MODELS AND SIMULATIONS: RISKS AND LESSONS LEARNED

Terry L. Hardy

Great Circle Analytics, LLC, 899 Pearl Street #21, Denver, Colorado, 80203 USA, thardy@gcirc.com

ABSTRACT

Organizations throughout the world are developing and operating space launch vehicles and systems for the purposes of furthering exploration, delivering services, and facilitating commercial human spaceflight. The operation of the launch vehicles and space systems creates the potential for harm to the crew, to flight participants, and to the uninvolved public. Therefore, it is imperative that comprehensive risk assessments be performed to characterize, evaluate, and reduce the risks of these endeavors. Analytical models and simulations are used in complex space systems to support decision making during development and operations. However, the risks associated with the use of models and simulations are often underestimated, and the hazards are often misunderstood. The failure to understand and address model and simulation risks can lead to poor decisions that may result in mishaps. This paper provides real-world examples and lessons learned to illustrate common concerns with the use of models and simulations.

1. INTRODUCTION

Analyses, models, and simulations play important roles in identifying and controlling space system hazards and reducing space system risk. Analysis is typically defined as technical or mathematical evaluation using mathematical models, simulations, and algorithms. A model is a physical, mathematical or otherwise logical representation of a system, entity, phenomenon, or process. Data that goes into a model is considered part of the model. A simulation is a method for implementing that model, and is typically considered to be an imitation of the characteristics of a system, entity, phenomena, or process using a computational model.

Critical risk decisions are often made on the basis of the results from models and simulations. Analyses and models may be used in the design of space systems, for example, to determine propellant requirements or to calculate launch vehicle trajectories. Models and simulations may be used to verify that safety requirements have been met, for example, to determine structural design margins or to calculate expected thermal loads. Simulations can be used to identify whether systems meet requirements and to allow operators to interact with the system prior to operation. Examples include the use of simulation tools to imitate cockpit conditions prior to flight or to analyze guidance, navigation and control system performance during reentry of a crew capsule. Models may be qualitative, such as system safety risk assessments, or quantitative, such as expected casualty analyses.

All analyses, models, and simulations contain assumptions and uncertainties which impact their usefulness. However, the analysis assumptions are not always understood, and the models and simulations may not be applied appropriately. While the use of any model or simulation requires judgment, safety assessments often do not consider the impact these analyses can have on the risk. Using accident reports from various industries, this paper describes safety risks applicable to analyses that support space system decision making, and provides lessons learned from those incidents.

2. LESSONS LEARNED

This section discusses a number of lessons learned related to analyses, models, and simulations, with corresponding accident examples to show where a flaw in the analysis process led to an undesirable outcome. These accidents are described in reports and investigative summaries from multiple industries and organizations, including those outside of the aerospace industry. This is done to broadly illustrate hazards and risks in models and simulations and to stress the importance of learning from other industries. Note that in discussing these accidents, this paper does not intend to oversimplify the events and conditions that led to the accidents or blame any individuals or organizations. There is rarely a single identifiable cause leading to an accident. Accidents are usually the result of complex factors that include hardware, software, human interactions, and procedures. Readers are encouraged to review the full accident and mishap investigation reports to understand the often complex conditions and chain of events that led to each accident discussed here.

2.1. Failure to incorporate the appropriate models in hazard identification and risk decision making

On January 31, 2006, an explosion occurred at a chemical manufacturing facility located in Morganton, North Carolina, in the United States. This company manufactured paint additives and polymer coatings, and conducted its operations in a large 1,500 gallon reactor. One worker was killed by the explosion, and 14 others were injured in the aftermath. The U.S. Chemical Safety

