METHODOLOGY FOR CONTRAIL INVENTORY ANALYSIS

by

Denis Avila
A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Systems Engineering and Operations Research

Committee:

________________________________ Dr. Lance Sherry, Dissertation Director

________________________________ Dr. George Donohue, Committee Member

________________________________ Dr. John F. Shortle, Committee Member

________________________________ Dr. Elise Miller-Hooks, Committee Member

________________________________ Dr. Hadi El-Amine, Committee Member

________________________________ Dr. John F. Shortle,Chair Systems Engineering and Operations Research

________________________________ Dr. Ariela Sofer, Associate Dean

________________________________ Dr. Kenneth S. Ball, Dean, Volgenau School of Engineering

Date:___________________________ Spring Semester 2019
George Mason University
Fairfax, VA

Methodology for Contrail Inventory Analysis

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

by

Denis Avila
Master of Science
George Mason University 2010

Director: Lance Sherry, Associate Professor
Department of System Engineering and Operations Research

Spring Semester 2019
George Mason University
Fairfax, VA

DEDICATION

To my wife Gabriela and my children Kendra, Alexander and Hannah.

ACKNOWLEDGEMENTS

It is with genuine appreciation that I would like to thank my thesis advisor Dr. Lance Sherry for his guidance and encouragement throughout this process. His unbending support for my research and expert knowledge of air transportation made this work a reality. My appreciation extends to my committee members Dr. George Donohue, Dr. John F. Shortle, Dr. Elise Miller-Hooks and Dr. Hadi El-Amine whose questions and comments helped shape my dissertation and made it stronger. I would also like to thank Dr Alexander Levis, whose questions and comments started my path down this road. Finally, I owe the gratest gratitude to my family. To my wife Gabriela, who has always supported me, and to my children Kendra, Alexander and Hannah who have been there with a smile.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

ABSTRACT

METHODOLOGY FOR CONTRAIL INVENTORY ANALYSIS

Denis Avila, Ph D.

George Mason University, 2019

Dissertation Director: Dr Lance Sherry

Condensation trails (aka "contrails") are high thin clouds that occur when hot exhaust gases from jet engines mix with cold, humid air. These anthropogenic clouds result in a net warming effect by blocking approximately 33% of outgoing longwave radiation emitted by the Earth, and allowing 77% of the incoming shortwave radiation from the Sun to be absorbed by the Earth and it's atmosphere. Even though contrails generate only 1% of total anthropogenic radiative forcing, they have an immediate effect on global warming. In this way, managing contrails can yield immediate global warming benefits today that can be used to buy-time for long-term CO_2 mitigations to take effect. To support "contrail management," contrails from flights will need to be inventoried. Previous models required pre-processed atmospheric data, were geographically limited, and did not take into account Sun Zenith Angle, optical depth, contrail width or ice crystal size. These models were not able to analyze diurnal, seasonal, or geographic effects for a large airspace (e.g. CONUS), and could not be used in an operational context for alternate flight path evaluations.

This paper describes a commercially scalable method for inventory of contrails in a national airspace system using publicly available weather and flight surveillance track data, and models of contrail formation and net radiative forcing. The method provides the means to generate Contrail Inventories as well as perform real-time alternative flight-path analysis and overcomes the limitations described above. The method is demonstrated with a case study for the U.S. National Airspace System across 365 historic weather days. The analysis yielded a daily average of 57.4K nautical miles of contrails with an estimated daily net radiative forcing of +7.08 mW/m2. Less than 25% of the flights generated contrails on a given day, and Summer months had three times the warming effect of the Winter months. Increasing Cruise Flight Levels to avoid generating a contrail by 2000' up to a maximum of 4000', reduced the average daily flights with contrails by 14.8%, reducing NRF by an average of 91%, with an average decrease in fuel-burn of 0.64%. The implications of these results and the limitation of this method are discussed.

1. INTRODUCTION

Air travel is a common part of our lives, with the world now more connected than ever the number of passengers continues to grow. The International Civil Aviation Organization (ICAO) estimates the airline industry today is formed by 1,400 commercial airlines and 4,130 airports. This industry served almost 3.5 billion passengers in 2015 through its 34 million departures. 9.5 million of those departures took place in the US alone. Along with the benefits of the forecast growth in the number of flights there is the need to manage a larger volume of engine exhaust placed directly in the Troposphere. Under certain circumstances the engine exhaust will generate condensation trails; these “condensation trails” are, long, thin, anthropogenic (i.e. human-made) clouds which trap outgoing longwave radiation from the Earth resulting in global warming.
Contrails were first noticed in the 1920’s when the first high-altitude flights began. They became of importance during WWII when bombers could be sighted from miles away identified by their contrails. From that time and until recently little attention has been paid to contrails. In 1953 H. Appleman (1953) established the conditions of relative humidity and temperature in which jet engine emissions will form contrails. In general, contrails form as a result of the hot humid exhaust from jet engines mixing with the cold low pressure atmosphere. The water vapor condenses and freezes on particles left by the engine’s exhaust creating an artificial cloud.

In 1999 the Intergovernmental Panel on Climate Change (IPCC) estimated that contrails covered 0.1% of the Earth’s surface and projected a growth of 5% per year until 2015 (Penner 1999, IPCC Special Report). This paper provides a means to generate an inventory of contrails and estimate the RF they induce. The paper will adhere to the IPCC's definition of radiative forcing which evaluates the anthropogenic forcing as the changes influenced by humans after the industrial era. The Radiative Forcing (RF) measurements refer to the period 1750 – present, unless otherwise noted.

In 2005, the IPCC estimated the total Industrial-era Anthropogenic Forcing (ERF) at 1.6 W/m^2 with a range of uncertainty from 0.6 to 2.4. (Penner J - IPPC AR4) More recently the IPPC report in 2011 estimated the total ERF at 2.3 W/m^2, with a range of uncertainty from 1.1 to 3.3 W/m^2 (Myhre, G – IPPC AR5). From that total Radiative Forcing; the total RF attributed directly to CO_2 is 1.82 W/m^2, while Contrails are estimated at 0.05 W/m^2.

The effect is created because as clouds, contrails present multiple interactions with climate, reflecting sunlight during the day and reflecting heat back to earth during the night. Typically, this will produce a net warming effect; however it is shown that clouds can reduce the magnitude of RF due to Green House Gases (GHG) by about 25% (Forster et al., 2005; Worden et al., 2011; Zhang et al., 2011) .

Contrails are aircraft induced *cirrus clouds* made of ice particles. Their shape and duration will depend on existing atmospheric conditions. Under the right conditions they "persist" for hours or days. These high and thin clouds are highly transparent to shortwave radiation, presenting a small albedo force, allowing most of the incoming energy to reach the surface. Although they do absorb a portion of the outgoing longwave radiation, a fraction is sent back to the surface adding to the shortwave energy (Penner 1999) . The overall effect is therefore to enhance atmospheric greenhouse warming. (Haywood et al., 2009).

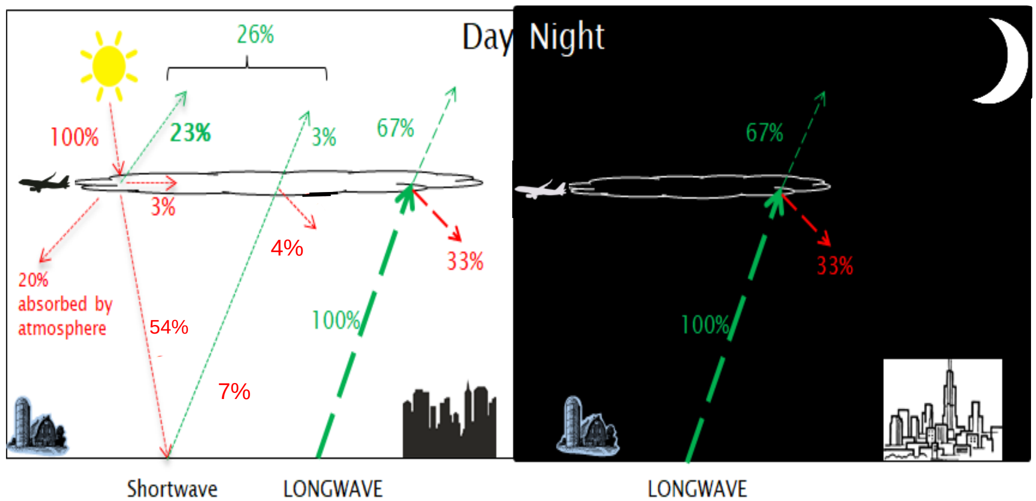


Figure 1: Effect of contrails from incoming shortwave and outgoing longwave radiation

Previous research calculated the contrail formation with a specific scope. For example, Lee et.al. (2009) used closed form model based on fuel-burn. Campbell et.al. (2008), Kaiser (2012), Soler & Hansen (2014), Yin et.al (2018) performed analysis for one route, Gao & Hansman (2013) included routes for 12 city pairs and Sridhar & Chen (2013) analyzed 278 flights, however only Soler & Hansen's (2014) model takes into account contrail persistence. These analyses used pre-processed or simulated atmospheric conditions ranging from one day to a month. The RF models used were simplified estimates and did not take into account Sun Zenith Angle, optical depth, contrail width or ice crystal size. Due to these limitations, previous models could not be used to analyze diurnal, seasonal, or geographic effects for a large airspace (e.g. CONUS).

For this reason the results to these models cannot be applied to an entire airspace. They do not provide enough information to decide if the treatment is effective for large variety of flights or which flights should be modified.

This paper describes a commercially scalable method to process real-time, publicly available track and weather data to estimate the contrail induced RF. At the same time it provides a methodology to perform automated reporting on archived weather and track data across the whole National Airspace for the Contiguous United Stated (CONUS).

The method described in this paper uses efficient data structures and algorithms to process: (1) basic National Oceanic & Atmospheric Administration (NOAA) Rapid Refresh (RAP) atmospheric data for the whole Contiguous United States (CONUS) airspace over 24 hours, 365 days, (2) processes the flight tracks for a full day of scheduled flight operations (e.g. 30,000), (3) implements a contrail formation model (Paoli & Shariff, 2016; Naiman et.al., 2011), and (4) implements a Net Radiative Forcing model that includes contrail formation, persistence, Sun Zenith Angle, optical depth, contrail width or ice crystal size (Schumann et.al., 1996, 2012, 2013; Burkhardt, 2011). The method, models and algorithms developed in this model allow analysis of diurnal, seasonal, and geographic effects for a large airspace (e.g. CONUS). These effects turn out to be important of contrail mitigation strategies.

1.1 BACKGROUND

1.1.1 SOLAR RADIATION

Solar radiation is out natural source of energy from the sun and is delivered to the Earth as electromagnetic radiation. According to the World Meteorological Organization our planet receives approximately 1360 W/m² of insolation. While the majority of our solar radiation is in the form of short wave radiation we receive a wide frequencies ranging from Shortwaves, as short

100nm to long waves as long as 1mm (1,000,000 nm). A portion of the long-wave infrared is absorbed by gasses and particles within the upper atmosphere. Both the atmosphere and clouds behave differently to different wavelengths. As the solar radiation travels through the atmosphere, some of it is absorbed and scattered (25%) by air molecules, water vapor, particles, aerosols and clouds. Approximately 20% of the radiation is reflected back out into space (a large cloud cover will reflect larger portions of the radiation). The remaining radiation arrives on the Earth's surface. Once the radiation arrives at the surface, some of it is reflected back into the sky. The reflection depends on the actual surface - fresh snow can reflect up to 95%, desert sands - 35-45%, grasslands - 15-25% and dense forest vegetation 5-10%. As the surface heats it radiates back to the atmosphere at a higher wavelength.

Scattering of the radiation drives the daylighting and makes the sky look bright. If the radiation were allowed to travel uninterrupted through the atmosphere, the day-sky would look just as the night-sky. The scattering is triggered by particles of approximately 0.5 microns in size. As radiation with longer wavelengths simply ignores these particles, higher frequency (shorter wavelength) radiation tends to be scattered more. This is what makes the sky appear blue - as lower frequency red and yellow light pass almost directly through whilst blue light is bounced about all over the place.

When the electromagnetic waves collide with the atmosphere and clouds they can be either transmitted, reflected, absorbed, refracted, polarized, diffracted, or scattered depending on the composition of the object and the wavelength. Shorter wavelengths react as if clouds were almost invisible while longer wavelengths reflect off the clouds. For the purpose of this analysis we review select properties only.

REFLECTION: Reflection occurs when incoming light hits an object and bounces off. Reflection occurs when irradiated hits very smooth surfaces, such as mirrors or sheets of metal. The wavelengths reflected off an object will give it it's "color".

ABSORPTION: Absorption occurs when photons from the incident light hit atoms and molecules causing them to vibrate. As the object's molecules are hit, they move and vibrate, generating heat. This heat is then emitted from the object as thermal energy.

DIFFRACTION: Diffraction is most common when a light wave hits an object similar in size to its own wavelength causing it to bend and spread around the obstacle..

SCATTER: Scattering occurs when solar radiation hits objects and bounces off in variety of directions. The amount of scattering that will be driven by the wavelength of the radiation and the size and structure of the object.

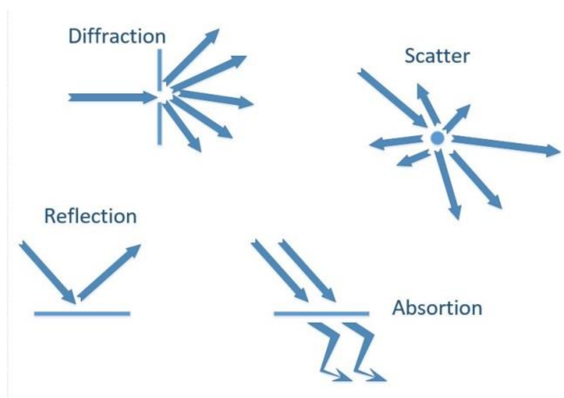


Figure 2: Light properties

This last effect is the reason the sky appears blue. The shorter wavelengths, such as blue and violet are scattered by nitrogen and oxygen as it passes through the atmosphere. Longer wavelengths of light—red and yellow—transmit through the atmosphere. Scattering of radiation of the shorter wavelengths causes the sky to illuminates with the light from the blue and violet end of the visible spectrum. While violet is scattered more than blue our sky appears blue to our eyes because we more sensitive to blue light.

The color of a cloud as seen from the below provides insight as to the cloud’s composition and density. Dense deep tropospheric clouds present a high reflectance (70% to 95%) throughout the visible spectrum. When the water particles are densely packed, sunlight cannot penetrate into the cloud before it is reflected out, giving clouds their characteristic white color, especially when viewed from the top. Water drops in clouds tend to scatter light efficiently, in this way the amount of solar radiation decreases as in penetrates the cloud. As a result, the cloud base can vary from a very light to very dark grey depending on the cloud's thickness and the amount of radiation reflected, absorbed or transmitted back to the observer. Thin clouds provide little resistance to radiation and can appear white or appear to take the color of their background. High tropospheric clouds - such as contrails - appear mostly white as they are formed entirely by ice crystals.

1.1.2 Contrails

Condensation trails (Contrails), are long, thin artificial clouds that can be created under certain conditions and left behind by aircrafts. Contrails form as hot humid exhaust from jet engines mixes with a cold low pressure atmosphere. The water vapor condenses and freezes on soot particles left by the engine’s exhaust creating a “man-made” cloud. Contrails were first noticed in the 1920’s when the first high-altitude flights began. Later in 1953 Appleman published a chart that can be used to determine when a jet airplane would or would not produce a contrail.

The regions of airspace that meet conditions for contrail formation are identified as Ice Super Saturated Regions (ISSR). When the aircraft travels through the ISSR the emissions inject water and other emissions into the air. The residual soot encompasses all primary, carbon-containing products from incomplete combustion processes in the engine. These particles acts as a means for the water to crystalize forming ice particles around them. As the exhaust water freezes around these particles forming contrails.

1.1.3 Conditions required for contrail formation

Contrail formation occurs at low temperatures, when there is an increase in relative humidity (RH) caused by the engine exhaust. Mixing the warm water vapor in a cool ambient causes an increase in saturation. The water droplets condensate on soot and volatile particles left by the exhaust and freeze forming ice particles. The trail of ice particles left behind by the engines creates a "man-made" cirrus cloud. Specifically contrails form in conditions where the ice saturation exceeded 100% (RHi), typically these conditions are found at altitudes above 8 km (26,000 ft) and temperatures below -40°C.

Contrail formation is directly related to the Relative Humidity. The Relative Humidity is typically expressed as a percentage and it represents the actual amount of water vapor in the air compared to the total amount of water vapor that can exist in the air at its current temperature without condensing. Warm air can hold more water vapor than cold air, therefore an equal amount of water will yield a higher relative humidity in colder air.

The amount of water introduced by the aircraft can be appreciated in an ideal balanced equation for combustion of kerosene:

$$2C_{12}H_{26}\,(\mathrm{l}) + \ 37O_2(g) \rightarrow \ 24CO_2(\mathrm{g}) + \ 26H_2\mathrm{O}(\mathrm{g})$$

Equation 1: Ideal balanced combustion of kerosene

The combustion yields approximately 1.24 tons of water for each ton of kerosene used. Under the appropriate atmospheric conditions, this introduction of water to a saturated atmosphere results in the contrail formation. Along with the CO_2 and H_2O an actual combustion will yield other emissions.

$$C_{10}H_{22} + \ 15.5O_2 \rightarrow \ 10CO_2 + \ 11H_2\mathrm{O} + 3.76(15.5)N_2 + 10.6kcal/g$$

Equation 2: Jet Engine Chemical Reaction

Table 1: Typical Emissions from Aero Engine at Cruise

Emission	From 1 kg fuel
CO_2	<3160 g
H_2O	<1290 g
NO_x	<15 g
SO_x	<0.8 g
CO	<0.6 g
Hydrocarbons	<0.01 g
Particulates	<0.05 g
Air	Large Amount

The formation of ice particles in these young contrails are usually smaller than typical cirrus particles (1–20 μm). Ice particles in contrails are non-spherical and these shapes of cirrus crystals vary in a "not-well-known manner ". These frozen droplets are described as droxtals which differ from spheres in their scattering phase functions. However as the contrail ages the size and shape of the crystals approach that of natural cirrus and can be modeled as spheres.

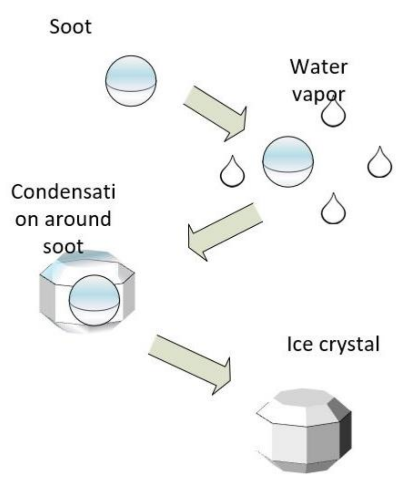


Figure 3: Ice crystal formation

Contrails are effectively cirrus clouds, and they act with similar properties. As cirrus clouds, they have a warming or cooling effect depending on conditions. In general they warm during night and cool during day depending on solar zenith angles. They also tend to cool over dark cold surfaces and warm over bright warm surfaces.

1.1.4 Radiative Forcing

Radiative Forcing (RF) is the net energy change on the Earth's atmosphere due to some perturbation. RF is used as a measure to determine the change from the time prior to industrial era (i.e. 1750) to present-day, and is typically measured as the change imposed at the top of atmosphere (TOA) in W/m^2. There natural sources of RF are: 1) solar irradiance, 2) volcanic forcing, and 3) asteroid impacts. During the period analyzed (1750 – 2012) no major asteroid impacts have occurred so this factor is not considered. The volcanic forcing is dramatic and highly episodic. Solar irradiance is the dominant source of RF. The IPCC has estimated the RF at the Top of Atmosphere at: 0.12 W/m^2 (0.06 to 0.30 W/m^2) [ref].

We have created anthropogenic sources of RF affect both the atmosphere and our land surfaces. A large number of Green House Gases (GHGs) have had a substantial increase over the Industrial Era, some of which are exclusively of anthropogenic origin. Additionally, human activities such as agriculture have modified the land and changed the surface albedo.

We have injected gases and aerosols into the atmosphere both directly and as a secondary product of chemical reactions. While water vapor is also injected to the atmosphere, the lifetimes of gases and water can be lengthy and vary substantially. The best estimate today for the total anthropogenic RF over the industrial era is of 2.4± 0.6W m–2, and according to the IPCC "*It is* *virtually certain that the anthropogenic RF is positive*". The latest estimates find that the RF is 50% higher compared to estimates of AR4 (2005) due primarily to reductions in estimated aerosol RF but also to continued growth in greenhouse gas and a greater uncertainty due to its inclusion of additional impacts on clouds.

The combined effect of well mixed greenhouse gases (WMGHGs) was estimated in AR4 (1750 to 2005) to be 2.63 W m–2 where the four most important gases were CO_2, CH_4, dichlorodifluoromethane (CFC-12) and N_2O in that order. (Myhre 2013). Hansen et al. (2005) estimated a mean radiative forcing of 0.03W/m2 with a maximum of about 2W/m2 over the United States.

1.1.5 Radiative Forcing due to Contrails and Contrail-Induced Cirrus

Contrails net radiative forcing is the result of the change in divergence of solar and infrared radiation fluxes within and below the contrail in the upper troposphere (Liou et al., 1990; Strauss et al., 1997; Meerk Otter et al., 1999). The area below a contrail has shown a change in heat source in the order of 0.3 K/day for 100% cover. Sassen (1997) found a reduction of solar radiation of 40 W /m2 when measured locally in the shadow of contrails. The effect of the contrail is negated when it is located above a thick cloud. Any heating would only occur above the lower cloud and the effect of the contrail on the surface is essentially zero. During the day, the radiative forcing by contrails is positive however it is strongest during the night because of the absence of negative SW forcing.

The role contrails play in climate has been elevated in AR5. In its evaluation during AR4 the IPCC assessed the RF of contrails as +0.01 (–0.007 to +0.02) W m–2 and provided no estimate for contrail induced cirrus. In AR5, the new estimate of RF due to contrails is set to +0.01

(+0.005 to +0.03) W m–2 and an Effective Radiative Forcing (ERF) is estimated to combine contrails and contrail-induced cirrus to +0.05 (+0.02 to +0.15) W m–2. The overall effect of subsonic aircraft was estimated in the Aviation and the Global Atmosphere, Special Report of IPCC to be 0.05 Wm-2 in 1992 and expected to grow to 0.19 Wm-2 by 2050, this includes the combined effect of carbon dioxide, ozone, methane, water vapor, contrails, and aerosols, but do not take into account possible changes in cirrus clouds. Other estimates of contrail induced RF have been made; for example Wilcox et al. (2012) estimated a contribution from civilian aircraft in 2005 of 0.0009 (0.0003 to 0.0013) W m–2 with high confidence in the upper limit. In 2011 Ulrike Burkhardt and Bernd Kärcher estimated that contrail coverage over the US exceeds 1% with coverage over the eastern corridor even higher. They estimate a net RF 0f .0375Wm-2 with areas over eastern US and central Europe reaching over 0.3Wm-2. They classify that contrail induced radiative forcing as "one of the largest single aviation-related radiative-forcing components". They estimate the contrail-cirrus radiative forcing offset by the natural-cloud feedback to yield a radiative by contrail induced cloudiness (CIC) of about 31 mW/ m2.

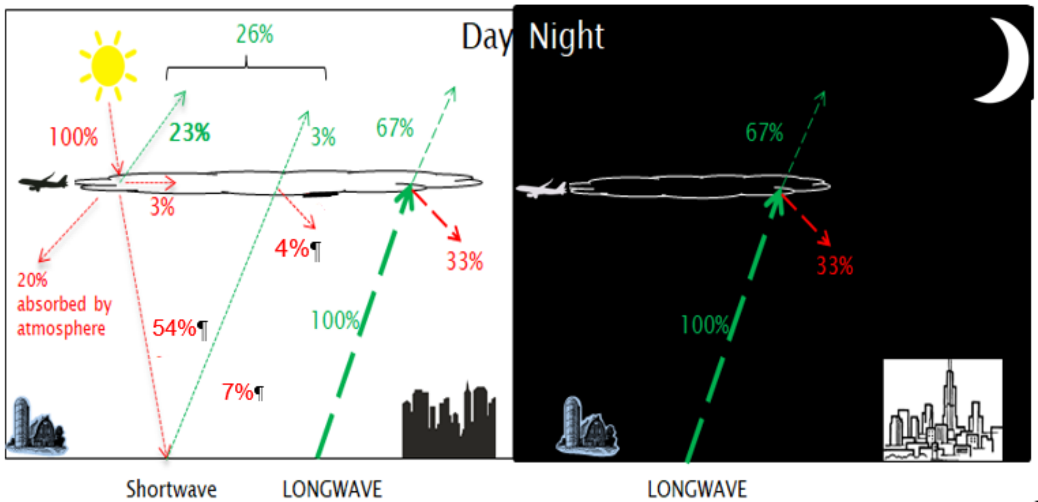


Figure 4: Contrail & CO_2 induced RF

The potential impact contrails could have on the environment has triggered multiple efforts. These efforts have used a reduced set of data focusing either on specific high days or routes; which facilitates the evaluation of the methods. The analysis of a full year of data suggests that the conditions to create ISSR change significantly day by day and at each altitude. This leads suggests there is benefit to analyzing the specific flight path an aircraft will take to predict contrail formation. This effort attempts to find alternative flight altitudes for all US flights during a day and estimate contrail formation to predict the RF impact of the flights would have as the combined effect of cloud cooling and warming.

According to the IPCC AR5 the global mean radiative forcing in w/ m^2. Well Mixed greenhouse gases (CO_2 , CH_4 , N_2O and halocarbons) account for 2.83 [w/m^2] while contrails and contrail induced cirrus account for 0.05 [w/ m^2]

Table 2: Radiative Forcin estimates for Contrail Cirrus

Radiative Forcing	Best Estimate or Range	Uncertainty Range with 2/3 Probability	Status of Understanding
1992			
Line-shaped contrail cirrus	0.02 Wm^{-2}	0.005-0.06 Wm^{-2}	fair
Additional aviation-induced cirrus clouds	0-0.04 Wm^{-2}	-	very poor
Other indirect cloud effects	-	either sign, unknown magnitude	very poor
2050			
Line-shaped contrail cirrus	0.10 Wm^{-2}	0.03-0.4 Wm^{-2}	fair
Additional aviation-induced cirrus clouds	0-0.16 Wm^{-2}	-	very poor
Other indirect cloud effects	-	either sign, unknown range	very poor

Based on IPPC AR5 ***Table 3-9:*** *Global radiative forcing by contrails and indirect cloud effects in 1992 and 2050 (scenario Fa1). No entry indicates insufficient information for best-estimate value*

Wilcox (2012) estimated the RF contribution from civilian aircraft to 0.0009 (0.0003 to 0.0013) W /m^2 with high confidence in the upper limit. In 2011 Ulrike Burkhardt & Kärcher (2011) estimated that contrail coverage over the US exceeds 1% with coverage over the eastern corridor even higher. They estimated the net RF to 0.0375W/m^2 with areas over eastern US and central Europe reaching over 0.3W/m^2.They identify that contrail induced radiative forcing as “one of the largest single aviation-related radiative-forcing components”. They estimate the contrail-cirrus radiative forcing offset by the natural-cloud feedback to yield a radiative by contrail induced cloudiness (CIC) of about 31 mW/m2.

The warming effect is strongly influenced by solar zenith angles. As the day progresses the change in the angle of the sun will generate a higher reflection of the shortwave radiation and none during the night. The longwave radiation reflected back to the atmosphere will remain throughout the day. This effect can be seen in the Table 3:

Table 3: Net RF by Hour

Time	Zenith	RFSW	RFLW	Net RF
0	0	-12.27	27	14.73
1	0	-12.27	27	14.73
2	0	-12.27	27	14.73
3	0	-12.27	27	14.73
4	60	-25.42	27	1.58
5	71	-25.85	27	1.15
6	80	-22.09	27	4.91
7	89	-21.87	27	5.13
8	99	-98.82	27	-71.82
9	111	-68.34	27	-41.34
10	126	-64.80	27	-37.80
11	147	-65.99	27	-38.99
12	176	-67.52	27	-40.52
13	207	-66.48	27	-39.48
14	230	-64.85	27	-37.85
15	246	-66.57	27	-39.57
16	258	-82.35	27	-55.35
17	268	-100.00	27	-73.00
18	277	-18.74	27	8.26
19	287	-25.47	27	1.53
20	297	-25.76	27	1.24
21	0	-12.27	27	14.73
22	0	-12.27	27	14.73
23	0	-12.27	27	14.73
24	0	-12.27	27	14.73

The data provided is charted in the first graph (March / Spring) we can see the effect of the reflection of shortwave radiation during the mid-day hours, while the longwave radiation remains constant. The four graphs provide a view of how the LW and SW radiation change throughout the year. Reference lines are added at 6:00 AM and 6:00 PM on all graphs. The solar angle (zenith) is provided on the left axis while the right provides RF in W/m^2.

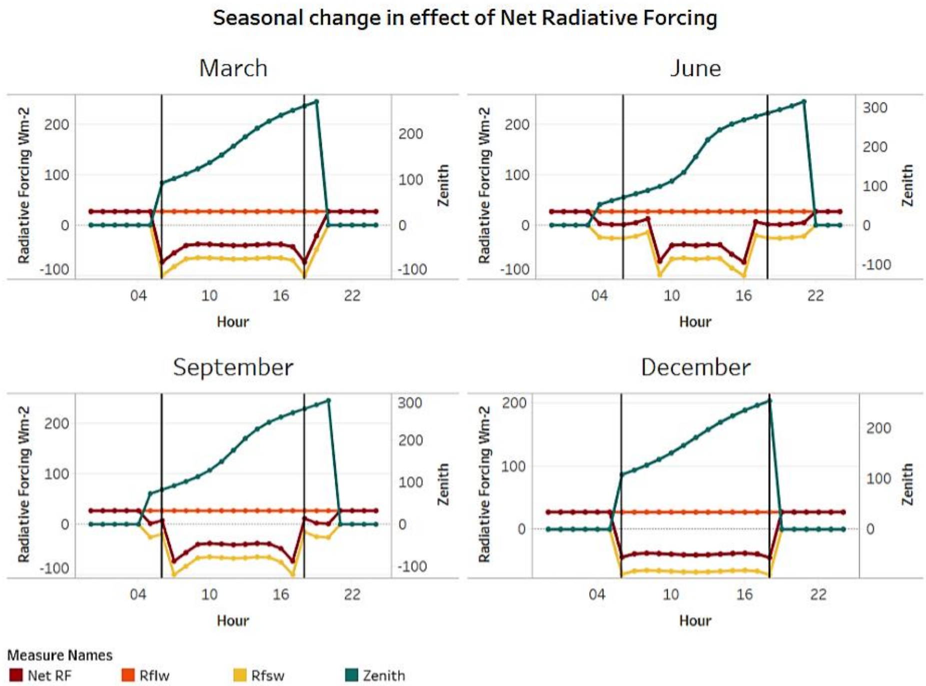


Figure 5: Contrails Day / Night and Seasonal effect.

The main driver for the longwave radiation reflected back is the size of the crystal. As the size of the crystal increases so does its capacity to reflect radiation. The next chart shows the change in radiative forcing as the crystal size increases.

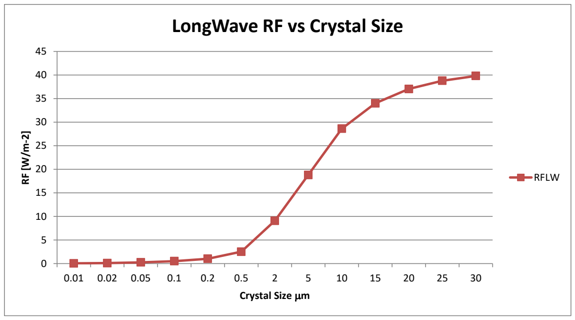


Figure 6: Longwave RF.

1.1.6 Contrails Frequency Forecast:

With the expected increase in passengers there is an inherent increase in the number of aircraft. The current volume of contrails is expected to rise along with the number of flights over the next 30 years, however contrail generation will also change as a function of the aircraft fleet and engine type. The change in the mix of aircraft size can provide some insight to this change. The Boeing fleet for example is expected to grow from 22,510 aircraft in 2015 to 45,240 by 2035.

Table 4: Boeing Market Forecast by Region.

2015	Asia	North Americ	Europe	Middle East	Latin	America2	C.I.S.	Africa
Large Widebody	270	100	170	140	-	50	10	740
Medium Widebody	540	320	360	320	20	20	60	1,640
Small Widebody	860	750	440	250	140	140	80	2,660
Single Aisle	4,540	4,010	3,370	590	1,280	650	430	14,870
Regional Jets	140	1,730	270	70	110	170	110	2,600
Tota l	6,350	6,910	4,610	1,370	1,550	1,030	690	22,510

2035	Asia	North Americ	Europe	Middle East	Latin	America2	C.I.S.	Africa
Large Wide-body	70	60	100	320	-	50	-	700
Medium Wide-body	1,590	460	610	840	40	70	80	3,690
Small Wide-body	2,340	1,150	1,140	610	350	170	300	6,060
Single Aisle	12,560	6,630	5,920	1,660	3,110	1,380	1,020	32,280
Regional Jets	310	1,520	150	80	160	230	60	2,510
Tota l	16,970	9,820	7,920	3,510	3,660	1,900	1,460	45,240

Airbus's fleet as of 2015 is reported as 18,020 aircraft and they forecast a fleet of 37,710 by 2035 with the following mix:

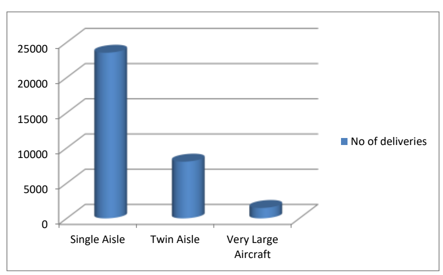


Figure 7: Airbus Expected Aircraft Deliveries 2016-2035

Small aircraft tend to be used for shorter, lower altitude flights and therefore generate less contrails. Focusing on mid to large aircraft and based on Boeing and Airbus estimates there will be an estimated 101% increase in “Contrail generating: size fleet by 2035.

Table 5: Fleet Change 2015-2035

		2015	2035	Pct Increase
Boeing	Large	740	700	-5%
	Twin Aisle	1,640	3,690	125%
Airbus	Large	806	1,688	109%
	Twin Aisle	4,392	9,191	109%
	Total	7,578	15,269	101%

Along with a change in the fleet there is an expected change in jet engines. Newer engines tend to run hotter and yield a higher RH. In 2000 Dr. Schumann lead an experiment which demonstrated this effect. Two aircraft, – an A340 and a B 707 - with different engines were flown side by side. The A340 was seen producing contrails while a B 707 does not. Due to the higher contrail generation by newer engines; we look at the expected volume of water to be introduced by aircraft in the future. Mauro Masiol, estimates the volume of water injected to the atmosphere will grow from 282 in 2008 to 390 by the year 2025.

1.2 Gaps in the current methodologies

Current models have focused on targeted trajectories and weather samples. A snapshot in weather provides no means to estimate contrail persistence in addition to preventing visibility to diurnal, seasonal, or geographic effects. Focusing on single or limited trajectories only provides insight to the behavior of one, of the thousands of aircraft in the air at any given time. These methods do not provide a holistic view of contrails being generated and therefore provide no guidance as to how to adapt contrail reduction policies at a regional level.

Additionally while over the past decade models have evolved and several methods have been used to evaluate the impact of contrails they have not taken into account Sun Zenith Angle, optical depth, contrail width or ice crystal size.

1.3 Problem Statement

The objective of this research is to produce a methodology to create an inventory of all contrails in a region enabling a systematic evaluation of the airspace. Additionally a methodology will be included tp evaluate the impact of changing produced by changing flight levels. The impact needs to be evaluated by flight to allow changes to be proposed on flight by flight biases. The evaluation needs to take into account contrail generation, persistence and the characteristics that affect the contrail's radiative forcing including Sun Zenith Angle, change in optical depth, contrail width and ice crystal size.

A case study is executed to validate the model using flight information from the ADS-B exchange. The flights are run through weather data matching the date and time of flight to calculate contrail generation, persistence and total RF induced buy the flight.

1.4 Summary of Results

The methodology proposed produces (1) Inventory of Contrails , (2) Impact of changing flight levels.

The process enables real-time assessment of contrails by its management of the data. The weather data required for this study provides 58 million data points per day creating a computational challenge. To lessen the computational requirements the methodology exploits new data structures and focuses on processing only critical information. By eliminating non-Ice Super Saturated (ISS) weather cells from the beginning the process reduces the overall processing effort in a scale of 50 to 1. The analysis of the ISSRs showed they can cover extensive areas resembling weather systems. They are most frequent over the Southeastern US air space and while they are present throughout the year they are most frequent from June to September concentrating between FL 320 to 370 with a max density of 30% at FL 340. Creating the inventory of identified that in average 7.4% of the volume of Contiguous United States (CONUS) air space between FL200 and FL400 exhibited Ice Super Saturated conditions. As a consequence, on the average day 6,160 flights (25%) traverse a CONUS cell with ISS conditions generating a daily average of 57.5K nautical miles of contrails (sigma 25K nm). The contrails generated an estimated average daily net radiative forcing of +7.08 mW/m^2. During the day, the albedo cooling of incoming shortwave radiation of –1.84 mW/m^2 by the contrails is negated by the +5.85 mW/m^2 warming of trapped outgoing longwave radiation. At night the contrails trap outgoing longwave radiation +3.07 mW/m^2.

On a per flights basis, the average flight generated 2.08 nautical miles of contrails. The average flight that traversed one or more CONUS ISS cells (contrail generating flight) generated 9.3 nautical miles of contrails. The average Net Radiative Forcing (NRF) per flight is 0.0003 mW/m^2. The average NRF per contrail flights is 0.00115 mW/m^2. Additionally, the average NRF per along-track distance nautical mile is 2.9 x 10-7 mW/m^2. The average NRF per contrail along track distance nautical mile is 0.00012 mW/m^2. An analysis of alternative flight levels suggests that increasing the flight level by 2000 ft provides a 62% decrease in contrails when accounting for their persistence (nautical miles – Hr). This change would theoretically decrease the average fuel burn by 2% (240 kg). An analysis on flights associating contrail NRF with their origin & destination and found that 30% of the NRF is generated by 22 flight routes. These findings show that in order to have an impact on contrail generated NRF it is not necessary to target all flights and the greatest impact can be achieved by raising the flight levels by 2000 ft during the summer months.

1.5 Unique Contribution

The methodology proposed provides a commercial scalable process to produce a contrail inventory.

Process achieved by

(1) Exploiting new data structures and focusing on processing only critical data. By eliminating non-Ice Super Saturated (ISS) weather cells at data extraction reduces the overall processing effort in a scale of 50 to 1.

(2) Preprocessing aircraft location to map into the weather grid and simplify the matching variables.
(3) Predetermining all potential aircraft locations and altitudes along the route providing a large reduction to weather space to be analyzed.

These items enable the process to benefit from efficient data structures and algorithms to process archived or live atmospheric data for an entire region over an extended period of time in combination with all flights present in the area. The effect is then calculated taking into account when and where contrails are formed as well as how they change over time. The methodology generates an inventory of contrails over the year providing insight on how the induced radiative forcing is affected by seasonality, geographic location and different flight levels.

The methodology provides details for:

(1) A Methodology to process basic National Oceanic & Atmospheric Administration (NOAA) Rapid Refresh (RAP) atmospheric data for the whole Contiguous United States (CONUS) airspace for 24 hours, 365 days
(2) A Methodology to processes the flight tracks for a full day of scheduled flights (e.g. 30,000) and evaluate multiple options for alternative flight levels simultaneously rather than evaluating multiple flight paths sequentially operations.
(3) A Methodology to implement a contrail formation model (Paoli & Shariff, 2016; Naiman et.al., 2011)
(4) A Methodology to implement a Net Radiative Forcing model that includes the aircrafts current position (Sun Zenith Angle) and contrail ageing (change in optical depth, contrail width and ice crystal size)

1.6 APPLICATIONS

The proposed methodology design can be used:

(1) By government agencies as a means to generate a holistic view of the contrail inventory in the region. Knowledge of the contrail formation and impact in Radiative Forcing can then be used to help guide policy to towards airline operations.

(2) By airlines to provide real-time evaluation tool to estimate the environmental impact of alternate flight plans.

2. LITERATURE REVIEW

This chapter provides a review the literature for Airpace Contrail Modleing. An evolution of the models is presented with the characteristics of selected models.

2.1 AIRSPACE CONTRAIL MODELS

Prior models have been built to estimate the formation and impact of contrails. These models have used varied trajectories and number of flights, but the scope of these has been limited. Campbell (2008), Kaiser (2012), Gao (2013) and Soler & Hansen (2014) all provided models based on a single trajectory; Campbell used a trajectory from Chicago O'Hare to Los Angeles LAX on Nov 17th 2001, Kaiser used a flight from Amsterdam Schiphol (EHAM) to Salzburg (LOWS) on Jan 19th 2012 and run alternative flight paths through the model using weather conditions that produce contrails in 3 of the 5 stations along the route. Soler & Hansen used of trajectory from SFO to JFK on June 30th 2012.

Other efforts included larger samples: In 2012 Chen & Sridhar used simulated flights in four sectors of the Kansas City air route traffic control center. Later in 2013 Chen & Sridhar (2013) and Gao (2013) used 12 pairs of trajectories with a total of 287 flights, in that same publication Chen & Sridhar (2013) simulated a larger set of flights into short medium and long range to determine their impact by distance.

While these methods provide insight to potential changes in contrails they present two limitations, the flights travel unique or specific routes and more importantly with they all analyze the flights through a single day of atmospheric conditions limiting the possibility to apply to provide guidance for an overall air traffic control strategy. A summary of these models as provided in Table 6.

Table 6: Contrail Model trajectory & Atmospheric conditions

Year	Author(s)	Trajectory(ies)	Atmospheric Conditions
2008	Campbell	1 -Trajectory: O'Hare to LAX	1 Day - Nov 17th 2001, RH > 100 (Via Temperature in RUC)
2012	Kaiser	1 -Trajectory: Amsterdam Schiphol (EHAM) to Salzburg (LOWS) . 19 Jan 2012	1 Day Jan 19th 2012, Weather in 5 stations along route @ flight time
2012	Chen /Sridhar	Simulated 3 sectors	1 Day April 23th 2010 - NOAA RUC 13X13 km 4/23/10 @ 8:00 , 34000ft
2013	Chen /Sridhar	12-Trajectory / 287 flights, All flights Simulated segmented into short, medium, long and transcontinental	1 Day - April 12th 2010 NOAA RUC 13X13 km
2013	Gao	12 City Pair /287 flights	1 Day - April 10th 2012 NOAA RUC 13X13 km
2014	Soler/Hansen	1 -Trajectory: SFO - JFK	1 Day Jun 30th 2012

2.2 CONTRAIL MODELING

Contrail formation has been based on either satellite observations or NOAA RUC files. Campbell (2008), Kaiser (2012), Chen & Sridhar (2012) and Soler & Hansen (2014) all use the RUC files to determine control formation and are consistently based on the Appleman criterion. Control persistence and spreading was estimated by Chen & Sridhar (2012) as 1000 m with and a contrail duration of 10,000s (~2:45 hr), and by Soler & Hansen (2014) by assigning a fixed 5hr duration to persistence.

Other than those estimates control persistence and spreading have not taken into account in contrail models leaving an open question as to the actual size and duration. Details on the individual models as provided in Table 7

Table 7: Contrail Model Contrail formation persistence and Spreading:

Year	Author(s)	Contrail Formation	Contrail Persistence	Contrail Spreading
2008	**Campbell**	RH > 100 (Via T) RUC	NA	NA
2012	**Kaiser**	ice saturation e*ice = 6.112e^(22.46Th/272.62+Th)	NA	NA
2012	**Chen /Sridhar**	rcontr<=RHw<100% and Rhi >=100%	1000m ; 10000s	
2013	**Chen /Sridhar**	Relative humidity with respect to ice greater than 100% from RUC Files	NA	Can spread up to 10 times linear contrail.
2013	**Gao**	Satellite observations	NA	Provided by NASA
2014	**Soler/Hansen**	r_{contr}<= RH_w < 100% and RH_i 100%.	Fixed: 5 Hr	NA

2.3 RADIATIVE FORCING MODELS

The environmental impact of the contrails has evolved over time. Campbell's (2008) model starts the aircrafts trajectory at FL340 and found the least contrail formation after descending to FL 300 and then climbing to FL 380 during a segments of the flight. The impact was estimated an additional 2.7% of fuel burn. Kaiser (2012), estimated radiative forcing induced by CO_2 as RF = 3.785 10-11 W/(m^2 t_{CO2}) and Contrail RF = 6.26 10-9 W/(m^2 hr)

As a means to quantify the impact of contrails Chen & Sridhar (2012) provide a model which defines a Contrail Frequency Index (CFI). Contrail frequency index is the number of aircraft in a defined volume that meet contrail persistence criteria. The CFI is zero for a space where conditions for persistent contrail formation do not exist. The CFIs were estimated at 20 US air traffic control centers at 8:00 AM at 34000ft. The RF induced is estimates fixed to 10 W/m^2 and use an Absolute Global Temperature Change Potential (AGTP) pulse which compensates for the climate effect over time measuring the change in global temperature at a particular time t due to an instantaneous disruption at t_0. Further details on AGTP can be found in appendix A.
Later in 2013 Chen & Sridhar (2013) simulate flights segmented into short, medium, long and transcontinental. After analyzing one month of data they find that due to their low altitude, short distance fights (< 500 miles) contribute the least to contrail reductions. Medium-distance flights (500 to 1000 miles), contribute the most to contrail reductions due to the combination of their altitude and the volume of flights itself. Long-distance (1000 to 1500 miles) and transcontinental flights had a present more varied result. The analysis concludes that for the top three contrail days in April, 2010, the contrail frequency index (CFI) per 1,000 miles for medium-range, long-range,

and transcontinental flights can be reduced by an average of 75%. The analysis suggests efforts to reduce contrails be focused on Medium-distance flights.

