19

Engineering Ethics When Lives Are on the Line: When Does Bad Engineering Become Bad Ethics?

Neil Siegel

19.1 Introduction

Numerous engineering projects create products and services that are important to society; many have explicit safety implications; some are distinguished by explicitly supporting national security. Failures and deficiencies that might be considered "routine" in some settings can in these cases directly cause injuries and lost lives, in addition to harming national security. In such a setting, decisions regarding quality, testing, reliability, and other "engineering" matters can become *ethical* decisions, where balancing cost and delivery schedule, for example, against marginal risks and qualities is not a sufficient basis for a decision. When operating in the context of an engineering project with such important societal implications, established engineering processes must therefore be supplemented with additional considerations and decision factors. In this chapter, long-time defense contractor executive and US National Academy of Engineering member Neil Siegel discusses specific examples of ways in which these ethical considerations manifest themselves. The chapter starts with his thesis, asserting that bad engineering risks transitioning into bad ethics under certain circumstances, which are described in the chapter. It then uses a story from the NASA manned space program to illustrate the thesis; unlike some stories, this one has a "happy ending." The author then moves to the main aspects of the chapter, starting by explaining the behavioral, evolutional, and situational factors that can tempt engineers into unethical behavior: how *do* engineers get into situations of ethical lapse? No one enters a career in engineering intended to put lives and missions at risk through ethical lapses; at the very least, this is not the path to promotion and positive career recognition. With the basis for such behavior established, the author then defines what he calls the *characteristics of modern systems that create risk of ethical lapse;* he identifies five specific traits of modern societal systems – systems of the sort that today's engineers are likely to be engaged in building – as being those that can allow people to slip from bad engineering into bad ethics. These characteristics are then illustrated with examples, from everyday engineering situations, such as working to ensure the reliability of the electric power grid, and designing today's automobiles. The very complexities and richness of features that distinguish many of today's products and critical societal systems are shown to become a channel through which bad engineering can transition into bad ethics. Lastly, the chapter discusses some of the author's ideas about how to correct these situations, and guard against these temptations.

19.2 Thesis

In my view, bad engineering risks transitioning into bad ethics when performing proper analyses *would* have indicated that major system problems are being overlooked in the specification and design of the system – but those steps are not performed.

Herein, I provide examples that illustrate the pervasiveness, subtlety, and potentially severe impact of such bad engineering ethics, and also identify a set of *specific system characteristics* that can trigger this type of ethic quandary. The systems that we engineers create – and will continue to create in the future – serve vital and increasingly pervasive roles in our society. We owe society, and ourselves, the very best that we can do. This is my motivation for talking about ethics in engineering.

19.3 A Story That I Was Told: The Return-to-Earth Orbit Design

I start with a story from the past; in my experience, such stories can help us form strong intuitions about what to do in the future.

My parents were both engineers who worked on the US manned space program: my father in propulsion, my mother in guidance. Through them, I met many other engineers who worked on that program. One of the more senior of these engineers[1] told me this story:

> "I was working on the *orbitology* team at Space Technology Laboratories"[2]; as the word orbitology implies, this team designed the orbits for the US manned space missions, including the Apollo program, which was intended to take humans to the moon, and return them safely to the Earth.
>
> The team conceived of the idea of designing the orbit so that if something went wrong, the space vehicle would coast around the moon and return to the Earth, even if for some reason no additional engine burns were possible. "We thought that this was a really good design, offering an entire additional layer of safety for the astronauts." So, the idea was presented to NASA.
>
> "To our surprise, NASA *hated* the idea"; the return-to-Earth orbit required more power at launch (that is to say, a slightly larger booster rocket) than some other orbit designs. It took a lot of time and arguing to convince NASA to adopt the idea. Fortunately, NASA eventually embraced the idea, and even published papers about it as if it were their idea in the first place.
>
> In the end, NASA selected a slight compromise: some of the earliest Apollo missions (which were only going to orbit the moon and return to the Earth, rather than land on it) used exactly the contractor-invented return-to-Earth orbit design, but for the actual moon landing missions, NASA used a variant of that design that required a small engine burn in order to get back to the Earth. NASA thought that this was okay, because the required engine burn was small – well within the capacity of the engine on the Apollo service module that they selected to use for this purpose – and they would program the computer in the command module to be ready, upon emergency, to do the required calculations.
>
> Everyone of course now knows that there was an explosion on the way out to the moon during the Apollo 13 mission. When that explosion took place, the *service module* was severely damaged, and because of that, the power to the *command module* was so limited that NASA had to turn off most of the command module systems (so as to conserve what little battery capacity remained for the actual re-entry phase of the mission). The astronauts were instructed to move to the *Lunar Excursion Module*. This move had its own issues: the Lunar Excursion Module was designed only to support two astronauts, not the three that had in fact to move into it during this emergency, and its batteries were sized only to support operations for about two days, not the five or so days that it would take to get the astronauts around the moon and back to the Earth. Dealing with the limited battery power capacity required turning off many of the devices and systems in the Lunar Excursion Module (in additional to having already turned off almost everything in the command module).
>
> One of the results of the explosion was that the engine on the service module that NASA had intended to perform the engine burn required to get into an actual return-to-Earth orbit was not available for use; the service module had suffered too much damage. Nor was the computer in the command module that NASA had intended to control this mid-course engine burn that would transform the orbit into a true return-to-Earth orbit available; the power to command module was largely turned off. NASA and the contractor team decided to perform this course-correcting engine burn using the engine on the Lunar Excursion Module that was intended to perform the actual moon landing; this was called the "Lunar Excursion Module *descent engine*."[3] NASA also needed a computer to control the timing and duration of the engine burn; since they were going to be using the engine in the Lunar Excursion Module, they decided to use a computer in the Lunar Excursion Module that was a part of what was termed the "abort guidance system"[4] to control the engine burn that would redirect the astronauts back to the Earth.[5]
>
> Fortunately, all worked well, and the astronauts managed to return safely to the Earth.[6] But to this day, some of the surviving contractor team members remain

> irate about NASA's attitude when the idea for a return-to-Earth orbit design was originally presented. NASA did not want to use the return-to-Earth orbit design because it would require a little more capacity in the booster-rocket system. The contractor team felt that the small additional expense was warranted; NASA did not. "I believe that we only convinced them to adopt this design by telling NASA that they were putting the lives of future astronauts at risk."