and Hazard Investigation Board (CSB) determined that the cause of the accident was a runaway reaction. To meet a sudden increase in demand, plant managers had scaled up the normal process by adding more constituents. Unfortunately, the managers failed to understand that scaling up the process resulted in an increase in energy release, leading to tank heating above what the cooling system could handle. The pressure inside the reactor increased due to the increased heating, leading to the venting of solvents inside the building. The vented solvents ignited, leading to the explosion. The CSB faulted the company for its lack of recognition of the hazards from scaling the process and its lack of safeguards to protect against a runaway reaction. The CSB stated that the company had not identified hazards in its operations and had not conducted formal hazard analyses. The CSB noted that safeguards were primarily procedural, but the company could have used high pressure alarms, automatic shut offs, and venting to mitigate the risk. CSB also stated that the company "had minimal safety information on its polymerization process, even though this was the core of its manufacturing business." Although analytical techniques were available, the company did not use analytical models to characterize the reaction process and the thermal aspects of that process, and the plant manager had relied on past experience to estimate batch sizes. This accident shows that, while past experience is important, that experience should be supplemented with data and analysis, especially when making changes to a system [1].

2.2. Failure to provide adequate training in the limitations of models

On August 6, 2007, the Crandall Canyon Mine in Emery County, Utah collapsed, trapping six workers. On August 16, 2007, the mine collapsed again when one of the walls of a tunnel exploded, killing three rescue workers. The original six workers trapped in the explosion were never recovered. According to U.S. Mine Safety and Health Administration (MSHA) investigators, the original collapse was caused by a flawed mine design. The investigation report stated that the stress level exceeded the strength of the pillars such that when one small failure occurred it created a ripple effect that caused widespread collapse, leading to the loss of the miners. The MSHA stated that the mine was "destined to fail" because the company failed to heed early warnings and previous failures. For example, on March 10, 2007, one of the pillars burst leading to a partial collapse of the mine. According to the MSHA the mine's design was based on improper analysis and models. The report stated that the operator's mine design incorporated flawed design recommendations from its contractor. The investigation team discovered that managers and mine safety personnel did not review input and output files for accuracy and completeness and were not appropriately trained in the details and limitations of the models. Therefore, evaluators could not provide adequate assessments of the risk. This accident illustrates that even valid models and simulations can be misused if those using or reviewing the models are not trained to understand the model's limitations [2].

2.3. Failure to document model assumptions and limitations

The Space Shuttle Endeavor was launched on September 7, 1995, on mission STS-69. One goal of the mission was to deploy and then retrieve the Shuttle Pointed Autonomous Tool for Astronomy 201 (SPARTAN-201). SPARTAN-201 was a spacecraft designed to provide short-term scientific observations related to solar winds and the solar atmosphere. During one of the first on-board targeted burns in the rendezvous sequence, ground crews noted that the Shuttle had used 4.3 times as much propellant as predicted. This propellant usage may have threatened the ability to retrieve the spacecraft. However, all burns after this maneuver were ultimately completed successfully and the spacecraft was successfully retrieved. Analyses after the mission found a performance limitation in a rendezvous software algorithm that led to the excess propellant usage. Apparently, this algorithm had been used on Apollo missions in the 1960s and adopted for use on the Space Shuttle. However, the limitations in the algorithm were not passed down to personnel on the Space Shuttle program, and had not been encountered on any previous missions. After the mission, the algorithm functionality and performance were documented and incorporated into flight rules, training, and procedures. This incident stresses the importance of documenting all model assumptions and limitations [3].

2.4. Analysis substituted for testing to reduce costs

The Mars Polar Lander (MPL) spacecraft was launched on a mission to the planet Mars on January 3, 1999. Upon arrival at Mars, communications ended according to plan as the vehicle prepared to enter the Martian atmosphere. Communications were scheduled to resume after the Lander and the probes were on the surface. However, repeated efforts to contact the vehicle failed, and eventually the program managers declared that the spacecraft was lost. The cause of the MPL loss was never fully identified, but the most likely scenario was that a failure occurred upon deployment of the three landing legs during the landing sequence. Each leg was fitted with a Hall Effect magnetic sensor that was designed to generate a voltage when the leg contacted the surface of Mars. The flight software issued a command to shut down the descent engines when touchdown was detected by this sensor. The MPL

investigators believed that when the landing legs deployed, the spacecraft software interpreted spurious signals from the motion of the vehicle as valid touchdown events. The software, upon receiving these signals, then prematurely shutdown the engines at approximately 40 meters above the surface of Mars, and the spacecraft crashed onto the surface and was destroyed. Although the MPL failure report noted that the verification and validation program was well planned and executed, the report also stated analysis was often substituted for testing to reduce costs. Such analysis may have lacked adequate fidelity to identify this system failure scenario. Also, the touchdown sensing software was not tested with the Lander in the flight configuration. The MPL investigators specifically recommended that system software testing in the future include stress testing and fault injection in a suitable simulation environment to determine the limits of capability and search for hidden flaws. As shown here, analyses are important verification tools, but the risks of using them in place of testing must be explicitly stated and understood [4].

