Minimizing the Search Area for the Location of Missing Planes in the Ocean

Team # 42042

February 9, 2015

Abstract

In this paper, we present a collection of methods to expedite and simplify the search for a missing plane believed to have crashed into open waters, such as an ocean. Our first technique creates a preliminary search zone based on the time between expected pings from the airplane while in flight and the calculated velocity of the plane. We then modify the search area by creating a simplified probability distribution based on the analysis of data from similar plane crashes. Our final modification of the search area utilizes a 100-point rating and ranking system of our own design that determines how similar each plane crash is to the the scenario of the missing plane. We use this rating to re-vamp our probabilities in the prior search area and then calculate a weighted average to give us an area of highest probability within the search zone created by the rating. We then test our methodology using the data available from the Air France Flight 447. Our model was highly effective in determining an accurate search zone for the missing flight. The zone our model created contained the location of the wreckage of Air France flight 447. We also explore a technique using a projectile motion in the horizontal direction to try to calculate a buffer zone that gives the maximum possible distance the wreckage could be from its last known position.

Contents

1 Introduction

With the development of recent events such as the disappearance of Malaysian Flight MH370 and Asia Air Flight 8501, the importance of being able to locate where these planes and others have crashed into the ocean has come to the foreground of the public eye. The initial problem of finding a plane after it disappears from radar is difficult enough due to the depth and scope of the ocean where the plane could have crashed into. But when you factor in the obstacles created by the ocean itself, the process of searching for and locating the plane wreckage becomes time-consuming, expensive, and tiresome.

In the past, many have relied upon the principles of Bayesian Search Theory [12, 27, 28] to narrow down the search area for the plane based upon the probability the plane fell into the ocean in a particular sector and the probability of being able to find the plane in that sector with current technology. Even then, the process remains long and expensive. In the more recent years, searchers have greatly relied upon the information available about the missing plane based upon its last set of **pings**, a term referring to an electronic "handshake" the plane's computer make with a satellite while in flight that includes information about location, altitude, and status [23, 12]. By knowing where and at what time the pings were given we can use the information to form a model to create a search zone for the plane.

In our model, we have utilized both of these techniques to create a preliminary search radius centered at the location of the plane's last ping. We then develop a rating and ranking system of one hundred points that analyzes past plane crashes of a similar nature in comparison to the dynamics of the current crash under investigation. This system paired with Bayesian principles is used to modify the initial search radius by weighting each region with a specific probability, thereby generating "hot zones" within our previous buffer zone that have a higher probability of containing the missing plane.

1.1 Plan of Attack

Our model must accurately predict a region of the ocean in which to search for a missing plane believed to have crashed with the assumption that the black boxes are defective and do not emit a signal that can be used to locate the plane. In order to cope with this predicament, in this paper we will:

- Determine the greatest search radius in which the plane could have crashed,
- Narrow down our search zone based off the parameters of the plane at the Last Known Position (LKP), and
- Determine which zones in our search area have the highest probability of containing the downed plane based off the analysis of previous plane crashes.

1.2 Assumptions

The problem cannot be sufficiently modeled without making several assumptions. We assume the following:

– Before crashing, the plane was able to emit a set of pings that contain information about longitude, latitude, altitude, and time of the ping.

– The time between pings is constant and known.

– The reason the plane did not emit an expected ping is because it has either crashed or started to crash.

– The expected path of the plane from its place of departure and its destination is known and accessible.

– The plane was intact when it impacted the water (i.e. the plane did not explode or break apart while still in flight).

– We have access to all available technology, information, and equipment in order to find the missing plane.

1.3 Key Terms

– **LKP** - Last Known Position

– **Pings** - data sent from the plane while in flight that contain geographic information about the flight

– **Bayes Theorem** - a theorem for establishing a statistical relationship between current probability and prior probability

– **Buffer Zone** - this is an area that shows the maximum distance the wreckage could be expected from the LKP

– **Side-scan Sonar** - this is a system of sonar that is efficient for developing basic images of the sea floor

– **SAS** - Synthetic Aperture Sonar; this is a form of sonar in which sophisticated post-processing methods are use to develop high-resoltuion images of the scanned area

– **Metron Corporation** - the company known for developing the search methodologies for tracking down Air France Flight 447 as well as developing the algorithm for SAROPS

– **SAROPS** - Search and Rescue Optimal Planning System; developd by Metron Corporation and used by the Coast Guard to aid in search and rescue missions

– **Projectile Motion** - the study of how a thrown object or particle travels under the influence of gravity and other forces such as air resistance

– **Black Box** - often called the flight recorder; this is the plane's computer that stores all of the flight information such as position and engine status

2 Developing a Search Zone

We now begin an explanation of our proposed technique to reduce the time and effort required to find the missing plane.

2.1 Initial Search Zone

The first set of data is from the lost plane. Using its last set of pings, we determine two radii to create a "buffer" zone for our search area. By calculating the last known air speed, we can use the time between pings to create a maximum distance the plane could have flown before impacting the water. We are able to do this because we assume

A the reason we have not received another ping is because the plane has either crashed or started to crash and

B the plane is limited by the right triangle formed by the maximum horizontal distance the plane can fly based on its last pings and its last known altitude.

We denote these two radii as P_{n+1} and P_{n+2}, where P_n is the last known ping, P_{n+1} is the next expected ping, and P_{n+2} is the second expected ping that was not received. The length of P_{n+1} is the estimated horizontal distance, using the plane's set of last pings, the plane can travel in the time between pings. The length of P_{n+2} is twice the length of P_{n+1}.

