On Unverifiable "Facts"

David K. Wittenberg

March 10, 2003

Abstract

We consider physical *facts* which cannot be verified in practice. We are not interested in classic metaphysical or philosophical arguments which cannot be decided even in principle, but in facts where the answer is in theory easy to measure, but in reality, essentially impossible to measure. Most of the time, the facts we are interested in are probabilities of events which are too rare to measure directly, or which would involve experiments which are far too damaging to contemplate. One example is estimating the probability of catastrophic failure in a space shuttle launch or nuclear reactor.

1 Introduction

There are always debates going on about science, both among scientists and in the political arena. Many of these debates are trying to use science outside of the area where science is applicable[13], but a good number are about things which should be solvable. Why the debates when we could simply do the experiment? Sometimes, the debates are political, and the answers one gets correlate perfectly with the opinions one started with.

We study the cases where the questions are clearly within the realm of science, and could be answered given enough time and money to do the experiments, but the experiments are too expensive, time consuming, or dangerous to do. In most of these cases, the question can be phrased as "What is the probability of some event occurring?" These problems, which share the difficulty in measuring reliability, include the accuracy of ballistic missiles[10], safety of space travel[12, 5], and safety of commercial aircraft. By limiting our study to problems which clearly should be solvable, and noting the great difficulties in solving these comparatively simple cases, we show that the difficulties we encounter are inherent in calculating probabilities of unlikely events.

In those cases one tries to combine results from experiments which can be done in order to estimate the probabilities of interest. This is where the difficulty lies. When calculating (as opposed to measuring) probabilities, it is crucial to understand how each event's probability is affected by the probabilities of other events, ie. which probabilities are independent of each other. Figuring out which experiments are germane to a particular calculation is the issue which Barnes[2]

addresses with *similarity judgments*. Any school of scientific thought is based on agreed upon similarity judgments. These are implicit agreements about what issues are relevant in predicting behaviour, and, just as important, what issues are not relevant in predicting behaviour. In arguing that a measurable example justifies a conclusion about an immeasurable one, we need to argue that they are, in fact, similar. Different schools of thought with different similarity judgments will use different examples of a "similar" event, which can result in wildly different estimates. Duhem[4], writing in the late nineteenth century, describes a more basic problem in trusting physical measurments by pointing out the requirement for what he called "auxiliary assumptions". Duhem points out that when a physicist says he made a measurement (say of voltage), he is actually reporting the output of a relatively sophisticated device.[1] As the measurement devices get more sophisticated, and farther from simple observation, more and more assumptions are required for one to accept the measurement. As long as everyone shares those assumptions, science can proceed normally, but if those assumptions are not shared, it is barely possible to even communicate ones results.

1.1 Problems of Interest

My interest in estimating probabilities of rare events comes from attempts to ensure the reliability of computer programs. People realized that debugging programs was hard almost as soon as they started to program in the late 1940s. Wilkes[17] said "[in 1949] the realization came over me with full force that a good part of the remainder of my life was going to be spent in finding errors in my own programs." This was a surprise. It had been thought that building computer hardware was difficult, but programming was assumed (in the late 1940's) to be straightforward. By 1974, Fred Brooks could write a chapter called *No Silver Bullet* (in *The Mythical Man Month*[3] a witty and widely read description of problems in software engineering) and expect everyone to understand how hard getting programs correct is. So, I started with the question "How likely is it that a particular program gets the right answer?" in cases where it's clear what a correct answer is. These issues are of great interest in making software highly reliable.

1.2 Non-Problems

We are not interested in questions unless they have (in principle) clear answers. So arguments about the odds of a particular team winning the next World Cup or what people actually believe when they lie to pollsters are not covered here. In the first example, the statistical class is not large enough for the probability

[1] The issue here is that most of the quantities physicists talk about are not directly measurable, but are abstractions. To measure the length of something one puts a ruler next to it, but to measure voltage, one chooses any of several different methods. It is an assumption that different voltmeters, using different measurement methods based on different physical assumptions, actually measure the same thing.

to be well defined[16], and in the second, the question falls outside of what Medawar calls "The Limits of Science"[13]. We are only interested in facts (mainly the probabilities of events which are too rare to observe) which could be gathered if not for the danger or expense involved. The figures that we are interested in are, therefore, well defined. While statistics can be used to confuse an audience, we are not concerned with that here. We are assuming that the estimates are made in good faith, though the issues we discuss can certainly be used to confuse people.

By limiting our study to calculating well defined values, we show that even in the comparatively easy cases the problems are intractable.