Gao's evaluation consists in two alternatives; a cost based optimization where a tax is imposed on a contrail generating flight path, and an AGTP based optimization. The AGTP valuation is performed in Kelvin per kg of fuel burnt or nautical-mile-contrail-produced and evaluated on RUC-20km weather data. The model set values of [2.1132E-15, 1.8207E-15, 1.5983E-15] AGTP fuel burn in K/kg at a time horizon of 25, 50 and 100 years and [3.3, 10, 30] mW/m^2 at 25, 50 and 100 years. (Values were generated by NASA Ames Model) The analysis finds that flight level adjustments are more efficient than lateral optimization in reducing contrails, however the long term effect of CO_2 is found to be more significant than the short term effect of contrails, therefore the author concludes to continue a fuel minimal profile.

In their model Soler & Hansen (2014) provide a 4D trajectory planning tool which assigns cost to passenger travel time, fuel, CO2 emissions, and contrail generation. The model estimates CO_2 based on Fuel burnt and contrails based on NOAA NCEP/DOE AMIP-II Reanalysis data with persistence fixed to 5 hours. The simulation finds that most contrails are minimized flying at FL 390 and 410 and suggests flying below FL 350. A summary is provided in Table 8

Table 8: Contrail Model Radiative Forcing evaluation:

Year	Author(s)	RF for Contrails
2008	Campbell	via FB
2012	Kaiser	$6.26\ 10^{-9}$ W/(m^2 hr contrail)
2012	Chen /Sridhar	10 W/m2
2013	Chen /Sridhar	10 to 80 mW/m2
2013	Gao	[3.3, 10, 30] mW/m2
2014	Soler/Hansen	15% of flight X αCO_2

3. METHODOLOGY

This section describes a commercially scalable method for inventory of contrails in a national airspace system using publicly available weather and flight surveillance track data, and models of contrail formation and net radiative forcing. The method provides the means to generate Contrail Inventories as well as perform real-time alternative flight-path analysis.

This method performs the pre-processing of atmospheric data for a large geographical area (e.g. CONUS), and takes into account:

- **Contrail Persistence**: Duration of contrail after if has been generated.
- **Contrail Width**: Lateral coverage of contrail as it spreads.
- **Sun Zenith Angle**: Calculated based on aircraft location at the time of contrail formation.
- **Optical Depth**: Change in optical depth of contrail over time.
- **Ice Crystal Size**. Change in crystal size of contrail over time.

With this method analysis can be done on diurnal, seasonal, or geographic effects for a large airspace (e.g. CONUS), and can be used in an operational context for real-time alternate flight path evaluations. The methodology‘s objective is to provide a systematic approach to determine the generation of contrails and potential prevention. The methodology is composed by two processes. The first process predicts contrails along a given path and to forecast their persistence to evaluate the contrail’s radiative forcing over its full lifespan.

Process 1: Airspace Contrail Inventory: The method to generate an Inventory of Contrails in a National Airspace System (NAS) includes six processes (Figure 8). The first three processes identify the regions in which the atmosphere meets the criteria for Ice Super Saturation (ISS). The fourth process generates the flight track data. The fifth process merges the weather and flight track data to estimate the contrails. The sixth process estimates the net radiative forcing.

Process 2: Evaluate Alternate Cruise FLs: The method is complimented with a process to simulate flights through actual weather conditions at various flight levels with the objective of comparing the tradeoff between contrail generation and change in fuel burn. The procedure provides a systematic approach with the intension of taking advantage of current data processing tools to create a process that can be used to estimate the fuel burn and the generation, size, persistence and radiative forcing induced by contrails of thousands of flights at a time. (Figure 16). Last, the model will calculate the fuel burn per route to enable a tradeoff analysis between climatic effect and fuel burn.

3.1. METHOD FOR INVENTORYING CONTRAILS IN THE NATIONAL AIRSPACE SYSTEM

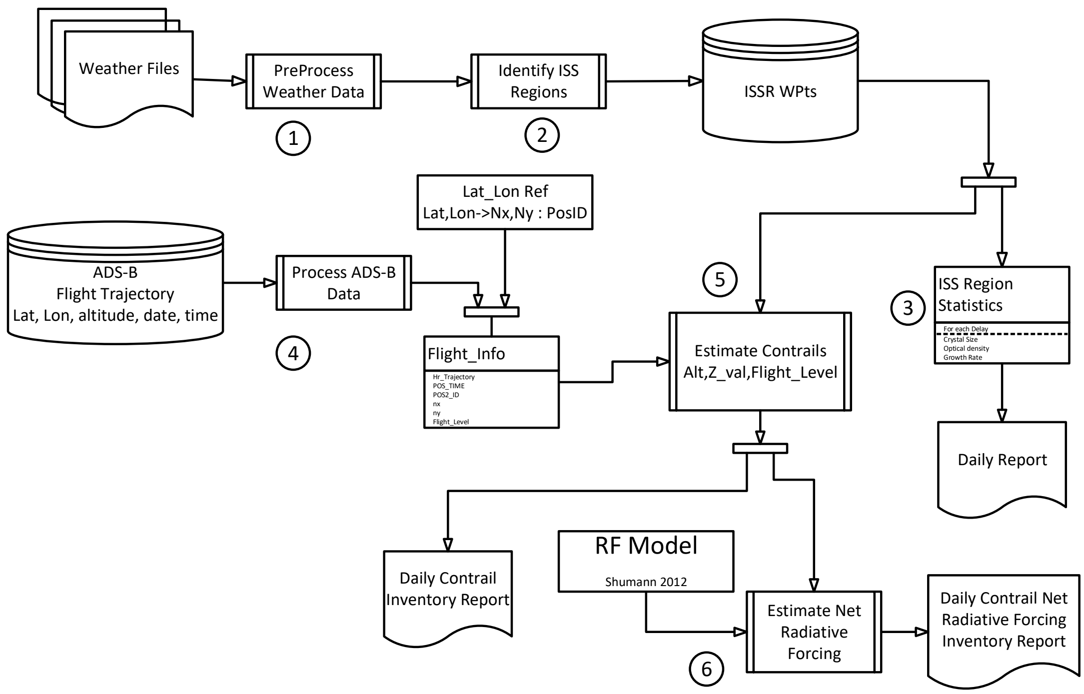


Figure 8: Summary of 6 processes to inventory contrails

3.1.1 PROCESS WEATHER DATA

Weather information is obtained from National Oceanic and Atmospheric Administration (NOAA) Rapid Refresh Products (RAP) weather files (http://www.nco.ncep.noaa.gov/pmb/products/rap/). These files provide information for a three-dimensional grid covering the US air space.

The data is provided in a GRIB (General Regularly-distributed Information in Binary form) file through the National Climatic Data Center (NCDC). The files provide weather indicators under a Lambert Conformal projection with a 13-km resolution (Figure 9). The data contains a grid of weather points with dimensions Nx = 451, Ny = 337 forming each plane and Z isobaric vertical levels (Z_Val). File details are included in Table 9.

Each hourly weather file is approximately 30MB of compressed data therefore each day entails over 58.3 million weather cells in 720 MB of compressed text, posing a challenge for both processing and storage. To mitigate the storage and processing issues, as each file is decoded it is swept once, by pressure level providing the data to an ISS identification data to a secondary process. The file is closed and no temporary space or files are required. To process one year worth of data, this initial step was taken to create an algorithm that would sweep through each of the 8,500 text files, decode the GRIB format and then extract values of interest.

Table 9: Geographic details of Weather information.

GRID DESCRIPTIONS	
Regional - CONUS (Lambert Conformal) - 13 km 130 ()	
Nx	451
Ny	337
La1	16.281N
Lo1	233.862E = 126.138W
Res. & Comp. Flag	0 0 0 0 1 0 0 0
Lov	265.000E = 95.000W
Dx	13.545087 km
Dy	13.545087 km
Projection Flag	0
Scanning Mode (bits 1 2 3)	0 1 0
Latin 1	25.000N
Latin 2	25.000N (tangent cone)
Lat/Lon values of the corners of the grid	
(1,1)	16.281N, 126.138W
(1,337)	54.172N, 139.856W
(451,337)	55.481N, 57.383W
(451,1)	17.340N, 69.039W
Pole point	
(I,J)	(249.315, 1051.539)

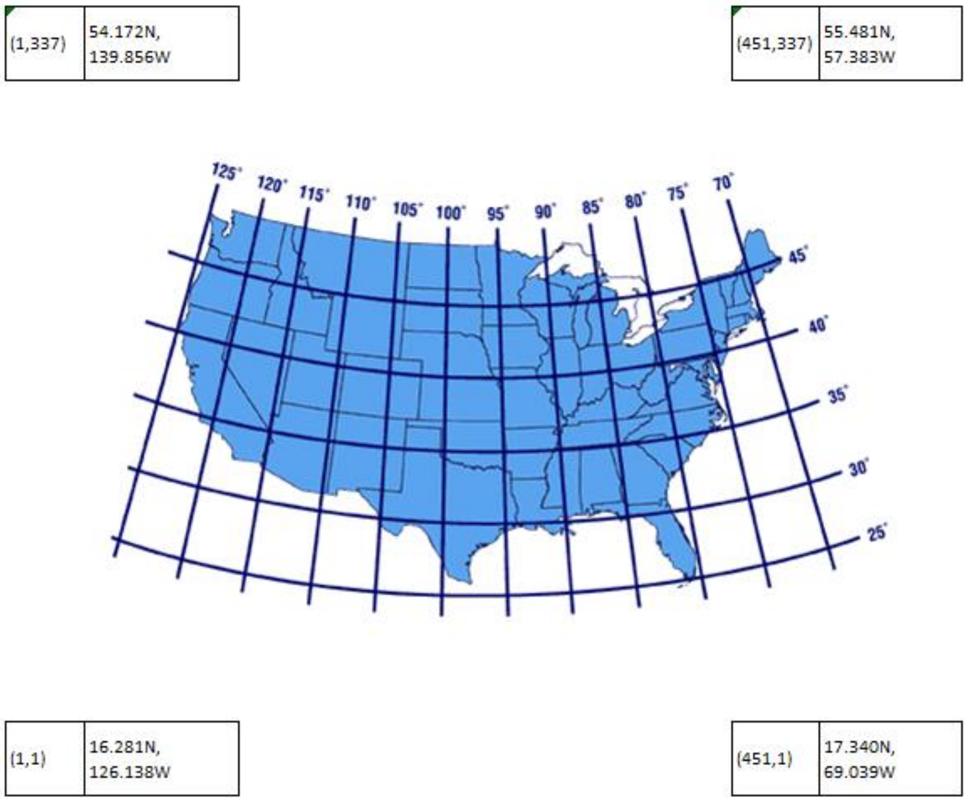


Figure 9: Weather Grid

The automation of the process requires weather files to be named with a convention that allows to identify the date and time of the file within the file name. The process then scans through a directory noting the date and time of file, sweeps through each RAP file, and extracts the needed data to identify ISSR. If the criteria is meet, the record is processed for loading.

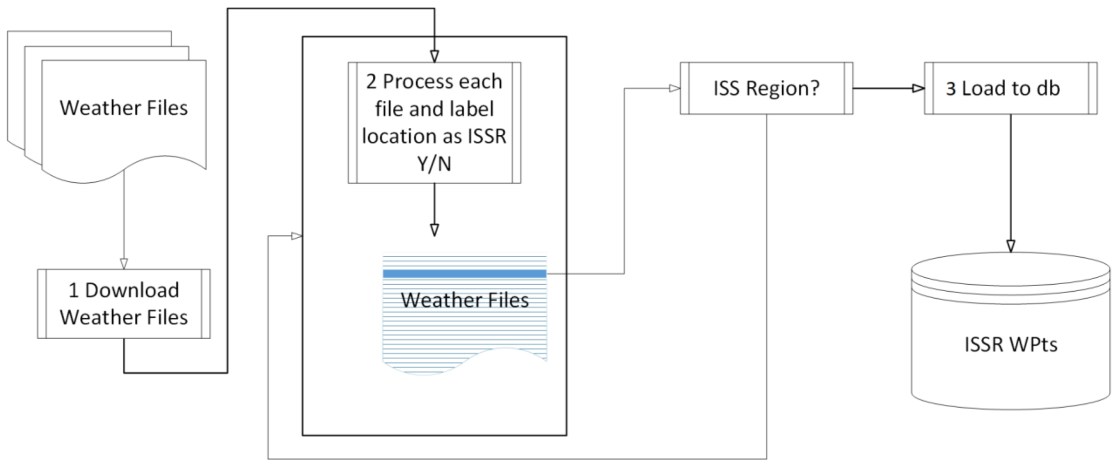


Figure 10: Weather data preprocessing

3.1.2 IDENTIFY ICE SUPER SATURATED (ISS) REGIONS

The second step uses weather parameters to determine if an ISSR is present and loads it to the database. To reduce the amount of data to be processed and loaded, at the time the GRIB file is read, only weather cells that meet the ISSR criteria are processed by this step. When the weather cell is read; the time and date from the file name along with the location along with its significant weather information (temperature, pressure, specific humidity) are all included in an insert query statement and loaded to the Contrail Inventory Database.(Figure 10).

For ISSR conditions to be met, atmospheric conditions in each cell are required to meet a temperature at or below –40°C (233.15 K) and a Relative Humidity of 100% (Appleman – Criterion). The temperature in Kelvin is provided within the file. Relative Humidity (RH), is estimated using vapor saturation tables and specific humidity. From the saturation tables at –40°C, 0.1 g/kg (or 0.0001 kg/kg) (Lnewam) of water vapor is sufficient to saturate the air. Any additional humidity, such as that provided by the engine's exhaust will result in contrails.

To simplify location data, the weather information is kept as provided in the weather information as Nx, Ny, Z. It is then necessary to locate the flight within the weather grid. For this purpose, a reference table is created which contains all Nx, Ny combinations along with the range of latitude and longitude that fall within each region.

The cells within the grid are 13 km by 13 km by 1000 feet cells. The grid is formed starting at 16.28N/126.13W and extends to 55.48N/57.38W. The vertical space that will be used ranges from FL200 to FL400. There are 151,897 cells at each Flight Level, and a total of 2,431,792 three dimensional cells per hour. The collection of all cells and flight levels creates an "ISSR weather Grid" over the CONUS (Figure 11).

Each weather cell is uniquely identified within the grid based on its latitude and longitude and assigned a location Identifier (Loc_ID). This way each cell can be identified by its Loc_ID (1 to 151,897) and a Flight Level.

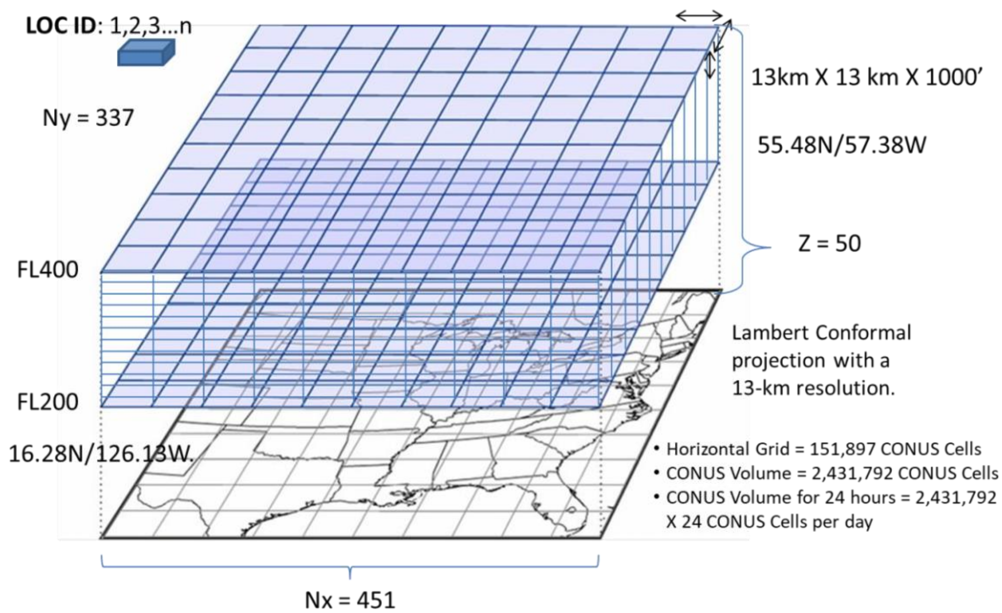


Figure 11: CONUS Cells of weather data; 13km X 13km X 1000'.

3.1.3 Ice Super Saturated Region Statistics Report

3.1.3.1 Atmospheric Metrics

The objective of the analysis of the ISSR regions is to measure the frequency of occurrence and coverage in latitude, longitude and altitude.

3.1.3.2 Horizontal Coverage Estimation

To estimate the coverage of the ISSR at each Flight Level the quantity of identified Ice Saturated weather cells is compared to the total weather cells on grid for the time period:

$$\%\ CONUS\ Volume = \frac{Count\ of\ ISSR}{Coordinates\ in\ Grid}$$

Equation 3: Percentage CONUS Vol

With this definition the % Conus Volume for a moth with 30 days is estimated as:

$$\boldsymbol{Count\ of\ ISSR\ Weather\ Cells}$$
$$= \text{ Weather Cells that satisfy the requirements within the month and Flight Level}$$

Equation 4: Count of ISSR Weather Cells

$$\begin{aligned}\boldsymbol{Coordinates\ in\ Grid} &= (\boldsymbol{Nx})(\boldsymbol{Ny})(\boldsymbol{Days\ in\ month})(\mathbf{24}\ \boldsymbol{hr}) \\ &= (\mathbf{451} * \mathbf{337})\ (\boldsymbol{Days\ in\ month})(\mathbf{24}) = (\mathbf{151},\mathbf{987})(\mathbf{30})(\mathbf{24}) \\ &= \mathbf{109},\mathbf{430},\mathbf{640}\end{aligned}$$

Once the Weather cells have been accumulated for the period a report is generated to provide statistics on the ISSR. A summary of the statistics for the ISS CONUS cells includes:

- Percentage of CONUS Cells presenting Ice Super Saturation (ISS) between FL200 to FL400
- Percentage of CONUS Cells presenting Ice Super Saturation (ISS) per Flight Level
- Ceiling and Floor of Ice Super Saturation (ISS) in CONUS Cells
- Geographic Coordinates of Ice Super Saturation (ISS) within the CONUS Cells
- Percentage of Rate of Change of Ice Super Saturation (ISS) within the CONUS Cells

A sample of the statistics is provided in setion 4. Additionally the ISS regions can also be visualzied in in 3-D vizualisations as shown in Figure 12

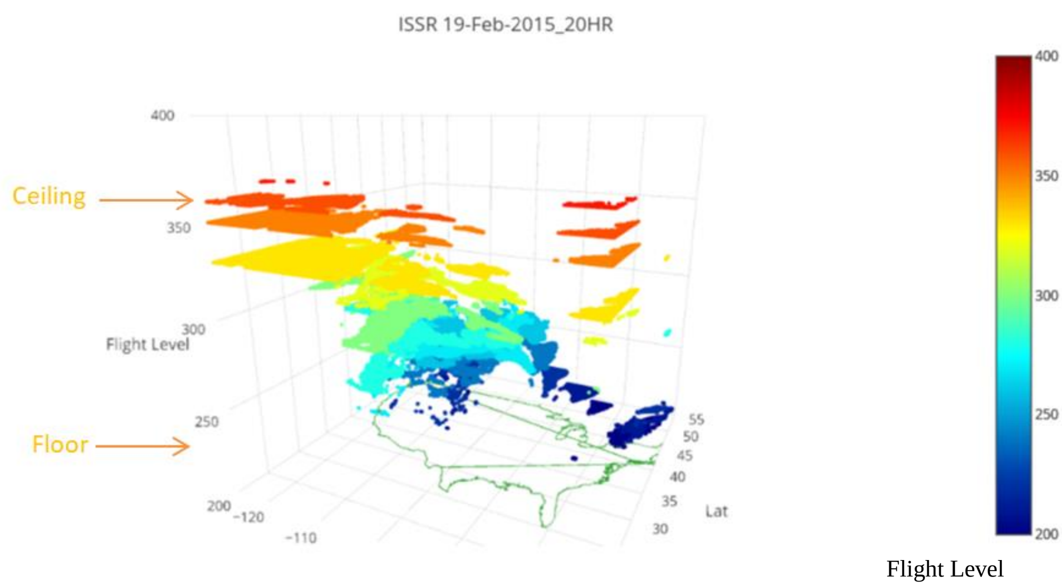


Figure 12: Visualization of ISS Regions in the CONUS (3D)

3.1.4 *Process ADS-B Data*

The trajectory information is obtained from the ADS-B Exchange (www.adsbexchange.com); a website that accumulates flight data from around the world and consolidates it to a single data source. The air traffic trajectories are loaded to the database and reference data is generated to convert it into the weather grid. The track data from the ADS-B files are processed. The data for each flight includes latitude, longitude, altitude, date and time. The update rate for records is variable from approximately 30 seconds to several minutes. Given the size of the CONUS cells, update rates within one minute provide at least one record in each cell. Flights with gaps in track records > 5nm are eliminated, as are flights with along-track distance of less than 50nm, and flights with missing records.

The flights are categorized into Day and Night flights by taking the mid-point time between takeoff and landing. If the time at the halfway point of the flight falls between 6AM and 6PM the flight is considered a “Day” flight, otherwise it is considered as “Night” flight.

3.1.4.1 Air Traffic Data

The air traffic data is used to fly the aircraft through the weather data to evaluate contrail generation and impact. To obtain an accurate evaluation of the impact of the actual flight trajectories flown.

Aircraft trajectory information model

Figure 13: Aircraft Trajectory data model

The process to enable trajectory information for integration with the weather data is summarized in the following steps:

1. Download DBS air traffic information.
2. Load air traffic Information to weather database.
3. Locate trajectory position within the weather grid Reference and associate a Loc_ID to each position.
4. Locate traffic altitude within Flight Level Reference and associate a Z_Value to each record.
5. Extract Hour from Date-Time information

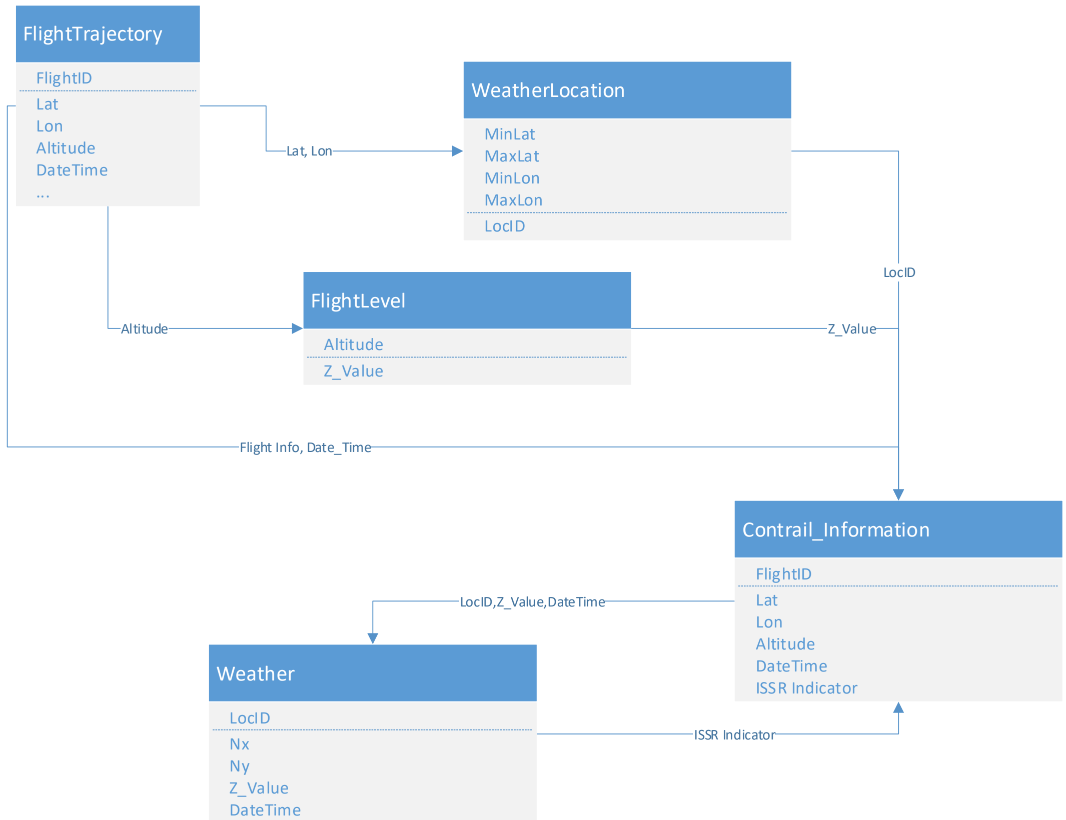


Figure 14: Aircraft trajectory information processing

3.1.5 Estimate Contrails

The flight path positions are mapped to the same grid as the atmospheric data, and each weather cell has been assigned the unique Position identifier (Loc_ID). With the mapping completed the traffic information is swept locating the aircraft position within the mapping and assigning the Position identifier to each record. To facilitate altitude matching, a minimum and maximum pressure is assigned to each flight level and then assigned a Z_Val from the atmospheric data.

The last variable: time, is provided at the top of each hour within the atmospheric data. For simplicity the minutes and seconds are truncated from the traffic data leaving both traffic and weather data in Hours.

Subsequent processes will only require three integers to locate the flight, a Position Identifier, a Z_val and the hour.

Once the atmospheric and trajectory information have been prepared, contrail estimation takes advantage of the following factors.

- Only ISS weather cells are included in weather data
- Each weather cell has been uniquely identified within its plane.
- Flight trajectory position mapped to Loc_ID.
- Weather Z levels mapped to flight level.

The combination of the data preparation enables the contrail identification to be simplified from matching: Latitude, Longitude, Altitude and time to Loc_ID, Z level, and Hour. This simplification is critical as it transforms a process of matching four real variables to matching 3 integers. Additionally because only ISS cells are included in the data no other validation is required. If a trajectory intersects with the weather cells it has met the criteria for contrail formation. The streamlined process can now be completed in 2 steps:

1. **Contrail Generation**

The aircraft position is evaluated in the weather information to see if ISSR conditions exist. The flight track is matched with the CONUS weather cells. When a track intersects with a CONUS Cells that meets the ISS criteria, a contrail is recorded.

2. Contrail Persistence

The contrail can persist and grow if the atmosphere is supersaturated with respect to ice (Jensen 1998). The expected duration of Contrails is up to 5 hours. The model assumes the contrail persists up to five hours as long as the ISSR conditions remain uninterrupted at the location.. When the ISSR conditions are not present, a majority of the contrails last only a few minutes and are not counted in this analysis.

Schumann (2011) and Chen, Sridhar (2012) estimate the size and duration of the contrail as 1000 m wide with a life of 10000 s (~2:45 hr) . Additionally Freudenthaler (1995) estimated the lateral growth of contrails to range between 18 m / min and 140 m / min (between 1 and 8 km per hour). To assign contrail locations the weather to trajectory intersections will be aggregated by summarizing to every 5° both in latitude and longitude while retaining the actual flight level. To prevent duplication, such as cases where an aircraft was recorded in the same section multiple times, unique instances of weather grid locations are counted.

3.1.6 Estimate Net Radiative Forcing

The effect of the radiative forcing (RF) at the top of atmosphere (TOA) produced by the contrails is calculated based on the model developed by Schumann (2011). This modification to the model assumes a spherical ice crystal throughout the life of the contrail. The contrail width is set to 500m for the first hour, 1000m, 2000m, 3000m, and 4000m for each subsequent hour that the ISS conditions exist. Optical depth parameters are set based on findings from Schröder [23] who finds the diameter of the crystals to grow from approx. 1 to 10 μ after that, the growth slows and stabilizes around D = 30 μ

The Contrail Net Radiative Forcing Model, illustrated in Figure 15, calculates the contrails and their associated net Radiative Forcing for a given flight trajectory and atmospheric conditions.

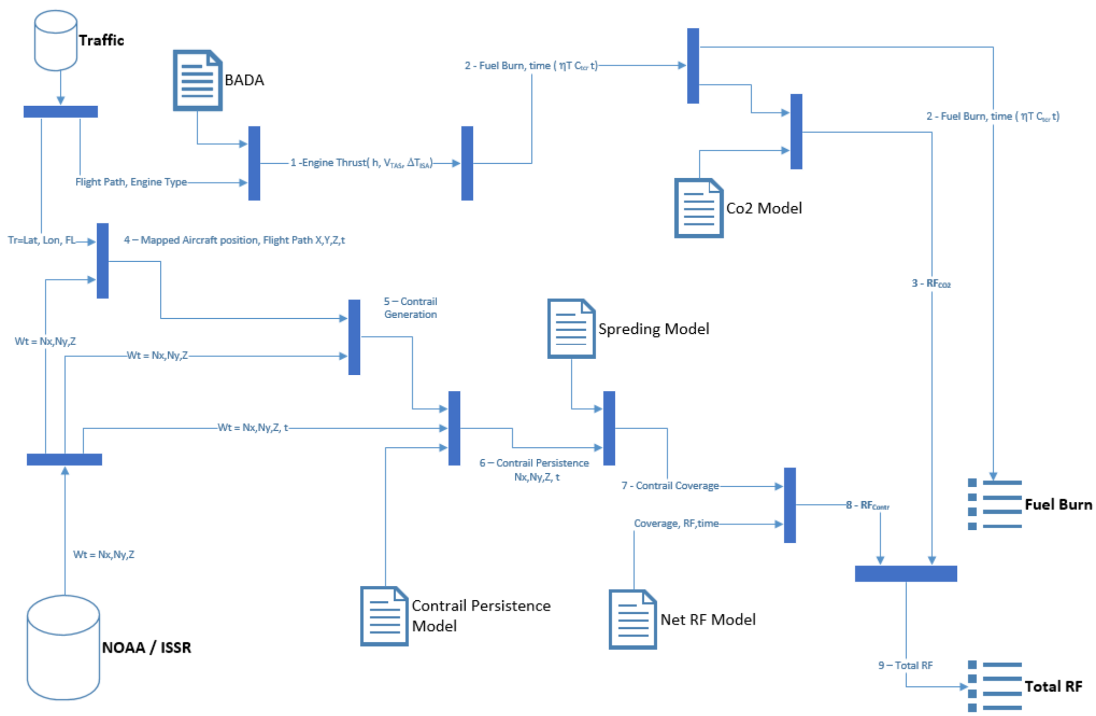


Figure 15: Net Radiative Forcing model

3.1.6.1 Contrail induced RF

The Net RF can then be calculated as the difference between the incoming solar shortwave radiation and the longwave radiation reflected back to space:

$$RF_{net}(\mathrm{t}) = \frac{1}{A_{Earth}} \sum_{flights} \int_{length} RF_{net}(t,s) W(t,s)\, \mathrm{ds}$$

$$\mathrm{EF} = \int_{lifetime} (RF_{net}(t',s) W(t',s))\mathrm{dt}'$$

Schumann (2011)

Equation 5: Radiative and Effective Forcing

The model estimates the net radiative forcing as the sum of the radiative forcing from longwave (RF_{LW}) and from shortwave (RF_{SW}). as shown in the equation below.

$RF_{Net} = RF_{LW} + RF_{SW}$

3.1.6.2 Radiative Forcing due to Longwave Radiation:

$RF_{LW} = [OLR - k_T(T-T_0)\ \{1-\exp[\delta_\tau F_{LW}(r_{eff})\tau]\}\ E_{LW}(\tau_c) \geq 0$

- The optical depth τ is estimated based on findings from Kärcher (2009)
- OLR is the outgoing longwave radiation. For this analysis we use OLR = 275 W/m^2
- T is temperature in Kelvin.
- r_{eff} = ¾ V/A [μm]
    - Where V is the particle volume and A the mean projected particle cross section area
- $F_{LW}(reff) = 1-\exp(-\delta_{lr}\, r_{eff})$
- $E_{LW}(\tau c) = \exp(-\delta_{lc}\, \tau_c)$

Parameters k_T, T_0, δ_τ , δ_{lr} are provided in Schumann (2005; Table 1– page 1396)

The Contrail optical depth τ vary between 0 and 2. Typically an optical depth τ will range from $\tau < 0.3$ for a thin cirrus to $\tau > 1$ in the case of thick cirrus clouds. The optical depth above the contrail τ_c ; ranges from0 to 10. For this analysis we will use the optical depths in Table 10.

Table 10: Conditions for contrail persistence

Time	Width	Diameter	Optical Depth τ	Optical Depth above contrail τc
0	0	0	0	0
1	500	10	0.4	0.36
2	1000	20	0.2	0.18
3	2000	25	0.08	0.072
4	3000	25	0.02	0.018
5	4000	25	0.01	0.009

Note: optical depths based on findings of Kärcher (2009)

3.1.6.3 Radiative Forcing due to Shortwave Radiation:

$$RF_{SW} = -SDR\ (t_A - A_{eff})^2\ \alpha_c(\mu,\tau,r_{eff})E_{SW}(\mu,\tau_c)$$

SDR can be calculated considering the SLR is the solar direct radiation in W/ m^2 and A_{eff} = RSR / SDR (reflected solar radiance / solar direct radiance). S_0 is the solar constant and μ defines the cosine of the solar zenith θ (SZA)

$$\alpha_c(\mu,\tau,r_{eff}) = R_c(\tau_{eff})[C_\mu + A_\mu R'C(\tau')F_\mu(\mu)]$$

$$\tau' = \tau F_{SW}\ (r_{eff})\ ,\ \tau_{eff} = \tau'/\mu$$

$F_{SW}(r_{eff}) = 1 – F_r$ [1-exp ($-\delta_{sr} r_{eff}$)]

$R_C(\tau_{eff})$ = 1-exp($-\Gamma\ \tau_{eff}$)

$R'_C(\tau_{eff})$ =exp($-\gamma\ \tau_{err}$)

$F\mu(\mu) = [(1-\mu)^{B\mu} / (1/2)^{B\mu}] -1$

$E_{SW}(\mu,\tau_c) = \exp(\delta_{SC}\tau_C – \delta'_{SC}\ \tau_{c,eff})$

$\tau_{c,eff} = \tau_c / \mu$

μ : cosine of the solar zenith angle (θ) can be found using the current flight latitude as follows:

$\mu = \cos[(latitude_{tr} /60)\ \pi/180]$

$\mu = \cos(\theta) = SDR / S_0$

SDR = $\cos(\theta)*S_0$

S_0 = 1,361 Wm-2

Cos(

where

- is the *solar zenith angle*
- is the *solar elevation angle*, = 90-
- the hour angle, in the local solar time.
- is the current declination of the Sun
- is the local latitude.

Parameters provided: Kt = 1.953, TO = 152, $\delta\tau$ = 0.941, δ_{1r} = 0.21, δ_{1c} = 0.16, t_A = 0.879, Γ= 0.242, Aμ = 0.361, Bμ = 0.709, Cμ = 0.709, Fr = 0.512, δ_{SR} = 0.157, δ_{SC} = 0.157, δ'_{SC} = 0.23

OLR value is based on the OLR recorded by NOAA during June 2016. Three sample were extracted to show high and low values. The OLRs sampled corresponded to 329, 194 and 274 W/m^2 . For the purpose of this analysis we will use al OLR or 275 $W/m^{2.}$

3.1.6.4 RF due to Fuel burn:

Fuel Burn and CO_2 generated RF are both calculated based on the Engine's Thrust. The thrust is converted into Fuel Burn and later to into RF.

1. **Engine Thrust:**

The flight path is used to calculate engine thrust using Eurocontrol's Base of Aircraft Data Revision 3.6 (BADA)

$$(T_{max})_{ISA} = C_{Tc1} (1 – h/C_{Tc2} + C_{Tc3}\, h^2)$$

Cruise Thrust is by definition equal to drag (T = D) The maximum cruise thrust can be calculated as

$$(T_{cruise})_{max} = C_{Tcr}\, T_{Max\ climb}$$

where C_{Tcr} is uniformly set to 0.95

2. **Fuel Burn:**

Based on the Engine Thrust, Fuel Consumption is estimated based on the Eurocontrol's BADA. The Nominal fuel flow for climb in [kg/min] can be found by multiplying the specific fuel consumption by Engine Thrust T:

$F_{nom} = \eta\, T$

Where

Jet: $\eta = C_{f1}\ (1+(V_{TAS}\ /\ Cf2))$

η in [hg/min/kN] ; V_{TAS} in [knots]

Cruise fuel flow is adjusted with a cruise fuel flow factor f_{fcr} :

$$f_{cr} = \eta\ T\ f_{fcr}$$

Descent fuel flow can be found as:

$$f_{min} = C_{f3}\ (1-h/C_{f4})$$

3. RF induced by CO_2

Emission of CO_2 can then be calculated using the same equation used in the FAA Aviation Environmental Design Tool (AEDT): $CO_2[g] = 3155\ FB[kg]$

Where CO_2 is in grams and fuel burn in kg. The impact on RF from CO_2 is estimated to 0.028 W/m^2 (Lee 2010)

$RF_{FuelBurn} = 0.028 * CO_2[g]$.

3.2. REAL-TIME ALTERNATIVE CRUISE FLIGHT LEVEL METHODOLOGY

This method sets a process to simulate flights through actual atmospheric conditions at various flight levels with the objective of determining the next lowest/highest cruise flight level to avoid generating a contrail while calculating the difference in fuel burn required to fully evaluate the alternate Cruise Flight Levels.

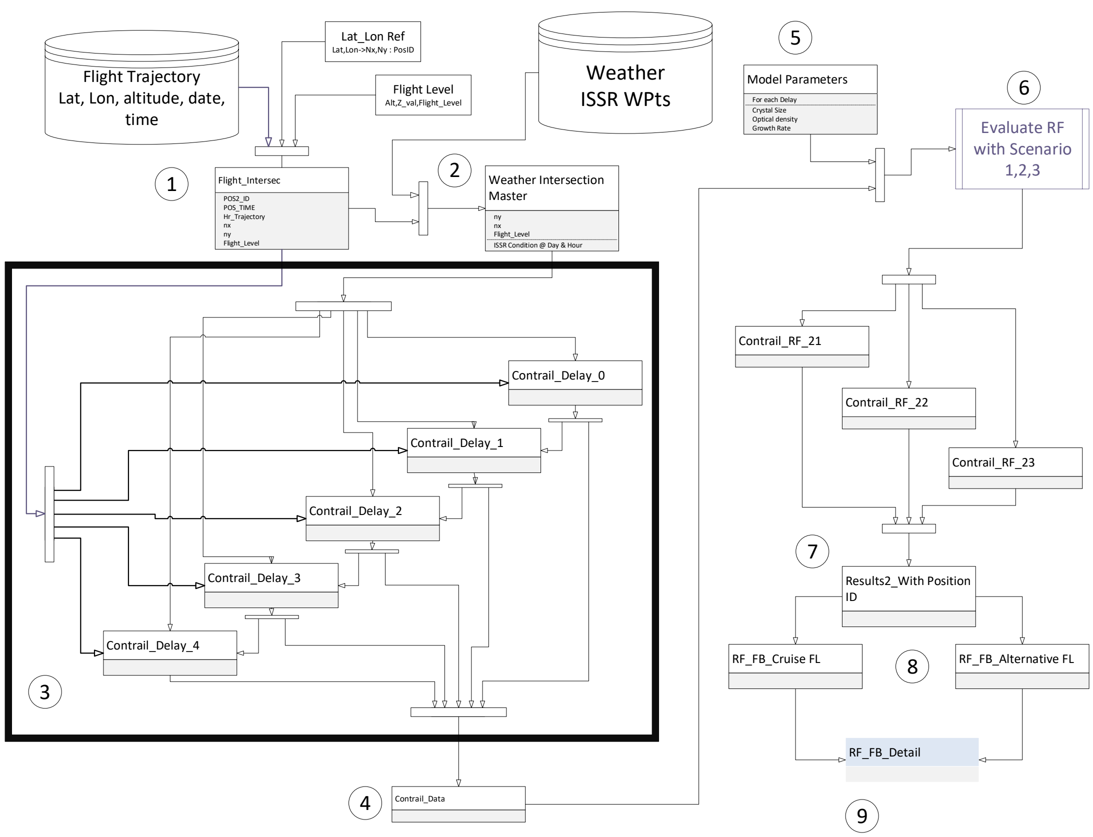


Figure 16: Contrail generation and fuel burn model

3.2.1 *ALTERNATIVE FLIGHT LEVEL EVALUATION:*

1. Create a universe of possible flight locations along the planned route as possible intersections with an Ice Super Saturated Region. The process uses the flight trajectory sequence of locations: longitude, latitude and altitude and assisted by a translation tables, maps them to a weather cell spaces. This allows the physical location to be replaced by a weather cell.
2. Using the potential weather cells the process creates a subset of the weather data. The subset will be limited to the latitude and longitude combinations found for flight trajectories while including all possible altitudes. This step proved critical at a later stage as it reduced the universe of the ISSR cells from 1.1 billion to 189 million.
3. Generate Contrails. The simulation of contrail generation is broken into a two step process:
   i) Contrail generation: The first ISSR penetration is determined by matching the flight intersection to the weather information to generate the possible contrail locations. The contrail space is generated matching the lat – lon, date and time while allowing the flight level to vary within cruise flight level +/- 8,000 ft.
   ii) Contrail persistence: Persistence is estimated from hours1 to 4 after the contrail formation: This process matches the intersection table to the next hour of weather to generate possible contrail persistence. The contrail persistence space is generated matching the lat – lon, date and extended time locations as long as a contrail was generated in the prior step(s).
4. Next we consolidated contrails generated during the hours 0 to 4 removing duplicates. This will yield a distinct combination of flight, date and location.
5. Set parameters for Optical depth: τ Optical depth above contrail: $\tau 2$ and contrail growth rate. The case study presented evaluates 3 sets of parameters:
   1- Expected RF.
   2- Rapid Crystal growth.
   3- Rapid Contrail spread.

Table 11: Parameters for RF scenarios

Treatment	Persistence [Hr]	Crystal Diameter [mm]	Contrail Optical Depth	Contrail Width [m]
1	0	10	0.4	500
	1	20	0.2	1000
	2	25	0.08	2000
	3	25	0.02	3000
	4	25	0.01	4000
2	0	10	0.5	500
	1	20	0.4	1000
	2	30	0.1	2000
	3	30	0.08	3000
	4	30	0.05	4000
3	0	8	0.4	1000
	1	15	0.2	4000
	2	20	0.08	8000
	3	20	0.02	16000
	4	20	0.01	32000

6. Estimate Radiative Forcing for each scenario.
   i) Evaluate each flight within the "Possible Contrail Space". The possible Contrail Space is created by limiting contrail formation in hours +1 to +4 to weather cells where a contrail was formed at the time of travel (hour 0). During the case study this limitation reduced the potential contrail cells from 53,000,000 to 18,000,000 facilitating the search and allowing the evaluation of 800 flights over 1 year in approximately on a desktop computer in 4 hours per scenario.
   ii) Contrails along Track: Generate RF along trajectory controlling FL to match actual trajectory
   iii) Alternative Contrail options: Generate RF along trajectory allowing FL to vary
   iv) Evaluate fuel burn along track.
   v) Evaluate fuel burn along alternative flight level.
   vi) Evaluate RF induced by CO2
7. Consolidate results for radiative forcing scenarios.
8. Evaluate fuel burn for path along track and alternative flight levels.
9. Consolidate and build *Radiative Forcing and Fuel Burn detail* for analysis.

3.2.2 *FUEL BURN EVALUATION:*

Fuel flow is calculated based on Eurocontrol's Base of Aircraft Data (BADA) B737 aircraft adjusted cruise fuel flow. The estimation of fuel burn is calculated assuming the ascent and descent are performed at a constant 3° angle and 250 VTAS with an initial and final altitude of zero. The distance is estimated along the track of the trajectory and cruise speed is assumed constant set to the mode of the speed recorded for the flight. This provides a direct measurement of the change in fuel burn due to flight level and departure - arrival.

Radiative Forcing is calculated for Fuel Burn via the CO_2 generated

1. Climb and descent per minute:

   Climb per min = TAN (3 /π) *(VTAS * 101.268) [ft/min]

   Descent per min = TAN (3 /π) *(VTAS * 101.268) [ft/min]

2. Climb and descent time

   Climb time = Evaluated Flight Level / Climb per min [min]

   Descent time = Evaluated Flight Level / Descent per min [min]

3. Cruise time

   Cruise Time = Total actual flight time – Climb Time – Descent Time

4. Engine Thrust:

   TClimb = CTc1 (1 – h/CTc2 + CTc3 h2)

   TDescent = CTc1 (1 – h/CTc2 + CTc3 h2) * 0.074

   TCruise = CTc1 (1 – h/CTc2 + CTc3 h2) * 0.95

Note:h set constant at 18000 for climb and descent; during cruise, h is set to the cruise flight level

5. Fuel Burn:

Based on the Engine Thrust, Fuel Consumption is estimated based on the Eurocontrol's BADA. The Nominal fuel flow for climb in [kg/min] can be found by multiplying the specific fuel consumption by Engine Thrust T. V_{TAS} is set to 250 for ascent and descent and set to mode of cruise speed for cruise portion.