Cost-versus-safety is always a design consideration and an important system trade-off, but in this case the incremental cost of employing the return-to-Earth orbit design to the overall Apollo mission was so small that it could not even be measured (e.g., a slightly larger booster rocket), whereas the situation that could develop if the return-to-Earth orbit (or the small variant that was eventually adopted) had *not* been used would have made it absolutely impossible to rescue the astronauts in the event of an incident like that which actually transpired on the Apollo 13 mission. NASA personnel were reported by my contact as saying that it was "absolutely impossible" for so many things to go wrong that a return-to-Earth situation would arise. In this, of course, NASA was absolutely wrong – this sort of situation *did* in fact arise (and only on the third attempt to land on the moon; so, not only was this situation not impossible, it was probably not even particularly rare). My contact always stated that he felt that the issue that caused NASA to reject the idea originally was actually rooted in a "not invented here" syndrome; that is, since the idea originated with a contractor, rather than within NASA's own engineering staff, the NASA engineering staff rejected it. My contact went on to state that the contractor decided to allow NASA to write about the return-to-Earth orbit concept *without* making reference to the idea having been developed by a contractor. He believed that this was an essential part of the "socialization" that allowed NASA eventually to accept and adopt the idea; that is, the contractor allowed NASA to transfer "emotional ownership" of the idea from the actual inventors (the contractor) to NASA's own engineering staff, and that this made it easier for NASA to adopt the idea. If my contact's assessment of the motivations involved is correct, this approach is an important lesson about how to get useful things done!

Since the overall cost impact was so small that it could not even be measured, the normal sort of cost-versus-benefit analysis did not apply, and this situation can be viewed almost entirely as a near-lapse of engineering ethics on the part of NASA; that is, they nearly let their "not invented here" mentality add an intolerable risk to the mission. It reflects credit on NASA, of course, that in the end they came to a good decision and adopted the suggestion of using the return-to-Earth orbit. We can all be thankful for that!

19.4 How Do Engineers Get into Situations of Ethical Lapse?

No one enters a career in engineering intended to put lives and missions at risk through ethical lapses; at the very least – as mentioned earlier – this is not the path to promotion and positive career recognition. So how did this happen?

In his book *Fooled by Randomness* Taleb (2004) writes about the tendency of humans to *underestimate the likelihood of low-probability events*. That is, if an event is reasonably rare, humans tend to act as if the probability actually approaches zero. Taleb even cites sources that attribute this tendency to the deep operation of our brains, as developed through evolution. If this is true, it takes active effort to overcome such a tendency.

The problem with this tendency is that rare events do occasionally occur. This is why we ought to buy fire insurance for our homes, and collision insurance for our cars.

This tendency to underestimate the likelihood of *low-probability events* was probably relevant to NASA's attitude in the situation described in section 19.3; since they deemed the likelihood that a set of events that could necessitate a return-to-Earth event was "absolutely impossible" (my contact's phrasing: to put it in the language I use herein, he felt that NASA underestimated the likelihood of this particular low-probability event), NASA may have therefore decided that they were not *actually* creating the *ethical dilemma* that they did in fact create. This may help explain their behavior. No one would deny, of course, that NASA did

in fact have the responsibility to assess the probabilities reasonably. The contractor did so, and therefore was worried about the potential of catastrophe; NASA at first avoided such a realistic assessment, and therefore essentially excluded the potential of this type of catastrophe, and in turn (at first) elected to exclude the only possible feature that could be incorporated into the system design that could mitigate against this particular low-probability event.

A large organization like NASA that is attempting novel and difficult enterprises will perhaps face these sorts of issues more often that the average organization. This is not to say that NASA is less ethical than other organizations; it just suggests that NASA perhaps is dealing with the complex and the unknown more often than some other organizations.

19.5 Characteristics of Modern Systems that Create Risk of Ethical Lapse

As already mentioned, I believe that *bad engineering* risks turning into *bad ethics* when drawing from experience and performing correct analyses *would* suggest that major system problems are being unobserved in the design and specification of the system. However, but those steps are *not* carried out. In the case described in section 19.3, NASA nearly fell into this position.

I further believe that it is the case that modern systems exhibit *specific technical and social characteristics* that can lead to this particular type of *ethical quandary*. Among the specific system characteristics that can trigger this problem, I have already discussed (in the discussion about the NASA moon-shot program), the apparently-normal human tendency to discount the likelihood of low-probability events to essentially zero probability, and how that discounting can lead to ethical quandaries, not just to engineering and management quandaries. In this section, I will discuss four additional system characteristics that can trigger a transition from engineering risks to ethical risks:

1. The *complexity* and *scale* of modern systems.
2. *Reliability* and *availability* tend to be under-emphasized, as compared to functionality and capability.
3. We tend to accept *operator-induced* and *user-induced* failures as being outside of our design responsibilities.
4. We ignore – or seriously under-emphasize – the potential for use of the system beyond the uses that were originally envisioned, and also do the same for potential use beyond the originally-specified conditions.

