajectory negotiation. The concept was evaluated as part of the Distributed Air-Ground Trajectory Negotiation Project, applying new technologies as CPDLC and ATC decision support tools to accommodate user preferred trajectories. Two human in the loop simulations were carried on in 2002 and 2003 with different objectives. The first focused on how through integration of air and

ground side decision support tools with data link could potentially improve efficiency, capacity, and workload distribution. The second focused on the interactions between pilot and controllers during trajectory negotiation. In the first simulation, the Concept Element 6 (CE-6) with Cockpit Displays of Traffic Information (CDTI) with conflict detection and resolution (CD&R) logic and Required Time of Arrival (RTA) capability was made available to pilots. For controllers they used Center-TRACON Automation System (CTAS) Traffic Management Advisor (TMA). General benefits in efficiency and capacity without compromising safety or significantly increasing workload were obtained. Under CE-6 aircraft flew more efficient paths at higher altitudes over a shorter period of time. A 5.5% increase over the baseline meter fix throughput was obtained (from 41.6 to 43.9 aircraft per hour). All the controllers, pilot and expert involved in the experiment witnessed acceptable levels of mental workload. Temporal demand and situation awareness were good. In the second simulation an improved controller Multi-Aircraft Control System with Display System Replacement (MACS/DSR) display and 3-D cockpit displays were used. The controller display incorporated improved DSTs including a better route planning tool, speed advisories, and the Transfer Of Communication (TOC) operations found in CPDLC I. Flight deck tools were also updated with better conflict detection capabilities and optional 3-d views that could be rotated to view from any angle. Overall, the results suggested that pilot-controller negotiation of trajectory changes are operationally feasible. En-route controllers approved most of the initial pilot requests and many of the remaining requests were also subsequently approved. The procedures were found to be simple and adequate, the controllers' only concern was about the potential adverse impact of silent ATC-Pilot communications. Under these conditions pilot unaware of the controller's CPDLC interactions with other pilots may overload a controller with requests.

A recent study (Prutzman 2007) analyzed the impact of the Tower Data Link System (TDLS) applied to 71 airport across the NAS. The study derived benefits and divided them into two categories, i.e. Controller Workforce and Stakeholder/Users, the salient results of the study are:

- Controller workload efficiency using tower ground control position salaries results

in approximately \$122,000 per year per site efficiency value totaling \$9M per year in controller workload avoidance for all the sites.

- For the Future Year (FY) 08 through the FY13 investment profile of 6.5M, the FAA workforce efficiency gain would total \$54.2M.
- The TDLS program would invest \$0.12 for every dollar of workforce efficiency gain achieved from FY08 through FY13 included in this investment request.
- The investment service unit cost per operation is less than \$0.03 per year, thereafter less than 1 cent per operation cost indefinitely for recurring sustainment for the life of the program.
- Compared to the average annual investment FAA will make of \$1.09M over the next 6 years, the airlines will invest over 4\$ million each year to subscribe for the service.
- The effect of this relationship is a benefit to users of cost savings and avoidance (derived through taxi/push back delays and efficiencies) of about \$55,013 per operation.

Some of the conclusions of the study should be proven more extensively but the trend showed was interesting.

## **2.4. *Controller Workload Studies***

In order to evaluate the impact of a new route structure around the New Tokyo International Airport and the Tokyo International Airport (TIA) a study (Tofukuji 1993) was carried on the controller workload levels. Both the en-route and terminal airspace capacity were considered in this research. They are usually evaluated as the number of aircraft that can be handled in a unit of time constrained by controller workload of the controllers. The Relative Capacity Estimating Process (RECEP) was used in order to perform the simulations and to evaluate workload. This quantitative method measures all the controller's countable activities. In the RECEP model the capacity is defined as the number of aircraft per hour, which corresponds to an upper limit in the controller's workload, that was found to be 48 man-minutes per hour for a radar controller and 66

man-minutes for a combined radar/manual controller team in several control centers evaluated at the Los Angeles Center. Different experiments were performed with different traffic loads to analyze trajectory data, controller's input, communications between controllers and pilots that could be used as drivers of controller's workload. In the end a correlation analysis of trajectory data and a regression analysis between the number of aircraft handled per hour and the frequency of the controller's intervention in the traffic flow. The result of the regression analysis gave a linear equation to represent the relationship between the number of aircraft and the frequency of intervention by the controllers. Seven trials were performed for different sectors with very high correlation coefficient all the times always explaining very well the variability of the model. One of the most affective operation for sector capacity was the transfer of control and consequently a reduction in the available airspace at the workload limit.

A European study (Majumdar 2001) used the Reorganized ATC Mathematical Simulator (RAMS) as a tool to evaluate the workload of air traffic controllers and to estimate the capacity of European airspace. The major problem in the European airspace is the lack of integration and harmonization on ATC services. Other implications of these factors are:

- A rise in flight delays in Europe,
- Nonoptimal flight profiles,
- Extra route lengths, and
- Possible safety issues.

This research effort provided a method to determine Europe's airspace capacity using a simulation model of air traffic controllers' workload. It was claimed that a measure for airspace capacity should be the controller workload especially for high air-traffic-density areas. Assuming this, the capacity of an ATC sector can be defined as the maximum number of aircraft that are controlled in a particular sector in a specified period while still permitting an acceptable level of controller workload. A definition of workload was given as "a process or experience that cannot be seen directly but must be inferred from what can be seen or measured". The basis to evaluate workload was the counting of tasks

both physical and mental, and their timing that the controller must do to control air traffic.

Two ways could be used to do so:

- Self-assessment of the controllers, either by instantaneous techniques or by non-instantaneous techniques (e.g., the NASA Task Load Index, NASA-TLX) method.
- Direct observation of the controllers by other controllers or by ATC system experts.

Based on a time-task definition, threshold controller loadings are defined for the number of minutes per hour controllers are occupied in their tasks as recorded by the models.

Workload is affected by a complex interaction of:

- The situation in the airspace
- The state of the equipment, and
- The state of the controller

The method proposed to estimate Airspace capacity followed these steps:

1. Define the physical characteristics of the air traffic system
2. Define the traffic demand through the sectors
3. Define the rules for assessing capacity of an individual ATC sector for use in the simulation model of air traffic controller workload
4. From the output of the runs of the simulation model, undertake statistical analysis to derive a functional relationship between airspace capacity and the airspace capacity drivers.

In order to evaluate workload based on tasks, the Reorganized ATC Mathematical Simulator (RAMS) was used. The controller tasks from the Bordeaux (Southwest France) Area Air Traffic Control Center were used in the simulation. The task base was divided into five categories:

1. Internal and external coordination tasks,
2. Flight data management tasks,
3. Radio or telephone communications tasks,
4. Conflict planning and resolution tasks, and
5. Radar tasks.

To model the ATC control process realistically, entry distributions were attributed to each controller that specify a time offset before the sector pierce in which the flight is



received into the controller's list. The distributions provided stochastic randomness and uncertainty mirrored in the real ATC environment.

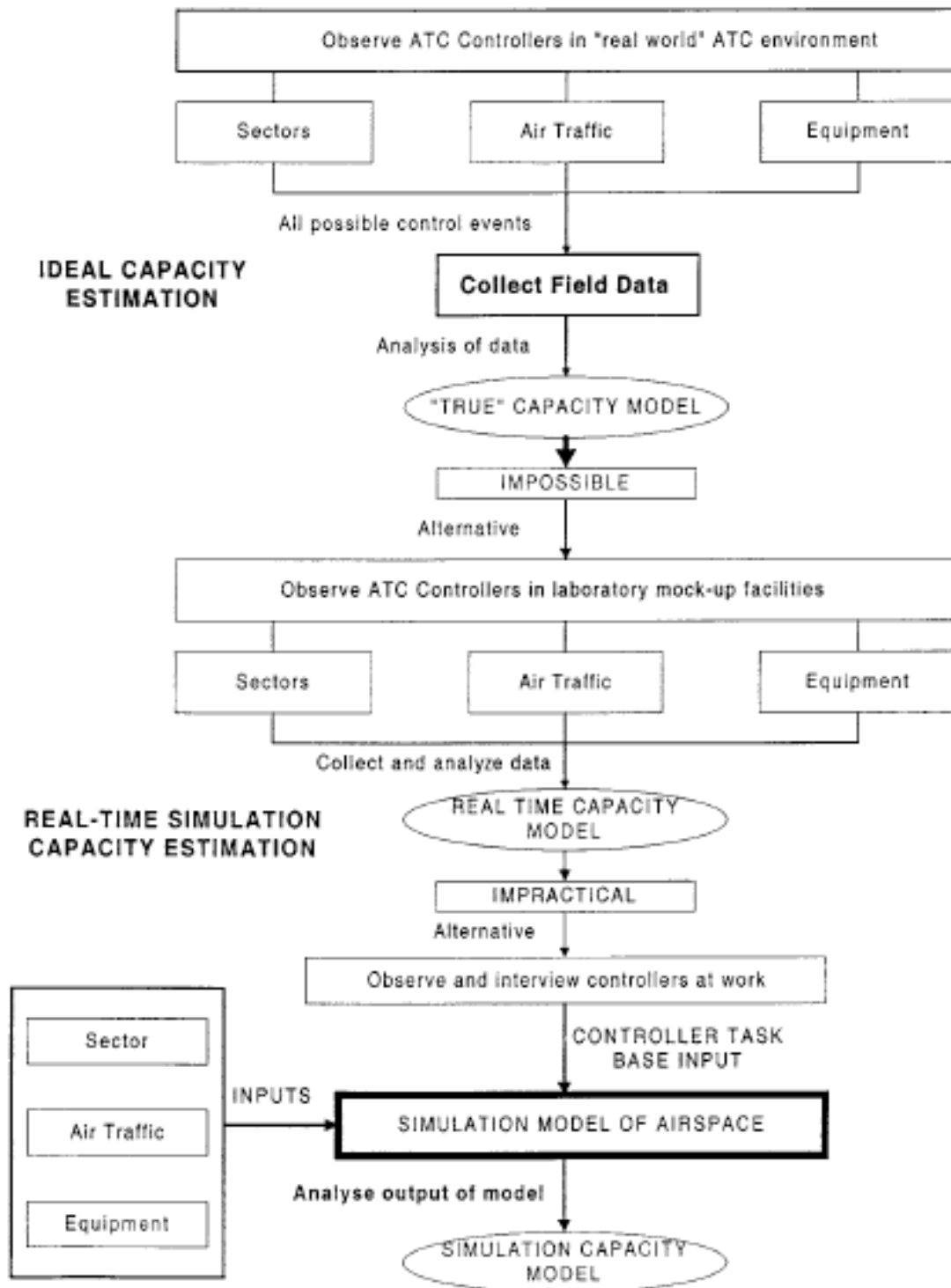


Figure 2.10 Flow Chart of the Study Process (Majumdar 2001, used with permission of the author).

In order to formulate a functional relationship between controller workload and appropriate air traffic and sector variables the model shown in Fig. 2.12 was used, with a series of possible regressors obtained from the RAMS simulation. The only five variables found to be significant at the 5% level of significance were:

- Number of Aircraft in cruise,
- Number of Aircraft in climb,
- Number of Aircraft in descend,
- Total flight time, and
- Average flight time.

Notice that these are all traffic related and not sector related variables.

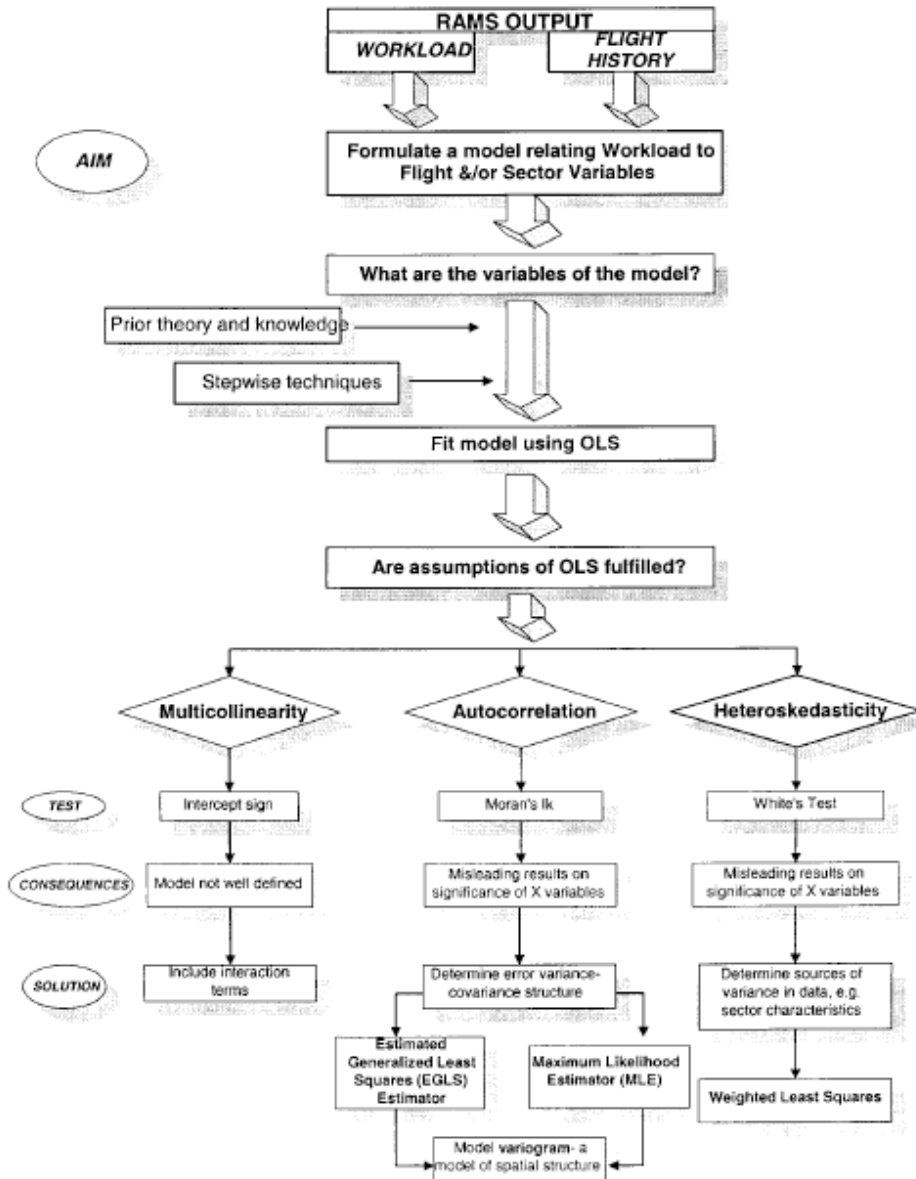


Figure 2.11 Flow Chart of the Study (Majumdar 2001, used with permission of the author).

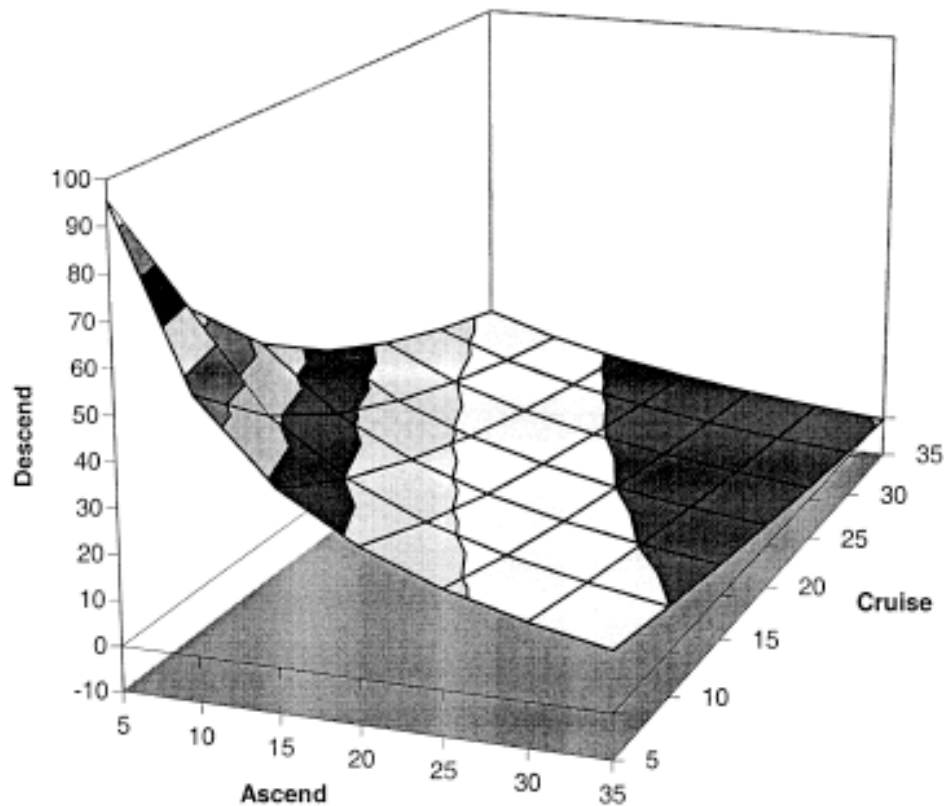
It was stated that the workload at capacity is equal to 42 minutes (2,520 s). A sample workload equation derived in the study is:

**Equation 1**

$$W = 193.92 + 45.23(\text{Cruise}) + 46.2(\text{Ascend}) + 3.82[(\text{Descend})(\text{Cruise})] + 1.56[(\text{Ascend})(\text{Cruise})] + 5.34[(\text{Descend})(\text{Cruise})]$$

Where: W=Total Acceptable Workload (s), Cruise=Total Time Aircraft Cruising (s), Ascend=Total Time Aircraft Climbing (s), Descend=Total Time Aircraft Descending (s).

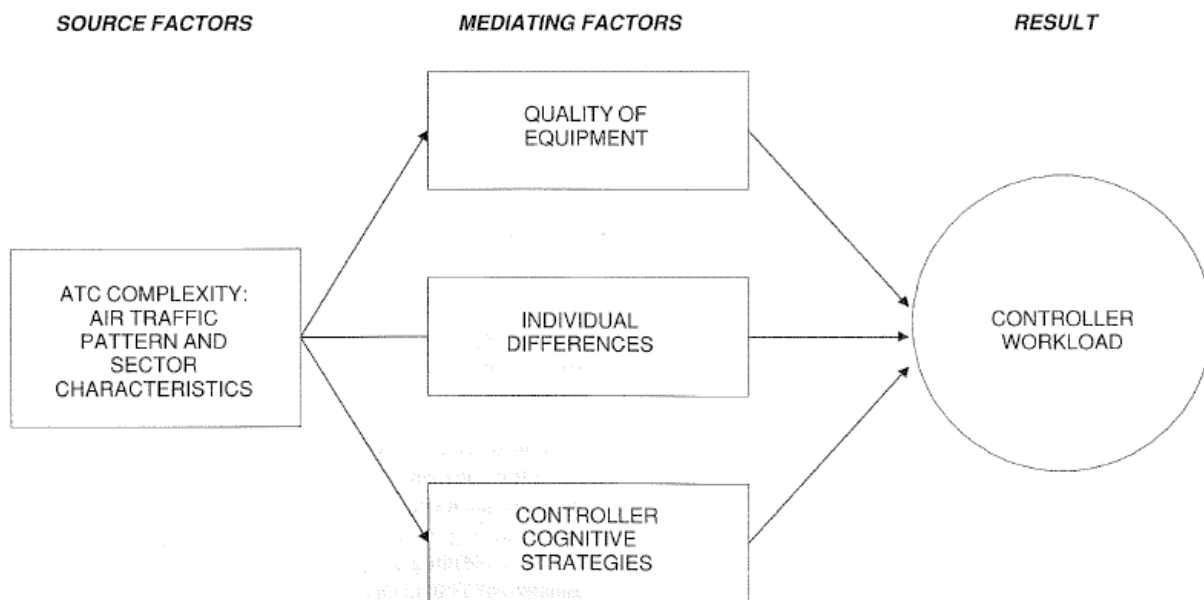
Plotting the results in a 3D curve for all the possible combinations of points in their various altitudes were obtained based on equation one corresponding to the controllers at capacity.



**Figure 2.12 Results in a 3d Curve Representation (Majumdar 2001, used with permission of the author).**

The analysis framework was applied to the EDDYLINO sector in the Maastricht airspace showing that the sector is approaching capacity using existing control technology and procedures.

Another study from the same author (Majumdar 2002) used another model, the European Airspace Model (EAM), to evaluate the factors affecting the ATC Controllers workload, the case studies were from five very different areas in the European Airspace. Using the same airspace capacity definition of the previous study a model for the factors affecting it was presented.



**Figure 2.13 Factors Influencing Controllers Workload (Majumdar 2002, used with permission of the author).**

It was found that the situation in the airspace is the primary factor affecting workload determined by:

- Physical aspects of the sector (e.g., size or airway configuration);
- Factors relating to the movement of air traffic through the airspace (e.g., the number of climbing and descending flights); and
- A combination of the above factors, which covers both sectors and traffic issues (e.g., required procedures and functions).

Some of the secondary factors founds are:

- The cognitive strategies the controller uses to process air traffic information (i.e., the state of the controller);
- The quality of the equipment, including the computer-human interface (i.e., the state of the equipment); and
- Individual differences such as age and amount of experience (i.e., the state of the controller).

A variety of multivariate analytical techniques were then applied using the EAM model, this model, like RAMS, assigns two controllers to each sector which is a three dimensional volume of airspace. The simulation model, through a series of inputs

including controller rules, permits to evaluate the workload levels for the specified sector. More than fifty sectors from five en-route area control centers were used for the experiment. Three type of analysis were carried out on sector-related and aircraft data on the peak hour:

- Principal components analysis,
- Factor analysis, and
- Regression analysis.

The Principal components analysis gave a result that accounted for 53% of the variance, with the two major components being the cruise element and the climb and descend aspects of the flights. The Factor analysis gave as major factor affecting workload the following:

- The entry and exit of aircraft in cruise, and cruising traffic in sector,
- The entry and exit of aircraft in climb, and climbing traffic in general, and
- Descending traffic in general.

The Ordinary Least of Square (OLS) regression analysis was undertaken from the pooled simulation results from the peak hour. No suitable linear regression relationships were obtained for workload with either of the following:

- Flight times by cruise, ascend, and descend, with the climb flight-time variable not significant and the intercept value too high; or
- Individual flight profiles, with the intercept value not significant.

Another simulation study (Leiden 2003) has analyzed the possible increase in airspace capacity connected with controller workload. The airspace sector capacity limit is reached when the demands of monitoring and controlling traffic exceed the human resources available. Two new concepts were implemented in the system the *En Route Free Maneuvering* and the *En Route Trajectory Negotiation*. The first concept through the application of new technologies, such as data-link communication, moves the locus of control to the flight crew, enabling them to freely maneuver in en route airspace subject to separating from traffic and area hazards. The greatest benefit of this concept is the scalability of the system. In fact, each equipped aircraft is self-separated increasing the amount of resources for other non-equipped aircraft. In the second

concept the locus of control remains on the ATC but now, through the integration of advanced flight deck automation, flight crew can exchange trajectory and clearance-related data with the ATC. It is claimed that all these features would eventually increase the controller productivity reducing the number of tasks for each flight. The controller task utilization was defined as the normalized time necessary to perform controller tasks and procedures. The amount of time not used in performing controller tasks is defined as idle utilization. In this concept, total utilization is defined,

Total Utilization = task utilization + idle utilization

An issue with this approach is that it's not easy to calculate the idle utilization because a lack of activity is very hard to be estimated. The results of the simulation were for some aspects surprising in fact with a complete implementation of both two new concepts on 85% of the Aircraft flying in the NAS the percentage increasing in capacity reach 330%, with 100% of aircraft equipped capacity is theoretically unlimited, at least from the controllers workload viewpoint. The unique limit in this scenario would be physical airspace and ability of aircraft equipment to find acceptable solutions.

## **3. METHODOLOGY**

### **3.1. *Model Choice***

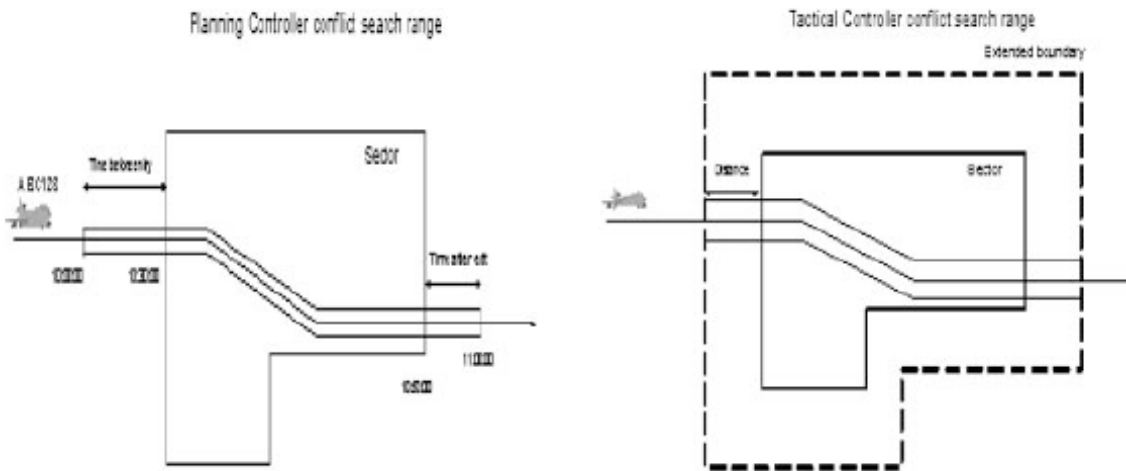
The first problem that a modeler has to face is the basic question of which model to use for his analysis. There are many different models in the market with different capabilities, Odoni et al. (Odoni 1997) presents advantages and disadvantages. The most commonly used models for Terminal Area simulation are SIMMOD, TAAM and RAMS. Because our study requires an evaluation of capacity and workload factors the choice was limited to TAAM and RAMS, SIMMOD Pro has added modeling capabilities to model ATC resources. We decided to use RAMS because of two important factors: availability and workload oriented features. The first reason is pretty obvious, the model was available in the Air Transportation System Laboratory at Virginia Tech and it was used in other studies. The second reason is the strong “controller oriented” framework behind RAMS, that in fact, was developed as an ATC workload model.

### **3.2. *RAMS Model***

The RAMS model is a fast-time, gate-to-gate, discrete-event simulation model that has been widely used for over 25 years in Europe for airspace planning. The model has been verified by controllers (EUROCONTROL 1999). RAMS generates flights profiles as four-dimensional trajectories through the airspace (3D plus time). Profiles are calculated using cruise, climb and descent speeds and rates provided via the 300-plus data-supplied aircraft models complemented by 60-plus performance tables, along with the option to define or modify additional aircraft models and performance (ISA-Software 2004). In the RAMS model, each control area is associated with a sector, which is a three-dimensional volume of airspace as defined in the real airspace situation. Each sector has an associated RAMS Planning Controller and a RAMS Tactical Controller, (see figure 3.1). These control areas maintain information regarding the flights that wish to penetrate them, and have associated separation minima and conflict resolution rules that need to be applied for each of the two RAMS control elements. The use of the Planning and Tactical control elements reflects the teamwork aspect of control observed in practice. The simulation engine also permits the input of our own rules to reflect



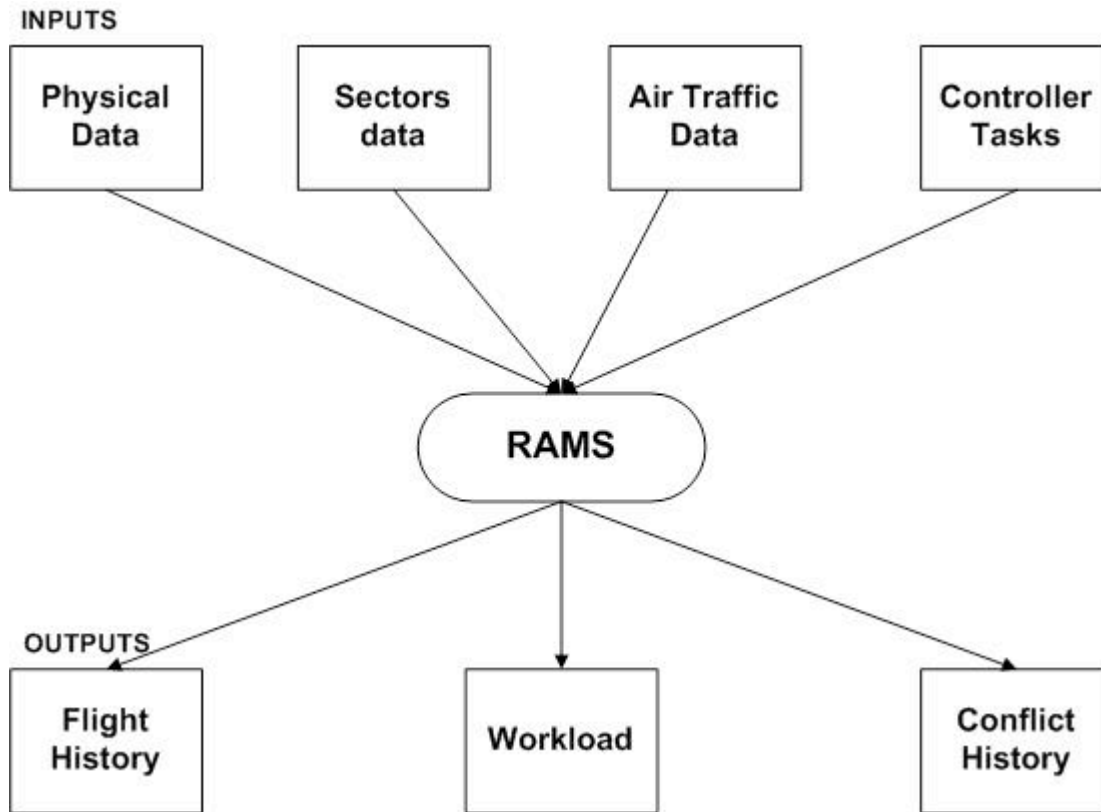
reality. The tasks for the controllers in RAMS are based on a total of 109 tasks, together with their timings and position, grouped into five major areas. These tasks are derived from a number of reference airspace regions of Europe, which include sectors in the London region, Benelux countries, France and Germany. Furthermore, a cloning engine enables future traffic demands to be generated, based upon the current air traffic patterns.



**Figure 3.1 The Control Elements in RAMS (Majumdar 2001, used with permission of the author).**

Using this type of model there are some issues that must be addressed in order to produce meaningful results. Some important “rules” that must be respected especially for input data are the following:

- the area of airspace simulated, the characteristics of the ATC sectors and the air routes through them are contained in the sector data input files ;
- the air traffic simulated, the characteristics of the aircraft and their performance capabilities are contained in the air traffic data input files ;
- the simulated controller tasks and procedures, the set of controller's tasks and their timings are contained in the controller task input files. The choice of an appropriate set and its implications are of the utmost importance in both undertaking as well as understanding the simulation results.



**Figure 3.2 Input and Output Structure in RAMS.**

The workload estimates obtained in the RAMS model are based upon task-time definitions derived from a detailed non-intrusive objective record of the controller's actions, aided by controller verification. Based upon these task-time definitions, threshold controller loadings of a control team (Tactical and Planning) at capacity being 42 minutes/hour must be utilized for RAMS.

### **3.3. *Building the Simulation Scenarios***

In order to create a scenario in RAMS many different kind of input are needed to create a detailed gate-to-gate simulation as the one built for our research effort. The input needed can be grouped into the following categories:

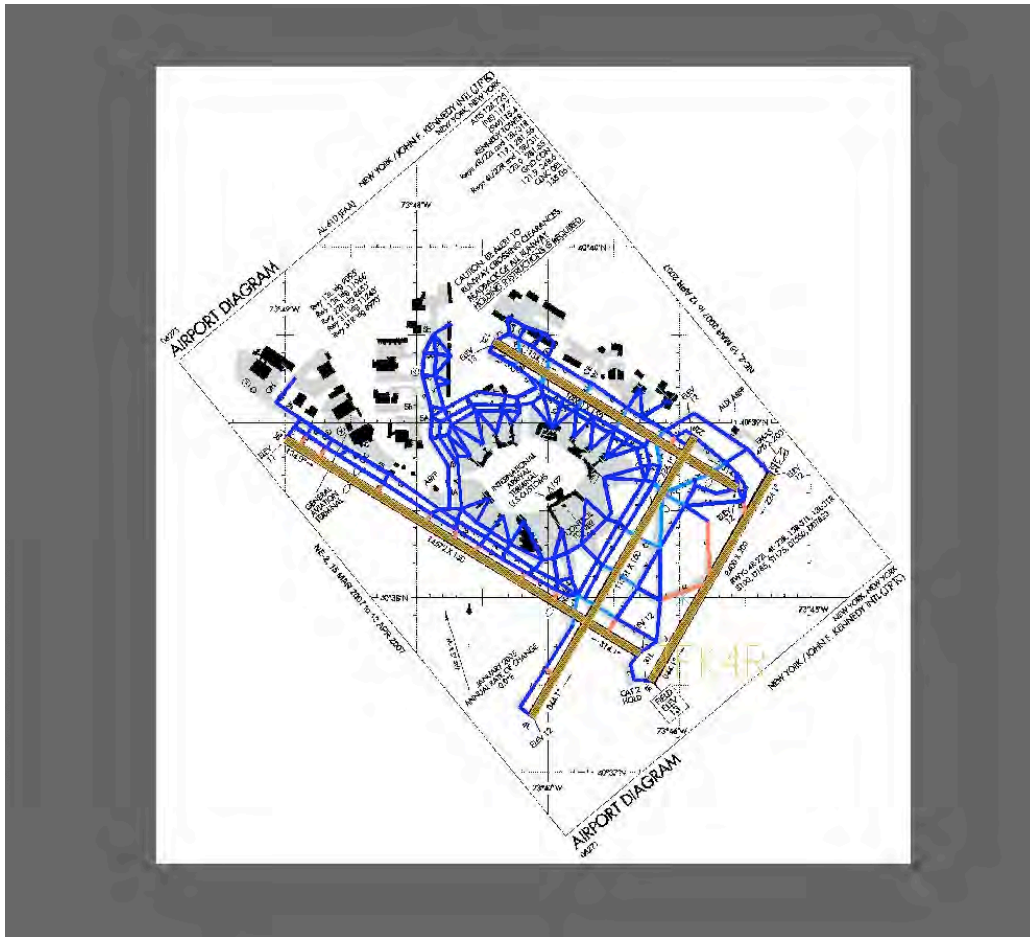
1. Physical data of the scenario (Airports layouts, approach and departure routes, etc),
2. Traffic demand data,
3. Airspace data, and

#### 4. Workload tasks data.

The datasets and sources of data to “feed” and validate the simulation come from various data sources. These are: FAA datasets (e.g. ASPM, ATO-P Forecasts, PDARS); other pieces of information were obtained from some general tools available in the public domain such as googleearth®, a very useful and powerful web-based tool based on satellite images. Other sources of information are websites that provide detailed information on airport layout, approach routes (STAR) and departure routes (SID). Sometimes mixing various sources and tools provides a powerful way to visualize operations. For example, combining PDARS data and googleearth®, is easy to obtain detailed representations of the flight paths flown by real aircraft traffic.

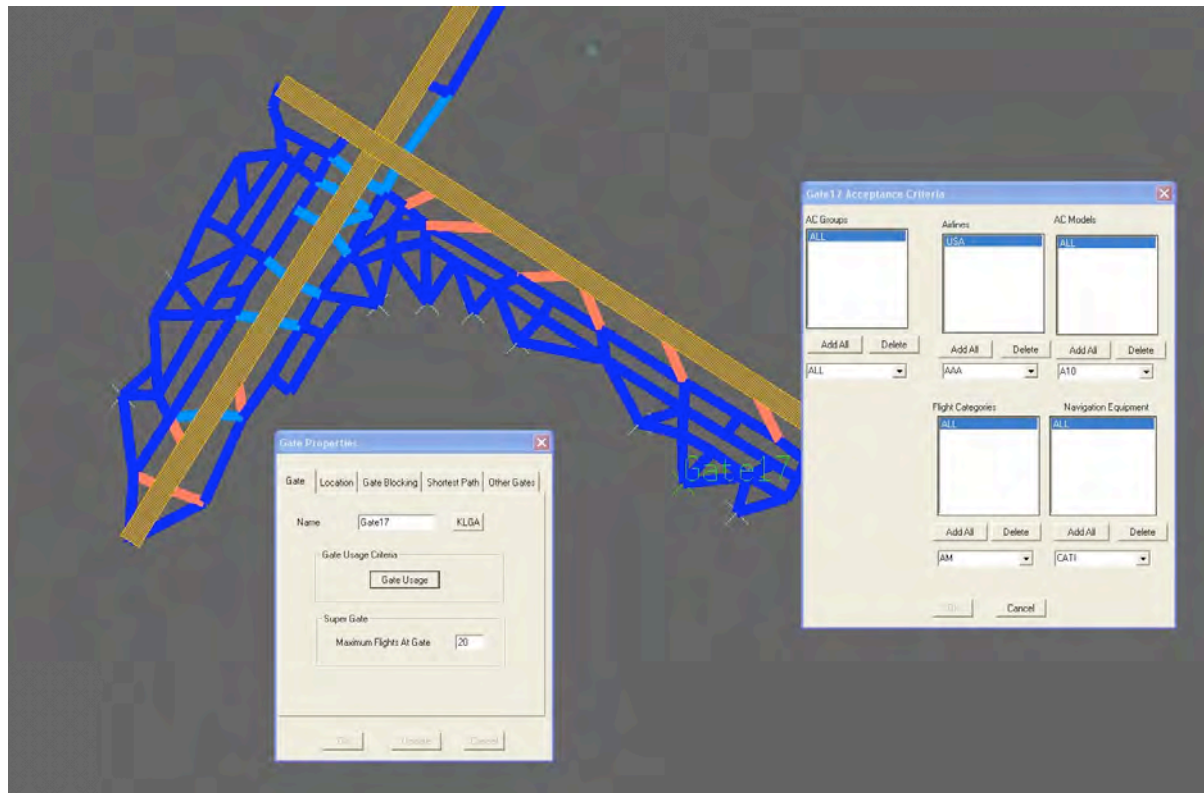
#### **3.4. *Drawing the Airports***

In order to build a gate-enabled airport simulation, a detailed representation of the airports in the study must be drawn. RAMS has a very useful capability that allows the superimposition of airport information the airport plates to the simulation scenario in order to guide the location of gates, ground links, and runways. The airport information plates were obtained from airnav.com (airnav.com 2007). Airnav includes all the airport plates, Standard Terminal Arrival (STAR) procedures, Standard Instrumental Departure procedures (SID), and Navaids.



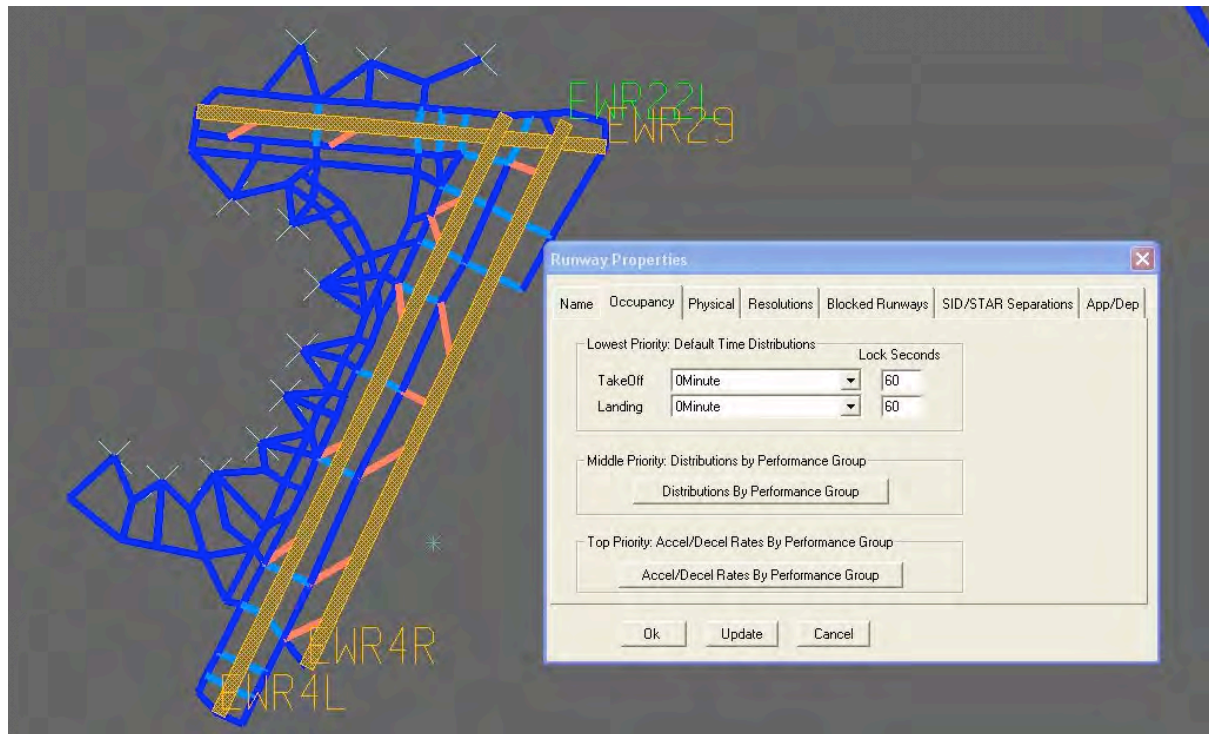
**Figure 3.3 JFK Airport Plate Superimposed in RAMS.**

During the drawing phase of the modeling work some simplifications are needed in order to make the scenario easier and faster to run. Large apron areas with multiple gates are difficult to represent in RAMS so a simple link structure was created. Each link has a capacity and other attributes such as link speed, utilization by different aircraft class, etc. Another simplification made was the grouping of several gates used by the same airlines into “Super Gates” with equivalent or higher capacity. It is important to have a realistic gate capacity and their allocation to airlines into gate. As we know in the USA, most of the hub airports have gates owned by airlines that can use them as they want. The information about the gate allocation were found at the airport websites, especially in the New York Port Authority one (NJ 2007) there is a detailed list of all the airlines using each gate. After the gates are physically drawn all the attributes must be input in the model in terms of capacity, utilization, etc.