$F_{nom} = \eta\ T$

$\eta_{Climb} = C_{f1}\ (1+(V_{TAS} / C_{f2}))$

Where

η in [hg/min/kN]

V_{TAS} in [knots]

Cruise fuel flow is adjusted with a cruise fuel flow factor f_{fcr} : 0.9737

$f_{cr} = \eta\ T\ f_{fcr}$

Fuel Burn $_{Climb} = (\eta_{Climb}\ T_{Climb} /1000)$ Climb Time

Fuel Burn $_{Desc} = C_{f3}\ (1\text{-}h\ / C_{f3})$ Descent Time

Fuel Burn $_{Cruise} = (\eta_{Cruise}\ T_{Cruise}\ /1000)$ Cruise Time * 0.9737

3.3 MODEL DATA

Analyzing weather and air traffic data to create the inventory requires dealing with large datasets. For this purpose a relational database is used to load and compare the weather and traffic information. The data is stored in the Contrail Inventory database. The database server to house the two main sources: weather and aircraft trajectory, along with several reference tables. A description of the data used for this analysis is provided in Table 12.

Table 12: Data Structure

ID	Table Name	Description	Source	Size (Records)
1	Airports	Airport Name and Location	Generated	4,227
2	Center Codes	Center Names	Generated	49
3	Flight_Levels	Flight levels being evaluated and mapping to Z_Val.	Generated	27
4	Lat_Lon	Mapping table to join weather to traffic data	Generated	151,987
5	Traffic	Aircraft trajectory sample 26-Jul-2006	ADS-B	675,750
6	Weather	Temperature, humidity by location and time	NOAA RAP files	1,176,676,072

1. Airports:

   This table provides a list of airports along with their location, city and country.

2. Center Codes:

   Center Names and codes for reference only.

3. Flight Levels:

   Translation table to convert altitude in feet to a flight level and Z_value. The table provides a min and max pressure altitude and its equivalent Z_Value. By predetermining these ranges a many to many join is reduced to a one to one match.

4. Lat Lon:

   Mapping table to convert latitude, longitude into Nx, Ny. This table allows a establish ranges by latitude and longitude to be assigned a unique Nx, Ny value. By prepossession these ranges and assigning them a Loc_ID subsequent comparison queries that will involve millions of records can be simplified from a many to many join to a one to one join.

5. Traffic:

   This table contains ADS-B trajectory information for one day or air traffic. This information is provided in a standard latitude, longitude and altitude..

6. Weather:

   This table provides weather information extracted from the NOAA RAP files. The data contains temperature, specific humidity, and pressure at each given position. To discretize the analysis the location, position is maintained in its original Nx, Ny, Z by date and hour. The size of this table far exceeds all others in this set with over 1.1 billion records.

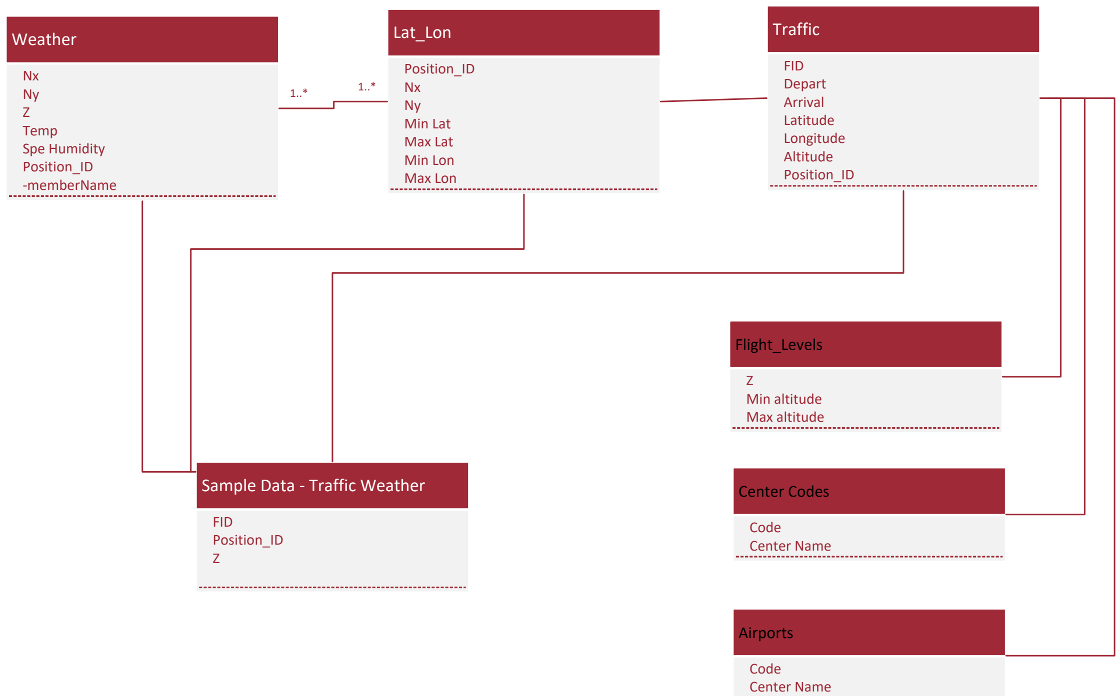


Figure 17:Weather and Traffic Database Model

Managing the data within a structured database enables us to interact with it via standard tools such as python and R. Additionally having the information in the database enables us to analyze the ISSR regions, compare these to flight records and provide insight to weather impact of alternative flight routes via standard Structured Query Language (SQL).

Given the volume of the data some steps were taken to preprocess the information to facilitate the analysis that the main challenge in this analysis will be matching weather to traffic data. Both provide information in location and time. Additionally, latitude, longitude and altitude are all continual dimensions. A direct analysis would therefore involve two four dimensional data sets, however some of this data is already discrete as is the location within the weather information.

- To simplify location, the weather information is kept as Nx, Ny, Z and the Lat_Lon reference was built to which contains all Nx, Ny combinations assigned a unique Loc_ID;

additionally a min and max latitude and longitude are assigned to create ranges. Next the traffic table swept locating the aircraft position within a lat-lon range and assigning the predefined Loc_ID to each record.

- To facilitate altitude matching a minimum and maximum pressure are assigned to each flight level and assigned a Z_Val from the weather data.
- The last variable: time, is provided at the top of each hour within the weather data. To allow matching, minutes and seconds are truncated from the trajectory data.

With these steps, analysis queries will only need to match Loc_ID, an integer within both sets, a Flight Level or Z_Val – integer in both sets and the hour.

4. CASE STUDY: DAILY CONTRAIL INVENTORY FOR U.S. AIRSPACE – 2015

The model is validated by running one day of flights though a year of weather data to generate a Contrail Inventory. The inventory proves several insights into contrail statistics. Based on these statistics, flights are selected and simulated through the weather with variable flight levels to evaluate potential for eliminating contrails by changing Cruise Flight Levels.

The NOAA provides RAP files with both actual and forecasted data. This analysis uses the actual weather files provided at the top of each hour. For the case-study atmospheric data was downloaded from the NOAA site. A total 276 days had complete data sets. Despite multiple efforts some days in January, September and October show incomplete atmospheric data.

4.1 CONTRAIL INVENTORY

4.1.1 ICE SUPER SATURATED REGION STATISTICS

In 2015, the monthly average of ISSR CONUS Cells was 116,865,118 (7.4%) with standard deviation of 37,399,966 (3.18%). The monthly ISSR CONUS Cells are shown in Table 13. The Summer months (Jun, July, Aug) have 1.9 times the ISSR CONUS Cell count of the winter months. The counts are accumulated over all locations in the CONUS and between FL200 and FL420.

Table 13: Count of 3D points by Month

Month	*ISSR Weather Cells*	*% of ISSR CONUS Cells*
Nov-14	74,790,763	5.90%
Dec-14	119,386,535	6.69%
Feb-15	75,086,577	5.68%
Mar-15	52,822,728	6.13%
Apr-15	117,735,997	6.82%
May-15	125,198,833	7.02%
Jun-15	141,545,934	8.20%
Jul-15	157,982,162	8.85%
Aug-15	159,774,273	8.96%
Sep-15	144,327,373	8.35%
*Oct-15**	6,639,306	5.80%
Total	**1,175,290,481**	**7.40%**
Avg	116,865,118	
Standard Deviation:	37,399,966	*3.18%*

*Partial data

The volumetric statistics from Table 13 are broken down by Flight Level for each month. Figure 18 provides an overall view of the seasonality by Flight Level. While ISSRs are present thought the year, seasonality can be seen with a high season from June to September. Flight Levels 320 and 360 have the highest percentage of ISSR CONUS Cells.

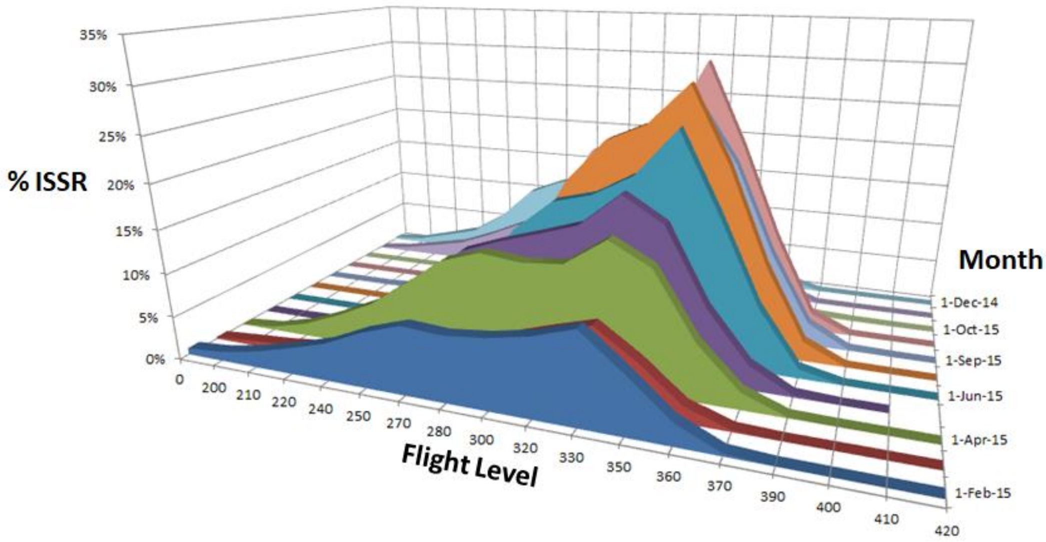


Figure 18: Percentage of ISSR Coverage in the US Airspace 2015

The ISSR CONUS Cells tend to have a volume in the shape of an octahedron (i.e. 3-D diamond) shape. The coverage of the CONUS at each altitude during the month of August 2015 is shown in Figure 19. ISSRs are present from Flight Levels 200 to 380, however the most frequent occurrence is at Flight Level 350.

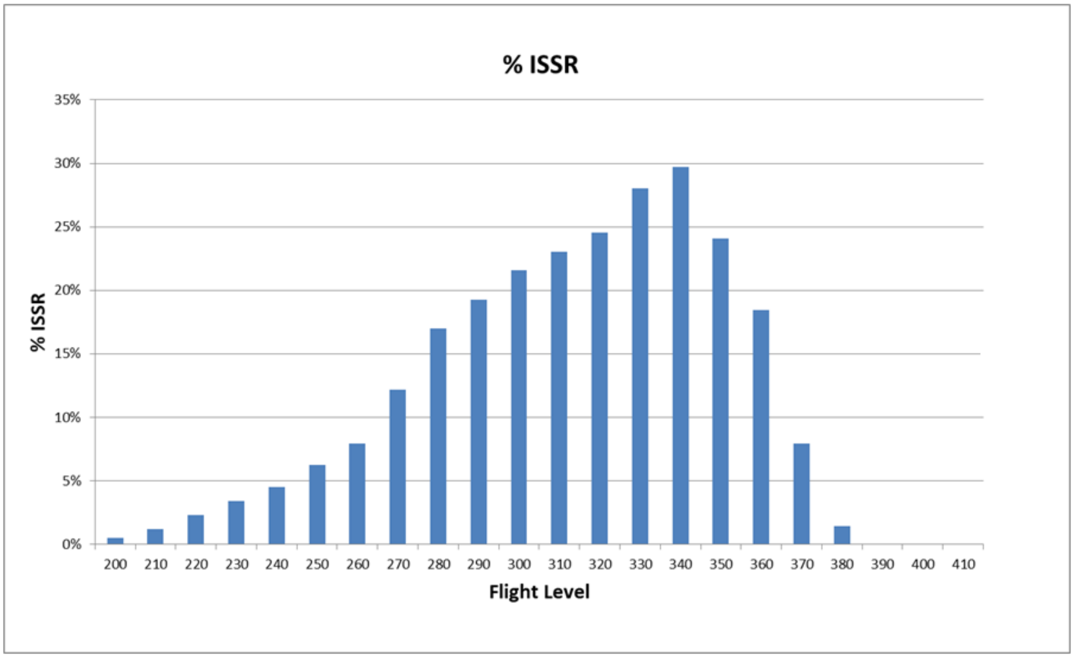


Figure 19: Average ISSR coverage by Flight Level in August 2015 (%)

The effect of seasonality by Flight Level is shown in Table 14. The table presents a concentration of ISSRs between FL 320 and 360 during the summer months, this frequency then drops within 2000 feet both above 360 and below 320. During February and March 2015 no ISSR were identified above Flight Level 350.

Table 14: ISSR frequency by Flight Level (Percentage of total air space)

Flight Level	Nov-14	Dec-14	Feb-15	Mar-15	Apr-15	May-15	Jun-15	Jul-15	Aug-15	Sep-15
380	0	0	0	0	0	0	2	4	4	4
370	2	1	0	0	2	3	9	13	14	13
360	1	1	0	0	2	2	9	12	11	12
350	5	6	3	2	6	9	18	23	23	19
340	10	14	8	6	15	17	26	28	30	26
330	2	3	2	2	4	7	11	11	11	11
320	12	22	15	12	21	24	24	26	24	23
310	8	11	9	5	10	10	9	9	10	7
300	4	6	3	2	5	7	5	5	6	4
290	8	13	10	5	14	15	9	5	7	6
280	3	7	4	2	7	7	4	1	2	2
270	4	8	5	3	9	5	3	1	1	2
260	2	3	2	1	3	2	1	0	0	1
250	3	5	4	2	4	1	0	0	0	1
240	2	2	2	1	2	1	0	0	0	0
230	1	1	1	0	1	0	0	0	0	0
220	1	1	1	0	1	0	0	0	0	0

Figure 20 combines the max and min Flight Levels in blue and red, along with the daily average ISSR coverage in green.

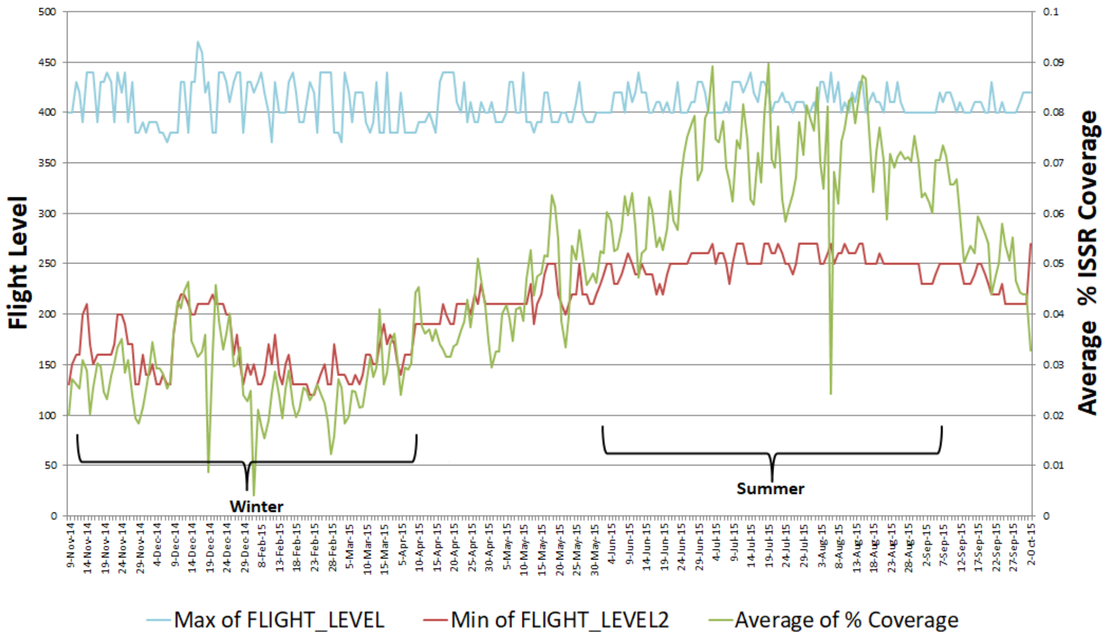


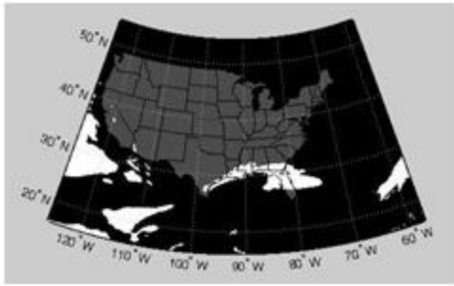
Figure 20: Average percentage of ISSR coverage by Flight Level (2014 – 2015)

The analysis of the atmospheric data showed the the ISSR can be found throughout the year with highest frequency from June to September. The monthly average % CONUS Volume is 20%, with aminimum of 1% and a maximum of 38%. Regarding altitudes, ISSRs were located covering 10% of the airspace at Flight Levels from 320 to 370 and occupied up to 30% of the airspace at Flight Level 340. The results indicate that super saturated ice regions constantly vary in size and location and can be covering a large geographic area on the CONUS.

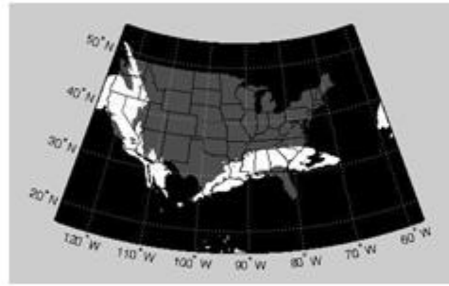
Geographic location

To ilustrate the ISSR seasonality and sensitivity to altitude two sequence of images are displayed. The images show ISSRs in white over a US map. The first sequence (Figure 21) illustrates differences by altitude. Hourly updates are show for the same time and date during the month of

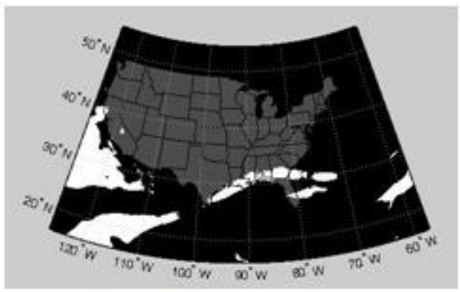
November for flight levels 340 and 360. The images illustate both the movement over time and how at FL 360 the ice saturated regions are larger over the sourhestern US air space. A similar sequence of saturate regions is produced in Figure 22 for the month of August. Comparing the two sequences the main change is that during August the ISSRs at flight level 340 cover the entire sourhestern US and a large section of the Atlantic ocean. While both sequences present differences between months, the difference from one flight level to another suggests a potential effect in contrail formation when changing flight levels.



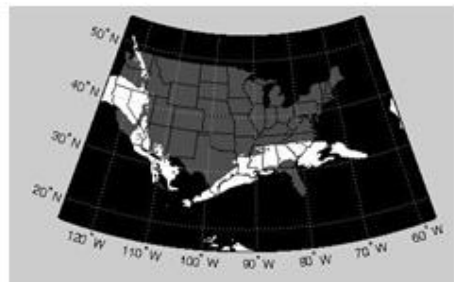
November 15th 2014 t:08:00 FL 340



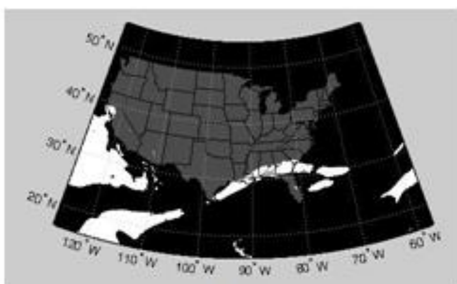
November 15th 2014 t:08:00 Z:26 FL 360



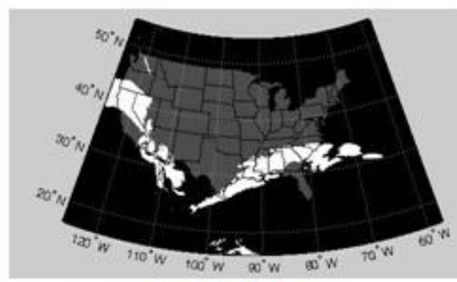
November 15th 2014 t:09:00 FL 340



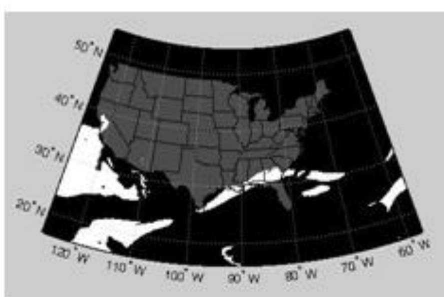
November 15th 2014 t:09:00 FL 360



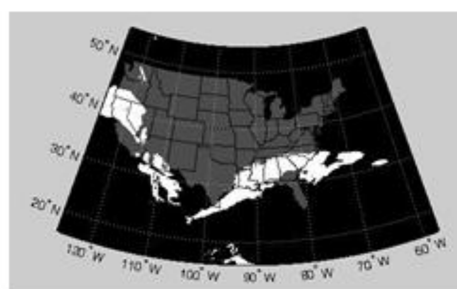
November 15th 2014 t:10:00 FL 340



November 15th 2014 t:10:00 FL 360

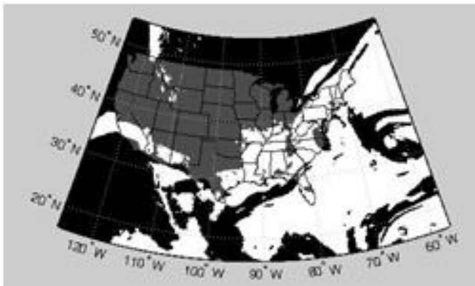


November 15th 2014 t:11:00 FL 340

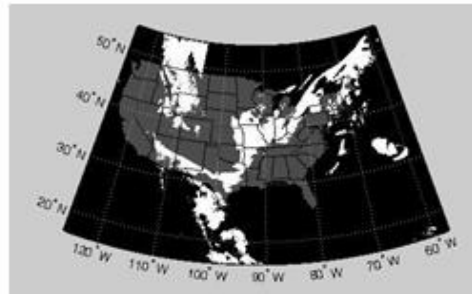


November 15th 2014 t:11:00 FL 360

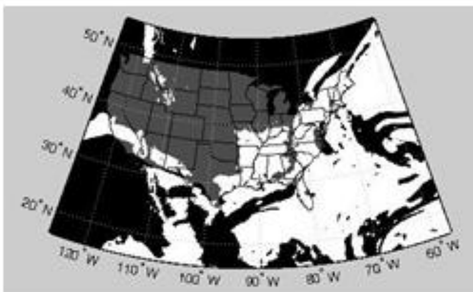
Figure 21: ISSR coverage November 2014 (%)



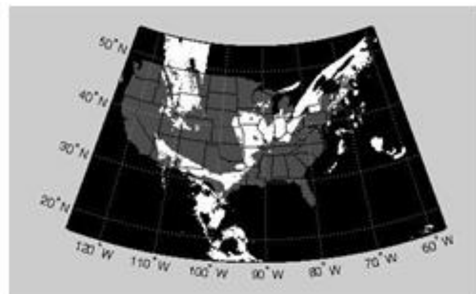
August 18th 2015 t:09:00 FL 340



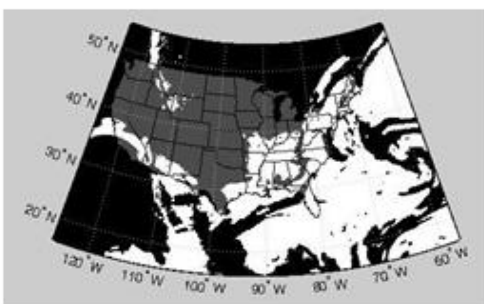
August 18th 2015 t:09:00 FL 360



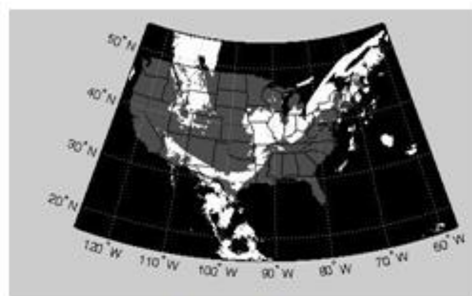
August 18th 2015 t:10:00 FL 340



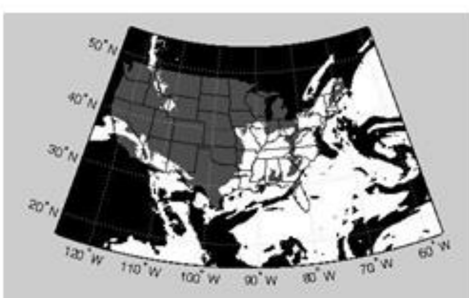
August 18th 2015 t:10:00 FL 360



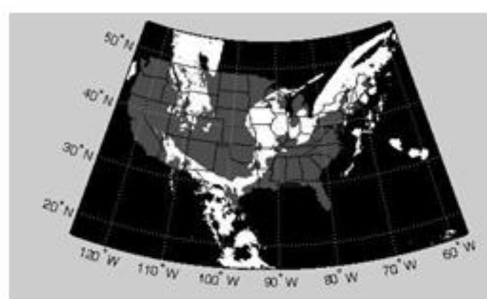
August 18th 2015 t:11:00 FL 340



August 18th 2015 t:11:00 FL 360



August 18th 2015 t:12:00 FL 340



August 18th 2015 t:12:00 FL 360

Figure 22: ISSR coverage August 2015 (%)

Figure 23 provides a snapshot of the ISSR regions on March 1st 2015. The weather is sampled at Z=29 time 0:00-23:25

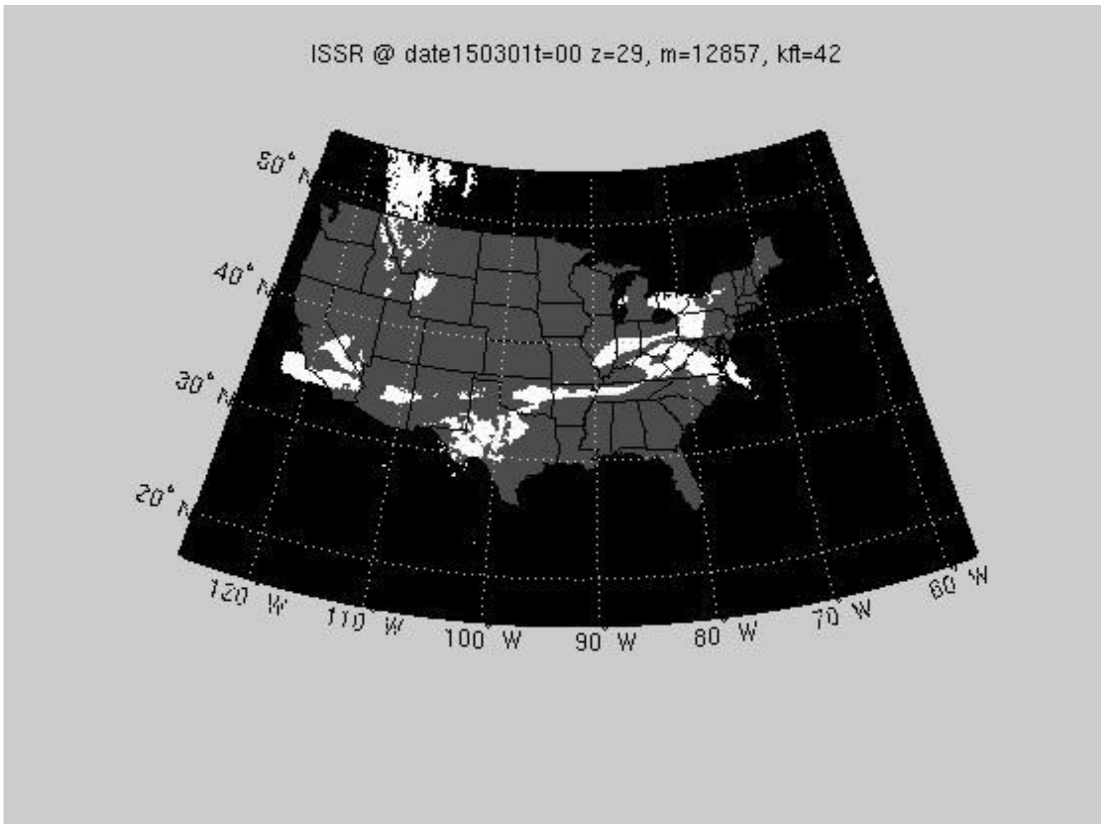


Figure 23: ISSR on March 1st 2015

Ice Super Saturated Region (ISSR) Statistics

Weather data from, November 9 2014 to October 2nd 2015, was analyzed. The weather data was downloaded from the National Oceanic and Atmospheric Administration (NOAA) Rapid Refresh

Products (RAP) weather files (http://www.nco.ncep.noaa.gov/pmb/products/rap/). Relative Humidity and Temperature were established for the 151,987 CONUS cells for each Flight Level, from FL200 to FL400.

Ice Super Saturated (ISS) conditions were present on average for each day in 7.4% of the CONUS cells with standard deviation 1.5 % of the cells (Table 15). During daytime, the CONUS cells with ISS conditions were normally distributed with an average and median of 6.2% of the CONUS with a standard deviation of 1.2%. During nighttime, the CONUS cells with ISS conditions were also normally distributed with an average and median of 8.7% with standard deviation of 1.6%. Due to diurnal effects, the nighttime exhibits an average of 2.5% more CONUS Cells with ISS conditions. The percent of the CONUS cells that exhibit ISS regions is consistently higher in the Summer months than other months of the year.

Table 15: ISSR Statistics for 365 days of weather

Statistic	**Daily Total**	**Distribution Description**	**Day Total**	**Night Total**	**Day/Night Ratio**
Daily Average ISS CONUS Cell Count	**μ =178,516 (7.4%) Std Dev = 33,796**	Normal Distribution Mean = 151,442 (7.4%) Std Dev = 29,745 (1.5%)	μ = 151,442 (6.2%) Std Dev = 29,745 (1.2%)	μ = 211,167 (8.7%) Std Dev = 38,316 (1.6%)	**0.72**

The “Summer” months running from May to October, and exhibited a higher percentage of ISS Conus cells than the “winter” months (Figure 24). The average percent of the CONUS with ISS Cells in the “summer” was 8.4% with an average in the “winter” of 6.3% (significant at $p < 0.05$).

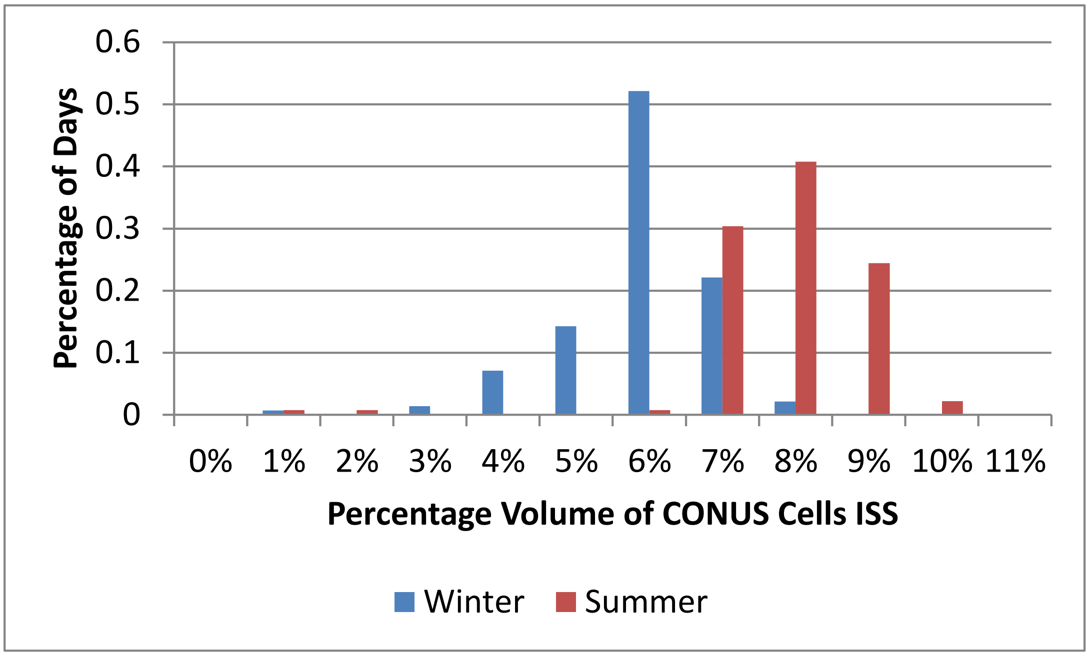


Figure 24: Percent of CONUS Cells between FL200 and FL400 that exhibit Ice Super Saturated Conditions in “summer” and “winter” months.

4.1.2 Flight Trajectories

One representative day of CONUS flights recorded by the ADS-B transmissions, June 20th 2016, was matched against daily hourly weather data for 267 days from November 9 2014 to October 2 2015. Contrail generation on all aircraft in the US air space was calculated. A comparison is presented by aircraft manufacturer. The two largest manufactures showed very little variation between. Boeing and Airbus aircraft generated contrails during 13.47% and 13.48% of their traveled trajectories respectively.

Manufacturer	Count of Aircraft	Contrail Loc	Total Loc	Contrails [km]	Pct Contrails per flight
Boeing	2,911	10,748	79,717	139,724	13.48%
Airbus	1,623	6,612	49,078	85,956	13.47%
Embraer	870	2,984	23,740	38,792	12.57%
Bombardier	714	1,881	17,408	24,453	10.81%
McDonnell Douglas	343	1,286	8,870	16,718	14.50%
Other	470	961	6,431	12,493	14.94%
Grand Total	**6,931**	**24,472**	**185,244**	**318,136**	**13.21%**

4.1.2.1 Flight Trajectory Statistics

The sample day of flights, downloaded from the ADS-B Exchange (https://www.adsbexchange.com/), had 30,813 flights that traveled through the CONUS. Flight trajectories of less than 50 nm and flight trajectories with missing data were eliminated, leaving 24,095 flights (Table 16).

These flights generated 24.3M nautical miles along-track distance. Sixty nine percent of the along-track distance were generated during the day.

Table 16: Summary Flight Trajectory Statistics for June 20th 2016

Statistic	Daily Total	Day Total	Night Total	Day/Night Ratio
Flight Count	24,095	13,628	10,467	56.56% Day
Flight Trajectory Along-track Distance	24,363,179 nm	16,553,906 nm	7,809,273nm	2.12 68.96 % Day
Flight Trajectory Count CONUS Cells	316,495 CONUS cells	218,254 CONUS Cells	98,241 CONUS Cells	1.72 63.27% Day

The distribution of total along-track distance for each flight exhibits an exponential distribution with a mean of 952 nautical miles and a median of 531 nautical miles.

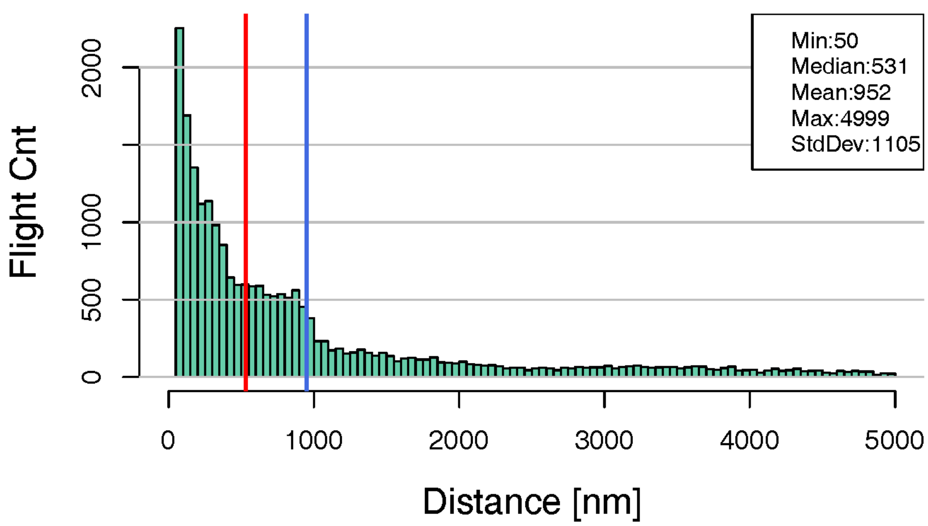


Figure 25: Flight Distance Along Track

The flight trajectories traversed 316,495 CONUS cells. Sixty three percent (63%) of the CONUS cells were traversed during daylight hours. The distribution of CONUS cells traversed is Exponential with median of 10 and mean of 13.3 CONUS cells.

The flights operated with Cruise Flight levels ranging from FL200 to FL 400. The Mean and median Cruise Flight Level was FL 350. Fifty-five percent (55%) of the flights had Cruise Flight levels between FL340 and FL360.

Flights are aggregated by summarizing to every 5° both in latitude and longitude and displayed in Figure 26 to gain a geographic distribution of flights. To prevent duplication, such as cases where an aircraft was recorded in the same section multiple times, unique flights are counted.

Number of Flights Identified by region

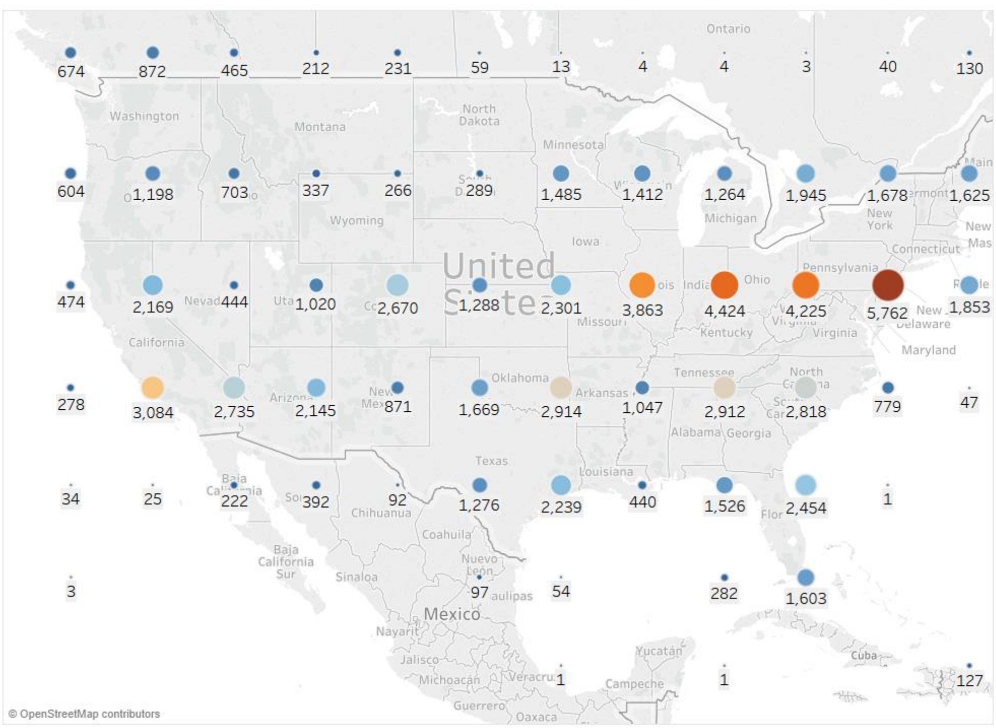


Figure 26: Geographic distribution of flights over the US Air Space.

4.1.3 Flight Trajectory Intersections with ISS CONUS Cells

4.1.3.1 Contrail Generation and persistence evaluation

Using the locations identified in the first process, contrail persistence is determined by evaluating each location at the time the aircraft passes through and over the four hours after. The contrail will be extended at each location during that time so long as the conditions for ISSR remain uninterrupted. As noted in section 3 weather cells used in the analysis all meet contrail generation conditions, therefore the match of a trajectory location and weather data implies contrail generation. While weather conditions continue the contrail is persisted for up to five hours. Contrail width and optical depth are specified in Table 11.

The sample day of 24,095 flights in the CONUS traversed a total of 242,704 CONUS cells. The average percentage of flights that traversed ISS CONUS Cells each day was 22.3% with standard deviation of 6.9% (Table 17). Fifty eight percent of flights that traversed ISS CONUS cells were daytime flights.

On the average day, the flights in the CONUS ISS cells generated 57,472 along-track nautical miles of contrails. The contrail along-track distance is above the annual average in the Summer months. The daytime contrail along-track distance is on average 21% greater than the nighttime contrail along-track distance.

The average contrail along-track distance (ATD) for flights that traverse a CONUS cell with ISS conditions is 9.3 nautical miles.

Table 17: Flight Trajectory Intersections with ISS CONUS Cells

Statistic	Daily Total	Day Total	Night Total	Day/Night Ratio
Flights that traversed ISS CONUS Cells	μ = 6,160 (22.3%) Std Dev = 1,931 (6.9%)	μ = 3,593 (12.99%) Std Dev = 1,034	μ = 2,567 (9.28%) Std Dev = 913	58.32% Day 1.47
Along-track Distance of Contrails Generated (nm) per day	μ = 57,472 (0.3%) Std Dev = 25,186	μ = 34,785 Std Dev = 14,386	μ = 22,879 Std Dev = 10,898	1.58

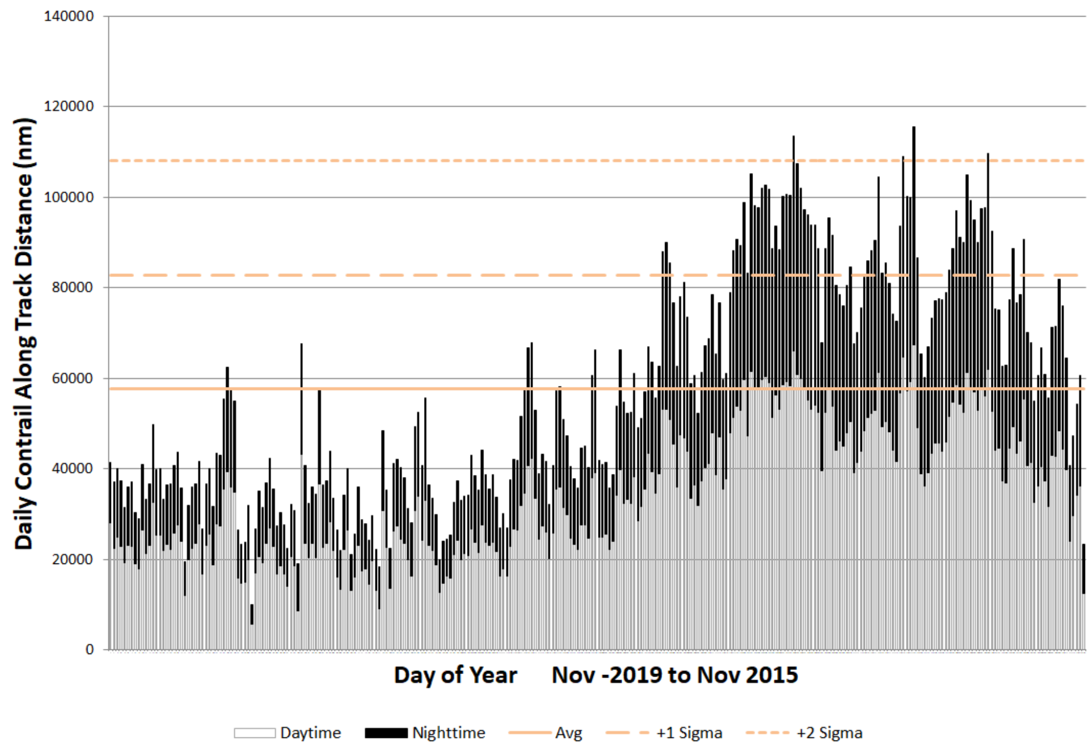


Figure 27: Daily Contrail Along-track Distance

The daily contrail along track distance increases linearly with the increase in the percentage of the CONUS cells that have ISS conditions (Figure 28). A linear fit is found at Contrail ATD = $1X10^6$(Conus ISS) - 43697, with an R^2 of 0.6305. This suggest that a small increase in percentage of ISS CONUS yields a larger increase in the contrail along-track distance.

