The Iron Laws of Air Traffic Control

Kevin Arnett
Jonathan S. Gibbs
John J. Horton
U.S. Military Academy
West Point, NY

Advisor: David Bailey

Introduction

We focus our analysis on two key system design specifications:

- how the Air Traffic Control System accomplishes its requirements (the safe routing of aircraft through a sector of airspace), and
- what computational and time demands are generated by the traffic load.

We develop a solution that takes into account factors such as knowledge of proposed flight paths, orientation of aircraft, and acceptable probability of an accident. We develop two separate models to analyze situations where in-flight conflicts arise.

- The first examines the position of aircraft through three-dimensional normal probability distributions to develop the likelihood of a collision.
- The second uses vector calculus and dynamics to develop real-time data on the likely trajectory of an aircraft, making no assumptions that the aircraft are flying along a predetermined path.

Two other models use analogies to other fields to provide metrics of complexity of the workload of the air traffic controller (ATC), one focusing on the inherent complexity of an airspace (analogous to fluid flow) and the other on number of aircraft.

Of our four models, we are able to validate only two, the probability distribution model for likelihood of collision and the airspace complexity model. The implementation and validation are straightforward and we omit them due to time considerations.

The UMAP Journal 21 (3) (2000) 269–282.

Assumptions

- All aircraft flight paths are filed with the FAA prior to departure and are available to all ATCs.
- We treat an aircraft as a sphere with a set radius, whose orientation is insignificant.
- Two distinct types of errors cause flight path deviation:
  - systematic error (pilot error, emergencies, failed equipment, etc.) and
  - random error (differences in weather, GPS signals, aircraft characteristics).
- Relativistic effects are negligible.
- The curvature of the Earth need not be accounted for explicitly (the ease with which transformation matrices can be designed precludes this from being of great import).

Requirement A

All aircraft are required by law to file flight plans with the Federal Aviation Administration. ATCs generally assume that an aircraft will follow its filed flight plan but are attentive to deviations. There is a duality in the ATC job: on one hand, ATCs plan as if everything will perform in a predictable manner, while they simultaneously must be vigilant in case things do not. The two different roles of the ATC are reflected in our model. Based on how an ATC monitors a sector, we break our model into two separate submodels:

- The Random Effects Model predicts potential conflicts between two aircraft based on flight path data and characteristics of the aircraft, weather, etc.
- The Contingency Model addresses routine real-time monitoring of aircraft. It makes no assumptions about aircraft following a given flight path. The model serves as an alert system for an ATC, warning of conflicts caused by systemic errors as they arise.

Both models run on real-time data, but the Contingency Model has a higher priority in terms of computing resources. The Random Effects Model is re-evaluated as fluctuations in velocity affect the flight path. The Contingency Model is more relevant in situations where airplanes are not well spaced and where there is a high probability for unplanned deviations in flight path. In contrast, the Random Effects Model is more suited to the operational planning of flight paths and the monitoring of major air corridors.

While certainly different, these two models both answer the question of what constitutes "too close," by examining situations that make a collision likely and advising the ATC of the need for intervention.

It may seem that we should determine a safe separation between two planes, but this question is so related to orientation as to be meaningless. The better approach is to determine what flight paths and velocities will cause a collision.

Random Effects Model

We represent an expected path with a vector-valued function $r_p(t)$ such that

$$r_p(t) = x(t)\vec{i} + y(t)\vec{j} + z(t)\vec{k},$$

where t is time and x, y, z are functions that describe the aircraft's position. However, due to a variety of factors such as weather and instrumentation inaccuracies, the actual position is not fixed but rather is dependent on three random variables. We define this actual position vector as

$$r_p(t) = x'(t)\vec{i} + y'(t)\vec{j} + z'(t)\vec{k} = [x(t) + \epsilon_x]\vec{i} + [y(t) + \epsilon_y]\vec{j} + [z(t) + \epsilon_z]\vec{k},$$

where each of the three error terms has normal distribution centered at 0, $\epsilon_i \sim N(0, \sigma_i^2)$ for dimension i, where each dimension can have a different variance. We assume independence of the error terms. We identify values for the variances based on data from the FAA (see **Appendix A**). Essentially, this model describes a probability shell surrounding the aircraft (**Figure 1**).