2 Motivation

Why does this matter to an engineer? In specifying a system, one often specifies a failure rate. If the system is small and inexpensive enough (say a $10 electronic component), one can test a large enough number of samples to determine if the failure rate is acceptably small. With systems, end to end testing[15] (ie. testing the entire system at once) is the obvious answer, but that works only when we can afford to test several prototypes before finalizing the design. We are concerned with expensive or dangerous systems where it is impossible to run enough tests. These issues frequently appear in discussions of safety in designing things like bridges, nuclear power plants, or airplanes. In the early 20th century, there were over 1000 boiler explosions each year in North America[1]. Today we are much less tolerant of that number of casualties. We are particularly demanding of aircraft, as the FAA specifies a failure rate of 1 structural failure per 10^9 hours of flight. Note that most models of commercial aircraft never achieve a total of a billion flight hours over all the planes made of that model. This leads us to the problem in making extremely reliable systems: One cannot test them long enough to measure the failure rate, so it is hard to trust that the system is as reliable as it is specified to be.

With structural materials one can make a test part (smaller and weaker than the part of interest), test it to destruction, and then argue that the part in use is stronger by some factor which can be calculated from theories based on experience with parts of the same shape but different size. Similarly, one can argue that the number of (say) flex cycles before failure corresponds to some number of hours of flight, and that the thicker piece in actual use will last some (known) factor longer than the test part. With this sort of argument, one can usually get a fairly good idea of how likely a part is to fail.

This works quite well, and it's rare for a steel and cement structure like a bridge or a building to fall down. In most of the cases where a structure does fall down, it turns out that either the material or the application were not understood as well as we thought. A classic example is the Tacoma Narrows Bridge (often called "Galloping Gertie") that collapsed in 1940 as a consequence of a resonance excited by the wind[7]. The enquiry into the collapse of the bridge found that the design was not at fault, because even in retrospect they

did not have enough understanding of wind driven oscillations to predict the failure. This led to serious consideration of building a replacement bridge to the same design as the failed bridge. Theodore von Kármán (a member of the commission investigating the collapse) pointed out that if they rebuilt the bridge in the same way it would fall down in the same way. von Kármán later explained the failure in terms of vortex shedding (sometimes called Kármán vortex street). Petroski[14] points out that this sort of failure of prediction is necessary for engineering to advance, so we will have buildings falling as long as we keep designing structures with new materials or novel designs.

3 Why This is Hard

In systems which are less well understood than steel and concrete, one can not make arguments of this sort. Instead, one has to calculate the failure rate more indirectly, and this requires careful judgment, and thus can never be entirely convincing.

A well documented example of the difficulty in this sort of prediction was the safety of the space shuttle. Before the space shuttle "Challenger" exploded in 1986, top NASA management claimed that a failure that killed the crew would occur once in 10^5 flights, while low level engineers thought the failure rate of the solid rocket boosters was likely to be 1 in 10^2 flights[5]. Management and engineers differed almost as much in their estimates for failure rates of the main (liquid-fueled) engines. McConnell[12] describes other problems with the shuttle. How could different groups (both with access to almost all the known data) come to such different conclusions? Donald MacKenzie[10] gives one answer with his description of a "Certainty trough" (figure 1). This refers to a rough graph of perceived uncertainty in a system (on the vertical axis) plotted against social or intellectual distance from the system. The graph is bathtub shaped. The low area in the center represents people who know something, but not a great deal about the technology and who tend to have the most faith in it. These are the managers or others who are committed to using the technology. People intimately involved in technology (the high area at the left) know the uncertainties, while people alienated from the institutions or committed to a different technology (the high area at the right) think it won't work. Mackenzie[11] later elaborated on the certainty trough. A particularly clear example of the certainty trough was in the huge range of estimates of the reliability of the Strategic Defense Initiative (SDI, often ridiculed as Star Wars).

Software reliability is particularly prone to this sort of difficulty in estimating. One cannot use the usual sort of argument, where one tests weaker systems, and uses a physical theory to claim that the stronger one should last some calculated factor longer than the tested one, and from that argue that the Mean Time Between Failures (MTBF) is acceptable. Most material systems exhibit continuous behaviour, where a small change in the forces on a piece results in a small change in the amount of deflection. Software, on the other hand, is non-continuous. Sometimes, as in passwords, we rely on this (A tiny change in

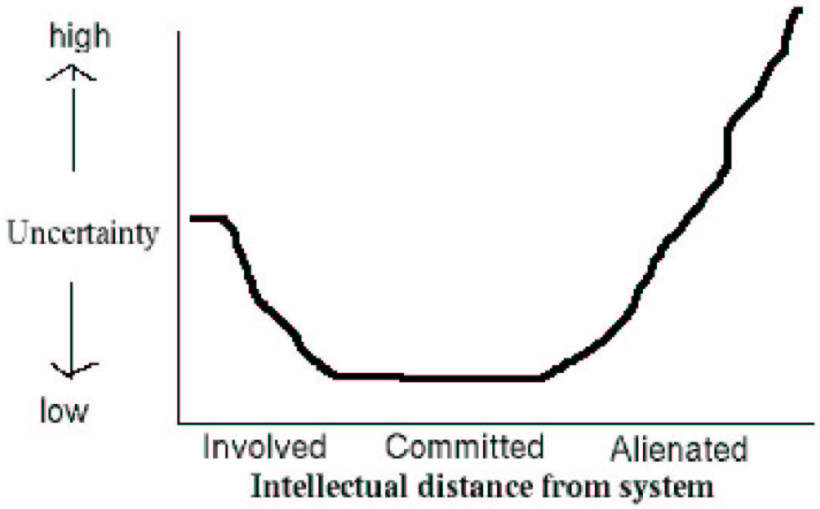

Figure 1: Certainty Trough (after Mackenzie[10])

input results in a huge change in output. A one bit error in my password would result in my not gaining access to my account.), but it makes testing extremely difficult.