**Figure 3.4 LGA Gate Attribute Window.**

After the ground network with the gates is completed the most important part of the airport drawing can be added, i.e. the runway system. In RAMS runway attributes control all the operations logic, separations and sequencing so a correct calibration of all these parameters is critical to perform a reliable and faithful simulation study. Drawing the runways in RAMS by superimposing the airport plates as layer eases the positioning of runway links and nodes. For example, the length of the runway compared to the real is almost exact, with an error that can be considered negligible. A function in RAMS straightens the runway in order to obtain a better graphical effect. After all physical runway attributes that can be implemented, the runway “occupancy” window is the focus of the sequencing criteria that will be applied in the simulation model.



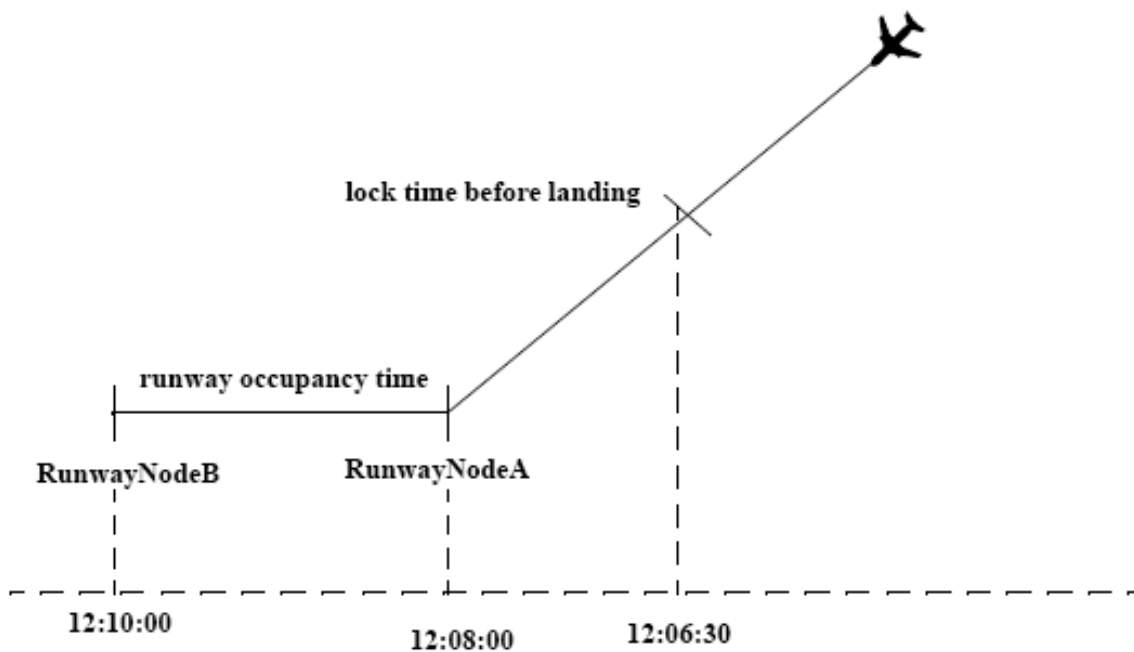
**Figure 3.5 Runway Attributes Window. EWR 22L Occupancy Characteristics.**

In RAMS the separation applied at the runway level are controlled by three levels of control:

- Acceleration/Deceleration rate driven by individual performance group,
- Distributions by performance group, and
- Time distributions (Take off and landing plus lock time).

This is one of the key concepts to understand RAMS operations modeling. From this priority list we can understand how “performance” oriented the model is. RAMS differs from other models like SIMMOD where all the separations are based on the link attributes. In fact, RAMS applies sequencing and separations based first on the performance of the leading and trailing aircraft; then it will check if there are any distributions applied to any of the performance group of the two flights and then it will add a lock time that is defined for each runway direction for take off and landing. The summation of these three factors will be the final separation applied by the model. Tweaking these three parameters produces separations at the runway level, which then will be also the one applied at the approach and departure routes level. With the conflict

resolution switch on, RAMS enforces separations to ensure that no violations and conflicts will occur in the simulation. The controllers' rules will then apply these criteria on the actual simulation. RAMS is controller centric. These rules can be pretty straightforward for non-intersecting runway operations like the ones observed at JFK and EWR. For an airport like LGA, where operations are always carried out at intersecting runways, these rules are more difficult to fine tune and became the key part to evaluate the fidelity of our simulation runs.



**Figure 3.6 Performance Time (ROT) Plus Locking Time Representation. (ISA-Software 2004)**

Other important input to the runway operation mask are the resolution criteria that must be given to RAMS in order to solve the conflicts and to send the aircraft if one of that happen. Holding stack, alternate runways and alternate approaches must be implemented here; especially the holding stack are very important to give RAMS the chance o separate aircraft that cannot be stopped in the air as SIMMOD does. Some extra holding stack delays must be then expected than reality because once the flight path is designed it cannot be changed except that with these tools.

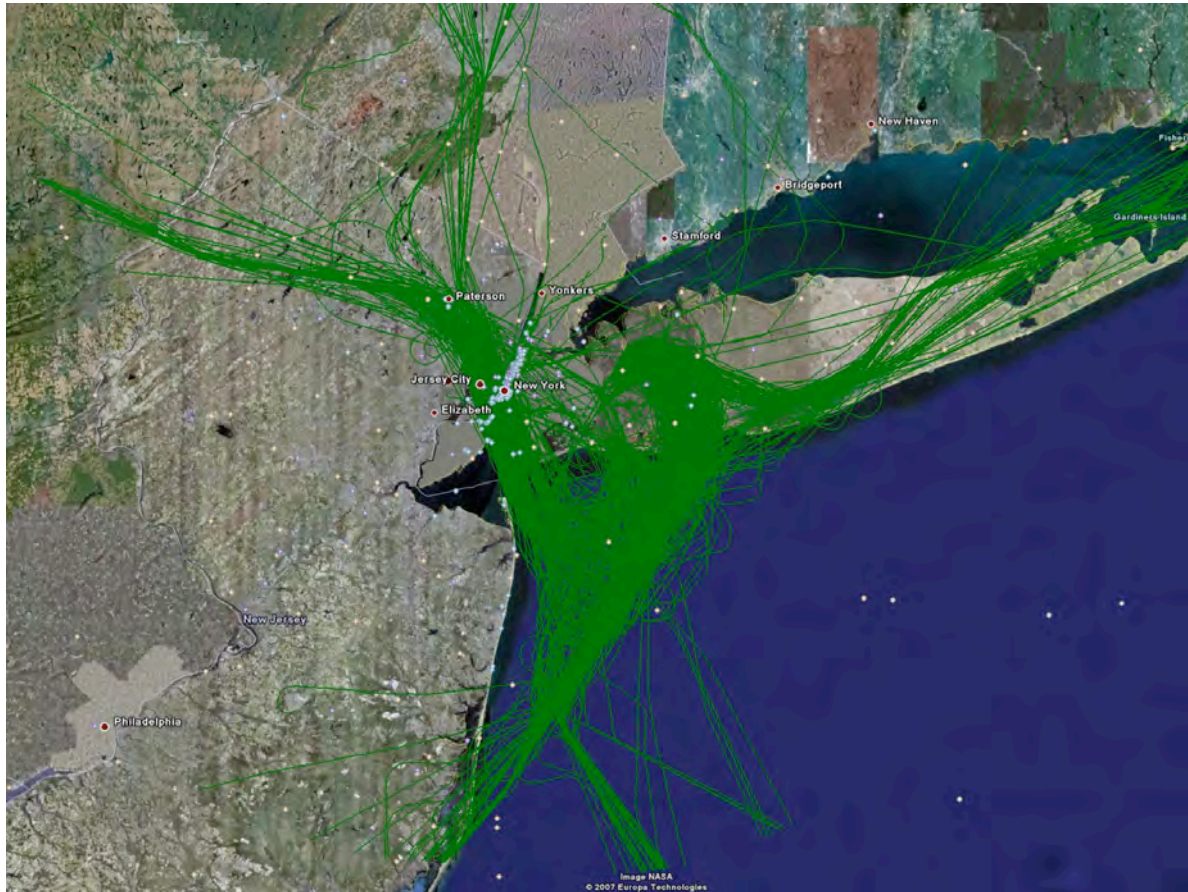
An important piece in modeling intersecting runway operation configuration like at LGA Airport, is the blocked runways mask. RAMS employs a runway blocking procedure where selected operations are input in a two-by-two matrix representing arrivals and

departures. In this way the model prevents the simultaneous occupation of two intersecting runways by two aircraft.

The last inputs in RAMS modeling are the approach and departure routes for each runway. Arrival and departure routes can be constructed using two methods:

- SID and STAR plates, or
- PDARS flight track data.

The second option was used in this project because PDARS track data is more realistic. PDARS data was obtained from FAA ATO Office for February 23, 2006. The data was visualized in googleearth®. As expected, simplification of the flight tracks was do to represent the actual tracks flown that day.



**Figure 3.7 JFK Arrivals from PDARS Data in Googleearth® (used by courtesy of Google).**



This combination of PDARS and a web-based tool such as googleearth® provided a powerful graphical tool to visualize complex flight paths of New York airports. The results were useful to compare actual flight paths with SID and STARS procedures.

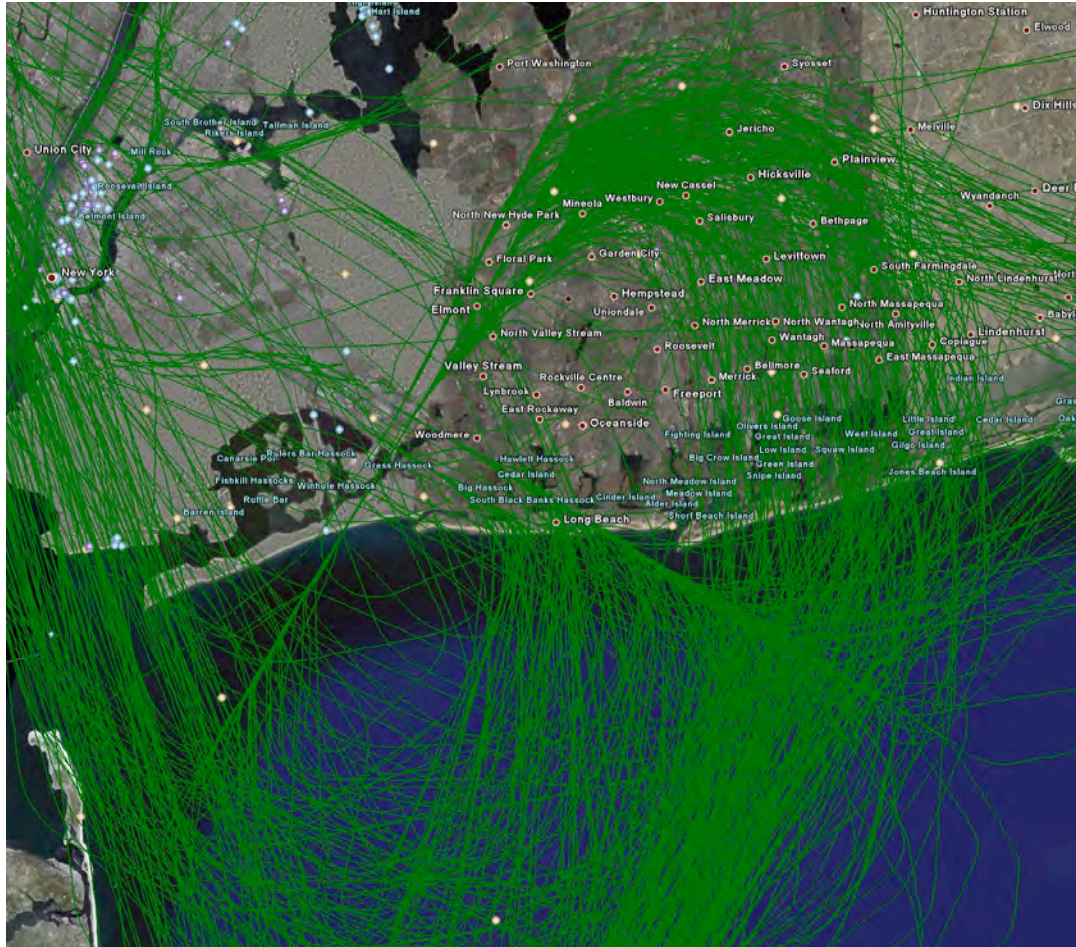


Figure 3.8 Detailed Image of JFK Arrival Paths in Googleearth® (used by courtesy of Google).

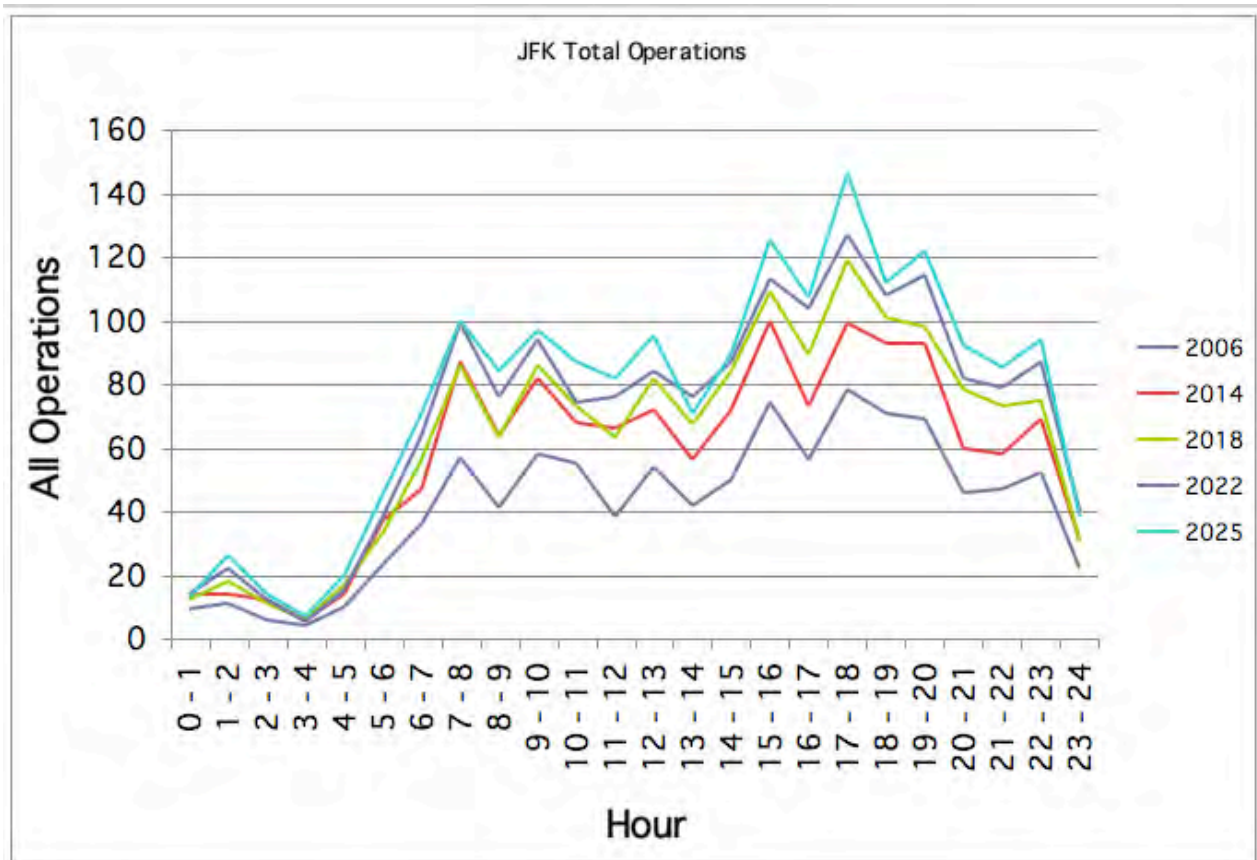
### 3.5. Traffic Demand Data

The traffic demand data used in the study was obtained from the FAA ATO-P office. Five distinct forecast years were available for the study: 2006, 2014, 2018, 2022 and 2025. The FAA data contains detailed information about each flight including flight level, speed and waypoint information. The baseline year corresponds to year 2006. For each year, two days of every quarter were available including peak and off-peak days. Peak days for air traffic are Monday and Friday. Off-peak days are the weekends. Each record of the database has the following information:

1. ID Number, unique flight number. The number is created by ATO-P during the forecasting process.
2. Date, contains the US date of the extraction for the baseline data used to create the forecast in YYYYMMDD format.
3. Aircraft ID, ETMS provided aircraft ID.
4. Flight Index, ATALAB created number.
5. Flight Plan Type: ORIGINAL\_FLIGHT, CLONE\_####, VFR, etc.
6. Departure Time.
7. Departure Time Flag, Flag that tells whether the departure time above is from the filed flight plan, or if the filed flight plan does not have a departure time, is the actual departure time.
8. Arrival Time.
9. Arrival Time Flag, as above.
10. Filed Altitude, in 100's of feet.
11. Filed Speed: Filed speed in knots (true airspeed).
12. Departure Airport: ETMS code (FAA for US, can be FAA or ICAO for international).
13. Arrival Airport.
14. Departure Latitude, Departure Longitude, Departure Elevation.
15. Departure Airport Country Code, 1 for US, 2 for Canada, others are in ATALAB Airports table.
16. Arrive Latitude, Arrive Longitude, Arrive Elevation.
17. Arrive Airport Country Code, as above.
18. Aircraft Type, Up to four character equipment code provided by the ETMS System.

19. Physical Class, J = jet, T = turboprop, P = piston.
20. User Class: one character indicator of the user class provided by the ETMS System.  
C = Commercial, F = Freight, G = GA, M = Military, T = Air Traffic, O = Other.
21. Flew Flag: ETMS flight table flew\_flag. >0 means etms is almost positive it actually flew. =0 means etms thinks it didn't fly.
22. Airspace Code, code to see if flight actually touched US airspace based on ATALAB boundary crossing table.
23. Departure ICAO, ICAO code for departure airport. Reverts to FAA code if cannot identify ICAO code.
24. Arrive ICAO, as above.
25. ATO User Class, New user class defined by airline code. It modifies the ETMS user class category to better reflect passenger/cargo as well as operations identified as regional or fractional.
26. BADA Aircraft ID: Aircraft type consistent with BADA version 3.6.
27. BADA Source: ATO-P or ACES.
28. ETMS Filed Waypoints: the field contains the filed waypoints from ETMS. These waypoints are latitude/longitude pairs represented in minutes with longitude measured from the prime meridian as West positive.

In order to convert all the information into a format that RAMS could read as a demand file, a parser file was created in Matlab. RAMS simulations require two demand files, e.g. traffic.dat and trafficprofile.dat. These files require fewer data fields than those contained in ATO-P demand files. Traffic.dat requires information on the flight ID, the simulation entry time. Trafficprofile.dat requires waypoint and route information. All the demand sets available from FAA-ATO for all years studied are plotted in figures 3.9-3.11 to see the differences in flight demand in the future.



**Figure 3.9 ATO-P Forecast for JFK Up to Year 2025.**

According to FAA forecasts, the demand at JFK Airport will grow substantially in the last year. The total flight count is expected to increase by 81% according to the FAA forecast. It is not clear how this demand could be serviced at a congested facility. The situation of EWR is very different. In the last year of the analysis the overall growth is expected to be 57% for arrival and departure operations combined. This reduced improvement could be explained by the fact that FAA claims that EWR is closer to its capacity limits than JFK.

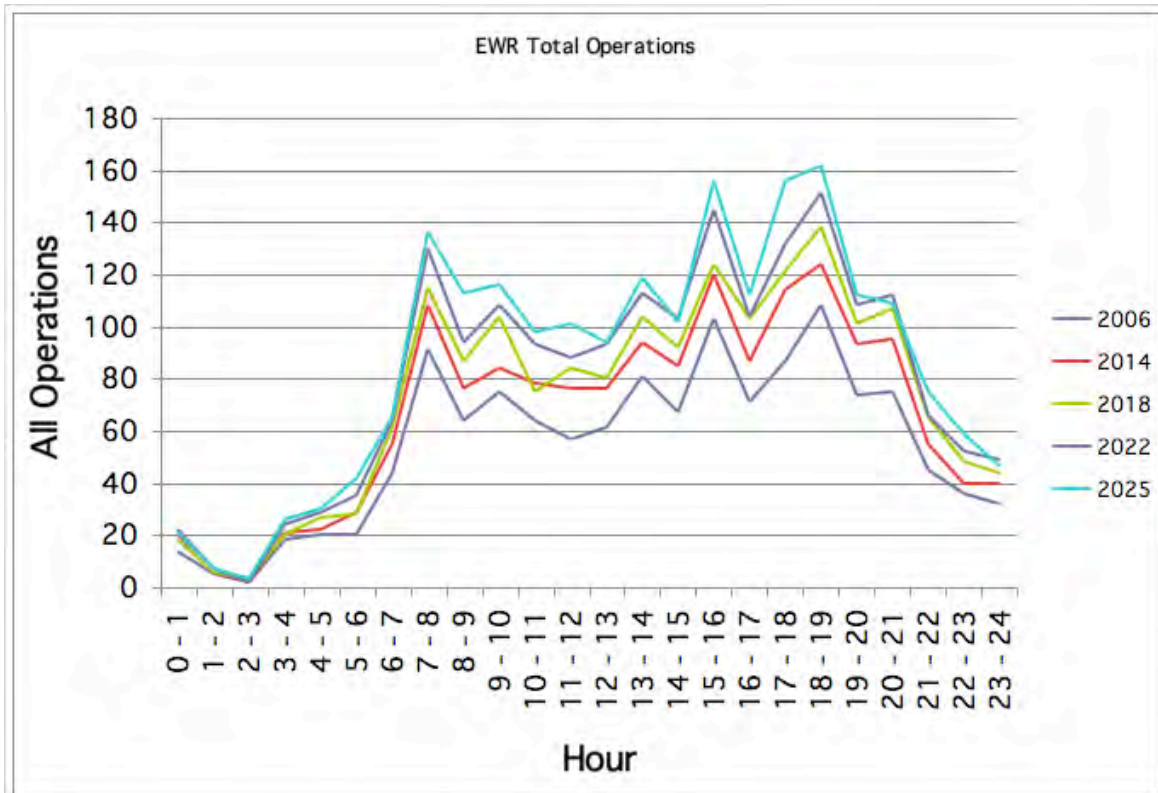


Figure 3.10 ATO-P Forecast for EWR Up to Year 2025.

The future demand for LaGuardia Airport is expected to remain unchanged through the year 2025 as in fig. 3.11. LGA is operating at near capacity and there is no room for more demand in the future.

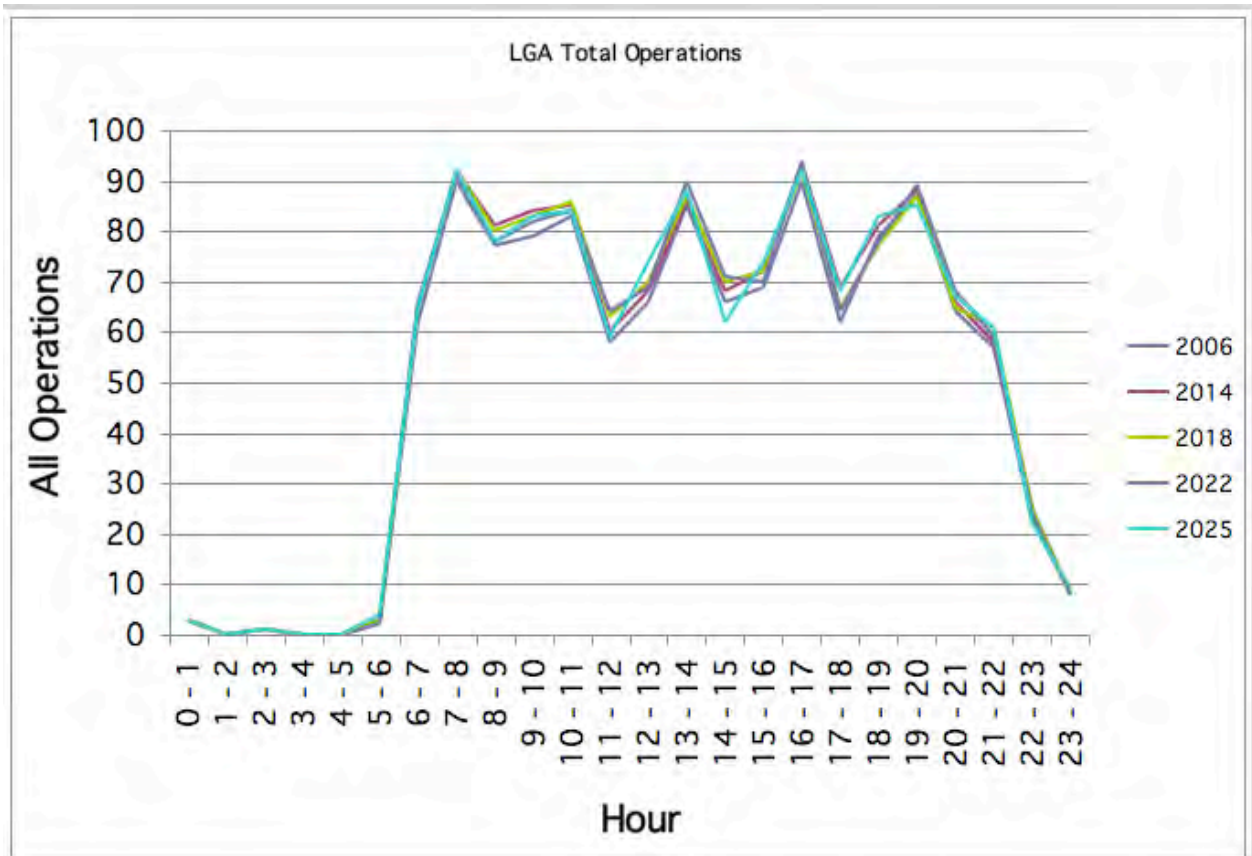


Figure 3.11 ATO-P Forecast for LGA Up to Year 2025.

### 3.6. *Building the sectors*

In RAMS, each control area is associated with a sector. Each sector is a three dimensional volume of airspace. The sector structure around the New York City Metroplex area was approximated by analyzing the arrival and departure approach procedures of each airport. An important consideration in RAMS is the number of controllers managing a sector. This number was approximated by counting the number of frequencies available at airport approach and terminal procedures. The results of this analysis are represented in Tables 3.1-3.3.

**Table 3-1 JFK Frequency List.**

<b>JFK</b>		
<b>Frequency Name</b>	<b>Frequency</b>	<b>OPS/RWY</b>
Tower 1	119.1	4R/22L & 13L/31R
Tower 2	123.1	4L/22R & 13R/31L
GND CON	121.9	all
CLNC DEL	135.05	all
ATIS DEP	115.1	Information Service
ATIS DEP	128.725	Information Service
NY DEP CON	135.9	DEP/31/4/22
NY APP CON	127.4	ARR/22
ATIS ARR	128.725	Information Service
ATIS ARR (NE)	117.7	Information Service
ATIS ARR (SW)	115.4	Information Service
NY APP CON	133.1	ARR/22
NY APP CON	125.7	ARR/22/31

JFK is the only airport that has two tower frequencies to independently control two main runways and, of two the ground area. There are three frequencies for arrivals and one for departures.

**Table 3-2 LGA Frequency List.**

<b>LGA</b>		
<b>Frequency Name</b>	<b>Frequency</b>	<b>OPS/RWY</b>
Tower	118.7	all
ATIS DEP	127.05	Information Service
ATIS ARR	125.95	Information Service
GND CON	121.7	all
CLNC DEL	135.2	all
NY DEP CON	120.4	DEP/ALL
NY APP CON	120.8	ARR/ALL
NY APP CON	127.3	Approaches
ATIS ARR	113.1	Information Service
NY APP CON	128.55	Approaches

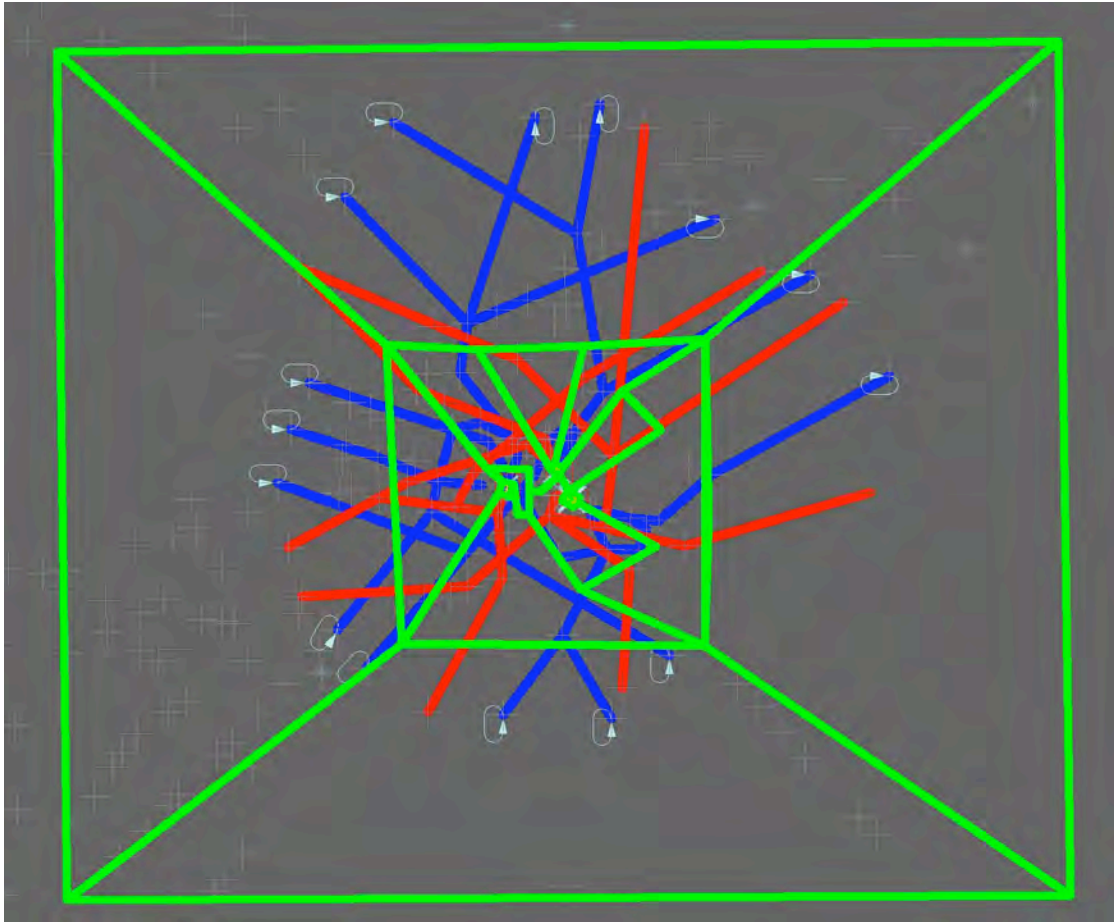
La Guardia and Newark have, like JFK, three frequencies for arrivals and one for the departure operations.

**Table 3-3 EWR Frequency List.**

<b>EWR</b>		
<b>Frequency Name</b>	<b>Frequency</b>	<b>OPS/RWY</b>
Tower	118.3	all
GND CON	121.8	all
CLNC DEL	118.85	all
ATIS	115.7	Information Service
ATIS	134.825	Information Service
NY DEP CON	119.2	DEP/ALL
NY APP CON	120.15	Approaches
NY APP CON	127.6	Approaches
NEWARK ATIS	134.82	Information Service
NY APP CON	127.3	Approaches

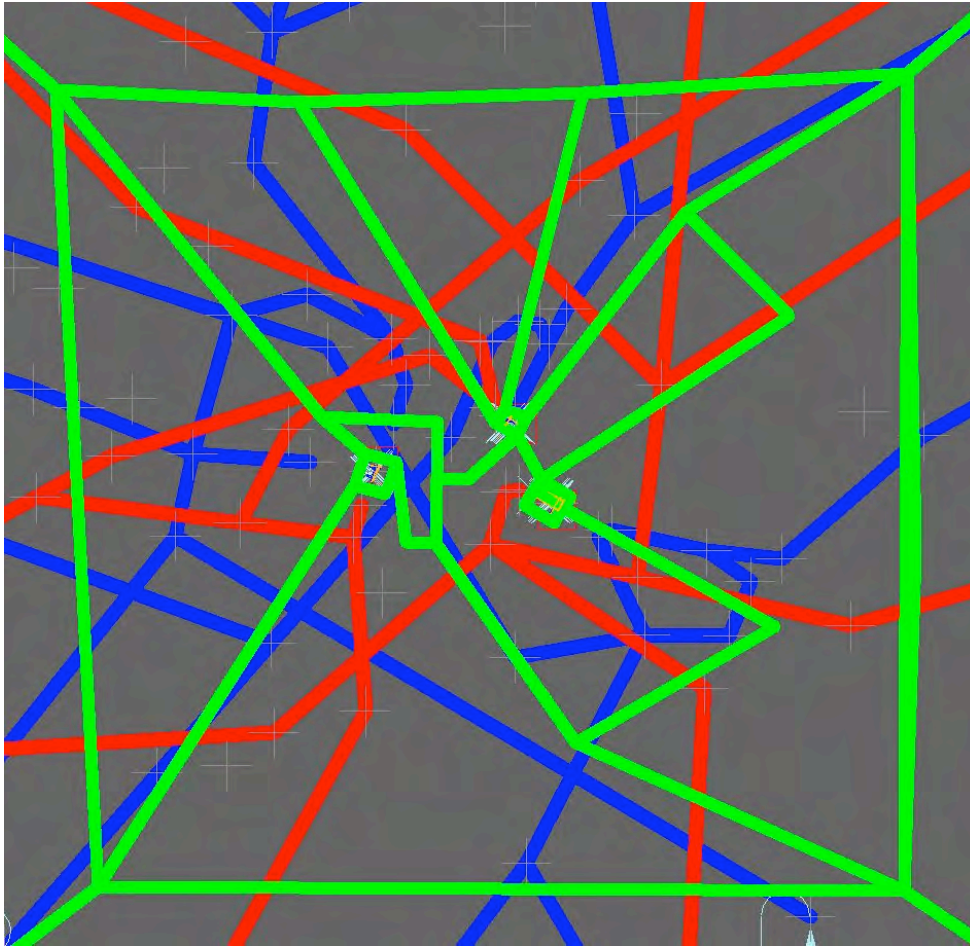
Studying approach and departure procedures, we obtained the sector structure in RAMS presented in Fig. 3.12 and 3.13. Four “super” sectors were added to control the arrival metering nodes. These sectors control a part of airspace system that is at the edge of the terminal area allowing the necessary control on the arrivals and departure operations in the RAMS simulation.





**Figure 3.12 The Sectors Structure Included in the Simulation.**

The separations applied inside the core sectors and that applied in the external sectors are different. The external sectors have the role of enforcing the Miles-in-Trail (MIT) separations of ten nautical miles, whereas in the core area the separation is reduced to the wake vortex values for sequencing of the aircraft in the final approach segments.



**Figure 3.13 Detail of The Core Sectors in the Simulation.**

### **3.7. *Workload Tasks Data***

The workload associated with ATC events is dynamically allocated during the simulation, refined by the unique airspace situation at the time of the ATC event. These workload calculated are data-defined allowing general-to-specific airspace conditions determine the weight attached to the workload event. Controllers handle the flow of traffic through the sectorised airspace, and therefore the window entries influence many simulation events, such as conflict detection, capacity, workload, etc. Controllers have separation values used to check spatial conflicts between flights upon entry into the information windows. The choice of the controllers' tasks weighted in workload calculations is important to have confidence in the final results of the simulation. RAMS divides tasks based on five different categories and then assigns to each task a weight that represents the duration to complete the task. The five categories are:

1. Conflict Search Activities
2. Coordination Activities (Internal and External)
3. Flight Data Management Activities
4. Communication Activities
5. Radar Resolution Activities

Default data was used in our simulation because we do not have additional information about the future data communication parameters. Assumptions were made about future communication systems analyzed in our study. For the study the task category with highest relevance is “Communication Activities”. All the tasks of this category deal with the communication process between controllers and pilots. These task allow controllers to apply control procedures to manage traffic flow, enforce separations, and apply safety restrictions.

Nine tasks belong to the *Communication* group, of a total number of forty-nine tasks. The weight of these tasks, i.e. the time necessary to perform them by the controllers, was gradually reduced from the baseline scenario to the Data-Communication Phase 1 and 2. Six of the *Coordination Activities* and one of the *Flight Data Management* were also reduced in the scenarios with the implementation of the future communication technologies. The tasks reductions were obtained from the CPDLC studies conducted by the FAA in the nineties (FAA 1995; FAA 1996).

### **3.8. Analyzing the New York Air Traffic Control Area**

A real world scenario is always more complicated than a simulation model. There are some simplifications that are necessary to make the simulation easy to run. Due to variation in meteorological condition runway configurations used may vary This type of behavior is complicated to be modeled in RAMS because once you have built SIDs and STARs they are fixed and you cannot change them. Furthermore these routes are associated to a single runway and cannot be easily associated to multiple runways. For this reason, several single configuration scenarios can represent the performance of the

system that continuously changes. As expected, when there are intersecting operations, the results of a simulation yield different results and controller workload conditions. To accurately model New York, three different scenarios were built to mimic the most common configurations in this airport system. Data from the ASPM database was collected to evaluate the most common configurations for all three airports considered in the analysis, i.e. LGA, EWR and JFK. The results of this analysis are summarized in Tables 3-4 to 3-8:

**Table 3-4 Scenario 1 Operations.**

Scenario 1		
Airport	Runway Set (ARR/DEP)	Coverage (%)
LGA	22/13	29.08
EWR	22L/22R	29.99
JFK	13L/13R	15.65

La Guardia Airport is used as the reference airport in the analysis. The first scenario reflects the most its most used configuration where arrival operations are performed on runway 22 and departure operations on runway 13. This is also the configuration with the highest capacity, in fact the two intersecting operations are operated with the lowest interference between each other. For Scenario 1, the configuration of JFK in use is 13L for arrivals and 13R for departures. In Newark the runways in use are 22L for arrivals and 22R for departures. In the case of JFK this configuration is not the most used but it's the most used paired to the LGA operations, stated in the table.

**Table 3-5 Scenario 2 Operations.**

Scenario 2		
Airport	Runway Set (ARR/DEP)	Coverage (%)
LGA	31/4	19.17
EWR	4R/4L	22.1
JFK	31R/31L	25.13

Table 3.2 summarizes the operations for Scenario 2. In this case, LGA is operating on runway 31 for arrivals and runway 4 for departures. This configuration is less frequently used of the three scenarios because is the configuration with the lowest capacity. The configuration at EWR uses runway 4R for arrivals and runway 4R for departures. In this scenario the configuration of JFK most frequently used is 31R for arrivals and 31L for departures. Table 3.6 represents the aggregated utilization values and total coverage of the two scenarios.

**Table 3-6 Total Coverage With Two Scenarios.**

With 2 Scenarios	
Airport	Total Coverage (%)
LGA	48.25
EWR	52.09
JFK	40.78

In the third scenario EWR runway 22L is used for arrivals and 22R for departures with a sporadic use of runway 11 for additional arrivals of small jets limited by the length of the runway. At LGA, runway 22 is used for arrivals and runway 31 for departures. At JFK, runway 31L is used for mixed operations (i.e. arrivals and departures), and runway 31R only for arrivals. As we can see from Table 3.8 the total coverage with three scenarios is around 75% of the total configurations in use. JFK has the lower percentage in the modeling.

**Table 3-7 Scenario 3 Operations.**

Scenario 3		
Airport	Runway Set (ARR/DEP)	Coverage (%)
LGA	22/31	19.66
EWR	11, 22L   22R	22.39
JFK	31L, 31R   31L	16.47

**Table 3-8 Total Coverage With Three Scenarios.**

With 3 Scenarios	
Airport	Total Coverage (%)
LGA	67.91
EWR	74.48
JFK	57.25

At the end of the study all the delays and workload data are weighted based on these utilization percentage values to estimate a value for capacity multiplier to be applied in the future. Each of the scenarios was tested with four demand sets coming from ATO-P Forecasts corresponding to 2006, 2014, 2022 and 2025. The last was set as the final year from the analysis corresponding to the year when NextGen should enter into service. These demand sets were also tested for an equipage to reflect conditions of segments 1 and 2 in DATACOM and evaluate the possible benefits that would eventually arise in terms of delays and workload with the implementation of this new communication concept.

**Table 3-9 Runs Performed for All the Scenarios and Equipment Levels.**

<b>Scenario 3</b>	<b>2006</b>	<b>2014</b>	<b>2022</b>	<b>2025</b>
Do Nothing	√	√	√	√
Data-Comm Segment 1	√	√	√	√
Data-Comm Segment 2	√	√	√	√

### **3.9. *Balancing the Operations in JFK***

ATO-P forecast data give only the arrival and departure airport of the demand that we needed but a problem that we had to face in the last scenario of our analysis was to balance the operations in JFK where two runways were used for arrivals, i.e. 31L and 31R, and one of these, i.e. 31L, for departures. As we know the problem of balancing operations could be very important to efficiently operate a runway system, and there have been numerous studies addressing this problem. The most common approach

used is the application of Linear Programming to solve an optimization problem. A series of studies (Gilbo 1993; Gilbo 1997; Gilbo 2001; Gilbo 2003) were performed at the Volpe Center in a time window of ten years and the final result was a practical application of this method to create a decision support tool for air traffic controllers to optimally allocate arrivals and departures on the airfield. The implementation of a heuristic runway balancing algorithm was developed and implemented outside of RAMS to assign traffic at JFK with both runways 31L and 31R processing arrivals demand file from the ATO-P Forecast data.

To test the runway balancing procedure we performed an analysis of the demand provided by ATO-P without runway balancing and using one runway for departures, (31L), and one for arrivals (31R).

**Table 3-10 JFK Original Demand.**

Original Demand			
Hourly Split	ARR	DEP	Total Operations
00:00 - 01:00	12	3	15
01:00 - 02:00	11	2	13
02:00 - 03:00	12	4	16
03:00 - 04:00	14	2	16
04:00 - 05:00	14	1	15
05:00 - 06:00	24	6	30
06:00 - 07:00	24	16	40
07:00 - 08:00	24	27	51
08:00 - 09:00	19	41	60
09:00 - 10:00	14	37	51
10:00 - 11:00	32	23	55
11:00 - 12:00	30	27	57
12:00 - 13:00	25	19	44
13:00 - 14:00	26	15	41
14:00 - 15:00	36	16	52
15:00 - 16:00	30	18	48
16:00 - 17:00	26	21	47
17:00 - 18:00	16	43	59
18:00 - 19:00	32	44	76
19:00 - 20:00	15	39	54
20:00 - 21:00	13	18	31
21:00 - 22:00	14	26	40
22:00 - 23:00	4	21	25
23:00 - 24:00	4	8	12
Tot	471	477	948

The arrivals demand was assigned to the two arrivals runways based on the distance between the arrival airport and the runway center. This first approximation approach resulted in an unbalanced demand between the two runways, with runway 31L absorbing most of the arrivals and also the most of the departures.