2.5. Improper model inputs

On November 12, 2008, a 2 million gallon liquid fertilizer tank at a company in Chesapeake, Virginia, United States, collapsed. Two workers performing welding operations at the site were seriously injured and an adjacent neighborhood was partially flooded as a result of the accident. The CSB found that the company had not assured that welds met accepted industry standards, and the CSB faulted the company for its failure to perform inspections of the welds. CSB also stated that proper procedures were not in place for filling the tanks following major facility modifications. In its report, the CSB also noted that the contractor hired by the company to calculate the maximum fill height had used some faulty assumptions in its analyses. The maximum liquid level was supposed to be calculated based on the minimum measured shell thicknesses and the extent of the weld inspection (full, spot, or no radiography). The contractor used the maximum (not minimum) measured thickness, and improperly assumed full inspection of the welds. The model inputs based on those assumptions led to an overestimation of the allowable liquid level. The tank failed at a fill level of 26.74 feet, below the calculated maximum of 27.01 feet. As shown in this accident, model inputs and data are just as critical as its algorithms to obtaining valid output [5].

2.6. Reliance on overly simplistic models

On October 29, 2006, the barge *OTM 3072* capsized off the coast of Bas-Caraquet, New Brunswick, Canada. The barge was in tow during strong, gale-force winds. The barge sunk after capsizing and was a complete loss, but no one was injured in the accident. The Transportation Safety Board of Canada (TSB) investigated this accident and found that the barge was overloaded and beyond its range of stability. The situation was made worse by the weather conditions. The TSB noted that the operators did not have the proper information or analysis tools to assess the stability of the barge. Preliminary calculations were done, but these calculations were simplistic and a comprehensive study was not performed to determine stability. The TSB made particular mention of the safety management practices of the barge owner and those of the managing owner. The accident report noted that the company's safety management policies, procedures, and practices were limited, which led to a lack of understanding of the risks and a failure to perform more detailed analyses. This accident shows that simple models may help in initial stages of development but may not be of sufficient fidelity to make critical safety decisions [6].

2.7. Failure to use conservative models and inputs

On October 4, 1992, the cargo plane El Al flight 1862 crashed into a neighborhood in Amsterdam, Netherlands. All four crew members and 39 people on the ground died, and many more were injured on the ground. The airplane was designed with fuse pins holding the engine to the wing. These fuse pins were designed to fracture cleanly in the event of a severe engine failure and excessive loads on the engine. The engine would then fall away cleanly and not damage the wing or the fuel tank, allowing the plane to continue flight. The Netherlands Aviation Safety Board found, however, that these pins did not fail properly. It was likely according to the accident investigation board that the pin suffered from gradual fatigue failure. The gradual failure led to engine No. 3 breaking free, knocking engine No. 4 out with it and severely damaging the wing and control surfaces. The pilot then could not keep the plane level or maintain stable flight. The board found that the design of the system to hold the engine to the wing was "inadequate to provide the required level of safety." In addition, the board faulted the inspection procedures at El Al. The report also faulted the certification process. The report stated that the aircraft certification process included a fail-safe analysis of the nacelle and pylon concept. This analysis however did not include the scenarios of fatigue failure or partial failure of a single structural element. Therefore, according to the board, the models and analyses were not conservative. An analysis was also done to establish maintenance requirements. However, this analysis did not provide a sufficient maintenance schedule under actual operating conditions, again making unrealistic assumptions. This accident shows the importance of using conservative models and inputs

such that the results provide for sufficient safety margins [7].