19.5.1 Complexity and Scale Introduce Non-Linearities

Another important characteristic of modern systems is their *complexity* and *scale*. Complexity and scale introduce *non-linearities* in system behavior,[7] so that our intuition – which many people suspect basically operates by linear or proportional extrapolation (see, e.g., Kilpatrick et al., 2001) – is no longer even approximately valid. This can cause lapses in consideration of failure modes (among many other system characteristics), for example – which can in turn manifest themselves as unsafe operation. Many systems in fact exhibit serious failure modes that come entirely from scale, complexity, and the resulting errors in their *dynamic behavior*[8] – that is, scale and complexity can be actual *sources* of failure modes. This therefore can become another path via which we can create an ethical lapse through incomplete or inadequate engineering – we understand the scale and complexity of the system we are designing, but fail to account for the failure modes that such scale and complexity introduce themselves, above normal engineering considerations.

19.5.2 Reliability and Availability are Under-Emphasized

Reliability and *availability*, when they are under-emphasized in favor of focusing on system capabilities and functionality, can also become a source for lapses in engineering ethics. There is a natural tendency to focus on functionality and the visible features and capabilities of our systems, and therefore to under-emphasize quality characteristics (such as reliability

and availability, which are in some sense "less visible" to the eye of the intended users than system capabilities and functions). This tendency is reinforced by our contracts and system specifications: a typical specification for a big system will have most of its listed requirements dealing with functionality and capability (at times, as much as 99 percent of the requirements by count), and only a small portion (sometimes just 1 percent of the requirements by count) dealing with quality and usability factors. But reliability and availability – and other quality factors, too – are quite likely to be involved in safety and other important societal considerations, even though they did not require very many words in the specification to define their requirements. As a result, many system-development efforts perform only rudimentary analysis during design of these quality parameters. For example, the fault-tree might only identify the most obvious types of faults, completely omitting many faults that are equally impactful, but harder to see. The result is that the realized system reliability often is actually far less than the predicted system reliability. Another result of this behavior is that even if the occurrence rate of faults is as predicted, the severity (e.g., the impact of those faults when they occur on system operational effectiveness) is often far higher than predicted. Having system reliability much lower than predicted, and/or the impact of system faults being more severe than predicted, can have serious safety and other consequences – and therefore, once again, we have created an ethical lapse through incomplete or inadequate engineering.

19.5.3 Treating *Operator-Induced Failures* as Being Outside of Our Design Responsibilities

Yet another characteristic of modern systems that can cause ethical lapses is that we tend to accept the idea that operator-induced failures are *outside* of our design responsibilities. That is, if the user or operator of our system does something wrong (or even just something unexpected) and a problem results, we tend to say that it was *his or her* responsibility, rather than saying that *we* should have foreseen the possibility of such a mistake, and made the system react in a safe and predictable fashion, even in the presence of such "wrong" inputs.

This is an ethical lapse because it is 100 percent certain that at some point in the life of a system the users will "punch the wrong button," or create an input outside of nominal range, or provide some other "wrong" input or action. A robust design is precisely one that protects the system and its users against excessive adverse consequences from such an action.

Early versions of the DOS computer operating system, for example, did not even have a simple "Are you sure?" check when a command to erase a file (or even an entire directory of files) was entered. Even this simple example constituted negligent design (quickly corrected by Microsoft in later versions of DOS), as such "Are your sure?" queries were standard practice in many competing computer operating systems at the time. "Negligence" evolves into an "ethical problem" when the consequences can harm people. For example, why should a motor–generator accept a command to spin faster than it is designed to tolerate, when the consequence of such a command is that the generator may physically come apart, and that people could be injured or killed by that action (see Schneier, 2007)? The creators of StuxNet, of course, took advantage of just such a lapse on the part of the designers of the centrifuges being used to distill out heavy uranium isotopes.[9] But why should the microcontrollers of those centrifuges have accepted commands to operate outside of their known physical limits?

Those examples are of *single-point* instances of "wrong" user inputs causing actual physical damage. Much more common, and much more subtle, is the creation of physical damage through a *combination* of commands – none of which may be intrinsically unreasonable – that only in *combination* result in physical damage (and thereby, can cause injuries or death to people). Think of the chain of valves and pumps (these days, all of which are capable of being controlled remotely by computer commands) that operate an oil refinery or a chemical plant. With heavy, hot fluids moving through pipelines, control commands to a succession of valves and pumps must be properly synchronized, else the momentum of the column of moving fluid can burst a pipe wall, or cause other damage. A command to an individual valve or pump may be within the range of operation for that single device, but in

the context of the *total operating picture* of the facility, that same command may be a disaster. I assert that proper design and good engineering ethics require that we design our systems with the appropriate dynamic checks-and-balances that prevent command sequences that can combine to cause damage; and do so whether those commands are intentional or accidental. Very few of today's complex systems meet such a standard. I believe that we have a responsibility to protect our systems even against such operator-induced failures, and therefore also against hacker-induced and bad-inside-actor-induced failures, as well.

19.5.4 Ignoring the Potential that Our Systems Will be Used in Ways Other Than We Intended

Another characteristic (somewhat related to the example in section 19.5.3) is that we often ignore – or seriously under-emphasize – the potential for using our systems in ways that we did not envision, or beyond the specified operating conditions.

One simple example is using a screwdriver as a chisel (or vice-versa). Many of us have done that.