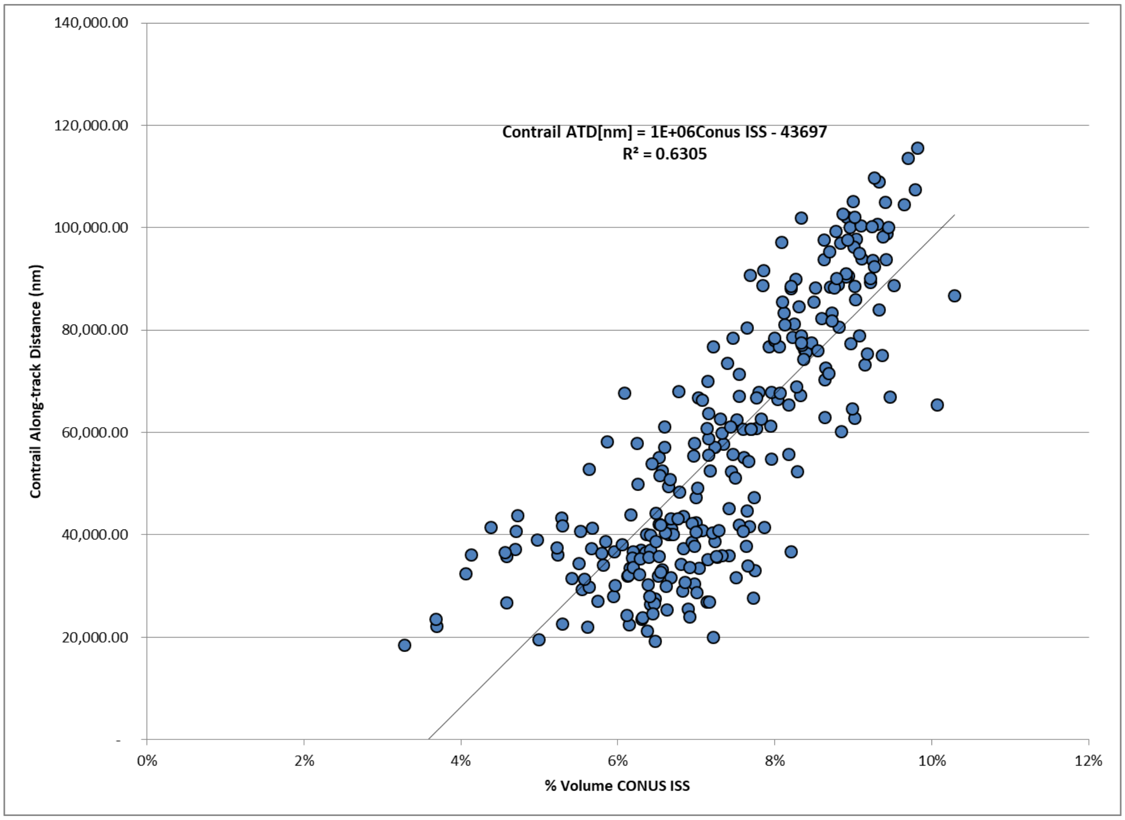


Figure 28: Percentage of daily ISS CONUS cells increases, the Contrail Along-track Distance increases exponentially

4.1.3.2 Net Radiative Forcing Statistics

The Net Radiative Forcing (NRF) is calculated as the sum of

- the outgoing longwave radiation reflected back to the Earth during the day (i.e. warming)
- the outgoing longwave radiation reflected back to the Earth during the night (i.e. warming)
- the incoming shortwave radiation reflected back out to space (i.e. cooling).

For the year, the estimated total NRF generated by the contrails had a net warming effect of magnitude 206 W/m^2. This includes an annual albdeo cooling during the day of 0.670 W/m2, warming of 2.1 W/m2 and 1.4 W/m2.

The estimated daily average was 7.08 mW/m^2 with a median 6.44 mW/m^2 (Table 18, Figure 29). The NRF exhibits a high degree of variance with a standard deviation of 3.59 mW/m^2 which is roughly half of the mean (i.e. coefficient of variation is 0.5). Since the analysis was conducted with the same daily flight schedule, the variance is derived from volume, altitude, and geographic location of the ISS regions.

Table 18: Daily Estimated Net Radiative Forcing

Statistic	Daily Total	Day Total	Night Total	Day/Night Ratio
Daily Estimated Net Radiative Forcing (mW/m^2)	μ =7.08 Std Dev =3.59	μ = 4.01 Std Dev = 1.99	μ = 3.07 Std Dev = 1.62	1.31 56.67% Day

The NRF in the U.S. is above average in the Summer months and below average in the Winter, Spring and Fall. There are 9 days in this period in which NRF exceeds 2 sigma.

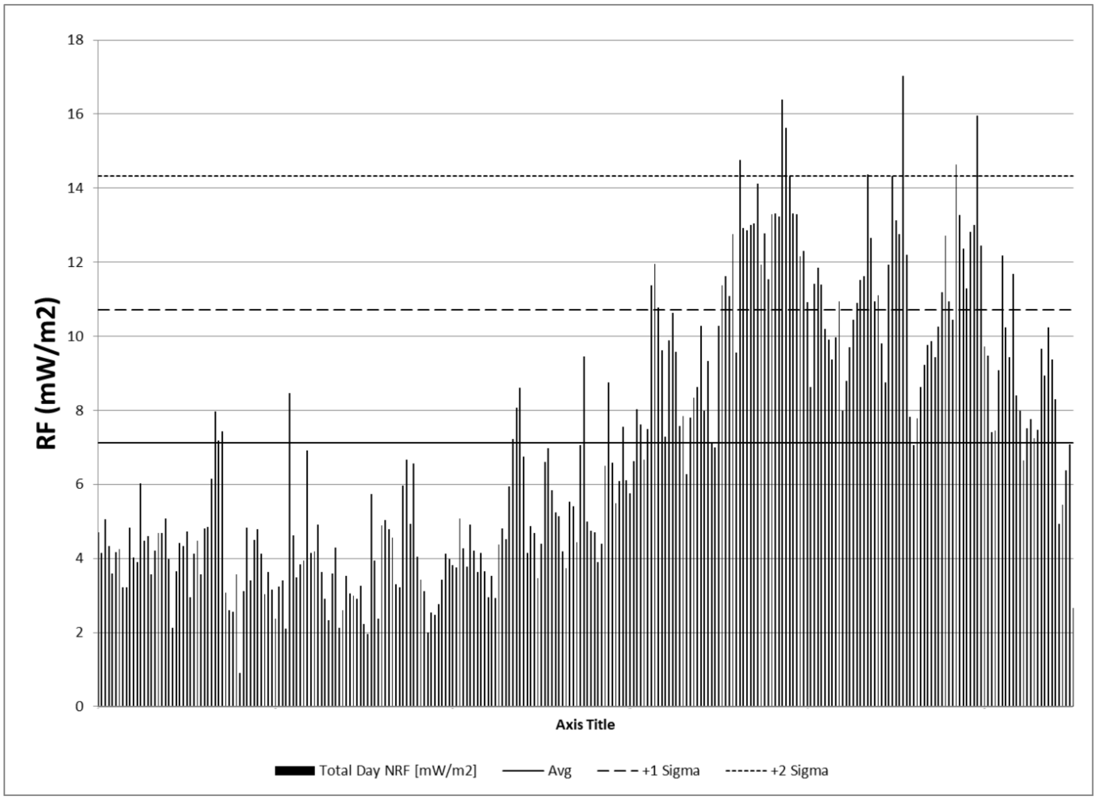


Figure 29: Daily average Net Radiative Forcing

The Daily Contrail Along Track Distance is an exponential function correlation with % Volume of CONUS Cells with ISS Conditions, NRF = $0.6423e^{15.238\% \text{ CONUS Cells}}$ ($R^2 = 0.6392$). The NRF is directly proportional to the Daily Contrail Along Track Distance:

$$\text{Daily NRF } (mW/m^2) = (0.0001 * \text{Contrail ATD (nm)}) - 1.0547$$

$$R^2 = 0.9876$$

As a consequence of the proportional relationship between NRF and Contrail ATD, the NRF is correlated non-linearly the % Volume of CONUS ISS Cells:

$$\text{Daily NRF } (mW/m^2) = 0.9706e^{(25.121 * \% \text{ CONUS ISS})}$$

$$R^2 = 0.473$$

The average NRF per flight is 0.0003 mW/m^2. The average NRF per contrail flights is 0.00115 mW/m^2.

The average NRF per along-track distance nautical mile is 2.9 x 10-7 mW/m^2. The average NRF per contrail along track distance nautical mile is 0.00012 mW/m^2.

4.2 ALTERNATIVE CRUISE FLIGHT LEVEL EVALUATION.

The ISSRs can be avoided by either flying around them or flying either under or over. Flying around ISSRs was deemed not practical. After visually analyzing the weather data, we find that even during days where the conus has a low ISSR volume count, the average width of an ISSR over the U.S. air space ranged from 100 to 250 nm. Figure 30 presents the coverage at a frequent flight level, during a low ISSR season. The image provides a view at flight level 360 at 9:00 AM on November 15th 2014. The area covered by these regions suggest it would not be efficient to fly around. A route around the ISSR at the same flight level would add over 100 nm to the trajectory. This view is in line with recent publications where rerouting traffic from Mexico City to Frankfurt (MMEX – EDDF) generated a detour of 367 nm (Rosenow 2018).

This research will focus on options to fly above and below the regions.

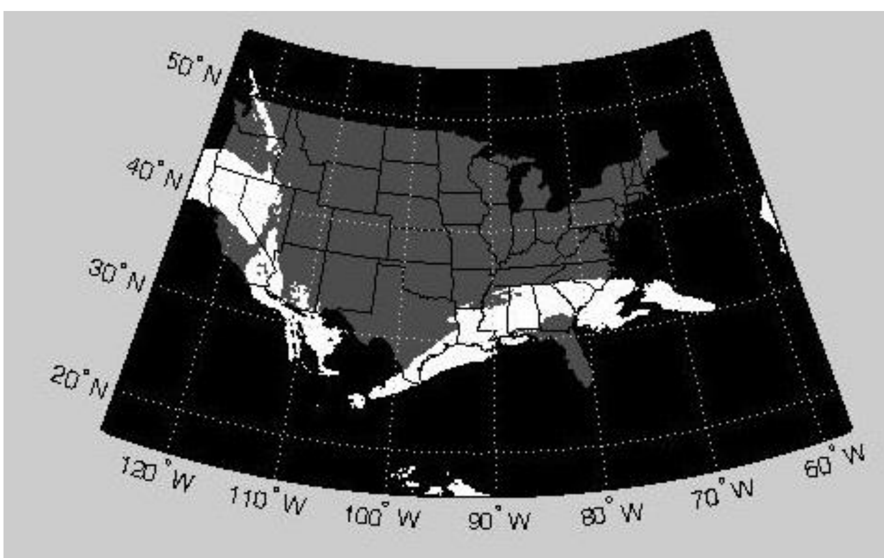


Figure 30: ISSR at flight level 360 at 9:00 AM on November 15th 2014.

The analysis uses the same set of flights used in Section 4.1. The selected flights are simulated through alternative cruise flight levels of the weather to determine the contrail generation and impact. Flights paths assume a 3° climb and descent. Additionally the following considerations are taken:

- Flights are selected to include only those with a complete trajectory during the sample time span and which provide both a departing and arriving airport within the USA.
- The distance of the flight is calculated along the recorded track. Any flight that did not present sufficient data points to provide a full trajectory when compared to its Great Circle distance was excluded.
- During the typical flight trajectory, a flight often modifies its cruising altitude. For the purpose of this case study the *original cruise flight level* for each flight is determined as the mode of the recorded flight levels (excluding all records under FL 200).
- When evaluating a flight for alternative flight levels the evaluation is done assuming the cruise flight level is constant, e.g. the flight climbs to the alternative flight level and remains there for the duration of the flight. This standard flight approach provides similar conditions for all flights so the difference in fuel burn is driven by distance and altitude only. After applying these restrictions, a set of 802 flights were obtained.

4.2.1 *CONTRAIL GENERATING FLIGHT STATISTICS*

To provide context for contrail formation Figure 31 presents the distribution of the ISSR by flight level. The high frequency of ISSR around the FL 350 has an impact on contrail formation. The figure suggests that as flights go above FL360 they will encounter less of ISSRs in their path.

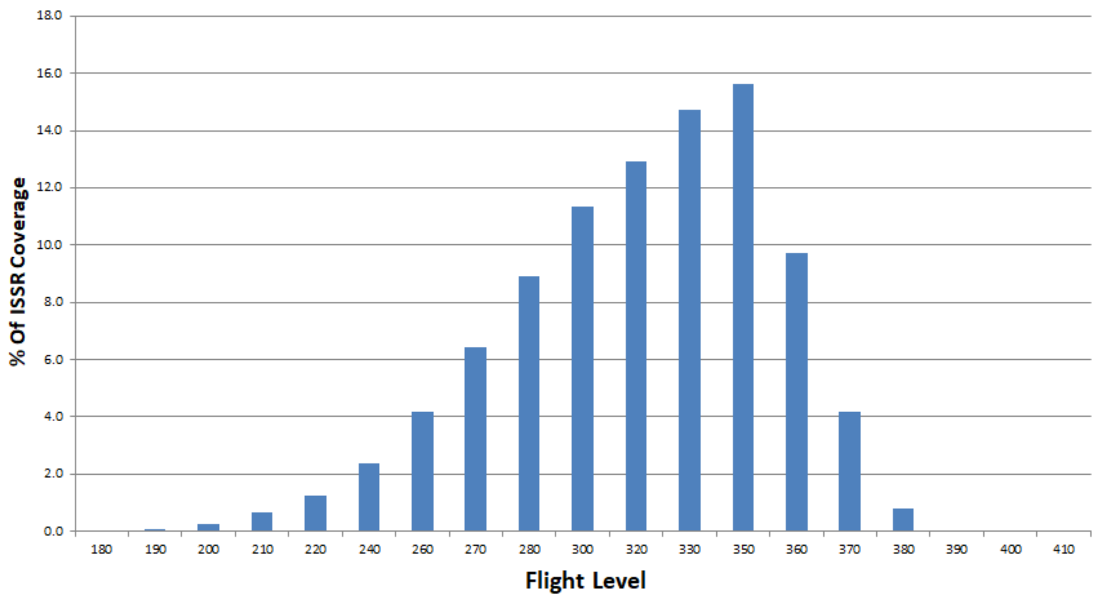


Figure 31: Average ISSR coverage by Flight Level 2014 - 2015 (%)

To locate the flights within the ISSR the distribution of contrails by original cruise flight level is shown in Figure 32. These are the altitudes where contrails were recorded from the original sample of flights.

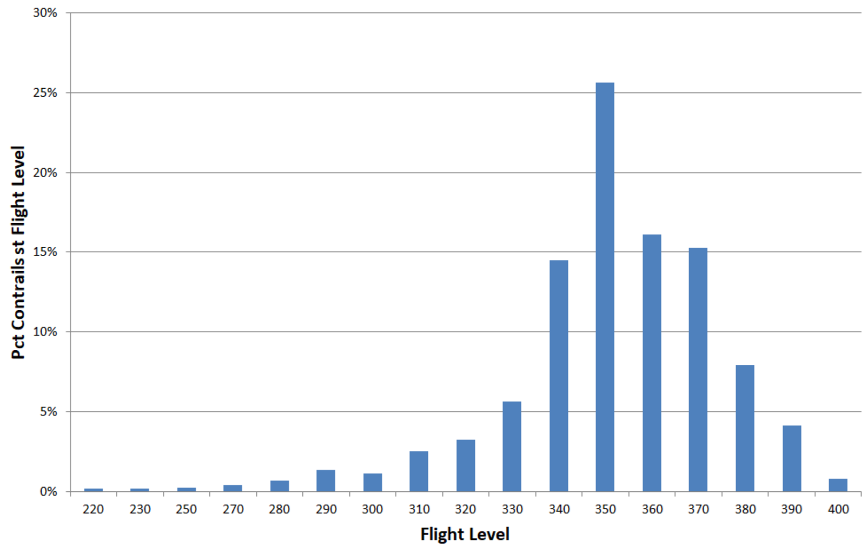


Figure 32: Distribution of Contrail generating flights by Actual Cruise Flight Level

To complete the context, Figure 33 presents contrails identified at all flight levels along the original path. Because flights often change altitude during their trajectory a single flight can present contrails at multiple altitudes. This set of altitudes reflects all flight levels recorded for flights, including but not limited to their cruise flight level.

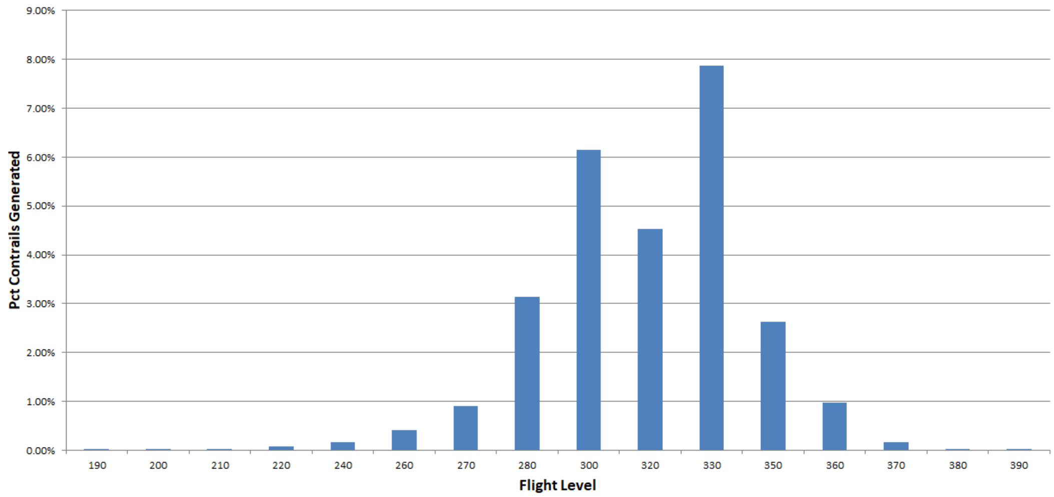


Figure 33: Distribution of contrail generation at all flight levels used along path.

The simulation assigns to each flight, its original cruise level and alternative levels. In each case the simulation climbs to the assigned fight level and remains there for the duration of the cruise phase before descending. The flight is then evaluated for contrail generation and fuel burn. After the initial cruise level, the flight is reproduced using alternative flight levels in 2,000 ft. increments up to 8,000 ft. both above and below the original cruise level. Additionally alternative cruise levels are limited to a floor and ceiling of FL 200 and FL 400 respectively. The analysis found that elevating the flight level by 2000 ft. resulted in an average decrease of 51.6% of contrail generating flights per day.

A summary of the alternative flight level evaluation is provided in Figure 34. At their original flight level; when "*Change in Flight Level*" = *0*, the graph shows that 17% of flights generated Contrails. The analysis showed that decreasing the flight level by 2000 ft. increased the number of flights generating contrails from 17 to 33%; however increasing the FL by 2000ft decreased the percentage of flights to 8.2%.

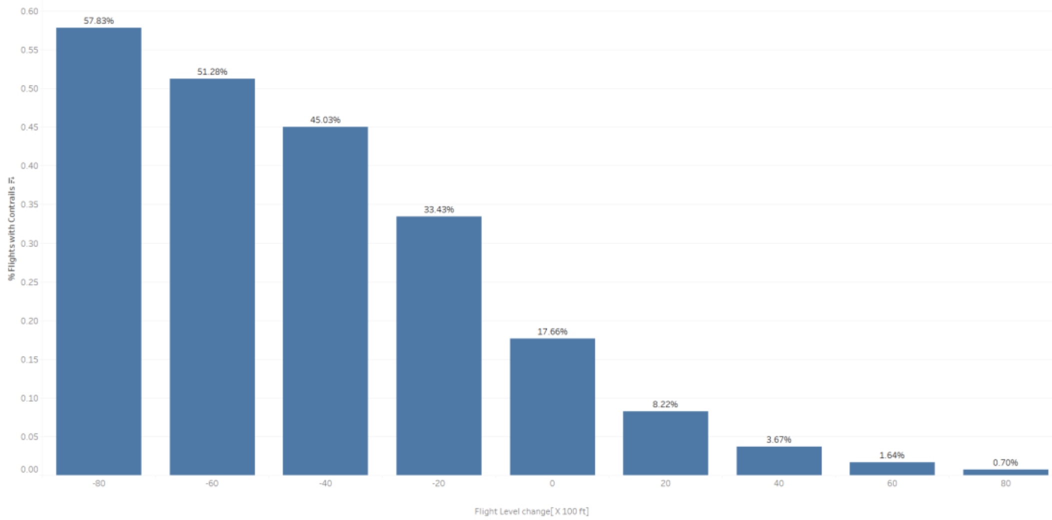


Figure 34: Percentage of flights generating contrails as cruise flight level varies.

Each flight was evaluated throught the year while reducing and increasing their flight level in increments of 2000 feet. The summary of the results is provided in Table 19. The table provides the percentage of flights at each flight level arraged by alternative flight level (verticaly) grouped by their original flight level (horizontally).

The middle row provides the distribution for the original flight level “0 – Original Cruise FL” with alternative levels above and below on the vertical axis. The table shows that decreasing all flights by 8000 ft (Flight Level -80) increased the contrail frequency to 57.98% while increasing the flight level by 8000 ft decreased the contrail frequency to 0.71%.

Table 19: Contrail generation as flight level varies as a percentage of total flights (x 100ft)

	Original Cruise FL																				
Flight Level	**200**	**210**	**220**	**230**	**240**	**250**	**260**	**270**	**280**	**290**	**300**	**310**	**320**	**330**	**340**	**350**	**360**	**370**	**380**	**390**	**400**	**Total**
-80									0.05	0.06	0.16	0.29	0.71	1.85	13.33	16.65	8.80	8.17	5.40	1.67	0.85	**57.98**
-60							0.02	0.02	0.07	0.10	0.31	0.51	2.78	5.43	8.50	8.48	8.90	7.05	7.88	1.13	0.20	**51.40**
-40					0.02	0.01	0.03	0.02	0.13	0.18	1.26	1.88	1.90	3.19	8.47	7.31	13.06	4.87	1.81	0.89	0.11	**45.14**
-20			0.03	0.04	0.03	0.02	0.07	0.05	0.50	0.62	0.88	1.20	2.09	2.98	11.89	5.02	2.93	4.09	1.01	0.07		**33.51**
0 - Original Cruise FL		0.03	0.04	0.05	0.06	0.04	0.21	0.20	0.35	0.40	0.91	1.20	3.35	2.22	2.49	4.03	1.64	0.44	0.03			**17.70**
20	0.01	0.02	0.07	0.09	0.27	0.14	0.10	0.15	0.36	0.37	1.31	0.86	0.79	1.94	1.27	0.43	0.05					**8.24**
40	0.02	0.04	0.23	0.29	0.26	0.09	0.09	0.15	0.53	0.26	0.22	0.77	0.44	0.21	0.05							**3.68**
60	0.09	0.13	0.14	0.20	0.33	0.09	0.11	0.10	0.09	0.20	0.06	0.09	0.01									**1.65**
80	0.09		0.11		0.35	0.06		0.04	0.02	0.02												**0.71**

The table shows the highest frequency at FL 350 which matches the preference in flight levels and ISSR density making contrails more likely.

Based on these results the focus of the analysis is centered on increasing the flight level to either 2000 or 4000 ft.

Change of contrails generated per year

To understand the effect of increasing the flight level a more detailed view is provided in Figure 35 showing the daily change in percentago of flights with contrails. The graph shows that by increasing the flight level by 2000ft there is a decrease in flights with contrails with a mode at 60% over 42% of the eyar.

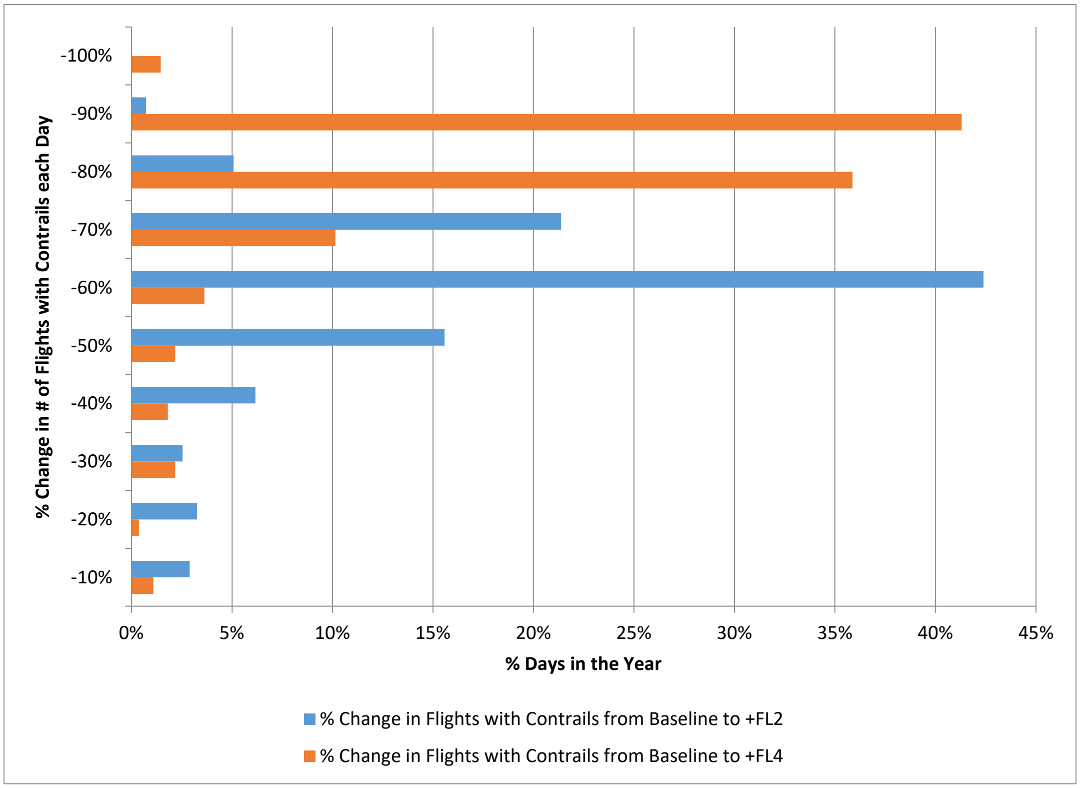


Figure 35: Change in flight generating contrails per year when increasing flight level

A further increase to 4000 ft. would decrease the days with contails by 80 – 90 %.

Impact to contrail ATD and persistence.

The analysis uses the weather information during the flight and the hours after the flight has crossed the region to determine the contrail generation and persistence. Tables 20 and 21 provide statistics for the nautical miles and nautical miles – hour of contrails generated per flight. In average flights generated 14.6 nautical miles of contrails per flight; this number could potentially be reduced to 12.9 nm, a reduction of 11.2% by raising all aircrafts by 2,000 ft. Taking into account the effect of persistence; the contrails-hr show a reduction from 35.7 to 30 nm hr a reduction of 14.8%.

Table 20: Contrails generated per flight as Flight Level varies in Nautical Miles

	Change in Flight Level (X100ft)								
	-80	-60	-40	-20	0	20	40	60	80
Min	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0
Median	20.2	21.2	20.2	16.7	13.8	11.7	11.3	11.0	12.2
Mean	21.4	22.5	21.6	18.4	14.6	12.9	12.6	12.1	13.3
Max	55.4	63.3	59.8	56.2	46.4	44.9	42.1	42.1	29.2
Std Dev	7.5	8.4	8.8	8.2	6.5	6.1	6.1	5.6	4.9

Table 21: Contrail and Persistence per flight by Flight Level in Nautical Miles – Hour

	Change in Flight Level (X100ft)								
	-80	-60	-40	-20	0	20	40	60	80
Min	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0
Median	55.9	60.0	55.8	43.0	28.6	23.4	21.6	21.8	30.4
Mean	61.2	65.3	61.1	49.4	35.7	30.4	28.6	28.3	35.2
Max	199.7	242.4	221.9	188.5	178.0	145.8	135.1	91.3	85.2
Std Dev	33.0	35.8	35.0	31.6	25.1	22.7	21.9	20.8	22.8

The change in daily contrails along track distance is presented in Figure 36. The figure shows how increasing the FL by 2000 ft will provide a reduction of 50% in daily contrails over 40% of the year. A further increase of 4000 ft would drive a reduction in contrails over 50% of the year of 80%

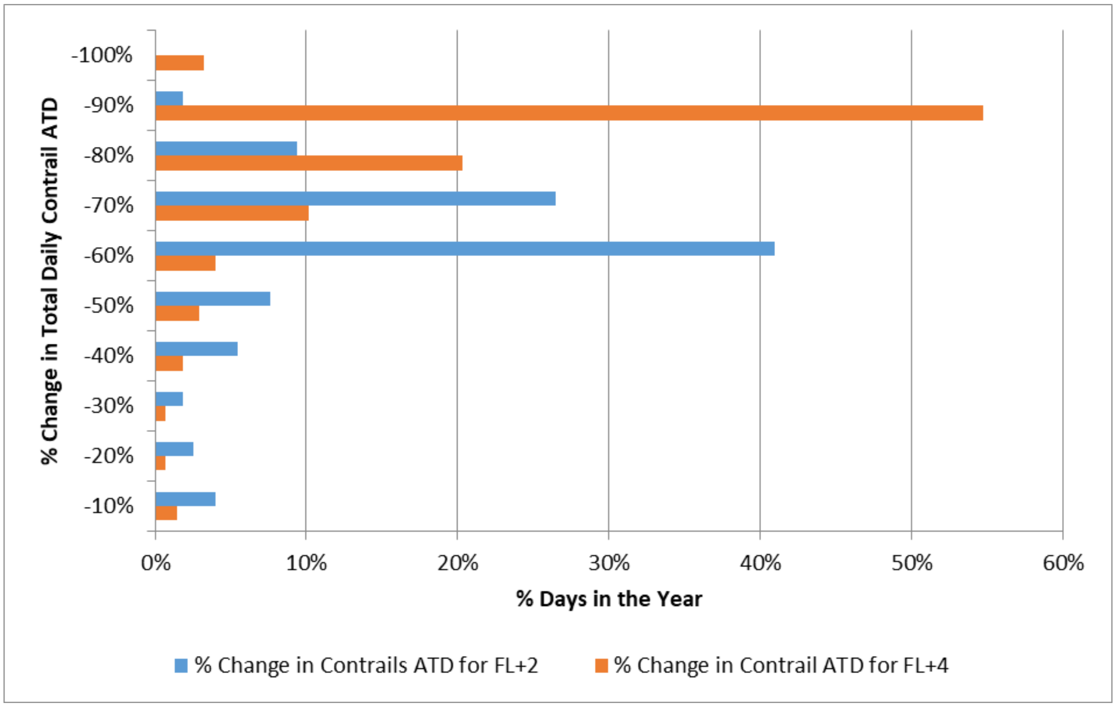


Figure 36: Change in daily contrail ATD

Last the contrails are persisted over time in Figure 37 reflecting a larger decrease in persisted contrails. Increasing the average flight level by 4000 ft would eliminate 80 % of the contrail-persistence however a change to 2000ft eliminates 50% of the contrails and their persistence.

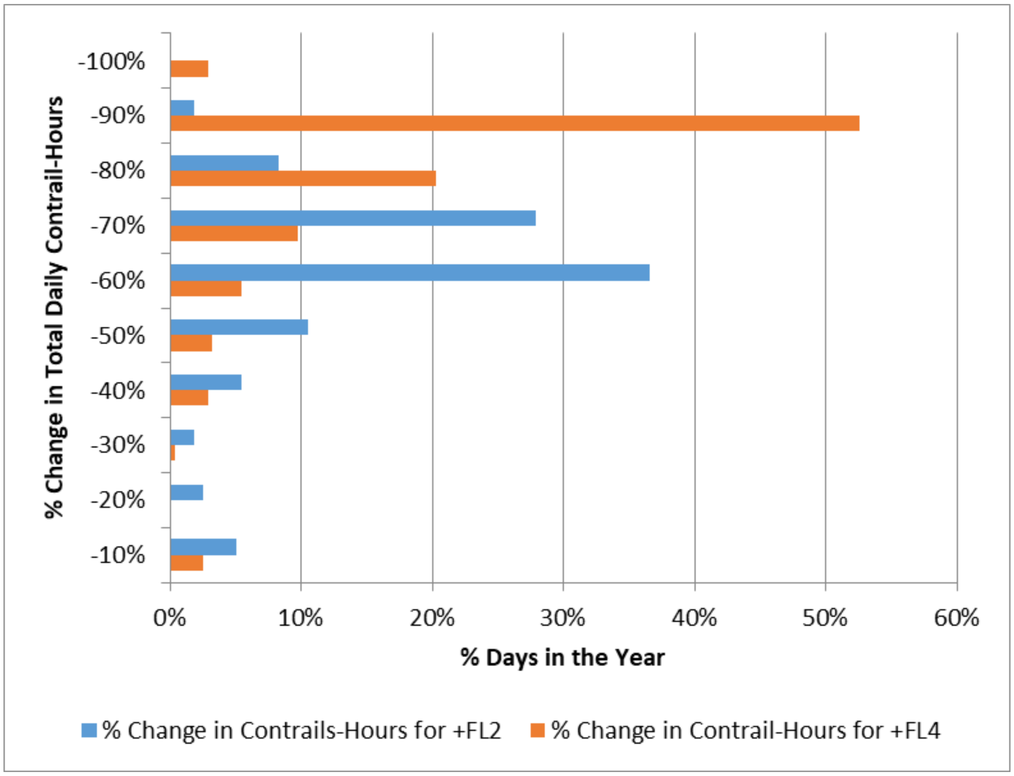


Figure 37: Distribution of daily contrial - hours

An analysis by flight for both the linear Nautical Miles and Nautical Miles – hour is provided in Figure 38. The graph shows that at lower levels the contrails will persist longer while increasing the flight level decreases the persistence.

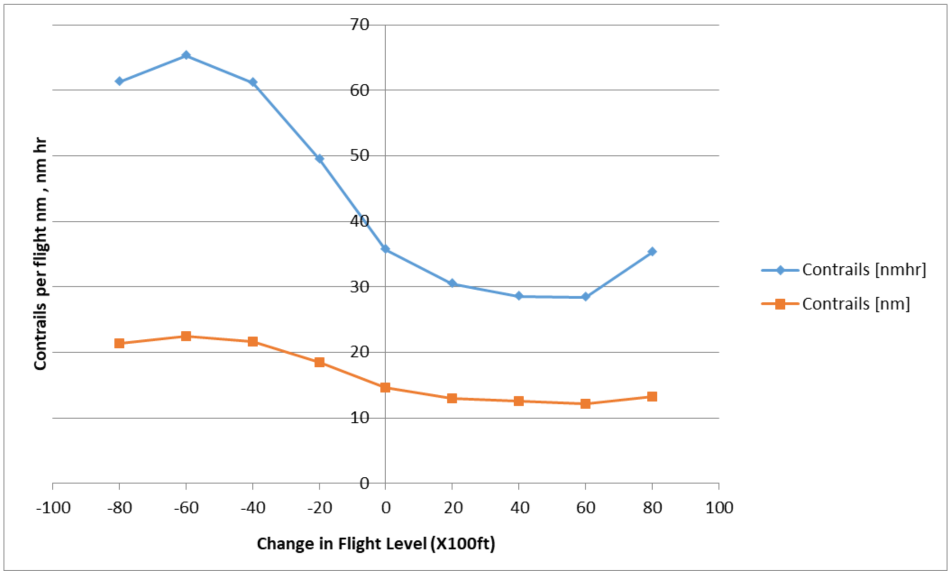


Figure 38: Comparison of linear contrails vs contrails – hour generated per flight

To obtain a geographic view of the contrail generation, the intersections of flight trajectories and ISSR are plotted over a US map. Contrails generated at their original flight level are aggregated by summarizing to every 5° both in latitude and longitude (Figure 39). To prevent duplication, such as cases where an aircraft was recorded in the same section multiple times, unique instances contrails are counted.

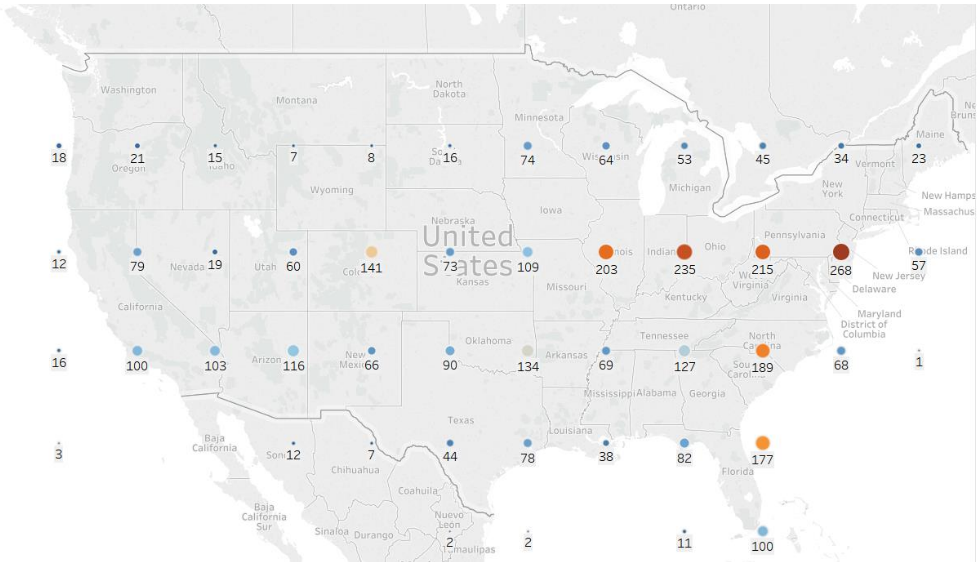


Figure 39: Count of contrails generated by location

4.2.2 NET RADIATIVE FORCING STATISTICS

The Net Radiative Forcing was estimated per flight at each flight level in mW/m^2. The change in total induced Net Radiative Forcing are grouped by alternative flight level and shown in Figure 40.

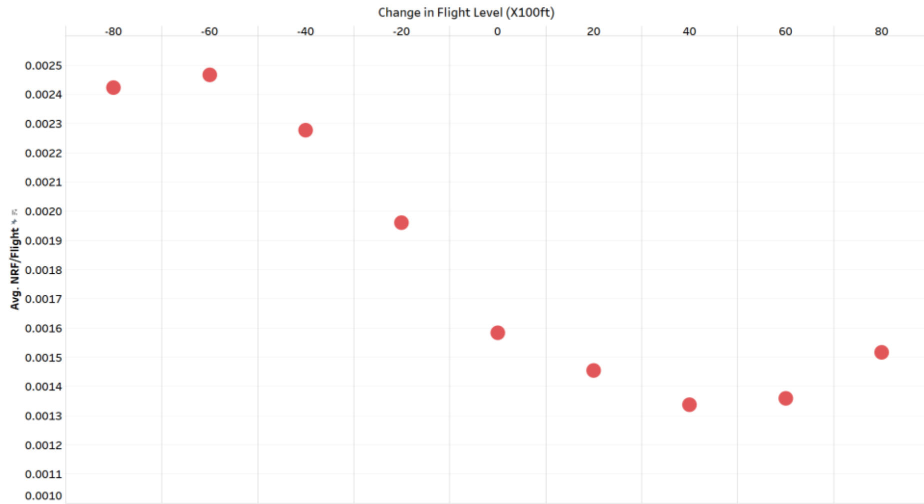


Figure 40: Radiative Forcing induced by flight as flight level varies.

The results shown correspond to the Expected Radiative Forcing (Treatment 1). In total 3 treatments were applied to the contrails to evaluate how sensitive the RF model is. The treatments modify parameters for Optical depth: τ; Optical depth above contrail: τ2; and contrail growth rate, and were set to evaluate:

1- Expected RF.
2- Rapid Crystal growth.
3- Rapid Contrail spread.

Details for the parameters are provided in section 3.1.16 in Table 11. The result of the analysis of the 3 treatments showed that both a rapid growth in Crystal and Contrail spread increased the average NRF from 0.00157 to 0.00256 mW/m2, an increase of 63% . Details are provided in Table 22.

Table 22: Mean Radiative Forcing for each Scenario by Flight Level in mW/m^2

	Net Radiative Forcing [mW/m2]								
	-80	-60	-40	-20	0	20	40	60	80
μ Treatment 1	0.00250	0.00268	0.00254	0.00211	0.00157	0.00137	0.00128	0.00125	0.00143
μ Treatment 2	0.00438	0.00471	0.00444	0.00363	0.00265	0.00228	0.00212	0.00208	0.00248
μ Treatment 3	0.00438	0.00471	0.00444	0.00363	0.00265	0.00228	0.00212	0.00208	0.00248

The expected Net Radiative Forcing is calculated per flight and indicates that increasing the average flight level by 2000 ft (+FL2) produces a decrease in Net RF per flight from 0.00157 to 0.00137 mW/m^2, this is a 12.8% per flight. The statistics for the Net RF are presented in Table 23.

Table 23: Net Radiative Forcing by Flight Level in mW/m^2

	Change in Flight Level (X100ft)								
	-80	**-60**	**-40**	**-20**	**0**	**20**	**40**	**60**	**80**
Min	0.00033	0.00033	0.00019	0.00006	0.00003	0.00003	0.00003	0.00004	0.00015
Median	0.00236	0.00248	0.00237	0.00193	0.00140	0.00112	0.00106	0.00108	0.00139
Mean	0.00250	0.00268	0.00254	0.00211	**0.00157**	**0.00137**	0.00128	0.00125	0.00143
Max	0.00767	0.01056	0.01009	0.00876	0.00637	0.00573	0.00581	0.00578	0.00348
Std Dev	0.00119	0.00131	0.00131	0.00122	0.00100	0.00097	0.00096	0.00090	0.00082
						12.80%			

The sample of flights was evaluated to determine the miminun altitude at which each flight would no longer generate a contrail. In cases where the flight level needs to be increased by 2000ft the flight is labeled as +FL2; when the flight level needs to be increased by 4000 ft the flight is labeled as +FL4.

Assuming all flight paths can be modified throughout the year to meet their +FL2 or +FL4 level; the overall effect of is evaluated shown in Figure 41. The baseline (grey) line displays the NRF induced before any action is taken. If +FL2 flights are modified the new NRF the average daily NRF decreases 37%. (+FL2 - blue). The effect obtained if both +FL2 and +FL4 flights are modified is a average daily decrease of 68% in NRF (+FL4 - orange).

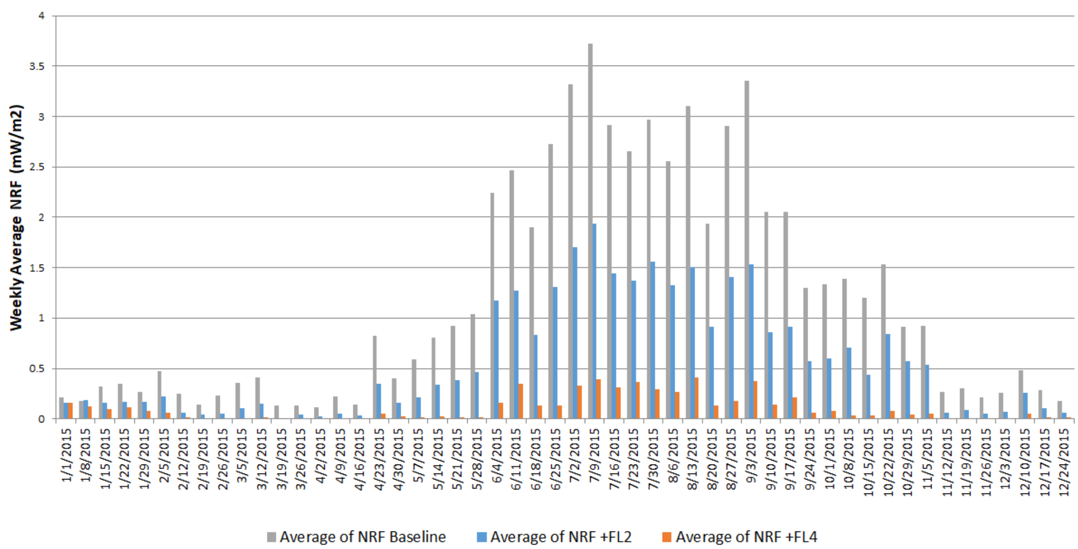


Figure 41 Weekly Net Radiative Forcing

The percentage change in NRF is calculated as a percentage of days in Figure 42. A change to +FL2 drives the Net RF decreased by 60% over almost 40% of the year while a change to +FL4 would drive the change to almost 90%.