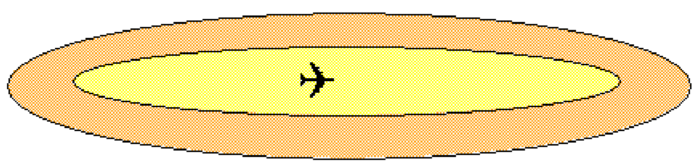

Figure 1. Probability shell for an aircraft in flight (not to scale).

The probability of being within Δ_i of a predicted location in dimension i is

$$P_i(\Delta_i) = \Phi(\Delta_i/\sigma_i) - \Phi(-\Delta_i/\sigma_i) = 2\Phi(\Delta_i/\sigma_i) - 1,$$

where Φ is the standard normal cumulative probability density function. Assuming independence of the error terms in the three directions, the probability that the plane will be within $(\Delta_x, \Delta_y, \Delta_z)$ of its projected position is

$$g(t, x', y', z') = P_x(\Delta_x)P_y(\Delta_y)P_z(\Delta_z).$$

Consider two aircraft with probability functions g and h, assumed independent. The probability that there will be a collision at a given point is the probability that both planes occupy that point at the same time, namely,

$$c(t, x', y', z') = g(t, x', y', z') \cdot h(t, x', y', z').$$

Thus, we determine the viability of two flight paths by examining the probabilities associated with each point at every instant. This is a very computationally intensive prospect, but there are methods that provide solutions in reasonable amounts of time.

Implementation and Validation

We wrote a program in Visual Basic for Applications on top of a Microsoft Excel spreadsheet. Flight data are placed in the worksheet, which the program uses as input. We used the program to validate the model for two scenarios:

- The two planes collide.
- The two planes cross paths but at different times.

The results of these simulations are presented and discussed in **Appendix B**.

Contingency Aircraft Tracking System

For aircraft deviating grossly from their flight plans, the Random Effects Model is not useful. The Contingency Aircraft Tracking System is designed to alert the ATC to any aircraft that could be on a collision course.

Using data collected on the positions of the aircraft, by either GPS or some other monitoring system, the system uses the path that the aircraft has been on to predict where it will be in the future. Those future positions are used to designate a sector of air space as off limits to other aircraft. When two or more aircraft are predicted to pass through the same sector, the ATC is alerted. This tracking system is based on several assumptions:

- Aircraft do not accelerate in the direction of travel while in the airspace.
- Aircraft turn at a constant normal acceleration or move in a straight path.
- The position locating systems are accurate and provide continuous updating.

Most major airports have the tracking ability described in the third assumption; continuous updating compensates for the other assumptions of zero tangential acceleration and constant normal acceleration.

To predict possible future positions of the aircraft, we use three past positions. We calculate the vectors from the first to the second and from the second to the third and determine the angle between them. If the angle is below a certain tolerance, we approximate the path of the aircraft by a straight line; otherwise, we approximate it by the arc of the circle defined by the three points. In the latter case, since the aircraft could also stop turning or turn less sharply, we generate a line for the path of the aircraft if it keeps its current velocity vector, tangent to the circle. Anywhere between the arc of the circle and the line—a planar, fin-shaped area—is a possible future position. This system approximates the fin with a triangle defined by the current aircraft position, the position the aircraft would occupy if it continued along the arc of the circle for the time step, and the position if it flies straight for the rest of the time step. [EDITOR'S NOTE: We omit the vector calculus details of the calculations involved.]

Requirement B

Applying vector calculus and multivariable analysis, we relate the characteristics of the flow of traffic in a sector to the complexity of the ATC's job in controlling the sector. By examining the influence of aircraft entering and exiting the sector in three different respects (instantaneous, over a time interval, and over a particular time of day), we refine the model.