A more relevant example of this that everyone knows about is the *Internet*. The Internet was designed to share small bits of textual information between academic and scientific researchers; no other use was envisioned or specified at the time of its creation. Certainly, the use of the Internet for *safety-critical missions* was not imagined (Kleinrock 2013); yet today, an almost countless number of safety-critical and societal-critical missions are operated over the Internet. I will discuss the implications of this specific example in the case studies in sections 19.6 and 19.7.

19.6 A Case Study: The Electric Power Grid

I will now talk a little bit about the electric power grid, as an example of the quandary that society has gotten itself into; in this case by using the Internet for the sort of societal-critical missions that we are considering herein.

A modern industrialized country such as the United States depends in an essential way on a complex set of *interconnected technical infrastructures*, such as water, electricity, natural gas, gasoline, sewage treatment, road building and maintenance, traffic signals, food production and distribution, and so forth.

Following the principles formalized in Ricardo's *Law of Comparative Advantage* (Ricardo, 1817), society has become *specialized*. Whereas at one time most people were fairly self-sufficient – digging their own water wells, growing their own food, building their own shelters, and making their own clothes – this is no longer the case for the majority of people in the United States and the industrialized world. Instead, each individual person specializes in performing one type of task (e.g., growing wheat, teaching, programming a computer), and in essence exchanges his or her contribution on that specialized task for the remaining goods and services that he or she needs. Money was long-ago recognized as a more efficient medium of exchange under such circumstances, as compared to barter. If I depend on barter and have grain and need meat, I must find someone who has meat and wants grain. As you can see, having a freely convertible medium of exchange (e.g., cash) making exchange easy is of enormous economic benefit. As a result, most of us work for cash wages, and purchase – rather than make – each of the aforementioned critical infrastructure services such as water, food, and so on.

While such specialization lies at the root of our economic progress over primitive societies, and hence at the root of increased human life-span[10] and other obvious benefits, we are most of us now dependent on the continuous operation of these *critical* technical infrastructures. Few of us have artesian water wells on our property, yet we can only live for about three days without potable water. Even fewer of us, especially in the cities, grow our own food.[11]

Those who design and operate these critical infrastructures are responsible people, and they have endeavored to make them reliable. What they have not done, however, is made

adequate provision to defend against *deliberate* attempts to undermine, disrupt, or destroy these infrastructures; nor in many ways have they dealt effectively with undesired *emergent behavior* that arises not from their individual component of the overall system, but from the complex interactions among all of the components of their system. In their defense, they have not been tasked to do this, or not authorized by their ratepayers to incur these costs.

In the "post-9/11" world, this is increasingly recognized as a gap that must be corrected. For example, multiple nations have tested or are believed to be developing *electro-magnetic-pulse* weapons that are designed explicitly to attack the electric power grid over a large area with a single attack (Foster et al., 2008). There are a few rare natural phenomena (e.g., extreme space weather) that could cause similar disruptions – they are rare, but they do periodically occur!

Electricity plays a key foundational role among these key infrastructure services. For example, water must be pumped to be available, and most of that pumping is directly powered by electricity. In turn, many of the electric power plants are powered by natural gas, but much of that natural gas is produced far from where it is used. This mean the natural gas must be pumped from its production locations to the locations of the power plants (several thousand separate sites in the United States alone) – and that pumping is in turn often powered by electricity. In particular, while many natural gas pumping stations are in fact powered by the gas in the pipeline itself (using about 3 percent of the gas energy for this purpose), increasingly (and in my view, unfortunately), recent installations and upgrades (including many in critical gas-distribution locations, such as Houston, Denver, and California) have instead used electricity to power such pumping (Judson, 2013).

Furthermore, electricity has another nearly unique characteristic: we have little capacity to store electricity upon rapid changes in demand. Yet demand must match generated power to an astonishingly accurate degree, and on sub-second response-times. As a result of these characteristics, in order to recover from a large-scale power outage, those facilities generating electricity must coordinate closely with the major users of electricity (water pumping, water treatment, etc.) as generation capacity is brought back online: generation and electric load must be brought online and then kept exactly in balance. If generation at any instant exceeds demand, voltage and/or frequency can go up, damaging equipment. If generation at any moment falls short of demand, voltage and/or frequency can go down, causing equipment to shut off, perform outside of specification, or be damaged. But with all forms of real-time communications off-line due to this same power failure, how is this coordination – which is necessary for the electric grid to be restarted – going to be performed? Once the grid is up and running in a steady state, generators can monitor voltage and frequency, and make small adjustments without having explicitly to coordinate with the users of that electricity, but to start the grid from "off", such coordination is required; otherwise, the fluctuations would be far too large, and cause damage to equipment.

The result is a complicated set of interdependencies among these critical infrastructures, and among the people and organizations that operate, pay for, and regulate them, as illustrated by Figure 19.1 (Siegel and Ferren, 2016).

Many of these mission-critical and safety-critical links are implemented via the Internet, but as already noted, the Internet was not in fact designed to support such critical applications. Specific ways in which the Internet falls short of properly supporting these sorts of critical missions include the following:

- The variance in packet delivery time on the Internet is practically unbounded, and is at best large as compared to networks that are designed specifically for critical-mission use
- The packet successful-delivery-on-first-attempt rate on the Internet is considerably smaller than that achieved by networks specifically designed for critical-mission use.
- The availability of service between any two points on the Internet is considerably lower than that achieved by networks designed for critical-mission use.

Any of these characteristics of the Internet can contribute to irregularities, errors, wrong answers, and failures of service for the mission-critical and safety-critical missions supported by the Internet, including the electric power grid and the critical infrastructure services (water, sewage treatment, etc.) that depend on the electric power grid.