**Table 3-11 Unbalanced Demand by Hour.**

Unbalanced Demand					
Hourly Split	31L ARR	31R ARR	31L DEP	31L Total	Total Ops.
00:00 - 01:00	12	0	3	15	15
01:00 - 02:00	9	2	2	11	13
02:00 - 03:00	7	5	4	11	16
03:00 - 04:00	8	6	2	10	16
04:00 - 05:00	3	11	1	4	15
05:00 - 06:00	12	12	6	18	30
06:00 - 07:00	13	11	16	29	40
07:00 - 08:00	17	7	27	44	51
08:00 - 09:00	12	7	41	53	60
09:00 - 10:00	10	4	37	47	51
10:00 - 11:00	24	8	23	47	55
11:00 - 12:00	26	4	27	53	57
12:00 - 13:00	19	6	19	38	44
13:00 - 14:00	23	3	15	38	41
14:00 - 15:00	33	3	16	49	52
15:00 - 16:00	27	3	18	45	48
16:00 - 17:00	22	4	21	43	47
17:00 - 18:00	15	1	43	58	59
18:00 - 19:00	28	4	44	72	76
19:00 - 20:00	15	0	39	54	54
20:00 - 21:00	12	1	18	30	31
21:00 - 22:00	9	5	26	35	40
22:00 - 23:00	2	2	21	23	25
23:00 - 24:00	4	0	8	12	12
Tot	362	109	477	839	948



In order to balance the arrival demand we used a very simple heuristic. Using runway 31L as the only departure runway we set different levels of acceptance rate for arrivals on the same runway. In order to obtain a realistic level of acceptance rate we used the PDARS data and constructed a capacity envelope with the actual data in hand to represent our balancing approach.

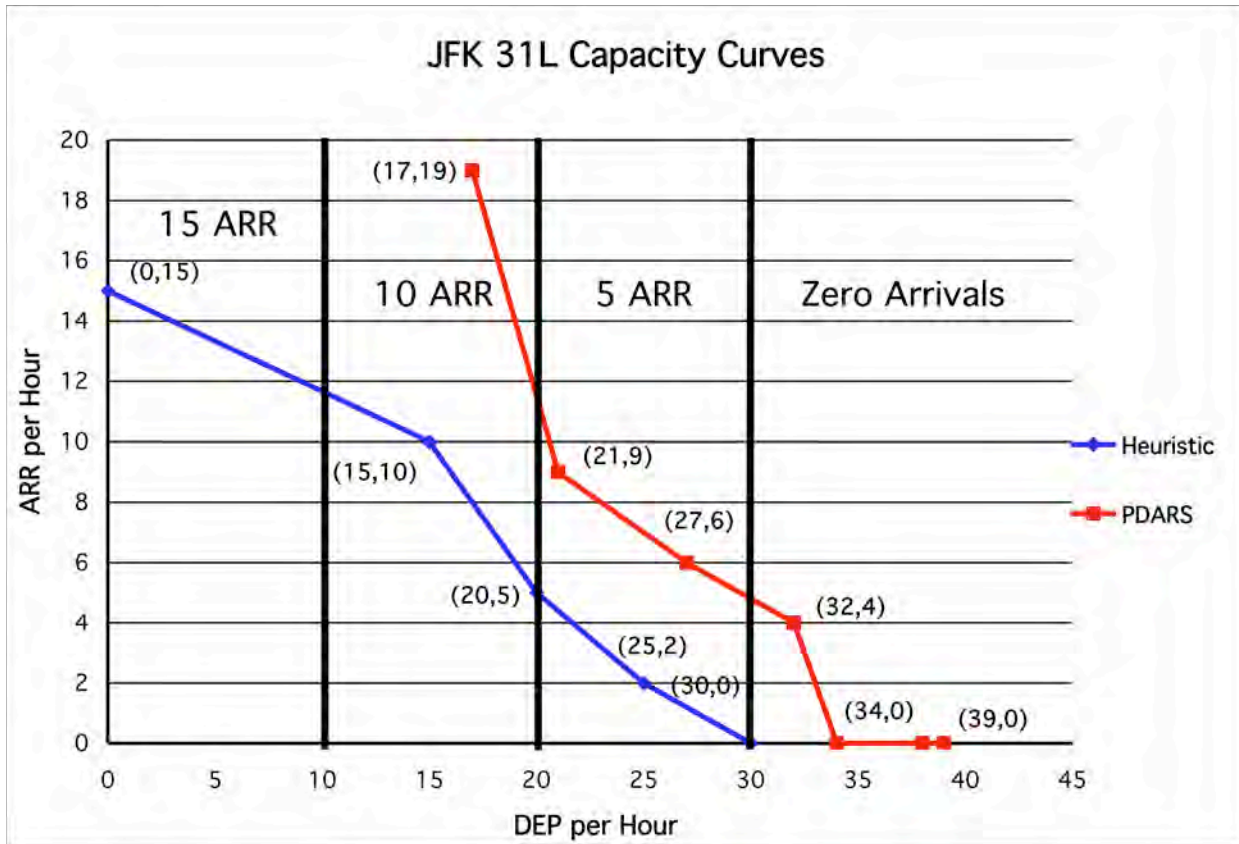


Figure 3.14 Capacity Diagram Derived from PDARS Data.

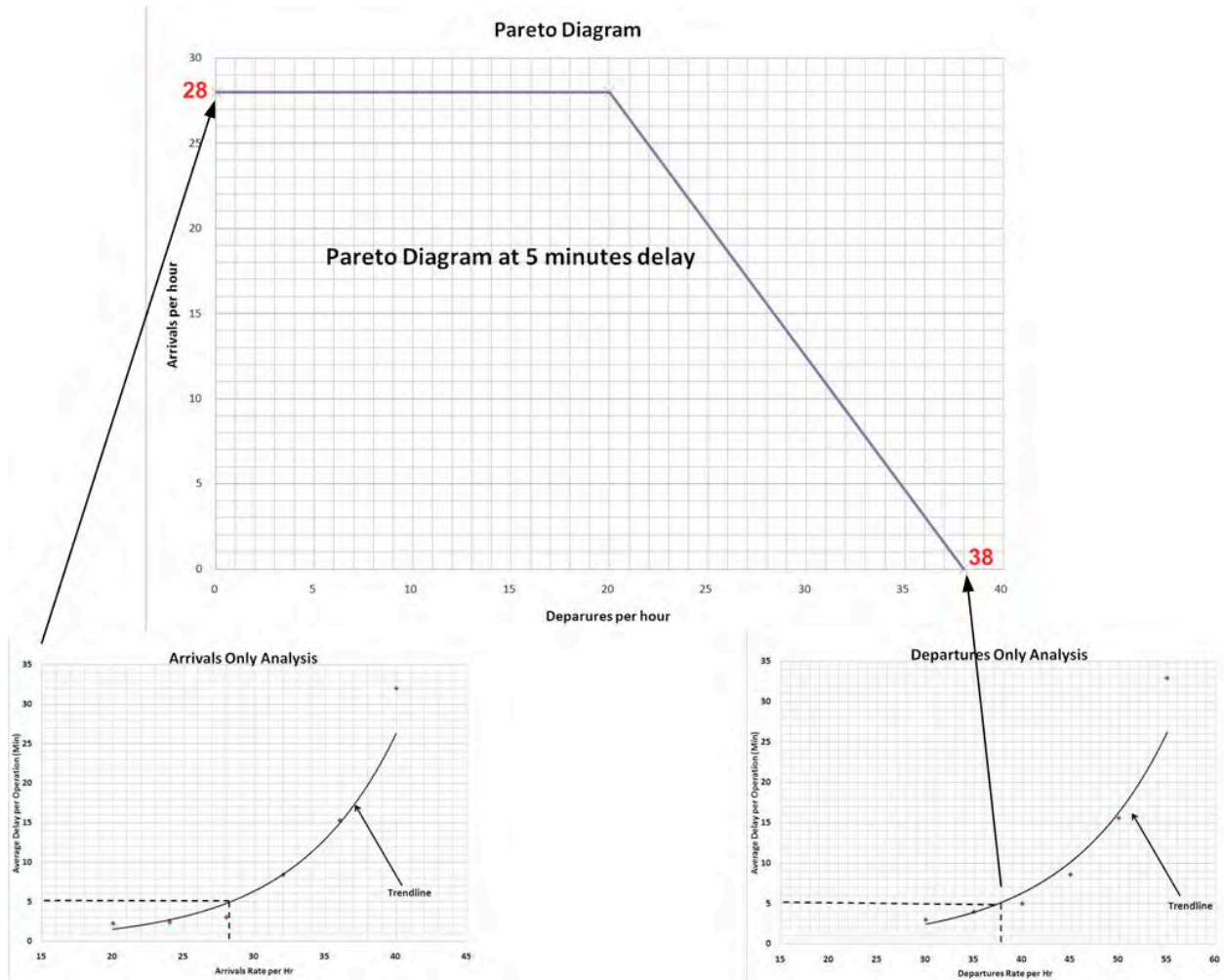
The final result of applying this balancing technique in terms of demand is showed in Table 3.12. The underlined cell represent the over capacity values that cannot be reduced because they come directly from the ATPO-P data set.

**Table 3-12 Demand After the Balancing Process.**

Balanced Demand					
Hourly Split	31L ARR	31R ARR	31L DEP	31L Total	Total Operations
00:00 - 01:00	12	0	3	15	15
01:00 - 02:00	9	2	2	11	13
02:00 - 03:00	7	5	4	11	16
03:00 - 04:00	8	6	2	10	16
04:00 - 05:00	3	11	1	4	15
05:00 - 06:00	12	12	6	18	30
06:00 - 07:00	4	20	16	20	40
07:00 - 08:00	5	19	27	32	51
08:00 - 09:00	0	19	41	41	60
09:00 - 10:00	0	14	37	37	51
10:00 - 11:00	7	25	23	30	55
11:00 - 12:00	8	22	27	35	57
12:00 - 13:00	6	19	19	25	44
13:00 - 14:00	7	19	15	22	41
14:00 - 15:00	10	26	16	26	52
15:00 - 16:00	8	22	18	26	48
16:00 - 17:00	7	19	21	28	47
17:00 - 18:00	0	16	43	43	59
18:00 - 19:00	0	32	44	44	76
19:00 - 20:00	0	15	39	39	54
20:00 - 21:00	4	9	18	22	31
21:00 - 22:00	3	11	26	29	40
22:00 - 23:00	1	3	21	22	25
23:00 - 24:00	4	0	8	12	12
Tot	125	346	477	602	948

### **3.10. Capacity Analysis**

The second analysis that we performed at the three major New York airports was to evaluate the saturation capacity in terms of operations per hour that could be processed with the implementation of the Data-Communication technologies. The analysis was modeled by creating an artificial demand based on the real aircraft mix that actually operates at each airport. Then the demand was gradually increased to evaluate the number of acceptable operations with a tolerable level of delay. The saturation capacity is obtained when the level of delay reaches 5 minutes per operation consistent with airport planning studies (Odoni 2003).



**Figure 3.15 Example of Capacity Diagram Calculation.**

To construct a Pareto diagram we investigated three points: arrivals only, departures only and 50/50 percent arrivals and departures for EWR and LGA. For JFK were evaluated points across two lines of capacity represented in Table 3.15.

**Table 3-13 La Guardia Airport Demand Levels Tested.**

<b>LGA</b>		
<b>DEP Only</b>	<b>ARR Only</b>	<b>Mixed Ops</b>
20 per Hour	20 per Hour	25/25 per Hour
25 per Hour	24 per Hour	30/30 per Hour
30 per Hour	28 per Hour	39/39 per Hour
35 per Hour	32 per Hour	45/45 per Hour
40 per Hour	36 per Hour	
45 per Hour	40 per Hour	
50 per Hour		

The demand levels tested were selected by probing real capacity of the airports in question stated in the FAA Benchmark Report of 2004 (FAA 2004). In Newark the mixed operations evaluated were up to 50/50 per hour, whereas in La Guardia due to intersecting runway operations the maximum number of operations level tested 45/45 per hour.

**Table 3-14 Newark Airport Demand Levels Tested.**

<b>EWR</b>		
<b>DEP Only</b>	<b>ARR Only</b>	<b>Mixed Ops</b>
20 per Hour	20 per Hour	25/25 per Hour
25 per Hour	24 per Hour	30/30 per Hour
30 per Hour	28 per Hour	42/42 per Hour
35 per Hour	32 per Hour	45/45 per Hour
40 per Hour	36 per Hour	50/50 per Hour
45 per Hour	40 per Hour	
50 per Hour		

The runway configuration tested at JFK included two arrival runways (31L and 31R) and one departure (31L) runway. This is the most used configuration at this airport.

Table 3-15 JFK Airport Demand Levels Tested.

JFK			
DEP Only	ARR Only	Mixed Ops (line 1) ARR/DEP	Mixed Ops (line 2) ARR/DEP
20 per Hour	20 per Hour	40/15 per Hour	20/60 per Hour
25 per Hour	24 per Hour	50/20 per Hour	27/65 per Hour
30 per Hour	28 per Hour	56/26 per Hour	30/70 per Hour
35 per Hour	32 per Hour	60/30 per Hour	
40 per Hour	36 per Hour		
45 per Hour	40 per Hour		
50 per Hour			

Each of the levels of demand tested included a baseline scenario, Data-Link Segment 1 and Data-Link Segment 2.

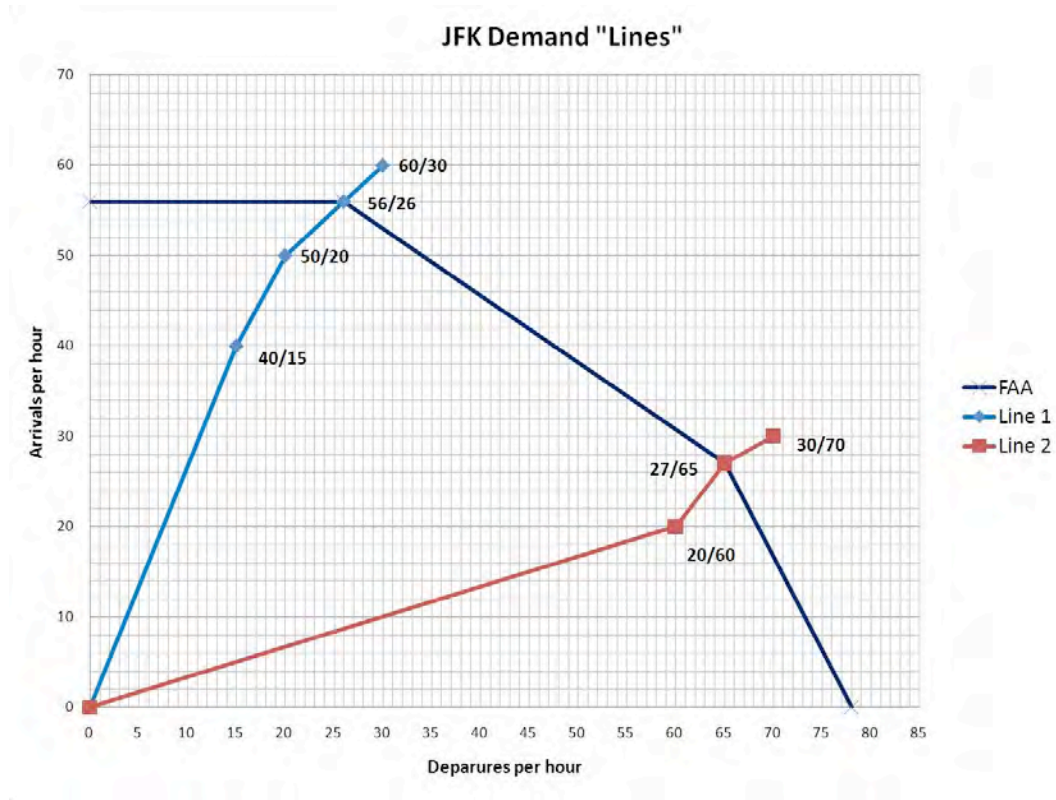


Figure 3.16 Mixed Operation Demands Tested.

### 3.11. Aircraft Analysis

The aircraft mix of each airport was derived from the FAA ATO-P demand files. Of the three airports simulated, LaGuardia has the most homogenous aircraft with no Heavy aircraft operating. The aircraft mix is shown in Figure 3.17.

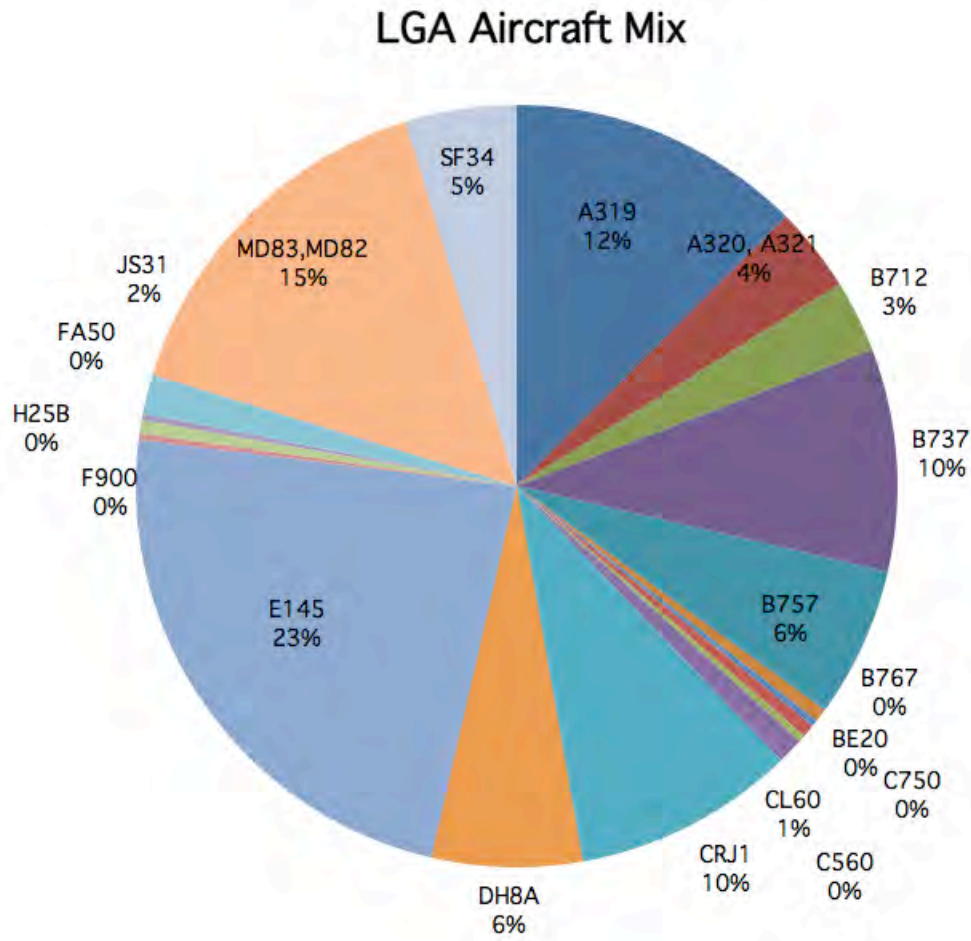
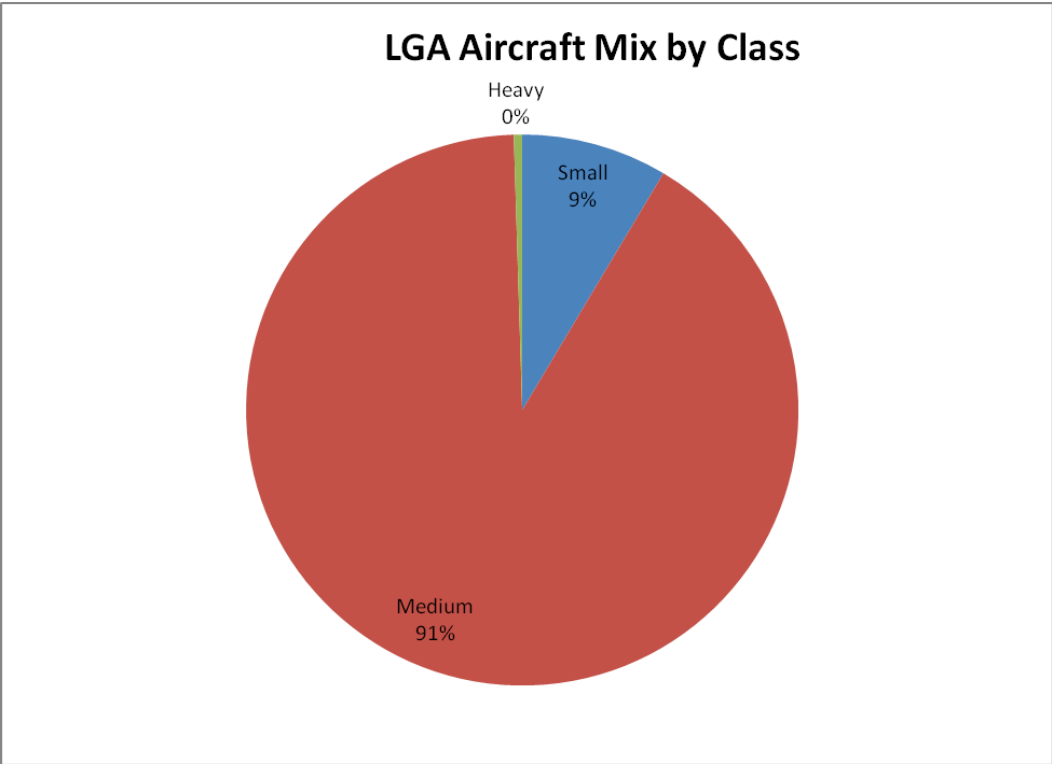


Figure 3.17 LGA Aircraft Mix.



**Figure 3.18 LGA Aircraft Mix by Weight Class.**

JFK has the higher percentage of heavy aircraft, with almost 30%. This is due to high number of international flights operating at this airport.

### JFK Aircraft Mix

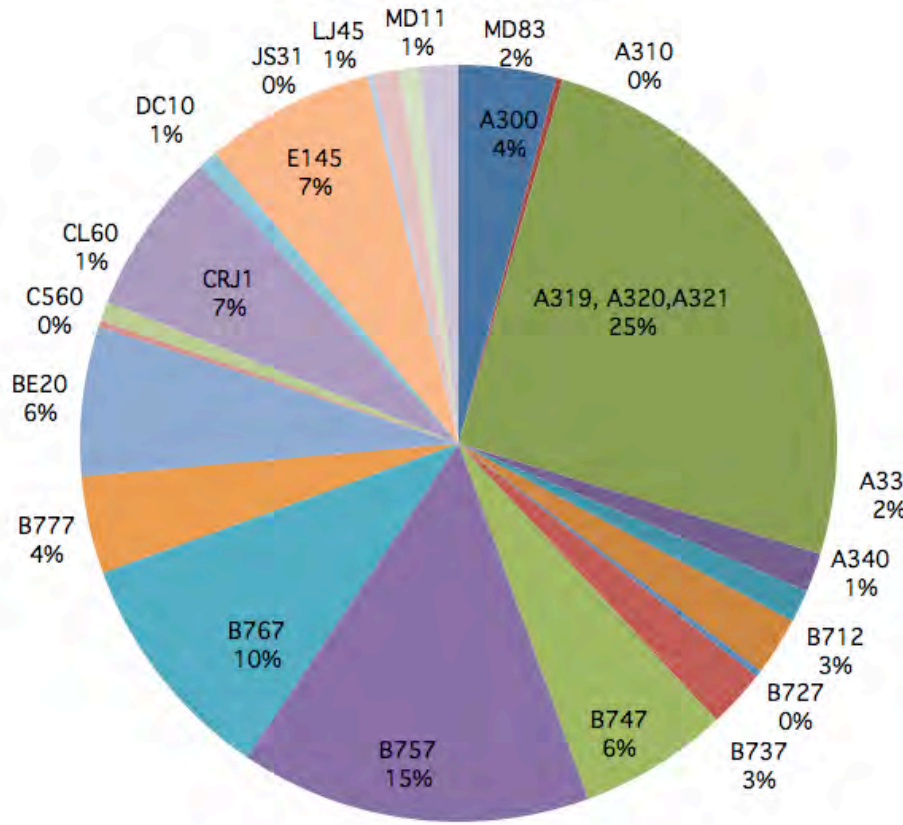
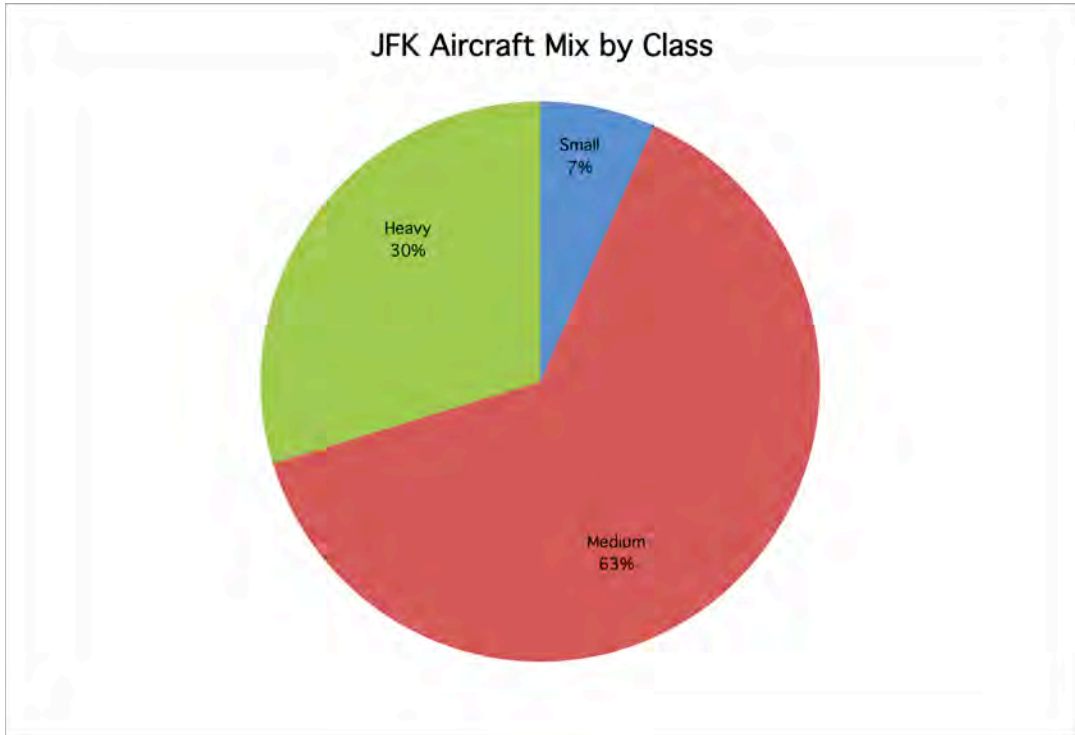


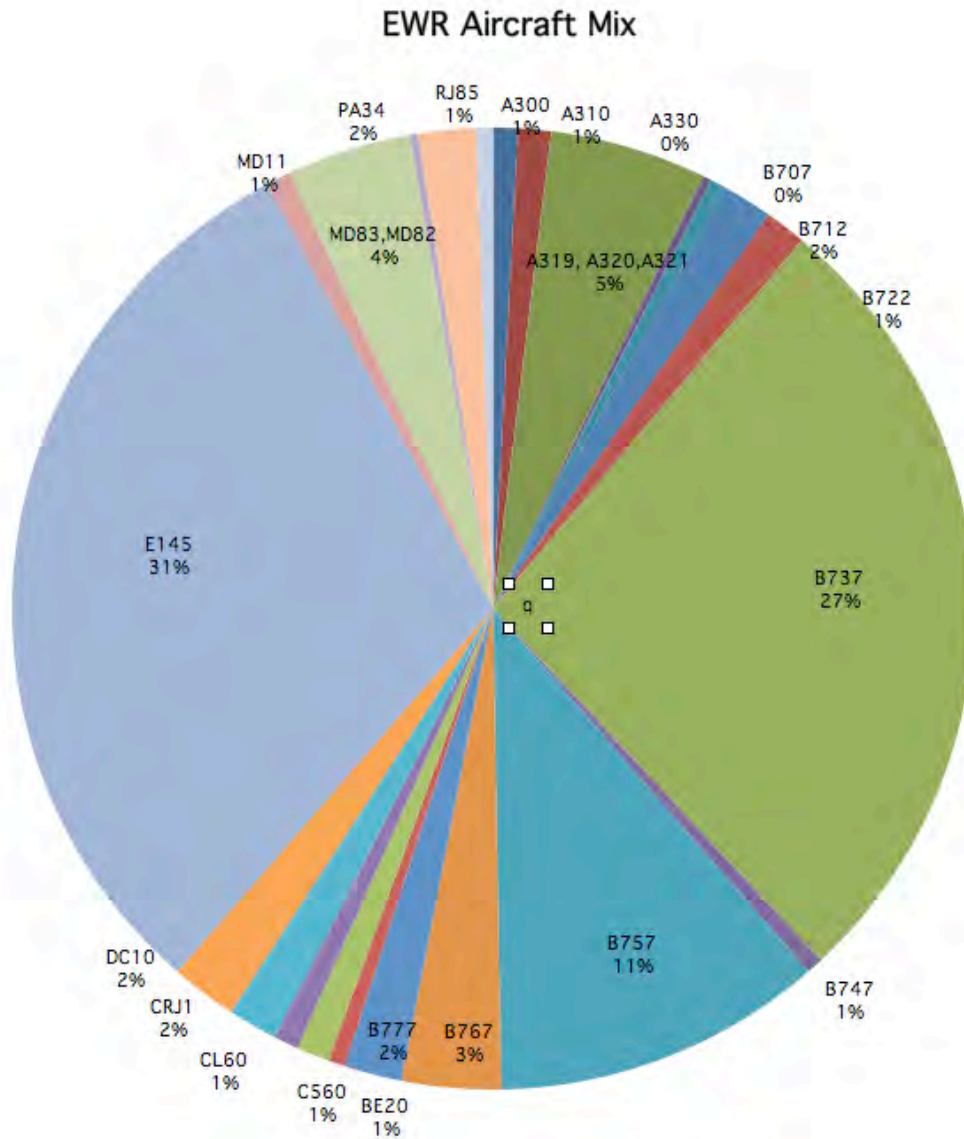
Figure 3.19 JFK Aircraft Mix.





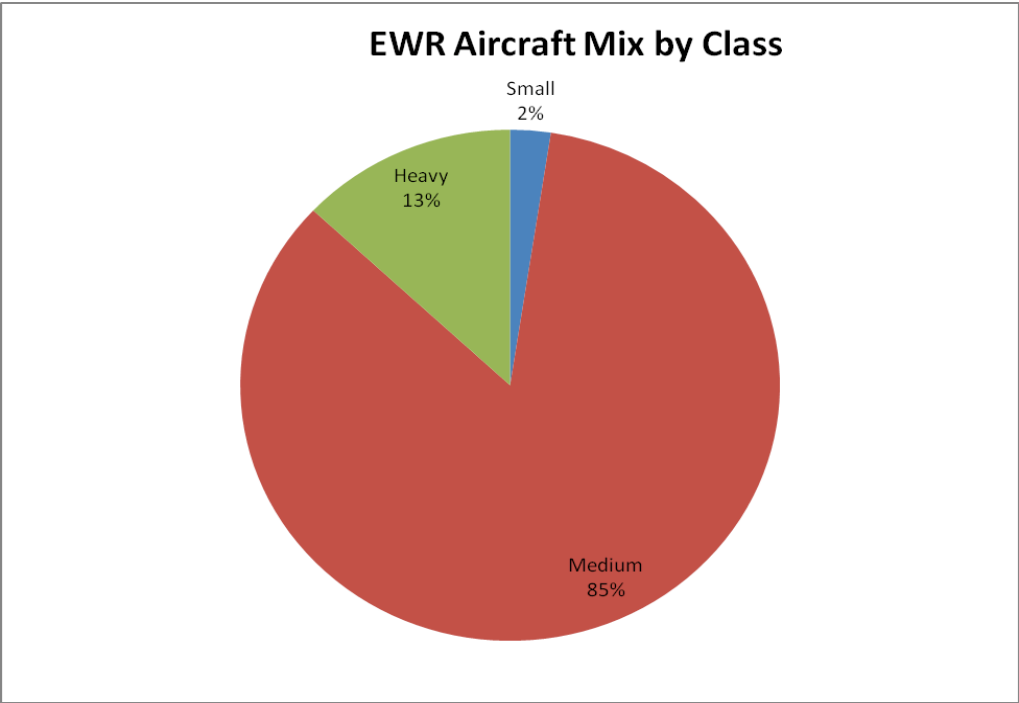
**Figure 3.20 JFK Aircraft Mix by Weight Class.**

Newark has the lowest percentage of small aircraft operating with only 2%. The majority of the aircraft operating at EWR are medium aircraft with numerous Embraer 145 and Boeing 737 (57% of population combined).



**Figure 3.21 EWR Aircraft Mix.**

It is interesting to notice how certain older aircraft like DC10 and Boeing 707, operate as freighters in small percentage from EWR.



**Figure 3.22 EWR Aircraft Mix by Weight Class.**

## 4. RESULTS

### 4.1. Capacity Analysis

There are two definitions of airport capacity that can be found in the literature (Hockaday 1974; Horonoeff 1993; Janic 2000):

- “Ultimate” or “Saturation” capacity, and
- “Practical” capacity

Ultimate capacity is the maximum number of aircraft that can be served in a given period of time with a constant demand for service. “Practical” capacity can be defined, as the maximum number of aircraft that can be served in a period of time considering the average delay experienced by each aircraft. Generally the level of permissible delay is prescribed in the analysis.

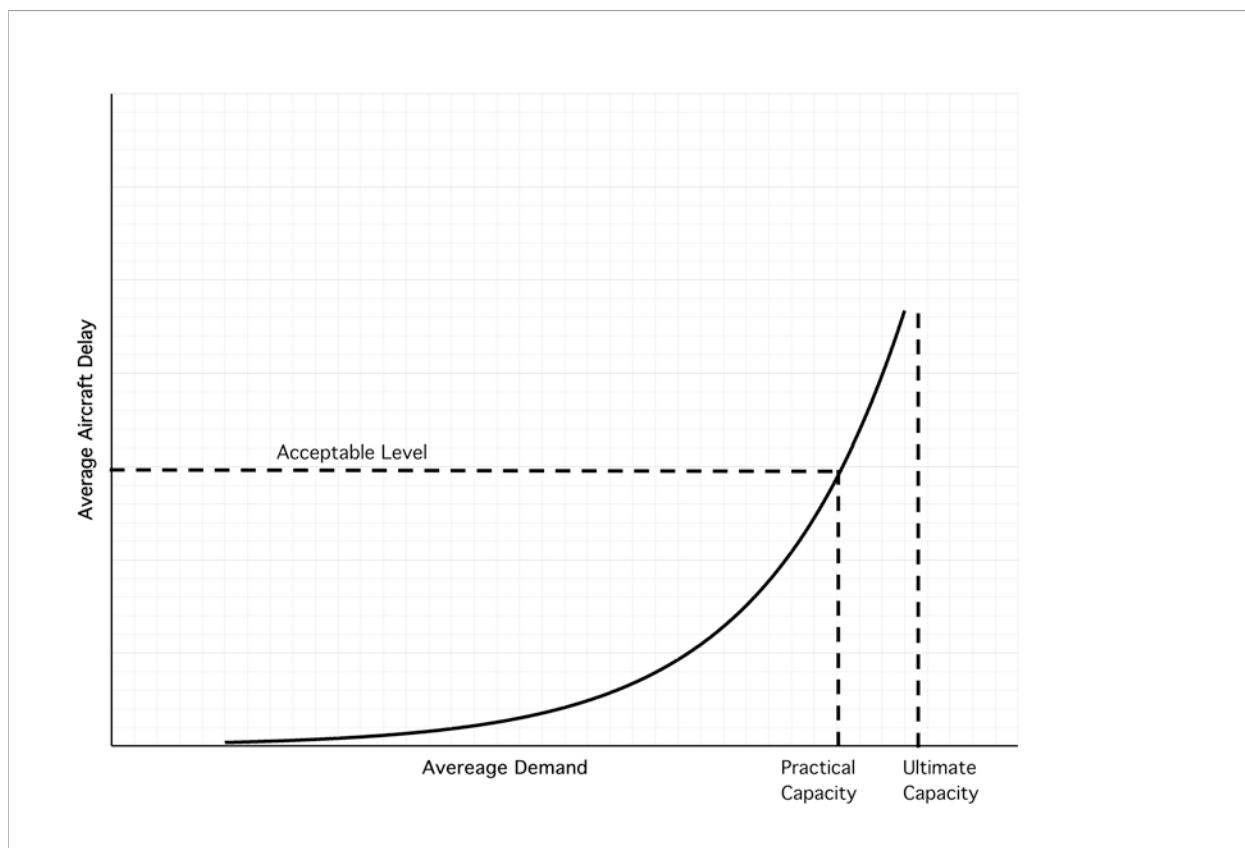


Figure 4.1 Relationship Between Delay-Related and Ultimate Capacity.

In our analysis we considered the practical capacity at an acceptable level of delay per operation of five minutes. To calibrate our model we used the results presented in the 2004 FAA Benchmark Capacity Report (FAA 2004). The analysis presented in the FAA Benchmark Report uses the Airfield Capacity Model (Swedish 1981). The ACM model does not consider an acceptable level of delay it is impossible to know the “practical” capacity from the FAA Benchmark Report. As a rule of thumb the “practical” capacity is considered to be around 80-90 percent of the maximum throughput capacity, depending of the specific conditions (Odoni 2003). Using such assumption the results of our simulation study correlate with the Benchmark numbers. These will be explained on a case-by-case.

The analysis was repeated for three different scenarios, the baseline, and two implementation of Data-Communications for segments 1 and 2. The introduction of communication new technologies was simulated changing two parameters in the simulation: the Miles-In-Trail separation (MIT) and the values of the minimum separation matrix. The MIT separation was reduced from 9 miles (baseline), to 8 and 7 miles to understand their effect on delays. These values were extrapolated from a previous study (Smith 2004). The separation matrix was reduced from the baseline values of 5% in the future scenarios according to the study (Smith 2004).

**Table 4-1 Wake Vortex Separation Matrix Baseline Scenario.**

		Leading		
		Light	Medium	Heavy
Trailing	Light	2.5	3.5	4
	Medium	2.5	2.7	3.7
	Heavy	2.5	3	3.5

**Table 4-2 Wake Vortex Separation Matrix Data-Communication Scenarios.**

		Leading		
		Light	Medium	Heavy
Trailing	Light	2.38	3.33	3.8
	Medium	2.38	2.57	3.52
	Heavy	2.38	2.85	3.33

In order to calculate the Capacity Diagram using the delay curves generated at airport, the first operational condition that was studied was the “arrivals only” case. A constantly increasing number of arrivals was loaded into the Airport models in RAMS. The runway configurations simulated for the three airports were the following:

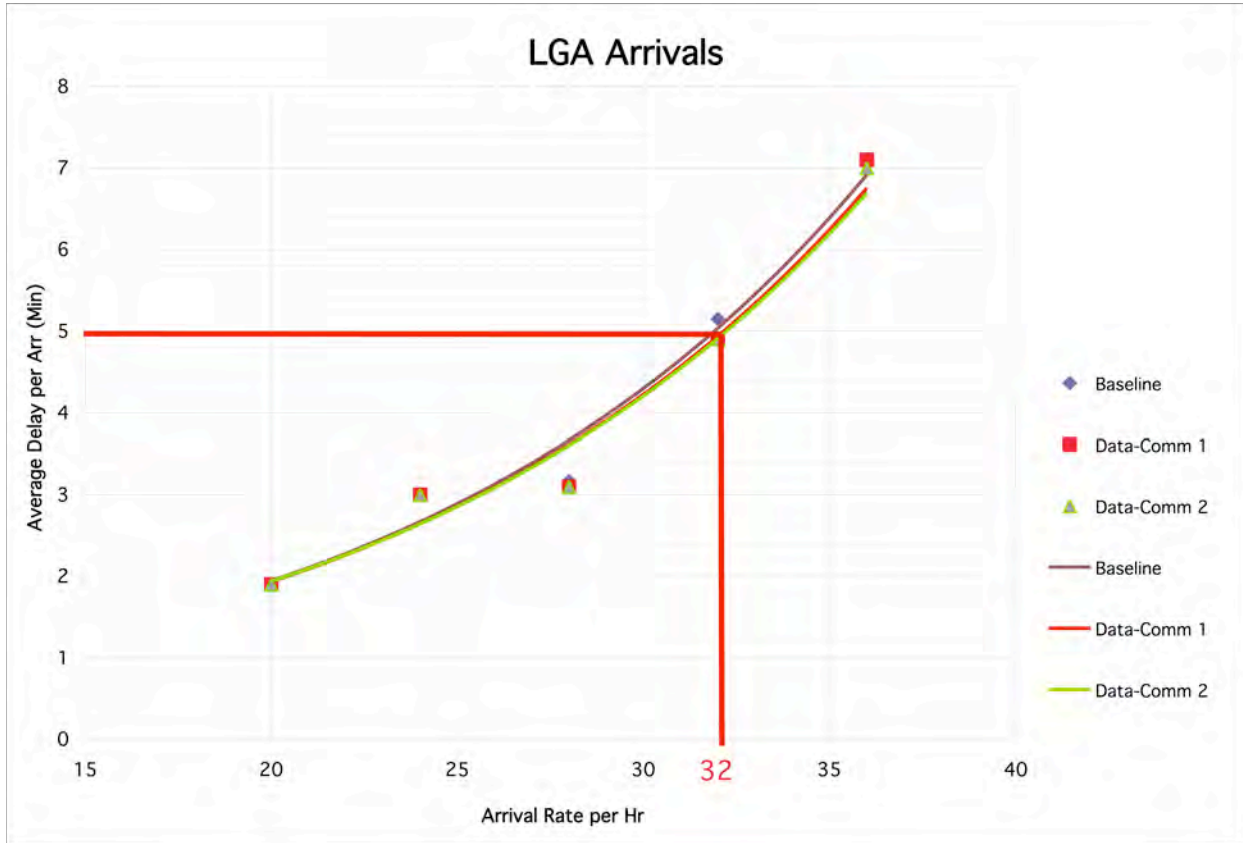
**Table 4-3 Simulated Scenarios.**

<b>Airport</b>	<b>Runway Set (ARR/DEP)</b>
LGA	22   31
EWR	22L   22R
JFK	31L, 31R   31L

JFK was simulated with ‘arrivals only” and “departures only” on a single runway; the independency of the operations between runway 31L and 31R made this possible. The results for a single runway were the doubled to obtain the total value. For the “mixed operations” analysis the complete runway configuration was simulated.