The pings given off by the airplane during its flight are modeled by P_m where m is the number of the ping emitted while in flight, labeled $(1, 2, ..., n)$, and n is the last ping given off by the airplane before it no longer gives out a signal. Contained in each ping is the longitude, latitude, and altitude of the airplane and the time the ping was given off such that

$$P_n = < x_n, y_n, z_n, t_n >$$

and x_n is the longitude, y_n is the latitude, z_n is the altitude, and t_n is the time of the ping. To determine the radius of the search area after the plane has fallen, we use the data included in the last two pings received from the plane, P_n and P_{n-1}, to generate a circle around the location at time t_n. Based on our assumptions, the distance from P_{n-1} to P_n is equivalent to the distance from P_n to P_{n+1}, so either distance can be used to determine the radius of the search area. However, in instances that the distance is different between the two, we recommend using the distance between the last ping and the next expected ping to generate a search area. The **search area** then becomes a circle centered at (x_n, y_n) with a radius equal to the distance between (x_n, y_n) and (x_{n-1}, y_{n-1}). We come up with the initial search area in Figure 1 modeled by the equation

$$(x - x_n)^2 + (y - y_n)^2 = r^2$$

where

$$r = \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2} \ .$$

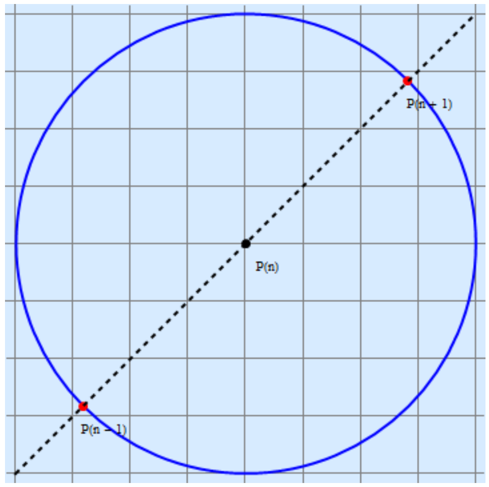


Figure 1: Initial search area based on data from last set of pings before assumed crash.

2.2 Modified Search Zone

While the first radius creates a decent field to look for the plane in, it does not consider the possibility that the plane malfunctioned right before the next expected ping and stayed in the air past the location and time where the next ping was scheduled to go off. To account for this, we modified our model to include the area after the next expected ping but before the second expected ping. This results in the model now having an increased radius of

$$(x - x_n)^2 + (y - y_n)^2 = 2d,$$

which is equal to the distance between P_{n-2} and P_n, based upon our assumption that the distance between P_{n-2} and P_{n-1} is the same as the distance between P_{n-1} and P_n, as shown in Figure 2.

But the search area, dependent upon the last calculated velocity of the airplane, is still too large for a feasible and efficient search.

2.3 Projectile Buffer Zone

To further narrow down our search area we began to examine the equation for the projectile motion of an object modeled by

$$s(t) = z_n - \frac{(v_t)^2}{g} ln(cosh(\frac{g}{v_t} t))$$

where $s(t)$ is the altitude for the plane at time t [29]. The altitude of the plane or last known altitude is given by the ping data and is represented by z_n. The velocity of the airplane, v_t can be approximated at time t_n

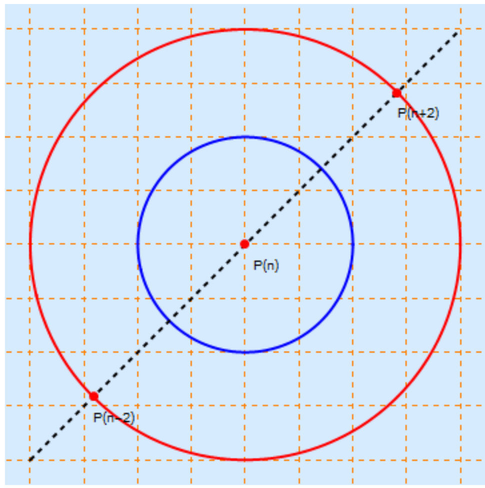


Figure 2: Projectile Buffer Zone as determined by the set of pings before the crash.

by using the terminal velocity of the plane which can be solved by the equation

$$v_t = \sqrt{\frac{gm}{c}}$$

where g is the gravity constant $32.2 feet\ sec^{-2}$, m is the mass of the plane, and c is the air drag coefficient of the plane. If all this data is known, then we can solve for t when $s(t) = 0$ to approximate how long it takes the airplane to crash into the ocean. Unfortunately, the air drag coefficient is unique to each plane model and is not readily available to find [33]. If we could solve for $s(t) = 0$ we would be able to use the time to narrow down our search area for the plane, but doing so is not possible at this time.

2.4 Rating and Ranking Search Zones

Our next step in modifying our search area was to take data from past plane crashes similar in nature to the missing plane currently being investigated and use it to create a simple probability map. In this probability map, the distance each plane was found from its start of emergency point (which we assume is equivalent to last known position) is used to create a radius around the LKP and give that particular radius a probability based upon how many planes were found that particular distance away. For the map in Figure 3, we used 9 plane crashes that were analyzed by the Metron Corporation while in search for Air France Flight 447 [27, 28] and a distance between pings (black dashed circles) of 10 NM (nautical miles).The other colored circles in Figure 3 are the various distances of the plane crashes from their start of emergency pings. The red circles represent areas where multiple crashes occurred within that zone, all other colors are single crashes in that zone and have distinguished colors purely for readability.

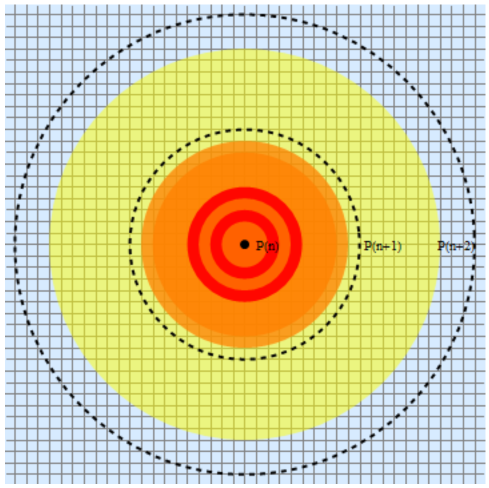


Figure 3: Modified Search Zone based on analysis of past crashes.

However the distances between the start of emergency ping and the plane wreckage still does not provide a feasible search zone. Therefore, to improve this method, we created a 100 point rating system to compare the past plane crashes with the missing plane currently under investigation. We then ranked the crashes according to the rating each crash received. This rating system took nine data points of comparison from the specifications of the airplane and its flight in question and then assigned a point value to each data point. We then compared the data points of previous crashes to the crash under investigation. The more similar the data points were to that of the flight being investigated the more points the crash zone of the previous accident received. The crash zone with the most points has the highest probability of containing the crash under investigation. To demonstrate how this works, we used the same nine plane crashes Metron Corporation used in their comparison to Air France Flight 447 [27, 28]. See Appendix A for all the data points that we used for our rating system.