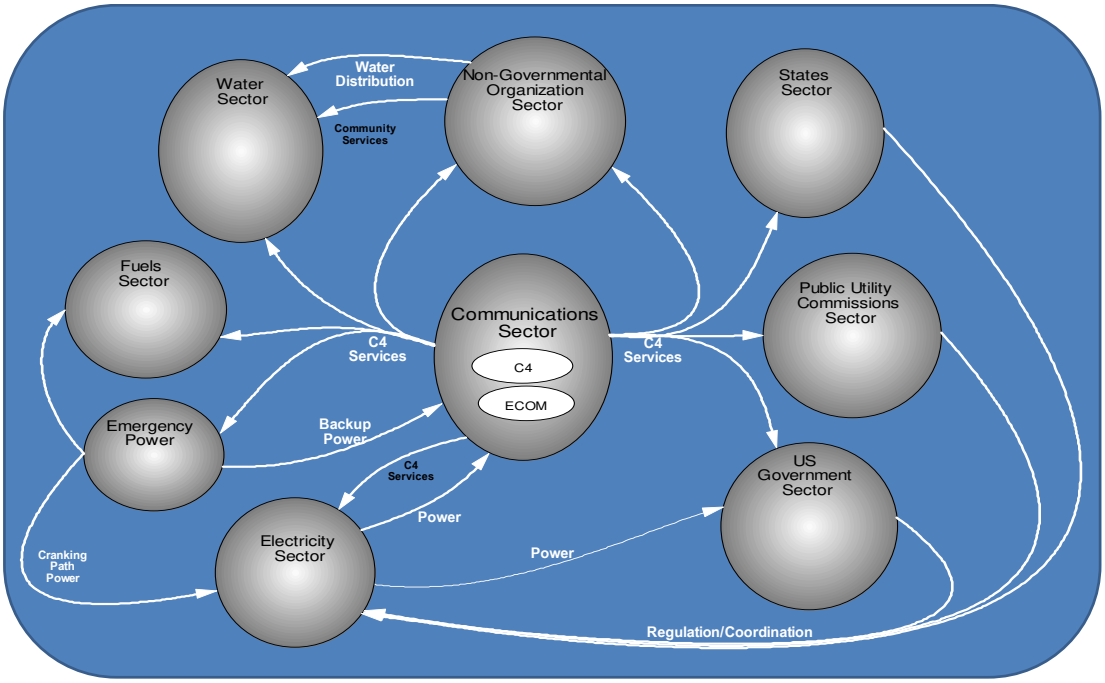

Figure 19.1 Complex interactions among the critical infrastructures, their stakeholders, and their regulators.

Source: Electric Infrastructure Security Council

Given this combination of nature of the electric grid, and the limitations of service for safety-critical missions provided by the Internet, it is easy to see the cascading disaster that could result from a large-scale electric outage in an industrialized country.

- Due to a man-made (e.g., electromagnetic pulse (EMP) attack or cyberattack) or natural (e.g., severe space weather) event, electric service would instantly go out, and (due to the recent substitution of computer-controlled switches to control the power grid, rather than the older electromagnetic relays) potentially over a far wider area than previous power outages. The computer-controlled switches offer great convenience when the power is on, but are vulnerable to EMP and cyberattack, whereas the older electromagnetic relays were not.
- Other critical infrastructure services that consume very large amounts of electricity (e.g., water pumping, water and sewage treatment, etc.) would go out, too. Some of these sites have diesel generators to replace a small portion of their electricity needs, but (due to safety concerns) most such sites are not allowed to store more than a nominal amount of diesel fuel.
- Communications systems (telephone, Internet, satellites and satellite phones, cellular phone base stations, and switching centers) might be instantly disrupted, depending on the cause of the event (e.g., if the cause of the event is an EMP attack, many of these services would go off the air instantly). Those that did not go off the air instantly would go off the air within four to eight hours, as their battery back-up systems exhaust their capacity. As a result, utility company emergency managers could not communicate with their field operatives or with the major users of electricity; government officials would not be able to communicate with utility company managers; the police and nongovernment organizations (NGOs) could not coordinate actions; no one would know where to send supplies and people with critical skills; and so forth.

- People would attempt to leave their cities, in search of somewhere where power might be on (and remember that with no electricity, no information about what to do or where to go could be provided to the public – radio and television stations, too, would be off the air in the event of a power outage). The roads would become completely clogged, and even emergency vehicles could not get through. After a few days, this congestion would get *worse*, as cars that are out of gasoline would be abandoned on the road. Even gas stations that have gasoline in their underground tanks need electricity to pump that gasoline up to the cars – and the electricity is off.
- High-rise buildings would become uninhabitable within a few days, as sewage would back up on every floor (sewage is *pumped* out of a high-rise building).
- Food in grocery stores, restaurants, homes, and storage centers would rot.
- The sewage overflows and the rotting food would become sources of infection; the combination of a lack of food and consistent sources of potable water (you can't provide water to a large city for very long out of little plastic bottles, even if necessary trucks could get fueled and get through the clogged roads) would reduce people's resistance to disease, and massive outbreaks of dysentery, cholera, and so on would occur.

Estimates are that such a scenario could kill *tens of millions* of people in the United States alone, and that it could take years (or even decades) before power was restored and the economy back to normal (EIS Council, 2017).