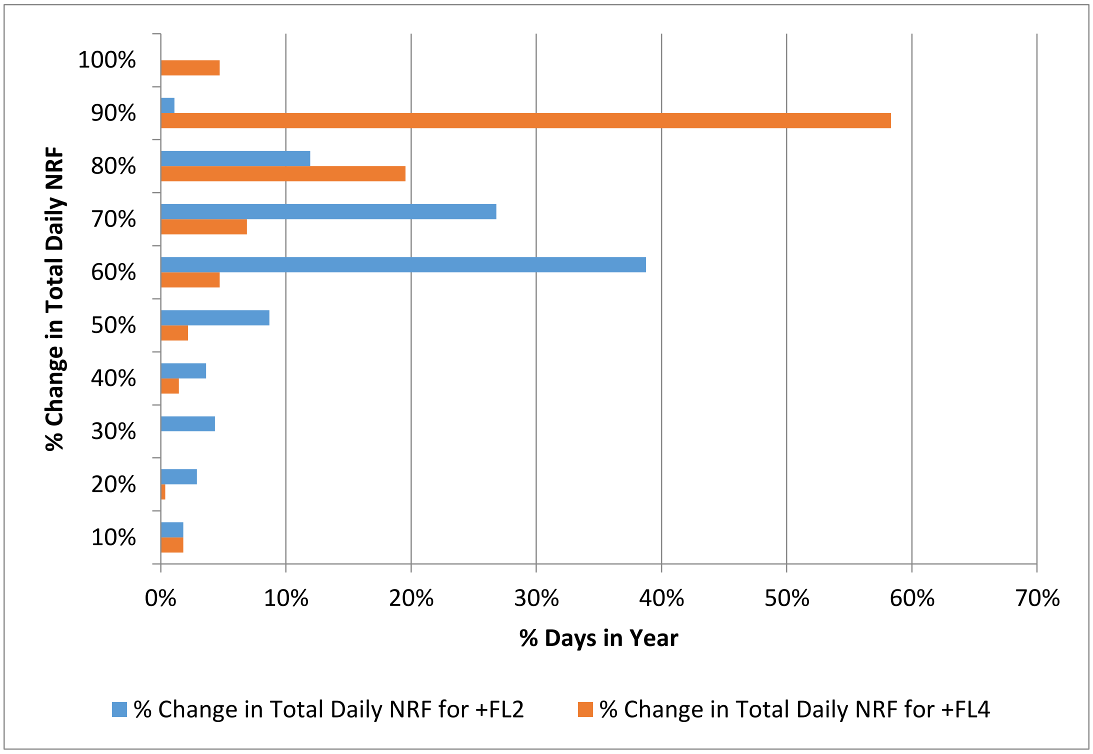


Figure 42: Change in Total Daily Net Radiative Forcing

4.2.3 FUEL BURN STATISTICS

The analysis of fuel burn follows a similar method. The original flight level is used as a baseline and as the flight level varies the average fuel consumption is compared to that at the original cruise flight level. The baseline (grey) line displays the Daily Fuel Burn induced before any action is taken. If +FL2 flights are modified the new Fuel Burn decreases 0.40%. (+FL2 - blue). The effect obtained if both +FL2 and +FL4 flights are modified is a daily decrease of 0.6% in Fuel Burn (+FL4 - orange). The effect over the year can be seen in Figure 43. Details for the change in Fuel Burn and NRF and provided in Table 24.

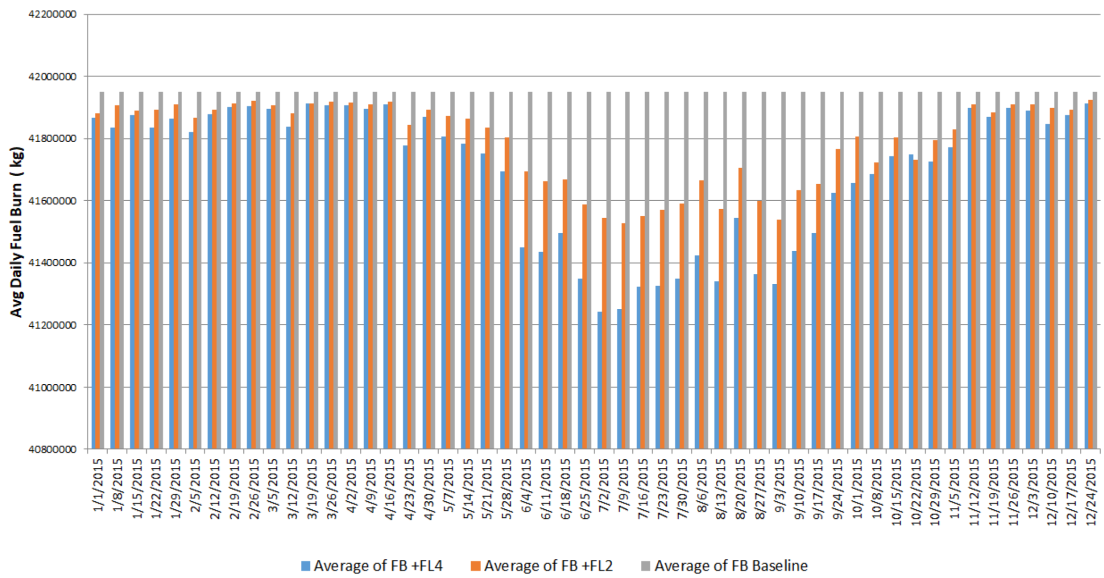


Figure 43: Change in average Fuel Burn due to change in Flight Level [kg]

Table 24: Statistics for change in fuel burn by Flight

Day	FB Baseline	NRF Baseline	FB +FL2	NRF +FL2	FB +FL4	NRF +FL4
MIN	5,992,831	-	5,914,715	-	5,870,268	-
MEDIAN	5,992,831	0.117940	5,977,217	0.040217	5,966,642	0.004891
MEAN	5,992,831	0.174944	5,969,971	0.079888	5,954,784	0.014266
MAX	5,992,831	0.863893	6,003,910	0.403080	6,001,569	0.111925
STD DEV	0	0.167868	20,250	0.088714	33,783	0.021584
% Change in Median from Baseline			0.261%	65.900%	0.437%	95.853%
% Change in Mean from Baseline			0.381%	54.335%	0.635%	90.531%

4.2.4 Fuel Burn Vs Net Radiative Forcing Statistics

Prior to comparing fuel burn to NRF. Their distributions are provided in Figures 44 and 45. The figures provide the total daily fuel burn and total daily NRF.

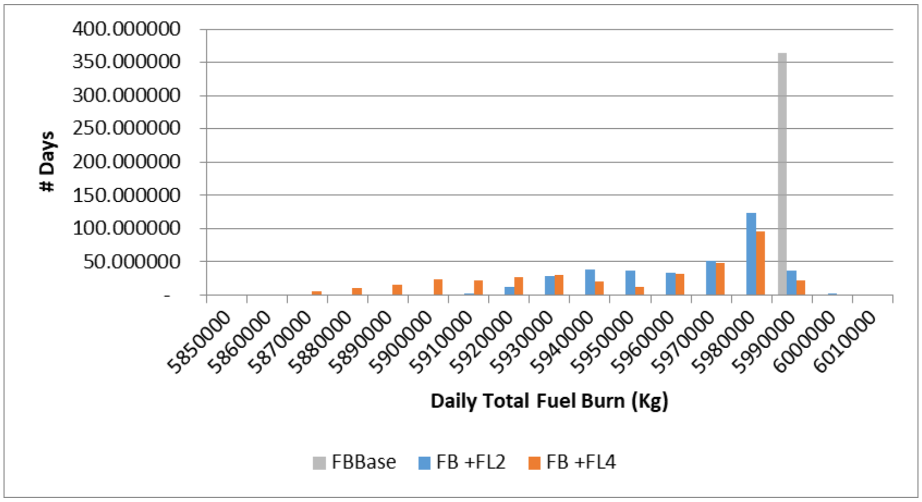


Figure 44: Histogram of Daily Total Fuel Burn

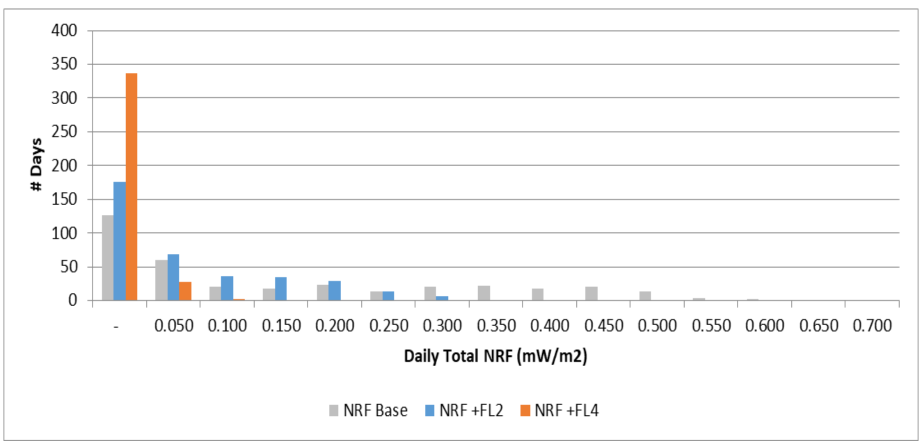


Figure 45: Histogram of Daily Total NRF

As in prior analysis the fuel burn to NRF relation is analyzed in three groups: baseline, +FL2 and +FL4. The relation of the Net Radiative forcing and fuel burn is provided in Figure 46. The figure shows a cuadratic relation where in average flying at a higher altitude would both reduce contrails and fuel burn.

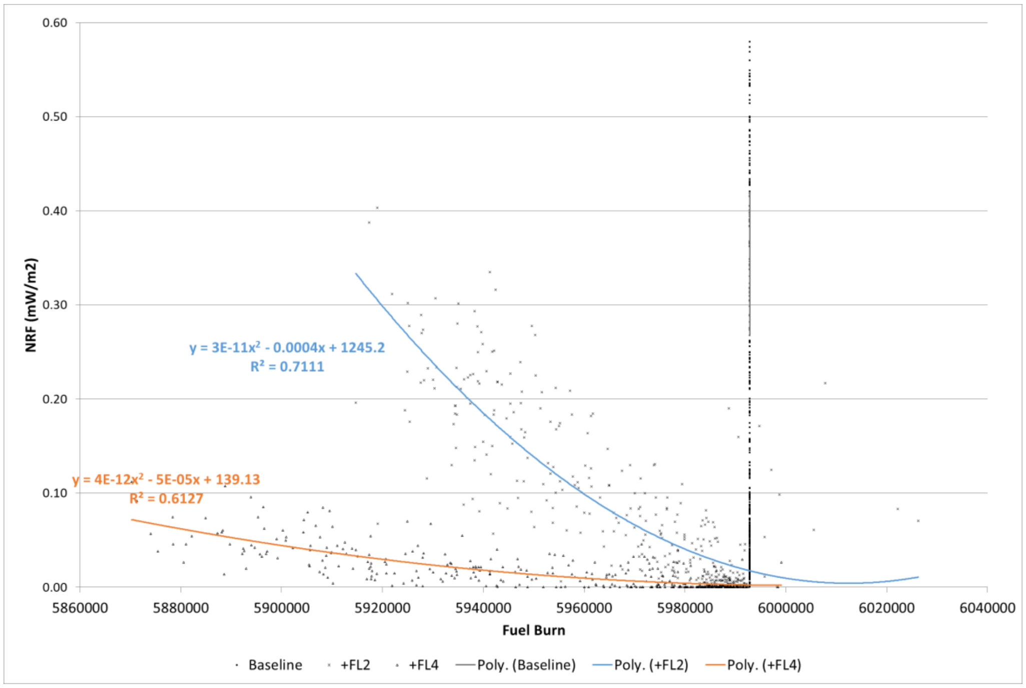


Figure 46: NRF to Fuel Burn

The relation of Fuel Burn to NRF is given by:

+FL2 - Selected flights increased flight level by 2000ft to eliminate contrails, remaining at orginial cruise flight level:

$$NRF = 4X10^{-12}FuelBurn^2 - 5X10^{-5}FuelBurn + 139.13{:}R^2 = 0.6127$$

Equation 6: FB vs NRF, +FL2

+FL4 - Selected flights increased flight level by 2000ft or 4000ft to eliminate contrails, remaining at orginial cruise flight level:

$$NRF = 3X10^{-11}FuelBurn^2 - 4X10^{-4}FuelBurn + 1245.2;\ R^2 = 0.7111$$

Equation 7: FB vs NRF, +FL4

Last the relation for Fuel Burn to NRF is aggregated by day to produce a scatterplot showing the change produced by the alteration of flight levels to +FL2 and +FL2+FL4.

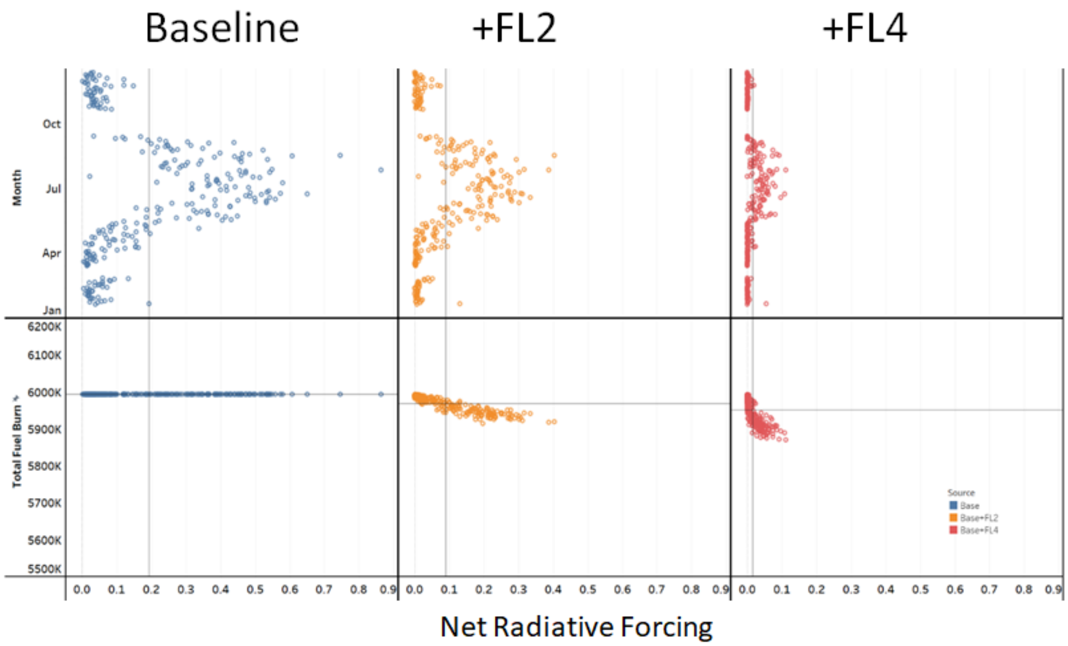


Figure 47: Daily effect to Fuel Burn and Net Radiative Forcing generated by increasing to +FL2 & +FL4

The graph shows how the reduction in frequency of contrail days for +FL2 and +FL4 drives a reduction in average NRF from 0.169 to 0.0816 mW/m^2 in the case of +FL2 and to 0.0147 for +FL4 while there is a reduction in average fuel burn of 0.39 and 0.64% respectively.

5. CONCLUSIONS

Condensation trails (aka "contrails") are high thin clouds that occur when hot exhaust gases from jet engines mix with cold, humid air. These anthropogenic clouds result in a net warming effect by capturing approximately 33% of outgoing longwave radiation emitted by the Earth, and allowing 77% of the incoming shortwave radiation from the Sun to be absorbed by the Earth and it's atmosphere. Even though contrails generate only 2% of total anthropogenic radiative forcing, they have an immediate effect on global warming. In this way, managing contrails can yield immediate global warming benefits today that can be used to buy-time for long-term CO_2 mitigations to take effect. To support "contrail management," contrails from flights will need to be inventoried. Previous models require pre-processed atmospheric data, were geographically limited, and did not take into account Sun's zenith angle, optical depth, contrail width or ice crystal size. These models were not able to analyze diurnal, seasonal, or geographic effects for a large airspace (e.g. CONUS), and could not be used in an operational context for alternate flight path evaluations.

The model presented here describes a commercially scalable method to generate an inventory of contrails in a national airspace system using publicly available weather, flight surveillance track data and models of contrail formation and net radiative forcing. The method provides the means to generate Contrail Inventories as well as perform real-time alternative flight-path analysis and overcomes the limitations mentioned above.

The method is demonstrated with a case study for the U.S. National Airspace System across 365 historic weather days. The analysis yielded a daily average of 57.4K nautical miles of contrails with an estimated daily net radiative forcing of +7.08 mW/m2. Less than 25% of the flights generated contrails on a given day, and Summer months had three times the warming effect of the Winter months. Increasing the Cruise Flight Level for contrail generating flights by 2000 feet eliminated 62% of the contrails with an average fuel savings of 2.2% per flight. An analysis on the yearly data to increase selected flights by 2000 ft (+FL2) provided a 54% decrease in average daily contrail induced NRF with a 0.90% decrease in fuel burn. Further advantages can be found when increasing the flight level of selected flights to 2000 and 4000 ft (+FL2, +FL4). These combined changes yielded a decrease in NRF of 91% and an average fuel burn decrease of 0.63%.

The methodology proposed enabled the analysis of publicly available weather and trajectory information to identify and quantify contrails and their radiative forcing. Furthermore the methodology is focused on processing only the necessary information, reducing uneeded efforts. Eliminating non-ISS weather cells from the beginning reduces the overall processing effort. Specifically the exclusion of non-critical cells reduced the required storage in a scale of 50 to 1, more importantly, it reduces the possible weather to trajectory intersections by a factor of 50. The last component is the preprocessing of data which allows to reduce the execution from matching 4 real variables: Longitude, Latitude, Altitude and time, to 3 integer variables: Loc ID, Z (altitude) and HR.

The method enabled the processing of a sample day of all aircraft trajectories in the CONUS along with a weather sample that represents all seasons which provides new insight to ISSR location, ISSR movements, contrail locations and potential mitigating strategies.

While the impact of contrails on global warming is relatively low (2% of total anthropogenic radiative forcing), contrails have the characteristic that their radiative forcing has immediate impact on the planets temperature. Whereas CO_2 emissions impact global warming on a 20 – 50 year time scale, contrails impact on the planet's temperature is immediate. In this way managing contrails could be a way to lower temperatures now to buy time to manage CO_2 emissions and other sources of global warming.

The methods described integrate the Schmidt-Appleman Criteria for Contrail Formation with the models of net radiative forcing (Schumann et.al., 1994, 2012, 2013; Burkhardt, 2011), and are guided by findings on contrail spread and growth from Freudenthaler (1995) [22] and Schröder [23]. The analysis of the ISSRs showed they can cover vast areas and the regions move resembling weather systems. While they are present throughout the year they are most frequent from June to September concentrating between FL 320 to 370 with a max density of 30% at FL 340.

Creating the inventory of contrails for the contiguous U.S. national airspace suggested that in average 7.4% of the volume of Contiguous United States (CONUS) air space between FL200 and FL400 exhibited Ice Super Saturated (ISS) conditions. As a consequence, on the average day 6,160 flights (25%) traverse a CONUS Cell with ISS conditions generating a daily average of 57.5K nautical miles of contrails (sigma 25K nm). The contrails generated an estimated average daily net radiative forcing of +7.08 mW/m^2. During the day, the albedo cooling of incoming shortwave radiation of –1.84 mW/m^2 by the contrails is negated by the +5.85 mW/m^2 warming of trapped outgoing longwave radiation. At night the contrails trap outgoing longwave radiation +3.07 mW/m^2.

On a per flights basis, the average flight generated 2.08 nautical miles of contrails. The average flight that traversed one or more CONUS ISS cells generated 38 nautical miles of contrails. The average NRF per flight in the CONUS is 0.0003 mW/m^2. The average NRF per contrail generating flights is 0.00115 mW/m^2.

The average NRF per along-track distance nautical mile is 2.9 x 10-7 mW/m^2. The average NRF per contrail along track distance nautical mile is 0.00012 mW/m^2. Additionally the evaluation of alternative flight levels showed that modifying the flight level has a large impact to contrail formation. Increasing the flight level by 2000 ft provides a 62% decrease in contrails when accounting for their persistence (nautical miles – Hr). This change would in average decrease the average fuel burn by 2%. This case will be limited due to flight level ceilings or traffic density at other flight levels. The decrease in contrails is driven by the fact that the altitudes at which ISSRs are more frequent coincides with frequently used flight levels. The drop in ISSR density from FL 350 to 360 causes a large number of flights to reduce or eliminate their contrails when raised above 35000 ft.

Results can vary greatly due to the constant movement of ISSR; therefore to accurately predict contrail formation and persistence a dynamic weather model is required. The proposed model provides a method to evaluate a commercial aircraft's impact for a specific flight path and weather conditions.

Future analysis could complement the flight trajectories to account for seasonal and less frequent flight paths by including all flights during a year. Similarly, the fuel consumption can be detailed by associating the engine type to each flight. Additionally, future work may include the evaluation of multiple flight level adjustments during the trajectory as well further evaluation of the impact of different crystal sizes. Last, the selection of alternative flight levels can be expanded to account for the airline's cost index.

APPENDIX A: AGTP

AGGREGATE GLOBAL TEMPERATURE POTENTIAL (AGTP)

Climate metrics are aimed at providing a common scale to compare different greenhouse gases. If the metrics are to be used as a tool in developing and evaluating aviation operations, they should be transparent and easy to apply. Global Warming Potential (GWP) and Aggregate Global Temperature Potential (AGTP) are some metrics used.

Aviation emissions and contrails have very different characteristics and influence the climate either directly or by decomposition into other chemical compounds. The impact of certain gases depends both on the amount of emission and the location of the emission. In addition, the impact of the emissions varies with the choice of the time horizon. These variations make it necessary to develop a common yardstick to measure the impact of various gases. Several climate metrics have been developed to assess the impact of aviation emissions. These metrics have been used to develop strategies for limiting the impact on the environment.

Pulse AGTP for CO_2 emission

The response of global-mean surface temperature to a unit impulse of ΔF ,using a second order model is given by

$$R(t) = \sum_{j=1}^{2} \frac{c_j}{d_j} e^{-t/d_j}$$

where the parameters cj and dj are given in Table 25.

The radiative forcing for CO_2 emission is made of a steady state component and three exponentially decaying components, and is given by:

$$\Delta F^{CO2}(t) = A^{CO2} \ \ (a_0 + \sum_{i=1}^{3} a_i \, e^{-\frac{t-to}{a_j}})$$

Where A^{CO2} =1.82X10^{-15} [W/m^2 kg] is the specific forcing due to CO_2 and its value is taken from past studies. The exponential component in parentheses is the decay of an impulse emission of CO_2 with time and the parameters are shown in Table 26. The expression for AGTP is evaluated after substituting. The pulse AGTP for 1 kg CO_2 emission for a time horizon H can then be found by analytical integration the equation, and is given by:

Table 25: Extracted from Sridhar - Integration of Linear Dynamic Emission and Climate

Index	0	1	2	3
ai (unitless)	0.217	0.259	0.338	0.186
αi (years)	172.9	18.51	1.186	
cj (K/Wm-2)		0.631	0.429	
dj (years)		8.4	409.5	

Table 26: Pulse AGTP coefficients

Horzion (years)	H =25	H =50	H =100
Pulse AGTP (Co2) H [K/kg]	A^{CO2} X 0.3686	A^{CO2} X 0.3176	A^{CO2} X 0.2788

(Based on Sridhar)

Where A^{CO_2}=1.82·10-15 W/m^2 kg

The contrail induced RF is calculated by:

$$RF_{net}(\mathrm{t}) = \frac{1}{A_{Earth}} \sum_{flights} \int_{length} RF_{net}\,(t,s) W(t,s)\,\mathrm{ds}$$

Schumann (2011)

Where A_{Earth} = Earth Surface area

$RF_{nets} = RF_{LS}+RF_{SW}$

W(t,s) = Contrail segment of geometrical width W

In his 2011 paper Schumann provides and compares two units of measure for the effect of contrails. The "Radiative Forcing" (RF) and the "Energy Forcing" (EF). The RF describes the net effect of flights at a given time, while the EF will describe the net effect for an individual flight over the life cycle of the contrail.

$$RF_{net}(\mathrm{t}) = \frac{1}{A_{Earth}} \sum\nolimits_{flights} \int_{length} RF_{net}\,(t,s) W(t,s)\,\mathrm{ds}$$

$$\mathrm{EF} = \int_{lifetime} (RF_{net}\,(t',s) W(t',s))\mathrm{dt}'$$

Schumann (2011)

APPENDIX B: EXTRACT AND PROCESSING CODE

WEATHER DATA EXTRACT:

```
Created on Mon Oct 12 12:14:08 2015
@author: Denis Avila

import time
import pygrib
import numpy as np
import os
import os.path
import mysql.connector
from mysql.connector import Error

try:
    conn = mysql.connector.connect(host='localhost',
                                   port=1234,
                                   database='Weather_db',
                                   user='user',
                                   password=pass)
```

```
        if conn.is_connected():
            print('Connected to MySQL database')
            cursor = conn.cursor()
            print('Cursor ready')
except Error as e:
        print(e)

dd1 = 01
dd2 = 33
dd = dd1
hh = 00
yy = 14
mnth = [11,12]
for mm in mnth:
        print( 'month', mm )
        for dd in range(dd1,dd2):
            for hh in range(0,25):
                for zz in range(15,34)
                    filestr= '20'+ str(yy).zfill(2) + '_' + str(mm).zfill(2) + '/' \
                        + str(yy).zfill(2) + str(mm).zfill(2) + str(dd).zfill(2) \
                        + str(hh).zfill(2) +'.rap.t' + str(hh).zfill(2) + 'z.awp130bgrbf00.grib2'
                    folderstr = '/media/denis/3T/Contrails/Weather_files/'
                    fileout ='ISSR_20' + str(yy).zfill(2) + '_' + str(mm).zfill(2) + \
                        '_' + str(dd).zfill(2) + '_' +str(hh).zfill(2) +'_at_' + str(zz).zfill(2) + '.csv'
                    folderout ='/location/' + str(yy).zfill(2) + '_' + str(mm).zfill(2)+'/'
                    url = folderstr + filestr
                    fout = folderout + fileout
                    grib=url ; #'multi_1.at_10m.t00z.f000.grib2';
                    fileout2 ='ISSR_20' + str(yy).zfill(2) + '_' + str(mm).zfill(2) + \
                        '_' + str(dd).zfill(2) + '_' +str(hh).zfill(2) +'_at_' + str(zz).zfill(2) + 'T.csv'
                    fout2 = folderout + fileout2

                    if os.path.isfile(grib) and os.access(grib, os.R_OK):

                        grbs=pygrib.open(grib)

                        fout = open(fout, 'w')
                        ISSR = np.zeros((337,451))
                        ISSRT = np.zeros((160000,10))#, dtype=float)
                        grb = grbs.select(name='Temperature')[zz] #[0] is level, valid values go from 0 to 50
                        T_var=grb.values
                        grb = grbs.select(name='Specific humidity')[zz]
                        H_var=grb.values
                        grb = grbs.select(name='U component of wind')[zz]
                        U_var=grb.values
                        grb = grbs.select(name='V component of wind')[zz]
                        V_var=grb.values
                        grb = grbs.select(name='Pressure')[zz]
                        P_var=grb.values
                        lat,lon = grb.latlons()
                        z = 0
                        print("Processing day:", dd, ' Hour:', hh, ' Z:',zz)
                        for i in range(0,450): #451 for LON
                            for j in range(0,336): #337 for LAT
                                if (T_var[j,i] <=233.15 and H_var[j,i] >=  0.0001):
                                    ISSR[j,i] = 1
                                    date_send = time.strftime('20' + str(yy).zfill(2) + '-' + str(mm).zfill(2) + \
                                        '-' + str(dd).zfill(2) + ' ' +str(hh).zfill(2) + ':00:00')
                                    args = \
                                    (
                                    i,
                                    j,
                                    zz,
                                    ("%.10f" % T_var[j,i]),
                                    ("%.10f" % H_var[j,i]),
                                    ("%.10f" % U_var[j,i]),
```

```
                        ("%.10f" % V_var[j,i]),
                        ("%.10f" % P_var[j,i]),
                        date_send,
                        hh
                        )
                        query = "INSERT INTO
      weather(LON,LAT,Z_VAL,TEMP,SPEC_HUM,U_VAR,V_VAR,PRESS,DATE,HOUR) " \
                        "VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)"
                        #print args

                        cursor.execute(query, args)
                    conn.commit()

                    np.savetxt(fout, ISSR, delimiter=",")  #fout.write(ISSR)
                    fout.close()
                    print('last insert id', cursor.lastrowid)
                else:
                    print "Skipping" , grib
        print('last insert id', cursor.lastrowid)
cursor.close()
conn.close()
print('Cursor closed. Done')
```

WEATHER DATA LOAD:

```
import sys, time
import mechanize

pullfiles = 1
M = '01'
Y = '2016'
d1 = int( Y + M + '01')
d2 = int( Y + M + '33')
pdays = xrange(d1,d2)

k = 1
def pulldown(myfiles):
  br1 = mechanize.Browser()
  for link in myfiles:
        if pullfiles == 1:
            url="http://weather file location/"
            response = br1.open(url)
            br1._factory.is_html = True
            sn = ("/media/…Weather_files/" + Y + "_" + M + '/' + link.text)
            request = br1.click_link(link)
            resp = br1.follow_link(link)
            global k

            buffer = resp.read()
            with open(sn, 'w') as fo:
                    fo.write(buffer)
            k = k+1
            print ('writing to file. k=' , k, ' link:', link.text)
            time.sleep(5)
            response.set_data(response.get_data().replace("<!---", "<!--"))
            br1.set_response(response)

def main ():
  br = mechanize.Browser()

  br.set_handle_refresh(False)

  for ix in range(len(pdays)):

      url="http://weather source/data/grib/rap/"
```

```
    response = br.open(url)
    for link in br.links():
        if link.url[0:8] == str(pdays[ix]):
            request = br.click_link(link)
            response = br.follow_link(link)
            print (link.url, ' Link with date in range')

            for link in br.links():
                #print link.text, link.url
                if link.url[:7] == 'hybrid/':
                    #print link.text, link.url
                    response = br.follow_link(link)
                    print (response.geturl())
                    myfiles = []
                    counter = 0
                    for link in br.links():
                        if link.url[-8:] == '00.grib2':
                            myfiles.append(link)
                            counter += 1
            pulldown(myfiles)
  print ("Done"       )
main()
```

Load ADS-B

```
#Get Traffic Data From abd-s exchange json file

#install.packages("jsonlite")

#RORACLE INSTALL
#setwd("E:/Contrails/R_Files/driver/")
#install.packages('ROracle_1.3-1.zip', repos = NULL)

library(ROracle)
library(jsonlite)

#json_file <- "http://api.worldbank.org/country?per_page=10&region=OED&lendingtype=LNX&format=json"

drv <- dbDriver("Oracle")
ORA_Con <- dbConnect(drv, "DB", "Pass", dbname='localhost:host')

getwd()
setwd('E:/Contrails/Docs/proposal/adsb_data')

json_file <- "\\path\2016-06-20-0000Z.json"
json_data <- fromJSON(paste(readLines(json_file), collapse=""))
adbs_exc <- fromJSON('2016-06-20-0000Z.json')
adbs_frameALL <- adbs_exc$acList
adbs_frame <- adbs_frameALL[colnames(adbs_frameALL)[c(1:45, 47:54, 56:61)]]
write.table(adbs_frame, file = 'file.csv', sep = '|', row.names = FALSE)
adbs_frame <- adbs_frameALL[colnames(adbs_frameALL)[c(1:45, 47:54 )]]

i = 1
files <- list.files(path='//adsb_data', pattern='*0Z.json', full.names=F, recursive=FALSE)
for(filename in files) {
  adbs_exc <- fromJSON(filename)
  adbs_frameALL <- adbs_exc$acList
  if (i == 1) {adbs_frame_DAY <- adbs_frameALL[c("Id",      "Rcvr",   "HasSig",  "Icao",    "Bad",    "Reg",    "FSeen",
        "TSecs",   "CMsgs",  "AltT",    "Call",    "Tisb",
                      "TrkH",    "Type",    "Mdl",     "Man",    "CNum",   "From",    "To",     "Op",     "OpIcao",
        "Sqk",     "VsiT",   "WTC",     "Species",
                      "Engines",   "EngType",        "EngMount",        "Mil",    "Cou",    "HasPic",
        "Interested",        "FlightsCount",      "SpdTyp",
                      "CallSus",   "TT",      "Trt",     "Year",    "Sig",    "Alt",    "GAlt",   "Gnd",    "InHg",
        "Lat",     "Long",   "Mlat",    "Spd",
                      "Trak",      "Vsi",     "ResetTrail","PosTime")          ]
```

```
        } else { adbs_frame <- adbs_frameALL[c("Id", "Rcvr",    "HasSig",  "Icao",    "Bad",    "Reg",    "FSeen",  "TSecs",
           "CMsgs",  "AltT",     "Call",     "Tisb",
                                "TrkH",     "Type",     "Mdl",      "Man",     "CNum",  "From",   "To",     "Op",      "OpIcao",
           "Sqk",      "VsiT",     "WTC",     "Species",
                                "Engines",  "EngType",              "EngMount",           "Mil",      "Cou",     "HasPic",
           "Interested",            "FlightsCount",         "SpdTyp",
                                "CallSus",  "TT",        "Trt",       "Year",     "Sig",      "Alt",      "GAlt",    "Gnd",      "InHg",
           "Lat",       "Long",     "Mlat",     "Spd",
                                "Trak",      "Vsi",      "ResetTrail","PosTime")            ]
          adbs_frame_DAY <- rbind(adbs_frame_DAY,adbs_frame )}
  i = i+1
}
write.table(adbs_frame_DAY, file = 'All_frame_DAY.csv', sep = '|',col.names = TRUE, row.names = FALSE)

names(adbs_frame_DAY)[1]<-paste("ADB_ID")
names(adbs_frame_DAY)[18]<-paste("L_FROM")
names(adbs_frame_DAY)[19]<-paste("L_TO")
dbWriteTable(ORA_Con,"TRAFFIC_ADB1",adbs_frame_DAY)
```