## 4.2. Arrivals Only Analysis

### 4.2.1. La Guardia Airport



**Figure 4.2 LGA “Arrivals Only” Results.**

The LGA simulation provided an improvement of one arrival per hour with the implementation of the new technologies. With an acceptable level of delays set to 5 minutes, the arrivals only capacity for La Guardia was calculated to be equal to 32 per hour for the baseline scenario, and with the implementation of Data-Communication. Between the segments 1 and 2 there was no capacity difference observed.

### 4.2.2. Newark Liberty Airport

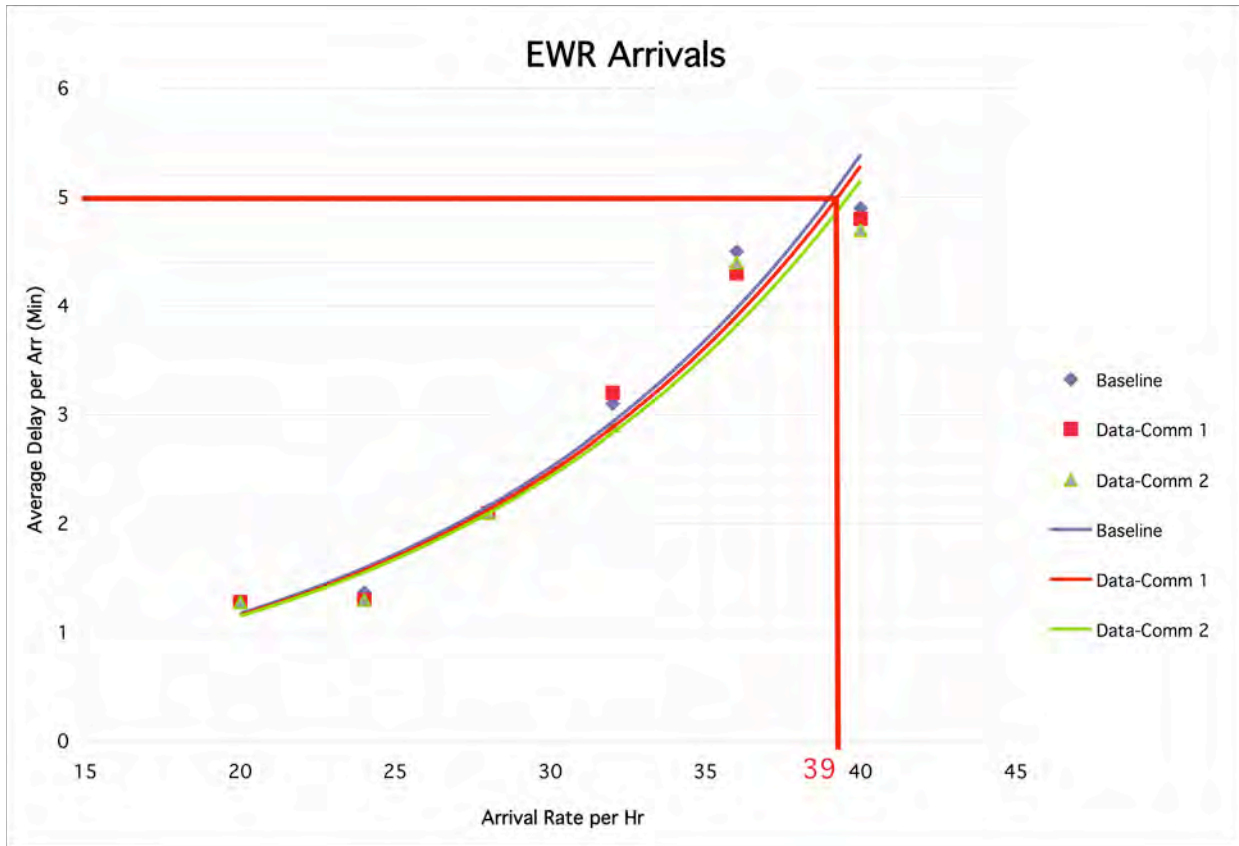


Figure 4.3 EWR “Arrivals Only” Results.

At Newark with the implementation of the Data-Communication technologies a gain of 0.7 operations per hour was found. For this airport the arrivals only capacity calculated was 39 per hour for the baseline, and 39.7 with Data-Communication segments 1 and 2.



### 4.2.3. John Fitzgerald Kennedy Airport

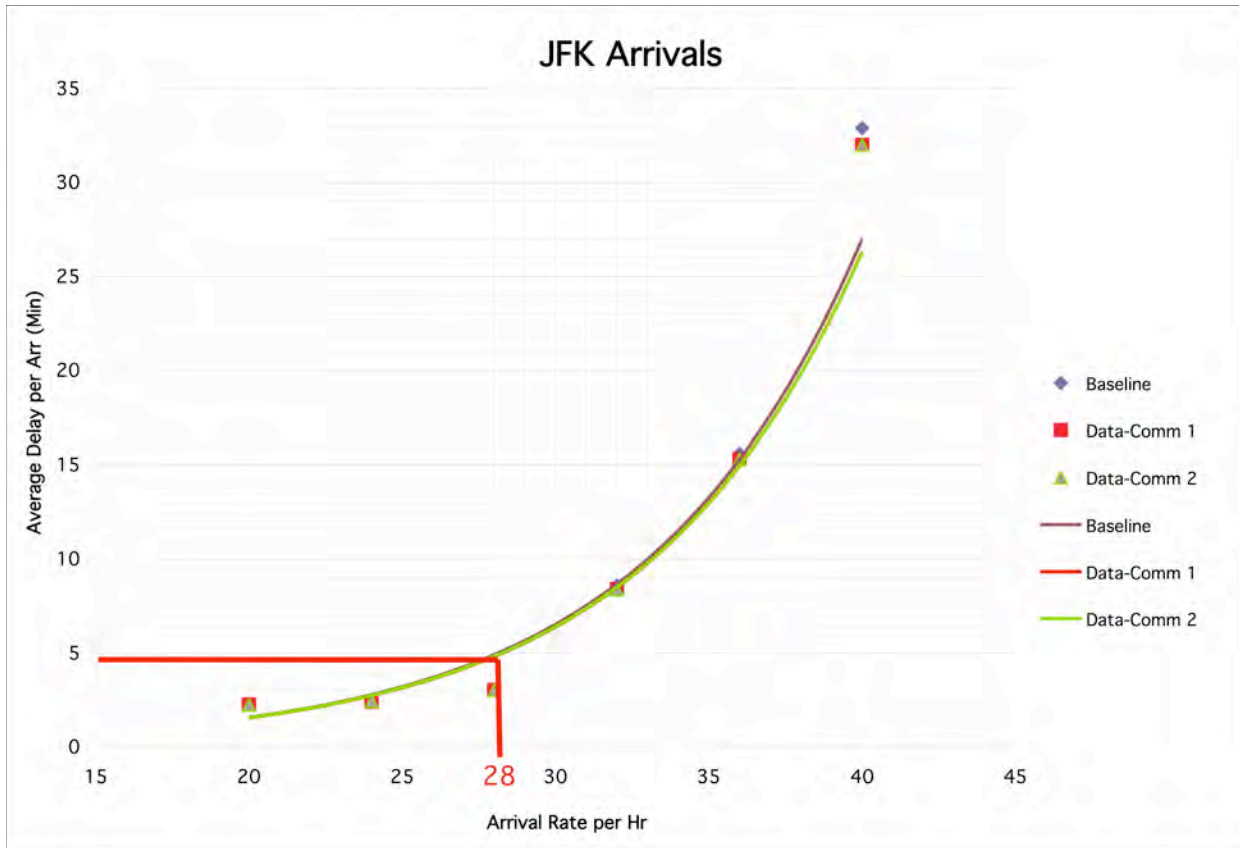


Figure 4.4 JFK “Arrivals Only” Results.

The simulation with the future implementation of Data-Communication at JFK did not produce any substantial benefits. The practical capacity at JFK calculated through RAMS simulation was 28 arrivals per hour for a single runway. The capacity of JFK can be doubled considering that the airport can operate two independent runways (e.g., runways are separated by 6,700 feet). Therefore the “arrivals only” point in the capacity diagram calculated was to be 56.

### 4.3. Departures Only Analysis

The departure analysis produced modest improvements with the Implementation of the data-communication technologies.

#### 4.3.1. La Guardia Airport

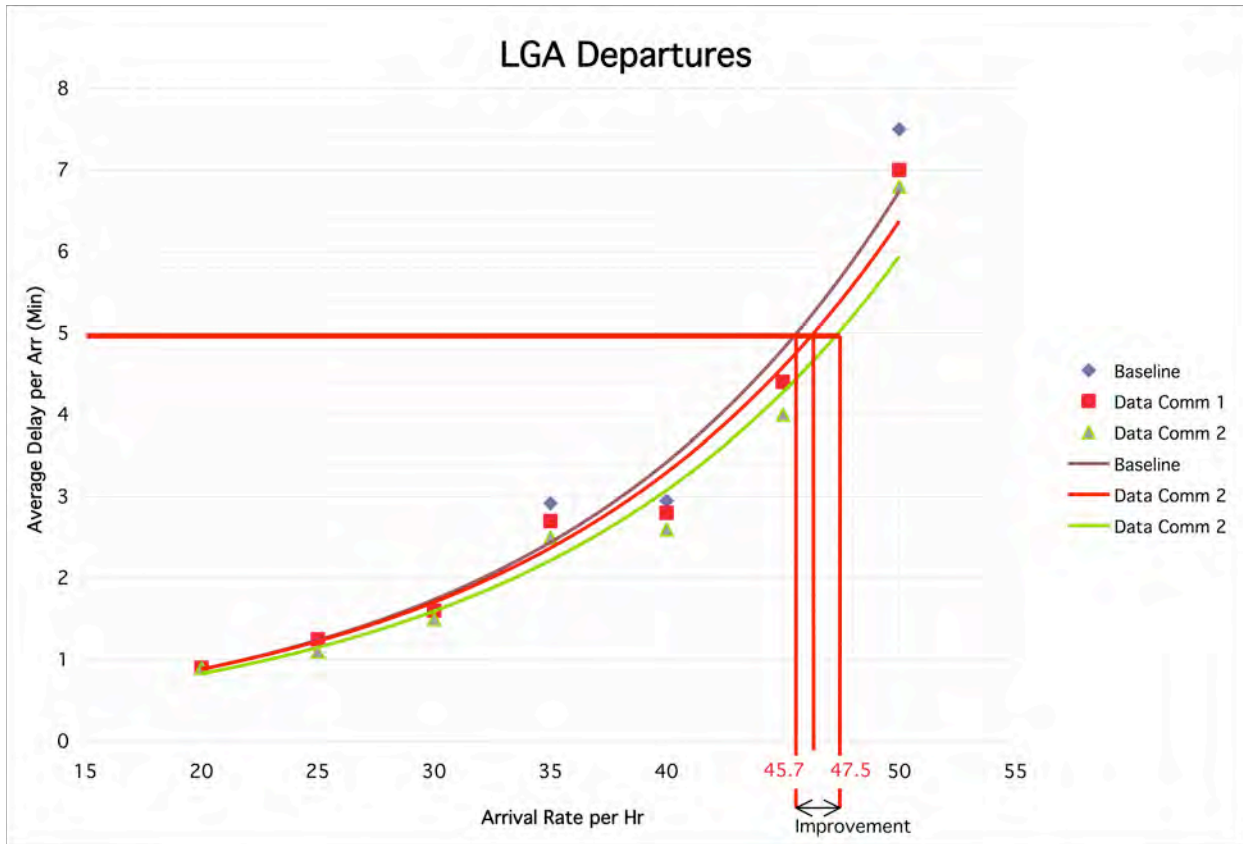


Figure 4.5 LGA “Departures Only” Results.

The improvements in the “departures only” operations were modest. In fact the “practical” capacity obtained from the simulation was 45.7 for the baseline, 46.5 for the Data-Comm segment 1, and 47.5 for segment 2. The total departures-only improvement is two operations per hour or to 4.4 percent.

### 4.3.2. Newark Liberty Airport

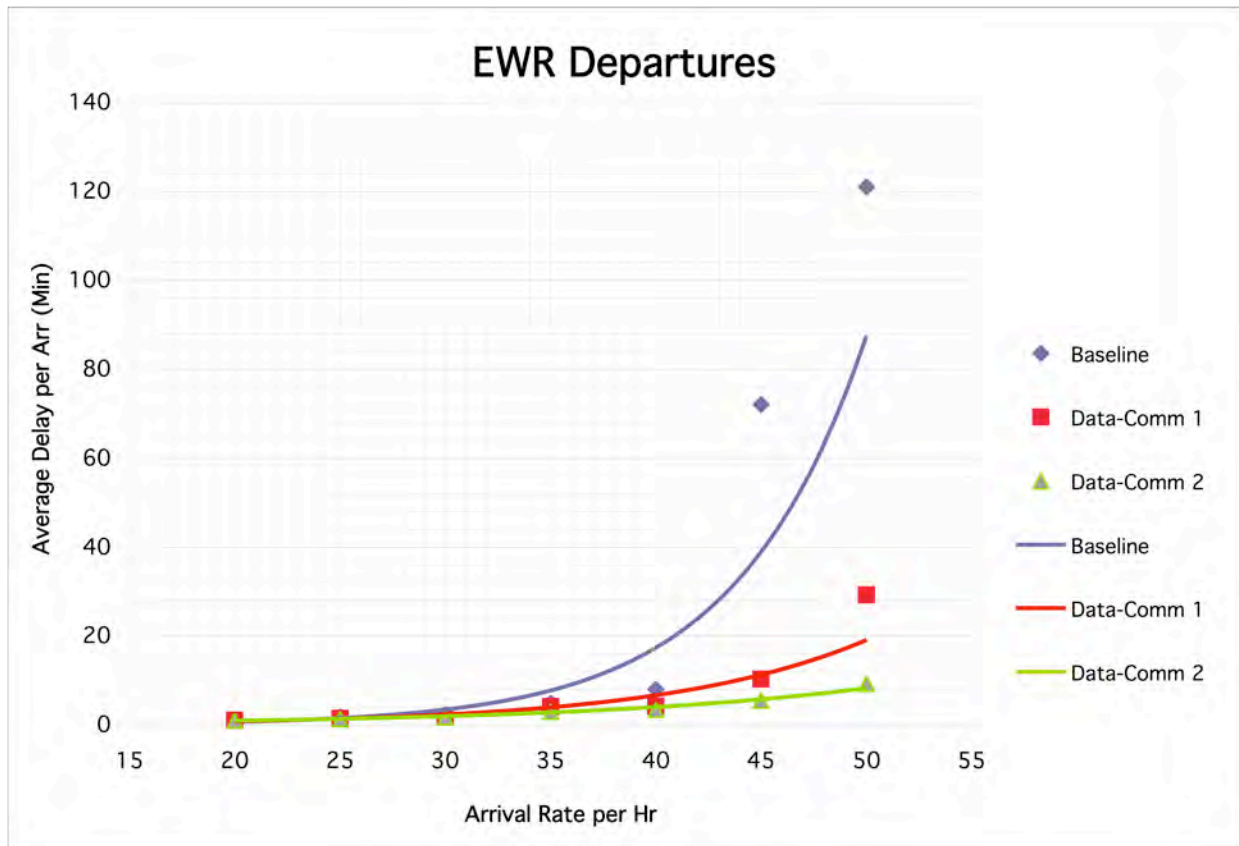


Figure 4.6 EWR “Departures Only” Results.

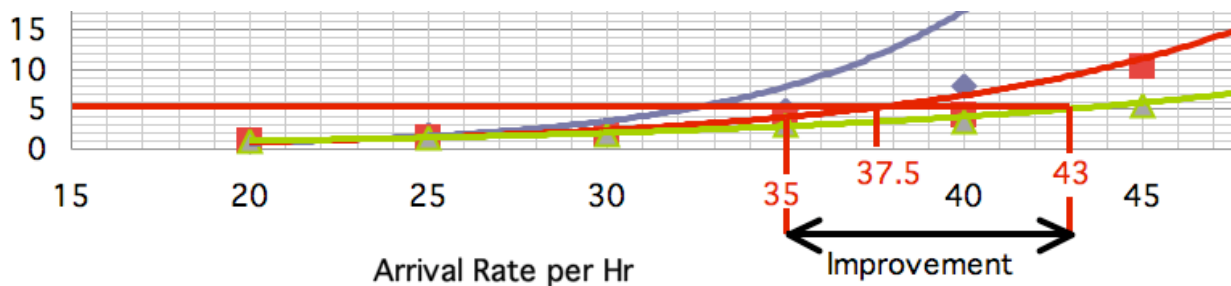


Figure 4.7 EWR “Departures Only” Results, Area of Interest.

The improvement at EWR is somehow surprising. The practical capacity calculated in RAMS was 35 departures per hour for the baseline scenario, 37.5 with Data-Comm segment 1 and 43 with segment 2. The total improvement was close to 23 percent.

### 4.3.3. John Fitzgerald Kennedy Airport

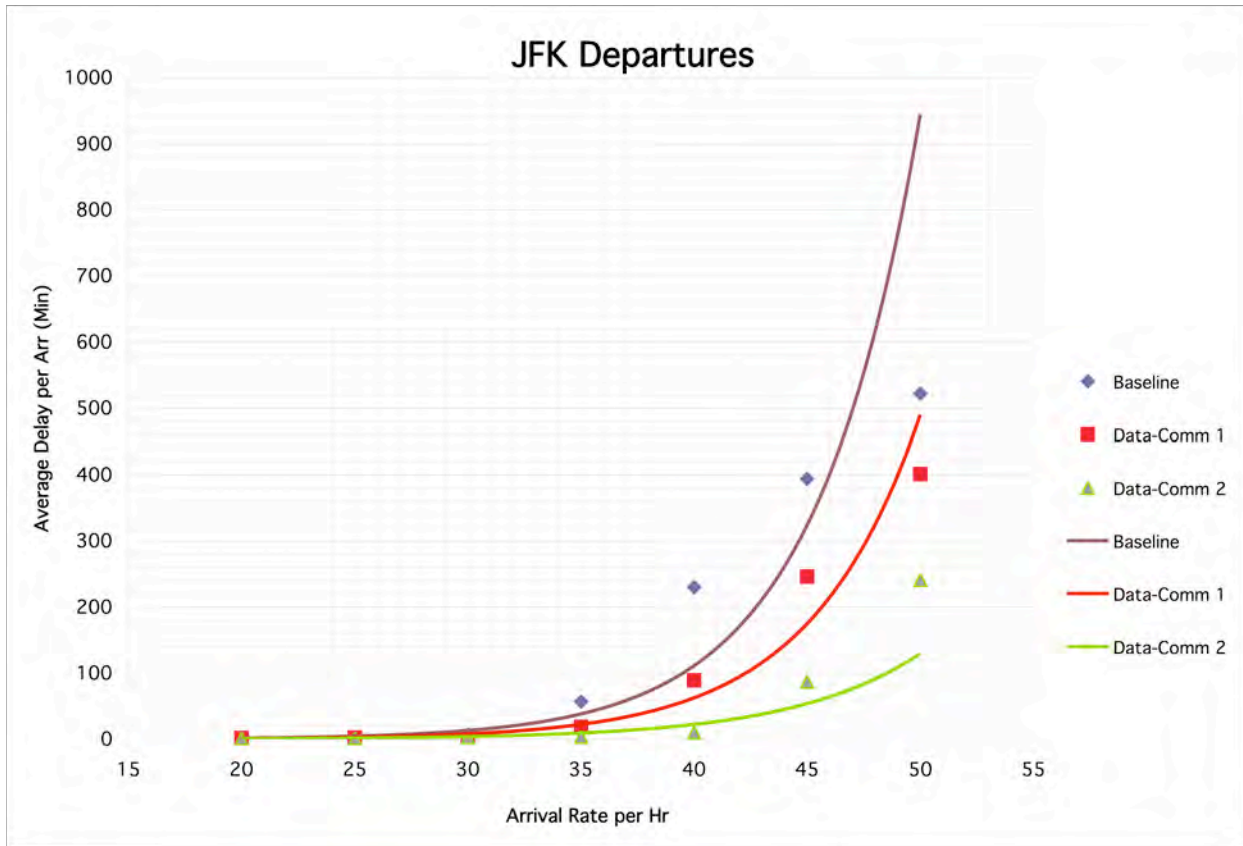


Figure 4.8 JFK “Departures Only” Results.

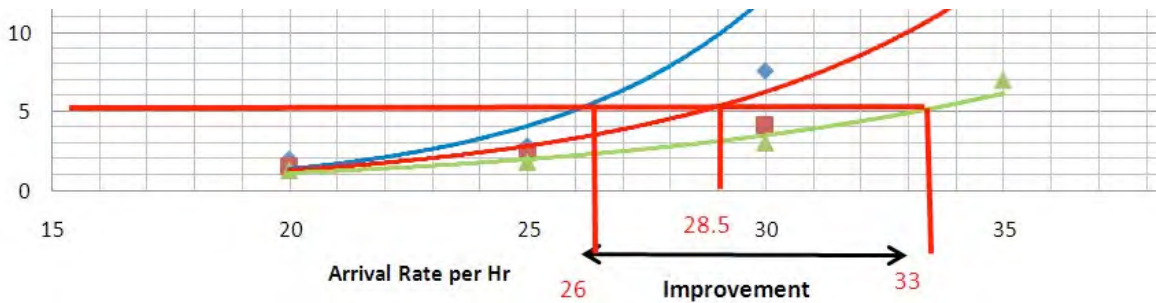


Figure 4.9 JFK “Departures Only” Results, Area of Interest.

Data-Communication technologies also showed good benefits in the JFK simulation. In fact from the baseline 26 departures per hour, the practical capacity calculated was 28.5 for segment 1 and 33 for segment 2, with an improvement of 27 percent.

## 4.4. Mixed Operations Analysis

### 4.4.1. La Guardia Airport

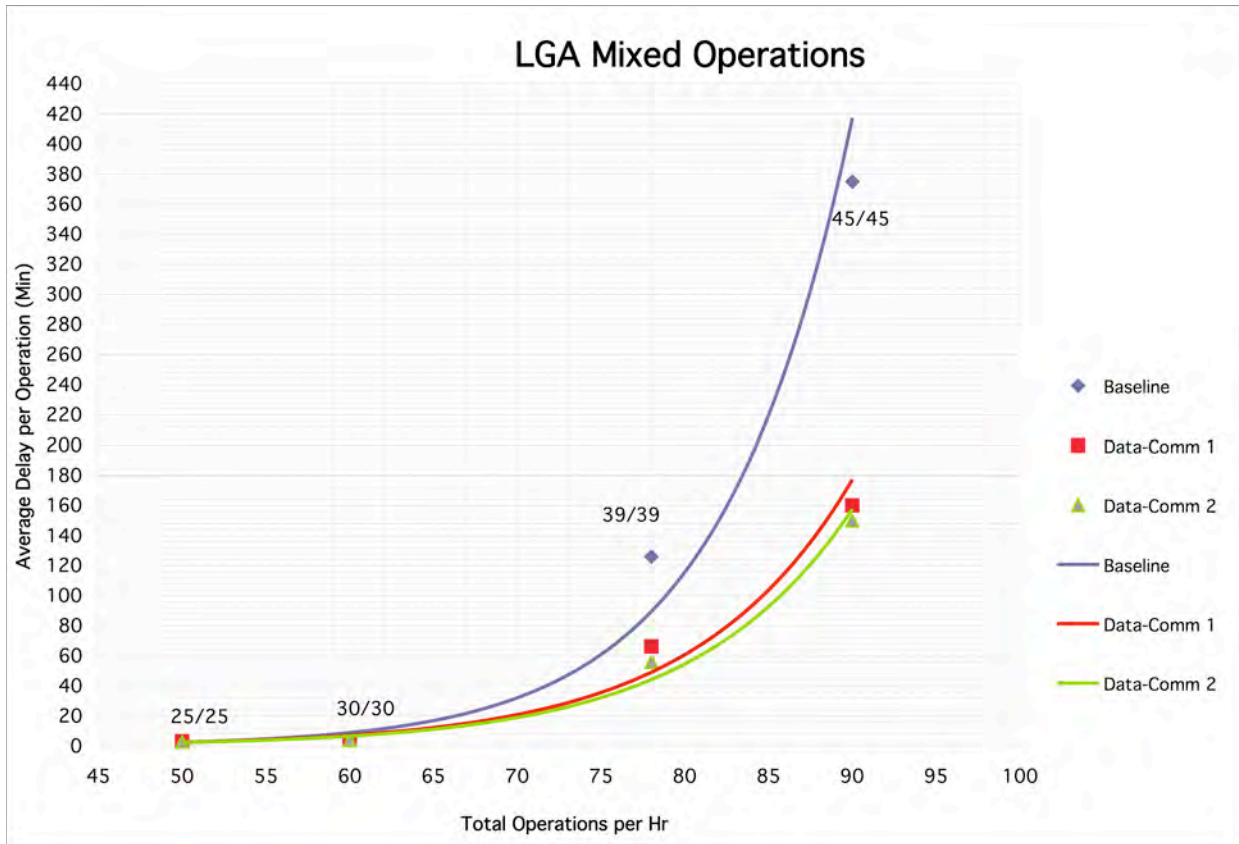


Figure 4.10 LGA Mixed Operation Results.



Figure 4.11 LGA Mixed Operation results, Area of Interest.

The practical capacity for mixed operations at LGA calculated using RAMS was 55 operations per hour for the baseline scenario (28/27 departures), 57 with Data-

Communication segment 1 (28/29), and 58 with Data-Communication segment 2 (29/29). The improvement was of 3 operations per hour equal or 5.1 percent.

#### 4.4.2. Newark Liberty Airport

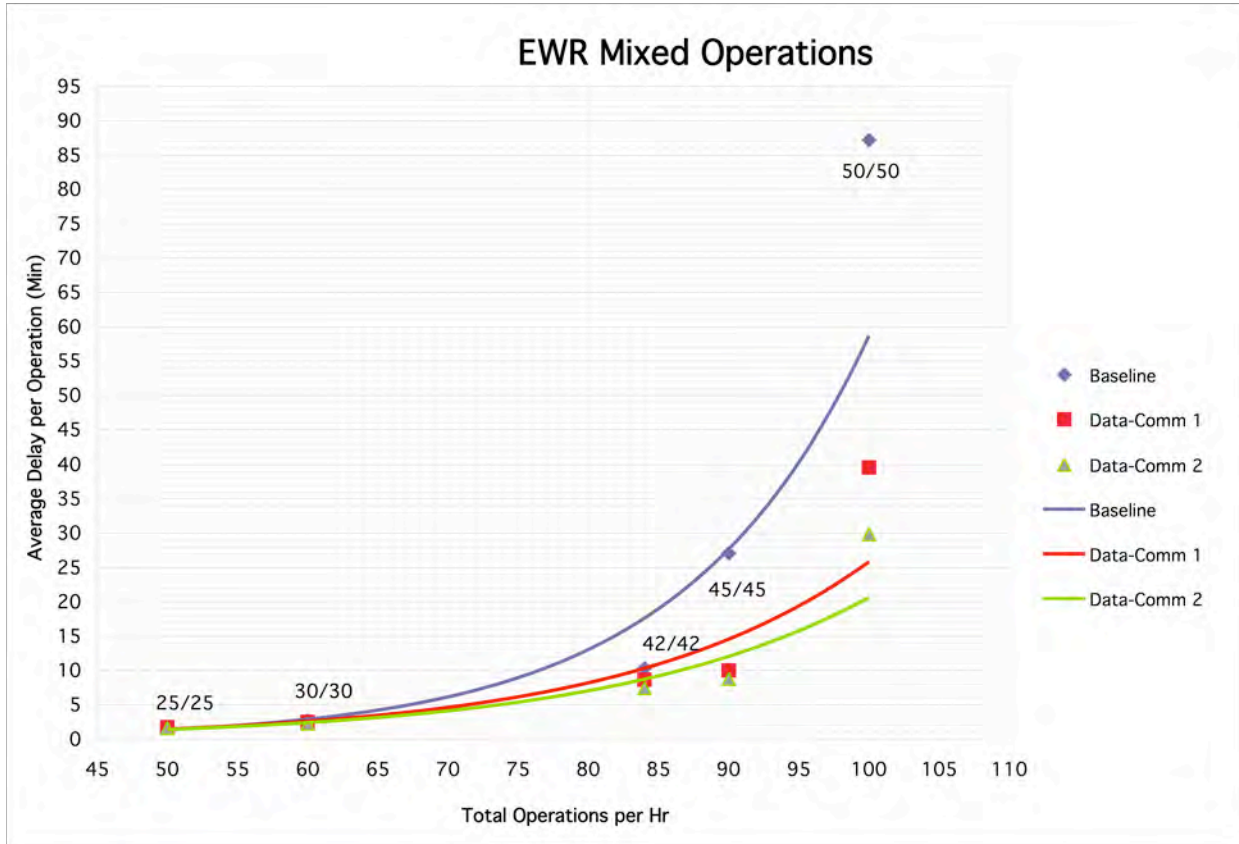


Figure 4.12 EWR Mixed Operation Results.

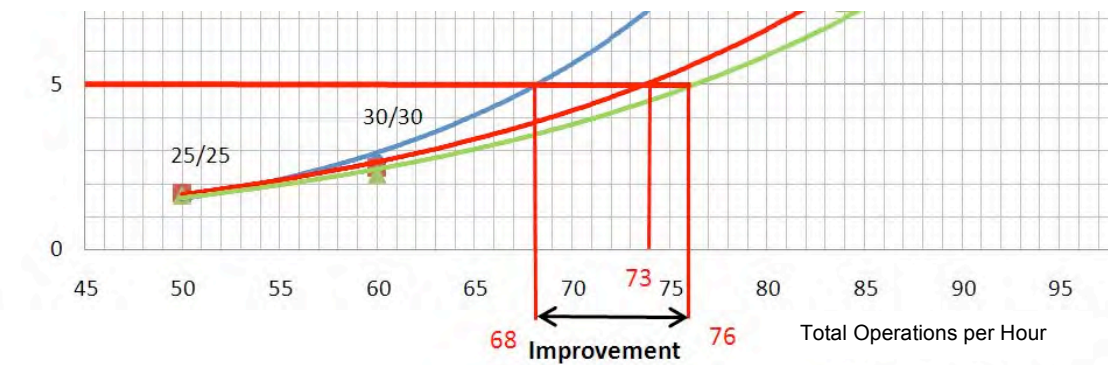


Figure 4.13 EWR Mixed Operation Results, Area of Interest.

The practical capacity at EWR using RAMS was calculated to be 68 operations per hour for the baseline scenario (34/34), 73 with Data-Comm segment 1 (37/36), and 76 for segment 2 (38/38). The total improvement was equal to 12 percent.

#### 4.4.3. John Fitzgerald Kennedy Airport

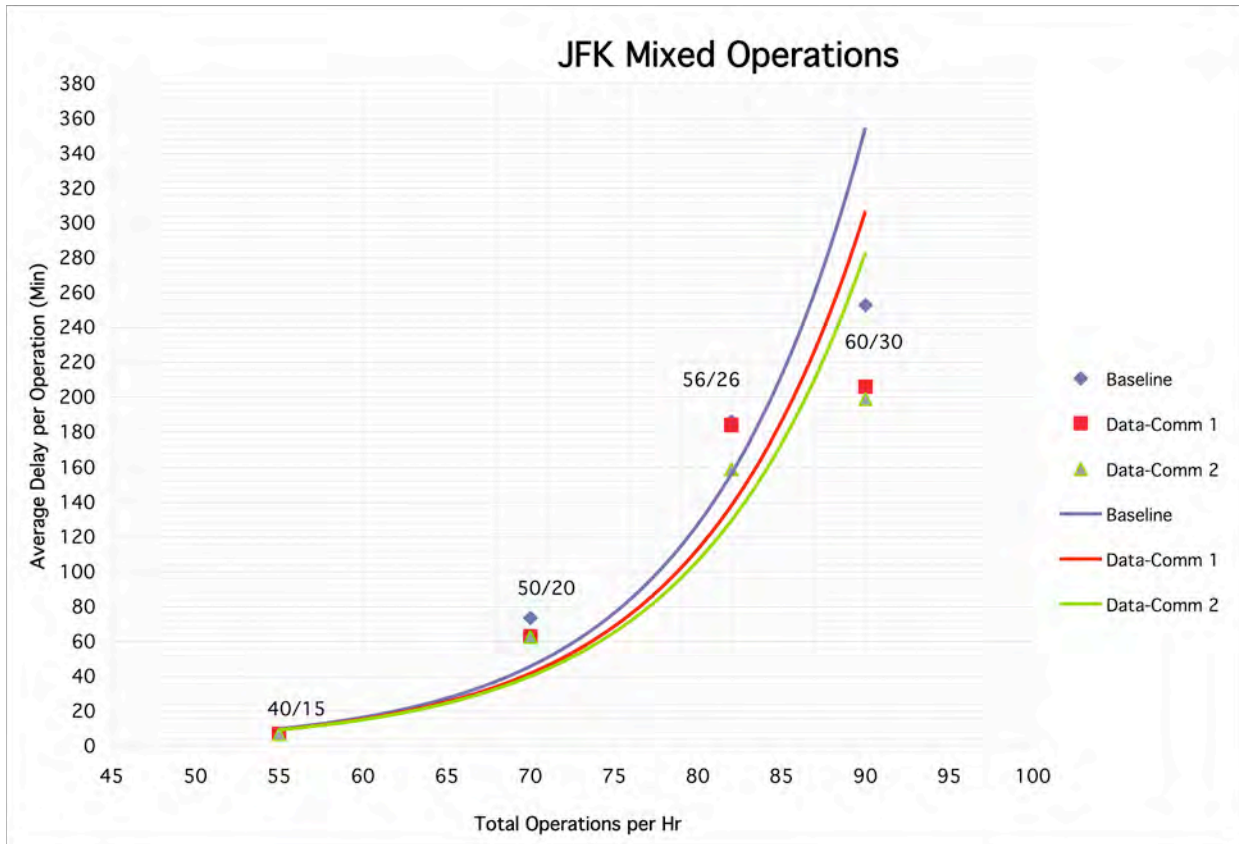
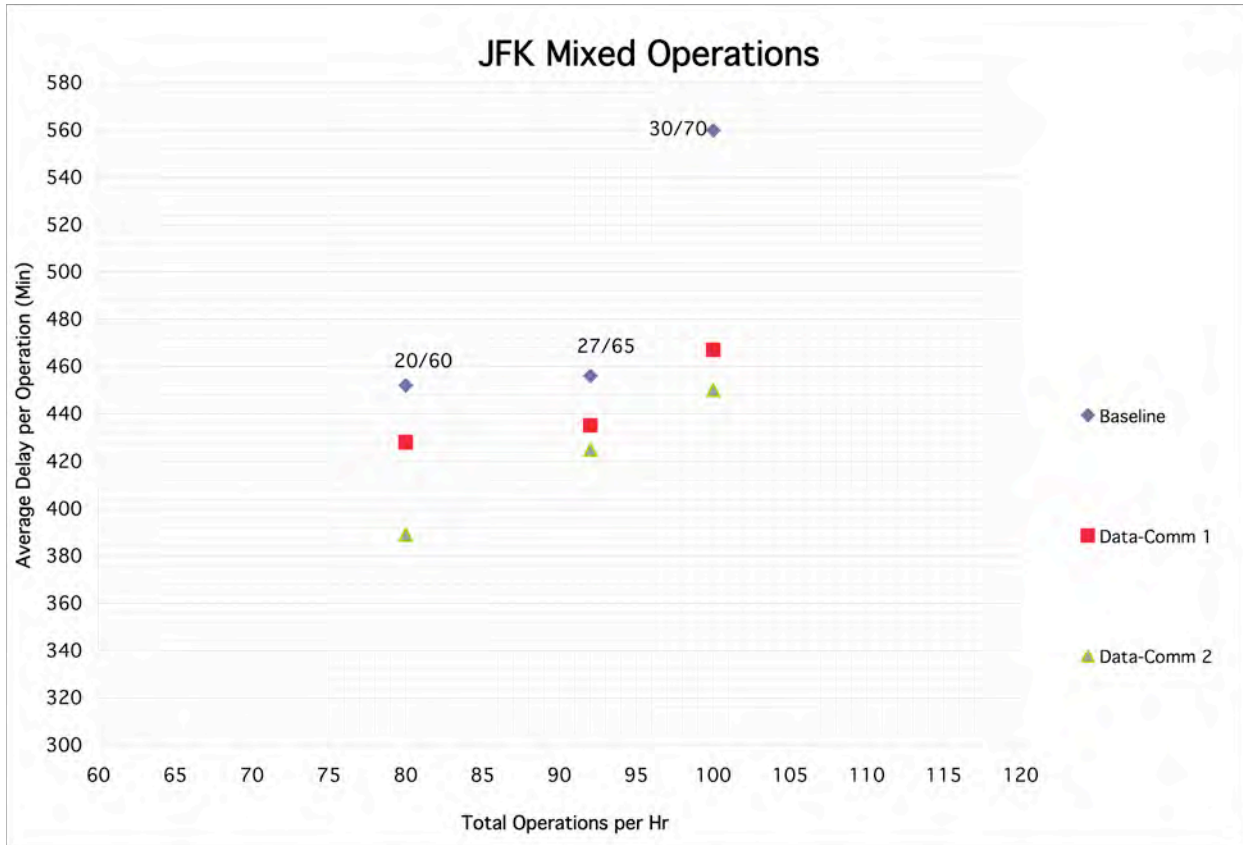


Figure 4.14 JFK Mixed Operation Results.

The choice of the arrival/departure combinations and relative points on the capacity diagram was based on the 2004 FAA Capacity Benchmark Report. Our analysis indicates that the benchmark shows better performance at New York airports. Capacity is estimated to be higher than the saturation capacity results. However, there are possible reasons for this discrepancy, the most important are:

- Combined Capacity Effects, and
- Runway Configurations Used.



**Figure 4.15 JFK Mixed Operations Results.**

The results considered acceptable for the mixed operations at JFK were only for 40 arrivals and 15 departures. The delay values for this combination were around six minutes, therefore they were considered as the middle point of the practical capacity diagram for all the three scenarios.

#### **4.5. Practical Capacity Envelope Diagrams**

As described in Chapter 3, the capacity envelope diagrams are obtained plotting the data calculated through the simulation runs made with RAMS for combinations of arrivals and departures.



### 4.5.1. La Guardia Airport

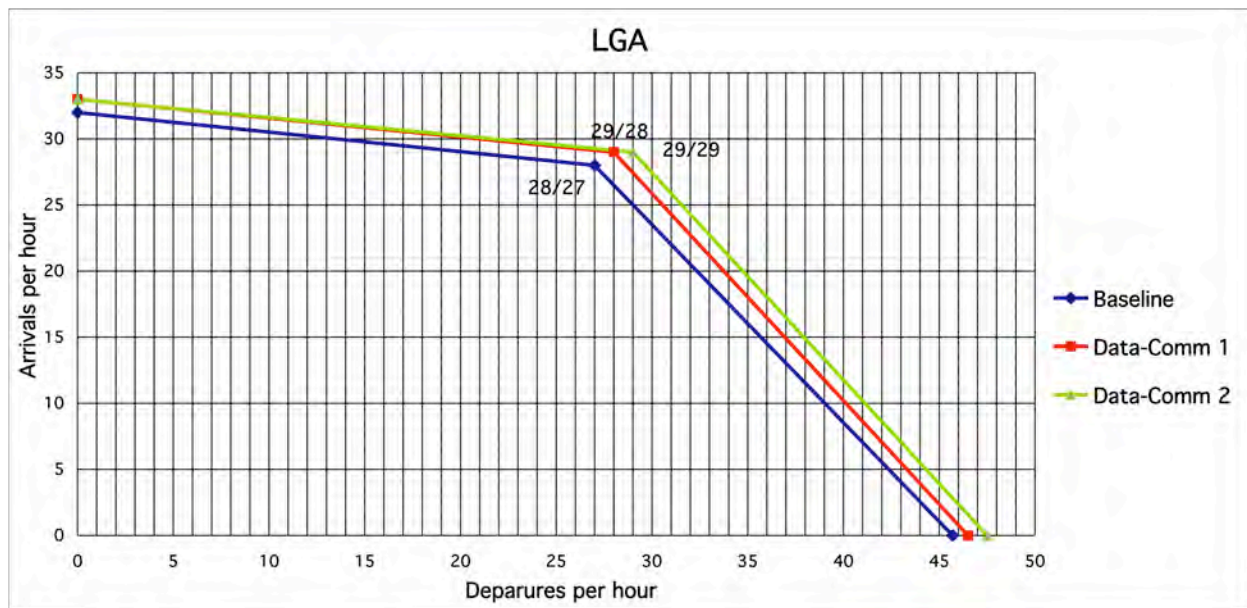


Figure 4.16 LGA Capacity Envelope Diagram.

The practical capacity of La Guardia Airport calculated through the simulation is significantly lower than the one calculated by the FAA through the Airfield Capacity Model. The differences can be attributed to:

- The configuration simulated is not the optimal for the airport,
- The FAA Benchmark Report uses the highest capacity,
- The non optimal behavior of turboprop aircraft in RAMS, and
- The simplifications made in the simulation.

The configuration simulated in RAMS, runway 22 for arrivals and runway 31 for departures, is not the optimal for LGA Airport. The optimal and most used runway configuration is arrivals on runway 22 and departures from runway 13. This configuration offers the least interaction between intersecting operations. We used the configuration 22/31 because this is part of Scenario 3 that in the end was the only one simulated. This restriction applies only for the mixed operations analysis case.