Determining the Complexity Inherent in a Sector

The sector is defined by its size (boundaries) and by the objects that impact the flow of traffic through it. We assume that the sector extends from the ground upwards through all space (no ceiling) and is bounded by cylinder walls following the shape of the base of the sector (ground projection).

We model the airport as a vector field with the following properties:

- Aircraft are equally drawn from all points toward the location of the airport.
- The size of the airport determines the magnitude of the attraction impact on aircraft traffic.

The simple field to address these requirements is

$$A(x,y) = k_i \cdot \frac{(x_i - x)\vec{i} + (y_i - y)\vec{j}}{|(x_i - x)\vec{i} + (y_i - y)\vec{j}|}.$$

The field A_i for airport i points from any point in the plane (x, y) toward the airport location (x_i, y_i). Furthermore, we assign each vector a magnitude k_i, representing the impact of the airport on the traffic in the sector.

The properties for obstacle fields in the sector are:

- An obstacle's influence on a point is limited by the distance from the point to the center of the obstacle (obstacles create local effects).
- Physically larger obstacles impact aircraft farther from their centers than smaller obstacles do (Dallas / Ft. Worth has a greater influence on traffic than a municipal landing strip).
- The impact of the obstacle on traffic is related to properties like permanence, physical height, and the way that it impacts air traffic (a small town should have a lesser impact than a similarly sized downtown of skyscrapers).

We use the following function for obstacle intensity:

$$O_i(x, y) = -h_i \exp\left[\frac{(x - q_i)^2 + (y - w_i)^2}{-l_i}\right].$$

Points more distant from the center of the obstacle (q_i, w_i) get values closer to zero, and the variable l_i reflects how large (geographically) the obstacle is. Our final property is upheld by the use of h_i, which determines the relative impact of the obstacle. The negative sign ensures that the object repulses traffic.

The impact of a number of obstacles is just the sum O_{TOT} of their impacts, and we get the vector field for the obstacles by taking the gradient $B(x, y) = \nabla O_{\text{TOT}}$ of the total impact. This creates a field in which traffic tends to flow away from obstacles in the radial direction.

We then combine the obstacle and airport fields into a single flow field through simple addition.

Total Complexity

We characterize the complexity of the flow field that we have created. The flow of aircraft through a sector is analogous to the flow of fluid particles during bulk flow. In fluid mechanics, a laminar flow is marked by smoothness and predictability and is irrotational (no eddies). Turbulent flow, far more difficult to analyze, is choppier, less predictable than laminar flow, and rotational. Turbulent flow through the sector serves as our model for high complexity, while laminar flow is analogous to a sector with very low complexity.

The curl of a vector field measures the level of rotationality of a flow. We evaluate the magnitude of the curl of the vector field at every point and use this as the measurement of the total complexity of a sector.

Complexity by Number of Aircraft

We examine the impact of traffic volume on complexity of the workload for ATCs. We delineate three separate components:

- instantaneous complexity,
- complexity over a time interval, and
- complexity over a particular time of day.

From most demanding to least demanding, the tasks of an ATC are:

- adjust a plane's trajectory to avoid a potential conflict or collision,
- create a minimum spanning tree that highlights the critical relationships among aircraft,
- calculate the distance between aircraft,

- receive and record data from each aircraft, and
- communicate with ATCs in adjacent sectors (hand off aircraft to one another).

Definition of Complexity

We propose that the complexity of monitoring a sector can be defined similarly to the time complexity of an algorithm, in terms of the number of reference functions required to produce the correct output.

