S E S 2 0 2 1
Seventeenth International Scientific Conference
SPACE, ECOLOGY, SAFETY
20 – 22 October 2021, Sofia, Bulgaria

SAFETY IN AEROSPACE ENGINEERING

Adelina Miteva

Space Research and Technology Institute – Bulgarian Academy of Sciences
e-mail: ad.miteva@gmail.com

Keywords: *Safety, aerospace engineering, reliability and safety in engineering design*

Abstract: *Safety is the most important aspect of all areas of technology. Particularly in aerospace engineering, safety is critical as it directly affects people's lives. However, modern systems engineering is becoming more complex and often includes multiple components, countless software lines, globally deployed development teams, and complex operating environments. As a result, it is becoming increasingly difficult to secure these complex systems. Here some aspects of system safety in aerospace engineering are presented.*

БЕЗОПАСНОСТ В АЕРОКОСМИЧЕСКАТА ТЕХНИКА

Аделина Митева

Институт за космически изследвания и технологии – Българска академия на науките
e-mail: ad.miteva@gmail.com

Ключови думи: *Безопасност, аерокосмическа техника, надеждност и безопасност в инженерното проектиране*

Резюме: *Безопасността е най -важният аспект на всички области на техниката. Особено в аерокосмическата техника безопасността е от решаващо значение, тъй като пряко засяга живота на хората. Съвременното системно инженерство става все по -сложно и често включва множество компоненти, безброй софтуерни линии, глобално разгърнати екипи за разработка и сложни операционни среди. В резултат на това става все по -трудно да се защитят тези сложни системи. Тук са представени някои аспекти на безопасността на системите в космическата техника.*

Introduction

Nowadays, the topic of system safety in aerospace engineering receives a lot of public discussion and attention: the investigation of aircraft accidents. Safety is the most important aspect of all areas of technology. In aerospace technology/industry in particular, safety is of paramount importance as it directly affects people's lives. However, modern engineering systems are becoming more and more complex. Thus, it becomes more and more difficult to ensure the safety of these complex systems [1-12].

Aerospace engineering is a type of engineering. Let us first briefly recall what these are engineering in general and science; what is the difference between engineers and scientists. Engineers apply principles of natural science in their work. They design and build structures, products, processes, or systems that improve people’s lives and make useful and necessary items. Items, that did not exist before or existed, but have already been improved. The greatest dream, goal and happiness of an engineer are to create something new, such as the Seven Wonders of the World and the International Space Station. And of course, happiness for a scientist is to understand something in the universe that people have never known or understood before. It is clear that in order to become a successful space scientist, you really need to understand the design of satellites, what is possible with them and what is not. And in a similar way, to be successful as a space engineer, you really need to understand the physical principles to understand the environment in which you will work, to understand the materials you will be working with and to develop systems that scientists will then use

to discover the secrets of the universe. Science and engineering are interconnected: a good engineer must understand science, and a good scientist must understand engineering.

Fig. 1. International Space Station [2, 3]

Safety is a system property that is dependent on many elements: hardware, software, human behavior (pilot error), and of coarse interactions between all of these three factors.

Hardware damage

In the past, before the advent of modern computers and software, security and safety were closely related to damage of equipment/hardware and components. These are actually things that can break or not work properly (like an engine failure, valves etc.). The traditional approach to mitigating the consequences of equipment failure is to provide redundancy for components. Indeed, if we had full redundancy in most hardware systems so that we can tolerate failures. It is used a lot. But it has made the system much more complex and several times more massive than the original system.

So there are always trade-offs. Redundancy is just one option. Another option that you can use is to try to remove the component. You might consider redesigning something in the system so that you do not need the often damaged part. Therefore, redundancy sometimes causes its own problems.

Safety and reliability is not the same thing. In the engineering context, however, they refer to two different concepts. Safety usually means that event X will never happen, while reliability usually means that event Y always happens. Understanding this distinction is crucial to our discussion of the various factors that affect system safety. A certain situation can be both safe and reliable, neither safe nor reliable, or any combination of the two. Reliability speaks of failures, especially of components. This may or may not have an impact on safety. Safety is not necessarily about failure. Safety is associated with accidents.

For example case of something that's unreliable but safe: suppose the engine of an airplane does not start on the ground.

When we talk about reliability, we usually talk about hardware components which either work or don't work, whereas safety issues that it is much more difficult to lead to accidents and you are really trying to look at it from a systems point of view. You can take reliability and decompose into properties individual components. All of these components are reliable, and you can put them together and find out the reliability of this assembly. Safety is a system property (an emergent property of the system), not a component property. Safety always depends on the context and on the environment. So if I ask you about any component - is it safe or not? Answer: it depends on the circumstances. What is the context? What kind of environment?