Our rating system uses the following data points for comparison with respective point values to determine how similar each plane crash is to the one we are investigating.

Variable	Point Value
Weight	20
Wing Span	10
Height	10
Length	5
Region of Ocean	10
Weather Conditions	2.5
Passengers on Board	10
Last Known Altitude	20
Cruise Speed	12.5

In order to make this process more efficient and accurate, we created a computer program written in the C++ programming language that reads the values for each plane crash from a file and calculates their rating based on our guidelines and then ranks them based on each score.(See Appendix B and C for the source code and the executed program.) To determine the ratings, we calculated each particular point value and added them together for the total points earned. To find each particular point value, we used the following formula:

$$points\ earned = (point\ value)\frac{value_A - |value_A - value_B|}{value_A},$$

where $value_A$ refers to the variable value for the missing plane and $value_B$ refers to the respective variable value for the plane in comparison. For example, for the Aeroflot plane crash in 1994[32], the points earned for weight would be

$$(20)\frac{232000-|232000-212000|}{213000} = 18.4348.$$

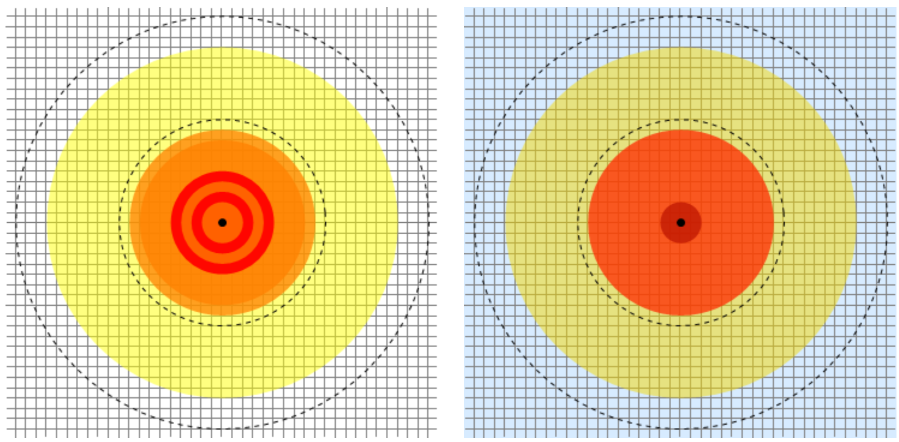
For the weather and ocean region categories, the following system was used to award points.

Region	Point Value
Same Ocean	10.0
Different Ocean	7.5
Lake	5.0
River	2.5
Land	0.0

Weather	Point Value
Same	2.5
Similar	1.25
Different	0.0

After determining these ratings, the computer program uses a sorting algorithm to rank them in descending order (greatest first and least last). We then use this ranking to revamp our probability map so that the more probable radii correlate with the plane crashes that were most similar to the crash in question. Figure 4 shows the map before and after our rating and ranking system is applied.

The darker section's outer radius is the most probable location while the orange area's outer radius is the second most probable location for the aircraft. Therefore, based upon this mapping, we would recommend



(a) Before rating and ranking. (b) After rating and ranking.

Figure 4: Distribution Maps

searching between the two radii for the wreckage and then move in towards the LKP.

Another way we can help narrow down the search efforts is to take the rankings from our new system and to find a weighted average as follows:

$$.60(top\ 3\ rankings) + .25(next\ 3\ rankings) + .15(last\ 3\ rankings).$$

The percentages can be changed based upon how many rankings there are. For example, in the case of 10 rankings, you can do a .50, .25, .15, .07, .03 weighted average using every two rankings. Doing so gives us an area of higher probability within our area of highest probability of which to search by, as shown in Figure 5.

3 Locating the Missing Aircraft

After determining the search area for the plane the next step is to begin searching for the plane in the ocean. There are several techniques for accomplishing this task. For our approach, we considered different forms of radar, most dominantly the Synthetic Aperture Sonar (SAS) [18].

3.1 Use of Radar

For a quick response and search after the disappearance of the plane, the best option is to send planes to fly over the preliminary search zone. The planes can use radar to try to locate the missing plane (if it is floating on the surface of the water) and/or debris from the wreck. If any debris is found, the United States Coast Guard's Search and Rescue Optimal Planning System (SAROPS) can be used to retrace the path of the debris based on the weather conditions and currents at the approximated time

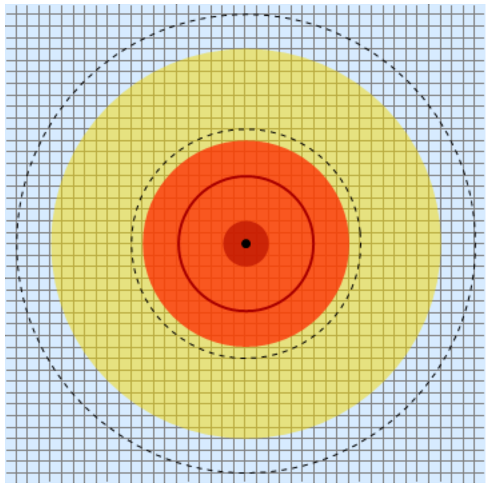


Figure 5: Weighted average radius on probability map.

of the wreck [19, 24]. Combined with Bayes Theorem, this can greatly improve the search area [27].

Bayes Theorem states

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

where $P(A|B)$ is the probability of the plane crashing in a certain sector given that the debris floated from that sector [21, 26, 15]. This is equal to the probability that the debris floated from some sector A given the plane crashed in sector A ($P(B|A)$) multiplied by the probability that the plane crashed in sector A ($P(A)$), all divided by the probability that the debris drifted from sector A ($P(B)$). SAROPS can be used to generate this probability with any debris found. In our methodology, we can utilize this technique in combination with our rating and ranking model to improve the probability of the sectors in the search zone and hone in on individual sectors with the highest probability of containing the crashed plan[19, 24]. But there is a **weakness** to using Bayes Theorem with the data. Some of the problems are as follows:

– The formula only creates a probability that the plane is in a particular sector, but there is still a chance the plane is not located in that sector with the highest probability.

– Erratic currents in the sea can inhibit tracking of the debris back to their point of origin.