Instantaneous Complexity

We assume that something similar to our flight plan validation model is used by the ATC, screening potential conflicts long before the concerned aircraft enter the sector. The real-time complexity of the ATC's workload is then related solely to the need for corrections and the management of aircraft in the sector. The instantaneous case handles deciding if corrections are necessary, which requires examining the relationship between every pair of aircraft. But checking all of the $\binom{n}{2} \sim \mathcal{O}\left(n^2\right)$ distances between pairs is inefficient; a human operator visually inspecting graphical output should be able catch dangerous interactions between aircraft, at least in simple or routine situations. To do so requires the ATC to look at only $n-1$ interactions, examining only the distance between an aircraft and its nearest neighbor. In more complex situations, we would need a better method for determining interactions of concern.

An improved process for extremely vexing scenarios would be to employ a minimum spanning tree algorithm to determine the "edges" of interest, namely, the distance between an aircraft and its closest neighbors. By the definition of a minimum spanning tree, all airplanes that are very close together are connected by an edge, whereas those that are relatively far away from each other are not. Minimum spanning tree algorithms, such as Prim's or Kruskal's algorithms, have a complexity of $\mathcal{O}\left(n^2\right)$. At first glance, this would not appear to have an advantage over checking the distances between all pairs. But the minimum spanning tree would not have to be determined at every iteration; the tree could be reused for some number of time steps without significant loss of accuracy. Hence, we conjecture that the instantaneous time complexity of monitoring n aircraft falls between $\mathcal{O}(n)$ and $\mathcal{O}\left(n^2\right)$.[1]

Time Interval Complexity

We examine how complexity is related to the number of aircraft passing through the sector over a given interval of time. The difference between this

[1] EDITOR'S NOTE: Several collision-detecting algorithms are known to be more efficient in practice than $\mathcal{O}\left(n^2\right)$ without having theoretical guarantees. For references, see Eppstein and Erickson [1999]. They also give an algorithm to solve the problem in time $\mathcal{O}\left(n^{0.6897}\right)$ per collision using space $\mathcal{O}\left(n^{1.6897}\right)$, by means of ray shooting structures.

scenario and the instantaneous example is that there are now additional tasks beyond simply monitoring positions:

- receive and record updated aircraft positions,
- deal with aircraft entering the sector,
- hand off aircraft leaving the sector, and
- issue flight adjustments.

As before, we are not interested in how long these tasks take, or even how many are performed; our concern is how the number of these operations that must be performed is related to the number n of planes passing through the sector.

- **Updates:** To update the positions of the airplanes in the sector, n transmissions from the aircraft must be received and recorded.
- **Flight Adjustments:** We assume that any given aircraft has a probability p of requiring a path readjustment at a given instant. It is reasonable to conjecture that this probability increases by (at least) some constant amount for each aircraft added to the system, so we postulate that $p = cn$, where c is some constant. The probability that a given plane will require an adjustment, multiplied by the total number of planes in the system, provides an estimated readjustment workload for the ATC of $W = cn^2$.
- **Handoffs and Receptions:** For a plane entering the sector, its relationship with other aircraft already in the sector must be evaluated. Thus, entering aircraft require distance calculations. Aircraft leaving the sector require that the ATC send a radio transmission to the adjacent sector's ATC. An exiting aircraft is no longer tracked, so complexity decreases. We ignore the cost of handing an aircraft off, as it is in terms of cost.
- **Total Analysis:** The complexity of a time interval is considerably greater than before, as there is the monitoring requirement (conjectured to be between $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$), the flight adjustment requirement ($\mathcal{O}(n^2)$), the reception requirement ($\mathcal{O}(n)$), and the handoff requirement (conjectured to be $\mathcal{O}(n^2)$).

Complexity During a Given Time of Day

We define the flux of a sector over an interval as $\Delta n = n_{\text{final}} - n_{\text{initial}}$. We interpreted an interval to be relatively short—10 to 15 min.

Entering planes have a high complexity cost, while departing planes reduce complexity. Therefore, the times of greatest complexity are not just when n is at a maximum, but rather when $d\Delta n/dt$ is at a maximum.