R-STATS

```
#Get Flight path length

sqlstr = paste("select traffic_id, fid,  ADB_ID ||'_' ||  ICAO ||'_' || CALL_ ID2,  dept_arpt, arr_Arpt, cur_lat, cur_lon

          from TRAFFIC_ADB_V

          where

         fid in

         (select distinct fid

          from TRAFFIC_ADB_V

          where pos2_id is not null

           )

           order by 2, POSTIME" , sep = "")

#Trajectory along track distance

sqlstr1 = paste("select

count(distinct(fid)) ,

sum(distance)  Distance ,
```

```
sum(case when day_night = 'Day' then  (distance) end)  Day_Distance,
sum(case when day_night = 'Night' then  (distance) end)  Night_Distance,
sum(case when day_night = 'Day' then  (distance) end) / sum(case when day_night = 'Night' then  (distance) end) Ratio,
sum(case when day_night = 'Day' then  (distance) end) / sum(distance) Ratio2
from Flight_Stats
where
include = 1" , sep = "")


Flight_Path<<-dbGetQuery(ORA_Con,sqlstr)

Flight_Path$Distance<-0

i = 1
while (i <= nrow(Flight_Path)){
  Curr_Flight  = Flight_Path[i,"FID"]
  Curr_Dist=0
  while (Curr_Flight == Flight_Path[i+1,"FID"] )  {
    lat1  = Flight_Path[i,"CUR_LAT"]
    lat2  = Flight_Path[i+1,"CUR_LAT"]
    lon1  = Flight_Path[i,"CUR_LON"]
    lon2  = Flight_Path[i+1,"CUR_LON"]

    Dist_flight = dist_a_b(lat1, lon1, lat2, lon2)  #in km
    Flight_Path[i+1,"Distance"] = Dist_flight
    Curr_Dist = Curr_Dist+Dist_flight
    i=i+1
  }
  i=i+1
  #print(Curr_Flight)
  #print(Curr_Dist)

}
```

```
write.csv(Flight_Path, file = paste("E:\\..\\Flight_DistanceN.csv", sep=""), row.names = TRUE)


Add_Dist <- function()
{
TRAJ_DATA$Distance_Full<-0


i = 1
while (i <= nrow(TRAJ_DATA))
{

   lat1  = TRAJ_DATA[i,"LAT1"]
   lat2  = TRAJ_DATA[i,"LAT2"]
   lon1  = TRAJ_DATA[i,"LON1"]
   lon2  = TRAJ_DATA[i,"LON2"]
   Dist_flight = 0
   if(is.numeric(lat1) & is.numeric(lat2) & is.numeric(lon1) & is.numeric(lon2)) {
   Dist_flight = dist_a_b(lat1, lon1, lat2, lon2)  #in km
   }
   TRAJ_DATA[i,"Distance_Full"] = Dist_flight
i=i+1

}
}

#-----------------------------------------
library(ggplot2)
library(ggmap)
library(maps)
#library(mapdata)

Exchange_LatLon <- function(d)
```

```
{

sqlstr = paste("select w.TRAFFIC_ID, w.ADB_ID, w.ICAO, w.CALL_, w.LAT, w.LON, w.Z_VAL, fl.flight_level
        from
        ADBS_Weather_1 w  join
        flight_level_ref2 fl
        on w.z_val = fl.z_val
        where w_Date >= '",format(d,'%d-%b-%Y') ,"' and w_Date < '",format(d+1,'%d-%b-%Y') ,"'" , sep = "")

sqlstr = paste("select w.TRAFFIC_ID, w.ADB_ID, w.ICAO, w.CALL_, w.LAT, w.LON, w.Z_VAL, fl.flight_level
        from
        ADBS_Weather_1 w  join
        flight_level_ref2 fl
        on w.z_val = fl.z_val" , sep = "")

 Contrail_Freqs<<-dbGetQuery(ORA_Con,sqlstr)



 usa <- map_data("usa")
 gg1 <- ggplot() +
  geom_polygon(data = usa, aes(x=long, y = lat, group = group), fill = NA, color = "black") +
  coord_fixed(1.3)

 gg1 <- gg1 +
  geom_point(data = Contrail_Freqs, mapping = aes(x = LON, y = LAT, color = factor(FLIGHT_LEVEL))) +
  geom_count()

 P_title = paste("Contrails simulation for " ,format(d,'%d-%b-%Y') ,"'" , sep = "")

 gg1 <- gg1 + labs(title = P_title , x = "Lon", y = "Lat") +
    labs ( color = "Flight Level") +
    coord_fixed(ylim=c(16.28,55.48), ratio=1/cos(pi*41.39/180)) +
    coord_fixed(xlim=c(-126.13,-57.38), ratio=1/cos(pi*41.39/180))
```

```
#  16.28N,126.13W and extends to 55.48N, 57.38W.

 print(gg1)

  filesaveas = paste("E:\\Contrails\\Docs\\proposal\\W_files\\Contrail_maps\\EXCHANGE_","_", d,".png", sep="")
  ggsave(filename=filesaveas, plot=gg1, scale = 3)
  return(gg1)
}

#Build Histograms

#to Save pics
SavePic <- function(Saveas)
{
  filename=paste("E:\\\...\\P2_IISE_Stats\\", Saveas , ".jpg", sep ="")

  jpeg(filename)
  plot(Hist1)
  dev.off()
}

#------------------------------------ Build Histogram Conus

#Flight track distance Day vs Night
HistoT<-function(){
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
```

```
Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)

for (k in (1:2)){


  for (DN in c('Day','Night'))

  {



    if (k ==1){   #Distance along track

      mainlabel=paste("ISSR Distribution - ", DN , sep="")

      xcelllimit = 5000

      labelx = "ISSR positive locations"

      #labely = "Flight Cnt"

      savedist = "ISSR_Distribution"

      ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");

      ISSR_DATA =subset(ISSR_CNT, Day_Night==DN)

      ISSR_DATA$W3D_POINTS <- as.numeric(ISSR_DATA$W3D_POINTS)

      ISSR_Hist <- aggregate(ISSR_DATA$W3D_POINTS, by=list(ISSR_DATA$DATE),  FUN=sum)

      #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;

    }

    if (k ==2){   #Distance along track

      mainlabel=paste("ISSR Distribution (%) - ", DN , sep="")

      xcelllimit = 5000

      labelx = "ISSR positive locations (%)"

      #labely = "Flight Cnt"

      savedist = "ISSR_DistributionPCT"

      ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_PCT.csv", colClasses  = "character");

      ISSR_DATA =subset(ISSR_CNT, Day_Night==DN)

      ISSR_DATA$W3D_POINTS <- as.numeric(ISSR_DATA$W3D_POINTS)

      ISSR_Hist <- aggregate(ISSR_DATA$W3D_POINTS, by=list(ISSR_DATA$DATE),  FUN=sum)

      #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;

    }

#-----------
```

```
#Generate Histogram by Counts of Days
Hist1 <-hist(ISSR_Hist$x , col = "aquamarine3", breaks=20,
main=mainlabel, las=0, xlab = label1, ylab = "Number of Days", cex.lab = 1.3)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")

abline(v = mean(ISSR_Hist$x ),col = "royalblue",lwd = 2)
abline(v = median(ISSR_Hist$x), col = "red", lwd = 2)

#Build Ledgend
led_text1=paste("Mean:", format(mean(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_Hist$x), digits=0,scientific = FALSE) , sep="")
legend("topleft",
    legend =c(led_text1,led_text2,led_text3,led_text4,led_text5)
    , cex=0.8
    , col=c("royalblue", "red", "black", "black", "black", "black"))

#-----------

dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',savedist,"_", DN , ".pdf", sep=""))

Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(ISSR_Hist$x, na.rm=TRUE);
Hist_out_median[j] = median(ISSR_Hist$x, na.rm=TRUE);
Hist_out_min[j] = min(ISSR_Hist$x, na.rm=TRUE);
Hist_out_max[j] =max(ISSR_Hist$x, na.rm=TRUE);
Hist_out_StdDev[j] =sd(ISSR_Hist$x, na.rm=TRUE);
```

```
Hist_out_var[j] = var(ISSR_Hist$x, na.rm=TRUE);
j=j+1
  }
  }
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_4",savedist,".csv", sep=""), row.names = TRUE)


}
#-----------


# Build ISSR By flight level   HIST3
HIST3 <- function(){


ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");
ISSR_CNT$W3D_POINTS <- as.numeric(ISSR_CNT$W3D_POINTS)
ISSR_CNT$FLIGHT_LEVEL <- as.numeric(ISSR_CNT$FLIGHT_LEVEL)


ISSR_CNT1<-ISSR_CNT




Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
```

```
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)

j = 1
for (DN in c('Day','Night'))
{
  i = 200
  for (i in seq(200,400, by = 10))
  {

    mainlabel=paste("ISSR Distribution: ", DN ," - FL:" , i, sep="")

   savefilename="ISSR_Dist2"

    labelx = "ISSR weather cells"
    labely = "Days per year"


    ISSR_CNT<-ISSR_CNT1

    if (DN == 'Day'){ISSR_CNT$W3D_POINTS <-ISSR_CNT$W3D_POINTS/13}
    if (DN == 'Night'){ISSR_CNT$W3D_POINTS <-ISSR_CNT$W3D_POINTS/11}

    ISSR_CNT =subset(ISSR_CNT, Day_Night==DN);
    ISSR_CNT =subset(ISSR_CNT, FLIGHT_LEVEL==i);

    if (nrow(ISSR_CNT)>0){
      ISSR_CNT <- aggregate(ISSR_CNT$W3D_POINTS, by=list(ISSR_CNT$DATE),  FUN=sum);
```

```
  #ISSR_Hist$x <- as.numeric(ISSR_Hist$x) ;

  #-----------

ISSR_CNT$x <- as.numeric(ISSR_CNT$x)

#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(ISSR_CNT$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNT$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNT$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNT$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNT$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNT$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(ISSR_CNT$x , col = "aquamarine3", breaks=20,
            main=mainlabel, las=0, xlab = labelx, ylab = labely,
             #xlim = c(0,80000),
            cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNT$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(ISSR_CNT$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
      col=c("royalblue", "red","black", "black", "black", "black" ))


dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',savefilename,"_", DN , " _FL",i, ".pdf", sep=""))
```

```
      Hist_out_SourceHist[j] =mainlabel
      Hist_out_Day_Night[j] =DN
      Hist_out_FL[j] = i
      Hist_out_mean[j] = mean(ISSR_CNT$x, na.rm=TRUE)
      Hist_out_median[j] = median(ISSR_CNT$x, na.rm=TRUE)
      Hist_out_min[j] = min(ISSR_CNT$x, na.rm=TRUE)
      Hist_out_max[j] =max(ISSR_CNT$x, na.rm=TRUE)
      Hist_out_StdDev[j] =sd(ISSR_CNT$x, na.rm=TRUE)
      Hist_out_var[j] = var(ISSR_CNT$x, na.rm=TRUE)
      j=j+1
    }
  }
}

Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_3",savefilename,".csv", sep=""), row.names = TRUE)


}
#-------------------------------------------------------
#-------------------------------------------------------
```

```
#Flight trajectories

#Distnace track, great circle, Weather Cells, flight count
Histof<-function(){

 #ISSR_DATA =subset(ISSR_CNT, Day_Night=='Night')
 #mainlabel="ISSR Distribution - Night"
 #-----------
 TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2.csv", colClasses  = "character")
 TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)
 TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)
 TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)
 TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)


 #Add_Dist()
 i = 1
 while (i <= nrow(TRAJ_DATA))
 {

  lat1  = TRAJ_DATA[i,"LAT1"]
  lat2  = TRAJ_DATA[i,"LAT2"]
  lon1  = TRAJ_DATA[i,"LON1"]
  lon2  = TRAJ_DATA[i,"LON2"]
  Dist_flight = 0
  if(is.numeric(lat1) & is.numeric(lat2) & is.numeric(lon1) & is.numeric(lon2)) {
   Dist_flight = dist_a_b(lat1, lon1, lat2, lon2)  #in km
  }
  TRAJ_DATA[i,"Distance_Full"] = Dist_flight
```

```
  i=i+1

}
TRAJ_DATA1<-TRAJ_DATA
write.csv(TRAJ_DATA1, file = "E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2GC.csv", row.names = TRUE)
TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST3GC.csv", colClasses  = "character")
TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)
TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)
TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)
TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)



TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA =subset(TRAJ_DATA, Include==1);
#TRAJ_DATA =subset(TRAJ_DATA, Distance>50);
TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)
```

```
 for (DN in c('Day','Night','All'))
  {
 for (k in (1:6))
  {

  #Distance along track
if (k ==1){
 TRAJ_DATA<-TRAJ_DATA1
 if (DN =='All'){
    mainlabel=paste("Flight Distance along track", sep="")}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Distance along track - ", DN , sep="")
  }


 xcelllimit = 5000
 labelx = "Distance [nm]"
 labely = "Flight Cnt"
 TrajFile = "FlightDistanceTrack"


 TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');

 TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
 TRAJ_Hist <- aggregate(TRAJ_DATA$Distance, by=list(TRAJ_DATA$ID2), FUN=sum);
 #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
  if (k ==2){ #Distance Great Circle
    TRAJ_DATA<-TRAJ_DATA1
    if (DN =='All'){
      mainlabel=paste("Flight Distance Distribution Great-Circle ", sep="")}
    else{
```

```
   TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
   mainlabel=paste("Flight Distance Distribution Great-Circle - ", DN , sep="")
  }


  # For Distance Origen to Dest

  xcelllimit = 5000

  labelx = "Distance [nm]"
  labely = "Flight Cnt"
  TrajFile = "FlightDistanceGC"


  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA$Distance_Full <- as.numeric(TRAJ_DATA$Distance_Full)*0.868976 #Distance along Track
  TRAJ_Hist <- aggregate(TRAJ_DATA$Distance_Full, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
 }
 if (k ==3){ #Trajectory Cell Count
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
   mainlabel=paste("Flight Intersection with Weather Grid ", sep="")}
   else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Intersection with Weather Grid - ", DN , sep="")
   }




 xcelllimit = 50
  labelx = "Weather cells"
```

```
  labely = "Flight Cnt"
  TrajFile = "Flight_to_WeatherGrid"

  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid
  TRAJ_Hist <- aggregate(TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
 }
if (k ==4){#Flight ISSR Intersec
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flight Intersection with ISSR" , sep="")}
    else{
      TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
      mainlabel=paste("Flight Intersection with ISSR - ", DN , sep="")
    }



    #DN = 'Day'

  xcelllimit = 15
  labelx = "ISSR cells"
  labely = "Flight Cnt"
  TrajFile = "Flight_to_ISSR"

  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0);

  TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
  TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
 }
```

```
  if (k ==5){#Flight ISSR Intersec as pct of flight
    TRAJ_DATA<-TRAJ_DATA1
    if (DN =='All'){
      mainlabel=paste("Percentage of Total Along-track Distance in ISSR ", sep="")}
      else{
        TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
        mainlabel=paste("Percentage of Total Along-track Distance in ISSR  - ", DN , sep="")
      }



    xcelllimit = 1
    labelx = "ISSR cells"
    labely = "Flight Cnt"
    TrajFile = "Flight_to_ISSR_PCT"

    TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
    TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

    TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
    TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

          TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum);

  }
  if (k ==6){#Percentage of Total Along-track Distance in ISSR Day and Night
    #DN = 'Day'
    TRAJ_DATA<-TRAJ_DATA1
    if (DN =='All'){
      mainlabel=paste("Percentage of Flight Distance along track in ISSR", sep="")}
    else{
      TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
      mainlabel=paste("Percentage of Flight Distance along track in ISSR - ", DN , sep="")
```

```
    }

    xcelllimit = 1
    labelx = "Percentage of Total Along-track in ISSR"
    labely = "Flight Cnt"
    TrajFile = "Flight_to_ISSR_PCT"

    TRAJ_DATA<-TRAJ_DATA1

    #TRAJ_DATA =subset(TRAJ_DATA1, Day_Night==DN);
    TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
    TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

    TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
    TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

    TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum) ;
   }

   #-----------
   #Build Ledgend
   led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
   led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

   #Generate Histogram by Counts of Days
```

```
Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
             main=mainlabel, las=0, xlab = labelx, ylab = labely,
             xlim = c(0,xcelllimit),
             cex.lab = 1.3)

 # H2 <-plot(Hist1)
 grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
 abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
 abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
 legend("topright",  legend =ledgD , cex=0.8 ,
        col=c("royalblue", "red","black", "black", "black", "black" ))

 dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))

 xlab=paste0(pretty(TRAJ_Hist$x) * 100, " %")
 #dev.print( device =jpeg, filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
 #          width=480, height=480, pointsize=72, quality=75, bg = "white", res = NA)
# jpeg(filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
#      width=480, height=480, pointsize=75, quality=75) ;
 #jpeg(paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"1_", DN , ".jpg", sep=""),width=480, height=480, pointsize=75, quality=75)



 #dev.cur()  was set to RStudioGD -2. moved to 4 = Jpeg
 #dev.list()
 # dev.set(dev.next())
 #dev.cur()
 #dev.set(2)
 #options("device");

 Hist_out_SourceHist[j] =mainlabel;
 Hist_out_Day_Night[j] =DN;
```

```
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
  }

k = k+1
}# end K
  Hist_Data_Out <- data.frame (
    Hist_out_SourceHist,
    Hist_out_Day_Night,
    Hist_out_FL,
    Hist_out_mean,
    Hist_out_median,
    Hist_out_min,
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"3_",DN,".csv", sep=""), row.names = TRUE)
 # end K (run till here)
}
#-------------------------------------------------
#-------------------------------------------------



#
#Flights and cells with contrails Delay = 0
```

```
{
sqlstr1 = paste("select to_char(date_time,'dd-mon-yy') Date_W, day_night, count(distinct(a.fid)) Flights, count(distinct(pos2_id))
Dis_Cells ,count((pos2_id)) Cells
        from Contrail_Work a join
        (select fid, day_night from flight_stats) b
        on a.fid=b.fid
        where traj_delay = 0
        group by
        to_char(date_time,'dd-mon-yy'), day_night" , sep = "")

TRAJ_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)



for (DN in c('Day','Night','All'))
{
  for (k in (1:2))
  {

    #DISTINCT Cells with contraiks
```

```
if (k ==1){
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("ISSR Cells traversed by Flight trajectory", sep="")
    xcelllimit = 10000}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);
    mainlabel=paste("ISSR Cells traversed by Flight trajectory - ", DN , sep="")
    xcelllimit = 10000



    }

  labelx = "Weather cells with contrails"
  labely = "Days per year"
  TrajFile = "Flight_to_ISSR_Hr0"

  TRAJ_DATA$DIS_CELLS <- as.numeric(TRAJ_DATA$DIS_CELLS)
  TRAJ_Hist <- aggregate(TRAJ_DATA$DIS_CELLS, by=list(TRAJ_DATA$DATE_W), FUN=sum);
}
#DISTINCT Flights with contrails
if (k ==2){
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flights with Contrails", sep="")
    xcelllimit = 10000}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);
    mainlabel=paste("Flights with Contrails - ", DN , sep="")
    xcelllimit = 7000
  }

  labelx = "Flight count with contrails"
```

```
  labely = "Days per year"
  TrajFile = "Flights_wContrail_Hr0"

  TRAJ_DATA$FLIGHTS <- as.numeric(TRAJ_DATA$FLIGHTS)
  TRAJ_Hist <- aggregate(TRAJ_DATA$FLIGHTS, by=list(TRAJ_DATA$DATE_W), FUN=sum);
}

  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
```

```
        col=c("royalblue", "red","black", "black", "black", "black" ))

    dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))



    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
  }
}
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"3_",DN,".csv", sep=""), row.names = TRUE)
# end K (run till here)
}
```

```
#Flight trajectories


#Distnace track, great circle, Weather Cells, flight count

Histof<-function(){



  #ISSR_DATA =subset(ISSR_CNT, Day_Night=='Night')

  #mainlabel="ISSR Distribution - Night"

  #-----------

  TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2.csv", colClasses  = "character")

  TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)

  TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)

  TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)

  TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)



  #Add_Dist()

  i = 1

  while (i <= nrow(TRAJ_DATA))

  {

    lat1  = TRAJ_DATA[i,"LAT1"]

    lat2  = TRAJ_DATA[i,"LAT2"]

    lon1  = TRAJ_DATA[i,"LON1"]

    lon2  = TRAJ_DATA[i,"LON2"]

    Dist_flight = 0

    if(is.numeric(lat1) & is.numeric(lat2) & is.numeric(lon1) & is.numeric(lon2)) {

      Dist_flight = dist_a_b(lat1, lon1, lat2, lon2)  #in km

    }

    TRAJ_DATA[i,"Distance_Full"] = Dist_flight

    i=i+1
```

```
}
TRAJ_DATA1<-TRAJ_DATA
write.csv(TRAJ_DATA1, file = "E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2GC.csv", row.names = TRUE)
TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST3GC.csv", colClasses  = "character")
TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)
TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)
TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)
TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)



TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA =subset(TRAJ_DATA, Include==1);
#TRAJ_DATA =subset(TRAJ_DATA, Distance>50);
TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)



for (DN in c('Day','Night','All'))
{
```

```
for (k in (1:6))
{

 #Distance along track
 if (k ==1){
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
   mainlabel=paste("Flight Distance along track", sep="")}
  else{
   TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
   mainlabel=paste("Flight Distance along track - ", DN , sep="")
  }



  xcelllimit = 5000
  labelx = "Distance [nm]"
  labely = "Flight Cnt"
  TrajFile = "FlightDistanceTrack"



  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');

  TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
  TRAJ_Hist <- aggregate(TRAJ_DATA$Distance, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
 }
 if (k ==2){ #Distance Great Circle
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
   mainlabel=paste("Flight Distance Distribution Great-Circle ", sep="")}
  else{
   TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
   mainlabel=paste("Flight Distance Distribution Great-Circle - ", DN , sep="")
```

```
 }

 # For Distance Origen to Dest

 xcelllimit = 5000

 labelx = "Distance [nm]"
 labely = "Flight Cnt"
 TrajFile = "FlightDistanceGC"

 TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
 TRAJ_DATA$Distance_Full <- as.numeric(TRAJ_DATA$Distance_Full)*0.868976 #Distance along Track
 TRAJ_Hist <- aggregate(TRAJ_DATA$Distance_Full, by=list(TRAJ_DATA$ID2), FUN=sum);
 #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==3){ #Trajectory Cell Count
 TRAJ_DATA<-TRAJ_DATA1
 if (DN =='All'){
   mainlabel=paste("Flight Intersection with Weather Grid ", sep="")}
 else{
   TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
   mainlabel=paste("Flight Intersection with Weather Grid - ", DN , sep="")
 }

 xcelllimit = 50
 labelx = "Weather cells"
 labely = "Flight Cnt"
 TrajFile = "Flight_to_WeatherGrid"
```

```
  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid
  TRAJ_Hist <- aggregate(TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==4){#Flight ISSR Intersec
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flight Intersection with ISSR" , sep="")}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Intersection with ISSR - ", DN , sep="")
  }



  #DN = 'Day'

  xcelllimit = 15
  labelx = "ISSR cells"
  labely = "Flight Cnt"
  TrajFile = "Flight_to_ISSR"

  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0);

  TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
  TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==5){#Flight ISSR Intersec as pct of flight
  TRAJ_DATA<-TRAJ_DATA1
```

```
    if (DN =='All'){
      mainlabel=paste("Percentage of Total Along-track Distance in ISSR ", sep="")}
    else{
      TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
      mainlabel=paste("Percentage of Total Along-track Distance in ISSR  - ", DN , sep="")
    }



    xcelllimit = 1
    labelx = "ISSR cells"
    labely = "Flight Cnt"
    TrajFile = "Flight_to_ISSR_PCT"

    TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
    TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

    TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
    TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

    TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum);

   }
   if (k ==6){#Percentage of Total Along-track Distance in ISSR Day and Night
    #DN = 'Day'
    TRAJ_DATA<-TRAJ_DATA1
    if (DN =='All'){
      mainlabel=paste("Percentage of Flight Distance along track in ISSR", sep="")}
    else{
      TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
      mainlabel=paste("Percentage of Flight Distance along track in ISSR - ", DN , sep="")
    }
```

```
    xcelllimit = 1
    labelx = "Percentage of Total Along-track in ISSR"
    labely = "Flight Cnt"
    TrajFile = "Flight_to_ISSR_PCT"

    TRAJ_DATA<-TRAJ_DATA1

    #TRAJ_DATA =subset(TRAJ_DATA1, Day_Night==DN);
    TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
    TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

    TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
    TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

    TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum) ;
   }



   #-----------
   #Build Ledgend
   led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
   led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

   #Generate Histogram by Counts of Days

   Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
```

```
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))

dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))

xlab=paste0(pretty(TRAJ_Hist$x) * 100, " %")
#dev.print( device =jpeg, filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
#          width=480, height=480, pointsize=72, quality=75, bg = "white", res = NA)
# jpeg(filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
#      width=480, height=480, pointsize=75, quality=75) ;
#jpeg(paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"1_", DN , ".jpg", sep=""),width=480, height=480, pointsize=75, quality=75)




#dev.cur()  was set to RStudioGD -2. moved to 4 = Jpeg
#dev.list()
# dev.set(dev.next())
#dev.cur()
#dev.set(2)
#options("device");

Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
```

```
      Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
      j=j+1;
    }

    k = k+1
  }# end K
  Hist_Data_Out <- data.frame (
    Hist_out_SourceHist,
    Hist_out_Day_Night,
    Hist_out_FL,
    Hist_out_mean,
    Hist_out_median,
    Hist_out_min,
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"3_",DN,".csv", sep=""), row.names = TRUE)
  # end K (run till here)
}
#--------------------------------------------------
#--------------------------------------------------



#
#------------
#contrails by Cell, KN nm
{
```

```
Delay =0

sqlstr1 = paste("select to_char(date_time,'dd-mon-yy'), day_night, count(distinct(a.fid)) Flights, count(distinct(pos2_id)) Dis_Cells
,count((pos2_id)) Cells,
count(distinct(pos2_id)) *13 DIS_km,
        count((pos2_id)) *13 km,
        count(distinct(pos2_id)) *13 *0.539957 Dis_nm,
        count((pos2_id)) *13*0.539957 nm,
        --count((pos2_id)) *13*0.539957 /count(distinct(a.fid)) nm_per_flight
        from Contrail_Work a join
        (select fid, day_night from flight_stats) b
        on a.fid=b.fid
        where traj_delay = 0
        group by
        to_char(date_time,'dd-mon-yy'), day_night" , sep = "")

TRAJ_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\Contrail_cell_km_nm.csv", colClasses  = "character")

TRAJ_DATA$DIS_KM <- as.numeric(TRAJ_DATA$DIS_KM)
TRAJ_DATA$DIS_nm <- as.numeric(TRAJ_DATA$DIS_nm)
TRAJ_DATA$km <- as.numeric(TRAJ_DATA$km)
TRAJ_DATA$nm <- as.numeric(TRAJ_DATA$nm)
TRAJ_DATA$DIS_CELLS <- as.numeric(TRAJ_DATA$DIS_CELLS)
TRAJ_DATA$CELLS <- as.numeric(TRAJ_DATA$CELLS)
TRAJ_DATA$W_Date <- as.Date(TRAJ_DATA$W_Date,"%m/%d/%Y")
TRAJ_DATA1<-TRAJ_DATA
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
```

```
Hist_out_FL <<- character(50)

Hist_out_mean <<-numeric(50)

Hist_out_median <<-numeric(50)

Hist_out_min <<-numeric(50)

Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)




for (DN in c('Day','Night','All'))

{

 for (k in (1:2))

 {


   #DISTINCT Cells with contraiks

   if (k ==1){

     TRAJ_DATA<-TRAJ_DATA1

     if (DN =='All'){

       mainlabel=paste("Yearly Contrails generated in the US Airspace", sep="")

       xcelllimit = 200000

       ycelllimit = 20}

     else{

       TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);

       mainlabel=paste("Yearly Contrails generated in the US Airspace - ", DN , sep="")

       xcelllimit = 150000

       ycelllimit = 20



     }


     labelx = "Nautical Miles"

     labely = "Days per year"
```

```
  TrajFile = "Contrail_nm"

 TRAJ_Hist <- aggregate(TRAJ_DATA$nm, by=list(TRAJ_DATA$W_Date), FUN=sum);
}
#DISTINCT Flights with contrails
if (k ==2){
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Yearly Contrails generated in the US Airspace per flight", sep="")
    xcelllimit = 150000
    ycelllimit = 30}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);
    mainlabel=paste("Yearly Contrails generated in the US Airspace per flight - ", DN , sep="")
    xcelllimit = 150000
    ycelllimit = 30
  }

  labelx = "Nautical Miles"
  labely = "Count of Flights"
  TrajFile = "Contrail_Flights_nm"

  T
  TRAJ_Hist <- aggregate(TRAJ_DATA$nm, by=list(TRAJ_DATA$FLIGHTS), FUN=sum);
}

#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
```

```
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
   col=c("royalblue", "red","black", "black", "black", "black" ))



plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,xcelllimit), ylim = c(0,ycelllimit), #.2,0.6,1
   # main = "ISSR Cell Count",
   #sub="(including FL200 to FL400)",
   main = "Yearly Contrails generated in the US Airspace\n(FL200 to FL400)",

   cex.sub =.8,
   xlab = "Nautical Miles",
   ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
```

```
    legend("topright",  legend =ledgD , cex=0.8 ,
          col=c("royalblue", "red","black", "black", "black", "black" ))


    dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))


    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
  }
}
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
# end K (run till here)
```

```
}

#------------
#------------


#
#------------
#contrails by FID in  nm
{
Delay =0

sqlstr1 = paste("select day_night, a.fid Flight_ID,
        count(distinct(pos2_id)) *13*0.539957  nm_per_flight
        from Contrail_Work a join
        (select fid, day_night from flight_stats) b
        on a.fid=b.fid
        where traj_delay = 0
        group by
        day_night, a.fid" , sep = "")

TRAJ_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
```

```
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)



for (DN in c('Day','Night','All'))
{
 for (k in (1:1))
 {

   #DISTINCT Cells with contraiks
   if (k ==1){
     TRAJ_DATA<-TRAJ_DATA1
     if (DN =='All'){
       mainlabel=paste("Contrails generated per Flight", sep="")
       xcelllimit = 350
       ycelllimit = 4000}
     else{
       TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);
       mainlabel=paste("Contrails generated per Flight - ", DN , sep="")
       xcelllimit = 350
       ycelllimit = 2000


     }

     labelx = "Nautical Miles"
     labely = "Flight Count"
     TrajFile = "Contrail_per FID_nm"

     TRAJ_Hist <- aggregate(TRAJ_DATA$NM_PER_FLIGHT, by=list(TRAJ_DATA$FLIGHT_ID), FUN=sum);
   }
   #DISTINCT Flights with contrails
```

```
# if (k ==2){
#   TRAJ_DATA<-TRAJ_DATA1
#   if (DN =='All'){
#     mainlabel=paste("Yearly Contrails generated in the US Airspace per flight", sep="")
#     xcelllimit = 150000
#     ycelllimit = 30}
#   else{
#     TRAJ_DATA =subset(TRAJ_DATA, DAY_NIGHT==DN);
#     mainlabel=paste("Yearly Contrails generated in the US Airspace per flight - ", DN , sep="")
#     xcelllimit = 150000
#     ycelllimit = 30
#   }
#
#   labelx = "Nautical Miles"
#   labely = "Count of Flights"
#   TrajFile = "Contrail_Flights_nm"
#
#   T
#   TRAJ_Hist <- aggregate(TRAJ_DATA$nm, by=list(TRAJ_DATA$FLIGHTS), FUN=sum);
# }




#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
```

```
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))



plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,xcelllimit), ylim = c(0,ycelllimit), #.2,0.6,1
    # main = "ISSR Cell Count",
    #sub="(including FL200 to FL400)",
    main = mainlabel,

    cex.sub =.8,
    xlab = labelx,
    ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))



dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))
```

```
    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
  }
}
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
# end K (run till here)
}
#---------------------------------End of By Fid

#Flight trajectories
```

```
#Distnace track, great circle, Weather Cells, flight count
Histof<-function(){


 #ISSR_DATA =subset(ISSR_CNT, Day_Night=='Night')
 #mainlabel="ISSR Distribution - Night"
 #-----------
 TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2.csv", colClasses  = "character")
 TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)
 TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)
 TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)
 TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)


 #Add_Dist()
 i = 1
 while (i <= nrow(TRAJ_DATA))
 {

  lat1  = TRAJ_DATA[i,"LAT1"]
  lat2  = TRAJ_DATA[i,"LAT2"]
  lon1  = TRAJ_DATA[i,"LON1"]
  lon2  = TRAJ_DATA[i,"LON2"]
  Dist_flight = 0
  if(is.numeric(lat1) & is.numeric(lat2) & is.numeric(lon1) & is.numeric(lon2)) {
   Dist_flight = dist_a_b(lat1, lon1, lat2, lon2)  #in km
  }
  TRAJ_DATA[i,"Distance_Full"] = Dist_flight
  i=i+1

 }
 TRAJ_DATA1<-TRAJ_DATA
 write.csv(TRAJ_DATA1, file = "E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST2GC.csv", row.names = TRUE)
```

```
TRAJ_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\TRAJECTORY_CNT_DIST3GC.csv", colClasses  = "character")
TRAJ_DATA$LAT1 <- as.numeric(TRAJ_DATA$LAT1)
TRAJ_DATA$LAT2 <- as.numeric(TRAJ_DATA$LAT2)
TRAJ_DATA$LON1 <- as.numeric(TRAJ_DATA$LON1)
TRAJ_DATA$LON2 <- as.numeric(TRAJ_DATA$LON2)



TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA =subset(TRAJ_DATA, Include==1);
#TRAJ_DATA =subset(TRAJ_DATA, Distance>50);
TRAJ_DATA1<-TRAJ_DATA
TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)




for (DN in c('Day','Night','All'))
{
  for (k in (1:6))
  {


    #Distance along track
```

```
if (k ==1){
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flight Distance along track", sep="")}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Distance along track - ", DN , sep="")
  }


  xcelllimit = 5000
  labelx = "Distance [nm]"
  labely = "Flight Cnt"
  TrajFile = "FlightDistanceTrack"


  #TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA =subset(TRAJ_DATA, Include==1);

  TRAJ_DATA$Distance <- as.numeric(TRAJ_DATA$Distance) #Distance along Track
  TRAJ_Hist <- aggregate(TRAJ_DATA$Distance, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==2){ #Distance Great Circle
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flight Distance Distribution Great-Circle ", sep="")}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Distance Distribution Great-Circle - ", DN , sep="")
  }
```

```
 # For Distance Origen to Dest

 xcelllimit = 5000

 labelx = "Distance [nm]"
 labely = "Flight Cnt"
 TrajFile = "FlightDistanceGC"


 TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
 TRAJ_DATA$Distance_Full <- as.numeric(TRAJ_DATA$Distance_Full)*0.868976 #Distance along Track
 TRAJ_Hist <- aggregate(TRAJ_DATA$Distance_Full, by=list(TRAJ_DATA$ID2), FUN=sum);
 #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==3){ #Trajectory Cell Count
 TRAJ_DATA<-TRAJ_DATA1
 if (DN =='All'){
   mainlabel=paste("Flight Intersection with Weather Grid ", sep="")}
 else{
   TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
   mainlabel=paste("Flight Intersection with Weather Grid - ", DN , sep="")
  }



 xcelllimit = 50
 labelx = "Weather cells"
 labely = "Flight Cnt"
 TrajFile = "Flight_to_WeatherGrid"

 TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
 TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid
```

```
  TRAJ_Hist <- aggregate(TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==4){#Flight ISSR Intersec
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Flight Intersection with ISSR" , sep="")}
  else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Flight Intersection with ISSR - ", DN , sep="")
   }



  #DN = 'Day'

  xcelllimit = 15
  labelx = "ISSR cells"
  labely = "Flight Cnt"
  TrajFile = "Flight_to_ISSR"

  TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
  TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0);

  TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
  TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT, by=list(TRAJ_DATA$ID2), FUN=sum);
  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
}
if (k ==5){#Flight ISSR Intersec as pct of flight
  TRAJ_DATA<-TRAJ_DATA1
  if (DN =='All'){
    mainlabel=paste("Percentage of Total Along-track Distance in ISSR ", sep="")}
  else{
```

```
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Percentage of Total Along-track Distance in ISSR  - ", DN , sep="")
   }



   xcelllimit = 1
   labelx = "ISSR cells"
   labely = "Flight Cnt"
   TrajFile = "Flight_to_ISSR_PCT"

   TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
   TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

   TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
   TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

   TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum);

  }
  if (k ==6){#Percentage of Total Along-track Distance in ISSR Day and Night
   #DN = 'Day'
   TRAJ_DATA<-TRAJ_DATA1
   if (DN =='All'){
    mainlabel=paste("Percentage of Flight Distance along track in ISSR", sep="")}
   else{
    TRAJ_DATA =subset(TRAJ_DATA, Day_Night==DN);
    mainlabel=paste("Percentage of Flight Distance along track in ISSR - ", DN , sep="")
   }



   xcelllimit = 1
   labelx = "Percentage of Total Along-track in ISSR"
```

```
labely = "Flight Cnt"
TrajFile = "Flight_to_ISSR_PCT"

TRAJ_DATA<-TRAJ_DATA1

#TRAJ_DATA =subset(TRAJ_DATA1, Day_Night==DN);
TRAJ_DATA =subset(TRAJ_DATA, Domestic_Flight=='Y');
TRAJ_DATA =subset(TRAJ_DATA, ISSR_AUG16_CNT>0); #Positive

TRAJ_DATA$ISSR_AUG16_CNT <- as.numeric(TRAJ_DATA$ISSR_AUG16_CNT); #Trajectory intersec Weather 16Aug15
TRAJ_DATA$POS2_ID_CNT <- as.numeric(TRAJ_DATA$POS2_ID_CNT);   # Posid2 found in weather grid

TRAJ_Hist <- aggregate(TRAJ_DATA$ISSR_AUG16_CNT/TRAJ_DATA$POS2_ID_CNT, by=list(TRAJ_DATA$ID2),
FUN=sum) ;
}


#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
      main=mainlabel, las=0, xlab = labelx, ylab = labely,
      xlim = c(0,xcelllimit),
      cex.lab = 1.3)
```

```
# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
      col=c("royalblue", "red","black", "black", "black", "black" ))

dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))

xlab=paste0(pretty(TRAJ_Hist$x) * 100, " %")
#dev.print( device =jpeg, filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
#         width=480, height=480, pointsize=72, quality=75, bg = "white", res = NA)
# jpeg(filename = paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".jpg", sep=""),
#      width=480, height=480, pointsize=75, quality=75) ;
#jpeg(paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"1_", DN , ".jpg", sep=""),width=480, height=480, pointsize=75, quality=75)



#dev.cur()  was set to RStudioGD -2. moved to 4 = Jpeg
#dev.list()
# dev.set(dev.next())
#dev.cur()
#dev.set(2)
#options("device");

Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
```

```
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
   }

   k = k+1
  }# end K
  Hist_Data_Out <- data.frame (
   Hist_out_SourceHist,
   Hist_out_Day_Night,
   Hist_out_FL,
   Hist_out_mean,
   Hist_out_median,
   Hist_out_min,
   Hist_out_max,
   Hist_out_StdDev,
   Hist_out_var
   ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"3_",DN,".csv", sep=""), row.names = TRUE)
  # end K (run till here)
}
#-------------------------------------------------
#-------------------------------------------------




#------------
#contrails RF
{

sqlstr1 = paste(" select
to_Char(date_time,'dd-mon-yy') Day1,
```

```
            day_night,
            count(distinct(a.fid)),
            Traj_delay,
            count(distinct(pos2_id)) ISSR_CELLS,
            sum(RFSW) RFSW ,
            sum( RFLW) RFLW,
            sum(RF) Total_RF_Wm2,
            sum(RFSW_W) RFSW_W ,
            sum( RFLW_W) RFLW_W,
            sum(RF_W) Total_RF_W
            from Contrail_Work a join
            (select fid, day_night from flight_stats) b
            on a.fid=b.fid
            group by
            to_Char(date_time,'dd-mon-yy') ,
            day_night,
            Traj_delay" , sep = "")

CONTRAIL_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

CONTRAIL_DATA1 <- read.csv("E:\\\...\\P2_IISE_Stats\\CONTRAIL_DATA2.csv", colClasses  = "character")
CONTRAIL_DATA<-CONTRAIL_DATA1
CONTRAIL_DATA$RFSW <- as.numeric(CONTRAIL_DATA$RFSW)
CONTRAIL_DATA$RFLW <- as.numeric(CONTRAIL_DATA$RFLW)
CONTRAIL_DATA$TOTAL_RF_WM2 <- as.numeric(CONTRAIL_DATA$TOTAL_RF_WM2)
CONTRAIL_DATA$RFSW_W <- as.numeric(CONTRAIL_DATA$RFSW_W)
CONTRAIL_DATA$RFLW_W <- as.numeric(CONTRAIL_DATA$RFLW_W)
CONTRAIL_DATA$TOTAL_RF_W <- as.numeric(CONTRAIL_DATA$TOTAL_RF_W)

CONTRAIL_DATA$W_Date <- as.Date(CONTRAIL_DATA$DAY1,"%d-%b-%y")
CONTRAIL_DATA1<-CONTRAIL_DATA
CONTRAIL_DATA<-CONTRAIL_DATA1
```

```
#write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\CONTRAIL_DATA.csv", sep=""), row.names = TRUE)
#Cn_Factor = 1000000000

#surface of earth to get W/m2
Cn_Factor =  510072000000000
# *1000 to get Miliwatts
Cn_Factor =  510072000000

CONTRAIL_DATA$RFSW <- as.numeric(CONTRAIL_DATA$RFSW)/Cn_Factor
CONTRAIL_DATA$RFLW <- as.numeric(CONTRAIL_DATA$RFLW)/Cn_Factor
CONTRAIL_DATA$TOTAL_RF_WM2 <- as.numeric(CONTRAIL_DATA$TOTAL_RF_WM2)/Cn_Factor
CONTRAIL_DATA$RFSW_W <- as.numeric(CONTRAIL_DATA$RFSW_W)/Cn_Factor
CONTRAIL_DATA$RFLW_W <- as.numeric(CONTRAIL_DATA$RFLW_W)/Cn_Factor
CONTRAIL_DATA$TOTAL_RF_W <- as.numeric(CONTRAIL_DATA$TOTAL_RF_W)/Cn_Factor
CONTRAIL_DATA1<-CONTRAIL_DATA
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)


for (DN in c('Day','Night','All'))
{
  for (k in (1:3))
  {
```

```
#SW
if (k ==1){
 CONTRAIL_DATA<-CONTRAIL_DATA1
 if (DN =='All'){
  mainlabel=paste("Radiative Forcing from Contrails in the US Airspace", sep="")
  seclabel = "Short Wave: (Reflexion of solar radiation)"
  xcelllimit = 0
  ycelllimit = 20
  xcellStart = -3
  ycellStart = 0}
 else{
  #
  #CONTRAIL_DATA =subset(CONTRAIL_DATA, DAY_NIGHT==DN);
  # its all SW - Day
  mainlabel=paste("Radiative Forcing from Contrails in the US Airspace - ", DN , sep="")
  seclabel = "Short Wave: (Reflexion of solar radiation)"
  xcelllimit = 0
  ycelllimit = 20
  xcellStart = -3
  ycellStart = 0

 }

 labelx = "Radiative Forcing mW/m2"
 labely = "Days per year"
 TrajFile = "RFSW_W"

 TRAJ_Hist <- aggregate(CONTRAIL_DATA$RFSW_W, by=list(CONTRAIL_DATA$W_Date), FUN=sum);
}
#LW
if (k ==2){
 CONTRAIL_DATA<-CONTRAIL_DATA1
```

```
  if (DN =='All'){
    mainlabel=paste("Radiative Forcing from Contrails in the US Airspace", sep="")
    seclabel = "Long Wave: (Greenhouse effect)"
    xcelllimit = 20
    ycelllimit = 20
    xcellStart = 0
    ycellStart = 0}
  else{
    CONTRAIL_DATA =subset(CONTRAIL_DATA, DAY_NIGHT==DN);
    mainlabel=paste("Radiative Forcing from Contrails in the US Airspace - ", DN , sep="")
    seclabel = "Long Wave: (Greenhouse effect)"
    xcelllimit = 15
    ycelllimit = 20
    xcellStart = 0
    ycellStart = 0



  }

  labelx = "Radiative Forcing mW/m2"
  labely = "Days per year"
  TrajFile = "RFLW_W"

  TRAJ_Hist <- aggregate(CONTRAIL_DATA$RFLW_W, by=list(CONTRAIL_DATA$W_Date), FUN=sum);
}
#SW+LW
if (k ==3){
  CONTRAIL_DATA<-CONTRAIL_DATA1
  if (DN =='All'){
    mainlabel=paste("Radiative Forcing from Contrails in the US Airspace", sep="")
    seclabel = "Net radiative forcing"
    xcelllimit = 15
    ycelllimit = 15
```

```
    xcellStart = 0
    ycellStart = 0}
   else{
    CONTRAIL_DATA =subset(CONTRAIL_DATA, DAY_NIGHT==DN);
    mainlabel=paste("Radiative Forcing from Contrails in the US Airspace - ", DN , sep="")
    seclabel = "Net radiative forcing"
    xcelllimit = 10
    ycelllimit = 15
    xcellStart = 0
    ycellStart = 0


   }

   labelx  = "Radiative Forcing mW/m2"
   labely = "Days per year"
   TrajFile = "RFSW_LW_W"

   TRAJ_Hist <- aggregate(CONTRAIL_DATA$TOTAL_RF_W, by=list(CONTRAIL_DATA$W_Date), FUN=sum);
  }


  #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
  #-----------
  #Build Ledgend
  led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
  led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
  led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
  led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
  led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
  led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
  led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , " mW/m2",
sep="")
```

```
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))



plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
    # main = "ISSR Cell Count",
    #sub="(including FL200 to FL400)",
    main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),

    cex.sub =.8,
    xlab = labelx,
    ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))



dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))
```

```
    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;
  }
}
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
# end K (run till here)
}

#------------
#------------
```

```
 # For Distance Origen to Dest
Histo<- function(){
    TRAJ_Hist <- aggregate(TRAJ_DATA$Distance_Full, by=list(TRAJ_DATA$ID2), FUN=sum);

   mainlabel=paste("Flight Distance Distribution Great-Circle - Night" , sep="")
   savefilename=paste("Flight Distance Distribution GreatCircle" , sep="")
   labely = "Flight Cnt"
   labelx = "Distance"

   #Build Ledgend
   led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
   led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
   led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")

   #Generate Histogram by Counts of Days

   Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=50,
             main=mainlabel, las=0, xlab = labelx, ylab = labely,
             xlim = c(0,5000),
             cex.lab = 1.3)

   # H2 <-plot(Hist1)
   grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
   abline(v = mean(TRAJ_Hist$x , na.rm=TRUE),col = "royalblue",lwd = 2)
```

```
    abline(v = median(TRAJ_Hist$x,  na.rm=TRUE), col = "red", lwd = 2)
    legend("topright",  legend =c(led_text1,led_text2,led_text3,led_text4,led_text5) , cex=0.8 ,
           col=c("royalblue", "red","black", "black", "black", "black" ))


    #-----------

write.csv(TRAJ_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\TRAJ_DATA", sep=""), row.names = TRUE)



  #Hist1;
# SavePic(savefilename)
}


  #-------------------------------------------------------
  #-------------------------------------------------------
  #Weather ISSR Intersection of ISSR - Trajectory Cell Count, %, Contrail km

  Histo2<-function(){
  TRAJ_ISSR_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_TRAJEC_INTER.csv", colClasses  = "character")
  TRAJ_ISSR_DATA$W_ISSR_PTS <- as.numeric(TRAJ_ISSR_DATA$W_ISSR_PTS)
  TRAJ_ISSR_DATA$Contrail_per_day <- as.numeric(TRAJ_ISSR_DATA$Contrail_per_day)
  TRAJ_ISSR_DATA$CONUS_Intersect_pct <- as.numeric(TRAJ_ISSR_DATA$CONUS_Intersect_pct)
  TRAJ_ISSR_DATA$Contrails_km <- as.numeric(TRAJ_ISSR_DATA$Contrails_km)
  TRAJ_ISSR_DATA1<-TRAJ_ISSR_DATA

  Hist_out_SourceHist <<- character(50)
  Hist_out_Day_Night <<- character(50)
  Hist_out_FL <<- character(50)
  Hist_out_mean <<-numeric(50)
  Hist_out_median <<-numeric(50)
```

```
Hist_out_min <<-numeric(50)

Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)


k =1

j = 1


for (k in (1:3)){


for (DN in c('Day','Night'))

{

  i = 200

  for (i in seq(200,400, by = 10))

  {


    TRAJ_ISSR_DATA<-TRAJ_ISSR_DATA1;

    TRAJ_ISSR_DATA =subset(TRAJ_ISSR_DATA, DAY_NIGHT==DN);

    TRAJ_ISSR_DATA =subset(TRAJ_ISSR_DATA, FLIGHT_LEVEL==i);


    if (nrow(TRAJ_ISSR_DATA)>0){


    if (k==1){

    mainlabel=paste("Weather Cells with Contrial - ", DN ," - FL:" , i, sep="")

    savefile = "ContrailCellsCount"

    labelx = "Cell Count with Contrails"

    labely = "Days per year"

    TRAJ_ISSR_DATA$W_ISSR_PTS <- as.numeric(TRAJ_ISSR_DATA$W_ISSR_PTS); #Trajectory intersec Weather 16Aug15

    TRAJ_Hist <- aggregate(TRAJ_ISSR_DATA$W_ISSR_PTS, by=list(TRAJ_ISSR_DATA$W_DATE), FUN=sum);

    }


    if (k==2){

    mainlabel=paste("Pct of FL with Contrail - ", DN ," - FL:" , i, sep="")
```

```
savefile = "PctFLwithContrail"
labelx = "Pct of FL with Contrail"
labely = "Days per year"
TRAJ_ISSR_DATA$CONUS_Intersect_pct <- as.numeric(TRAJ_ISSR_DATA$CONUS_Intersect_pct); #% of FL with ISSR