There are accidents that are caused by:

- component failures;
- component interaction accidents.

Component interaction accidents are of two types:

- accidents caused by interactions among components;
- accidents with no component failure.

Software failure

Software plays an irreplaceable role in many engineering systems. However, software can also contribute to system crashes. But what do we mean when we say "software glitch"? Actually, the problem is not in the software itself. In fact, the "curse of software" is that it always does exactly what it is told to do. To understand the role of software in accidents, it is important to view software in the context of how it interacts with other components and with human operators.

Software is absolutely essential for a successful mission. Software is simply a sequence of instructions, a process, a sequence of steps that a computer must follow. Software doesn't crash like hardware. This is a completely different failure. This problem has existed in software from the very beginning. The software design failed. The bottom line is that we'd better tell the software to do the right thing from the beginning and by a systems viewpoint (how all parts of the system interact with one another).

In software, redundancy is a very different issue, from hardware, and can add significant complexity to the system. It is very, very difficult to test, because the software is so complex that it can be tested any possible combination of inputs it can take years or decades. Redundancy for software also adds a lot of complexity. Only to synchronize the computers it takes a lot of time and effort.

Software — and its interactions with hardware and humans — also played a role in aviation accidents. Again, the curse of software shocks: the software did exactly what it was designed to do, but it wasn't designed for all possible operating conditions.

The aviation software design did not accommodate all of the possible system situations that it could be in. And therefore, it prevented the pilot from doing what the pilot really wanted to do. The interaction between software and people is critical.

Human error

Here we look at some of the problems that can be caused by people or their interactions with software. The software is executed in accordance with the project. In this case, the software engineers were not mistaken. The way it usually works is software engineers need to be told what software needs to do, and this is done through software requirements. And requirements are of course a very important part of systems engineering.

One of the first things you should do when you start designing an aerospace system is figuring out what your system is supposed to do. And this is what we call requirements writing. Until you have not a set of requirements, you really don't know what exactly you should design and build. It turns out that the software developers didn't just forget to write this command down in the software sequence. They really met the requirements for them. In other words, the original system requirements, which were then transformed into software requirements, did not really indicate the whole situation in which the system could arise.

Who is responsible for translating system requirements into software requirements? Engineers are people. Everything comes back to people. People talk about human error and talk about drivers or pilots - operators. But there are still people who design, and people make mistakes. If we want to understand why the software was wrong, because software is just design, we really need to understand a thing or two about the people who created it. Software is always design. This is always what the engineers thought about. But we are checking something. Before you launch something into space, be it hardware or software, you test it many times. Many tests have been done, but they weren't perfect. We never know exactly the conditions in space.

We should try to test the systems according to the philosophy of testing aerospace systems:

- Test it like you're going to fly it;
- Fly it like you tested it.

We must not forget the importance of taking a systems view of the entire safety and security situation because we have the interplay of all these different pieces of physical hardware, software and people, and that is really a lot of where your approach to system safety comes from. And it's really hard to get a complete picture of the safety system if you look at only one of the areas, such as software. One really has to look at the whole system.

We often hear that the cause of the accident was "human error". This is especially true in the aerospace industry, where pilot error is cited as a contributing factor to many aircraft accidents. Human error is not a cause, but rather a symptom of the context in which the error occurred. This explanation highlights the difference between the traditional view and the systemic view of the human factor. And if you take a closer look at some of the common mistakes you encounter in your daily life (For example, when entering a store, do you always open the door correctly - push or pull?), consider whether they are due to human error or poor design.

We conclude with a few words of wisdom about designing with people in mind from Donald Norman. This is from a book called “The Design of Everyday Things”:

“Of course, people do make errors. Complex devices will always require some instruction, and someone using them without instruction should expect to make errors and to be confused. But designers should take special pains to make errors as cost-free as possible. Here is my credo about errors:

If an error is possible, someone will make it. The designer must assume that all possible errors will occur and design so as to minimize the chance of the error in the first place, or its effects once it gets made. Errors should be easy to detect, they should have minimal consequences, and, if possible, their effects should be reversible.” [1, 2; (pp. 34, 35)]

The designer always strives for immediate safety engineering. Three principles are applied to achieve safety in Engineering Design [7].

- The principle of safe existence (safe-life behavior) implies that that all components and their relationships within the product will survive the intended stress and operational life without failing or generating a fault.

- The principle of limited failure (fail-safe behavior) implies that a functional fault or damage can occur during the operational life of the product without causing serious damage to the product.