Because RAMS simulates the aircraft behavior based on their performance the injection of the turboprop aircraft slows down the arrival and departure flows observed in the

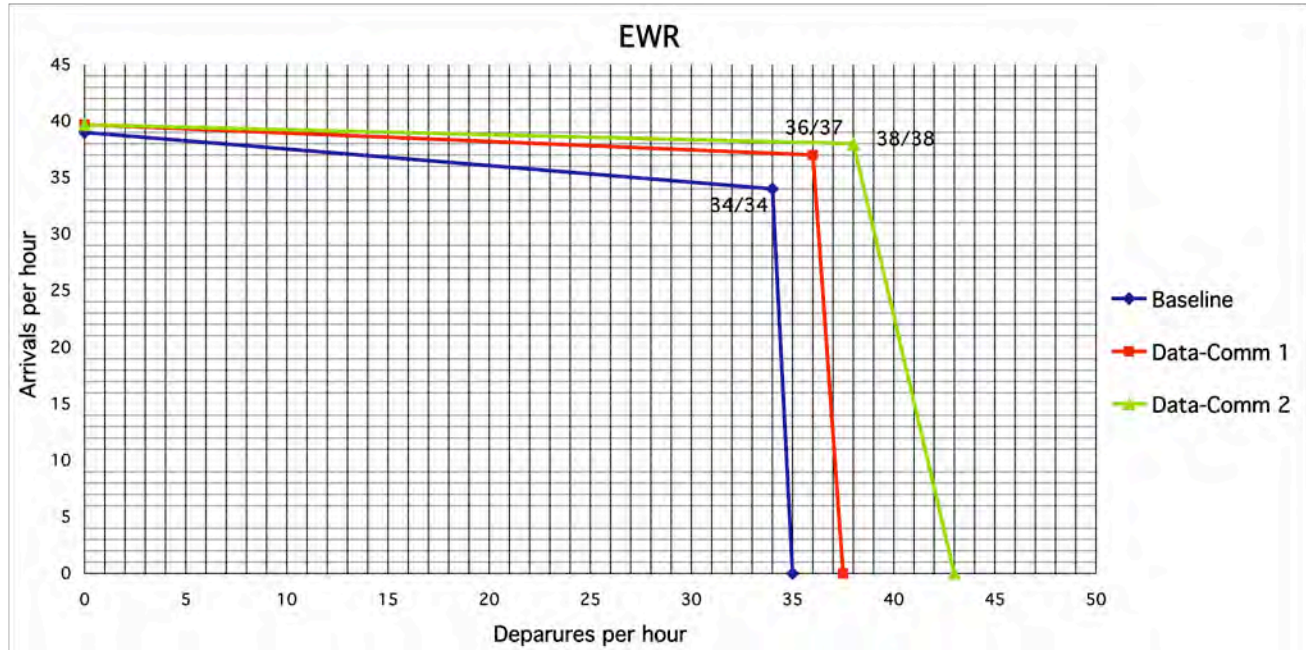
simulation. The discrepancy is mostly present in LGA because the percentage of Turboprop in the aircraft mix is significant, i.e. 8.5 percent.

The simplifications adopted in the simulation are explained in Chapter 3. The usage of super-gates instead of the real gates, that has no influence in ACM here plays a role. The unique approach and departure route, etc. justify the remaining level of capacity missing.

In order to evaluate the order of magnitude of the results we obtained we have also to remember that the practical capacity is usual from 80 to 90 percent of the saturation capacity (Odoni 2003). This is a gross comparison useful only to compare order of magnitude of results obtained using different configurations and with different initial assumptions. For the 'arrivals only' capacity our baseline result is equal to 76.2% of the FAA value, we obtained in fact 32 arrivals per hour instead of 42 of the report. For the "departures only" capacity we calculated 45.7 arrivals per hour instead of 52, inside the range expected. Combining these two results in the mixed operations our simulated value was equal to 62.5 percent of the FAA' result.

With the implementation of the Data-Communication technologies the increment in capacity is small for the departure operations where the benefit of the segment 2 reaches almost 4 percent. In the mixed operations we obtained 5.5 percent of increment.

## 4.5.2. Newark Liberty Airport



**Figure 4.17 EWR Capacity Envelope Diagram.**

The limitation on the capacity results for EWR derives mostly from:

- The configuration simulated is different from the report,
- The taxiway system of EWR,
- The non optimal behavior of turboprop aircraft in RAMS, and
- The simplification adopted in the simulation.

In the EWR case the difference between the configuration evaluated on the FAA' report and the one we simulated is significant because we didn't simulate the operations on the intersecting runway 11/29, that, also if usually limited, give some additional operations per hour to the facility performances. The operations simulated in RAMS were only on the two parallel runways, departures from runway 22R and arrivals to runway 22L.

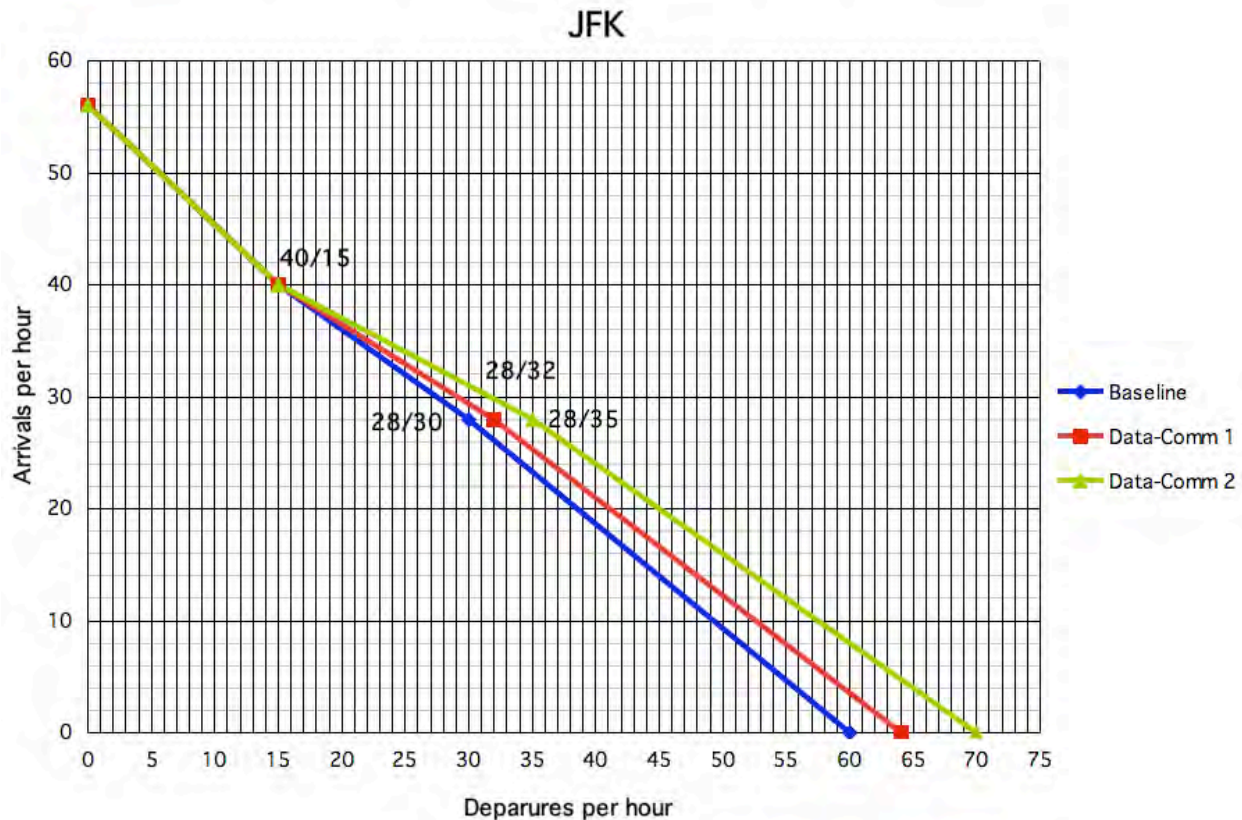
The non-optimal taxiway system of EWR penalized the departures operation more than the arrivals one because of the interaction of the aircraft on the ground not present in the "arrivals only" scenario. The delays calculated from RAMS were in this case mostly on the ground causing the low value of departure capacity.

The explanation of the last two bullets is the same as for LGA but in the Newark's aircraft mix there are only 2.4 percent of Turboprop aircraft.

The results obtained were close again to the one we expected. The “arrivals only” capacity calculated with RAMS was equal to 39 operations per hour whereas in the FAA’ report the capacity was equal to 42. We obtained almost 93 percent of the saturation capacity value. The “departures only” capacity obtained through the simulation was low for the reasons explained above; we obtained 67.3 percent of the “ultimate” capacity obtained through ACM. The mixed operations value was equal to 79 percent of the FAA’ report value, close to the one expected.

With the new communication technologies between controllers and pilot the benefit obtained was significant especially for the departure operations where it grew from 35 to 43 operations per hour with a 22% improvement. This result could be partially explained with the enormous benefit that the very homogeneous departure flow, almost 90% of medium and only 2.6% of Turboprops, had with the new reduced separations minima. With the mixed operations the total benefit was equal to 11.8% mostly driven by the departures improvement.

### 4.5.3. John Fitzgerald Kennedy Airport



**Figure 4.18 JFK Capacity Envelope Diagram.**

In the JFK' case all the facts mentioned for the other airports apply too but the reason why the capacity is strongly below the one calculated with ACM is that we simulated a complete different scenario from the FAA. In fact, analyzing the ASPM data and the airport plates we noticed that JFK never operates with two simultaneous runways for departures. We simulated arrivals on both runway 31L and 31R, and departures from runway 31L only. For the "arrivals only" and "departures only" capacity we simulated in RAMS operations to only one runway and then we doubled the results because the operations on the two parallel runways are independent.

The "arrivals only" practical capacity simulated was equal to 28 operations per hour that doubled give the 56 operation of the Benchmark Report. The low value of arrivals per hour at JFK is easily explained with the operations of heavy aircraft that require larger separations. At JFK 28.8 percent of the aircraft mix are heavy, like Boeing 747, Boeing

777 and Airbus A340. The comparison with the FAA Capacity Benchmark Report results in this case is meaningless for the difference in the scenario simulated.

What is interesting here is to point out the improvement with the Data-Communication technologies that for the “departures only” capacity is equal to 25%. Here the improvement can be explained with the more smooth flow obtainable with the reduced separations that Data-Comm allows. The improvement became smaller for the mixed operations with only 8.6 percent. This benefit reduction is due to the null improvement on the arrival operations.

#### 4.6. Delay Analysis

The results of the delay analysis proved that the three New York airports are already close to their saturation capacity today and that, if the future demand will keep growing, the level of delays will become intolerable very soon. Four levels of demand were simulated, i.e. the baseline year of the study 2006, 2014, 2022, and 2025. Different behaviors were observed in each airport because the level of demand growth predicted by the FAA on the ATO Forecast is different case by case.

##### 4.6.1. La Guardia Airport

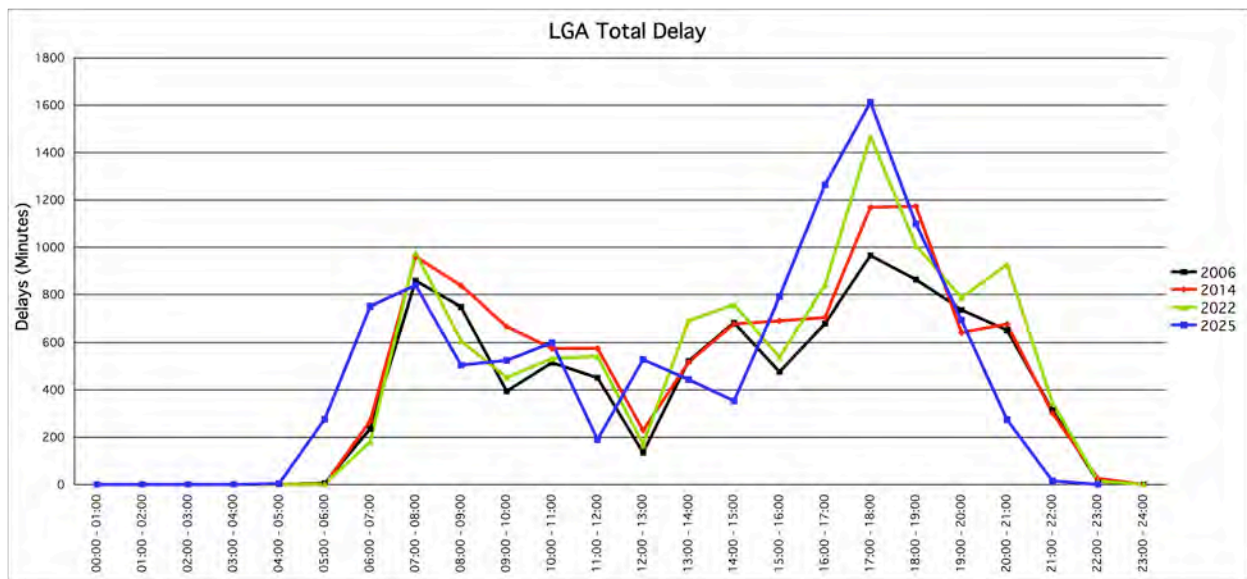


Figure 4.19 LGA Total Delays Growth, Baseline Scenario.

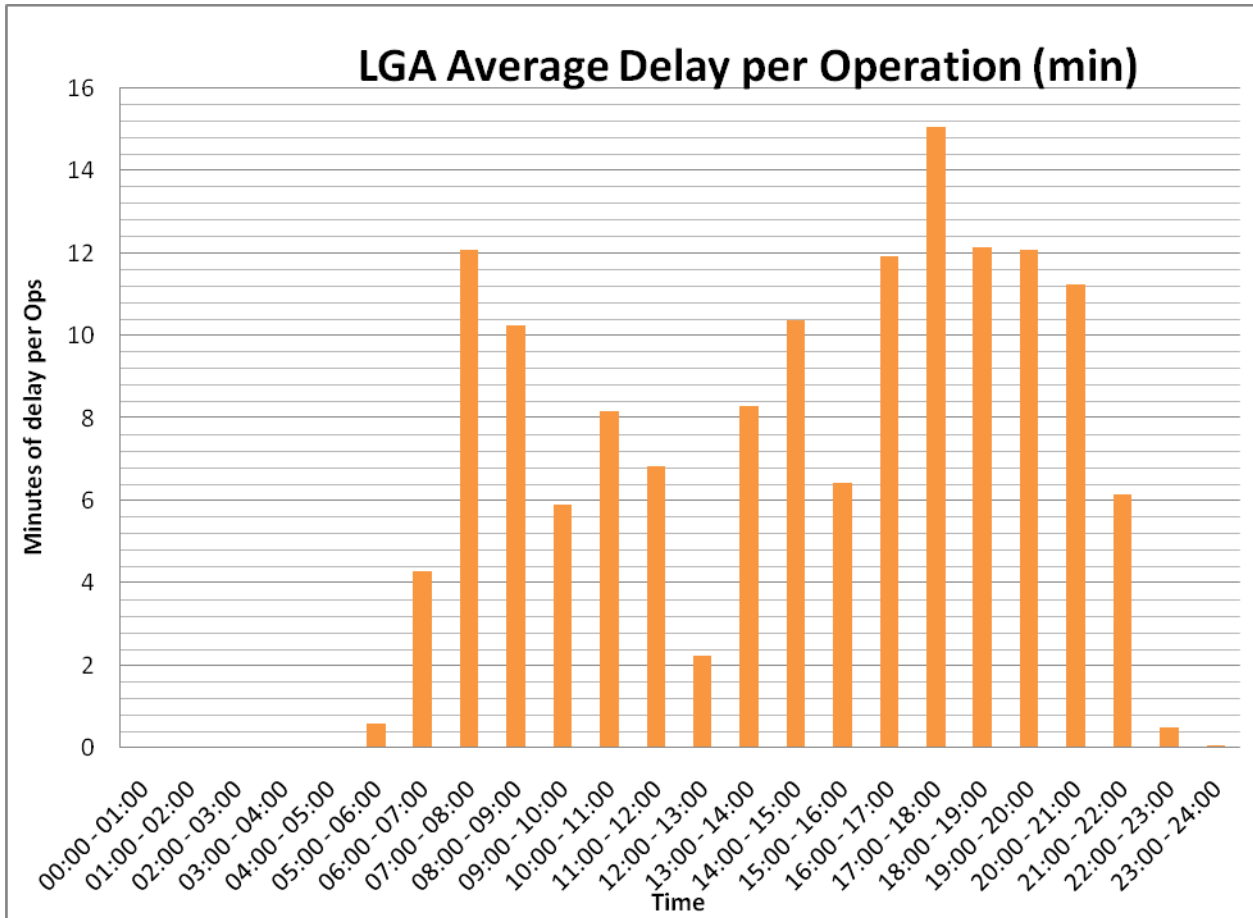
With 2014 and 2022 demand the shapes of the delays remains pretty similar only with 2025 there are slightly different peaks and especially in the first hours of the day. The most important peaks are around 7:00 AM and 5:00PM as it could have been expected. The LGA demand as predicted by the FAA will grow almost insignificantly, in fact the maximum demand we loaded in the scenario is the 2014 one, but the difference is very small as we can see in table 4.4.

**Table 4-4 LGA Average Delays and Total Number of Flights, Baseline Scenario.**

<b>Year</b>	<b>Avg Delay per Ops (min)</b>	<b>% Diff</b>	<b>Number of Flights (24 hrs)</b>	<b>% Diff</b>
2006	6.03		1211	
2014	6.64	10.11	1238	2
2022	6.86	13.76	1236	2
2025	6.7	11.11	1220	1

The slight difference in the average delay results can be explained with the different distribution over the 24 hours of the demand, in fact in terms of operations per hour it's almost irrelevant. The average delay for all the scenarios simulated is around six minutes per operation that, with the four minutes rule, set La Guardia as already over the practical capacity. It is interesting to notice how a 2 percent increase in demand causes more than 10 percent of increase in delays; this happens because we are already very close to the knee of the delay curve were small variations in the demand create large increments in the delay values.

The distribution of the average delay per operation during the 24 hours is similar to the total delay one. The maximum average delay experienced at LGA is around 15 minutes per operation. For the FAA any record under this threshold is not even considered as a delay.



**Figure 4.20 LGA Average Delay per Operation Baseline Scenario 2006 Demand.**

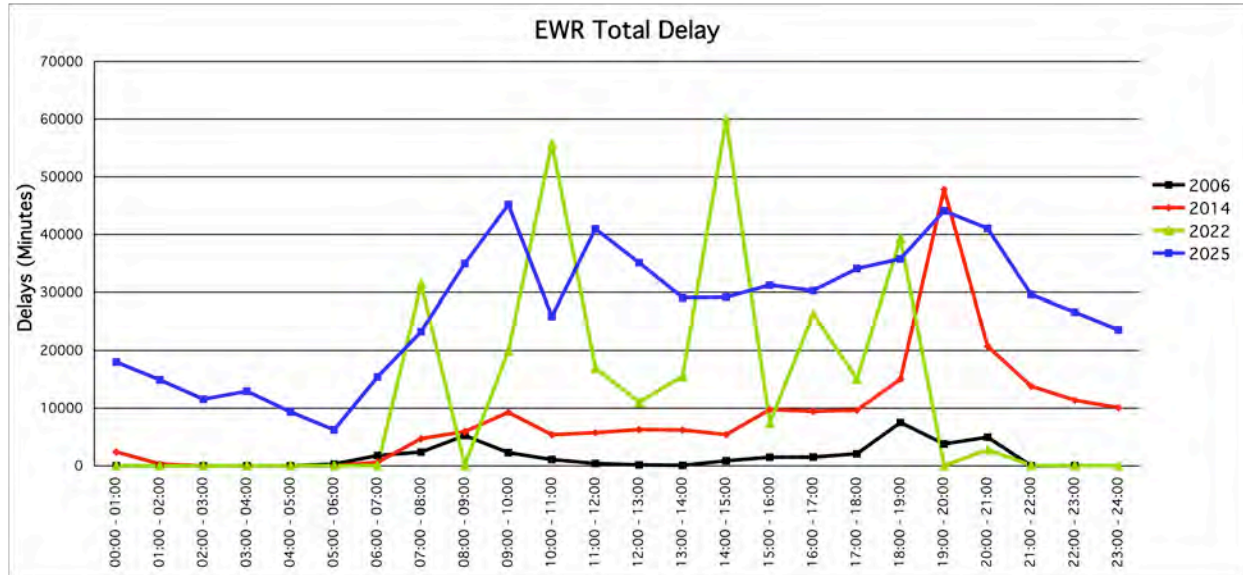
Comparing the baseline scenario with the Data-Communication one interesting benefits were recorded. From 6.03 minutes per operation of the baseline the average with Data-Communication segment 2 was 4.25, with an improvement of 29.5 percent. The differences are pretty similar with the following year demands because, as we said before, for LGA there is not a significant growth predicted.

**Table 4-5 LGA Benefit Summary.**

	<b>Baseline</b>	<b>Data-Comm 1</b>	<b>Data-Comm2</b>
Year	Avg Delay per Ops (min)	Avg Delay per Ops (min)	Avg Delay per Ops (min)
2006	6.03	5.47	4.25
2014	6.64	6.27	6.27
2022	6.86	6.45	5.78
2025	6.7	5.95	5.09



## 4.6.2. Newark Liberty Airport



**Figure 4.21 EWR Total Delays Growth, Baseline Scenario.**

The results in the EWR simulation are less continuous and only for 2006 and 2014 they follow the expected trends. The fact that we simulated a scenario with only one runway for departure (22R) and one for arrivals (22L) and we didn't add the intersecting runway operations caused a decrement in the overall performance. Newark, in fact is the only airport that in the calibration step gave us some problems. With the 2022 and 2025 demand the delays are so high and the operations so crowded that the future demand scenarios results are not significant.



**Figure 4.22 Queue Accumulation in EWR 2025 Scenario.**

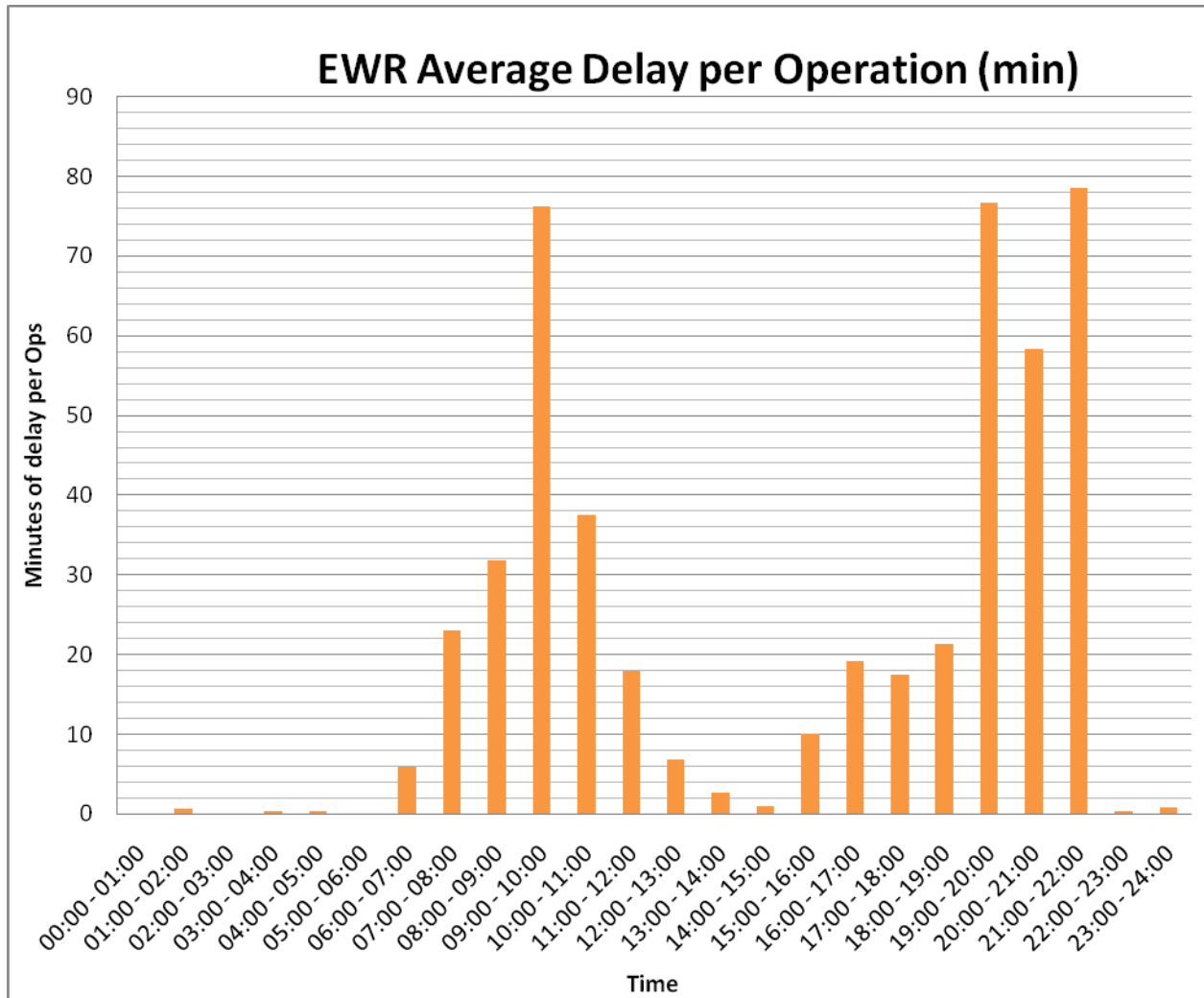
For the lack of capacity that we didn't simulate, already the baseline scenario gave us high levels of delays; these results were also driven by the not ideal runway and taxiway configuration of the airport. The fact that both parallel runways are on the same side of the gates make necessary the crossing of runway 22R by the arrival traffic on 22L.

**Table 4-6 EWR Average Delays and Total Number of Flights, Baseline Scenario.**

Year	Avg Delay per Ops (min)	% Diff	Number of Flights (24 hrs)	% Diff
2006	20		1314	
2014	116	480	1451	10
2022	228	1040	1549	18
2025	327	1535	1584	21

The baseline delay per operation is 20 minutes and with the increased capacity the situation gets worse very quickly. With the 2025 demand the delay would grow to 327 per minutes, clearly not feasible or sustainable. Also if in our simulation we slightly

underestimated the capacity of the airport there is no way, without any significant infrastructural improvement, in which EWR could process a 21 percent increase of demand without experiencing heavy delays.



**Figure 4.23 EWR Average Delay per Operation Baseline Scenario 2006 Demand.**

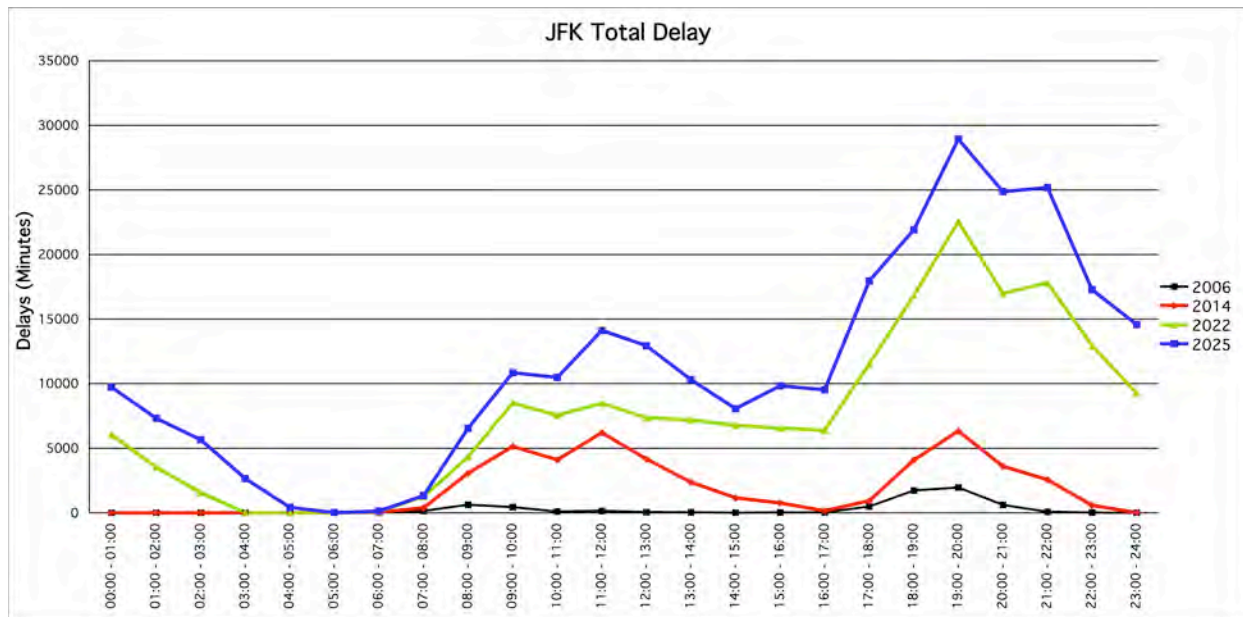
The average delay per operation in the peak hours goes up to more than one hour per flight. This Level Of Service (LOS) would not be acceptable in any western country airport.

**Table 4-7 EWR Benefit Summary.**

	<b>Baseline</b>	<b>Data-Comm 1</b>	<b>Data-Comm 2</b>
Year	Avg Delay per Ops (min)	Avg Delay per Ops (min)	Avg Delay per Ops (min)
2006	20	14	12.3
2014	124	119	116
2022	228	218	197
2025	327	281	258

With the implementation of the new technologies there would be a benefit that can be considered interesting only for the baseline year because for the other years the LOS is so deteriorated that it doesn't make any sense to comment it. The benefit in the baseline demand year would be around 38.5 percent, from 20 minutes per operation to 12.3.

### **4.6.3. John Fitzgerald Kennedy Airport**



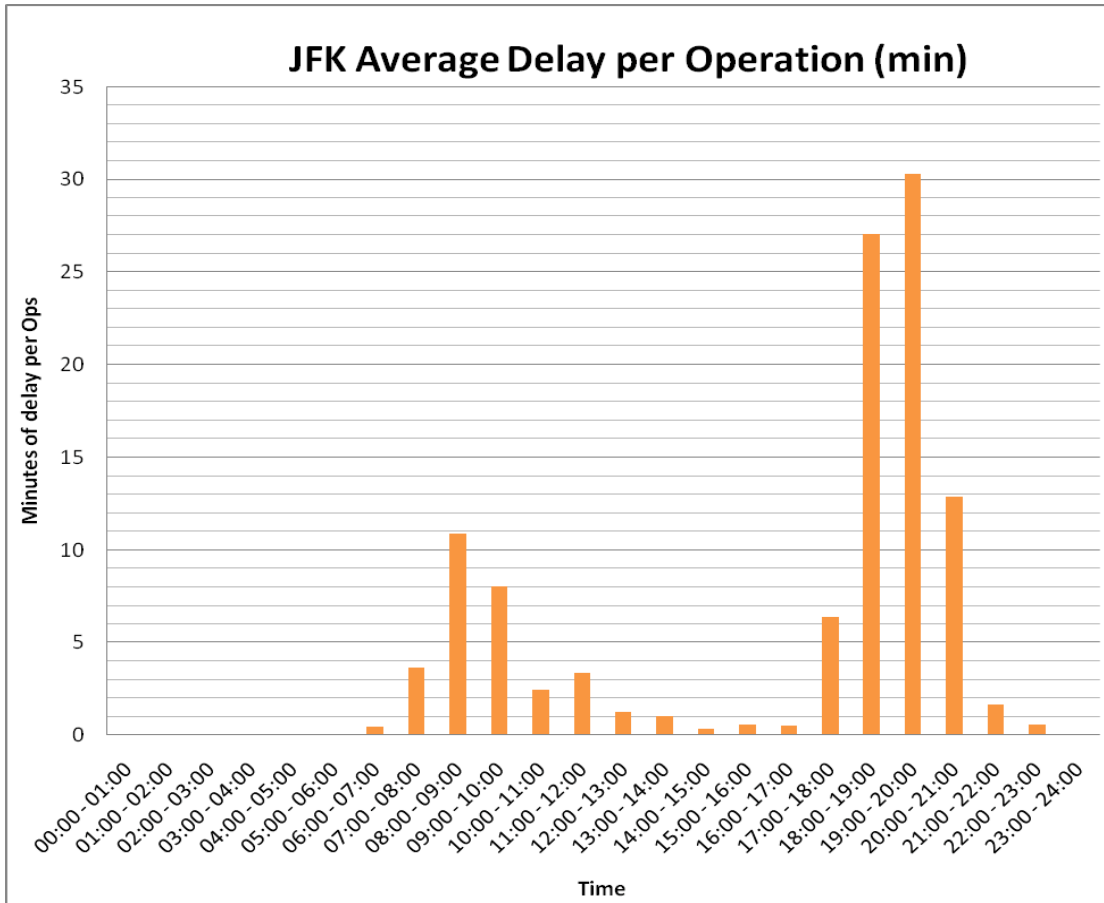
**Figure 4.24 JFK Total Delays Growth, Baseline Scenario.**

The delay trends in JFK present the typical double peaks in the morning around 10:00 AM and in the evening around 7:00 PM. The growth of the delays from 2006 demand to 2025 is the most significant compared to the other airports, but it's explained by the fact that the demand predicted by the FAA for the future will increase tremendously.

**Table 4-8 JFK Average Delays and Total Number of Flights, Baseline Scenario.**

	<b>Avg Delay per Ops (min)</b>	<b>% Diff</b>	<b>Number of Flights (24 hrs)</b>	<b>% Diff</b>
2006	4.64		1009	
2014	26.87	479	1351	34
2022	88.92	1810	1550	54
2025	217.89	4590	1588	57

The FAA claims that in 2025 JFK would accommodate 1588 flights per day, an increase of 57 percent from 2006. This airport doesn't have the performance and the capacity to accommodate this demand without intolerable level of delays. Our simulation with two runways for arrivals (31L and 31R), and one for departures (31L) showed that in 2025 the average delay per flight would be more than 200 minutes. Also in 2014 with an increment of 34 percent the level of delays would be very high, i.e. 26.87 per flight. The fact that the baseline delay level is more than 4 minutes shows that this infrastructure is actually operating at capacity.



**Figure 4.25 JFK Average Delay per Operation Baseline Scenario 2006 Demand.**

The plot in figure 4.25 shows that also if the average is around 5 minutes during the peak hour the delays per operation go up to 30 minute, especially during the second high peak of the day when all the delays of the day sum up.

**Table 4-9 JFK Benefit Summary.**

	<b>Baseline</b>	<b>Data-Comm 1</b>	<b>Data-Comm 2</b>
Year	Avg Delay per Ops (min)	Avg Delay per Ops (min)	Avg Delay per Ops (min)
2006	4.64	4.73	2.77
2014	26.87	25.05	16.21
2022	88.92	84.01	59.15
2025	217.89	147.27	94.47

With the last segment of implementation of Data-Communication the benefit achievable would be significant. The reduction from 4.64 to 2.77 minutes per operation represents

more than 40 percent of improvement. Also with Data-Communication completely in service 2022 and 2025 demands could not be served with a decent level of delays.

#### **4.7. Workload Analysis**

The last analysis that we did on the New York City Metroplex scenario was on the impact and benefits of the implementation of the Data-Communication technologies, that on the FAA plan would be introduced in two steps, on the controllers' workload. A step-by-step implementation approach of these new concepts is required in a sensitive field like controller-pilot communications. As described in chapter 3 we simulated this new concept in RAMS following the double objective of the project, i.e. reducing delays and controllers' workload.

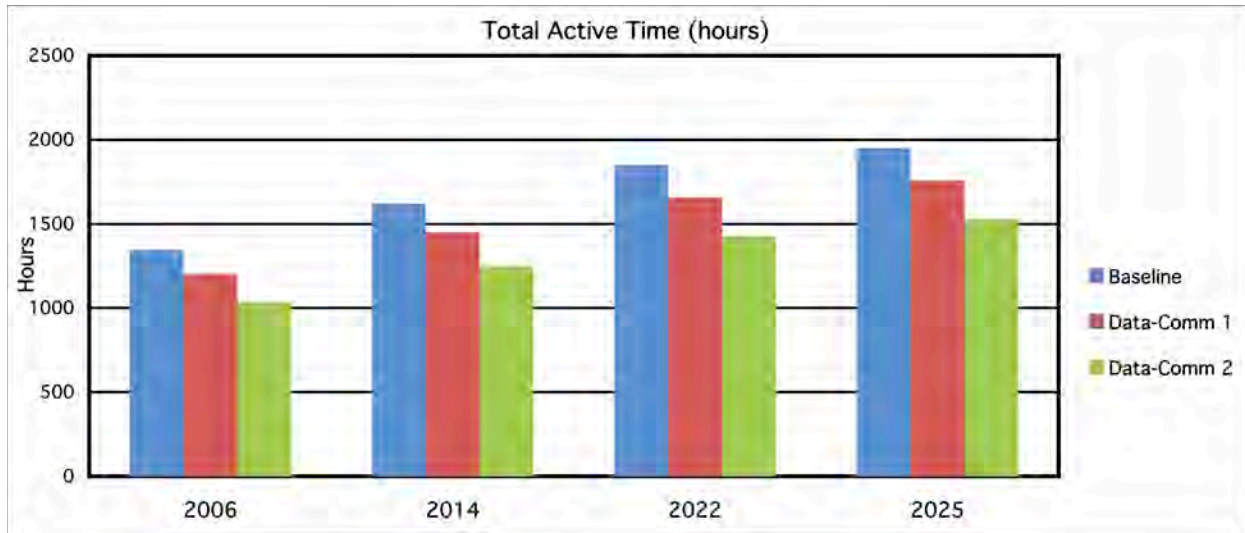
The reduced workload was simulated in RAMS reducing the task weights gradually from the baseline scenario to the segment 2 future concepts. The tasks reduced were extrapolated from the FAA studies (FAA 1995; FAA 1996). The complete list of RAMS controllers tasks is represented in Table 4-9. The tasks reduced in the future scenarios are underlined in grey.

**Table 4-10 Reduced Controllers Task in RAMS.**

Task Number	Category	Task Description	Controller	Weights in Seconds		
				Baseline	Data-Comm 1	Data-Comm 2
Task1	RT Communication	Transmit New FL	Tactical	10	5	0
Task10	Conflict Search	Establish Initial FL Clearance FI on the Ground	Planning	15	15	15
Task11	Conflict Search	Establish Initial FL Clearance FI Un Stable	Tactical	10	10	10
Task12	Conflict Search	Establish Initial FL Clearance FI on the Ground	Tactical	10	10	10
Task13	Conflict Search	Establish Initial FL Clearance FI on the Ground	Tactical	10	10	10
Task14	Conflict Search	Establish Initial FL Clearance FI Stable	Planning	8	8	8
Task15	Conflict Search	Establish Initial FL Clearance FI Stable	Tactical	5	5	5
Task16	Conflict Search	Establish Sector Exit Clearance FI Stable	Planning	5	5	5
Task17	Conflict Search	Establish Sector Exit Clearance FI Un Stable	Planning	5	5	5
Task18	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Planning	5	5	5
Task19	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Planning	5	5	5
Task2	RT Communication	Receive Flight Level Reached Report	Tactical	5	3	0
Task20	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Planning	5	5	5
Task21	Conflict Search	Establish Sector Exit Clearance FI Stable	Tactical	5	5	5
Task22	Conflict Search	Establish Sector Exit Clearance FI Un Stable	Tactical	5	5	5
Task23	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Tactical	5	5	5
Task24	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Tactical	5	5	5
Task25	Conflict Search	Establish Sector Exit Clearance FI on the Ground	Tactical	5	5	5
Task26	Coordination	Input ACT Message To Computer FI Stable	Planning	15	15	15
Task27	Coordination	Input ACT Message To Computer FI Stable	Planning	15	15	15
Task28	Coordination	Input ACT Message To Computer FI Un Stable	Planning	15	15	15
Task29	Coordination	Input ACT Message To Computer FI Un Stable	Planning	15	15	15
Task3	RT Communication	Transmit New FL	Planning	1	0	0
Task30	Coordination	Input ACT Message To Computer FI on the Ground	Planning	15	15	15
Task31	Coordination	Input ACT Message To Computer FI on the Ground	Planning	15	15	15
Task32	Coordination	Receive Airborne Time From Local Airport FI on the Ground	Planning	6	3	0
Task33	Coordination	Receive Airborne Time From Local Airport FI on the Ground	Tactical	3	1	0
Task34	Coordination	Receive Clearance Request From Local Airport FI on the Ground	Planning	15	10	5
Task35	Flight Data Management	Receive Flight Progress Strip FI Stable	Planning	10	10	10
Task36	Flight Data Management	Receive Flight Progress Strip FI Un Stable	Planning	10	10	10
Task37	Flight Data Management	Receive Flight Progress Strip FI on the Ground	Planning	10	10	10
Task38	Flight Data Management	Receive Flight Progress Strip FI Stable	Tactical	5	5	5
Task39	Flight Data Management	Receive Flight Progress Strip FI Un Stable	Tactical	5	5	5
Task4	Conflict Search	Conflict Situation Identified	Planning	15	15	15
Task40	Flight Data Management	Receive Flight Progress Strip FI on the Ground	Tactical	5	3	0
Task41	Coordination	Received Take Off Request FI on the Ground	Planning	10	5	0
Task42	Coordination	Received Take Off Request FI on the Ground	Tactical	5	3	0
Task43	Coordination	Received Time Level Estimate FI Stable	Planning	15	15	15
Task44	Coordination	Received Time Level Estimate FI Un Stable	Planning	25	25	25
Task45	RT Communication	Received 1st Call Airspace 1st Simulated Center	Tactical	10	5	0
Task46	RT Communication	Received 1st Call Airspace Inter Center	Tactical	10	5	0
Task47	RT Communication	Received 1st Call Airspace Inter Sector	Tactical	8	4	0
Task48	RT Communication	Received Handoff	Tactical	5	3	0
Task49	RT Communication	Transmitted Change Of Frequency	Tactical	5	3	0
Task5	Conflict Search	Planning Conflict Search	Planning	5	5	5
Task50	RT Communication	Transmitted Handoff	Tactical	8	4	0
Task51	Flight Data Management	Remove Flight Progress Strip	Planning	3	3	3
Task52	Flight Data Management	Remove Flight Progress Strip	Tactical	2	2	2
Task6	Conflict Search	Initial FL Clearance For Local Airport Departure	Planning	5	5	5
Task7	Conflict Search	Initial FL Clearance For Local Airport Departure	Tactical	5	5	5
Task8	Conflict Search	Establish Initial FL Clearance FI Un Stable	Planning	15	15	15
Task9	Conflict Search	Establish Initial FL Clearance FI on the Ground	Planning	15	15	15

The sectors simulated were thirteen in the core area plus four around the core area. Each sector in RAMS has two controllers, a tactical and a planning controller; so a total of 34 controllers were simulated.