TRAJ_Hist <- aggregate(TRAJ_ISSR_DATA$CONUS_Intersect_pct, by=list(TRAJ_ISSR_DATA$W_DATE), FUN=sum);
}

if (k==3){
mainlabel=paste("Contrails Generated [km] - ", DN ," - FL:" , i, sep="")
savefile = "Contrailskm"
labelx = "Contrails [km]"
labely = "Days per year"
TRAJ_ISSR_DATA$Contrails_km <- as.numeric(TRAJ_ISSR_DATA$Contrails_km); #Contrail km
TRAJ_Hist <- aggregate(TRAJ_ISSR_DATA$Contrails_km, by=list(TRAJ_ISSR_DATA$W_DATE), FUN=sum);
}

 #TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;



 #-----------
 #Build Ledgend
 led_text1=paste("Mean:", format(mean(TRAJ_Hist$x,na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
 led_text2=paste("Median:", format(median(TRAJ_Hist$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
 led_text3=paste("Min:", format(min(TRAJ_Hist$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
 led_text4=paste("Max:", format(max(TRAJ_Hist$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
 led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
 led_text6=paste("Var:", format(var(TRAJ_Hist$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")

 #Generate Histogram by Counts of Days

 Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        # xlim = c(0,00),
```

```
            cex.lab = 1.3)

      # H2 <-plot(Hist1)
      grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
      abline(v = mean(TRAJ_Hist$x,na.rm=TRUE ),col = "royalblue",lwd = 2)
      abline(v = median(TRAJ_Hist$x ,na.rm=TRUE), col = "red", lwd = 2)
      legend("topright",  legend =c(led_text1,led_text2,led_text3,led_text4,led_text5) , cex=0.8 ,
          col=c("royalblue", "red","black", "black", "black", "black" ))


      dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',savefile,"_", DN , " _FL",i, ".pdf", sep=""))



      Hist_out_SourceHist[j] =mainlabel;
      Hist_out_Day_Night[j] =DN;
      Hist_out_FL[j] = i;
      Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
      Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
      j=j+1




    }
  }
}

k = k+1
```

```
}# end K
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist2_",savefile,".csv", sep=""), row.names = TRUE)

# end K (run till here)

#-------------------------------------------------
#-------------------------------------------------
}

#-------------------------------------------------------
#-------------------------------------------------------
#Weather ISSR Intersection COMPARE Cell count , %, Contrail KM
Histo2Comp<-function(){

TRAJ_ISSR_DATA <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_TRAJEC_INTER.csv", colClasses  = "character")
TRAJ_ISSR_DATA$W_ISSR_PTS <- as.numeric(TRAJ_ISSR_DATA$W_ISSR_PTS)
TRAJ_ISSR_DATA$Contrail_per_day <- as.numeric(TRAJ_ISSR_DATA$Contrail_per_day)
TRAJ_ISSR_DATA$CONUS_Intersect_pct <- as.numeric(TRAJ_ISSR_DATA$CONUS_Intersect_pct)
TRAJ_ISSR_DATA$Contrails_km <- as.numeric(TRAJ_ISSR_DATA$Contrails_km)
TRAJ_ISSR_DATA1<-TRAJ_ISSR_DATA
```

```
Hist_out_SourceHist <<- character(50)

Hist_out_Day_Night <<- character(50)

Hist_out_FL <<- character(50)

Hist_out_mean <<-numeric(50)

Hist_out_median <<-numeric(50)

Hist_out_min <<-numeric(50)

Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)


k =1

j = 1


for (k in (1:3)){


 # for (DN in c('Day','Night'))

  {

    #i = 200

    #for (i in seq(200,400, by = 10))

    {


      TRAJ_ISSR_DATAD<-TRAJ_ISSR_DATA1;

      TRAJ_ISSR_DATAN<-TRAJ_ISSR_DATA1;

      TRAJ_ISSR_DATAD =subset(TRAJ_ISSR_DATAD, DAY_NIGHT=="Day");

      TRAJ_ISSR_DATAN =subset(TRAJ_ISSR_DATAN, DAY_NIGHT=="Night");

      #TRAJ_ISSR_DATA =subset(TRAJ_ISSR_DATA, FLIGHT_LEVEL==i);


      if (nrow(TRAJ_ISSR_DATA)>0){


        if (k==1){

          mainlabel=paste("Weather Cells with Contrial -  Day Vs Night", sep="")

          savefile2 = "ContrailCellsCountComp"

          labelx = "Cell Count with Contrails"
```

```
      labely = "Days per year"
      xlimsize = 22000
      TRAJ_ISSR_DATAD$W_ISSR_PTS <- as.numeric(TRAJ_ISSR_DATAD$W_ISSR_PTS); #Trajectory intersec Weather
16Aug15
      TRAJ_HistD <- aggregate(TRAJ_ISSR_DATAD$W_ISSR_PTS, by=list(TRAJ_ISSR_DATAD$W_DATE), FUN=sum);
      TRAJ_ISSR_DATAN$W_ISSR_PTS <- as.numeric(TRAJ_ISSR_DATAN$W_ISSR_PTS); #Trajectory intersec Weather
16Aug15
      TRAJ_HistN <- aggregate(TRAJ_ISSR_DATAN$W_ISSR_PTS, by=list(TRAJ_ISSR_DATAN$W_DATE), FUN=sum);
    }

    if (k==2){
      mainlabel=paste("Pct of CONUS with Contrail -  Day Vs Night" , sep="")
      savefile2 = "PctCONUSwithContrailComp"
      labelx = "Pct of CONUS with Contrail"
      labely = "Days per year"
      xlimsize = 0.0008
      TRAJ_ISSR_DATAD$CONUS_Intersect_pct <- as.numeric(TRAJ_ISSR_DATAD$CONUS_Intersect_pct); #% of FL with
ISSR
      TRAJ_HistD <- aggregate(TRAJ_ISSR_DATAD$CONUS_Intersect_pct, by=list(TRAJ_ISSR_DATAD$W_DATE),
FUN=sum);
      TRAJ_ISSR_DATAN$CONUS_Intersect_pct <- as.numeric(TRAJ_ISSR_DATAN$CONUS_Intersect_pct); #% of FL with
ISSR
      TRAJ_HistN <- aggregate(TRAJ_ISSR_DATAN$CONUS_Intersect_pct, by=list(TRAJ_ISSR_DATAN$W_DATE),
FUN=sum);
    }

    if (k==3){
      mainlabel=paste("Contrails Generated [km] - Day Vs Night", sep="")
      savefile2 = "ContrailskmComp"
      labelx = "Contrails [km]"
      labely = "Days per year"
      xlimsize = 300000
      TRAJ_ISSR_DATAD$Contrails_km <- as.numeric(TRAJ_ISSR_DATAD$Contrails_km); #Contrail km
```

```
  TRAJ_HistD <- aggregate(TRAJ_ISSR_DATAD$Contrails_km, by=list(TRAJ_ISSR_DATAD$W_DATE), FUN=sum);
  TRAJ_ISSR_DATAN$Contrails_km <- as.numeric(TRAJ_ISSR_DATAN$Contrails_km); #Contrail km
  TRAJ_HistN <- aggregate(TRAJ_ISSR_DATAN$Contrails_km, by=list(TRAJ_ISSR_DATAN$W_DATE), FUN=sum);
}

#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;


#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_HistD$x,na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_HistD$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_HistD$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_HistD$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_HistD$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_HistD$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c('Day:',led_text3,led_text2,led_text1,led_text4,led_text5)

led_text1=paste("Mean:", format(mean(TRAJ_HistN$x,na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_HistN$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_HistN$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_HistN$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_HistN$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_HistN$x,na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgN = c('Night:',led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_HistD$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xlimsize),ylim = c(0,50),
        cex.lab = 1.3);
Hist2 <-hist(TRAJ_HistN$x , col = "aquamarine3", breaks=20,
```

```
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,xlimsize),ylim = c(0,50),
        cex.lab = 1.3);
plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,xlimsize), ylim = c(0,50), #.2,0.6,1
    main = mainlabel,
    cex.sub =.8,
    xlab =labelx,
    ylab =labely)
plot( Hist2, col=rgb(.6,0.2,.8,1/2), xlim=c(0,xlimsize),ylim = c(0,50), add=T) # second

# H2 <-plot(Hist1)
#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_HistD$x, na.rm=TRUE ),col = rgb(.2,0.6,1,1/2),lwd = 2)
abline(v = median(TRAJ_HistD$x, na.rm=TRUE), col = rgb(.2,0.6,1,1/2), lwd = 2)
legend("topleft",  legend =ledgD , cex=c(.8)  ,
    #text.col=c(rgb(.2,0.6,1,1/2), "black", "black", "black", "black", "black", "black" ),
    box.col =rgb(.2,0.6,1,1/2),
    box.lwd = 4
)

#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_HistN$x, na.rm=TRUE ),col =rgb(.6,0.2,.8,1/2),lwd = 2)
abline(v = median(TRAJ_HistN$x, na.rm=TRUE), col = rgb(.6,0.2,.8,1/2), lwd = 2)
legend("topright",  legend =ledgN , cex=c(.8) ,
    #text.col=c("royalblue", "red","black", "black", "black", "black" )
    box.col=rgb(.6,0.2,.8,1/2),
    box.lwd = 4
)



dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',savefile2,".pdf", sep=""))
```

```
        Hist_out_SourceHist[j] =mainlabel;
        Hist_out_Day_Night[j] =DN;
        Hist_out_FL[j] = i;
        Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
        Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
        Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
        Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
        Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
        Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
        j=j+1





      }
    }
  }

  k = k+1
}# end K
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);
```

```
write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist2_",savefile2,".csv", sep=""), row.names = TRUE)

# end K (run till here)

#--------------------------------------------------
#--------------------------------------------------
}

# Build ISSR combined HIST3 DAY VS Night

Histo3<-function(){
ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");
ISSR_CNT$W3D_POINTS <- as.numeric(ISSR_CNT$W3D_POINTS)
ISSR_CNT$FLIGHT_LEVEL <- as.numeric(ISSR_CNT$FLIGHT_LEVEL)

ISSR_CNT1<-ISSR_CNT

Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)

j = 1
```

```
mainlabel=paste("Yearly ISSR Distribution within US Airspace", sep="")

savefilename=paste("ISSR_Dist" , DN, sep="")

labelx = "ISSR weather cells"
labely = "Number of Days"


ISSR_CNTD<-ISSR_CNT1
ISSR_CNTN<-ISSR_CNT1


ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/13
ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/11
# ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/16 # FL
#  ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/16 # FL

ISSR_CNTD =subset(ISSR_CNTD, Day_Night=="Day");
#ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);
ISSR_CNTN =subset(ISSR_CNTN, Day_Night=="Night");
#ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);



  ISSR_CNTD <- aggregate(ISSR_CNTD$W3D_POINTS, by=list(ISSR_CNTD$DATE),  FUN=sum);
  #ISSR_Hist$x <- as.numeric(ISSR_Hist$x) ;
  ISSR_CNTN <- aggregate(ISSR_CNTN$W3D_POINTS, by=list(ISSR_CNTN$DATE),  FUN=sum);

  #-----------

  ISSR_CNTD$x <- as.numeric(ISSR_CNTD$x)
```

```
ISSR_CNTN$x <- as.numeric(ISSR_CNTN$x)


#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(ISSR_CNTD$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c('Day:',led_text3,led_text2,led_text1,led_text4,led_text5)


led_text1=paste("Mean:", format(mean(ISSR_CNTN$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgN = c('Night:',led_text3,led_text2,led_text1,led_text4,led_text5)
#Generate Histogram by Counts of Days


Hist1 <-hist(ISSR_CNTD$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
         xlim = c(0,300000),ylim = c(0,60),
        cex.lab = 1.3);
Hist2 <-hist(ISSR_CNTN$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,300000),ylim = c(0,60),
        cex.lab = 1.3);
plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,300000), ylim = c(0,50), #.2,0.6,1
   # main = "ISSR Cell Count",
    #sub="(including FL200 to FL400)",
   main = "Yearly ISSR Cell Count in US Airspace\n(FL200 to FL400)",
```

```
     cex.sub =.8,
      xlab = "ISSR Cell Count",
      ylab ="Number of Days")
plot( Hist2, col=rgb(.6,0.2,.8,1/2), xlim=c(0,300000),ylim = c(0,50), add=T) # second

# H2 <-plot(Hist1)
#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTD$x, na.rm=TRUE ),col = rgb(.2,0.6,1,1/2),lwd = 2)
abline(v = median(ISSR_CNTD$x, na.rm=TRUE), col = rgb(.2,0.6,1,1/2), lwd = 2)
legend("topleft",  legend =ledgD , cex=c(.8)  ,
       #text.col=c(rgb(.2,0.6,1,1/2), "black", "black", "black", "black", "black", "black" ),
       box.col =rgb(.2,0.6,1,1/2),
       box.lwd = 4
       )

#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTN$x, na.rm=TRUE ),col =rgb(.6,0.2,.8,1/2),lwd = 2)
abline(v = median(ISSR_CNTN$x, na.rm=TRUE), col = rgb(.6,0.2,.8,1/2), lwd = 2)
legend("topright",  legend =ledgN , cex=c(.8) ,
       #text.col=c("royalblue", "red","black", "black", "black", "black" )
       box.col=rgb(.6,0.2,.8,1/2),
       box.lwd = 4
       )

dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\ISSR_CNT_Day_Night.pdf', sep=""))



Hist_out_SourceHist[j] =mainlabel
Hist_out_Day_Night[j] =DN
Hist_out_FL[j] = i
Hist_out_mean[j] = mean(ISSR_CNT$x, na.rm=TRUE)
```

```
    Hist_out_median[j] = median(ISSR_CNT$x, na.rm=TRUE)
    Hist_out_min[j] = min(ISSR_CNT$x, na.rm=TRUE)
    Hist_out_max[j] =max(ISSR_CNT$x, na.rm=TRUE)
    Hist_out_StdDev[j] =sd(ISSR_CNT$x, na.rm=TRUE)
    Hist_out_var[j] = var(ISSR_CNT$x, na.rm=TRUE)
    j=j+1

Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_3",savefilename,".csv", sep=""), row.names = TRUE)

#--------------------------------------------------------
#--------------------------------------------------------
}
```

```
# Build ISSR ALL

Histo3<-function(){
  ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");
  ISSR_CNT$W3D_POINTS <- as.numeric(ISSR_CNT$W3D_POINTS)
  ISSR_CNT$FLIGHT_LEVEL <- as.numeric(ISSR_CNT$FLIGHT_LEVEL)

  ISSR_CNT1<-ISSR_CNT



  Hist_out_SourceHist <<- character(50)
  Hist_out_Day_Night <<- character(50)
  Hist_out_FL <<- character(50)
  Hist_out_mean <<-numeric(50)
  Hist_out_median <<-numeric(50)
  Hist_out_min <<-numeric(50)
  Hist_out_max <<-numeric(50)
  Hist_out_StdDev <<-numeric(50)
  Hist_out_var <<-numeric(50)

  j = 1



  mainlabel=paste("ISSR Distribution", sep="")

  savefilename="ISSR_Dist_ALL"

  labelx = "ISSR weather cells"
  labely = "Number of Days"
```

```
ISSR_CNTD<-ISSR_CNT1
ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/ 24

ISSR_CNTD <- aggregate(ISSR_CNTD$W3D_POINTS, by=list(ISSR_CNTD$DATE),  FUN=sum);


#-----------

ISSR_CNTD$x <- as.numeric(ISSR_CNTD$x)



#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(ISSR_CNTD$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c('Day:',led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(ISSR_CNTD$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,300000),ylim = c(0,60),
        cex.lab = 1.3);

plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,300000), ylim = c(0,50), #.2,0.6,1
   # main = "ISSR Cell Count",
   #sub="(including FL200 to FL400)",
   main = "Yearly ISSR Cell Count in US Airspace\n(FL200 to FL400)",
```

```
     cex.sub =.8,
     xlab = "ISSR Cell Count",
     ylab ="Number of Days")

# H2 <-plot(Hist1)
#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTD$x, na.rm=TRUE ),col = rgb(.2,0.6,1,1/2),lwd = 2)
abline(v = median(ISSR_CNTD$x, na.rm=TRUE), col = rgb(.2,0.6,1,1/2), lwd = 2)
legend("topleft",  legend =ledgD , cex=c(.8)  ,
      #text.col=c(rgb(.2,0.6,1,1/2), "black", "black", "black", "black", "black", "black" ),
      box.col =rgb(.2,0.6,1,1/2),
      box.lwd = 4
)


dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\TotCellCountDay_Night.pdf', sep=""))



Hist_out_SourceHist[j] =mainlabel
Hist_out_Day_Night[j] =DN
Hist_out_FL[j] = i
Hist_out_mean[j] = mean(ISSR_CNT$x, na.rm=TRUE)
Hist_out_median[j] = median(ISSR_CNT$x, na.rm=TRUE)
Hist_out_min[j] = min(ISSR_CNT$x, na.rm=TRUE)
Hist_out_max[j] =max(ISSR_CNT$x, na.rm=TRUE)
Hist_out_StdDev[j] =sd(ISSR_CNT$x, na.rm=TRUE)
Hist_out_var[j] = var(ISSR_CNT$x, na.rm=TRUE)
j=j+1



Hist_Data_Out <- data.frame (
```

```
    Hist_out_SourceHist,
    Hist_out_Day_Night,
    Hist_out_FL,
    Hist_out_mean,
    Hist_out_median,
    Hist_out_min,
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_3",savefilename,".csv", sep=""), row.names = TRUE)

  #-------------------------------------------------------
  #-------------------------------------------------------
}

  # Build ISSR Distritribution % COMPARE  HIST5_pctconus

Histo3Comp_pct<-function(){
  ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");
  ISSR_CNT$W3D_POINTS <- as.numeric(ISSR_CNT$W3D_POINTS)
  ISSR_CNT$FLIGHT_LEVEL <- as.numeric(ISSR_CNT$FLIGHT_LEVEL)

  ISSR_CNT1<-ISSR_CNT

  Hist_out_SourceHist <<- character(50)
  Hist_out_Day_Night <<- character(50)
  Hist_out_FL <<- character(50)
  Hist_out_mean <<-numeric(50)
```

```
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)

j = 1



 mainlabel=paste("ISSR Distribution: ", DN ," - FL:" , i, sep="")

 savefilename="ISSR_Dist"

 labelx = "ISSR weather cells"
 labely = "Days per year"


 ISSR_CNTD<-ISSR_CNT1
 ISSR_CNTN<-ISSR_CNT1


 ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/ 31613296  # 151987*13HR day
 ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/ 26749712  #151987*11HR night
 #ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/16 # FL
 #ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/16 # FL

 ISSR_CNTD =subset(ISSR_CNTD, Day_Night=="Day");
 #ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);
 ISSR_CNTN =subset(ISSR_CNTN, Day_Night=="Night");
 #ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);
```

```
ISSR_CNTD <- aggregate(ISSR_CNTD$W3D_POINTS, by=list(ISSR_CNTD$DATE),  FUN=sum);
#ISSR_Hist$x <- as.numeric(ISSR_Hist$x) ;
ISSR_CNTN <- aggregate(ISSR_CNTN$W3D_POINTS, by=list(ISSR_CNTN$DATE),  FUN=sum);

#-----------

ISSR_CNTD$x <- as.numeric(ISSR_CNTD$x)
ISSR_CNTN$x <- as.numeric(ISSR_CNTN$x)

#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(ISSR_CNTD$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c('Day:',led_text3,led_text2,led_text1,led_text4,led_text5)

led_text1=paste("Mean:", format(mean(ISSR_CNTN$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgN = c('Night:',led_text3,led_text2,led_text1,led_text4,led_text5)
#Generate Histogram by Counts of Days

Hist1 <-hist(ISSR_CNTD$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,0.15),
        ylim = c(0,60),
```

```
        cex.lab = 1.3);
Hist2 <-hist(ISSR_CNTN$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,0.15),
        ylim = c(0,60),
        cex.lab = 1.3);
plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,0.15), ylim = c(0,50),
    main = "Yearly % of US Airspace with ISSR\n(FL200 to FL400)",
    #sub="(including FL200 to FL400)",
    xlab = "% CONUS with ISSR",
    ylab ="Number of Days")
plot( Hist2, col=rgb(.6,0.2,.8,1/2), xlim=c(0,0.15),ylim = c(0,50), add=T) # second

# H2 <-plot(Hist1)
#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTD$x, na.rm=TRUE ),col = rgb(.2,0.6,1,1/2),lwd = 2)
abline(v = median(ISSR_CNTD$x, na.rm=TRUE), col = rgb(.2,0.6,1,1/2), lwd = 2)
legend("topleft",  legend =ledgD , cex=0.8 ,
     col=c("royalblue", "red","black", "black", "black", "black" ),
     box.col =rgb(.2,0.6,1,1/2),
     box.lwd = 4)

#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTN$x, na.rm=TRUE ),col =rgb(.6,0.2,.8,1/2),lwd = 2)
abline(v = median(ISSR_CNTN$x, na.rm=TRUE), col = rgb(.6,0.2,.8,1/2), lwd = 2)
legend("topright",  legend =ledgN , cex=0.8 ,
     col=c("royalblue", "red","black", "black", "black", "black" ),
     box.col =rgb(.6,0.2,.8,1/2),
     box.lwd = 4)

dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\PCTCellCountDay_Night.pdf', sep=""))
```

```
    Hist_out_SourceHist[j] =mainlabel

    Hist_out_Day_Night[j] =DN

    Hist_out_FL[j] = i

    Hist_out_mean[j] = mean(ISSR_CNT$x, na.rm=TRUE)

    Hist_out_median[j] = median(ISSR_CNT$x, na.rm=TRUE)

    Hist_out_min[j] = min(ISSR_CNT$x, na.rm=TRUE)

    Hist_out_max[j] =max(ISSR_CNT$x, na.rm=TRUE)

    Hist_out_StdDev[j] =sd(ISSR_CNT$x, na.rm=TRUE)

    Hist_out_var[j] = var(ISSR_CNT$x, na.rm=TRUE)

    j=j+1




Hist_Data_Out <- data.frame (

  Hist_out_SourceHist,

  Hist_out_Day_Night,

  Hist_out_FL,

  Hist_out_mean,

  Hist_out_median,

  Hist_out_min,

  Hist_out_max,

  Hist_out_StdDev,

  Hist_out_var

  ,stringsAsFactors=FALSE);


write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_3",savefilename,".csv", sep=""), row.names = TRUE)


#-------------------------------------------------------

#-------------------------------------------------------

}
```

```
#-------------------------------------------------

#-------------------------------------------------


# ISSR Day and Night

Histo6 <- function(){

ISSR_CNT <- read.csv("E:\\\...\\proposal\\EXPER\\ISSR_CNT.csv", colClasses  = "character");

ISSR_CNT$W3D_POINTS <- as.numeric(ISSR_CNT$W3D_POINTS)

ISSR_CNT$FLIGHT_LEVEL <- as.numeric(ISSR_CNT$FLIGHT_LEVEL)


ISSR_CNT1<-ISSR_CNT


Hist_out_SourceHist <<- character(50)

Hist_out_Day_Night <<- character(50)

Hist_out_FL <<- character(50)

Hist_out_mean <<-numeric(50)

Hist_out_median <<-numeric(50)

Hist_out_min <<-numeric(50)

Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)


j = 1


mainlabel=paste("ISSR Distribution: ", DN ," - FL:" , i, sep="")
```

```
savefilename="ISSR_Dist"

labelx = "ISSR weather cells"
labely = "Number of Days"


ISSR_CNTD<-ISSR_CNT1
ISSR_CNTN<-ISSR_CNT1



ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/ 13
ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/ 11
# ISSR_CNTD$W3D_POINTS <-ISSR_CNTD$W3D_POINTS/16 # FL
#  ISSR_CNTN$W3D_POINTS <-ISSR_CNTN$W3D_POINTS/16 # FL

ISSR_CNTD =subset(ISSR_CNTD, Day_Night=="Day");
#ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);
ISSR_CNTN =subset(ISSR_CNTN, Day_Night=="Night");
#ISSR_CNT =subset(ISSR_CNTD, FLIGHT_LEVEL==i);




ISSR_CNTD <- aggregate(ISSR_CNTD$W3D_POINTS, by=list(ISSR_CNTD$DATE),  FUN=sum);
#ISSR_Hist$x <- as.numeric(ISSR_Hist$x) ;
ISSR_CNTN <- aggregate(ISSR_CNTN$W3D_POINTS, by=list(ISSR_CNTN$DATE),  FUN=sum);

#-----------

ISSR_CNTD$x <- as.numeric(ISSR_CNTD$x)
ISSR_CNTN$x <- as.numeric(ISSR_CNTN$x)

#-----------
#Build Ledgend
```

```
led_text1=paste("Mean:", format(mean(ISSR_CNTD$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTD$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgD = c('Day:',led_text3,led_text2,led_text1,led_text4,led_text5)

led_text1=paste("Mean:", format(mean(ISSR_CNTN$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(ISSR_CNTN$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
ledgN = c('Night:',led_text3,led_text2,led_text1,led_text4,led_text5)
#Generate Histogram by Counts of Days

Hist1 <-hist(ISSR_CNTD$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,500000),ylim = c(0,60),
        cex.lab = 1.3);
Hist2 <-hist(ISSR_CNTN$x , col = "aquamarine3", breaks=20,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(0,500000),ylim = c(0,60),
        cex.lab = 1.3);
plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(0,350000), ylim = c(0,50), #.2,0.6,1
    # main = "ISSR Cell Count",
    #sub="(including FL200 to FL400)",
    main = "Yearly ISSR Cell Count in US Airspace\n(FL200 to FL400)",

    cex.sub =.8,
    xlab = "ISSR Cell Count",
    ylab ="Number of Days")
```

```
plot( Hist2, col=rgb(.6,0.2,.8,1/2), xlim=c(0,35000),ylim = c(0,50), add=T) # second


# H2 <-plot(Hist1)
#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTD$x, na.rm=TRUE ),col = rgb(.2,0.6,1,1/2),lwd = 2)
abline(v = median(ISSR_CNTD$x, na.rm=TRUE), col = rgb(.2,0.6,1,1/2), lwd = 2)
legend("topleft",  legend =ledgD , cex=c(.8)  ,
      #text.col=c(rgb(.2,0.6,1,1/2), "black", "black", "black", "black", "black", "black" ),
      box.col =rgb(.2,0.6,1,1/2),
      box.lwd = 4
)


#grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(ISSR_CNTN$x, na.rm=TRUE ),col =rgb(.6,0.2,.8,1/2),lwd = 2)
abline(v = median(ISSR_CNTN$x, na.rm=TRUE), col = rgb(.6,0.2,.8,1/2), lwd = 2)
legend("topright",  legend =ledgN , cex=c(.8) ,
      #text.col=c("royalblue", "red","black", "black", "black", "black" )
      box.col=rgb(.6,0.2,.8,1/2),
      box.lwd = 4
)


dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\TotCellCountDay_Night.pdf', sep=""))




Hist_out_SourceHist[j] =mainlabel
Hist_out_Day_Night[j] =DN
Hist_out_FL[j] = i
Hist_out_mean[j] = mean(ISSR_CNT$x, na.rm=TRUE)
Hist_out_median[j] = median(ISSR_CNT$x, na.rm=TRUE)
Hist_out_min[j] = min(ISSR_CNT$x, na.rm=TRUE)
Hist_out_max[j] =max(ISSR_CNT$x, na.rm=TRUE)
Hist_out_StdDev[j] =sd(ISSR_CNT$x, na.rm=TRUE)
```

```
 Hist_out_var[j] = var(ISSR_CNT$x, na.rm=TRUE)
 j=j+1



 Hist_Data_Out <- data.frame (
   Hist_out_SourceHist,
   Hist_out_Day_Night,
   Hist_out_FL,
   Hist_out_mean,
   Hist_out_median,
   Hist_out_min,
   Hist_out_max,
   Hist_out_StdDev,
   Hist_out_var
   ,stringsAsFactors=FALSE);

 write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist_3",savefilename,".csv", sep=""), row.names = TRUE)

 #--------------------------------------------------------
 #--------------------------------------------------------
 }



 #--------------------------------------------------------
 #--------------------------------------------------------



#fl BY fid
{{

 sqlstr1 = paste("Select * from
```

```
(
 SELECT FID , -- count(distinct(FID)),
 sum(POS2_ID_CNT_CONTRAILS) Contrail_pts,
 INCLUDE,
 distance,
 FID1,
 sum(CNT_POS2_ID) Pos2_cnt,
 sum(POS2_ID_FULL_TRAJECTORY ) full_cnt
 FROM FLIGHT_STATS
 where include = 1
 and distance is not null
 group by
 FID,INCLUDE, distance, FID1 ) a
 join
 (
 select
 (ADB_ID ||  CALL_ ) FID3,
 STATS_MODE(round(ALTITUDE,-1)) mode_FL
 from
 TRAFFIC_ADB_V
 group by
 (ADB_ID ||  CALL_ )) FL2
 on a.FID = FL2.FID3" , sep = "")

FL_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

# CONTRAIL_DATA1 <- read.csv("E:\\\...\\P2_IISE_Stats\\CONTRAIL_DATA2.csv", colClasses  = "character")
FL_DATA<-FL_DATA1
FL_DATA$INCLUDE <- as.numeric(CONTRAIL_DATA$INCLUDE)
FL_DATA1<-FL_DATA
FL_DATA<-FL_DATA1

 #write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)
```

```
  #-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)



      mainlabel=paste("Flight Count by Flight Level", sep="")
      seclabel = "FL 200 - FL 400"
      xcellStart = 200
      xcelllimit = 400
      ycelllimit = 2000
      ycellStart = 0}
    labelx = "Flight Level"
    labely = "Count of Flights"
    TrajFile = "FL_by_FID"



    FL_DATA =subset(FL_DATA, INCLUDE==1);
    FL_DATA =subset(FL_DATA, MODE_FL>=200);
    FL_DATA =subset(FL_DATA, MODE_FL<=400);

    TRAJ_Hist <- aggregate(FL_DATA$MODE_FL, by=list(FL_DATA$FID), FUN=sum);
```

```
#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
#  led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , " mW/m2",
sep="")
led_text7 = "
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
         main=mainlabel, las=0, xlab = labelx, ylab = labely,
         xlim = c(xcellStart,xcelllimit),
         cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
     col=c("royalblue", "red","black", "black", "black", "black" ))


plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
    # main = "ISSR Cell Count",
    #sub="(including FL200 to FL400)",
    main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),
```

```
        cex.sub =.8,
        xlab = labelx,
        ylab =labely)

    abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
    abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
    legend("topright",  legend =ledgD , cex=0.8 ,
         col=c("royalblue", "red","black", "black", "black", "black" ))



    dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))



    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1;

Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
```

```
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
  # end K (run till here)



#------------
#------------
}



#RF BY fid
{



  #--------------------------------------------------------
  #--------------------------------------------------------



  {


    sqlstr1 = paste("SELECT
SCENARIO,
            FLIGHT_ID,
            sum(TOTAL_RF_W)/510072000000 NRF, -- surface of earth to get W/m2  510,072,000,000,000 then * 1000 to get
miliwatts
            DATE_YEAR,
            DAY_NIGHT,
            LAT,
            LON,
            b.fl_mode,
```

```
            sum(ISSR_CELLS) Cells,
            sum(ISSR_CELLS) * 13 * 0.539957 nm
            FROM RF_RESULTS_T FR join
            (select fid fsfid, fl_mode from flight_stats ) b
            on (fr.FLIGHT_ID = b.fsfid)
            where scenario = 'Scenario_1'
            group by
            SCENARIO,
            FLIGHT_ID,
            DATE_YEAR,
            DAY_NIGHT,
            LAT,
            LON,
            b.fl_mode
            ;" , sep = "")

sqlstr1 = paste("SELECT
FID,
DISTANCE,
POS2_ID_CNT_CONTRAILS,
INCLUDE,
POS2_ID_FULL_TRAJECTORY,
FL_MODE
FROM FLIGHT_STATS" , sep = "")

FL_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

FL_DATA1 <- read.csv("E:\\\...\\P2_IISE_Stats\\RF_by_Flight.csv", colClasses  = "character")
FL_DATA<-FL_DATA1
FL_DATA$NRF <- as.numeric(FL_DATA$NRF)
FL_DATA$W_Date <- as.Date(FL_DATA$DATE_YEAR,"%d-%b-%y")
FL_DATA$CELLS <- as.numeric(FL_DATA$CELLS,"%d-%b-%y")
FL_DATA$NM <- as.numeric(FL_DATA$NM,"%d-%b-%y")
```

```
FL_DATA1<-FL_DATA
FL_DATA<-FL_DATA1

#write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)
#-----------------------------------
k = 1
j = 1
Hist_out_SourceHist <<- character(50)
Hist_out_Day_Night <<- character(50)
Hist_out_FL <<- character(50)
Hist_out_mean <<-numeric(50)
Hist_out_median <<-numeric(50)
Hist_out_min <<-numeric(50)
Hist_out_max <<-numeric(50)
Hist_out_StdDev <<-numeric(50)
Hist_out_var <<-numeric(50)



mainlabel=paste("Net Radiative Forcing by Flight", sep="")
seclabel = ""
xcellStart = 0
xcelllimit = 1

ycellStart = 0
ycelllimit = 5000
}
labelx = "Flights"
labely = "Net Radiative Forcing"
TrajFile = "FL_by_FID"
```

```
#FL_DATA =subset(FL_DATA, INCLUDE==1);
#FL_DATA =subset(FL_DATA, MODE_FL>=200);
FL_DATA =subset(FL_DATA, SCENARIO=='Scenario_1');

TRAJ_Hist <- aggregate(FL_DATA$NRF, by=list(FL_DATA$FLIGHT_ID), FUN=sum);
TRAJ_Hist <- bin(FL_DATA$NRF,nbins = 80);

#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , " mW/m2",
sep="")
led_text7 = ''
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(xcellStart,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
```

```
       col=c("royalblue", "red","black", "black", "black", "black" ))



plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
      # main = "ISSR Cell Count",
      #sub="(including FL200 to FL400)",
      main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),

      cex.sub =.8,
      xlab = labelx,
      ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
       col=c("royalblue", "red","black", "black", "black", "black" ))

barplot(h$counts, horiz = TRUE)



dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))



Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
j=j+1;
```

```
  Hist_Data_Out <- data.frame (
    Hist_out_SourceHist,
    Hist_out_Day_Night,
    Hist_out_FL,
    Hist_out_mean,
    Hist_out_median,
    Hist_out_min,
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
  # end K (run till here)


  #------------
  #------------


}



#counts of FIDs included and not
{
  #-------------------------------------------------------
  #-------------------------------------------------------
  {


    sqlstr1 = paste("select distinct fid, date_time, include from
(SELECT FID FID1,
                    DISTANCE,
                    DAY_NIGHT,
                    POS2_ID_CNT,
```

```
            INCLUDE
            FROM FLIGHT_DIST_NM ) a
            ,
            (SELECT
            FID,
            date_time,
            --  CUR_LAT,
            --  CUR_LON,
            count(distinct(POS2_ID)) Cont_pts,
            sum(RF_W) NRF
            FROM CONTRAIL_RF_1
            group by fid,date_time ) b
            where
            (a.FID1=b.fid)" , sep = "")

 FL_DATA1<<-dbGetQuery(ORA_Con,sqlstr1)

 FL_DATA<-FL_DATA1
# FL_DATA$NRF <- as.numeric(FL_DATA$NRF)
 FL_DATA1$DATE1 <- substring(FL_DATA1$DATE_TIME,1,10)
 FL_DATA$W_Date <- as.Date(FL_DATA$DATE1,"%Y-%m-%d")
#  FL_DATA$CELLS <- as.numeric(FL_DATA$CELLS,"%d-%b-%y")
# FL_DATA$NM <- as.numeric(FL_DATA$NM,"%d-%b-%y")
 FL_DATA1<-FL_DATA


 #write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)
 #-----------------------------------
 k = 1
 j = 1
 Hist_out_SourceHist <<- character(50)
 Hist_out_Day_Night <<- character(50)
 Hist_out_FL <<- character(50)
```

```
  Hist_out_mean <<-numeric(50)
  Hist_out_median <<-numeric(50)
  Hist_out_min <<-numeric(50)
  Hist_out_max <<-numeric(50)
  Hist_out_StdDev <<-numeric(50)
  Hist_out_var <<-numeric(50)




  mainlabel=paste("Net Radiative Forcing by Flight", sep="")
  seclabel = ""
  xcellStart = 0
  xcelllimit = 5000

  ycellStart = 0
  ycelllimit = 5000
}
labelx = "Flights"
labely = "Net Radiative Forcing"
TrajFile = "FL_by_FID"




FL_DATA =subset(FL_DATA, INCLUDE==1);
#FL_DATA =subset(FL_DATA, MODE_FL>=200);
FL_DATA =subset(FL_DATA, SCENARIO=='Scenario_1');

TRAJ_Hist <- aggregate(FL_DATA$FID, by=list(FL_DATA$W_Date), FUN=sum);
TRAJ_Hist<- data.frame(table(FL_DATA$W_Date))
#TRAJ_Hist <- bin(FL_DATA$NRF,nbins = 80);

#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;
#-----------
```

```
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , " mW/m2",
sep="")
led_text7 = ''
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(xcellStart,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))


plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
   # main = "ISSR Cell Count",
   #sub="(including FL200 to FL400)",
   main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),

   cex.sub =.8,
```

```
      xlab = labelx,
      ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
       col=c("royalblue", "red","black", "black", "black", "black" ))



ggplot(TRAJ_Hist, aes(Var1,Freq), Freq)
barplot(h$counts, horiz = TRUE)


dev.print(pdf, paste('E:\\\...\\P2_IISE_Stats\\',TrajFile,"_", DN , ".pdf", sep=""))


Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
j=j+1;

Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
```

```
    Hist_out_median,
    Hist_out_min,
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
  # end K (run till here)



  #------------
  #------------



}



#RESULTS1
{

  #-------------------------------------------------------
  #-------------------------------------------------------



  {



    RF_DATA1 <- read.csv("E:\\\...\\proposal\\Final_Results\\Results_RF1.csv", colClasses  = "character")
    RF_DATA<-RF_DATA1
    RF_DATA$TOTAL_RF_W <- as.numeric(RF_DATA$TOTAL_RF_W)
    RF_DATA$NRF <- as.numeric(RF_DATA$NRF)
    RF_DATA$DISTANCE <- as.numeric(RF_DATA$DISTANCE)
```

```
RF_DATA$PCT_NRF <- as.numeric(RF_DATA$PCT_NRF)

RF_DATA$PCT_Flights <- as.numeric(RF_DATA$PCT_Flights)

#FL_DATA$W_Date <- as.Date(FL_DATA$DATE_YEAR,"%d-%b-%y")

RF_DATA$PCT_NRF <- (RF_DATA$PCT_NRF)*100

RF_DATA$PCT_Flights <- (RF_DATA$PCT_Flights)*100

RF_DATA1<-RF_DATA




# FL_DATA$NRF <- as.numeric(FL_DATA$NRF)

#FL_DATA1$DATE1 <- substring(FL_DATA1$DATE_TIME,1,10)

#FL_DATA$W_Date <- as.Date(FL_DATA$DATE1,"%Y-%m-%d")

#  FL_DATA$CELLS <- as.numeric(FL_DATA$CELLS,"%d-%b-%y")

# FL_DATA$NM <- as.numeric(FL_DATA$NM,"%d-%b-%y")


#write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)

#-----------------------------------

k = 1

j = 1

Hist_out_SourceHist <<- character(50)

Hist_out_Day_Night <<- character(50)

Hist_out_FL <<- character(50)

Hist_out_mean <<-numeric(50)

Hist_out_median <<-numeric(50)

Hist_out_min <<-numeric(50)

Hist_out_max <<-numeric(50)

Hist_out_StdDev <<-numeric(50)

Hist_out_var <<-numeric(50)


RF_DATA<-RF_DATA1


mainlabel=paste("Net Radiative Forcing by Flight", sep="")

seclabel = ""
```

```
  xcellStart = 0
  xcelllimit = 5000

  ycellStart = 0
  ycelllimit = 5000
}
labelx = "% Flights / day"
labely = "% Net Radiative Forcing"
TrajFile = "pct_fl_pct_RF"




#FL_DATA =subset(FL_DATA, MODE_FL>=200);
RF_DATA =subset(RF_DATA, SCENARIO=='Scenario_1');




#abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
#abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
#legend("topright",  legend =ledgD , cex=0.8 ,
#      col=c("royalblue", "red","black", "black", "black", "black" ))


# add a title and subtitle
title("% RF vs % Flights", "")


plot(RF_DATA$PCT_Flights,RF_DATA$PCT_NRF, type = "S" ,xlab = labelx, ylab = labely,
    #col=rgb(.1,0.6,1,1)
    col = "dodgerblue4"
    )
```

```
abline( v = 10 ,col = "royalblue",lwd = 1)
abline( v = 5 ,col = "royalblue",lwd = 1)
abline( v = 3 ,col = "royalblue",lwd = 1)
abline( v = 2 ,col = "royalblue",lwd = 1)
abline( v = 1 ,col = "royalblue",lwd = 1)



led_text1=paste(" 1% (  54 flights ) ->  4.82% NRF" , sep="")
led_text2=paste(" 2% ( 108 flights ) ->  8.21% NRF" , sep="")
led_text3=paste(" 3% ( 163 flights ) -> 11.07% NRF", sep="")
led_text4=paste(" 5% ( 270 flights ) -> 15.96% NRF", sep="")
led_text5=paste("10% ( 541 flights ) -> 25.61% NRF" , sep="")
#led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
#led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , " mW/m2",
sep="")
led_text7 = "
ledgD = c(led_text1,led_text2,led_text3,led_text4,led_text5)

#legend("bottomright",  legend =ledgD , cex=0.8 , col=c("royalblue", "red","black", "black", "black", "black" ))
legend(53, 53,  legend =ledgD , cex=0.8 ,
      col=c("royalblue", "red","black", "black", "black", "black" ))

title("% RF vs % Flights")



dev.print(pdf, paste('E:\\\...\\proposal\\Final_Results\\',TrajFile,"_", DN , ".pdf", sep=""))



Hist_out_SourceHist[j] =mainlabel;
Hist_out_Day_Night[j] =DN;
Hist_out_FL[j] = i;
Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
```

```
Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
j=j+1;

Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
  Hist_out_max,
  Hist_out_StdDev,
  Hist_out_var
  ,stringsAsFactors=FALSE);

write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\",TrajFile,"_",DN,".csv", sep=""), row.names = TRUE)
# end K (run till here)


#------------
#------------

}


#RF_FB_2
{


#-------------------------------------------------------
```

```
#------------------------------------------------------

{

  RF_DATA1 <- read.csv("E:\\\...\\proposal\\Final_Results\\RF_FB_V2_values.csv", colClasses  = "character")
  RF_DATA<-RF_DATA1
  RF_DATA$CNT <- as.numeric(RF_DATA$CNT)
  RF_DATA$NRF <- as.numeric(RF_DATA$NRF)
  RF_DATA$NMC <- as.numeric(RF_DATA$NMC)
  RF_DATA$FB_TOTAL <- as.numeric(RF_DATA$FB_TOTAL)
  RF_DATA$FB_C <- as.numeric(RF_DATA$FB_C)
  RF_DATA$FB_D <- as.numeric(RF_DATA$FB_D)
  RF_DATA$FB_CO <- as.numeric(RF_DATA$FB_CO)
  RF_DATA$FB_RF <- as.numeric(RF_DATA$FB_RF)
  RF_DATA$FL_CHG2 <- as.numeric(RF_DATA$FL_CHG2)
  RF_DATA$Include <- as.numeric(RF_DATA$Include)
  RF_DATA$Limits <- as.numeric(RF_DATA$Limits)
  RF_DATA$NRF_per_Flight <- as.numeric(RF_DATA$NRF_per_Flight)
  RF_DATA$NRF_per_Flight_per_year <- as.numeric(RF_DATA$NRF_per_Flight)/276
  RF_DATA$FB_TOTAL_Per_Flight <- as.numeric(RF_DATA$FB_TOTAL_Per_Flight)


  RF_DATA1<-RF_DATA


  # FL_DATA$NRF <- as.numeric(FL_DATA$NRF)
  #FL_DATA1$DATE1 <- substring(FL_DATA1$DATE_TIME,1,10)
  #FL_DATA$W_Date <- as.Date(FL_DATA$DATE1,"%Y-%m-%d")
  #  FL_DATA$CELLS <- as.numeric(FL_DATA$CELLS,"%d-%b-%y")
  # FL_DATA$NM <- as.numeric(FL_DATA$NM,"%d-%b-%y")

  #write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)
```

```
#-----------------------------------

k =1
j = 1

for (k in (1:2)){
 for (s in c('Winter','Summer','Autumn','Spring')){

    RF_DATA<-RF_DATA1
    RF_DATA =subset(RF_DATA, Include==1);
    RF_DATA =subset(RF_DATA, Limits==1);
    RF_DATA =subset(RF_DATA, RF_SOURCE=='FL_AT');
    RF_DATA =subset(RF_DATA, SEASON==s);

    if (nrow(RF_DATA)>0){

      if (k==1){



        mainlabel=paste("Net Radiative Forcing by Flight",'_' ,s, sep="")
        seclabel = ""
        xcellStart = 0
        xcelllimit = 0.005
        ycellStart = 0
        ycelllimit = 140

        labelx = "Net Radiative Forcing"
        labely = "Flight Count"
        savefile =  paste("NRF_Flight" ,'_' ,s, sep="")

        TRAJ_Hist <- aggregate(RF_DATA$NRF_per_Flight, by=list(RF_DATA$FLIGHT_ID), FUN=mean);
      }
```

```
if (k==2){

  mainlabel=paste("Contrails per Flight",'_' ,s, '[nautical miles]', sep="")
  seclabel = ""
  xcellStart = 0
  xcelllimit = 150
  ycellStart = 0
  ycelllimit = 120

  labelx = "Nautical Miles"
  labely = "Flight Count"

  savefile = paste("Contrial_Flight_nm" ,'_' ,s, sep="")
  TRAJ_Hist <- aggregate(RF_DATA$NMC, by=list(RF_DATA$FLIGHT_ID), FUN=mean);
}

if (k==3){
  mainlabel=paste("Contrails Generated [km] - ", DN ," - FL:" , i, sep="")
  savefile = "Contrailskm"
  labelx = "Contrails [km]"
  labely = "Days per year"
  TRAJ_ISSR_DATA$Contrails_km <- as.numeric(TRAJ_ISSR_DATA$Contrails_km); #Contrail km
  TRAJ_Hist <- aggregate(TRAJ_ISSR_DATA$Contrails_km, by=list(TRAJ_ISSR_DATA$W_DATE), FUN=sum);
}

#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;



#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
```

```
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , "
mW/m2", sep="")
led_text7 = "
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)

#Generate Histogram by Counts of Days

Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(xcellStart,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
legend("topright",  legend =ledgD , cex=0.8 ,
    col=c("royalblue", "red","black", "black", "black", "black" ))



plot( Hist1, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
   # main = "ISSR Cell Count",
   #sub="(including FL200 to FL400)",
   main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),

   cex.sub =.8,
   xlab = labelx,
   ylab =labely)

abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
```

```
    abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
    legend("topright",  legend =ledgD , cex=0.8 ,
        col=c("royalblue", "red","black", "black", "black", "black" ))


    dev.print(pdf, paste('E:\\\...\\proposal\\Final_Results\\',savefile,".pdf", sep=""))



    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1


  }
}
  k = k+1

}# end K
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
```

```
    Hist_out_max,
    Hist_out_StdDev,
    Hist_out_var
    ,stringsAsFactors=FALSE);

   write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist2_",savefile,".csv", sep=""), row.names = TRUE)




 #------------
 #------------
 }

}



#RF_FB_3 Flight level change
{

 #-------------------------------------------------------
 #-------------------------------------------------------

 {


   RF_DATA1 <- read.csv("E:\\\...\\proposal\\Final_Results\\RF_FB_V4_values.csv", colClasses  = "character")
   RF_DATA<-RF_DATA1
   RF_DATA$CNT <- as.numeric(RF_DATA$CNT)
   RF_DATA$NRF <- as.numeric(RF_DATA$NRF)
   RF_DATA$NMC <- as.numeric(RF_DATA$NMC)
   RF_DATA$FB_TOTAL <- as.numeric(RF_DATA$FB_TOTAL)
```

```
RF_DATA$FB_C <- as.numeric(RF_DATA$FB_C)

RF_DATA$FB_D <- as.numeric(RF_DATA$FB_D)

RF_DATA$FB_CO <- as.numeric(RF_DATA$FB_CO)

RF_DATA$FB_RF <- as.numeric(RF_DATA$FB_RF)

RF_DATA$FL_CHG2 <- as.numeric(RF_DATA$FL_CHG2)

RF_DATA$Include <- as.numeric(RF_DATA$Include)

RF_DATA$Limits <- as.numeric(RF_DATA$Limits)

RF_DATA$NRF_Per_Flight <- as.numeric(RF_DATA$NRF_Per_Flight)

RF_DATA$NRF_per_Flight_per_year <- as.numeric(RF_DATA$NRF_Per_Flight)/276

RF_DATA$FB_TOTAL_Per_Flight <- as.numeric(RF_DATA$FB_TOTAL_Per_Flight)



RF_DATA1<-RF_DATA




# FL_DATA$NRF <- as.numeric(FL_DATA$NRF)

#FL_DATA1$DATE1 <- substring(FL_DATA1$DATE_TIME,1,10)

#FL_DATA$W_Date <- as.Date(FL_DATA$DATE1,"%Y-%m-%d")

#  FL_DATA$CELLS <- as.numeric(FL_DATA$CELLS,"%d-%b-%y")

# FL_DATA$NM <- as.numeric(FL_DATA$NM,"%d-%b-%y")


#write.csv(CONTRAIL_DATA, file = paste("E:\\\...\\P2_IISE_Stats\\FL_DATA.csv", sep=""), row.names = TRUE)

#-----------------------------------


k =1

j = 1


for (k in (1:2)){

  for (s in c('All', 'Winter','Summer','Autumn','Spring')){


    RF_DATA<-RF_DATA1

   # RF_DATA =subset(RF_DATA, Include==1);
```

```
# RF_DATA =subset(RF_DATA, Limits==1);
 RF_DATA =subset(RF_DATA, RF_SOURCE=='FL_CHG');


 if (nrow(RF_DATA)>0){

   if (k==1){
    # RF_DATA =subset(RF_DATA, SEASON==s);
     mainlabel=paste("Flight Contrails per ",'_' ,s, '[nautical miles]', sep="")
     seclabel = ""
     xcellStart = -80
     xcelllimit = 80
     ycellStart = 0
     ycelllimit = 10000

     labelx = "Nautical Miles"
     labely = "Flight Count"

     savefile = paste("Contrial_Flight_nm" ,'_' ,s, sep="")
     TRAJ_Hist <- aggregate(RF_DATA$CNT, by=list(RF_DATA$FL_CHG2), FUN=sum);

      p<-ggplot(data=TRAJ_Hist, aes(x=Group.1, y=x)) +
       geom_bar(stat="identity", color="black", fill="white")+
       #scale_fill_manual(values=c(-80,-60,-40,-20,0,20,40,60,80))+
       theme_minimal()


     plot( p, col=rgb(.2,0.6,1,1/2), xlim=c(xcellStart,xcelllimit), ylim = c(ycellStart,ycelllimit), #.2,0.6,1
           # main = "ISSR Cell Count",
           #sub="(including FL200 to FL400)",
           main = paste(mainlabel, "\n" , seclabel,led_text7 ,sep=""),
           cex.sub =.8,
           xlab = labelx,
```

```
       ylab =labely)

  abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
  abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
  legend("topright",  legend =ledgD , cex=0.8 ,
        col=c("royalblue", "red","black", "black", "black", "black" ))

  dev.print(pdf, paste('E:\\\...\\proposal\\Final_Results\\',savefile,"Bar.pdf", sep=""))

}

if (k==2){
  RF_DATA =subset(RF_DATA, SEASON==s);

  mainlabel=paste("Induced Radiative Forcing",'_' ,s, sep="")
  if (SEASON=="All"){ mainlabel=paste("Net Radiative Forcing by Flight",sep="")}

  seclabel = ""
  xcellStart = -80
  xcelllimit = 80
  ycellStart = 0
  ycelllimit = 0.0015

  labelx = "Net Radiative Forcing"
  labely = "Change in Flight Level [X100 ft]"
  savefile =  paste("NRF_FlightLevel" ,'_' ,s, sep="")

  TRAJ_Hist <- aggregate(RF_DATA$NRF_Per_Flight, by=list(RF_DATA$FL_CHG2), FUN=mean);
}
```

```
if (k==3){
  if (c =='All'){

    mainlabel=paste("Net Radiative Forcing by Flight",'_' ,s, sep="")
    seclabel = ""
    xcellStart = 0
    xcelllimit = 0.005
    ycellStart = 0
    ycelllimit = 140

    labelx = "Net Radiative Forcing"
    labely = "Flight Change"
    savefile =  paste("NRF_Flight" ,'_' ,s, sep="")

    TRAJ_Hist <- aggregate(RF_DATA$NRF_Per_Flight, by=list(RF_DATA$FL_CHG2), FUN=mean);
  }
}

if (k==4){
  if (c =='All'){
  mainlabel=paste("Net Radiative Forcing by Flight",'_' ,s, sep="")
  seclabel = ""
  xcellStart = 0
  xcelllimit = 0.005
  ycellStart = 0
  ycelllimit = 140

  labelx = "Net Radiative Forcing"
  labely = "Flight Count"
  savefile =  paste("NRF_Flight" ,'_' ,s, sep="")

  TRAJ_Hist <- aggregate(RF_DATA$NRF_Per_Flight, by=list(RF_DATA$FL_CHG2), FUN=mean);
  }
```

```
}
#TRAJ_Hist$x <- as.numeric(TRAJ_Hist$x) ;

TRAJ_Hist <- aggregate(RF_DATA$CNT, by=list(RF_DATA$FL_CHG2), FUN=sum);

#-----------
#Build Ledgend
led_text1=paste("Mean:", format(mean(TRAJ_Hist$x, na.rm=TRUE), digits =2,scientific = FALSE) , sep="")
led_text2=paste("Median:", format(median(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text3=paste("Min:", format(min(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text4=paste("Max:", format(max(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text5=paste("StdDev:", format(sd(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text6=paste("Var:", format(var(TRAJ_Hist$x, na.rm=TRUE), digits=2,scientific = FALSE) , sep="")
led_text7=paste("Total RF:", format(sum(TRAJ_Hist$x, na.rm=TRUE),big.mark =",", digits=0,scientific = FALSE) , "
mW/m2", sep="")
led_text7 = ''
ledgD = c(led_text3,led_text2,led_text1,led_text4,led_text5)


#Generate Histogram by Counts of Days
Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3")


Hist1 <-hist(TRAJ_Hist$x , col = "aquamarine3", breaks=80,
        main=mainlabel, las=0, xlab = labelx, ylab = labely,
        xlim = c(xcellStart,xcelllimit),
        cex.lab = 1.3)

# H2 <-plot(Hist1)
grid(nx=NA,ny=NULL,lty=1,lwd=1,col="gray")
abline(v = mean(TRAJ_Hist$x, na.rm=TRUE ),col = "royalblue",lwd = 2)
abline(v = median(TRAJ_Hist$x, na.rm=TRUE), col = "red", lwd = 2)
```

```
    legend("topright",  legend =ledgD , cex=0.8 ,
         col=c("royalblue", "red","black", "black", "black", "black" ))



    dev.print(pdf, paste('E:\\\...\\proposal\\Final_Results\\',savefile,".pdf", sep=""))



    Hist_out_SourceHist[j] =mainlabel;
    Hist_out_Day_Night[j] =DN;
    Hist_out_FL[j] = i;
    Hist_out_mean[j] = mean(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_median[j] = median(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_min[j] = min(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_max[j] =max(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_StdDev[j] =sd(TRAJ_Hist$x, na.rm=TRUE);
    Hist_out_var[j] = var(TRAJ_Hist$x, na.rm=TRUE);
    j=j+1


   }
  }
  k = k+1

}# end K
Hist_Data_Out <- data.frame (
  Hist_out_SourceHist,
  Hist_out_Day_Night,
  Hist_out_FL,
  Hist_out_mean,
  Hist_out_median,
  Hist_out_min,
```

```
    Hist_out_max,

    Hist_out_StdDev,

    Hist_out_var

    ,stringsAsFactors=FALSE);


  write.csv(Hist_Data_Out, file = paste("E:\\\...\\P2_IISE_Stats\\Hist2_",savefile,".csv", sep=""), row.names = TRUE)


  #------------

  #------------

 }


}
```