The electric grid and its vulnerability is therefore another example of the creation of physical damage through a *combination* of activities. Each portion of the electric grid is designed sensibly within its own small domain of operations and responsibility, but no one apparently has the responsibility to assess and protect the grid against the damage that comes from such a *combination* of interactions. Yet the critical infrastructure of the industrialized world depends on such a combination of interactions. For example, if electric power goes out, natural gas stops being pumped, and since most of the electricity in the United States comes from burning natural gas, the power plants cannot be re-started because they don't have natural gas. The natural gas pumping stations either don't have back-up diesel generators or they are not allowed to store enough diesel fuel on site to restore steady-state operations. The trucks that were planned to deliver that extra diesel fuel cannot get through because of clogged roads; and so forth. I again assert that proper design (and good engineering ethics) requires that we find a way for society to account for (and to pay for!) methods to prevent and/or recover from these sort of multi-domain/multi-step events. As we have seen, the power grid does not yet meet this standard (although I am glad to report that this problem is at last beginning to attract interest[12]).

19.7 An Additional Case Study: The CAN Bus

Computer hacking and sabotage are also just beginning to be understood as a real threat to society's safety and well-being. Hacking is not just a threat to private data. Consider the physical damage that could be caused by hacking traffic signals, water pumping stations, and so on and so on. I believe that we in the engineering profession have a responsibility to design our systems so that the impact of attacks on their capability is minimized, at least to the point of protecting human lives. Yet when I have approached industries (e.g., banking) offering to improve their cybersecurity posture, a common response is that "We are waiting for the government to establish guidance via statutes, because if we do things in the absence of such laws, our stockholders and regulators will attack us for spending money needlessly." We engineers have to take a role in leading society out of this dilemma; driving some of the engineering studies into our systems designs in advance of the law requiring it; providing information to professional societies and others who can work to get appropriate laws and regulations enacted; creating demand for all of the above through educating the general public; and so forth.

I provide one last example of what I think is a bad design enabled by a lack of thinking about the ethical implications of a design: the automotive CAN bus.

A Controller Area Network (CAN) bus is an electronic communications path, intended for use in a vehicle (like an automobile) that is designed to allow microcontrollers and other devices to communicate with each other without the use of a host computer. The specific version used in cars was apparently started at Robert Bosch GmbH (a German automotive electronics company), and was officially released for use by the automotive industry in 1986 (CAN in Automation Group, retrieved 2018). The design decision adopted throughout the automobile industry was to place *all* of the microcontrollers and electronics in the *entire* car on a *single* such bus: engine controls, brake controls, transmission controls, and other motion-related (and hence safety-related) items are on the same bus as the radio and air conditioning.

In contrast, in other types of vehicles that employ extensive electronic controls (such as warships), the state of practice has long required that *multiple independent* buses be employed, so as to separate the control of safety-critical and/or mission-critical items from "convenience items." For example, on the typical US warship, there is one bus that connects all of the devices that control the basic movement of the ship (e.g., power generators, engines, steering apparatus, etc.), a second bus that connects all of the devices that control the military functions on the ship (e.g., sensors, weapons, etc.), and still a third bus that connects the "convenience items" (e.g., non-emergency lighting, recreational devices, etc.). Good engineering – and good engineering ethics – would have had the designers who were implementing an automotive CAN bus in a specific vehicle looking at the rationale behind this well-known warship design practice, and considering how those rationale and lessons-learned would apply to passenger cars. This *ought* to have resulted in a design for a communications bus system for automobiles that did not, for example, allow hackers to gain remote wireless access to a car via the audio entertainment system and use that access (since there is only a single bus) remotely to "take over" motion-critical and safety-critical items – such as being able to accelerate the car without any action on the part of the driver (and the driver not being able to "override" this acceleration), to prevent the driver from turning off the ignition or taking the transmission out of "drive," and even to disable the brakes. Yet this is precisely what has come to pass (for examples, see Greenberg, 2015; Koscher et al., 2010). Such poor design, especially when well-known examples were available of better approaches (e.g., warship control systems), verges, in my view, into a serious lapse of engineering ethics. We are just beginning to learn the cost of this particular lapse.

In addition, there are "second-order" effects that should have been considered, but evidently were not. For example, the audio system in most modern cars is so powerful that it can create sounds that are so loud as to constitute a serious distraction to the driver; actually causing pain in the ears, and so forth. Therefore, a remote hacker who "only" was able to seize control of the sound system of a car (even without trying to take over the engine, brakes, etc.), could still seriously degrade the ability of the driver to safely control the vehicle.

19.8 Corrective Actions

Having defined and illustrated via examples the problem, I wish to present some ideas about how to avoid the problem.

First and foremost, in my opinion, is the matter of placing proper emphasis on good design. At present, I believe that it is fair to say that most engineering projects place their primary emphasis on developing good requirements. Many texts and corporate guidelines about performing systems engineering, for example, are heavily focused on the matter of requirements. Many academic papers that examine the question of problems in the system development process focus on requirements, too, citing factors like incomplete requirements or "requirements creep" (e.g., the problem that the requirements continue to change, even as the design and implementation progress) as the root cause of the large number of engineering projects that have significant problems. It is natural for the customers and eventual users of the system to focus on the requirements, too; after all, the requirements are something that they can understand, and are also something that they have a natural reason for wishing to influence.

There is no doubt that such problems occur, for example, engineering projects often end up costing much more than promised, usually accompanied by taking significantly longer than predicted, failing to implement all of the promised capabilities. The data indicates that a large portion of engineering projects are terminated before their completion, due to these factors. But I spent several years of my engineering career as a sort of "designated engineering project fix-it person," and what I found was *not* that engineering projects that were in trouble had bad requirements; instead, what they had consistently was *bad designs*.