Potential Conflicts

The most critical aspect of the ATC's job is to reroute aircraft when a potential conflict arises. Once a trajectory is corrected, we expect that the complexity from that problem returns to 0 but the complexity of correcting the next aircraft increases, because of more limited options. We therefore conjecture that the complexity of the addition of another needed correction is related to both the total number of aircraft and the number q of previously corrected aircraft that have not departed the sector. If k is the number of aircraft still requiring course corrections, each additional potential conflict increases total complexity by $\mathcal{O}\left(aq + bk^2\right)$, with a, b constants. We expect that increasing k has a greater impact than increasing q, because of the added demand of each correction.

Effect of Software Tools

The automated tracking of aircraft would safely allow more aircraft to operate in a given sector. The motivation would primarily be economic. Air traffic would become more complex as automated software increased the ability of the system to handle a complex situation. For an ATC, the situation would be no more complex, since it could be handled with the same effort as before.

Discussion

Part A: Random Effects Model

We develop a numerical method and implement a software program to analyze simulated flight paths. The results convincingly show the strength of this approach. In all cases, the model correctly predicts what would occur.

This model has three main strengths:

- It can easily accommodate the addition of aircraft to the system.
- It allows the ATC to be confident that the sector is devoid of conflicts before aircraft even enter.
- It is general enough so that we can make refinements as new data become available for different types of aircraft and situations.

There are, however, a number of important weaknesses in this model:

- Although the theoretical approach seems straightforward and simple, actually finding the largest value across numerous three-dimensional arrays is computationally foreboding. Since this program would ideally be real-time, the solutions must be achieved quickly (under 30 s).
- The model is not based on any actual data except for FAA guidelines. Ideally, the nature of the random effects could be determined and realistic standard deviations could be used.

- We lack data as to what constitutes a dangerous probability.

Part A: The Contingency Model

This model is used when aircraft are not following specified flight plans. The model has some very strong points:

- It allows the ATC to keep track of many unplanned paths at once.
- Using an array of blocks of space facilitates the addition of multiple aircraft into the tracking system.
- Continuous updating maintains a current view of where all aircraft are projected to be.
- The model accounts for aircraft turning, when pilots are more likely to miss seeing another aircraft.

The weaknesses of this model are due to some of the constraints on its operation and some of the approximations it makes:

- Because the model approximates possible future positions with a triangle, some space is considered that the aircraft could not possibly enter. We also do not account for space that the aircraft could get to by turning less sharply.
- After a certain amount of time, the position of an aircraft along a circle begins to return to its original position, making the triangle of future positions a bad approximation. Therefore, the model is limited in how long into the future it is useful.
- The model does not account well for sharper and sharper turns.

Part B: Inherent Complexity of a Sector

The only validation possible is to ensure that the model is consistent with intuition. We expect that sectors containing more objects (airports and obstacles) are more complex, and the model supports this by suggesting that these features add to the rotationality of the sector flow. In our implementation, the magnitude of the impact a particular object (storm, mountain, airport etc.) is based purely on conjecture. With experimental data and the input of actual ATCs, the model could be calibrated.

The strength of this model is that it provides a metric for complexity that is completely general. Obviously, calibrating the model would make it more useful, but it seems unlikely that new considerations would appear that would invalidate the basic approach and the related assumptions. The weakness of this model is that although analytically sound, it is extremely difficult to implement numerically.

Part B: Aircraft-Based Complexity of a Sector

The main strength lies in providing a method for analyzing how the role of the ATC would be affected by automated tracking software. The flaws are that

- the model relies on a loose analogy between algorithmic and air traffic handling complexities,
- the model has not been calibrated, and
- the mathematics of complexity for the operations that we have defined are not well understood (that is, our operations are not numerical calculations but procedures for which the numbers of calculations are not known).