2.8. Using models outside their valid range

On July 5, 2004, a new barge was carrying out its third trip with a load when it folded in half and sunk in the Middensluis lock of Ijmuiden, the Netherlands. All crew members were able to escape without injury, but the ship was lost and the lock was closed to remove the debris. The investigation by the Dutch Safety Board found that the sagging of the vessel was due to improper design of the barge. Structural design tools were used to determine aspects such as bending moment and permissible tensions in components. While the design tools used were common to the industry, this design was different from conventional barge designs. This barge was 40 percent longer than most designs (110 meters long) with a similar width to other barges, so the aspect ratio was unique. In addition, the hold area concentrated the load in the middle of the vessel unlike other designs. Existing tables were used to provide inputs to the design calculations. Because of the unique design, some of the inputs had to be extrapolated from those tables, increasing the uncertainty in the prediction. Also, the design formulas contained implicit assumptions regarding the hold length, and the dimensions of this vessel exceeded the range where the formulas were valid. As a result of the extrapolated inputs and the use of formulas outside their valid range, the calculations produced results with a bending moment that was too low. This meant that the vessel design was too weak and it could fold when carrying a load. Independent calculations had been performed prior to construction, but those turned out to be insufficient as well, relying on many of the same assumptions as the original design calculations. In the words of the safety board, "While extrapolation of the tables used is not uncommon, in this case, however, insufficient account was taken of the validity of the formulas used. The designer's assumption with regard to the calculation for the vessel construction was incorrect. Therefore the vessel was not designed for the stresses to which it could be exposed." The report stated that none of the parties involved understood the limitations on the formulas. All models have applicable ranges of use, and these ranges must be understood to assure that the model is properly employed [8].

2.9. Failure to properly validate models

On June 2, 2001, NASA launched the X-43A "Hyper X" vehicle. The X-43A was a subscale prototype NASA had developed to obtain information on a supersonic combustion (scramjet) engine. The prototype vehicle was a hybrid rocket consisting of the Hyper-X research vehicle and a modified Pegasus launch vehicle. During the mission, the entire hybrid rocket was released nominally from a B-52 carrier aircraft. The solid rocket motor ignition occurred approximately 5 seconds after release, and shortly thereafter the vehicle experienced a control anomaly. After several seconds the vehicle began to break up, and the vehicle was destroyed by a range safety command when it started to veer off course. The mishap investigation board found that the root cause was that the vehicle control system design was deficient for the trajectory flown due to inaccurate analytical models. These models overestimated the system margins. The board found that modeling inaccuracies existed in the fin actuation system and in aerodynamics. The fin actuation system inaccuracies resulted from discrepancies in modeling the electronic and mechanical fin actuator system components and underprediction of the fin actuation system compliance. Aerodynamic modeling inaccuracies resulted from errors in incorporation of wind tunnel data into the model, misinterpretation of the wind tunnel results due to insufficient data, and unmodeled changes associated with the thermal protection system. The modeling did not include sufficient uncertainty analysis of the modeling parameters, according to the report. The investigation report also discussed insufficient wind tunnel testing to validate the model; the testing did not take into account changes made in the thermal protection system. This example shows the importance of assuring that models are properly validated using tests that simulate actual operation [9].

2.10. Failure to compare model results to real-world experience

On January 18, 1978, the Hartford, Connecticut area in the United States experienced a large snowstorm, and the roof of the Hartford Civic Center Coliseum was covered with heavy snow from the storm. At approximately 4:15 AM the roof of the arena collapsed. Five thousand basketball fans had been in the arena just a few hours before, but had all left for the evening. Had the collapse occurred earlier hundreds may have died or been injured. An analysis after the accident showed the cause of the collapse to be inadequate bracing in the exterior rods supporting the roof and an underestimation of the loads. The engineers had relied on an oversimplified computer analysis to assess the loads. During construction workmen had noticed a sag in the roof, calling into question the model used. The designers apparently believed their analysis instead of the physical evidence which was showing potential failure, and they disregarded the information from the workmen. While the empirical evidence was not formal, this information should have alerted the engineers to a potential problem with the models. As this mishap shows, analysts should use all available information in validating their models and simulations; sometimes the best information is operator experience in the field [10].