I have also seen cases of two completed systems that do approximately the same thing, where one runs 100 times faster than the other. Similarly, I have seen cases of two completed systems that do approximately the same thing, where one is 1000 time more reliable than the other. Having for these examples also had the opportunity to examine the root cause for the slower and less reliable performance, I can assert that the systems at the bad end of these examples had bad designs.

This finding has many interesting implications. First of all, having a 100x or 1000x range of outcomes for a critical parameter from an engineering project is shocking; mature engineering disciplines simply do not have such large range of outcomes. Consider mid-sized family sedans offered for sale that meet US emissions-control requirements; the variation from best to worst, for example, in gas mileage is no more than 25 percent, not 100x (10,000 percent) or 1000x (100,000 percent). Something is going radically wrong inside the designs of the systems that exhibit such bad performance on such an important metric.

I drew on this experience in fixing troubled engineering projects in my Ph.D. research. In my Ph.D. dissertation I develop and attempt to validate a hypothesis describing exactly in what way are these designs bad, and how could they have been improved. I am not the only person to have examined the question of how to accomplish an effective design for a complex engineering system.

Creating a good design for an engineered system is, in my experience, far more difficult that developing the requirements for that same system. Furthermore, we get a lot of "help" as we develop the requirements for an engineering system; after all, our customers and our users understand well *what* they want the new system to do, and such *what* constitutes a major portion of the requirements.

We do not usually have such a resource pool to help us with the design. The design is far subtler than the requirements, interactions between elements of the design are far more likely to have significant impacts on the system and its performance than are interactions between elements of the requirements, and the design is far more technical (and hence opaque to many observers). Furthermore, most projects do not have reasonable technical metrics for measuring the progress of the design; they tend to use only management metrics for measuring progress on the design (e.g., we held these reviews, we produced these documents, etc.).

Engineered systems almost always aspire to create some sort of emergent behavior, a sort of "1 + 1 = 3", where useful things happen due to the interaction of formerly separate elements. But what also happens is, while creating a design that produces the desired emergent behaviors, the design fails to *prevent* the arising of unplanned emergent behaviors that appear as unintended adverse consequences.

As a result, unintended adverse emergent behavior often creeps into our systems through such incomplete designs. In my judgment and experience, this is the *true root cause* of most failures of engineering project developments (rather than requirements creep, etc.). Such incomplete design – that is, a design that does *not* incorporate features explicitly aimed at preventing such unintended emergent behavior (in my Ph.D. dissertation and other writings, I often call this "unplanned dynamic behavior") – is likely to exhibit the poor characteristics that I have described, and therefore these are the projects most likely to fail and/or be cancelled.

So, improving the design and the design process is "step 1," in my view, towards avoiding the problem of bad engineering transitioning into bad ethics.

After the design, I believe the next most important corrective is the risk management process. Most big projects have some type of formal risk management process; the process itself is usually pretty rigorous. What is lacking, in my experience, is *content*. I have found that the risks contained on the risk register of an engineering project are often completely superficial and general. I have actually seen the statement "The software might be late" as an entry on the risk register in a multi-billion-dollar engineering project. Such a statement is useless as a risk register entry. First of all, it is true on every project that has a material amount of software (as almost every engineering project does these days), therefore it is not specific in any fashion about this particular project. But far worse is the fact that it contains no insight about what the project should be measuring every month in order to determine if in fact the risk is coming to pass, or what steps should be taken to mitigate the impact if and when they determine, through those measurements, that the risk is coming to pass.

So the next corrective is generating much more specific and far more measurable entries on the risk register, and then taking the corresponding actions in the rest of the risk management process; for example, figure out what to measure, work out how to make those measurements, create mitigation plans if the risk materializes, and so forth.

Of course, doing this takes intense (and expensive) effort, expertise, and a lot of time. It also results in there being many more items on the risk register! These are probably the reasons that it is not done properly more often.

19.9 Conclusion

In my view, *bad engineering* can transition into *bad ethics* when proper analyses and experience *would* have shown that serious system problems are being overlooked in the specification and design of the system – yet those steps are not performed.

Examples have been provided that illustrate the pervasiveness, subtlety, and potentially severe impact of such bad engineering ethics.

I further believe that it is the case that modern systems exhibit *specific technical and social characteristics* that can lead to this specific type of ethical quandary. Examples of such system characteristics that can trigger this quandary were discussed, including the following:

- The human tendency to discount the likelihood of low-probability events to essentially zero probability
- The complexity and scale of modern systems
- Reliability and availability tend to be under-emphasized, as compared to functionality and capability
- We tend to accept *operator-induced* and *user-induced* failures as being outside of our design responsibilities
- We ignore – or seriously under-emphasize – the potential for use beyond the uses of the system that were originally envisioned, and for potential use beyond the originally-specified conditions.

Societal expectations for engineering are very high; whereas a baseball player has only to succeed (e.g., get a hit) 30 percent of the time to be considered a major success, in contrast, society's expectations for engineered products and systems is near 100 percent availability and correctness. I believe that, in turn, this expectation grants us license to insist on proper designs, based on proper analyses, especially in safety-critical and mission-critical situations.