Conclusions

DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION AGENCY
AIR TRAFFIC CONTROL DIVISION
ANALYSIS SUBDIVISION
WASHINGTON, DISTRICT OF COLUMBIA

DOT-FAA 7 FEBRUARY 2000

MEMORANDUM THRU: Dr. W. Roland Hamilton, Chief of Analysis, Air Traffic Control Division

FOR: Jane Garvey, Administrator, Federal Aviation Agency

SUBJECT: Summary of report conclusions from Project Star-Chaser

1. The purpose of this memorandum is to outline the findings of Project Star-Chaser and the ramifications that for operations and personnel of the Federal Aviation Agency and Air Traffic Control System.

2. Project Star-Chaser has arrived at two computational methods for alerting ATCs to instances where two aircraft come dangerously close to one another.

   (a) Random Effects Model: This model, if applied in ATC hardware, will allow operators to determine when the flight paths of two aircraft come dangerously close. This should give ATCs confidence that, provided aircraft stick to their flight plans, there is no chance that a multi-aircraft accident can occur.

   (b) Contingency Model: This model, if implemented in ATC software/-hardware, will provide a real-time analysis of the movement of aircraft in an ATC'S sector. If an aircraft deviates from its flight plan (having to circle above an airport during severe delays, for example), the ATC has a useful tool to help determine how the real-time behavior of aircraft lead directly to possible aircraft proximity conflict.

3. Perhaps more important than developing these computational tools, however, Project Star-Chaser has been largely focused on the systemic complexity associated with Air Traffic Control. We have broken the analysis of this complexity into several portions.

   (a) Inherent Complexity of a Sector: By examining the impact that objects have on a sector of airspace (namely, airports and obstacles), we have developed a means to analyze the complexity of a sector. This is the key to evaluating our current sector arrangement and determining if geographical boundaries should necessarily determine how the National Airspace is subdivided.

   (b) Instantaneous, Time Interval, and Time of Day Complexity: These three areas are iterative refinements of how the aircraft in the sector lead to increased complexity. Even with employing a minimum spanning tree as a best case scenario, the complexity of an ATC's workload is still $\mathcal{O}\left(n^2\right)$. This means that, under the present circumstance, increasing the number of aircraft leads to a much higher complexity for the ATC.

   (c) Impact of Collision Corrections: The Project further examined the complexity model to account for how complexity is affected by the number of course corrections needed. We find that this complexity is $\mathcal{O}\left(q+k^2\right)$, where q is the number of corrections already made and k is the number of corrections outstanding. This result suggests that a backlog of corrections greatly increases the complexity of the ATC's job.

   (d) Impact of Advanced Information Systems: We predict that the use of more advanced and autonomous hardware/software packages will reduce the job complexity for the ATC.

4. Advances will reduce the workload placed on ATCs and will increase efficiency and the volume of air travel.

   Unfortunately for ATCs, the end result of system improvements is no change. Though advances in guidance, tracking, and other technological areas seem to offer hope to reduce the stress and worry of the job, market and other economic equilibrium forces will quickly return the system to its maximum safe carrying capacity.

5. The Point of Contact for this memorandum is the undersigned at `J_Gibbs@Star_Chase.org`.

JOE W. GIBBS, Ph.D.
Head Functional Analyst,
Project Star-Chaser

Appendix A: Identification of σs

We assume that the net effect of variations in an aircraft's actual position compared to its predicted position is zero, since they are nonsystemic. In addition, the additive combination of random variations strongly suggests that the error term is normally distributed.

We do not have a good way of measuring the standard deviations in the three coordinate directions. Instead, we "reverse engineer" the standard deviations based on FFA guidelines for the minimum separation of aircraft in flight.

Table A1.

FAA guidelines and corresponding standard deviations.

Orientation	Altitude	Minimum separation	Standard deviation (km)
Laterally	N / A	5 mi	1.563
Vertically	<29,000 ft	1,000 ft	0.304
Vertically	>29,000 ft	2,000 ft	0.430

We determine the values for the standard deviations in **Table A1** by assuming that the FAA set guidelines so that there would be a .999 chance that the actual position of the plane would be within the accepted range; hence we use for each dimension the equation

$$\frac{1}{\sigma\sqrt{2\pi}} \int_{-\text{min dist}}^{\text{min dist}} \exp\left(\frac{-s^2}{2\sigma^2}\right) ds = 1 - .999$$

and solve for σ using MathCad.