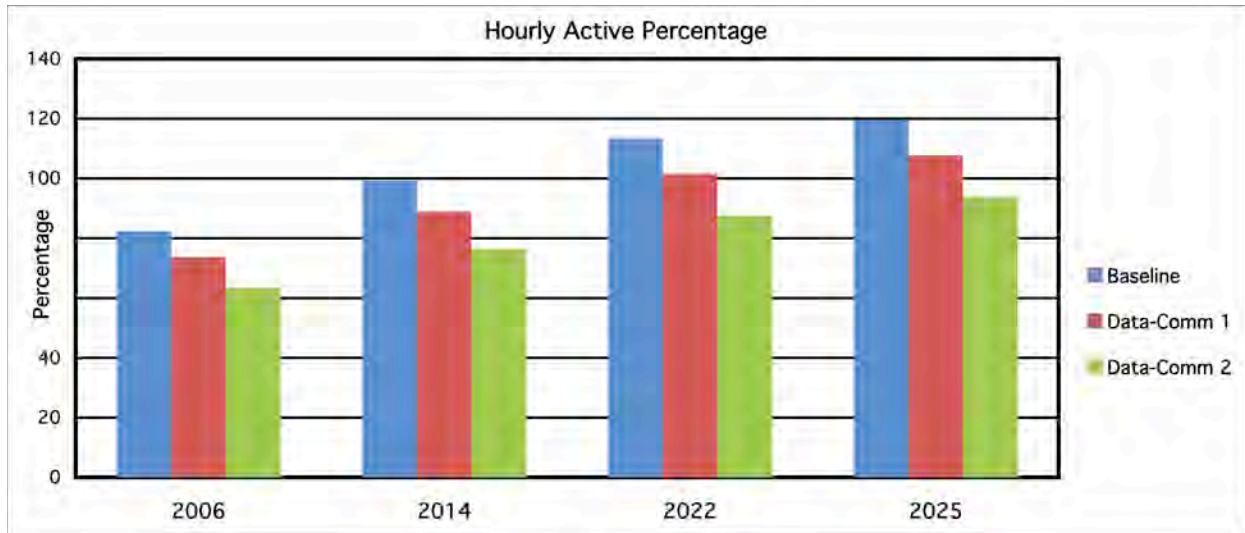
**Figure 4.26 Total Occupation Time.**

The results of the simulation with Data-Communication gave a general reduction in the total hours in which the controllers were busy handling traffic (Fig. 4.19). The benefit was evaluated in the baseline year, i.e. 2006, in 2014, 2022, and 2025. The reduction was significant and it's similar for all the four levels of demand simulated. The benefit is always calculated on the baseline scenario with the same level of demand and with the actual level of communication technologies.

**Table 4-11 Benefits with Data-Communication Technologies on the Total Occupation Time.**

	2006	2014	2022	2025
<b>Scenario</b>	% Benefit	% Benefit	% Benefit	% Benefit
Data-Comm 1	10.54	10.47	10.44	9.86
Data-Comm 2	23.03	22.94	22.86	21.56

The reduction for all the demand levels loaded into the scenario were around 10 percent with Data-Communication segment 1 and almost 23 percent with the complete implementation of the new communication standard. Only in 2025 scenario the benefit is slightly lower, around 21.5 percent.



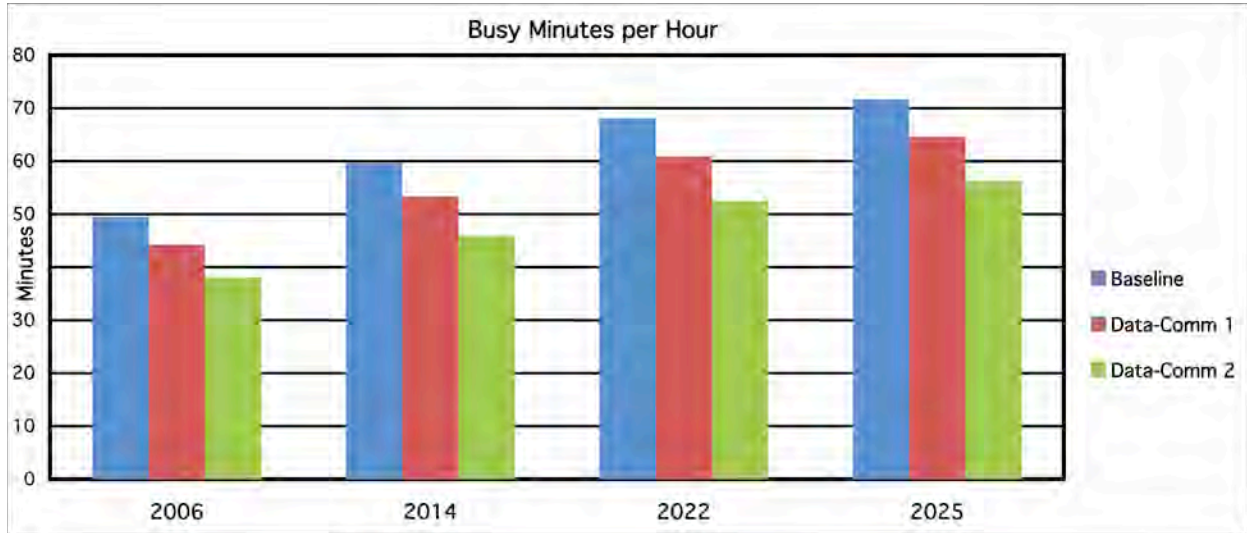
**Figure 4.27 Percentage of Occupation Time.**

The percentage of the time per hour in which the controllers were busy with the 2006 demand was close to 82.5 percent. With the full implementation of Data-Communication this would have lowered to 63.48 percent. Without any new communication tool already in 2014 the percentage would reach almost 100 percent with an unbearable workload or with the need of adding controllers to the airspace. The situation would become completely unsustainable with 2022 and 2025 demand. With the new communication paradigm the increment of workload would be around five percent in 2022, probably an increment in the manpower would be necessary but the number of controllers to add would also be less.

**Table 4-12 Occupation Time by Scenario.**

	<b>2006</b>	<b>2014</b>	<b>2022</b>	<b>2025</b>
	Occupation %	Occupation %	Occupation %	Occupation %
Baseline	82.48	99.34	113.45	119.52
DC1	73.79	88.94	101.60	107.74
DC2	63.48	76.55	87.51	93.75

The same percentages had been translated in minutes in figure 4.21.



**Figure 4.28 Occupation Time in Minutes per Hour.**

From the 49.9 minutes with the actual communication procedures the baseline value goes down to 38.9. In 2014 the busy time would reach the 59.6 minutes per hour, a level not acceptable at all especially for continued period of time. With the full implementation of Data-Communication technologies the actual level of workload would be experienced with 2022 demand.

**Table 4-13 Busy Minutes per Hour by Scenario.**

	2006	2014	2022	2025
	Busy Minutes/hr	Busy Minutes/hr	Busy Minutes/hr	Busy Minutes/hr
Baseline	49.9	59.61	68.07	71.71
DC1	44.27	53.37	60.96	64.64
DC2	38.09	45.93	52.51	56.25

The baseline value of workload is very close to the one calculated in the FAA man-in-the-loop simulation of the Newark terminal area (FAA 1996), in fact in that study it was claimed that during peak hour the controllers were busy for more than 54 minutes, we obtained almost 50 minutes. Our value is lower because it's an average over 48 hours, but the hourly demand is higher because it's the 2006 one. The workload calculated with RAMS includes also tasks other than voice communication, e.g. flight data

management, conflict search, and coordination; this explains why our value would be higher during peak hours.

## 5. RESULTS VALIDATION

The validation process for simulation models is a complex and time-consuming process. To calibrate our results we considered two parameters: the total ground delays experienced by departure operations at each airport and the separation time for arrival and departure operations. The FAA Aviation System Performance Metrics (ASPM) database was used to obtain delays used in the calibration. PDARS data was used to verify inter-arrival and inter-departure separations. For the purpose of delay calibration, it is important to compare the same types of delays. In the ASPM database there are many different delays recorded but one in particular that resembles the delays recorded in RAMS (“taxi out” delays). RAMS does not record airborne delays for the departure operations. All the other delays related to the flight plan in ASPM cannot be used because they represent some variability that cannot be inputted into RAMS, such as delays caused by passengers loading process or delays accrued from a previous leg flown by the same aircraft. The demand simulated in RAMS is relative to February 23 2006, so we used the same day in ASPM to compare the results.

## 5.1. La Guardia Airport

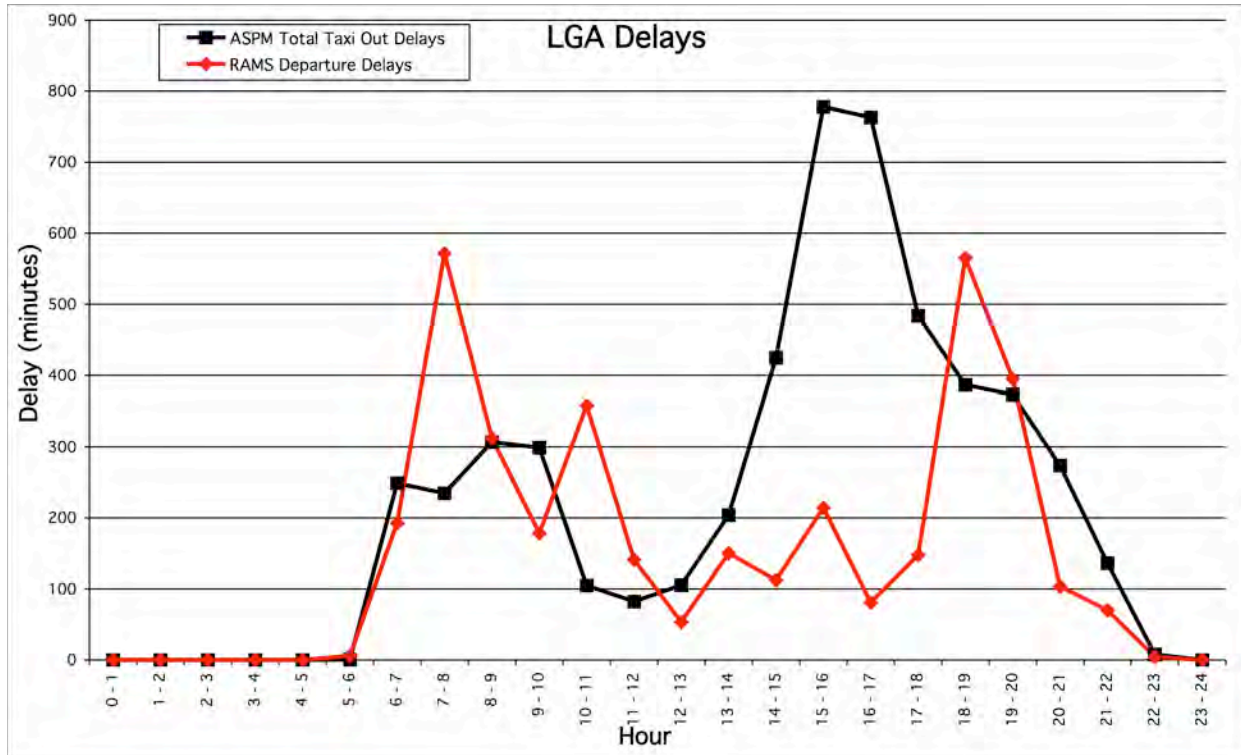
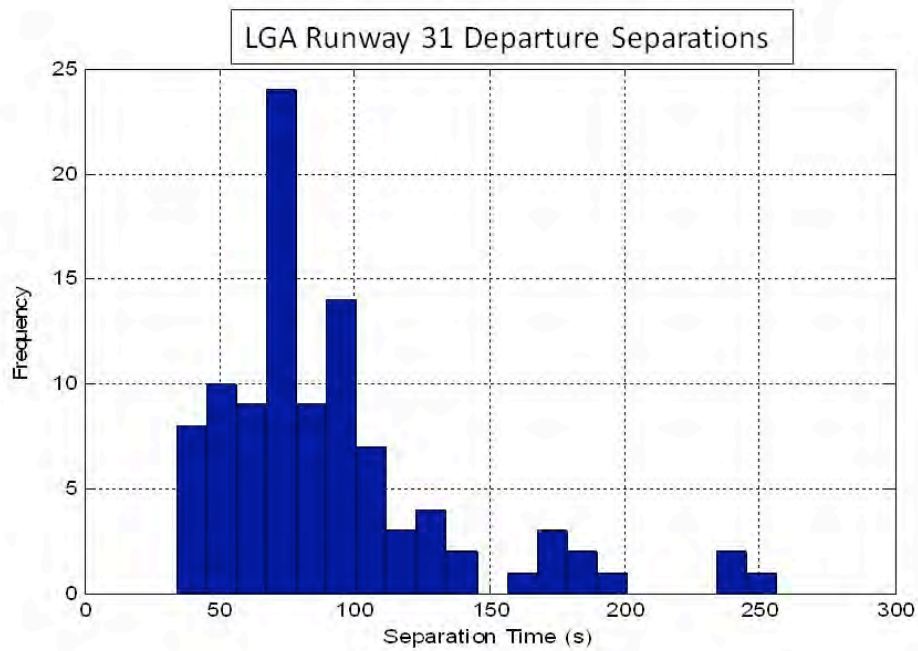


Figure 5.1 LGA Delays Comparison with ASPM.

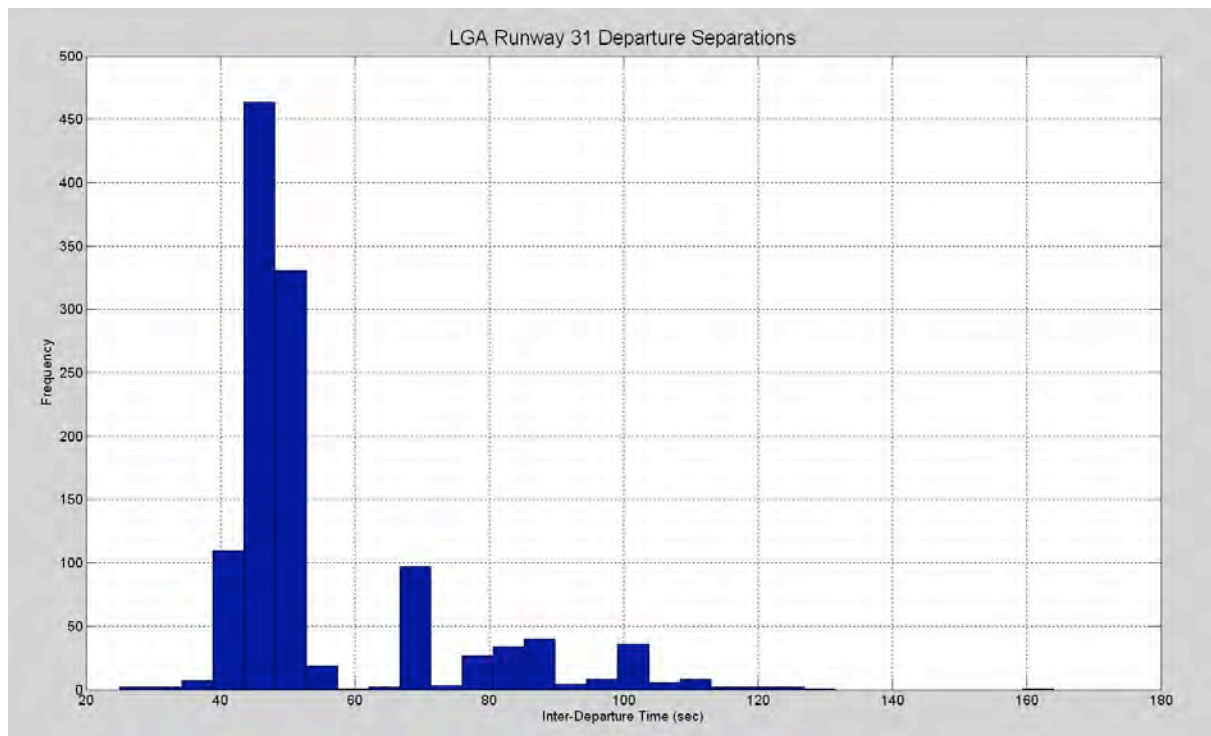
La Guardia Airport shows a delay trend that is very similar to the ASPM data one both in the order of magnitude and in the peaks and valleys. The RAMS output is smaller in the total value than that observed. The total values for the all day, i.e. the area under the two curves, ASPM value is equal to 5,212 minutes whereas RAMS value is 3792 minutes. The difference is 27 percent on the total delays value. The reason why this difference exist can be:

- The unique configuration simulated in RAMS, and
- The absence of any possible “human error” in the ground operations.



**Figure 5.2 LGA Departure Separations from PDARS Data Analysis.**

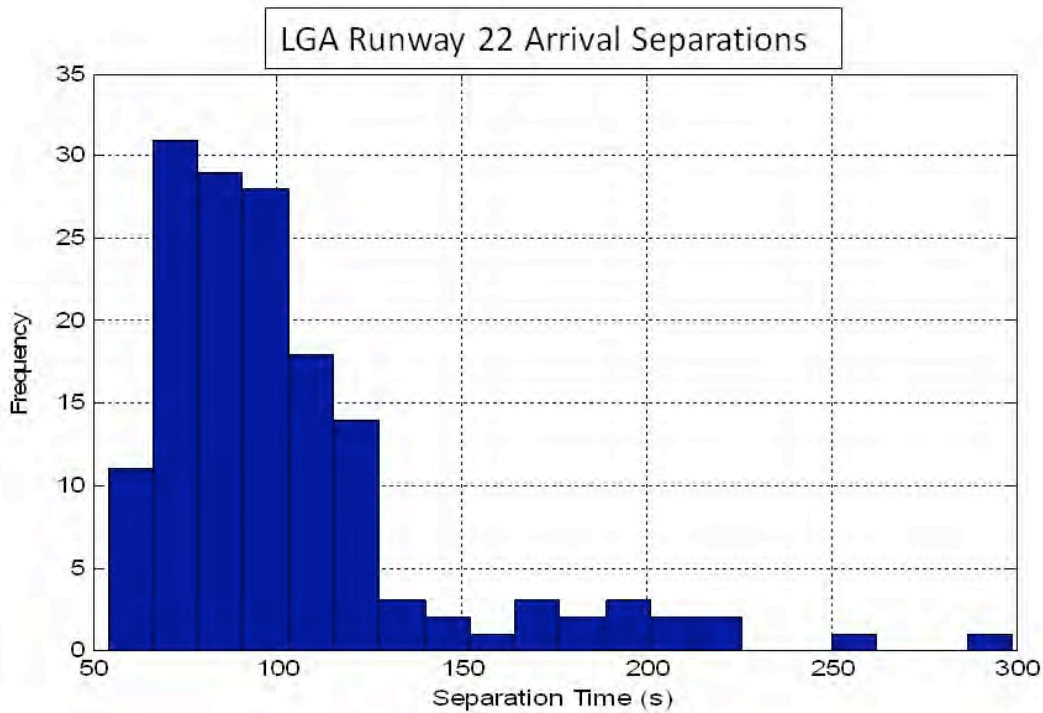
In the PDARS data of February 23 2006 there are only 108 departures from runway 31 that was not heavily used that day. This low utilization is probably due to the direction of the wind blowing that day on LGA. The average time between successive arrivals is 91.35 seconds.



**Figure 5.3 LGA Departure Separations from RAMS.**

The average departure-departure headway at LGA is 67.5 seconds. Figure 5.3 illustrates the probability density function for inter-departure times. There are some cases when the simulation program is not able to de-conflict the flights and consequently the separation minima are violated bringing down the average.





**Figure 5.4 LGA Arrival Separations from PDARS Data Analysis.**

There are only 160 records of arrivals to runway 22 in the PDARS data examined and the average time between arrivals is 102.33 seconds. There are numerous records scattered over the day that lowers the average as can be seen in Fig. 5.4. The total records in RAMS are 1192 that, filtered from the records over 300 seconds and under 20 seconds, gave an average of 92.07 seconds.

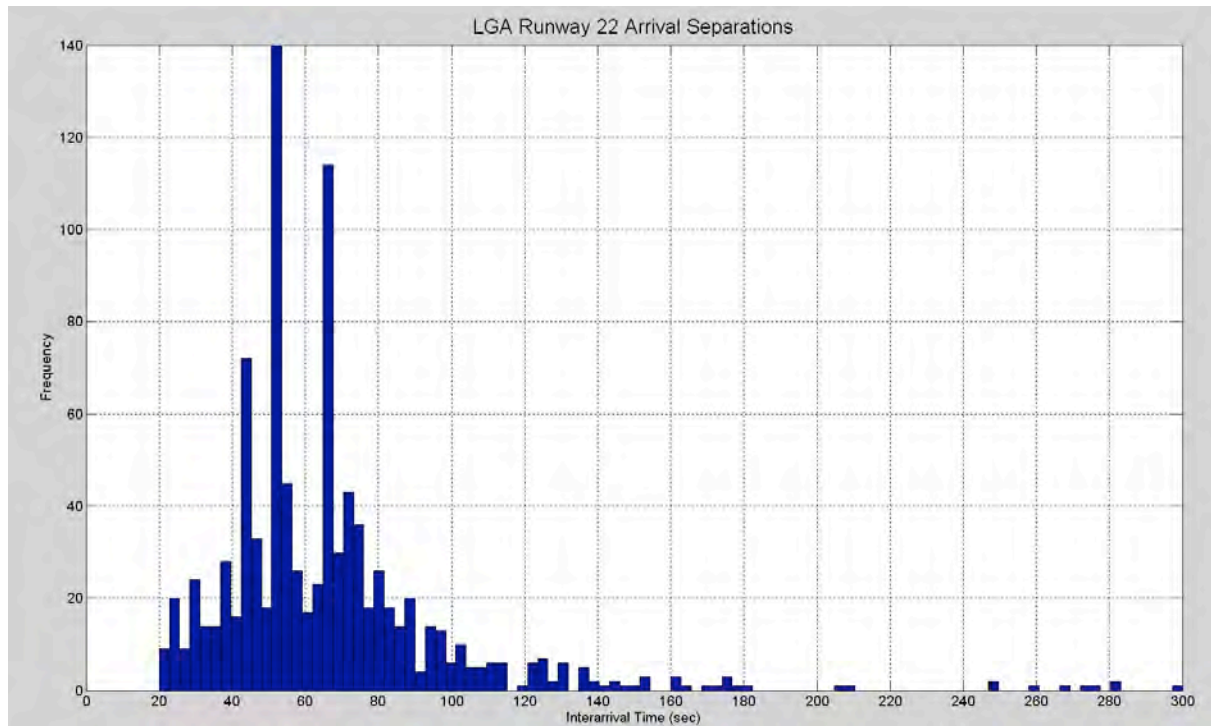


Figure 5.5 LGA Arrival Separations from RAMS.

## 5.2. Newark Liberty Airport

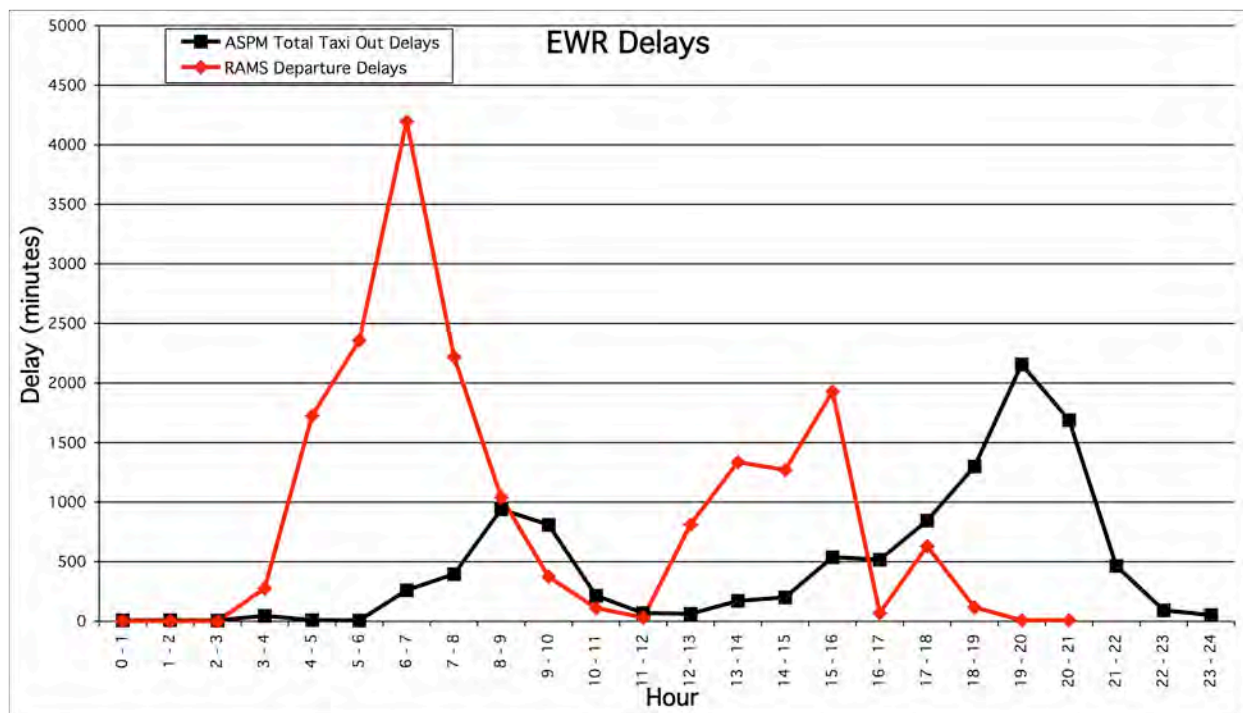
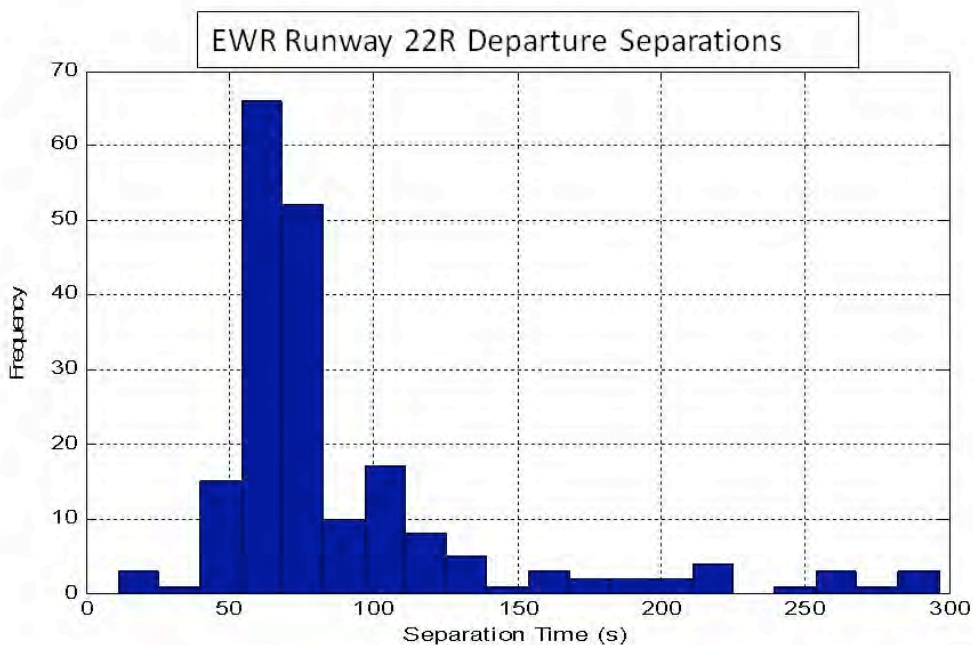


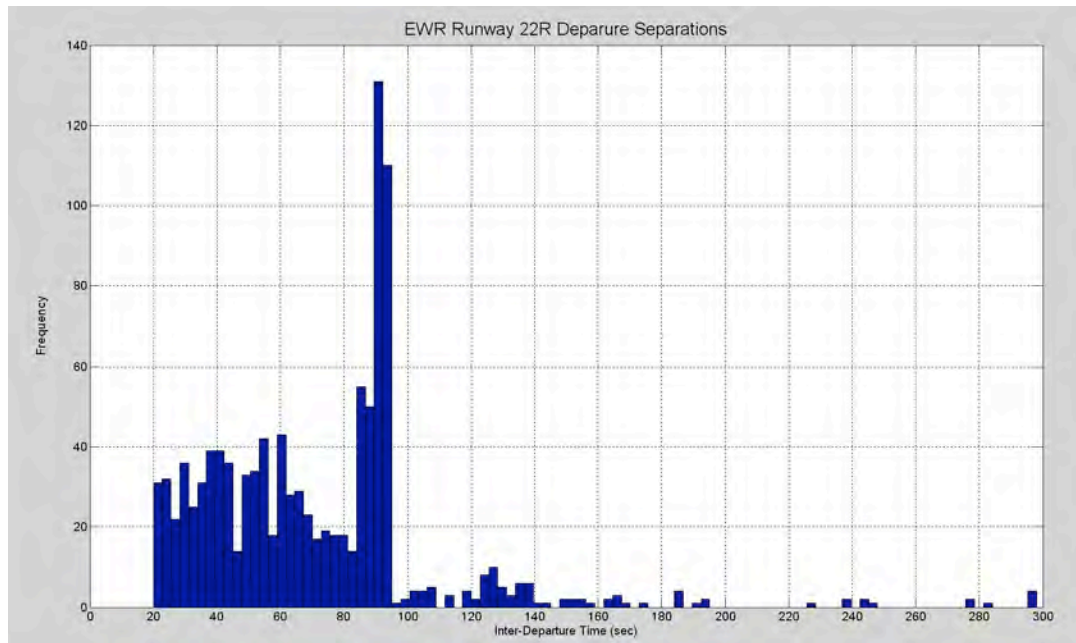
Figure 5.6 EWR Delays Comparison with ASPM.

As mentioned before Newark was the airport that had the largest discrepancies. The reason is that we used a single runway for arrivals. Often in Newark the intersecting runway 11 is used to land small regional jets. Not simulating that we lost some capacity capabilities of the airport. In Fig. 5.6 it can be seen that the delays obtained from the simulation model are higher as a total that the ASPM one and that the highest peak in the simulation is around 7:00 AM instead in reality, that day the highest delays were experienced around 7:00 PM. The order of magnitude is still comparable but the results are not completely satisfying.



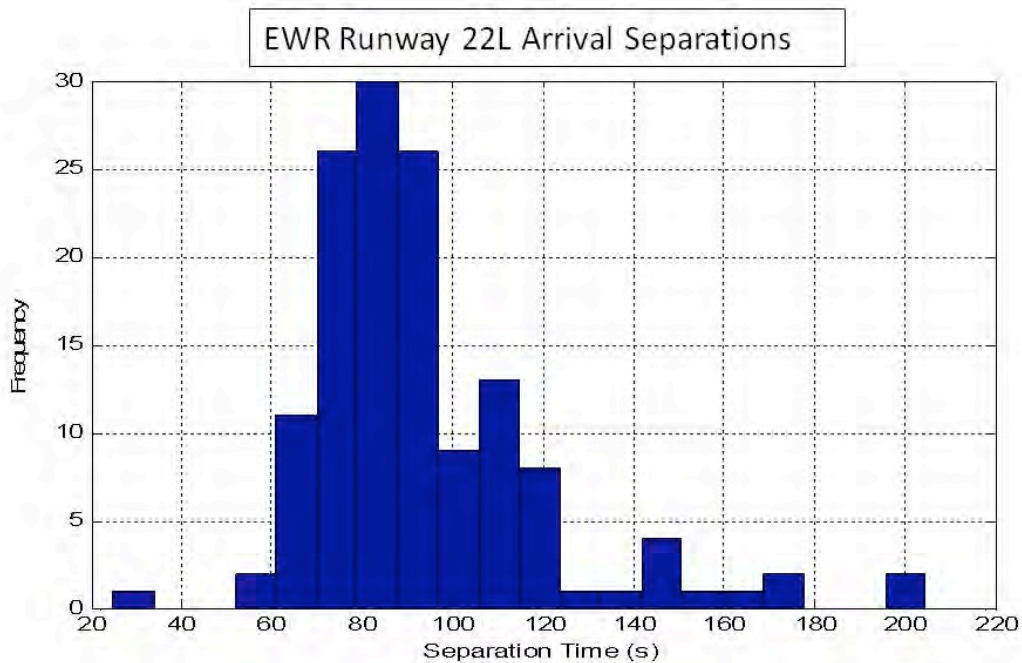
**Figure 5.7 EWR Departure Separations from PDARS Data Analysis.**

From the PDARS data there are 212 observations. The average inter-departure time was 89.19 seconds. The total number of simulated flights was 1097 with an average of 85.35 seconds between successive departures.



**Figure 5.8 EWR Departure Separations from RAMS Output.**

For arrival operations, the day analyzed from PDARS data had only 138 landings to runway 22L. The calculated average was 93.85 seconds.



**Figure 5.9 EWR Arrival Separations from PDARS Data Analysis.**

The mean on the 838 arrival operations simulated in RAMS was 85.54 seconds, 8.85 percent different from the PDARS data.

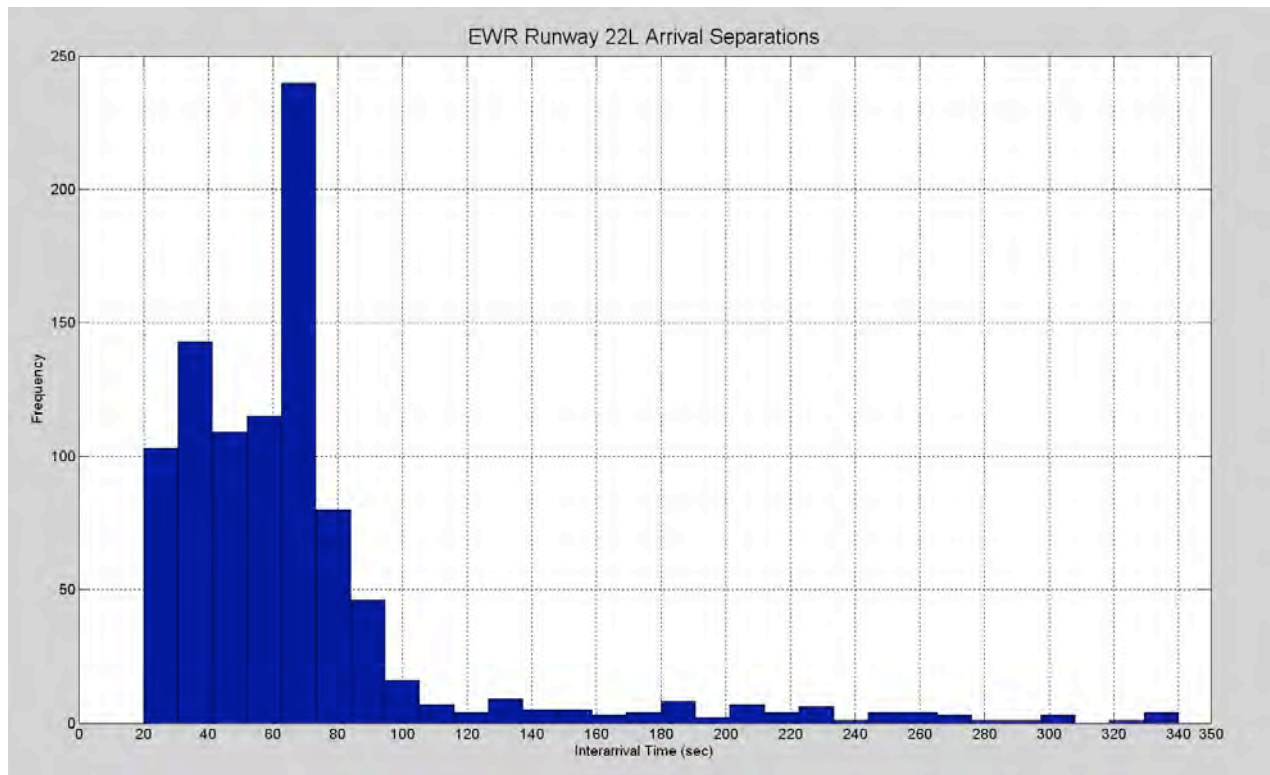
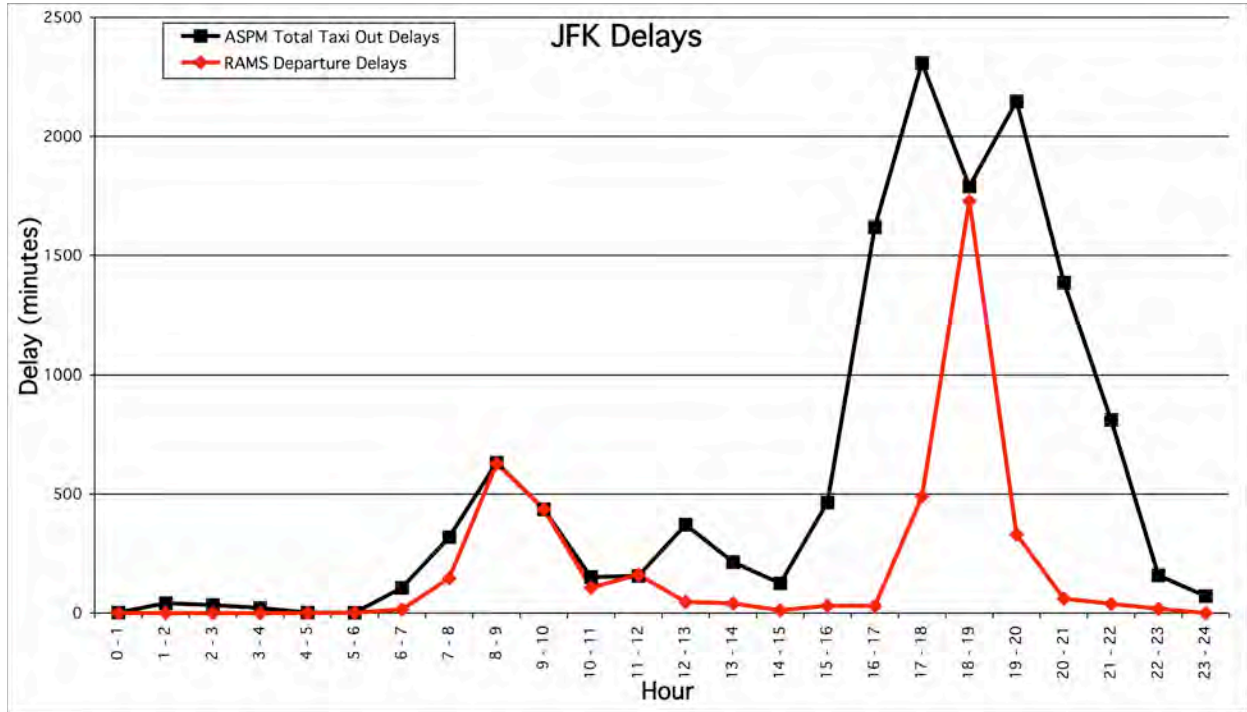


Figure 5.10 EWR Arrival Separations from RAMS Output.

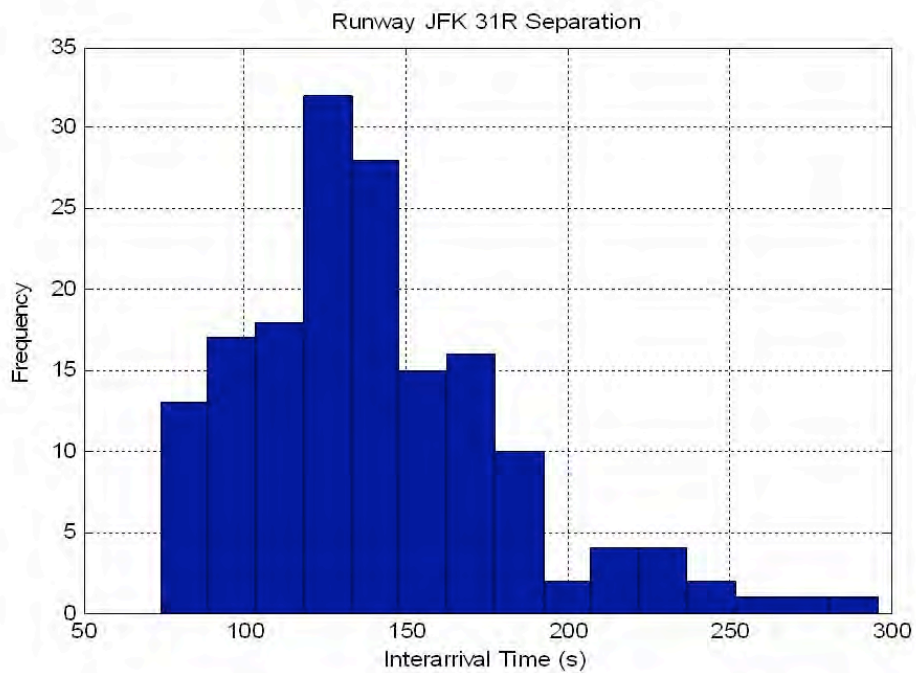
### 5.3. *John Fitzgerald Kennedy Airport*

In terms of delays, JFK gave the best fit between the simulated results and the real data both in magnitude and in the trends.