In the case there is no debris found, Bayes Theorem can still be used, but not as accurately or efficiently. After performing an initial search of the hot zones in the search area, if the plane is not found Bayes Theorem can be run again, this time with $P(A)$ still being the probability of finding the plane in some sector A based on our hot zones but now $P(B)$ is the probability of the plane being located in some sector A having already searched the sector but not finding the plane. This method was used by the Metron in their search for the missing Air France Flight 447 [12].

3.2 Use of Synthetic Aperture Sonar (SAS)

If the plane has not been found after the initial sweep, the next step would involve doing a scan of the ocean floor using Synthetic Aperture Sonar (SAS). **SAS** is a type of sonar that uses a remote controlled transmitter that gives off consecutive pings along a straight track to map the ocean floor [16, 18]. SAS, while slow, is able to create a detailed view that provides a better picture than sidescan sonar.

It is worth noting however that since SAS is a slow process, we recommend using our Search Area Model and then applying Bayes Theorem before going to search the ocean floor. This way the area to cover can be reduced by starting with areas of high probability.

3.3 Use of Sidescan Sonar

Without restrictions on funds or time we recommend the use of SAS to locate the missing plane. However, SAS does not work in all topographical situations and not all real world scenarios have unlimited funds. Underwater topography does not always allow for a straight line which is required for use of the SAS system. Also, SAS is an expensive resource to use. In case the search team is limited by curved underwater surfaces or funds we recommend using sidescan sonar. Sidescan does not create as clear a picture, but it does not require travelling in a straight line and also is a cheaper alternative to SAS [18, 16].

4 Tests and Results

4.1 Air France Flight 447

To test our algorithm, we used data from Air France Flight 447. After completing some preliminary research, we were able to find the LKP for AF447 at 2.98deg N latitude and 30.59deg W longitude [27]. We were also able to find the time between pings to be 10 minutes [27]. Based on the analysis of Metron Corporation, we modified this time so that the radius of the P_{n+2} circle would be 20 NM [28], yielding the following search zone in Figure 6:

Then, using the same plane crashes Metron Corporation used [28], we re-defined our search area using the distance between the wreckage and the start of emergency location for each plane to give a rough probability of where AF447 could be. The technique we used to do so was to make a circle with radius equal to that distance for each plane crash centered at the LKP for AF447. Then, if more than one wreck was found at that radius, we gave it a higher probability than those with only one plane. The outlier was the 17 NM distance, which we gave a lower probability to counter that (Figure 7).

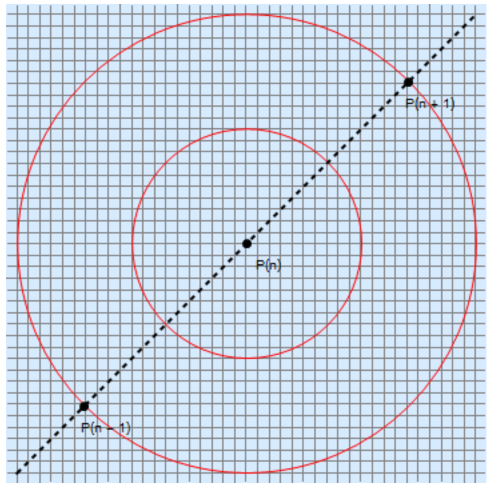


Figure 6: Preliminary Search Zone for AF447.

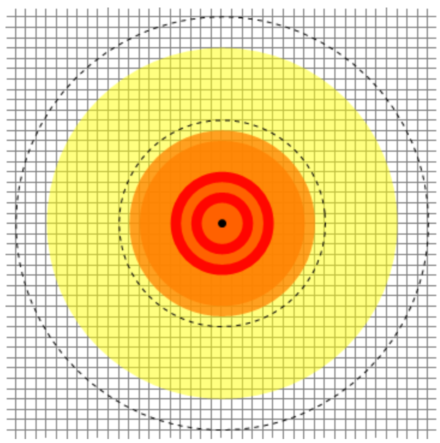


Figure 7: Search Zone with Rough Probabilities from Initial Analysis

Next, we incorporated our proposed rating and ranking technique to modify the search zone once more. As mentioned in previous sections, we analyzed each plane crash based on nine components (weight, wingspan, height, length, region, weather, passengers, altitude, and cruise speed), each with a respective point value. The ratings are based upon how well each plane crash compares to AF447. The data for AF447 is as follows [20, 9]:

Variable	Value
Weight	230000
Wingspan	60.30
Height	17.39
Length	58.82
Region	Atlantic
Weather	Fair
Passengers	228
Altitude	38000
Cruise Speed	880

We then ran the data in Appendix A through our computer program (Appendix B) and created the following rankings:

Ranking	Plane Crash	Rating	Distance*
1	Aeroflot 1994	72.9445	3
2	Pulkovo 2006	64.5623	3
3	Adam Air 2007	64.3475	9
4	Silk Air 1997	62.4505	5
5	Caspian Airlines 2009	60.1324	5
6	West Caribbean 2005	58.7991	17
7	IRS Aero 2001	50.1154	4
8	Aeroflot 1995	45.9966	8
9	Trans Asia	35.1410	2

*Distance is the nautical miles from start of emergency.

Using this ranking, we were able to once again re-define our search area. We define the zone between radius 3 and radius 9 to be the "hot zone" because it is within the donut type image made by the top three ratings in Figure 8. We believe there is a much higher probability of finding the plane in this zone due to our ranking system and therefore, by searching there first, we can save time and money.

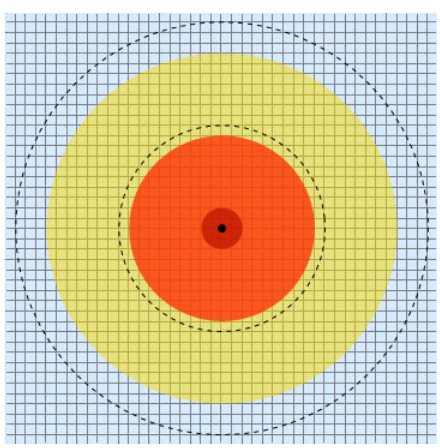


Figure 8: After rating and ranking.

Our final modification to the search area is a weighted average. By giving our top three ratings a weight of 60%, our next three a weight of 25%, and our bottom three a weight of 15%, we are able to find the following average:

$$.60(\frac{3+3+9}{3}) + .25(\frac{5+5+17}{3}) + .15(\frac{4+8+2}{3}) = 5.88.$$

We then modify our search area to include the weight average radius and define it as the region of the highest probability within our hot zone in Figure 9.