- The principle of redundancy implies that the safety and security of the product is enhanced by including reserve elements that can fulfill some or all of the product’s functions in case of failure. In the case of passive redundancy, the reserve element supports required functionality even when all components are functioning normally. With passive redundancy, the reserve element is only activated in the case of a failure. When the original and reserve elements operate according to differing modi operandii it is known and is the principle redundancy. Back-up elements can be employed in parallel, serial, quartet, cross-quartet, two-out-of-three and comparative redundancy.

If risk cannot be excluded by applying the three principles listed above, complementary indirect and indicative safety equipment is incorporated.

Nanotechnology Safety in the Aerospace Industry

Nanotechnology, the science of materials and devices with at least one dimension in the 1 nm to 100 nm range, can be applied to any field of industry and everyday life. The aerospace industry finds technology that reduces component scale and weight of particular interest. Nanomaterials can also provide corrosion, weathering, and thermal resistance that would allow aerospace products to function in a variety of environments. As the use of nanomaterials increases, questions arise about the safety of nanotechnology, especially in applications where living things are at risk of exposure. In aerospace manufacturing, it means preventing pollution of an object, the people inside it, and the environment around it. Even some materials that are not toxic in bulk or at microscale, such as gold, silver, and platinum, can become lethal at nanoscale. The aerospace industry may be better prepared for the safety of nanotechnology than other industries because it is already adapted to the use of composites, and it is in the composition of composites, coatings and other sensitive devices / sensors that nanomaterials are most likely to be found in this area. In summary, the aerospace industry is expected to implement nanomaterials on a large scale in the future. Because of the prior adoption of composite materials in construction, and given the strict safety requirements of both composites and nanomaterials, aerospace companies may already have the perspective required to use these products safely [8, 9, 10].

Conclusion and future plans

The easiest way to correct mistakes is the undo button in the word processor. This is what we really need to do. It is not necessary to always prevent human error if that is not possible. We can also make it recoverable and easier to fix. This is absolutely important, especially in aerospace systems, because these are systems on which human lives depend, and safety is absolutely essential if we have an air transport system and if we are going to explore space.

Accidents can happen with or without component failure. Software always does what you tell it to do. It is better to design it properly.

Redundancy doesn't always solve problems. Redundancy can cause new problems and it is not always the answer. You must be very careful.

We must take a systematic view of accidents. You really need to understand the whole system, how these components are interacting and interacting with the human. And human behavior is always influenced by both design and context.

Safety and reliability they are not identical.

So once again, the software did exactly what it was designed to do. But the design didn't incorporate all of the human factors that people really needed to use it.

The system safety is a very important field and it can be applied to many areas of human life besides just aerospace. E.g.: nuclear power, health care, nanotechnology etc.

Each country has official websites for the results of aviation accident investigations [3, 4]. When you read about an accident, try to see if you can understand something about what happened.

But there are many other aspects of the safety in aerospace engineering and the effect of material properties on it. Nanotechnology safety in the aerospace Industry will be the subject of our next future study [8, 9].

References:

1. Норман, Д. Дизайн привычных вещей, Litres, 2020.
2. Norman, D. A. The Design of Everyday Things, 1988. Currency Doubleday, New York, 2013.
3. https://www.mtitc.government.bg/bg/category/193/okonchatelni-dokladi-ot-priklyuchili-razsledvaniya-na-aviacionni-subitiya-prez-2020-godina; Доклади от авиационно разследване в България.
4. НАРЕДБА № 13 от 27.01.1999 г. за разследване на авиационни произшествия.
5. Alderliesten, R. (2018). Introduction to Aerospace Structures and Materials. Delft University of Technology; DOI: https://doi.org/10.5074/t.2018.003; ISBN 978-94-6366-075-4.
6. Stapelberg, R. F. (2009). Handbook of reliability, availability, maintainability and safety in engineering design. Springer Science & Business Media.
7. Grote, K.-H., H. Hefazi (Eds.). Springer Handbook of. Mechanical Engineering; ISBN 13: 9783030470340; Springer International Publishing; 2021.
8. Miteva, A..On the microstructure and mechanical properties of nanocomposites, Proceedings SES 2012, ISSN 1313–3888, Sofia, Bulgaria; 220–225, 2013.
9. Asmatulu, R. (Ed.). Nanotechnology safety, Newnes, 2013.
10. Miteva, A.. Nanotechnology in military applications, Proceedings SES 2020, Sofia; p-ISSN 2603–3313; e-ISSN 2603–3321; 362–366, 2020.
11. https://courses.edx.org/courses/course-...
12. https://youtu.be/8jjVmWHWLP0.