DATA PREPARATION

```
--Prepare data
create or replace
PROCEDURE set_traffic_pos2_id
(traff_id in number)
is
  Begin
  update traffic2
        set pos2_id =
             (SELECT l.POS2_ID
             FROM TRAFFIC2 t  join
              LAT_LON2 l
             on
             (   t.cur_lat/60 >= l.lat_from and t.cur_lat/60 < l.lat_to
             and t.cur_lon/60 <= l.lon_from and t.cur_lon/60 > l.lon_to
             )
         where traffic_id = traff_id and l.pos2_id is null );
         commit;
END set_traffic_pos2_id;
```

CONTRAIL TO TRAJECTORY INTERSECTION

```
update Contrail_Work
set T_K= Temp
;

/*
  #RFLW = [OLR –kT(T_K-T0) {1-exp[-dt FLW(reff)t]} ELW(tc)
  #       Fact1          Fact2                Fact3
  Fact1 = OLR -( Kt*(T_K-T0) )
  */
update Contrail_Work
set LW_FACT1 = OLR -( Kt*(T_K-T0) );

 /* #FLW(reff) = 1-exp(-dlr reff)
  #reff = ¾ V/A [mm]
```

```
  if (diam == 0) {
    reff=0
  } else {
    V =4/3*pi*(diam/2)^3
    A =pi*(diam/2)^2
    reff=3/4*V/A}
*/
update Contrail_Work
set
VOL =power(4/3*PI*(DIAMETER/2),3),
AREA =power(PI*(DIAMETER/2),2)
;

update Contrail_Work
set reff=3/4*VOL/AREA;

update Contrail_Work
set  flw_reff =  1-exp(-dlr*reff);

update Contrail_Work
set  LW_FACT2 =  1-exp(-dt*flw_reff*tao);


update Contrail_Work
set  LW_FACT3 = exp(-dlc*tao2)  ;

update Contrail_Work
set  RFLW = 0;

update Contrail_Work
set  RFLW = LW_FACT1*LW_FACT2*LW_FACT3
where LW_FACT1*LW_FACT2*LW_FACT3 > 0;


--------------------LW RF CALC   END--------------

--------------------SW RF CALC--------------

update Contrail_Work
set  N_DAY = to_char(date_time,'DDD') ;

update Contrail_Work
set --decl = 23.44*cos((360/365) *(N_Day+10)); old
      decl = 23.44*sin(((360/365) *(284 + N_Day))*PI/180);

/*
Zenith
cos¿H=coslcoshcosd+sinlsind

l: north latitudes are positive, south latitudes are negative
d: the declination is positive when the sun's rays are north of the equator, and negative when the sun's rays are south of the equator.
h: the hour angle is negative before solar noon and positive after solar noon
f: the sun's azimuth angle is negative east of south and positive west of south
*/


update Contrail_Work
set Cos_Theta = cos(Cur_LAT*PI/180)*cos(decl*PI/180)*cos(Curr_HourAngle*PI/180)+sin(Cur_LAT*PI/180)*sin(decl*PI/180)
where
"HOUR" between 6 and 18 ;

update Contrail_Work
set Cos_Theta = 0
where
"HOUR" < 6 ;
```

```
update Contrail_Work
set Cos_Theta = 0
where
"HOUR" > 18;

select distinct "HOUR", Cos_Theta from  Contrail_Work  where "HOUR"  not between 6 and 18 ;
select Cur_LAT, cos(Cur_LAT*PI/180),
cos(Cur_LAT*PI/180)*cos(decl*PI/180)*cos(Curr_HourAngle*PI/180)+sin(Cur_LAT*PI/180)*sin(decl*PI/180) from
            Contrail_Work;


update Contrail_Work
set mu =
 case when Cos_Theta < 0 then 0
 else Cos_Theta
 end

update Contrail_Work
set SDR = S0 * mu;

update Contrail_Work
set SW_FACT1 = -SDR*power((tA- Aeff),2);

/*#Fact2
  #ac(m,t,reff) = Rc(teff)[Cm + AmR'C(t')Fm(m)]
  #RC(teff) = 1-exp(-G teff )
  #tc,eff = tc / m
  #Fsw(reff)=1-Fr[1-exp(-dsr*reff)]*/

--(8)
update Contrail_Work
set   FSW_REFF=1-Fr*(1-exp(-dsr*reff));

----------------------------------NEGATIVE mu, decl
--(7)

update Contrail_Work
set    TAO_PRIME = tao*FSW_REFF;

update Contrail_Work
set   teff =
case when mu <> 0 then TAO_PRIME/mu
else TAO_PRIME
end
;

--update Contrail_Work  set   teff = tao*FSW_REFF;

-------------------------------------
--(9)
update Contrail_Work
set    RCTEFF = 1 - exp(-GA * teff);
--  #R'C(terr) =exp(-g terr )


update Contrail_Work
set   RcTerr = exp(-ga2*TAO_PRIME);
-------------------------------------------

--------------------------------------------
--(11)
 -- #Fm(m) = [(1-m)Bm / (1/2)Bm ] -1
 update Contrail_Work
set  FM_M =((power((1-mu),Bm))/power((1/2),Bm))-1;

--  #ESW(m,tc) = exp (dSCtC – d'SC tc,eff)
```

```
 update Contrail_Work
 set   SW_FACT2 = RcTeff*(Cm+Am*RcTerr*FM_M) ;
  ---------------------------------------------
--(11)
--  #Fact3
update Contrail_Work
set tceff =
       case when mu <> 0 then tao2/mu
       else tao2
end;

 update Contrail_Work
set   SW_FACT3  = exp(dSC * tao - dsc1 * tceff );

----------------------
  --RFSW = Fact1*Fact2*Fact3

 update Contrail_Work
set RFSW =   SW_FACT1*SW_FACT2*SW_FACT3;

  --RF = RFLW + RFSW

update Contrail_Work
set RF = RFLW +RFSW ;
```

CALCULATE RF PROCEDURES

```
create or replace
PROCEDURE RF_LOOP AS

BEGIN

for x in (select distinct FID from Contrail_Work2)
  loop

      CALCULATE_RF(x.FID);

  end loop;
END RF_LOOP;

create or replace
PROCEDURE Calculate_RF(FIDX in varchar2)
is
  Begin

     UPDATE CONTRAIL_WORK2  Y
     set T_K= Temp
     where y.fid = FIDX;

     update Contrail_Work2 y
     set LW_FACT1 = OLR -( Kt*(T_K-T0) )
     where y.fid = FIDX;

     update Contrail_Work2 y
     set
     VOL =4/3*PI*power((DIAMETER/2),3),
     AREA =PI*power((DIAMETER/2),2)
     where y.fid = FIDX;

     update Contrail_Work2 Y
     set reff=3/4*VOL/AREA    where y.fid = FIDX;

     update Contrail_Work2 y
     set  flw_reff =  1-exp(-dlr*reff) where y.fid = FIDX;
```

```
update Contrail_Work2 y
set  LW_FACT2 =  1-exp(-dt*flw_reff*tao) where y.fid = FIDX;

update Contrail_Work2 y
set  LW_FACT3 = exp(-dlc*tao2)  where y.fid = FIDX;

update Contrail_Work2 y
set  RFLW = 0 where y.fid = FIDX;

update Contrail_Work2 y
set  RFLW = LW_FACT1*LW_FACT2*LW_FACT3
where LW_FACT1*LW_FACT2*LW_FACT3 > 0 and y.fid = FIDX;

update Contrail_Work2 y
set  N_DAY = to_char(date_time,'DDD') where y.fid = FIDX;

update Contrail_Work2  y
set decl = 23.44*sin(((360/365) *(284 + N_Day))*PI/180) where y.fid = FIDX;

update Contrail_Work2  y
set Cos_Theta = cos(Cur_LAT*PI/180)*cos(decl*PI/180)*cos(Curr_HourAngle*PI/180)+sin(Cur_LAT*PI/180)*sin(decl*PI/180)
where
"HOUR" between 6 and 18 and y.fid = FIDX;

update Contrail_Work2  y
set Cos_Theta = 0
where
"HOUR" < 6 and y.fid = FIDX;

update Contrail_Work2  y
set Cos_Theta = 0
where
"HOUR" > 18 and y.fid = FIDX;

update Contrail_Work2 y
set mu =
 case when Cos_Theta < 0 then 0
 else Cos_Theta
 end
 where y.fid = FIDX;

update Contrail_Work2  y
set SDR = S0 * mu where y.fid = FIDX;
update Contrail_Work2  y
set SW_FACT1 = -SDR*power((tA- Aeff),2) where y.fid = FIDX;

--(8)-----------------------------------------------
update Contrail_Work2  y
set   FSW_REFF=1-Fr*(1-exp(-dsr*reff)) where y.fid = FIDX;
-----------------------------------NEGATIVE mu, decl
--(7)

update Contrail_Work2  y
set    TAO_PRIME = tao*FSW_REFF where y.fid = FIDX;

update Contrail_Work2  y
set   teff =
case when mu <> 0 then TAO_PRIME/mu
else TAO_PRIME
end
 where y.fid = FIDX;

-------------------------------------
--(9)
update Contrail_Work2  y
```

```
    set    RCTEFF = 1 - exp(-GA * teff) where y.fid = FIDX;
    --  #R'C(terr) =exp(-g terr )


    update Contrail_Work2  y
    set   RcTerr = exp(-ga2*TAO_PRIME) where y.fid = FIDX;
    -------------------------------------------



    ---------------------------------------------
    --(11)
     -- #Fm(m) = [(1-m)Bm / (1/2)Bm ] -1
     update Contrail_Work2  y
    set  FM_M =((power((1-mu),Bm))/power((1/2),Bm))-1 where y.fid = FIDX;

     update Contrail_Work2  y
     set   SW_FACT2 = RcTeff*(Cm+Am*RcTerr*FM_M) where y.fid = FIDX;
     --(11)
    --  #Fact3
    update Contrail_Work2  y
    set tceff =
          case when mu <> 0 then tao2/mu
          else tao2
    end
    where y.fid = FIDX;

     update Contrail_Work2  y
    set   SW_FACT3  = exp(dSC * tao - dsc1 * tceff ) where y.fid = FIDX;

    update Contrail_Work2  y
    set RFSW =   SW_FACT1*SW_FACT2*SW_FACT3 where y.fid = FIDX;

    update Contrail_Work2  y
    set RF = RFLW +RFSW
    where y.fid = FIDX;

    update Contrail_Work2  y
    set
    RFSW_W =  RFSW * 13000 * Contrail_Width,
    RFLW_W = RFLW * 13000  * Contrail_Width
    where y.fid = FIDX;

    update Contrail_Work2  y
    set RF_W = RFLW_W +RFSW_W
    where y.fid = FIDX;

commit;
END Calculate_RF;
```

CREATE ALTERNATIVE FLIGHT LEVELS

```
drop table Flight_Intersec2  ;
create table Flight_Intersec2 as  --_415
select distinct
t.fid,
t.POS_TIME,
t.POSTIME,
t.POS2_ID,
to_char(t.pos_time,'HH24') Hr_Trajectory,
t.cur_lat, t.cur_lon,
l.nx,
l.ny,
fl.flight_level,
fl.z_val Z_VAL,
```

```
fs.DISTANCE,
fs.dept_arpt,
fs.arr_arpt,
fl_mode
FROM  TRAFFIC_ADB_V t,
FLIGHT_STATS FS,
final_sample f,
LAT_LON2 l,
FLIGHT_LEVEL_REF2 FL
where-- fs.fid = '10530084UAL385' and
FS.FID = T.FID
AND FS.FID = F.FID
AND F.L_800 = 1
--and fs.include2 = 1
AND T.POS2_ID =L.POS2_ID
and  t.z_val=fl.z_val
--and t.POS2_ID is not null
;

--SELECT COUNT(DISTINCT (FID)) FROM FLIGHT_INTERSEC2;

--SELECT z_val, COUNT (*) FROM WEATHER_INTER_MASTER GROUP BY z_val ORDER BY 1
 --------------------------------------
 --------------------------------------

drop table weather_Inter_Master;

--Find Weather for all altitudes and times
create table weather_Inter_Master as
select *
from
WEATHER2 w
where LAT ||'-' || LON in (select distinct NX ||'-' || NY from Flight_Intersec2);

drop table weather_Inter_Master2;
create table weather_Inter_Master2 as
select w.*, fl.flight_level from weather_Inter_Master w
join FLIGHT_LEVEL_REF2 FL
on fl.z_val = w.z_val;

drop table Contrail_20;
drop table Contrail_21;
drop table Contrail_22;
drop table Contrail_23;
drop table Contrail_24;
 --------------------------------------
 --------------------------------------

create table Contrail_20 as
select  'Flight'  Contrail_Type, 0 Traj_Delay, t.*,
        to_char(t.pos_time,'DD-MON-YYYY') Tr_Date,  to_char(t.pos_time,'HH24:Mi:SS') HHMISS,-- t.postime,
         -- fl.Z_VAL Z_VAL1  , fl.KFT     , fl.FLIGHT_LEVEL  , fl.MIN_PRESS        , fl.MAX_PRESS       ,
         w.ID          , w.LON    , w.LAT    , w.Z_VAL                W_Z_VAL, w.TEMP , w.SPEC_HUM        , w.U_VAR
          , w.V_VAR               , w.PRESS, w.DATE_TIME       , w.HOUR , w.flight_level wflight_level
       from
        Flight_Intersec2 t,
        weather_Inter_Master2 w,
         LAT_LON2 l
        where
        t.pos2_ID =l.pos2_id
         and l.nx= w.lat
         AND L.NY = W.LON
         and  w.flight_level between (fl_mode  - 80) and (fl_mode  + 80)
         AND TO_CHAR(T.POS_TIME,'HH24')  = W.HOUR
         -- and t.FID = '10541794AAL1857' and  w.flight_level  > 330
        order by fid, t.postime;
```

```
create table Contrail_21 as
select  'Persistence1' Contrail_Type, 1 Traj_Delay, t.*,
             to_char(t.pos_time,'DD-MON-YYYY') Tr_Date,  to_char(t.pos_time,'HH24:Mi:SS') HHMISS,-- t.postime,
              -- fl.Z_VAL Z_VAL1  , fl.KFT     , fl.FLIGHT_LEVEL  , fl.MIN_PRESS        , fl.MAX_PRESS       ,
              w.ID          , w.LON    , w.LAT    , w.Z_VAL               W_Z_VAL, w.TEMP , w.SPEC_HUM        , w.U_VAR
               , w.V_VAR               , w.PRESS, w.DATE_TIME       , w.HOUR  , w.flight_level wflight_level
            from
             Flight_Intersec2 t,
             weather_Inter_Master2 w,
              LAT_LON2 l
             where
             (FID || date_time|| t.Pos2_ID || w.flight_level ) in (select distinct FID || date_time || Pos2_ID || wflight_level from
           Contrail_20)
              and t.pos2_ID =l.pos2_id
              and l.nx= w.lat
              and l.ny = w.lon
              and  w.flight_level between (fl_mode  - 80) and (fl_mode  + 80)
              and to_char(t.pos_time,'HH24')+ 1  = w.hour
              order by fid, t.postime;

 create table Contrail_22 as
select  'Persistence1' Contrail_Type, 2 Traj_Delay, t.*,
             to_char(t.pos_time,'DD-MON-YYYY') Tr_Date,  to_char(t.pos_time,'HH24:Mi:SS') HHMISS,-- t.postime,
              -- fl.Z_VAL Z_VAL1  , fl.KFT     , fl.FLIGHT_LEVEL  , fl.MIN_PRESS        , fl.MAX_PRESS       ,
              w.ID          , w.LON    , w.LAT    , w.Z_VAL               W_Z_VAL, w.TEMP , w.SPEC_HUM        , w.U_VAR
               , w.V_VAR               , w.PRESS, w.DATE_TIME       , w.HOUR  , w.flight_level wflight_level
            from
             Flight_Intersec2 t,
             weather_Inter_Master2 w,
              LAT_LON2 l
             where
            (FID || date_time|| t.Pos2_ID || w.flight_level ) in (select distinct FID || date_time || Pos2_ID || wflight_level  from
           Contrail_21)
              and t.pos2_ID =l.pos2_id
              and l.nx= w.lat
              AND L.NY = W.LON
              and  w.flight_level between (fl_mode  - 80) and (fl_mode  + 80)
              and to_char(t.pos_time,'HH24') + 2 = w.hour
              order by fid, t.postime;


create table Contrail_23 as
select  'Persistence1' Contrail_Type, 3 Traj_Delay,t.*,
             to_char(t.pos_time,'DD-MON-YYYY') Tr_Date,  to_char(t.pos_time,'HH24:Mi:SS') HHMISS,-- t.postime,
              -- fl.Z_VAL Z_VAL1  , fl.KFT     , fl.FLIGHT_LEVEL  , fl.MIN_PRESS        , fl.MAX_PRESS       ,
              w.ID          , w.LON    , w.LAT    , w.Z_VAL               W_Z_VAL, w.TEMP , w.SPEC_HUM        , w.U_VAR
               , w.V_VAR               , w.PRESS, w.DATE_TIME       , w.HOUR  , w.flight_level wflight_level
            from
             Flight_Intersec2 t,
             weather_Inter_Master2 w,
              LAT_LON2 l
             where
              (FID || date_time|| t.Pos2_ID || w.flight_level ) in (select distinct FID || date_time || Pos2_ID || wflight_level  from
           Contrail_22)
              and t.pos2_ID =l.pos2_id
              and l.nx= w.lat
              and l.ny = w.lon
              and  w.flight_level between (fl_mode  - 80) and (fl_mode  + 80)
              and to_char(t.pos_time,'HH24') + 3 = w.hour
              order by fid, t.postime;


 create table Contrail_24 as
select  'Persistence1' Contrail_Type, 4 Traj_Delay, t.*,
             to_char(t.pos_time,'DD-MON-YYYY') Tr_Date,  to_char(t.pos_time,'HH24:Mi:SS') HHMISS,-- t.postime,
```

```
    -- fl.Z_VAL Z_VAL1  , fl.KFT     , fl.FLIGHT_LEVEL  , fl.MIN_PRESS        , fl.MAX_PRESS       ,
    w.ID          , w.LON    , w.LAT    , w.Z_VAL               W_Z_VAL, w.TEMP , w.SPEC_HUM        , w.U_VAR
     , w.V_VAR               , w.PRESS, w.DATE_TIME       , w.HOUR  , w.flight_level wflight_level
   from
    Flight_Intersec2 t,
    weather_Inter_Master2 w,
     LAT_LON2 l
    where
    (FID || date_time|| t.Pos2_ID || w.flight_level ) in (select distinct FID || date_time || Pos2_ID || wflight_level  from
  Contrail_23)
     and t.pos2_ID =l.pos2_id
     and l.nx= w.lat
     and l.ny = w.lon
      and  w.flight_level between (fl_mode  - 80) and (fl_mode  + 80)
     and to_char(t.pos_time,'HH24') + 4 = w.hour
     ORDER BY FID, T.POSTIME;

--SELECT COUNT(DISTINCT (FID)) FROM CONTRAIL_DATA2;

drop table Contrail_Data2;
 create table Contrail_Data2 as
 select * from Contrail_20
 union
 select * from Contrail_21
 union
 select * from Contrail_22
 union
 select * from Contrail_23
 union
 select * from Contrail_24;

 --------------------------------------
 --------------------------------------

select distinct lon, lat from Contrail_Data2 order by 1;
select distinct * from Contrail_20;
select * from Contrail_Work2
/*
drop table Contrail_0;
drop table Contrail_21;
drop table Contrail_22;
drop table Contrail_23;
drop table Contrail_24;
drop view Contrail_Data;*/


drop table Contrail_Work2;
create table Contrail_Work2 as
SELECT * FROM  CONTRAIL_DATA2
where fid in (select distinct fid from final_sample where L_800 = 1);


SELECT DISTINCT  FID FROM  CONTRAIL_DATA2
WHERE FID IN (SELECT FID FROM FINAL_SAMPLE WHERE L_1000 = 1);

truncate table  Contrail_Work2

insert into  Contrail_Work (CONTRAIL_TYPE,          TRAJ_DELAY,         TRAFFIC_ID,          DEPT_ARPT,
             ARR_ARPT,            FID,         CUR_LAT,             CUR_LON,              LAT_C,    LON_C,    ALT_100,
             GROUNDSPEED,   TR_DATE,             HHMISS,  POSTIME,              NY,          NX,          LAT_FROM,
             LON_FROM,           POS2_ID,  LAT_TO,  LON_TO,  ID,            LON,         LAT,         Z_VAL,    TEMP,
             SPEC_HUM,           U_VAR,    V_VAR,     PRESS,     DATE_TIME,            HOUR)
select * from  Contrail_Data;


SELECT  'Flight'  Contrail_Type, 0 Traj_Delay, t.*,
```

```
  TO_CHAR(T.POS_TIME,'DD-MON-YYYY') TR_DATE,  TO_CHAR(T.POS_TIME,'HH24:Mi:SS') HHMISS,
T.POSTIME,
  -- FL.Z_VAL Z_VAL1               , FL.KFT   , FL.FLIGHT_LEVEL               , FL.MIN_PRESS      , FL.MAX_PRESS
  ,
  W.ID         , W.LON   , W.LAT   , W.Z_VAL              W_Z_VAL, W.TEMP, W.SPEC_HUM        , W.U_VAR
   , W.V_VAR              , W.PRESS              , W.DATE_TIME      , W.HOUR , W.FLIGHT_LEVEL WFLIGHT_LEVEL
--  t.fl_mode, w.flight_level wflight_level
 FROM
  FLIGHT_INTERSEC2_415 T,
  weather_Inter_Master2 w, --Contrail_Data2 w, --
   LAT_LON2 l
  where
  t.pos2_ID =l.pos2_id
   AND L.NX= W.LAT
   and l.ny = w.lon
   --and  w.flight_level between fl_mode  - 80 and fl_mode  + 80
   AND TO_CHAR(T.POS_TIME,'HH24')  = W.HOUR
  order by t.fl_mode, to_char(t.pos_time,'DD-MON-YYYY') --fid, t.postime;
```

CREATE RF TREATMENTS

```
drop table Contrail_Work2;
CREATE TABLE CONTRAIL_WORK2 AS
SELECT * FROM  CONTRAIL_DATA2;
CREATE INDEX CONTRAIL_WORK2_INDEX1 ON CONTRAIL_WORK2 (FID, POSTIME);

Alter table Contrail_Work2

add OLR number(30,10)
add Kt number(30,10)
add T_K number(30,10)
add T0 number(30,10)
add dt number(30,10)
add FLW_reff number(30,10)
add dlr number(30,10)
add reff number(30,10)
add Vol number(30,10)
add Area number(30,10)
add tao number(30,10)
add tao2 number(30,10)
add ELW_tc number(30,10)
add dlc number(30,10)
add tc number(30,10)
add Diam number(30,10)
add LW_Fact1 number(30,10)
add LW_Fact2 number(30,10)
add LW_Fact3 number(30,10)
add RcTerr number(30,10)
add  N_Day number(30,10)
add decl number(30,10)
add mu number(30,10)
add Curr_HourAngle number(30,10)
add Cos_Theta number(30,10)
add SDR number(30,10)
add cos_q_S0 number(30,10)
add S0 number(30,10)
add tA number(30,10)
add Aeff number(30,10)
add Rcteff number(30,10)
add GA number(30,10)
add teff number(30,10)
add tao_prime number(30,10)
add FSW_reff number(30,10)
```

```
add Fr number(30,10)
add dsr number(30,10)
add Cm number(30,10)
add Am number(30,10)
add RprimeC_tao_prime number(30,10)
add ga2 number(30,10)
add Fm_m number(30,10)
add Bm number(30,10)
add dSC number(30,10)
add dSC1 number(30,10)
add tceff number(30,10)

add SW_Fact1 number(30,10)
add SW_Fact2 number(30,10)
add SW_Fact3 number(30,10)

add RF number(30,10)
add RFLW number(30,10)
add RFSW number(30,10)
add RF_W number(30,10)
add RFLW_W number(30,10)
add RFSW_W number(30,10)
add DIAMETER number(30,10)
add Contrail_Width number(30,10);


update Contrail_Work2
set
---------- LW Parameters
OLR = 275,
Kt=     1.935,
T0 =152,
dt=0.941,
dlr     =0.211,
dlc     =0.16,
-----------------------

---------- SW Parameters

tc =    0,
tA=     0.879,
Aeff=   0.36, --earth Albeado,
S0 = 1361, --cte of sun ,
GA=     0.242,
Fr=     0.512,
dsr=    0.15,
Cm=     0.709,
Am=     0.361,
ga2=    0.323,
Bm=     1.676, --0.709
dSC=    0.157,
dsc1=   0.23
;


update Contrail_Work2 set curr_hourangle = -90 where hour = 0 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 1 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 2 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 3 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 4 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 5 ;
update Contrail_Work2 set curr_hourangle = -90 where hour = 6 ;
update Contrail_Work2 set curr_hourangle = -75 where hour = 7 ;
update Contrail_Work2 set curr_hourangle = -60 where hour = 8 ;
update Contrail_Work2 set curr_hourangle = -45 where hour = 9 ;
update Contrail_Work2 set curr_hourangle = -30 where hour = 10 ;
```

```
update Contrail_Work2 set curr_hourangle = -15 where hour = 11 ;
update Contrail_Work2 set curr_hourangle = 0 where hour = 12 ;
update Contrail_Work2 set curr_hourangle = 15 where hour = 13 ;
update Contrail_Work2 set curr_hourangle = 30 where hour = 14 ;
update Contrail_Work2 set curr_hourangle = 45 where hour = 15 ;
update Contrail_Work2 set curr_hourangle = 60 where hour = 16 ;
update Contrail_Work2 set curr_hourangle = 75 where hour = 17 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 18 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 19 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 20 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 21 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 22 ;
update Contrail_Work2 set curr_hourangle = 90 where hour = 23 ;
commit;


drop table CONTRAIL_RF_21;
drop table CONTRAIL_RF_22;
drop table CONTRAIL_RF_23;
-----------------------------------
-----------------------------------

--SET Crystal Diameter , optical depth and width
update Contrail_Work2  Set DIAMETER = 10 , tao = 0.4, tao2 = 0 , Contrail_Width = 500 where TRAJ_DELAY = 0; -- Scenario 1
update Contrail_Work2  Set DIAMETER = 20 , tao = 0.2, tao2 = 0 , Contrail_Width = 1000 where TRAJ_DELAY = 1; -- Scenario 1
update Contrail_Work2  Set DIAMETER = 25 , tao = 0.08, tao2 = 0 , Contrail_Width = 2000 where TRAJ_DELAY = 2; -- Scenario 1
update Contrail_Work2  Set DIAMETER = 25 , tao = 0.02, tao2 = 0 , Contrail_Width = 3000 where TRAJ_DELAY = 3; -- Scenario 1
update Contrail_Work2  Set DIAMETER = 25 , tao = 0.01, tao2 = 0 , Contrail_Width = 4000 where TRAJ_DELAY = 4; -- Scenario 1

execute RF_LOOP;


create table CONTRAIL_RF_21 as
select * from
Contrail_Work2;

update Contrail_Work2  Set DIAMETER = 10 , tao = 0.5, tao2 = 0 , Contrail_Width = 500 where TRAJ_DELAY = 0; -- Senario 2
update Contrail_Work2  Set DIAMETER = 20 , tao = 0.4, tao2 = 0 , Contrail_Width = 1000 where TRAJ_DELAY = 1; -- Senario 2
update Contrail_Work2  Set DIAMETER = 30 , tao = 0.1, tao2 = 0 , Contrail_Width = 2000 where TRAJ_DELAY = 2; -- Senario 2
update Contrail_Work2  Set DIAMETER = 30 , tao = 0.08, tao2 = 0 , Contrail_Width = 3000 where TRAJ_DELAY = 3; -- Senario 2
update Contrail_Work2  Set DIAMETER = 30 , tao = 0.05, tao2 = 0 , Contrail_Width = 4000 where TRAJ_DELAY = 4; -- Senario 2

execute RF_LOOP;

create table CONTRAIL_RF_22 as
select * from
Contrail_Work2;

update Contrail_Work2  Set DIAMETER = 8 , tao = 0.4, tao2 = 0 , Contrail_Width = 1000 where TRAJ_DELAY = 0; -- Senario 3
update Contrail_Work2  Set DIAMETER = 15 , tao = 0.2, tao2 = 0 , Contrail_Width = 4000 where TRAJ_DELAY = 1; -- Senario 3
update Contrail_Work2  Set DIAMETER = 20 , tao = 0.08, tao2 = 0 , Contrail_Width = 8000 where TRAJ_DELAY = 2; -- Senario 3
update Contrail_Work2  Set DIAMETER = 20 , tao = 0.02, tao2 = 0 , Contrail_Width = 16000 where TRAJ_DELAY = 3; -- Senario 3
update Contrail_Work2  Set DIAMETER = 20 , tao = 0.01, tao2 = 0 , Contrail_Width = 32000 where TRAJ_DELAY = 4; -- Senario 3


create table CONTRAIL_RF_23 as
select * from
Contrail_Work2;
```

DATA EXTRACT FOR CONTRAIL MAP

```
Select * from
(
```

```
SELECT FID -- count(distinct(FID)),
FROM FINAL_SAMPLE
where L_800 = 1
group by
FID) a
join

(
select
(ADB_ID ||  CALL_ ) FID3,
round(cur_lat*2,-1)/2  Lat,
round(Cur_lon*2, -1)/2 lon1,
360+round(Cur_lon*2, -1)/2  lon
from
TRAFFIC_ADB_V
group by
(ADB_ID ||  CALL_ ),
round(cur_lat*2,-1)/2 ,
round(Cur_lon*2, -1)/2,
360+round(Cur_lon*2, -1)/2
) FL2
on a.FID = FL2.FID3
join
(
select
(ADB_ID ||  CALL_ ) FID3,
STATS_MODE(round(ALTITUDE,-1)) mode_FL
from
TRAFFIC_ADB_V
GROUP BY
(ADB_ID ||  CALL_ )) FL3
ON A.FID = FL3.FID3
;
```

REFERENCES

Abramson C. F. Almofeez, J. Carroll and C. Margopoulos , "Design of a decision support system to reduce net radiative forcing via optimal contrail generation,"2017 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, 2017, pp. 330-335.

Appleman, H. The formation of exhaust condensation trails by jet aircraft, B. Am. Meteor. Soc., 34, 14–20 (1953)

Banavar S, Hok K , Integration of Linear Dynamic Emission and Climate Models with Air Traffic Simulations, NASA Ames Research Center 2012

Banavar S, Hok Ng, Energy efficient contrail mitigation strategies for reducing the environmental impact of aviation , Tenth USA/Europe Air Traffic Management Research and Development Seminar, 2013

Brasseur, G. P. et al. Impact of Aviation on Climate FAA's Aviation Climate Change Research Initiative (ACCRI) Phase II. Bull. Am. Meteorol. Soc. 97, 561–583 (2016).

Bureau of Transportation Statistics, Web 2018

Burkhardt U, B Kärcher, Global radiative forcing from contrail cirrus, Nature Climate Change1,54–58 (2011) doi:10.1038/nclimate1068 Nature Climate Change 2011

Campbell S, Neogi N, Bragg M, An Optimal Strategy for Persistent Contrail Avoidance University of Illinois Urbana-Champaign, Urbana, IL, 61801, 2008

Chen N, Sridhar B., Evaluation of Contrail Reduction Strategies Based on Aircraft Flight Distances, NASA Ames Research Center, 2012

Chen N, Sridhar, B. S, Tradeoff between Contrail Reduction and Emissions under Future US Air Traffic Scenarios, NASA Ames Research Center, 2012 doi: 10.1109/SIEDS.2017.7937741

European Organization for the safety of air navigation. User manual for the base of aircraft data (BADA) 2004. Rev 3.7

Fichter, C., S. Marquart, R. Sausen, and D. S. Lee (2005), The impact of cruise altitude on contrails and related radiative forcing, Meteorol. Z., 14, 563–572.

Forster, P., Ramaswamy, V., Artaxo, P., Berntsen, T., Betts, R., Fahey, D.W., Haywood, J., Lean, J., Lowe, D.C., Myhre, Dowling, David R.; Radke, Lawrence F. (September 1990). "A Summary of the Physical Properties of Cirrus Clouds". Journal of Applied Meteorology. 29 (9): 970. Bibcode:1990JApMe..29..970D. doi:10.1175/1520-0450(1990)029<0970:ASOTPP>2.0.CO;2.

Fourth Assessment Report of the Intergovernmental Panel on Climate Change. Cambridge University Press, Cambridge.

Freudenthaler, V., Homburg, F. & Jäger, H. Annales Geophysicae (1994) 12: 956. doi:10.1007/s00585-994-0956-9

Gao H, Hansman J, Aircraft cruise phase altitude optimization considering contrail avoidance. Massachusetts Institute of Technology, 2013

Gierens K, et al, A Review of Various Strategies for Contrail Avoidance, Institute of Atmospheric Physics, DLR Oberpfaffenhofen, Germany, The Open Atmospheric Science Journal, 2008, 2, 1-7

Grewe, Volker. (2014) Aircraft routing with minimal climate impact: the REACT4C climate cost function modelling approach (V1.0), Geosci. Model Dev., Oberpfaffenhofen, Germany.

Hansen J, et al. Efficacy of climate forcings, J Geophys Res, 110 D18104, doi: 10.1029/2005JD005776

Haywood, J. M., et al., 2009: A case study of the radiative forcing of persistent contrails evolving into contrail-induced cirrus. J. Geophys. Res., 114, D24201.

ICAO, Continuing Traffic Growth and Record Airline Profits Highlight 2015 Air Transport Results, http://www.icao.int/Newsroom/Pages/Continuing-Traffic-Growth-and-Record-Airline-Profits-Highlight-2015-Air-Transport-Results.aspx, 2017. Web 2018

Jay Madigan, NASA Official: Dr. Lin Chambers, 07/2/2013, Contrail Science. Retrieved, http://science-edu.larc.nasa.gov/contrail-edu/science.php

Jensen E, Environmental conditions required for contrail formation and persistence. Journal Of Geophysical Research, Vol. 103, No. D4, Pages 3929-3936, February 27, 1998

Kaiser Michael, (2012), Tradeoff between optimum altitude and contrail layer to ensure maximum ecological en-route performance using the Enhanced Trajectory Prediction Model (ETPM), Department of Air Transport Technology and Logistics. Technische Universität Dresden, Germany

Kärcher B, U Burkhardt, S. Unterstrasser, and P. Minnis. Factors controlling contrail cirrus optical depth, Atmospheric Chemistry and Physics. 9, 6229–6254, 2009

Leahy, John. Mapping Demand 2016-2035 Global Market Forecast, http://www.airbus.com/company/market/global-market-forecast-2016-2035. 2017

Lee, D, Transport impacts on atmosphere and climate: Aviation. 2010 doi.org/10.1016/j.atmosenv.2009.06.005

Liou K.N., Ou S.C., Koenig G. (1990) An Investigation on the Climatic Effect of Contrail Cirrus. In: Schumann U. (eds) Air Traffic and the Environment — Background, Tendencies and Potential Global Atmospheric Effects. Lecture Notes in Engineering, vol 60. Springer, Berlin, Heidelberg

Lnewman, "Notes on Water Vapor Capacity of Air", Web 2016

Mannstein H, et al, A note on how to avoid contrail cirrus Transportation Research Part D: Transport and Environment. Volume 10, Issue 5, September 2005

Marsh Andrew J. Solar Radiation, Natural Frequency, ISSN 1833-7570. May 2017, http://naturalfrequency.com/wiki/solar-radiation

Masiol, Mauro, Roy M. Harrison, Aircraft engine exhaust emissions and other airport-related contributions to ambient air pollution: A review Division of Environmental Health and Risk Management, School of Geography, Earth and Environmental Sciences, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom

Meerk Otter R, Schumann U, Doelling, D.R. et al. HAL - Radiative forcing by contrails. Annales Geophysicae, European Geosciences Union, 1999, 17 (8), pp.1080-1094. <hal-00316645>

Myhre, G., D et. al , 2013: IPCC AR5 Anthropogenic and Natural Radiative Forcing. In: Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, pp. 659–740, doi:10.1017/ CBO9781107415324.018.

Naiman, A. D., Lele, S. K. & Jacobson, M. Z. Large eddy simulations of contrail development: sensitivity to initial and ambient conditions over first twenty minutes. J. Geophys. Res.116, D21208(2011).

NASA Earth Observations is part of the EOS Project Science Office located at NASA Goddard Space Flight Center . https://neo.sci.gsfc.nasa.gov/analysis/index.php

National Aeronautics and Space Administration, Science Mission Directorate. (2010). Wave Behaviors. Retrieved [insert date - e.g. August 10, 2016], from NASA Science website: http://science.nasa.gov/ems/03_behaviors

Nganga G, J., Prinn, R., Raga, G., Schulz, M., Van Dorland, R., 2007a. Changes in atmospheric constituents and in radiative forcing. In: Solomon, S., et al. (Eds.), Climate Change 2007: the Physical Science Basis. Contribution of Working Group I to the

Paoli, R. & Shariff, K. Contrail modeling and simulation. Annu. Rev. Fluid Mech. 48, 393–427 (2016).

Penner J, Lister D, et al, Aviation and the Global Atmosphere, Special Report of IPCC Working Groups I and III, Scientific Assessment Panel to the Montreal Protocol on Substances that Deplete the Ozone Layer, Published for the Intergovernmental Panel on Climate Change, 1999

Rosenow, Judith & Fricke, Hartmut & Luchkova, Tanja & Schultz, Michael. (2018). Minimizing contrail formation by rerouting around dynamic ice-supersaturated regions. Aeronautics and Aerospace Open Access Journal. 2. 10.15406/aaoaj.2018.02.00039.

Royal Geographical Society, Increasing Cloud Reflectivity Archived April 2, 2015, at the Wayback Machine., 2010

Sassen Kenneth. Contrail-Cirrus and Their Potential for Regional Climate Change, Department of Meteorology, University of Utah, Salt Lake City, Utah 1997

Schröder F, Kärcher B, On The Transition Of Contrails Into Cirrus Clouds, Journal Of The Atmospheric Science, 1999

Schumann U et al . Potential to reduce the climate impact of aviation by flight level changes. 3rd AIAA Atmospheric Space Environments Conference 27 - 30 June 2011, Honolulu, Hawaii

Schumann U. Formation, Properties and climatic effects of contrails. Institut für Physik der Atmosphäre,Deutsches Zentrum für Luft- und Raumfahrt (DLR),Oberpfaffenhofen,82230 Wessling,Germany. ulrich.schumann@dlr.de (2005)

Schumann U., B; K. Graf, H. Mannstein. A Parametric Radiative Forcing Model for Contrail Cirrus. Institute for Atmospheric Physics - Ulrich Schumann, Germany . 2012

Schumann, U. On conditions for contrail formation from aircraft exhausts. Meteorologische Zeitschrift. Z. 5, 4–23 (1996).

Schumman U. (2000) Experimental Test of the Inuence of Propulsion Efciency on Contrail Formation DLR, German Aerospace Center, 82230 W essling, Germany

Soler M, Zou B, Hansen M, Flight trajectory design in the presence of contrails: Application of a multiphase mixed-integer optimal control approach. Transportation Research Part C: Emerging Technologies doi.org/10.1016/j.trc.2014.08.009 2014

Solomon, S., D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M. Tignor and H.L. Miller (eds.)IPCC, 2007: Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change . Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 996 pp.

Sridhar B, Chen N, Energy Efficient Contrail Mitigation Strategies for Reducing the Environmental Impact of Aviation , Tenth USA/Europe Air Traffic Management Research and Development Seminar ATM2013

Strauss, B., Meerkoetter, R., Wissinger, B. et al. On the regional climatic impact of contrails: microphysical and radiative properties of contrails and natural cirrus clouds. Annales Geophysicae (1997) 15: 1457. https://doi.org/10.1007/s00585-997-1457-4

"Table Specific humidity of a kilogram of air" Web 2016

Tanveer H, Gauntlett D , Diaz J, Yeh P, "Design of a flight planning system to reduce persistent contrail formation to reduce greenhouse effects," 2014 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, 2014, pp. 84-89. doi: 10.1109/SIEDS.2014.6829890

Wilcox L, et al, Radiative forcing due to aviation water vapor emissions, Atmospheric Environment 63:1–13 · December 2012

Williams V, Nolan R., Variability of contrail formation conditions and the implications for policies to reduce the climate impacts of aviation. https://doi.org/10.1016/j.trd.2005.04.003 2005

Worden, H., K. Bowman, S. Kulawik, and A. Aghedo, 2011: Sensitivity of outgoing longwave radiative flux to the global vertical distribution of ozone characterized by instantaneous radiative kernels from Aura-TES. J. Geophys. Res. Atmos., 116, D14115

Yin F, Grewe V, Fromming C, Yamashita H, Impact on flight trajectory characteristics when avoiding the formation of persistent contrails for transatlantic flights. Transportation Research Part D: Transport and Environment //doi.org/10.1016/j.trd.2018.09.017

Zhang Y, Mackeb A, AlbersaF, Effect of crystal size spectrum and crystal shape on stratiform cirrus radiative forcing.. Atmospheric Research. doi.org/10.1016/S0169-8095(99)00026-5

BIOGRAPHY

Denis Avila graduated with honors in Industrial Engineering from the National Autonomus Universityu in Mexico in 1998. He received his M.S. in Systems Engineering from George Mason University, Fairfax VA in 2010. Denis has fifteen years of experience developing functional architectures and implementing efficient data solutions for large-scale systems. His research intrests focus on air traffic and its relation to weather data.