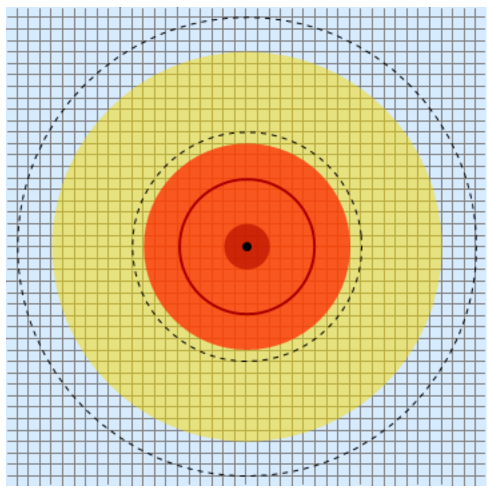


Figure 9: Weighted average radius on probability map.

4.1.1 Analysis of Results

Based upon our calculations, the wreckage of the Air France Flight 447 should have been found between three and nine nautical miles from the last known position, close to 5.88 nautical miles away. In reality, they found AF447 less than 6 miles from its LKP [11], which translates into 5.21 nautical miles. Therefore, our methodology worked well for this particular plane crash as it was within our search zone in Figure 10.

5 Improvements to the Model

5.1 Strengths and Weaknesses of Projectile Model

5.1.1 Strengths

The projectile model appealed most to our team because it offered an approach that would provide a gross over-estimation of where the plane would have come down, creating a solid buffer zone from where the last ping was sent. By using common variables, we wanted to figure out approximately how long this plane was in the air before it impacted the

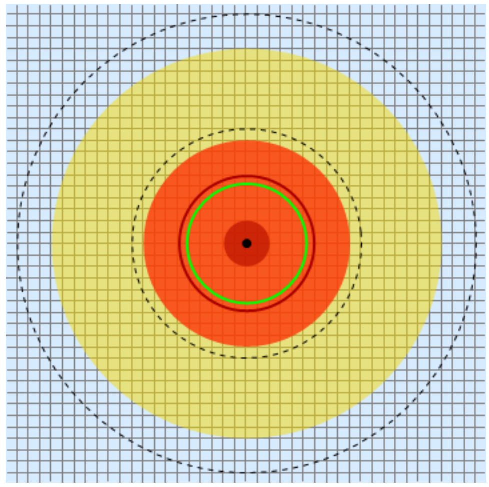


Figure 10: Final map of AF447 and our search area.

ocean. The strength of this approach was that we were able to take into account physical conditions and apply those to how the plane would have plummeted out of the sky. Since the plane is in flight just like a horizontal projectile with an initial velocity greater than zero, our approach was to break the plane up into three vector components to figure out an approximation of how long the plane would have remained airborne and then figure out a radius for a search zone from its LKP. The other benefit of using a projectile equation was that it would have been exceptionally easy to create and work with a computer visual or model. The other apparent strength we found in our model was that our buffer zone that the projectile model created was approximately twelve nautical miles tighter than the buffer zone of Metron which was 40 NM compared with out zone of 32 NM [27, 28]. Unfortunately, we were unable to address exactly why our zone was just over 75% of the size of Metron's zone. However, if we were able to access the necessary information, such as the correct drag coefficient and the cross-sectional area of the airplane, then we believe we would have had a much better clue as to why our data was different from theirs.

5.1.2 Weaknesses

One weakness of this model is the minimal amount of variables we take into consideration for discovering how long the plane remained airborne to calculate our radius. While this did provide an over-estimation for the general area the wreckage could be located in, we did not have access to enough data to narrow down the radius and shrink the search zone. Part of the reason this cannot be done is because the necessary data points for a more accurate analysis exist within the plane's black box which can only be found in the wreckage. But, with the right information, such as the air drag coefficient, we should be able to modify this approach in the future (See Future Work).

5.2 Strengths and Weaknesses of the Rating and Ranking System

5.2.1 Strengths

One of the strengths for this model was how accessible the data points were. This model works in a way that is very similar to that of a Rating Percentage Index (RPI) for sports teams. We used the collection of flights Metron [28] used in their analysis for tracking down Air France Flight 447 and then tracked down key data points including the take off weights of the aircraft and its dimensions. These data points were considered key because of the role they played in the projectile motion formula that strongly influences the time the aircraft will be airborne. By having easy access to each of these data points we were able to create a very basic rating system to each of the flights to establish a comparison those the crash locations of previous accidents to create a search zone for AF447. During this process we discovered another strength of the model, which is the ability to easily convert our ranking system into a computer program that calculates each rating and then ranks them accordingly. This method was simple and easy to work with, especially since the program is user friendly and the math is algorithmic in nature.

5.2.2 Weaknesses

The weakness of the Rating and Ranking System is how basic our estimation for the impact radius is. Even though we were able to use nine data points for evaluation, the estimations were still preliminary since essential data for this project would be located in the black box of the plane. Just like the projectile motion model, there are many data points we do not have access to that would allow for a more complex and elaborate comparison. The other weakness from this comes from how long it takes for SAS to cover the "hot zone" we have marked with a high probability. Therefore, to improve this method, it would be necessary to develop an optimized technique for searching zones marked high priority efficiently, effectively, and accurately.

6 Future Work

In the future, we would not only like to continue modifying and perfecting our current search method, but we would also like to continue development of the equation:

$$s(t) = z_n - \frac{(v_t)^2}{g} ln(cosh(\frac{g}{v_t}t))$$

where g is the gravity constant, v_t is the terminal velocity, and z_n is the last known altitude of the plane [29]. This would allow us to pinpoint the time it took for the plane to crash using only the information from the plane's last set of pings. The terminal velocity v_t, the variable that made it impossible to continue with this model, can be calculated using $\sqrt{\frac{gm}{c}}$ [29]. By finding the approximate time it took for the plane to start its descent and impact the water, we would be able to approximate a radius by finding the corresponding horizontal distance the plane would have covered. The only problem with this equation lies in the air drag coefficient c. This value can only be found if one of two things are known; the drag force upon the plane or the chord length of the airplane wing [33]. We currently are not able to find valid sources to supply either piece of data. Therefore, in the future, we would like to find this data, if possible, or create our own data using experimental results from wind tunnel tests and run experiments.