Appendix B: Results of Simulations

Figure B1 (next page) shows the result for the collision probabilities procedure on data for planes whose paths intersect at the same point at the same time. **Figure B2** shows the result for a near-miss, paths that intersect in space but not at the same time. The maximum probability in the near-miss case differs from the maximum probability in the collision case by 11 orders of magnitude.

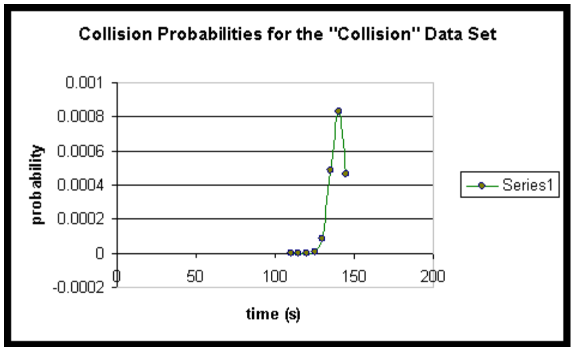

Figure B1. Collision probabilities for the “collision” data set.

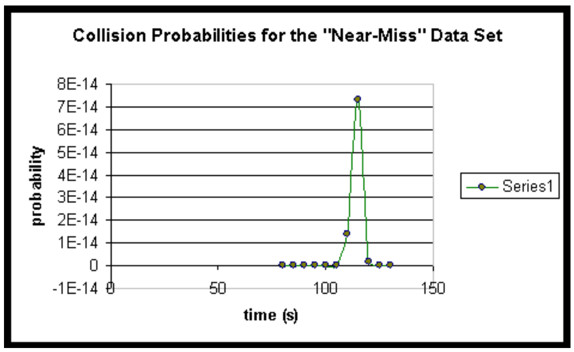

Figure B2. Collision probabilities for the “near-miss” data set.

References

Baird, D.C. 1995. *Experimentation: An Introduction to Measurement Theory and Experimental Design*. 3rd ed. Englewood, NJ: Prentice Hall.

Denver Air Route Traffic Control Center (ARTCC) Homepage. 2000. `http://www.tc.faa.gov/ZDV/` .

Devore, Jay L. 1995. *Probability and Statistics for Engineering and the Sciences*. 4th ed. Boston, MA: Duxbury Press.

Eppstein, David, and Jeff Erickson. 1999. Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions. Extended abstract at *Proceedings of the 14th Annual ACM Symposium on Computational Geometry* (1998), 58–67. *Discrete and Computational Geometry* 22 (4): 569–592. `\http://www.uiuc.edu/ph/www/jeffe/pubs/cycles.html` .

Hibbeler, R.C. 1998. *Engineering Mechanics: Dynamics*. 8th ed. Upper Saddle River, NJ: Prentice Hall.

Horowitz, Ellis, Sartaj Sahni, and Sanguthevar Rajasekaran. 1997. *Computer Algorithms / C++*. New York: Computer Science Press.

Rosen, Kenneth H. 1999. *Discrete Mathematics and Its Applications*. Boston, MA: WCB McGraw-Hill.

Schey, H.M. 1997. *Div, Grad, Curl, and All That: An Informal Text on Vector Calculus*. 3rd ed. New York: W.W. Norton.

Spiegel, Murray, and John Liu. 1999. *Mathematical Handbook of Formulas and Tables*. 2nd ed. New York: McGraw-Hill.

Stewart, James. 1998. *Calculus II. Special Edition for the United States Military Academy, Fall/Spring 1998/1999*. Boston, MA: Brooks / Cole Publishing Company.

Ugural, Ansel, and Saul Fenster. 1995. *Advanced Strength and Applied Elasticity*. 3rd ed. Upper Saddle River, NJ: Prentice Hall.