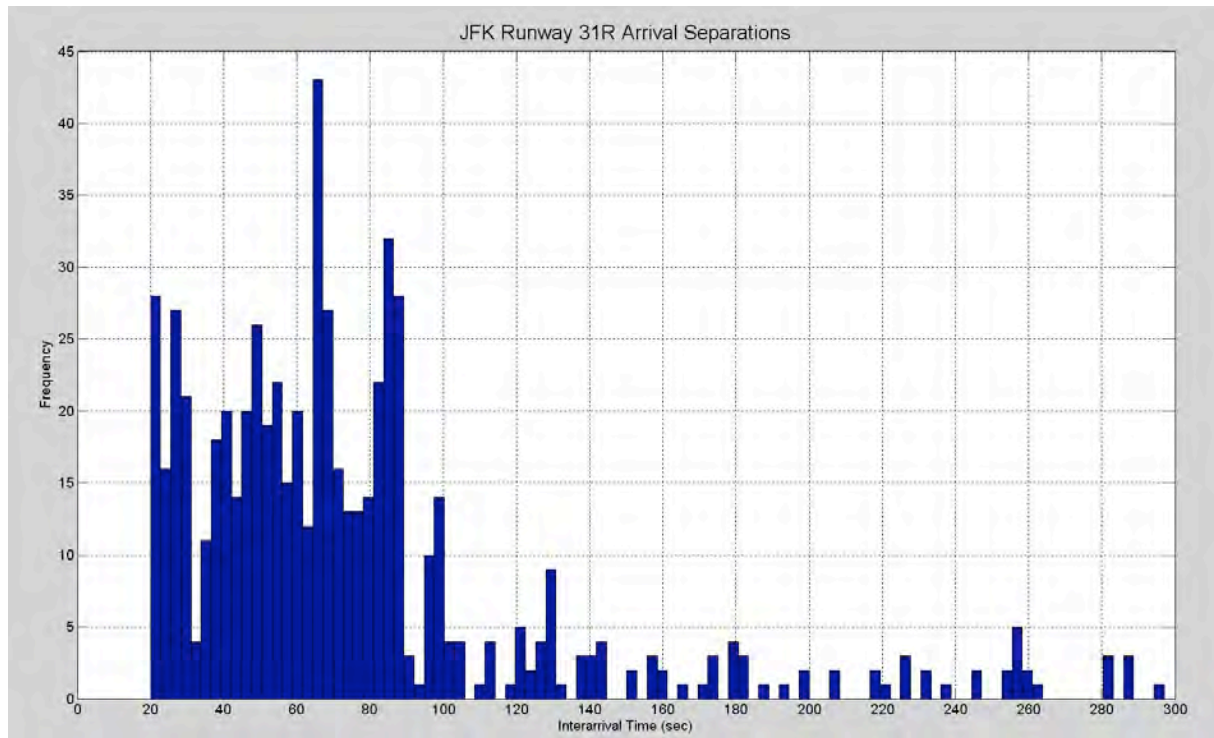
**Figure 5.11 JFK Delays Comparison with ASPM.**

Arrivals operations on both runway 31R and 31L were used for the calibration. From PDARS data the average on runway 31R is equal to 140.4 seconds. The total observations recorded were 173.



**Figure 5.12 JFK 31R Arrival Separations from PDARS Data.**

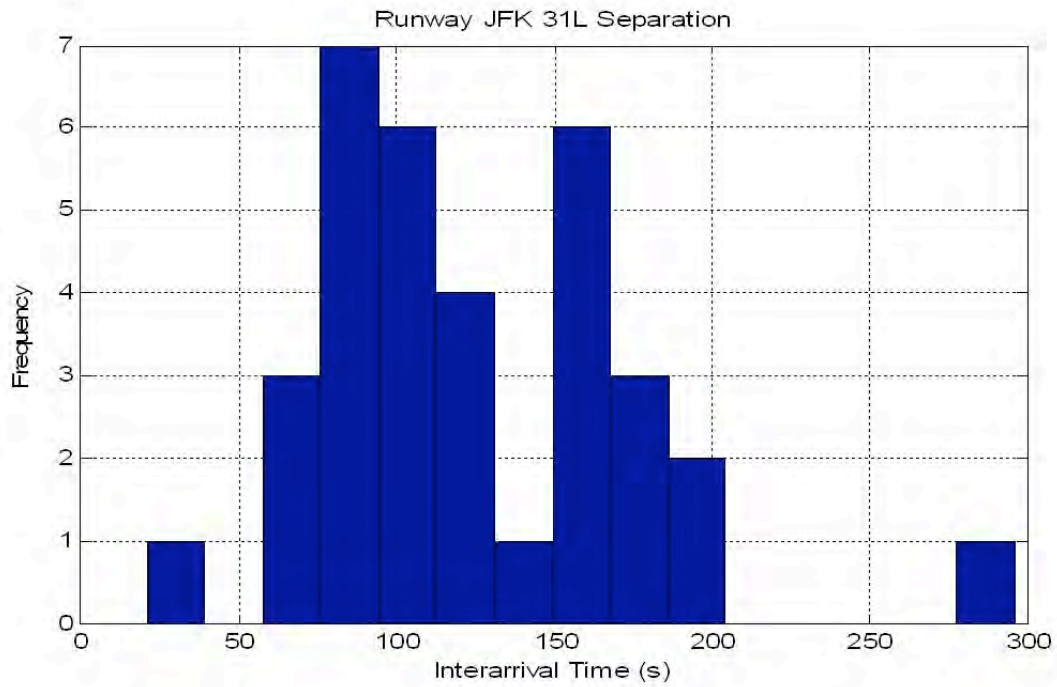
The simulated arrival operations on this runway were 850 with an average of 94.56 seconds. More realistic than the one obtained from PDARS data were during the whole day less than 200 aircraft landed on this runway.



**Figure 5.13 JFK 31R Arrival Separations from RAMS Output.**

Fewer arrival operations were recorded on runway 31L, i.e. 40. The data is not statistically significant. One hundred and forty four operations simulated in RAMS to runway 31R with an average inter-arrival time of 79.31 seconds.





**Figure 5.14 JFK 31L Arrival Separations from PDARS Data.**

The results from the simulation model are very scattered because 31L was heavily used for departures. For this reason the arrivals were few over the day.

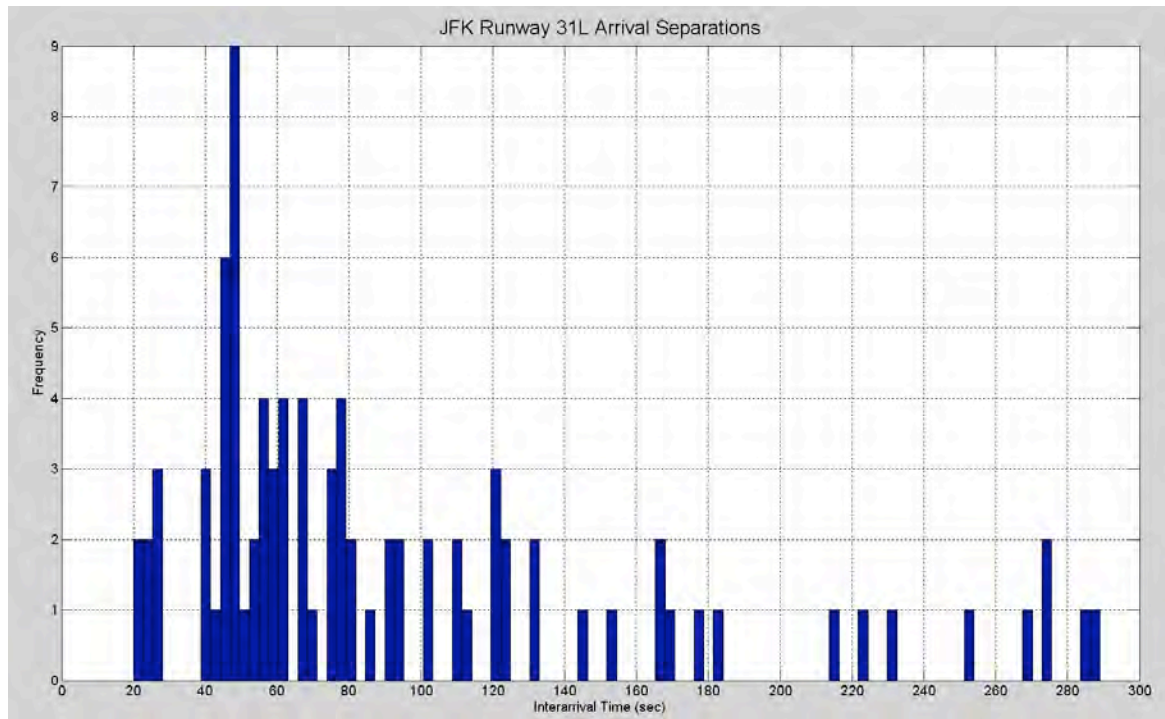


Figure 5.15 JFK 31L Arrival Separations from RAMS Output.

## 6. CONCLUSIONS AND RECOMMENDATIONS

### 6.1. Summary of Results

This study has proven that Data-Communication technologies could yield modest increases in performance of terminal area airspace. Benefits in terms of the total delays experienced by each airport were analyzed using a simulation study. Furthermore reduced workload was recorded in the scenarios with the new communication protocol implemented. Overall a modest gain in the hourly capacity was obtained.

The benefits in terms of total delays were different for the three airports analyzed. LGA simulation provided a gain in the baseline scenario (2006) with Data-Communication segment 2 of 29.5 percent on the average delay per flight. EWR average delay per operation in 2006 was of 38.5 percent. JFK simulation provided the highest benefit with 40.3 percent on the average delay per operation. With the future demand these benefit shrunk significantly.

The workload analysis provided a benefit of 23.7 percent in the hourly controller occupation time. This benefit will remain constant with the increased demand in the future also if the overall values will be significantly higher.

The capacity analysis provided a slight increment that could be translated into capacity multipliers different for each airport. The results are summarized in Table 6-1.

**Table 6-1 Capacity Multipliers Obtained.**

	LGA	EWR	JFK
Arrivals Only (%)	0	1.7	0
Departures Only (%)	4.4	23	27
Mixed Ops. (%)	5.1	12	0

## **6.2. Recommendations**

For the complexity and time-consuming simulation runs it was not possible to complete all the simulations stated in the methodology. This fact demonstrated that probably RAMS was not the right choice for such type of simulation. For the continuous need of calibration between different scenarios RAMS showed to be not completely mature as an airport ground model.

Further analysis should be performed on other terminal area systems where different and maybe higher benefits will be achieved. The complexity of the New York Metroplex area and the interactions between the three major airports yielded to limited benefits in terms of capacity. A higher level of details should be embedded in the simulation scenario such as:

- Detailed gates configuration, and
- Separate tracks for Jets and Turboprops.

The results of our analysis showed also that without any structural improvement none of the hubs studied could sustain any significant improvement in demand. All of them are close to their saturation capacity levels.

## REFERENCES

1. airnav.com. (2007). from [www.airnav.com](http://www.airnav.com).
2. Anderegg, A. (2006). "Creating an Enterprise Architecture for Next Generation Air Transportation System." The Journal of Air Traffic Control **Vol. 48**(Issue 1): P. 22-26.
3. Andreatta, G., Brunetta, L., Righi, L. (2005). "Evaluating Terminal Management Performances Using SLAM: The Case of Athens International Airport." Computers & Operations Research **34**: p. 1532-1550.
4. Blumstein, A. (1957). "A Monte Carlo Analysis of the Ground Controlled Approach System." Operations Research **5**(3): 397-408.
5. Blumstein, A. (1959). "Landing Capacity of a Runway." Operations Research **7**(6): 752-763.
6. Borener, S., Carr, G., Ballard, D., Hasan, S. (2006). "Can NGATS Meet the Demands of the Future?" The Journal of Air Traffic Control **Vol. 48**(Issue 1): P. 34-38.
7. Darby, E. R. (1999). Controller Evaluation of CPDLC Services implemented on Display System Replacement (DSR) Workstation, Study 1-- Initial Assessment of Services transitioned from PVD and Design Development of Additional Services.
8. Darby, E. R. (2000). Controller Evaluation of CPDLC Services implemented on Display System Replacement (DSR) Workstation, Study 1-- Assessment of Build I and IA Human Computer Interfaces, Training and Procedures.
9. Darby, E. R. (2001). CPDLC Build I: Phase 1 - Air Traffic Operational Effectiveness Test.
10. Dunlay, W. J. (2001). Simulations for Redesigned Airspace Flight Procedures. Transportation Research Board, Washington D. C., USA.
11. EUROCONTROL (1999). Model Simulation of the Czech Republic Airspace. Bretigny-sur-Orge, France, EUROCONTROL.
12. FAA (1995). User Benefits of Two-Way Data Link ATC Communications: Aircraft Delay and Flight Efficiency in Congested En Route Airspace.
13. FAA (1996). Benefits of Controller-Pilot Data Link ATC Communications in Terminal Airspace.

14. FAA (2004). Airport Capacity Benchmark Report. Washington, D.C., FAA.
15. Faison, W. E., Given, J. J., Meisner, M. P., Petersen, P. H. (1970). Analytical study of air traffic capacity in the New York metropolitan area Final Report, FAA.
16. Ferra, S. E., Darby, E. R. (2000). CPDPC Build I: Air Ground Integration Study Test Report.
17. Gangel, M., Mehta, G., Muhammad, C., Sekhavat, S., Vora, G., Donohue, G. L. (2004). Terminal Area Capacity Enhancement Concept. System and Information Engineering Design Symposium, Charlottesville, Virginia.
18. Garot, J. M., Ky, P. (2003). The Future Air Transport System in Europe: Vision and Perspectives\ AIAA/ICAS International Air and Space Symprosyum and Exposition: The Next 100 Years. Dayton, Ohio, Usa.
19. Gilbo, E., P. (1993). "Airport capacity: Representation, estimation, optimization." IEEE Transactions on Control Systems Technology **V. 1**(No. 3): p. 144-154.
20. Gilbo, E., P. (1997). "Optimizing airport capacity utilization in air traffic flow management subject to constraints at arrival and departure fixes." IEEE Transactions on Control Systems Technology **V. 5**(No. 5): p. 490-503.
21. Gilbo, E., P. (2003). Arrival/Departure Capacity Tradeoff Optimization: a Case Study at the St. Louis Lambert International Airport (STL). 5th USA/Europe Air Traffic Management R&D Seminar. Budapest.
22. Gilbo, E., P., Howard, K., W. (2001). "Collaborative optimization of arrival and departure traffic flow management strategies at airports." Air transportation systems engineering: p.289-303.
23. Hockaday, S. L. M., Kanafani, K. (1974). "Developments in Airport Capacity Analysis." Transportation Research **Vol. 8**: pp. 171-180.
24. Hoffman, J. (2001). Demand Dependence of Throughput and Delay at New York LaGuardia Airport. McLean, VA, The MITRE Corporation.
25. Horonieff, R., MCKelvey, F. X. (1993). Planning & Design of Airports. New York, USA, McGraw Hill
26. Huang, Y., Xiang, X., Madey, G. (2004). "A Self Manageable Infrastructure for Supporting Web-based Simulations." Proceedings of the 37th Annual Simulation Symposium (ANSS'04)(1080-241X).

27. Hughes, D. (2006). " A Billion a Year for ATC." Aviation Week & Space Technology **Vol. 164**(18): P. 37.
28. Hughes, D. (2006). "Eight Necessities for 2025 (Air Traffic Control)." Aviation Week & Space Technology **Vol. 164**: P. 48-49.
29. ISA-Software (2004). RAMS User Manual Version 5.20.
30. Janic, M. (2000). Air Transport System Analysis and Modelling. Amsterdam, Gordon and Bleach Science Publishers.
31. Janic, M. (2004). "Expansion of Airport Capacity at London Heathrow Airport." Transportation Research Record 1888: p. 7-14.
32. Janic, M., Totic, V. (1982). "Terminal airspace Capacity Model." Transportation Research A V. 16A No. 4: p. 253-260.
33. Leiden, K., Kopardekar, P., Green, S. (2003). Controller Workload Analysis Methodology to Predict Increases in Airspace Capacity. AIAA 3rd Annual Aviation Technology, Intergration, and Operations (ATIO) Conference. Denver, Co, Usa.
34. Majumdar, A., Ochieng, W.Y. (2002). "Factors Affecting Air Traffic Controllers Workload, Multivariate Analysis Based on Simulation Modeling of Controller Workload." Transportation Research Record 1788.
35. Majumdar, A., Polak, J. (2001). "Estimating Capacity of Europe's Airspace Using a Simulation Model of Air Traffic Controller Workload." Transportation Research Record 1744.
36. Martinez, J. C., Trani, A. A., Ioannou, P. G. (2001). "Modeling Airside airport Operations Using General-Purpose, Activity-Based, Discrete- Event Simulation Tools." Transportation Research Record 1744.
37. Massimini, P. A., Dieudonne, J. E., Monticone L. C., Lamiano, D. F., Brestle, E. A. (2000). "Insertion of Controller-Pilot Data Link Communications into the National Airspace System: Is It More Efficient?" IEEE AES System Magazine **September 2000**: p. 25-29.
38. NJ, T. P. A. o. N. (2007). from <http://www.panynj.gov/CommutingTravel/airports/html/>.
39. Odoni, A. R. (1971). Modeling for Air Traffic Control. Cambridge, Mass.

40. Odoni, A. R. (1991). Issues in modeling a national network of airports. Winter Simulation Conference Proceedings. Phoenix, AZ, USA.
41. Odoni, A. R., de Neufville, R. (2003). Airport Systems. Planning, Design and Management, McGraw Hill.
42. Odoni, A. R., Deyst, J., Feron, E., Hansman, R., Khan, K., Kuchar, J., Simpson, R., (1997). Existing and Required Modeling Capabilities for Evaluating ATM Systems and concepts, International Center for Air Transportation, MIT.
43. Prutzman, A. (2007). Tower Data Link System (TDLS), Business Case Update, Northrop Grumman.
44. Rakas, J., Mumayiz, S. (2001). Airport-Airspace Simulations for Capacity Evaluation. Transportation Research Board 79th Meeting, Washington, DC, USA.
45. Riley, V. (1992). Human Factors Issues of Data Link: Application of a System Analysis. S. International. Warrendale, PA, Usa.
46. Rodgers, M., Embt, D., Colligan, W. (1997). A Multi-Center NASSIM Analysis of the Effects of Data Link Equipage Rates on voice Communications, Federal Aviation Administration and CSSI.
47. Ryan, P. R. (1992). Airline Perspective on Data Link. S. International. Warrendale, PA, Usa.
48. Slattery, H. F. (1970). New York air traffic capacity study Final Report. Atlantic City, FAA.
49. Smith, N. M., Lee, P. U., Prevot, T., Mercer, J., Palmer, E.A., Battiste, V., Johnson, W. (2004). A Human-in-the-Loop Evaluation of Air-Ground Trajectory Negotiation. AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum. Chicago, IL, Usa.
50. Stamatopoulos, M. A., Konstantinos, Z. G., Odoni, A. R. (2004). "A Decision Support System for Airport Strategic Planning." Transportation Research C **12**: p. 91-117.
51. Swedish, W., J. (1981). Upgraded FAA Airfield Capacity Model Vols. 1 & 2. MITRE Corp. MTR-81W16, MITRE Corporation.
52. Swedish, W., J., Barrer, J. N., Kuzminski, P. (2005). Analyzing the Runway Capacity of Complex Airports. AIAA 5th Aviation, Technology, Integration, and Operations Conference. Arlington, Virginia, Usa.



53. Terrab, M., Odoni, A. (1993). "Strategic Flow Mangement for Air Traffic Control." Operations Research **Vol. 41 N. 1 Jan/Feb.**
54. Tofukujl, N. (1993). "An Enroute ATC Simulation Experiment for Sector Capacity Estimation." IEEE Transactions on Control Systems Technology **Vol. I, No. 3:** pp. 138-143.
55. Trani, A. A., Hobeika, A. G., Sherali, H. D., Kim, B. J. (1993). "Microcomputer Model for Design and Location of Runway Exits." Journal of Transportation Engineering **Vol. 119, n. 3:** p. 385- 401.
56. Venkatakrisnan, C. S., Barnett, A., Odoni, A. O. (1993). "Landings at Logan Airport: Describing and increasing airport capacity." Transportation Science **V. 27**(No. 3): p. 211-227.
57. Waller, M. C., Lohr, G. W. (1989). A piloted simulation of data link ATC message exchange. NASA Langley Research Center, Hampton, VA.

## **APPENDIX A ACRONYMS**

ACM Airfield Capacity Model  
ATO Air Transport Office  
ATSL Air Transportation Systems Laboratory  
ASPM Aviation System Performance Metrics  
CPDLC Controller Pilot Data-Link Communications  
ETMS Enhanced Traffic Management System  
FAA Federal Aviation Administration  
IFR Instrument Flight Rule  
IMC Instrument Meteorological Conditions  
MMC Marginal Meteorological Conditions  
NAS National Airspace System  
NASA National Aeronautics and Space Administration  
NGATS Next Generation Air Transportation System  
PDARS Performance Data Analysis and Reporting System  
RAMS Reorganized ATC Mathematical Simulator  
ROT Runway Occupancy Time  
TAAM Total Airspace and Airport Modeler  
TAF Terminal Area Forecast  
TRACON Terminal RADar CONtro  
VFR Visual Flight Rule  
VMC Visual Meteorological Conditions

# APPENDIX B FLOWCHART OF THE ANALYSIS

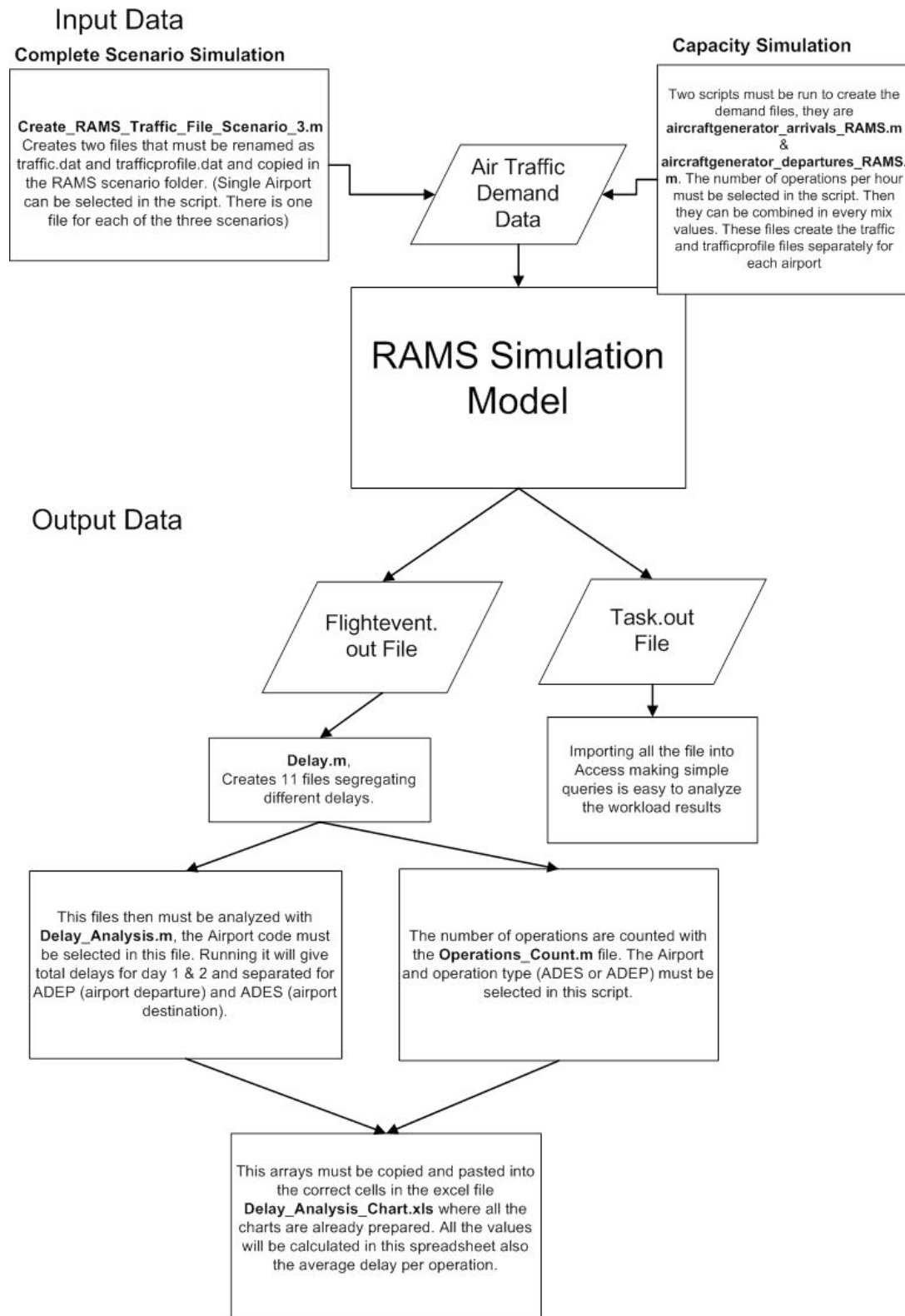


Figure B. Flowchart of the Analysis with the Matlab Files Used.

# APPENDIX C 1 MATLAB FILE TO CREATE RAMS DEMAND FILES

```
%-----  
% This m-file creates a RAMS Traffic File from the ATO Forecast struct  
% array  
%-----  
  
clc;  
% clear all;  
global ATO_Forecast  
  
Airport_ID_List = ['LGA'; 'EWR'; 'JFK'];  
  
% -----  
% Loading files  
% -----  
FolderName = 'E:\Gabriele\MATLAB Files\ATO-P Forecasts\  
FileName = 'forecast_Q2_PEAK_2006_2006_smoothed_filtered';  
disp('Loading...');  
load([FolderName, FileName, '.mat']);  
  
% -----  
% This function will count the number of departures at each airport  
% -----  
[Number_of_Departures_JFK] = Departures_count('JFK');  
[Number_of_Arrivals_JFK_31L] = Arrivals_Balance(Number_of_Departures_JFK);  
[Total_Number_of_Arrivals_JFK] = Arrivals_count('JFK');  
Percentage_of_Arrivals_to_31L = Number_of_Arrivals_JFK_31L ./  
Total_Number_of_Arrivals_JFK;  
Percentage_of_Arrivals_to_31L = min(Percentage_of_Arrivals_to_31L,1);  
RWY31L_Count = zeros(24,1);  
  
% -----  
% Setup box around airports  
% -----  
Polygon_Latitudes = [42.34207 42.39581 41.88247 41.62858 41.16985 39.94778  
39.67062 39.6875 39.89369 40.06081 40.70777 40.94836 41.98255 42.3156];  
Polygon_Longitudes = [-73.99614 -73.60829 -72.92616 -72.36090 -71.91034 -  
73.22402 -73.56126 -74.18774 -74.97654 -75.14857 -75.50947 -75.43555 -75.1346  
-74.84523];  
  
% -----  
% Writing file  
% -----  
Number_Of_Airports = length(Airport_ID_List);  
Number_Of_Flights = length(ATO_Forecast);  
Number_Of_Departures = 0;  
Number_Of_Arrivals = 0;  
Number_Of_Military_Flights = 0;  
Number_Of_VFR_Flights = 0;  
Number_Of_Skipped_Flights = 0;  
Unique_Traffic_Callsign_Counter = 0;  
Unique_Traffic_Callsigns = [];
```

```

Unique_Traffic_Callsigns_Count = [];
fid1 = fopen(['RAMS_Traffic_File_NYC_Complete_Scenario_1', FileName, '.txt'],
'w');
fid2 = fopen(['RAMS_Traffic_Profile_File_NYC_Complete_Scenario_1', FileName,
'.txt'], 'w');

for j = 1:Number_Of_Airports

    Airport_ID = Airport_ID_List(j,:);

    if strcmp(Airport_ID, 'LGA') == 1 % LGA
        Approach_Fixes_Names = {'APP_N', 'APP_N_W', 'APP_N_E',
'APP_S', 'APP_S_W'}';
        Approach_Entry_Fixes_Names = {'LGA_N_ENTRY_22', 'LGA_NW_ENTRY_22',
'LGA_NE_ENTRY_22', 'LGA_S_ENTRY_22', 'LGA_SW_ENTRY_22'}';
        Approach_Entry_Fixes_Longitude = [-73.60829 -74.84523 -72.36090 -
74.97654 -75.50947];
        Approach_Entry_Fixes_Latitude = [42.39580 42.3156 41.62858 39.89369
40.70777];
        Approach_Path_Names = {'LGA_APP_N_22', 'LGA_APP_NW_22',
'LGA_APP_NE_22', 'LGA_APP_S_22', 'LGA_APP_SW_22'}';

        Departure_Path_Names = {'LGA_DEP_S_13', 'LGA_DEP_NE_13',
'LGA_DEP_NW_13'}';
        Departure_Fixes_Longitude = [-74.618 -72.151 -75.471]';
        Departure_Fixes_Latitude = [39.702 41.527 41.464]';
        elseif strcmp(Airport_ID, 'EWR') == 1 % EWR
            Approach_Fixes_Names = {'APP_N', 'APP_N_W', 'APP_N_E',
'APP_S', 'APP_S_E'}';
            Approach_Entry_Fixes_Names = {'EWR_N_ENTRY_22L',
'EWR_NW_ENTRY_22L', 'EWR_NE_ENTRY_22L', 'EWR_SW_ENTRY_22L',
'EWR_SE_ENTRY_22L'}';
            Approach_Entry_Fixes_Longitude = [-73.99614 -75.1346 -72.92616 -
75.14857 -73.22402 ];
            Approach_Entry_Fixes_Latitude = [42.99614 41.98255 41.88247 40.06081
39.94778];
            Approach_Path_Names = {'EWR_APP_N_22L', 'EWR_APP_NW_22L',
'EWR_APP_NE_22L', 'EWR_APP_SW_22L', 'EWR_APP_SE_22L'}';

            Departure_Path_Names = {'EWR_DEP_NE_22R', 'EWR_DEP_S_22R',
'EWR_DEP_W_22R'}';
            Departure_Fixes_Longitude = [-72.6550 -74.7247 -75.4397]';
            Departure_Fixes_Latitude = [41.6468 39.7016 40.4238]';
            elseif strcmp(Airport_ID, 'JFK') == 1 % JFK
                Approach_Fixes_Names = {'APP_S_W', 'APP_N_E', 'APP_S',
'APP_S_E'}';
                Approach_Entry_Fixes_Names = {'JFK_NW_ENTRY_13L',
'JFK_NE_ENTRY_13L', 'JFK_SE_ENTRY_13L', 'JFK_N_ENTRY_13L'}';
                Approach_Entry_Fixes_Longitude = [-75.374 -71.9313 -74.19525 -
73.8087];
                Approach_Entry_Fixes_Latitude = [41.188 41.2568 39.6804 42.3533];
                Approach_Path_Names = {'JFK_APP_NW_13L', 'JFK_APP_NE_13L',
'JFK_APP_SE_13L', 'JFK_APP_N_13L'}';

                Departure_Path_Names = {'JFK_DEP_W_13R', 'JFK_DEP_S_13R',
'JFK_DEP_E_13R', 'JFK_DEP_NE_13R', 'JFK_DEP_N_13R', 'JFK_DEP_NW_13R'}';
                Departure_Fixes_Longitude = [-75.3591 -73.4974 -72.0373 -72.1718 -
73.3378 -75.4002]';

```

```

        Departure_Fixes_Latitude    = [40.2065 39.8046 40.6533 41.5027 42.2895
41.6671]';
    end

    Number_Of_Approach_Fixes    = length(Approach_Fixes_Names);
    Number_Of_Departure_Paths    = length(Departure_Path_Names);

    for i = 1:Number_Of_Flights

        Error = 'False';

        if mod(i, 1000) == 0
            disp(['Number of flights processed for ', Airport_ID, ' : ',
num2str(i)]);
        end

        Departure_Airport = ATO_Forecast(i).etms_departure_airport;
        Arrival_Airport   = ATO_Forecast(i).etms_arrival_airport;

        if strcmp(Departure_Airport, Airport_ID) == 1 ||
strcmp(Arrival_Airport, Airport_ID) == 1 % Only save flight to/from Airport_ID

            % -----
            % Get Operation Type
            % -----
            if strcmp(Departure_Airport, Airport_ID) == 1 % Departure
                Number_Of_Departures = Number_Of_Departures + 1;
                OperationType = 'Departure';
            else %Arrival
                Number_Of_Arrivals = Number_Of_Arrivals + 1;
                OperationType = 'Arrival';
            end % if strcmp(etms_departure_airport, Airport_ID) == 1

            % -----
            % Format Output
            % -----
            if strcmp(OperationType, 'Arrival') == 1 % Adjust Entry time of
arrivals
                if strcmp(ATO_Forecast(i).flight_plan_type, 'VFR') == 0 % Not
VFR Flight
                    EntryTime = ATO_Forecast(i).departure_time(end - 7:end); %
HH:MM:SS
                    EntryTime_InDays = datenum(EntryTime); % Time in Fraction
of Days
                else % VFR Flight
                    EntryTime = ATO_Forecast(i).arrival_time(end - 7:end); %
HH:MM:SS
                    EntryTime_InDays = datenum(EntryTime) - 30/60/24; % Time
in Fraction of Days
                end

                EntryTime = datestr(EntryTime_InDays, 'HHMMSS');
                EntryTime_Plus24Hrs = EntryTime;
                EntryTime_Plus24Hrs(1:2) = num2str(str2num(EntryTime(1:2)) +
24);
            elseif strcmp(OperationType, 'Departure') == 1
                EntryTime = ATO_Forecast(i).departure_time(end - 7:end); %

```

```

HH:MM:SS
Fraction of Days
    EntryTime_InDays = datenum(EntryTime) + randn/60/24; % Time in
    EntryTime = datestr(EntryTime_InDays, 'HHMMSS');
    EntryTime_Plus24Hrs = EntryTime;
    EntryTime_Plus24Hrs(1:2) = num2str(str2num(EntryTime(1:2)) +
24);
    end

    Traffic_Callsign = ATO_Forecast(i).acid;
    % Skip if Military Flights
    if strcmp(ATO_Forecast(i).user_class, 'M') == 1
        disp(['WARNING: Military flight skipped for index: ',
num2str(i)]);
        Number_Of_Military_Flights = Number_Of_Military_Flights + 1;
        continue;
    end

    % Save unique traffic callsigns. If duplicate found, add letter
    % to end of it. This is to avoid RAMS getting crazy.
    if isempty(strmatch(Traffic_Callsign, Unique_Traffic_Callsigns,
'exact')) == 1 % Unique traffic callsign found
        Unique_Traffic_Callsign_Counter =
Unique_Traffic_Callsign_Counter + 1;
        Unique_Traffic_Callsigns{Unique_Traffic_Callsign_Counter,1} =
Traffic_Callsign;

Unique_Traffic_Callsigns_Count(Unique_Traffic_Callsign_Counter,1) = 1;
        Traffic_Callsign = [Traffic_Callsign, 'A1'];
        Traffic_Callsign_Plus24Hrs = Traffic_Callsign;
        Traffic_Callsign_Plus24Hrs(end) = '2';
    else % Duplicate traffic callsign found
        Index_Of_Unique_Traffic_Callsign = strmatch(Traffic_Callsign,
Unique_Traffic_Callsigns, 'exact');

Unique_Traffic_Callsigns_Count(Index_Of_Unique_Traffic_Callsign,1) =
Unique_Traffic_Callsigns_Count(Index_Of_Unique_Traffic_Callsign,1) + 1;
        % Add letter A B C... to end of traffic call sign
        Traffic_Callsign = [Traffic_Callsign, char(64 +
Unique_Traffic_Callsigns_Count(Index_Of_Unique_Traffic_Callsign,1)), '1'];
        Traffic_Callsign_Plus24Hrs = Traffic_Callsign;
        Traffic_Callsign_Plus24Hrs(end) = '2';
    end

    if isempty(ATO_Forecast(i).bada_type) == 0
        Aircraft_Model = ATO_Forecast(i).bada_type;
    else
        if strcmp(ATO_Forecast(i).flight_plan_type, 'VFR') == 1
            Aircraft_Model = 'BE20';
        else
            Aircraft_Model = '???';
            disp(['ERROR: Invalid aircraft model type for flight
index: ', num2str(i)]);
            Number_Of_Skipped_Flights = Number_Of_Skipped_Flights + 1;
            continue;
        end
    end

```

```

        end
    end
    if ATO_Forecast(i).user_class == 'C'
        Flight_Category = 'C';
    elseif ATO_Forecast(i).user_class == 'F'
        Flight_Category = 'F';
    elseif ATO_Forecast(i).user_class == 'G'
        Flight_Category = 'G';
    elseif ATO_Forecast(i).user_class == 'M'
        Flight_Category = 'M';
    elseif ATO_Forecast(i).user_class == 'T'
        Flight_Category = 'O';
    elseif ATO_Forecast(i).user_class == 'O'
        Flight_Category = 'O';
    elseif strcmp(ATO_Forecast(i).flight_plan_type, 'VFR') == 1
        Flight_Category = 'G';
    else
        Flight_Category = '???';
        disp(['ERROR: Unknown user class "',
ATO_Forecast(i).user_class, '" for flight index: ', num2str(i)]);
        Number_Of_Skipped_Flights = Number_Of_Skipped_Flights + 1;
        continue;
    end
    NAV_Equipment = 'DefaultACNavEquipment';

    % Find closest approach entry point from departure airport
    if strcmp(OperationType, 'Arrival') == 1

        % Skip if VFR flight
        if strcmp(ATO_Forecast(i).flight_plan_type, 'VFR') == 0
            Altitude_Entry_Level = ATO_Forecast(i).dept_elev/100; %
Convert to Flight Level
            Altitude_Cruise_Level = ATO_Forecast(i).filed_altitude;
            Altitude_Exit_Level = ATO_Forecast(i).arr_elev/100; %
Convert to Flight Level

            % Remove points inside box
            if isempty(ATO_Forecast(i).waypoints) == 1
                disp(['ERROR: Flight has no waypoints! Flight index:
', num2str(i)]);
                Number_Of_Skipped_Flights = Number_Of_Skipped_Flights
+ 1;
                continue
            end
            Waypoints_Latitude = ATO_Forecast(i).waypoints.latitude;
            Waypoints_Longitude = ATO_Forecast(i).waypoints.longitude;
            [InOut_Box_Points] = inpolygon(Waypoints_Longitude,
Waypoints_Latitude, Polygon_Longitudes, Polygon_Latitudes);
            Index_Of_Outside_Points = find(InOut_Box_Points == 0);
            Waypoints_Latitude =
Waypoints_Latitude(Index_Of_Outside_Points);
            Waypoints_Longitude =
Waypoints_Longitude(Index_Of_Outside_Points);

            % Find closest approach entry point based on last waypoint
            if length(Waypoints_Longitude) == 0
                hold on;
                plot(Polygon_Longitudes, Polygon_Latitudes);

```



```

        plot(ATO_Forecast(i).dept_lon,
ATO_Forecast(i).dept_lat, 'x')
        plot(ATO_Forecast(i).arr_lon, ATO_Forecast(i).arr_lat,
'+')
        disp(['ERROR: Flight to short! Flight index: ',
num2str(i)]);
        Number_Of_Skipped_Flights = Number_Of_Skipped_Flights
+ 1;
        continue;
    end
    Last_Waypoint_Longitude = Waypoints_Longitude(end);
    Last_Waypoint_Latitude = Waypoints_Latitude(end);
    Distance_LastWaypoint_To_Approach =
deg2nm(distance(Last_Waypoint_Latitude, Last_Waypoint_Longitude,
Approach_Entry_Fixes_Latitude, Approach_Entry_Fixes_Longitude));
    [Closest_Approach_Fix_Distance,
Closest_Approach_Fix_Index] = min(Distance_LastWaypoint_To_Approach);
    % Closest_Approach_Fix_Name =
char(Approach_Fixes_Names{Closest_Approach_Fix_Index});
    Closest_Approach_Entry_Fix_Name =
char(Approach_Entry_Fixes_Names{Closest_Approach_Fix_Index});
    Closest_Approach_Path_Name =
char(Approach_Path_Names{Closest_Approach_Fix_Index});
    else % VFR Flight

        Number_Of_Waypoints = 0;
        Waypoints_Latitude = [];
        Waypoints_Longitude = [];
        Number_Of_VFR_Flights = Number_Of_VFR_Flights + 1;

        Altitude_Entry_Level = 180; % Convert to Flight Level
        Altitude_Cruise_Level = 180;
        Altitude_Exit_Level = 0; % Convert to Flight Level

        % Assign VFR to random approach
        Approach_Fix_Index =
find(randperm(Number_Of_Approach_Fixes) == 1);
        Closest_Approach_Entry_Fix_Name =
char(Approach_Entry_Fixes_Names(Approach_Fix_Index));
        Closest_Approach_Path_Name =
char(Approach_Path_Names{Approach_Fix_Index});
    end

    Departure_Airport = 'ADEP'; % Default Departure Airport

    elseif strcmp(OperationType, 'Departure') == 1 % Find closest
departure exit point to destination airport

        % Skip if VFR flight
        if strcmp(ATO_Forecast(i).flight_plan_type, 'VFR') == 0

            Altitude_Entry_Level = ATO_Forecast(i).arr_elev/100; %
Convert to Flight Level
            Altitude_Cruise_Level = ATO_Forecast(i).filed_alititude;
            Altitude_Exit_Level = ATO_Forecast(i).dept_elev/100; %
Convert to Flight Level

            % Find closest departure exit point to destination airport

```

```

        Arrival_Airport_Longitude =
double(ATO_Forecast(i).arr_lon);
        Arrival_Airport_Latitude =
double(ATO_Forecast(i).arr_lat);
        Distance_Departure_To_Destination =
deg2nm(distance(Arrival_Airport_Latitude, Arrival_Airport_Longitude,
Departure_Fixes_Latitude, Departure_Fixes_Longitude));
        [Closest_Departure_Fix_Distance,
Closest_Departure_Fix_Index] = min(Distance_Departure_To_Destination);
        Closest_Departure_Path_Name =
char(Departure_Path_Names{Closest_Departure_Fix_Index});

        else % VFR Flight

            Number_Of_Waypoints = 0;
            Waypoints_Latitude = [];
            Waypoints_Longitude = [];
            Number_Of_VFR_Flights = Number_Of_VFR_Flights + 1;

            Altitude_Entry_Level = 0; % Convert to Flight Level
            Altitude_Cruise_Level = 180;
            Altitude_Exit_Level = 180; % Convert to Flight Level

            % Assign VFR to random departure
            Departure_Paths_Index =
find(randperm(Number_Of_Departure_Paths) == 1);
            Closest_Departure_Path_Name =
char(Departure_Path_Names{Departure_Paths_Index});

        end

        Arrival_Airport = 'ADES'; % Default Departure Airport
    end

    if strcmp(OperationType, 'Arrival') == 1
        Departure_Runway = 'RWY';
        if strcmp(Airport_ID, 'LGA') == 1
            Arrival_Runway = 'LGA22';
        elseif strcmp(Airport_ID, 'EWR') == 1
            Arrival_Runway = 'EWR22L';
        elseif strcmp(Airport_ID, 'JFK') == 1
            if strcmp(Closest_Approach_Path_Name, 'JFK_APP_NE_13L') ==
1
                Arrival_Runway = 'JFK13L';
            elseif strcmp(Closest_Approach_Path_Name,
'JFK_APP_NW_13L') == 1 || strcmp(Closest_Approach_Path_Name, 'JFK_APP_N_13L')
== 1 || strcmp(Closest_Approach_Path_Name, 'JFK_APP_SE_13L') == 1
                Random_Number = rand;
                Arrival_Time =
(datenum(ATO_Forecast(i).arrival_time(end - 7:end)) - datenum('00:00:00')) *
24; % Hrs

                Arrival_Time_Ceil = ceil(Arrival_Time);
                if Arrival_Time_Ceil > 24 % Assign past midnight to
first bin

                    Arrival_Time_Ceil = 1;
                end
                Percentage_of_Arrivals_to_31L_ThisHour =