7 Conclusion

The problem of trying to pin-point the wreckage of a lost plane over open waters proved to be demanding of creative thought. Since the most necessary data to determine an approximate location with a high degree of accuracy is all contained on the flight's black box, which is inaccessible until the wreckage is located, we found ourselves with a need to answer a very complicated question with minimal amounts of crucial data. Our first approach used a relatively simple model of projectile motion with the influence of gravity and air resistance. While we had a high level of confidence in this approach, we lacked essential data to carry out the calculations and tests for this model and had to abandon it. This put us back at the drawing board for developing a new idea for creating a search radius and we chose to do so through a system of rating and ranking. We discovered that through our methodology of creating search areas by assigning ratings to other crashes throughout history according to the commonalities shared between them and the crash under investigation we were able to significantly narrow down the search area produced by Metron Corporation with a fairly high degree of accuracy. While Metron established a search radius of 40 NM [28], we were able to create a maximum probable radius of 17 NM with our most probable radii being 3 NM and 9 NM. We were able to maintain confidence in the accuracy of our second approach since the Air France Flight 447 was 5.21 NM from its LKP [11].

8 Press Release Report

Tracking down the location of a crashed plane in open waters proves to be very difficult. Most of the information needed to help track down the location of the wreckage is located on the plane's computer, known as the black box, which stores all the information about the flight while the plane is in the air. The importance of finding the black box is to help provide insight as to what might have caused the plane to crash in the first place.

Since the accuracy of a mathematical model is heavily dependent on the amount of data and what kind of data mathematicians have access to, creating a model to aid searchers in finding the wreckage calls for various, and often creative, approaches. The approach we have taken creates a comparison of multiple plane crashes with the missing plane currently being investigated. This comparison allows us to find multiple similarities between the lost plane and previous crashes to create a zone of high probability of where the wreckage will be. The calculations for this are fairly simple, but deciding how much certain commonalities matter in the model is the difficult part because they are so sensitive. Since characteristics such as size and weight greatly affect how an object falls from the sky, we decided to give these the highest weightings.

After determining which plane crashes possess the most commonalities with the plane we are currently investigating, a search will begin much like the searches conducted for the previous crashes. Since where the previous crashes were in relation to the last known location of the flight are known, we are able to assert, with some certainty, that the lost flight will likely be in a similar area.

Therefore, a flyover of the preliminary search zone will be issued in hopes to locate debris from the wreckage of the plane. While finding debris would be helpful, whether it is found or not we can utilize our data from previous crashes to reduce the size of the search zone. In doing so, we can increase the effectiveness of our attempts to find the wreckage using sonar and thus find the plane in a timely manner.

We hope that with our method for locating the missing plane we can help provide closure for those that have missing loved ones that were on the plane.

A Particulars for Comparison Planes

Plane Crash	Plane Model	Weight (kg)	Wingspan (m)	Height (m)	Length (m)
Aeroflot 1994	A310-300	212000	43.89	15.80	46.66
Aeroflot 1995	TU-154	82600	37.55	11.40	48.00
Silk Air 1997	B737	79010	34.30	12.50	39.50
IRS Aero 2001	IL87	64000	37.40	10.17	35.90
Trans Asia 2002	ATR72	21500	27.05	7.65	27.17
West Caribbean 2005	MD82	67810	32.87	9.02	45.06
Pulkovo 2006	TU154	100000	37.55	11.40	48.00
Adam Air 2007	B737	68040	34.30	12.50	39.50
Caspian Airlines 2009	TU154	100000	37.55	11.40	48.00

Plane Crash	Region	Weather	Passengers	Altitude (ft)	Cruise Speed (km hr^{-1})
Aeroflot 1994	Land	Fair	75	31000	850
Aeroflot 1995	Land	Winter Storm	98	10600	850
Silk Air 1997	River	Fair	104	35000	960
IRS Aero 2001	Land	Fair	100	26000	625
Trans Asia 2002	Pacific	Winter Storm	3	18000	526
West Caribbean 2005	Land	Fair	160	33000	813
Pulkovo 2006	Land	Thunderstorm	170	39000	850
Adam Air 2007	Pacific	Fair	112	35000	960
Caspian Airlines 2009	Land	Fair	168	24000	850

See Reference for the sources used to compile the crash data, [1, 2, 3, 4, 5, 6, 7, 9, 10, 13, 17, 22, 30, 31, 32, 34].