The data we have on software failures shows a cumulative mathematical error of 1% per four thousand lines of code in extremely carefully written programs[8], so it is not possible to argue that some piece of code will work for 10^9 hours as FAA regulations require for critical parts of commercial aircraft. Other data gives the number of errors (found) per 1000 lines of code, but it is not clear how to calculate MTBF (mean time between failures) from that measurement.

4 Possible Solutions

Once we realize that good estimates of probabilities are both necessary and impossible, we have to find some way to make estimates.

Fault Tree Analysis is a favorite. One lists all the possible ways the system can fail, and then for each of those lists all the possible causes, and so on. If one is sufficiently prescient as to make these lists complete, fault tree analysis works fine. In a system of non-trivial size, this is an enormous undertaking, and one is likely to miss some faults (most famously the Maginot line, whose builders failed to realize that the German army could go around the line). Fault tree analysis or some variant of it form the basis of most of the arguments for high reliability. While it is hard to do a fault tree analysis of a system which has

not failed, it is relatively easy to point out the cause of a failure in retrospect. Arguments that something has low reliability can often be made by pointing out partial failures.

In the end, all techniques for predicting the probability of extremely rare events depend on a complete understanding of the problem. In some cases (say the probability of a particular atom of a radioactive isotope decaying in the next second), we have sufficient understanding to make reliable predictions. The cases of interest are, or course, those where we don't have a perfect understanding, either because the system is simply too complex to describe accurately, or in rare cases, because we must change the underlying model (what Kuhn[9] calls a paradigm shift).

5 Conclusion

There are decisions which we must make (for example: should we build nuclear power plants) which depend on unreliable estimates of the safety of a particular system. The decisions must be made, and the costs of the wrong decision are high, but the decisions can not be put off indefinitely.

So we are left with a problem: We can not make accurate estimates of the failure rate of some technology, but we need accurate estimates.

As there are decisions which must be made, we will continue to try to estimate risks, and we will live with the results. We can understand the difficulties in measuring the risks accurately, and we can hope to improve our estimates.

6 Acknowledgements

My thanks to Tim Hickey for suggesting this line of research, and to Alex Feinman, Seth Landsman, and Heather Quinn for their comments on a draft of this paper.

References

[1] American Society of Mechanical Engineers. The history of ASME international. http://www.asme.org/history/asmehist.html, July 2002.

[2] B. Barnes. *T.S. Kuhn and Social Science.* Columbia University Press, 1982.

[3] F. P. Brooks, Jr. *The Mythical Man Month.* Addison-Wesley, Reading, Mass., anniversary edition, 1995.

[4] P. M. M. Duhem. *The Aim and Structure of Physical Theory.* Princeton University Press, 1914. 1954 translation by Philip P. Wiener from the second (1914) edition.

[5] R. P. Feynman. *Richard P. Feynman's Minority Report to the Space Shuttle* Challenger *Inquiry*, pages 152–169. In [6], 1986.

[6] R. P. Feynman. *The Pleasure of Finding Things Out: The Best Short Works of Richard P. Feynman.* Perseus Books, 1999.

[7] A. F. Gunns. The first Tacoma Narrows Bridge: A brief history of Galloping Gertie. *Pacific Northwest Journal*, 72(4):162–169, Oct 1981.

[8] L. Hatton and A. Roberts. How accurate is scientific software? *IEEE Transactions on Software Engineering*, 20(10):785–797, October 1994.

[9] T. S. Kuhn. *The Structure of Scientific Revolutions.* University of Chicago Press, 1962.

[10] D. MacKenzie. *Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance.* MIT Press, 1990.

[11] D. Mackenzie. *The Certainty Trough*, pages 325–329. In Williams et al. [18], 1998.

[12] M. McConnell. *Challenger: A Major Malfunction.* Doubleday, 1987.

[13] P. B. Medawar. *The Limits of Science.* Harper and Row, 1984.

[14] H. Petroski. *To Engineer is Human: The Role of Failure in Successful Design.* St. Martin's Press, 1985.

[15] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems*, 2(4):277–288, November 1984.

[16] R. von Mises. *Probability, Statistics and Truth.* Dover, second revised English edition, 1957. Translated by Hilda Geiringer from the third German edition (1951).

[17] M. V. Wilkes. *Memoirs of a Computer Pioneer.* MIT Press, 1985.

[18] R. Williams, W. Faulkner, and J. Fleck, editors. *Exploring Expertise: Issues and Perspectives.* Macmillan, 1998.