I also wish to plea for practicing engineers to believe that developing a personal reputation for thoroughness, diligence, and good engineering ethics is a boon, not a liability, to one's individual career. I cannot "prove" this, but most of my experience over nearly forty years as a practitioner supports that conclusion. One last little story: When I was the vice-president and general manager of an operating division at a large aerospace company, we elected to bid on a competition for a new type of system for the US Marine Corps. The system specification had a "hard limit" of 11,000 pounds for the complete system, because this was the capacity of the specific vehicle chassis that we were to use in building the

system. When it came time to submit our bid, my proposal manager pointed out that when they added up all of components of our proposed design (which we thought was wonderful, and would offer the Marines a lot of operational advantages), we were slightly over that weight limit; as I recall, 11,045 pounds. My proposal manager asked what we should do; the implication was for me to choose between fudging the analysis to make it say "10,999" pounds, or to submitting the proposal as-is (i.e., over the specification weight limit) and assume that the Marines would appreciate the honesty, understand that 1000 design decisions remained between the proposal and the fielded system, and that there was plenty of time to solve the weight problem before we were done. I said to submit it as-is. And we won – despite being overweight. Many, many years later the proposal manager came to me and said how much he and the proposal team had appreciated my having taken the ethical approach in that situation.

Notes

chapter-references

References

CAN in Automation Group (retrieved 2018). History of CAN technology. Retrieved from www.can-cia.org/can-knowledge/can/can-history/

EIS Council (2017). EPRO™ Black Sky Event Simulation Project. Some of the key results of this simulation scenario are reported at the website. Retrieved from EIS Council http://eiscouncil.org/Epro/SimulationProject

Foster, J. et al. (2008). Report of the Commission to Assess the Threat to the United States from Electromagnetic Pulse Attack. A commission chartered by United States public law 106-398, Title XIV.

Greenberg, A. (2015). Hackers Remotely Kill a Jeep on the Highway – with Me in it. Retrieved from www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

Koscher, K. et al. (2010). Experimental Security Analysis of a Modern Automobile. IEEE 2010 Symposium on Security and Privacy.

Judson, N. (2013). Interdependence of the Electricity Generation System and the Natural Gas System and Implications for Energy Security, MIT Lincoln Laboratory Technical Report 1173.

Kilpatrick, J., Swafford, J., a7 Findell, B. (Eds.) (2001). *Adding it up: Helping children learn mathematics*. National Academies Press.

Kleinrock, L. (2013). Personal communication. Kleinrock, still a professor at UCLA and who made important contributions to the design of the Internet (e.g., hierarchical routing), often says this in his lectures and speeches.

Kushner, D. (2013). The real story of Stuxnet. Retrieved from http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet

Ricardo, D. (1817). *On the principles of political economy and taxation*. London, UK: John Murray.

Schneier, B. (2007). Staged attack causes generator to self-destruct. Retrieved from www.schneier.com/blog/archives/2007/10/staged_attack_c.html

Siegel, N., & Ferren, B. (2016). The figure is from "Emergency Communications System (ECOM), A technical report for the electric Infrastructure Security Council. Retrieved from www.eiscouncil.org

Siegel, N. (2011). Organizing projects around the mitigation of risks arising from system dynamic behavior. *International Journal of Software Informatics* 5(3).

Taleb, N. N. (2004). *Fooled by randomness*. Random House.

[1] Who prefers that I not use his name.

[2] A NASA contractor. This company was later re-named "Thompson Ramo Woolridge" and was usually known by its initials "TRW." TRW was acquired by Northrop Grumman in 2002.

[3] The author's father was a member of the team at STL that developed this Lunar Excursion Module descent engine.

[4] The abort guidance system was intended to guide the Lunar Excursion Module to a rendezvous with the command module in the event that the mission commander decided that he could not finish a moon landing, and instead had to abort the landing and return to the command module, which was waiting for them in an orbit around the moon. Since the motions of the LEM near the moon were "free-flight," under the control of the LEM commander, the return-and-rendezvous course could not be pre-planned, but instead had to be calculated in real-time using telemetry and instrumentation; to do that required a guidance computer.

[5] The author's mother was a member of the development team for this abort guidance system.

[6] Among the "thank-you" visits that the Apollo 13 astronauts made after safely returning to Earth was a trip to Space Technology Laboratories in Redondo Beach, California, to thank the teams that created the LEM descent engine and the abort guidance computer. Both of my parents were part of the teams that met the Apollo 13 astronauts during this thank-you visit. The author was a teenager, and had the opportunity to accompany his mother during this visit.

[7] That is, small changes in an input can lead to more than a small change in an output; at times, small changes in an input can lead to gigantic changes in an output.

[8] The subject of "unplanned dynamic behavior" in a system, and how to design systems that exhibit markedly less on this undesirable behavior, is discussed in Siegel (2011).

[9] StuxNet was a computer-based attack on the Iranian nuclear program, which allegedly caused physical damage to Iranian centrifuges being used to weaponize uranium, by commanding those centrifuges to operate outside of their specified physical limits, for example, to spin too fast, to change speed too fast, and so forth. Many descriptions and analyses of the StuxNet endeavor are available. More information is available in Kusher (2013).

[10] After remaining constant for several hundred thousand years at a life-expectancy of about thirty-five years, human life-span has doubled (from about thirty-five years to approximately seventy years) more or less exactly in coordination with the creation and adoption of the industrial revolution, which is simply the period where humanity first intensely implemented the economic specialization described here. Note that the US National Academy has studied the causes of such increased life-span, and attributes most of that increase to the *engineering accomplishments* that created these technical infrastructures: water, sewage treatment, refrigerators, self-propelled tractors (and the fuel and parts supply chains that keep them running) – and *not* to modern medicine. In substantiation of this conclusion, the Academy notes that locations and countries that by and large do *not* have modern medicine have still achieved much the same life-span increases if they have adopted economic specialization and these technical infrastructures.

[11] As recently as during the Theodore Roosevelt administration, more than 90 percent of the US population engaged in farming or animal husbandry of some sort. Today, that figure is about 3 percent.

[12] As evidenced by the formation of the Electric Infrastructure Security Council (http://www.eiscouncil.com/), among other indicators.