B C++ Program Code

```
//***************************************************************************
//
//     Lindsay Bradley (Justin Groves, Mark Hubbard)
    February 8th, 2015       COMAP MCM 2015 Problem B
//
//     This program assists in our Ranking and Rating method
    for finding a missing plane at sea.
//
//****************************************************************************

//header files
#include <iostream>
#include <iomanip>
#include <fstream>
#include <cmath>
#include <cstdlib>
#include <stdlib.h>
#include <ctime>
#include <string>
using namespace std;

#define SIZE 9//number of airplanes being analyzed

class PlaneCrash
{
private:
   double weight;//weight of plane in kilograms
   double wingSpan;//wingspan of plane in meters
   double height;//height of plane in meters
   double length;//length of plane in meters
   double oceanRegion;//see Region chart
   double weather;//see weather chart
   double passengers;//number of passengers on plane
   double altitude;//altitude of plane in feet
   double cruiseSpeed;//cruising speed of the plane in
       kilometers per hour
   double totalPoints;//rating of plane
   string planeName;//name of plane crash

public:
   PlaneCrash() {};//default constructor
   void setData(ifstream &inFile)//reads in plane
       information from file
   {
      inFile >> planeName;
```

```
        inFile >> weight >> wingSpan >> height >> length >>
            oceanRegion >> weather >> passengers >> altitude
            >> cruiseSpeed;
    }//end setData()

    void createRating(double arg_weight, double arg_wingSpan,
        double arg_height, double arg_length, double
        arg_oceanRegion, double arg_weather, double
        arg_passengers, double arg_altitude, double
        arg_cruiseSpeed)
    {//creates the rating for the plane
        weight = 20 * ((arg_weight - abs(weight -
            arg_weight))/arg_weight);
        wingSpan = 10 * ((arg_wingSpan - abs(wingSpan -
            arg_wingSpan)) / arg_wingSpan);
        height = 10 * ((arg_height - abs(height - arg_height))
            / arg_height);
        length = 5 * ((arg_length - abs(length - arg_length))
            / arg_length);
        oceanRegion = 10 * ((arg_oceanRegion - abs(oceanRegion
            - arg_oceanRegion)) / arg_oceanRegion);
        weather = 2.5 * ((arg_weather - abs(weather -
            arg_weather)) / arg_weather);
        passengers = 10 * ((arg_passengers - abs(passengers -
            arg_passengers)) / arg_passengers);
        altitude = 20 * ((arg_altitude - abs(altitude -
            arg_altitude)) / arg_altitude);
        cruiseSpeed = 12.5 * ((arg_cruiseSpeed -
            abs(cruiseSpeed - arg_cruiseSpeed)) /
            arg_cruiseSpeed);
        totalPoints = weight + wingSpan + height + length +
            oceanRegion + weather + passengers + altitude +
            cruiseSpeed;
    }//end createRating()

    double getTotalPoints()//returns the total points
    {
        return totalPoints;
    }//end getTotalPoints()

    string getPlaneName()//returns the name of the plane crash
    {
        return planeName;
    }//end getPlaneName()

};//end PlaneCrash class


int main()
{
```

```
PlaneCrash planes[SIZE];//array of plane crashes being
    analyzed

//public variables equivalent to private class variables
string c_planeName;
double c_weight;
double c_wingSpan;
double c_height;
double c_length;
double c_oceanRegion;
double c_weather;
double c_passengers;
double c_altitude;
double c_cruiseSpeed;
double c_totalPoints;

ifstream inFile;//input file
inFile.open("RatingAndRanking.dat");

ofstream outFile;//output file
outFile.open("RatingAndRanking.out");

//prompts user to enter data for missing plane
cout << "Please input the weight of the plane: ";
cin >> c_weight;
cout << endl << "Please input the wingspan of the plane:
    ";
cin >> c_wingSpan;
cout << endl << "Please input the height of the plane: ";
cin >> c_height;
cout << endl << "Please input the length of the plane: ";
cin >> c_length;
cout << endl << "Please input the ocean Region of the
    plane: ";
cin >> c_oceanRegion;
cout << endl << "Please input the weather variable of the
    plane: ";
cin >> c_weather;
cout << endl << "Please input the passengers in the
    plane: ";
cin >> c_passengers;
cout << endl << "Please input the last known altitude: ";
cin >> c_altitude;
cout << endl << "Please input the cruise speed: ";
cin >> c_cruiseSpeed;

for (int i = 0; i < SIZE; i++)//rates each plane crash in
    comparison to missing plane
{
   planes[i].setData(inFile);//reads data from file
```

```
        planes[i].createRating(c_weight, c_wingSpan, c_height,
            c_length, c_oceanRegion, c_weather, c_passengers,
            c_altitude, c_cruiseSpeed);//creates rating
    }//end for loop

    for (int j = 1; j <= SIZE; j++)//sorts ratings into
        descending order
    {
        for (int i = 0; i < SIZE - j; i++)
        {
            double rankingOne =
                planes[i].getTotalPoints();//temporary
                variables to call private total points for
                comparison
            double rankingTwo = planes[i + 1].getTotalPoints();
            if (rankingOne<rankingTwo)//if that plane's rating
                is lower than the next this will swap their
                positions
            {
                PlaneCrash p = planes[i];
                planes[i] = planes[i + 1];
                planes[i + 1] = p;
            }//end if statement
        }//end inner for loop
    }//end outer for loop


    outFile << setw(15) << "Ranking" << setw(25) << "Plane"
        << setw(25) << "Rating" << endl << endl;//headers for
        output file
    for (int i = 0; i < SIZE; i++)//outputs rankings of each
        plane
    {
        c_planeName = planes[i].getPlaneName();//gets the name
            of the plane
        c_totalPoints = planes[i].getTotalPoints();//gets the
            rating of the plane
        outFile << setw(15) << i + 1 << setw(25) <<
            c_planeName << setw(25) << c_totalPoints <<
            endl;//prints out results
    }

    //close files
    inFile.close();
    outFile.close();

    return 0;
}//end main()
```

C Executed Program

```
C:\Windows\system32\cmd.exe
Please input the weight of the plane: 230000
Please input the wingspan of the plane: 60.30
Please input the height of the plane: 17.39
Please input the length of the plane: 58.82
Please input the ocean Region of the plane: 1
Please input the weather variable of the plane: 1
Please input the passengers in the plane: 228
Please input the last known altitude: 38000
Please input the cruise speed: 880
Press any key to continue . . .
```

Figure 11: This is the window that prompts the user for input data about the downed flight

Ranking	Plane	Rating
1	Aeroflot-1994	72.9445
2	Pulkovo-2006	64.5623
3	AdamAir-2007	64.3475
4	SilkAir-1997	62.4505
5	Caspian-2009	60.1324
6	WestCar-2005	58.7991
7	IRSAero-2001	50.1154
8	Aeroflot-1995	45.9966
9	TransAsia-2002	35.141

Figure 12: This is the output file produced from the program. It is the complete list of ratings ordered in a descending ranking.

References

[1] 737 Family. (2015, January 1). Retrieved February 9, 2015, from http://www.boeing.com/boeing/commercial/737family/pf/pf_600tech.page?

[2] 747 Family. (2015, January 1). Retrieved February 8, 2015, from http://www.boeing.com/boeing/commercial/747family/747-8_fact_sheet.page?