```

```

Percentage_of_Arrivals_to_31L(Arrival_Time_Ceil);
    if Random_Number >
Percentage_of_Arrivals_to_31L_ThisHour
        Arrival_Runway = 'JFK13L';
    else
        RWY31L_Count(Arrival_Time_Ceil) =
RWY31L_Count(Arrival_Time_Ceil) + 1;
        Arrival_Runway = 'JFK13L';
        Closest_Approach_Path_Name(end) = 'L';
    end
end
end
elseif strcmp(OperationType, 'Departure') == 1
    if strcmp(Airport_ID, 'LGA') == 1
        Departure_Runway = 'LGA13';
    elseif strcmp(Airport_ID, 'EWR') == 1
        Departure_Runway = 'EWR22R';
    elseif strcmp(Airport_ID, 'JFK') == 1
        Departure_Runway = 'JFK13R';
    end
    Arrival_Runway = 'RWY';
end

% -----
% Write Output
% -----

% -----
% Traffic File
% -----
% First day
fprintf(fid1, '%s %s %s %s %s %s %s %s %s %s %s %s %.0f %.0f %.0f\n',
EntryTime, Traffic_Callsign, Departure_Airport, Departure_Runway,
Arrival_Airport, ...
    Arrival_Runway, Aircraft_Model, Flight_Category,
NAV_Equipment, Altitude_Entry_Level, ...
    Altitude_Cruise_Level, Altitude_Exit_Level);
% Second day
fprintf(fid1, '%s %s %s %s %s %s %s %s %s %s %s %s %.0f %.0f %.0f\n',
EntryTime_Plus24Hrs, Traffic_Callsign_Plus24Hrs, Departure_Airport,
Departure_Runway, Arrival_Airport, ...
    Arrival_Runway, Aircraft_Model, Flight_Category,
NAV_Equipment, Altitude_Entry_Level, ...
    Altitude_Cruise_Level, Altitude_Exit_Level);

% -----
% Traffic Profile File
% -----
% First day
if strcmp(OperationType, 'Arrival') == 1
    Number_Of_Waypoints = length(Waypoints_Latitude);
    for Waypoint_Number = 1:Number_Of_Waypoints
        fprintf(fid2, '%s %f %f %s %.0f\n', Traffic_Callsign,
Waypoints_Latitude(Waypoint_Number), Waypoints_Longitude(Waypoint_Number),
EntryTime, Waypoint_Number);
    end
end

```

```

        fprintf(fid2, '%s %s %s %.0f\n', Traffic_Callsign,
Closest_Approach_Entry_Fix_Name, EntryTime, Number_Of_Waypoints + 1);
        fprintf(fid2, '%s %s %s %.0f\n', Traffic_Callsign,
Closest_Approach_Path_Name, EntryTime, Number_Of_Waypoints + 2);
        elseif strcmp(OperationType, 'Departure') == 1
            fprintf(fid2, '%s %s %s %s\n', Traffic_Callsign,
Closest_Departure_Path_Name, EntryTime, '1');
        end % if strcmp(OperationType, 'Arrival') == 1
        % Second day
        if strcmp(OperationType, 'Arrival') == 1
            Number_Of_Waypoints = length(Waypoints_Latitude);
            for Waypoint_Number = 1:Number_Of_Waypoints
                fprintf(fid2, '%s %f %f %s %.0f\n',
Traffic_Callsign_Plus24Hrs, Waypoints_Latitude(Waypoint_Number),
Waypoints_Longitude(Waypoint_Number), EntryTime_Plus24Hrs, Waypoint_Number);
            end
            %
            fprintf(fid2, '%s %s %s %.0f\n',
Traffic_Callsign, Closest_Approach_Fix_Name, EntryTime, Number_Of_Waypoints +
1);
            fprintf(fid2, '%s %s %s %.0f\n', Traffic_Callsign_Plus24Hrs,
Closest_Approach_Entry_Fix_Name, EntryTime_Plus24Hrs, Number_Of_Waypoints +
1);
            fprintf(fid2, '%s %s %s %.0f\n', Traffic_Callsign_Plus24Hrs,
Closest_Approach_Path_Name, EntryTime_Plus24Hrs, Number_Of_Waypoints + 2);
            elseif strcmp(OperationType, 'Departure') == 1
                fprintf(fid2, '%s %s %s %s\n', Traffic_Callsign_Plus24Hrs,
Closest_Departure_Path_Name, EntryTime_Plus24Hrs, '1');
            end % if strcmp(OperationType, 'Arrival') == 1

        end % if strcmp(etms_departure_airport, Airport_ID) == 1 ||
strcmp(etms_arrival_airport, Airport_ID) == 1
        end % for i = 1:Number_Of_Flights
    end % for j = 1:Number_Of_Airports
fclose('all');

disp(' ');
disp(['Number of departures from NYC: ', num2str(Number_Of_Departures)]);
disp(['Number of arrivals from NYC : ', num2str(Number_Of_Arrivals)]);
disp(['Number of skipped military flights in NYC : ',
num2str(Number_Of_Military_Flights)]);
disp(['Number of VFR flights in NYC : ', num2str(Number_Of_VFR_Flights)]);
disp(['Number of skipped flights in NYC : ',
num2str(Number_Of_Skipped_Flights)]);

```

# APPENDIX C 2 MATLAB FILE TO CREATE DETAILED DELAY FILES

```
clear all
clc

RAMS_Version = '5.27.33';

% Field 1: Time;
% Field 2: CallSign;
% Field 3: EventCategory;
% Field 4: Event;
% Field 5: String1;
% Field 6: String2;
% Field 7: String3;
% Field 8: String4;
% Field 9: String5;
% Field 10: String6;
% Field 11: Num1;
% Field 12: Num2;
% Field 13: Num3;
% Field 14: Num4;
% Field 15: Num5;
% Field 16: Num6;
% Field 17: Lat;
% Field 18: Long;
% Field 19: AFL;
% Field 20: RFL;
% Field 21: CFL;
% Field 22: SpeedNMhr;
% Field 23: GCHeading;
% Field 24: VelXnmhr;
% Field 25: VelYnmhr;
% Field 26: VelZftmin;
% Field 27: deltaTime;
% Field 28: fuelburnrate;
% Field 29: fuelburn;
% Field 30: deltaSpeed;
% Field 31: deltaAlt;
% Field 32: deltaDist;
% Field 33: deltaAngle;
% Field 34: Attitude;
% Field 35: ADEP;
% Field 36: ADES;
% Field 37: model;
% RAMS 5.27.20:
%   Field 38: center;
%   Field 39: PhySector;
%   Field 40: ContSector;
%   Field 41: Route;
% RAMS 5.27.33:
%   Field 38: cat;
%   Field 39: center;
%   Field 40: PhySector;
%   Field 41: ContSector;
%   Field 42: Route;
```

```

fid = fopen('C:\Documents and Settings\genea\Desktop\flightevent.out','r');

count = 0;
Simulation_Time_End = '25:00:00';

DelayHoldInAir = [];
EODelayHoldInAir = [];
HS = [];
EOHS = [];
GRD_DepartureQ = [];
GRD_EODepartureQ = [];
GRD_DelayRunwayCrossing = [];
GRD_EODelayRunwayCrossing = [];
GRD_GLDelayCapacity = [];
GRD_EOGLDelayCapacity = [];
GRD_GLDelaySeparation = [];
GRD_EOGLDelaySeparation = [];
GRD_DEPGATEblocked = [];
GRD_EODEPGATEblocked = [];
GRD_GATEDEP = [];
RWY_TOUCHDOWN = [];
RWY_NODEA = [];
RWY_NODEB = [];

DelayHoldInAir_Count = 0;
EODelayHoldInAir_Count = 0;
HS_Count = 0;
EOHS_Count = 0;
GRD_DepartureQ_Count = 0;
GRD_EODepartureQ_Count = 0;
GRD_DelayRunwayCrossing_Count = 0;
GRD_EODelayRunwayCrossing_Count = 0;
GRD_GLDelayCapacity_Count = 0;
GRD_EOGLDelayCapacity_Count = 0;
GRD_GLDelaySeparation_Count = 0;
GRD_EOGLDelaySeparation_Count = 0;
GRD_DEPGATEblocked_Count = 0;
GRD_EODEPGATEblocked_Count = 0;
GRD_GATEDEP_Count = 0;
RWY_TOUCHDOWN_Count = 0;
RWY_NODEA_Count = 0;
RWY_NODEB_Count = 0;

field_names = fgetl(fid);
while feof(fid) == 0

    if mod(count, 10000) == 0
        disp(['Read Records: ', num2str(count)]);
    end

    %     if count == 100000
    %         break;
    %     end

```

```

aline = fgetl(fid);
count = count + 1;

if strcmp(RAMS_Version, '5.27.20') == 1
    Split_Line = textscan(aline,
'%s%s%s%s%s%s%s%s%s%s%f%f%f%f%f%f%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s'
's%s', 'delimiter', ';');
    elseif strcmp(RAMS_Version, '5.27.33') == 1
        Split_Line = textscan(aline,
'%s%s%s%s%s%s%s%s%s%s%f%f%f%f%f%f%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s'
's%s%s', 'delimiter', ';');
    end

TimeStamp = Split_Line{1};
CallSign   = Split_Line{2};
Event      = Split_Line{4};
ADEP       = Split_Line{35};
ADES       = Split_Line{36};
Model      = Split_Line{37};
% Extract Events
if strcmp(Event, '#DelayHoldInAir') == 1
    DelayHoldInAir_Count = DelayHoldInAir_Count + 1;
    DelayHoldInAir(DelayHoldInAir_Count).CallSign = CallSign;
    DelayHoldInAir(DelayHoldInAir_Count).TimeStamp = TimeStamp;
    DelayHoldInAir(DelayHoldInAir_Count).ADEP = ADEP;
    DelayHoldInAir(DelayHoldInAir_Count).ADES = ADES;
elseif strcmp(Event, '#EODelayHoldInAir') == 1
    EODelayHoldInAir_Count = EODelayHoldInAir_Count + 1;
    EODelayHoldInAir(EODelayHoldInAir_Count).CallSign = CallSign;
    EODelayHoldInAir(EODelayHoldInAir_Count).TimeStamp = TimeStamp;
    EODelayHoldInAir(EODelayHoldInAir_Count).ADEP = ADEP;
    EODelayHoldInAir(EODelayHoldInAir_Count).ADES = ADES;
elseif strcmp(Event, '#HS') == 1
    HS_Count = HS_Count + 1;
    HS(HS_Count).CallSign = CallSign;
    HS(HS_Count).TimeStamp = TimeStamp;
    HS(HS_Count).ADEP = ADEP;
    HS(HS_Count).ADES = ADES;
elseif strcmp(Event, '#EOHS') == 1
    EOHS_Count = EOHS_Count + 1;
    EOHS(EOHS_Count).CallSign = CallSign;
    EOHS(EOHS_Count).TimeStamp = TimeStamp;
    EOHS(EOHS_Count).ADEP = ADEP;
    EOHS(EOHS_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_DepartureQ') == 1
    GRD_DepartureQ_Count = GRD_DepartureQ_Count + 1;
    GRD_DepartureQ(GRD_DepartureQ_Count).CallSign = CallSign;
    GRD_DepartureQ(GRD_DepartureQ_Count).TimeStamp = TimeStamp;
    GRD_DepartureQ(GRD_DepartureQ_Count).ADEP = ADEP;
    GRD_DepartureQ(GRD_DepartureQ_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_EODepartureQ') == 1
    GRD_EODepartureQ_Count = GRD_EODepartureQ_Count + 1;
    GRD_EODepartureQ(GRD_EODepartureQ_Count).CallSign = CallSign;
    GRD_EODepartureQ(GRD_EODepartureQ_Count).TimeStamp = TimeStamp;
    GRD_EODepartureQ(GRD_EODepartureQ_Count).ADEP = ADEP;
    GRD_EODepartureQ(GRD_EODepartureQ_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_DelayRunwayCrossing') == 1
    GRD_DelayRunwayCrossing_Count = GRD_DelayRunwayCrossing_Count + 1;

```

```

GRD_DelayRunwayCrossing(GRD_DelayRunwayCrossing_Count).CallSign =
CallSign;
GRD_DelayRunwayCrossing(GRD_DelayRunwayCrossing_Count).TimeStamp =
TimeStamp;
GRD_DelayRunwayCrossing(GRD_DelayRunwayCrossing_Count).ADEP = ADEP;
GRD_DelayRunwayCrossing(GRD_DelayRunwayCrossing_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_EODelayRunwayCrossing') == 1
GRD_EODelayRunwayCrossing_Count = GRD_EODelayRunwayCrossing_Count + 1;
GRD_EODelayRunwayCrossing(GRD_EODelayRunwayCrossing_Count).CallSign =
CallSign;
GRD_EODelayRunwayCrossing(GRD_EODelayRunwayCrossing_Count).TimeStamp =
TimeStamp;
GRD_EODelayRunwayCrossing(GRD_EODelayRunwayCrossing_Count).ADEP =
ADEP;
GRD_EODelayRunwayCrossing(GRD_EODelayRunwayCrossing_Count).ADES =
ADES;
elseif strcmp(Event, '#GRD_GLDelayCapacity') == 1
GRD_GLDelayCapacity_Count = GRD_GLDelayCapacity_Count + 1;
GRD_GLDelayCapacity(GRD_GLDelayCapacity_Count).CallSign = CallSign;
GRD_GLDelayCapacity(GRD_GLDelayCapacity_Count).TimeStamp = TimeStamp;
GRD_GLDelayCapacity(GRD_GLDelayCapacity_Count).ADEP = ADEP;
GRD_GLDelayCapacity(GRD_GLDelayCapacity_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_EOGLDelayCapacity') == 1
GRD_EOGLDelayCapacity_Count = GRD_EOGLDelayCapacity_Count + 1;
GRD_EOGLDelayCapacity(GRD_EOGLDelayCapacity_Count).CallSign =
CallSign;
GRD_EOGLDelayCapacity(GRD_EOGLDelayCapacity_Count).TimeStamp =
TimeStamp;
GRD_EOGLDelayCapacity(GRD_EOGLDelayCapacity_Count).ADEP = ADEP;
GRD_EOGLDelayCapacity(GRD_EOGLDelayCapacity_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_GLDelaySeparation') == 1
GRD_GLDelaySeparation_Count = GRD_GLDelaySeparation_Count + 1;
GRD_GLDelaySeparation(GRD_GLDelaySeparation_Count).CallSign =
CallSign;
GRD_GLDelaySeparation(GRD_GLDelaySeparation_Count).TimeStamp =
TimeStamp;
GRD_GLDelaySeparation(GRD_GLDelaySeparation_Count).ADEP = ADEP;
GRD_GLDelaySeparation(GRD_GLDelaySeparation_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_EOGLDelaySeparation') == 1
GRD_EOGLDelaySeparation_Count = GRD_EOGLDelaySeparation_Count + 1;
GRD_EOGLDelaySeparation(GRD_EOGLDelaySeparation_Count).CallSign =
CallSign;
GRD_EOGLDelaySeparation(GRD_EOGLDelaySeparation_Count).TimeStamp =
TimeStamp;
GRD_EOGLDelaySeparation(GRD_EOGLDelaySeparation_Count).ADEP = ADEP;
GRD_EOGLDelaySeparation(GRD_EOGLDelaySeparation_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_DEPGATEblocked') == 1
GRD_DEPGATEblocked_Count = GRD_DEPGATEblocked_Count + 1;
GRD_DEPGATEblocked(GRD_DEPGATEblocked_Count).CallSign = CallSign;
GRD_DEPGATEblocked(GRD_DEPGATEblocked_Count).TimeStamp = TimeStamp;
GRD_DEPGATEblocked(GRD_DEPGATEblocked_Count).ADEP = ADEP;
GRD_DEPGATEblocked(GRD_DEPGATEblocked_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_EODEPGATEblocked') == 1
GRD_EODEPGATEblocked_Count = GRD_EODEPGATEblocked_Count + 1;
GRD_EODEPGATEblocked(GRD_EODEPGATEblocked_Count).CallSign = CallSign;
GRD_EODEPGATEblocked(GRD_EODEPGATEblocked_Count).TimeStamp =
TimeStamp;
GRD_EODEPGATEblocked(GRD_EODEPGATEblocked_Count).ADEP = ADEP;

```



```

GRD_EODEPGATEblocked(GRD_EODEPGATEblocked_Count).ADES = ADES;
elseif strcmp(Event, '#GRD_GATEDEP') == 1
GRD_GATEDEP_Count = GRD_GATEDEP_Count + 1;
GRD_GATEDEP(GRD_GATEDEP_Count).CallSign = CallSign;
GRD_GATEDEP(GRD_GATEDEP_Count).TimeStamp = TimeStamp;
GRD_GATEDEP(GRD_GATEDEP_Count).ADEP = ADEP;
GRD_GATEDEP(GRD_GATEDEP_Count).ADES = ADES;
GRD_GATEDEP(GRD_GATEDEP_Count).Model = Model;
elseif strcmp(Event, '#RWY_TOUCHDOWN') == 1
RWY_TOUCHDOWN_Count = RWY_TOUCHDOWN_Count + 1;
RWY_TOUCHDOWN(RWY_TOUCHDOWN_Count).CallSign = CallSign;
RWY_TOUCHDOWN(RWY_TOUCHDOWN_Count).TimeStamp = TimeStamp;
RWY_TOUCHDOWN(RWY_TOUCHDOWN_Count).ADEP = ADEP;
RWY_TOUCHDOWN(RWY_TOUCHDOWN_Count).ADES = ADES;
RWY_TOUCHDOWN(RWY_TOUCHDOWN_Count).Model = Model;
elseif strcmp(Event, '#RWY_NODEA') == 1
RWY_NODEA_Count = RWY_NODEA_Count + 1;
RWY_NODEA(RWY_NODEA_Count).CallSign = CallSign;
RWY_NODEA(RWY_NODEA_Count).TimeStamp = TimeStamp;
RWY_NODEA(RWY_NODEA_Count).ADEP = ADEP;
RWY_NODEA(RWY_NODEA_Count).ADES = ADES;
RWY_NODEA(RWY_NODEA_Count).Model = Model;
elseif strcmp(Event, '#RWY_NODEB') == 1
RWY_NODEB_Count = RWY_NODEB_Count + 1;
RWY_NODEB(RWY_NODEB_Count).CallSign = CallSign;
RWY_NODEB(RWY_NODEB_Count).TimeStamp = TimeStamp;
RWY_NODEB(RWY_NODEB_Count).ADEP = ADEP;
RWY_NODEB(RWY_NODEB_Count).ADES = ADES;
RWY_NODEB(RWY_NODEB_Count).Model = Model;

end

end

fclose(fid);

disp(['Total Records: ', num2str(count)]);
disp(['DelayHoldInAir Records: ', num2str(DelayHoldInAir_Count)]);
disp(['EODelayHoldInAir Records: ',
num2str(EODelayHoldInAir_Count)]);
disp(['HS Records: ', num2str(HS_Count)]);
disp(['EOHS Records: ', num2str(EOHS_Count)]);
disp(['GRD_DepartureQ Records: ', num2str(GRD_DepartureQ_Count)]);
disp(['GRD_EODepartureQ Records: ',
num2str(GRD_EODepartureQ_Count)]);
disp(['GRD_DelayRunwayCrossing Records: ',
num2str(GRD_DelayRunwayCrossing_Count)]);
disp(['GRD_EODelayRunwayCrossing Records: ',
num2str(GRD_EODelayRunwayCrossing_Count)]);
disp(['GRD_GLDelayCapacity Records: ',
num2str(GRD_GLDelayCapacity_Count)]);
disp(['GRD_EOGLDelayCapacity Records: ',
num2str(GRD_EOGLDelayCapacity_Count)]);
disp(['GRD_GLDelaySeparation Records: ',
num2str(GRD_GLDelaySeparation_Count)]);
disp(['GRD_EOGLDelaySeparation Records: ',
num2str(GRD_EOGLDelaySeparation_Count)]);
disp(['GRD_DEPGATEblocked Records: ',

```

```

num2str(GRD_DEPGATEblocked_Count)];
disp(['GRD_EODEPGATEblocked Records: ',
num2str(GRD_EODEPGATEblocked_Count)];
disp(['GRD_GATEDEP Records: ', num2str(GRD_GATEDEP_Count)];
disp(['RWY_TOUCHDOWN Records: ', num2str(RWY_TOUCHDOWN_Count)];
disp(['RWY_NODEA Records: ', num2str(RWY_NODEA_Count)];
disp(['RWY_NODEB Records: ', num2str(RWY_NODEB_Count)];

% -----
% DelayHoldInAir
% -----
disp('Saving DelayHoldInAir...');
Number_Of_Records_Start = length(DelayHoldInAir);

fid = fopen('C:\Documents and Settings\genea\Desktop\DelayHoldInAir.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = DelayHoldInAir(i).CallSign;
    Number_Of_Records_End = length(EODelayHoldInAir);
    ADEP_Start= DelayHoldInAir(i).ADEP;
    ADES_Start= DelayHoldInAir(i).ADES;
    End_Delay_Found = 0;

    for j = 1:Number_Of_Records_End

        CallSign_End = EODelayHoldInAir(j).CallSign;
        ADEP_End = EODelayHoldInAir(j).ADEP;
        ADES_End = EODelayHoldInAir(j).ADES;
        if strcmp(CallSign_Start, CallSign_End) == 1
            Time_Start = datenum(DelayHoldInAir(i).TimeStamp); % Days
            Time_End = datenum(EODelayHoldInAir(j).TimeStamp); % Days
            Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
            fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(DelayHoldInAir(i).TimeStamp), char(EODelayHoldInAir(j).TimeStamp),
Delay_Secs, char(ADEP_Start), char(ADES_Start));
            EODelayHoldInAir(j) = []; % Delete Record to avoid duplications
            End_Delay_Found = 1;
            break;
        end

    end % for j = 1:Number_Of_Records_End

    % No end delay found - assume end delay is end of simulation
    if End_Delay_Found == 0
        Time_Start = datenum(DelayHoldInAir(i).TimeStamp); % Days
        Time_End = datenum(Simulation_Time_End); % Days
        Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(DelayHoldInAir(i).TimeStamp), char(Simulation_Time_End), Delay_Secs,
char(ADEP_Start), char(ADES_Start));
    end

end % for i = 1:Number_Of_Records_Start

fclose(fid);

```

```

% -----
% HS
% -----
disp('Saving HS...');
Number_Of_Records_Start = length(HS);

fid = fopen('C:\Documents and Settings\genea\Desktop\HS.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = HS(i).CallSign;
    Number_Of_Records_End = length(EOHS);
    ADEP_Start = HS(i).ADEP;
    ADES_Start = HS(i).ADES;
    End_Delay_Found = 0;

    for j = 1:Number_Of_Records_End

        CallSign_End = EOHS(j).CallSign;
        ADEP_End = EOHS(j).ADEP;
        ADES_End = EOHS(j).ADES;
        if strcmp(CallSign_Start, CallSign_End) == 1
            Time_Start = datenum(HS(i).TimeStamp); % Days
            Time_End = datenum(EOHS(j).TimeStamp); % Days
            Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
            fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(HS(i).TimeStamp), char(EOHS(j).TimeStamp), Delay_Secs, char(ADEP_Start),
char(ADES_Start));
            EOHS(j) = []; % Delete Record to avoid duplications
            End_Delay_Found = 1;
            break;
        end

    end % for j = 1:Number_Of_Records_End

    % No end delay found - assume end delay is end of simulation
    if End_Delay_Found == 0
        Time_Start = datenum(HS(i).TimeStamp); % Days
        Time_End = datenum(Simulation_Time_End); % Days
        Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(HS(i).TimeStamp), char(Simulation_Time_End), Delay_Secs,
char(ADEP_Start), char(ADES_Start));
        end

    end % for i = 1:Number_Of_Records_Start

fclose(fid);

% -----
% GRD_DepartureQ
% -----
disp('Saving GRD_DepartureQ...');
Number_Of_Records_Start = length(GRD_DepartureQ);

fid = fopen('C:\Documents and Settings\genea\Desktop\GRD_DepartureQ.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

```

```

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_DepartureQ(i).CallSign;
    Number_Of_Records_End = length(GRD_EODepartureQ);
    ADEP_Start = GRD_DepartureQ(i).ADEP;
    ADES_Start = GRD_DepartureQ(i).ADES;
    End_Delay_Found = 0;

    for j = 1:Number_Of_Records_End

        CallSign_End = GRD_EODepartureQ(j).CallSign;
        ADEP_End = GRD_EODepartureQ(j).ADEP;
        ADES_End = GRD_EODepartureQ(j).ADES;

        if strcmp(CallSign_Start, CallSign_End) == 1
            Time_Start = datenum(GRD_DepartureQ(i).TimeStamp); % Days
            Time_End = datenum(GRD_EODepartureQ(j).TimeStamp); % Days
            Delay_Seconds = round((Time_End - Time_Start) * 24 * 3600); % seconds
            fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(GRD_DepartureQ(i).TimeStamp), char(GRD_EODepartureQ(j).TimeStamp),
Delay_Seconds, char(ADEP_Start), char(ADES_Start));
            GRD_EODepartureQ(j) = []; % Delete Record to avoid duplications
            End_Delay_Found = 1;
            break;
        end

    end % for j = 1:Number_Of_Records_End

    % No end delay found - assume end delay is end of simulation
    if End_Delay_Found == 0
        Time_Start = datenum(GRD_DepartureQ(i).TimeStamp); % Days
        Time_End = datenum(Simulation_Time_End); % Days
        Delay_Seconds = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(GRD_DepartureQ(i).TimeStamp), char(Simulation_Time_End), Delay_Seconds,
char(ADEP_Start), char(ADES_Start));
    end

end % for i = 1:Number_Of_Records_Start

fclose(fid);

% -----
% GRD_DelayRunwayCrossing
% -----
disp('Saving GRD_DelayRunwayCrossing...');
Number_Of_Records_Start = length(GRD_DelayRunwayCrossing);

fid = fopen('C:\Documents and
Settings\genea\Desktop\GRD_DelayRunwayCrossing.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Seconds,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_DelayRunwayCrossing(i).CallSign;
    Number_Of_Records_End = length(GRD_EODelayRunwayCrossing);
    ADEP_Start = GRD_DelayRunwayCrossing(i).ADEP;

```

```

ADES_Start = GRD_DelayRunwayCrossing(i).ADES;
End_Delay_Found = 0;

for j = 1:Number_Of_Records_End

    CallSign_End = GRD_EODelayRunwayCrossing(j).CallSign;
    ADEP_End = GRD_EODelayRunwayCrossing(j).ADEP;
    ADES_End = GRD_EODelayRunwayCrossing(j).ADES;

    if strcmp(CallSign_Start, CallSign_End) == 1
        Time_Start = datenum(GRD_DelayRunwayCrossing(i).TimeStamp); % Days
        Time_End = datenum(GRD_EODelayRunwayCrossing(j).TimeStamp); %
Days
        Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%.0f,%s,%s\n', char(CallSign_Start),
char(GRD_DelayRunwayCrossing(i).TimeStamp),
char(GRD_EODelayRunwayCrossing(j).TimeStamp), Delay_Secs, char(ADEP_Start),
char(ADES_Start));
        GRD_EODelayRunwayCrossing(j) = []; % Delete Record to avoid
duplications
        End_Delay_Found = 1;
        break;
    end

end % for j = 1:Number_Of_Records_End

% No end delay found - assume end delay is end of simulation
if End_Delay_Found == 0
    Time_Start = datenum(GRD_DelayRunwayCrossing(i).TimeStamp); % Days
    Time_End = datenum(Simulation_Time_End); % Days
    Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
    fprintf(fid, '%s,%s,%s,%.0f,%s,%s\n', char(CallSign_Start),
char(GRD_DelayRunwayCrossing(i).TimeStamp), char(Simulation_Time_End),
Delay_Secs, char(ADEP_Start), char(ADES_Start));
end

end % for i = 1:Number_Of_Records_Start

fclose(fid);

% -----
% GRD_GLDelayCapacity
% -----
disp('Saving GRD_GLDelayCapacity...');
Number_Of_Records_Start = length(GRD_GLDelayCapacity);

fid = fopen('C:\Documents and
Settings\genea\Desktop\GRD_GLDelayCapacity.txt', 'w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_GLDelayCapacity(i).CallSign;
    Number_Of_Records_End = length(GRD_EOGLDelayCapacity);
    ADEP_Start = GRD_GLDelayCapacity(i).ADEP;
    ADES_Start = GRD_GLDelayCapacity(i).ADES;
    End_Delay_Found = 0;

```

```

for j = 1:Number_Of_Records_End

    CallSign_End = GRD_EOGLDelayCapacity(j).CallSign;
    ADEP_End = GRD_EOGLDelayCapacity(j).ADEP;
    ADES_End = GRD_EOGLDelayCapacity(j).ADES;

    if strcmp(CallSign_Start, CallSign_End) == 1
        Time_Start = datenum(GRD_GLDelayCapacity(i).TimeStamp); % Days
        Time_End = datenum(GRD_EOGLDelayCapacity(j).TimeStamp); % Days
        Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(GRD_GLDelayCapacity(i).TimeStamp),
char(GRD_EOGLDelayCapacity(j).TimeStamp), Delay_Secs, char(ADEP_Start),
char(ADES_Start));
        GRD_EOGLDelayCapacity(j) = []; % Delete Record to avoid
duplications
        End_Delay_Found = 1;
        break;
    end

end % for j = 1:Number_Of_Records_End

% No end delay found - assume end delay is end of simulation
if End_Delay_Found == 0
    Time_Start = datenum(GRD_GLDelayCapacity(i).TimeStamp); % Days
    Time_End = datenum(Simulation_Time_End); % Days
    Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
    fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(GRD_GLDelayCapacity(i).TimeStamp), char(Simulation_Time_End), Delay_Secs,
char(ADEP_Start), char(ADES_Start));
end

end % for i = 1:Number_Of_Records_Start

fclose(fid);

% -----
% GRD_GLDelaySeparation
% -----
disp('Saving GRD_GLDelaySeparation...');
Number_Of_Records_Start = length(GRD_GLDelaySeparation);

fid = fopen('C:\Documents and
Settings\genea\Desktop\GRD_GLDelaySeparation.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_GLDelaySeparation(i).CallSign;
    Number_Of_Records_End = length(GRD_EOGLDelaySeparation);
    ADEP_Start = GRD_GLDelaySeparation(i).ADEP;
    ADES_Start = GRD_GLDelaySeparation(i).ADES;
    End_Delay_Found = 0;

    for j = 1:Number_Of_Records_End

        CallSign_End = GRD_EOGLDelaySeparation(j).CallSign;
        ADEP_End = GRD_EOGLDelaySeparation(j).ADEP;

```

```

        ADES_End = GRD_EOGLDelaySeparation(j).ADES;

        if strcmp(CallSign_Start, CallSign_End) == 1
            Time_Start = datenum(GRD_GLDelaySeparation(i).TimeStamp); % Days
            Time_End = datenum(GRD_EOGLDelaySeparation(j).TimeStamp); % Days
            Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
            fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
                char(GRD_GLDelaySeparation(i).TimeStamp),
                char(GRD_EOGLDelaySeparation(j).TimeStamp), Delay_Secs, char(ADEP_Start),
                char(ADES_Start));
            GRD_EOGLDelaySeparation(j) = []; % Delete Record to avoid
duplications
            End_Delay_Found = 1;
            break;
        end
    end % for j = 1:Number_Of_Records_End

    % No end delay found - assume end delay is end of simulation
    if End_Delay_Found == 0
        Time_Start = datenum(GRD_GLDelaySeparation(i).TimeStamp); % Days
        Time_End = datenum(Simulation_Time_End); % Days
        Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
        fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
            char(GRD_GLDelaySeparation(i).TimeStamp), char(Simulation_Time_End),
            Delay_Secs, char(ADEP_Start), char(ADES_Start));
        end
    end % for i = 1:Number_Of_Records_Start

fclose(fid);

% -----
% GRD_DEPGATEblocked
% -----
disp('Saving GRD_DEPGATEblocked...');
Number_Of_Records_Start = length(GRD_DEPGATEblocked);

fid = fopen('C:\Documents and
Settings\genea\Desktop\GRD_DEPGATEblocked.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,Time_End,Delay_Secs,ADEP,ADES');

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_DEPGATEblocked(i).CallSign;
    Number_Of_Records_End = length(GRD_EODEPGATEblocked);
    ADEP_Start= GRD_DEPGATEblocked(i).ADEP;
    ADES_Start= GRD_DEPGATEblocked(i).ADES;
    End_Delay_Found = 0;

    for j = 1:Number_Of_Records_End

        CallSign_End = GRD_EODEPGATEblocked(j).CallSign;
        ADEP_End = GRD_EODEPGATEblocked(j).ADEP;
        ADES_End = GRD_EODEPGATEblocked(j).ADES;
        if strcmp(CallSign_Start, CallSign_End) == 1 % Found End Delay
            Time_Start = datenum(GRD_DEPGATEblocked(i).TimeStamp); % Days
            Time_End = datenum(GRD_EODEPGATEblocked(j).TimeStamp); % Days
            Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
            fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),

```

```

char(GRD_DEPGATEblocked(i).TimeStamp),
char(GRD_EODEPGATEblocked(j).TimeStamp), Delay_Secs, char(ADEP_Start),
char(ADES_Start));
    GRD_EODEPGATEblocked(j) = []; % Delete Record to avoid
duplications
    End_Delay_Found = 1;
    break;
end

end % for j = 1:Number_Of_Records_End

% No end delay found - assume end delay is end of simulation
if End_Delay_Found == 0
    Time_Start = datenum(GRD_DEPGATEblocked(i).TimeStamp); % Days
    Time_End = datenum(Simulation_Time_End); % Days
    Delay_Secs = round((Time_End - Time_Start) * 24 * 3600); % seconds
    fprintf(fid, '%s,%s,%s,%0f,%s,%s\n', char(CallSign_Start),
char(GRD_DEPGATEblocked(i).TimeStamp), char(Simulation_Time_End), Delay_Secs,
char(ADEP_Start), char(ADES_Start));
end

end % for i = 1:Number_Of_Records_Start

fclose(fid);

%
% -----
% GRD_GATEDEP
% -----
disp('Saving GRD_GATEDEP...');
Number_Of_Records_Start = length(GRD_GATEDEP);

fid = fopen('C:\Documents and Settings\genea\Desktop\GRD_GATEDEP.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,ADEP,ADES,Model');

for i = 1:Number_Of_Records_Start

    CallSign_Start = GRD_GATEDEP(i).CallSign;
    ADEP_Start= GRD_GATEDEP(i).ADEP;
    ADES_Start= GRD_GATEDEP(i).ADES;
    Model_Start= GRD_GATEDEP(i).Model;
    Time_Start = datenum(GRD_GATEDEP(i).TimeStamp); % Day
    fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(GRD_GATEDEP(i).TimeStamp), char(ADEP_Start), char(ADES_Start),
char(Model_Start));

end % for i = 1:Number_Of_Records_Start
fclose(fid);

% -----
% RWY_TOUCHDOWN
% -----
disp('Saving RWY_TOUCHDOWN...');
Number_Of_Records_Start = length(RWY_TOUCHDOWN);

fid = fopen('C:\Documents and Settings\genea\Desktop\RWY_TOUCHDOWN.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,ADEP,ADES,Model');

```



```

for i = 1:Number_Of_Records_Start

    CallSign_Start = RWY_TOUCHDOWN(i).CallSign;
    ADEP_Start= RWY_TOUCHDOWN(i).ADEP;
    ADES_Start= RWY_TOUCHDOWN(i).ADES;
    Model_Start= RWY_TOUCHDOWN(i).Model;
    Time_Start = datenum(RWY_TOUCHDOWN(i).TimeStamp); % Day
    fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(RWY_TOUCHDOWN(i).TimeStamp), char(ADEP_Start), char(ADES_Start),
char(Model_Start));

end % for i = 1:Number_Of_Records_Start
fclose(fid);

% -----
% RWY_NODEA
% -----
disp('Saving RWY_NODEA...');
Number_Of_Records_Start = length(RWY_NODEA);

fid = fopen('C:\Documents and Settings\genea\Desktop\RWY_NODEA.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,ADEP,ADES,Model');

for i = 1:Number_Of_Records_Start

    CallSign_Start = RWY_NODEA(i).CallSign;
    ADEP_Start= RWY_NODEA(i).ADEP;
    ADES_Start= RWY_NODEA(i).ADES;
    Model_Start= RWY_NODEA(i).Model;

    Time_Start = datenum(RWY_NODEA(i).TimeStamp); % Day
    fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(RWY_NODEA(i).TimeStamp), char(ADEP_Start), char(ADES_Start),
char(Model_Start));

end % for i = 1:Number_Of_Records_Start
fclose(fid);

% -----
% RWY_NODEB
% -----
disp('Saving RWY_NODEB...');
Number_Of_Records_Start = length(RWY_NODEB);
fid = fopen('C:\Documents and Settings\genea\Desktop\RWY_NODEB.txt','w');
fprintf(fid, '%s\n', 'CallSign,Time_Start,ADEP,ADES,Model');

for i = 1:Number_Of_Records_Start
    CallSign_Start = RWY_NODEB(i).CallSign;
    ADEP_Start= RWY_NODEB(i).ADEP;
    ADES_Start= RWY_NODEB(i).ADES;
    Model_Start= RWY_NODEB(i).Model;
    Time_Start = datenum(RWY_NODEB(i).TimeStamp); % Day
    fprintf(fid, '%s,%s,%s,%s,%s\n', char(CallSign_Start),
char(RWY_NODEB(i).TimeStamp), char(ADEP_Start), char(ADES_Start),
char(Model_Start));
end % for i = 1:Number_Of_Records_Start
fclose(fid);
disp('Finsihed!');

```

# APPENDIX C 3 MATLAB FILE TO ANALYZE THE DELAY FILES

```

clear all
clc

Files = {'DelayHoldInAir'; 'GRD_DelayRunwayCrossing'; 'GRD_DepartureQ';
'GRD_GLDelayCapacity'; 'GRD_GLDelaySeparation'; 'HS'; 'GRD_DEPGATEblocked'};
Number_Of_Files = length(Files);

for File_Number = 1:Number_Of_Files

    fid = fopen(['C:\Documents and Settings\genea\Desktop\'],
Files{File_Number}, '.txt', 'r');

    field_names = fgetl(fid); % Extract Field Names

    Output = textscan(fid, '%s%s%s%f%s%s', 'delimiter', ',');

    CallSigns = Output{1,1};
    StartTime = (datenum(Output{1,2}) - datenum('00:00:00')) * 24; % Convert
to decimal hrs
    EndTime    = (datenum(Output{1,3}) - datenum('00:00:00')) * 24; % Convert
to decimal hrs
    Delay_Sec = Output{1,4};
    ADEP      = Output{1,5};
    ADES      = Output{1,6};
    clear Output;

    Bins = [0:1:48];
    Number_of_Bins = length(Bins) - 1;

    for n = 1:Number_of_Bins;

        Indices      = find(StartTime > Bins(n) & StartTime <= Bins(n+1) &
strcmp(ADEP, 'EWR') == 1 );
        Delay_Sum_ADEP(n,File_Number) = sum(Delay_Sec(Indices)); % Sum of
delays in bin in seconds
    end % for n = 1:Number_of_Bins;
    for n = 1:Number_of_Bins;
        Indices      = find(StartTime > Bins(n) & StartTime <= Bins(n+1) &
strcmp(ADES, 'EWR') == 1 );
        Delay_Sum_ADES(n,File_Number) = sum(Delay_Sec(Indices)); % Sum of
delays in bin in seconds

    end % for n = 1:Number_of_Bins;
    Delay_Sum = Delay_Sum_ADEP + Delay_Sum_ADES;
    fclose(fid);

end % for File_Number = 1:Number_Of_Files
% Split Days
% Rearrange Bins to convert from UTC to EST
Delay_Sum_Day1 = Delay_Sum(1:24,:);
Delay_Sum_Day1 = [Delay_Sum_Day1(6:end,:); Delay_Sum_Day1(1:5,:)];
Delay_Sum_Day2 = Delay_Sum(25:48,:);
Delay_Sum_Day2 = [Delay_Sum_Day2(6:end,:); Delay_Sum_Day2(1:5,:)];

```

## APPENDIX C 4 MATLAB FILE TO COUNT THE HOURLY OPERATIONS

```
% -----  
% Function to count the number of operations per hour at any airport  
% -----  
clear all  
clc  
%  
%  
Files = {'GRD_GATEDEP'; 'RWY_TOUCHDOWN'};  
Number_Of_Files = length(Files);  
  
for File_Number = 1:Number_Of_Files  
  
    fid = fopen(['C:\Documents and Settings\genea\Desktop\  
Files{File_Number}', '.txt'], 'r');  
  
    field_names = fgetl(fid); % Extract Field Names  
  
    Output = textscan(fid, '%s%s%s%s', 'delimiter', ',');  
  
    CallSigns = Output{1,1};  
    StartTime = (datenum(Output{1,2}) - datenum('00:00:00')) * 24; % Convert  
to decimal hrs  
    ADEP      = Output{1,3};  
    ADES      = Output{1,4};  
  
    clear Output;  
  
    Bins = [0:1:48];  
    Number_of_Bins = length(Bins) - 1;  
  
    for n = 1:Number_of_Bins;  
        Indices = find(StartTime > Bins(n) & StartTime <= Bins(n+1) &  
strcmp(ADES, 'LGA') == 1 );  
        Operations_Sum(n, File_Number) = length(Indices); % Sum of delays in  
bin in seconds  
    end % for n = 1:Number_of_Bins;  
  
    fclose(fid);  
  
end % for File_Number = 1:Number_Of_Files  
Total_Operations = sum(Operations_Sum, 2);  
% Split Days  
% Rearrange Bins to convert from UTC to EST  
Total_Operations_Day1 = Total_Operations(1:24, :);  
Total_Operations_Day1 = [Total_Operations_Day1(6:end, :);  
Total_Operations_Day1(1:5, :)];  
Total_Operations_Day2 = Total_Operations(25:48, :);  
Total_Operations_Day2 = [Total_Operations_Day2(6:end, :);  
Total_Operations_Day2(1:5, :)];
```