[3] ASN Aircraft accident Boeing 737-4Q8 PK-KKW Pambauang. (2015, January 1). Retrieved February 9, 2015, from http://aviation-safety.net/database/record.php?id=20070101-0

[4] ASN Aircraft accident McDonnell Douglas MD-82 HK-4374X Machiques. (2015, January 1). Retrieved February 9, 2015, from http://aviation-safety.net/database/record.php?id=20050816-0

[5] ASN Aircraft accident Tupolev 154M RA-85185 Donetsk. (2015, January 1). Retrieved February 9, 2015, from http://aviation-safety.net/database/record.php?id=20060822-0

[6] ATR 72. (2012, January 1). Retrieved February 9, 2015, from http://www.airlines-inform.com/commercial-aircraft/ATR-72.html

[7] Accidents, incidents and the crash in the USSR and Russia. (2015, January 1). Retrieved February 9, 2015, from http://www.airdisaster.ru/reports.php?id=13

[8] AirSafe.com. (2015, January 1). Retrieved February 9, 2015, from http://www.airsafe.com/events/airlines/transasia.htm

[9] Airbus. A330-200 Prestige specifications. Retrieved February 8, 2015, from http://www.airbus.com/fileadmin/media_gallery/files/tech_data/jetFamily/media_object_file_A330_200_specifications.pdf

[10] AviationKnowledge. (2012, October 15). Retrieved February 9, 2015, from http://aviationknowledge.wikidot.com/asi:aeroflot-flight-593:child-in-the-cockpit

[11] Bodies from Air France Flight 447 are found. (2011, April 4). Retrieved February 9, 2015, from http://www.airfrance447.com/04/04/bodies-from-air-france-flight-447-are-found/

[12] Brumfiel, G. (2014, March 25). Can A 250-Year-Old Mathematical Theorem Find A Missing Plane? Retrieved February 9, 2015, from http://www.npr.org/blogs/thetwo-way/2014/03/25/294390476/can-a-250-year-old-mathematical-theorem-find-a-missing-plane

[13] Commercial Airplanes. (2015, January 1). Retrieved February 9, 2015, from http://www.boeing.com/boeing/commercial/md-80/product.page

[14] Crash of SilkAir Flight MI 185. (n.d.). Retrieved February 9, 2015, from http://eresources.nlb.gov.sg/infopedia/articles/SIP_1576_2009-09-30.html

[15] Gigerenzer, G. (2002). Calculated risks: How to know when numbers deceive you. New York: Simon & Schuster.

[16] Hansen, Roy Edgar. (2011). Introduction to Synthetic Aperture Sonar, Sonar Systems, Prof. Nikolai Kolev (Ed.), ISBN: 978-953-307-345-3, InTech, Available from: http://intechopen.com/books/sonar-systems/introduction-to-synthetic-aperture-sonar

[17] IL-18 Passenger Aircraft, Russia. (2015, January 1). Retrieved February 9, 2015, from http://www.aerospace-technology.com/projects/il18-aircraft/

[18] Kovacs, E. (2012, Fall). The Art of Shipwreck Hunting. Retrieved February 9, 2015, from http://www.alertdiver.com/Shipwreck-Hunting

[19] Kratzke, Thomas M., Lawrence D. Stone, and John R. Frost. Search and Rescue Optimal Planning System. Retrieved from Metron Scientific Solutions website: http://www.metsci.com/Portals/0/Search%20and%20Rescue%20Optimal %20Planning%20System.pdf

[20] Live Science Staff. (2014, May 2). Facts About Flight 370: Passengers, Crew & Aircraft. Retrieved February 8, 2015, from http://www.livescience.com/44248-facts-about-flight-370-passengers-crew-aircraft.html

[21] Lucertini, M., Gasca, A., and Nicolô, F. (Eds.). (2004). Technological concepts and mathematical models in the evolution of modern engineering systems: Controlling, managing, organizing. Basel: Birkhäuser Verlag.

[22] Map: Malaysia Airlines Flight 370. (2014, June 26). Retrieved February 9, 2015, from http://www.cnn.com/interactive/2014/03/world/malaysia-flight-map/

[23] Ostrower, J., and Pasztor, A. (2014). 'Partial Ping' Under Review for Clues About Flight 370. The Wall Street Journal. Retrieved February 9, 2015, from http://www.wsj.com/articles/SB20001424052702304679404579461900800102412

[24] Netsch, Robert. (2004, June) The USCG Search and Rescue Optimal Planning System (SAROPS) Via the Commercial/Joint Mapping Tool Kit (C/JMTK). Retrieved from: http://proceedings.esri.com/library/userconf/proc04/docs/pap1185.pdf

[25] Shaikh, T. (2011, July 29). Air France crash pilots lost vital speed data, say investigators. Retrieved February 8, 2015, from http://www.cnn.com/2011/WORLD/americas/05/27/air.france.447.crash/ index.html?hpt=T1

[26] Spiegel, M., Lipschutz, PhD, S., and Liu, PhD, J. (2013). Probability. In Mathematical Handbook of Formulas and Tables (4th ed.). New York: McGrawHill.

[27] Stone, L. (2015, January 1). In Search of Air France Flight 447. Retrieved February 9, 2015, from https://www.informs.org/ORMS-Today/Public-Articles/August-Volume-38-Number-4/In-Search-of-Air-France-Flight-447

[28] Stone, Lawrence D., Colleen Keller, Thomas L. Kratzke, and Johan Strumpfer. (2011, 20 January). Search Analysis for the Location of the AF447 Underwater Wreckage. Retrieved from the Bureau d'Enquêtes et d'Analyses website: http://www.bea.aero/fr/enquetes/vol.af.447/metron.search.analysis.pdf

[29] Tabor-Morris, A. (2014, May 6). Wolfram Demonstrations Project. Retrieved February 9, 2015, from http://demonstrations.wolfram.com/FallingBodyWithZeroLinearOrQuadraticAirResistance/

[30] TUPOLEV TU-154 RUSSIAN 3 ENGINE COMMERCIAL AIRLINER AIRCRAFT INFORMATION FACTS HISTORY AND PHOTOS. (2003, January 1). Retrieved February 9, 2015, from http://www.aviationexplorer.com/Tupolev_TU-154_Airliner_Facts_History_Photos.html

[31] The ATR ATR-72. (2015, January 1). Retrieved February 9, 2015, from http://www.airliners.net/aircraft-data/stats.main?id=42

[32] The Airbus A310. (2015, January 1). Retrieved February 9, 2015, from http://www.airliners.net/aircraft-data/stats.main?id=20

[33] The Drag Coefficient. (2014, July 16). Retrieved February 9, 2015, from http://www.grc.nasa.gov/WWW/k-12/airplane/dragco.html

[34] Tupolev Tu-154. (2012, January 1). Retrieved February 9, 2015, from http://www.airlines-inform.com/commercial-aircraft/Tu-154.html

[35] Earth. (2015, January 1). Retrieved February 9, 2015, from http://earth.nullschool.net/#current/ocean/surface/currents/orthographic=-245.10,3.37,303