2.11. Failure to independently check model or results

On August 5, 2008, a Sikorsky helicopter operated by the U.S. Forest Service crashed into trees and terrain while transporting firefighters near Weaverville, California, United State. Nine people were killed in the crash and four others were seriously injured. The NTSB report stated that the pilots had significantly overestimated the helicopter's load-carrying capacity and therefore did not have an adequate performance margin for a successful takeoff, leading to the crash. The NTSB stated that the overestimation of the performance capability was the result of improper inputs to the load capacity model. The manufacturer had provided an incorrect empty weight to the pilot-in-command, resulting in his overestimation of the helicopter's load carrying capacity. In addition, the helicopter's available power chart, also provided by the manufacturer, overestimated the emergency reserve power available, reducing the aircraft's load safety margin. Plus, the pilot-in-command followed an unapproved calculation procedure (also provided by the manufacturer) that used an above-minimum specification torque. This procedure increased the error in the load capacity estimations. These errors taken together led the pilots to believe that could carry a heavier load than they actually could. The report also faulted the oversight provided by the U.S. Forest Service. The report stated, "effective oversight would likely have identified that Carson Helicopters was using improper weight and performance charts for contract bidding and actual load calculations and required these contractual breaches to be corrected." As this accident shows, good models are not enough. Care must be taken with model inputs, and results and model use must be independently checked [11].

2.12. Failure to update analyses after design or operational changes

On October 2, 2005, the ship *Ethan Allen*, carrying 47 passengers and one operator, capsized while on a cruise of Lake George, New York. Twenty passengers died in the accident. The NTSB determined that the probable cause of the capsizing was that the *Ethan Allen* was unstable in the rough waters that day. When the ship had made a sharp turn, the waves and the involuntary shifting of the passengers due to the boat's motion led to the overturning of the ship. The ship was unstable, according to the NTSB, because it carried more people than it should have. The NTSB noted that the ship's stability had not been reassessed after it had been modified. The *Ethan Allen* had been modified to include an all-wood canopy with Plexiglas windows. This modification reduced the calculated stability limit from 59 people without the canopy to 14 with the canopy, according to calculations performed after the accident. Therefore, the vessel was carrying 34 people more than it should have, according to the NTSB. The NTSB stated that there was no record of detailed calculations showing stability margins nor was there any record of a stability test to verify the maximum number of people allowed on board the *Ethan Allen*. The NTSB stated that assessments and simplified stability tests should have been performed after installation of the canopy. In this case, the failure to perform additional tests or analyses following a design change masked significant risks [12].

2.13. Models not implemented correctly in software and computing systems

On February 25, 1991, a Patriot missile defense system operating in Saudi Arabia during Operation Desert Storm failed to track and intercept an incoming enemy Scud missile. This missile hit an American Army barracks, killing 28 Americans. The defense system failed to track and intercept the missile because of a software problem in the system's weapon control computer. Tracking an incoming missile required knowledge of both time and velocity. Time was kept continuously by the system's internal clock in tenths of seconds but was expressed as an integer. Because of limits on the operating system, inaccuracies were introduced when converting an integer to a real number. These inaccuracies increased over time. The Patriot system had been operating for 100 hours straight prior to the failure, and large inaccuracies had developed in the targeting system over that time. Therefore, while the tracking algorithm was correct, the implementation of the model on the computing system led to inaccuracies that resulted in the system failure. As shown here, it is important to not only consider the validity of the model but also how the model will actually work once it is implemented in software [13].

2.14. Biased application of model results

On February 1, 2003, the Space Shuttle Columbia disintegrated over Texas during re-entry, resulting in the loss of all seven crew members. The vehicle was lost as a result of damage sustained during launch when a piece of foam insulation from the external tank broke off and struck the leading edge of the wing, damaging the thermal protection system tiles. The Crater model was used to predict tile damage. This model predicted significant penetration of the tiles based on available flight information. "This seemingly alarming result suggested that the debris that struck Columbia would have exposed the Orbiter's underlying aluminum airframe to extreme temperatures, resulting in a possible burn-through during re-entry," according to the Columbia Accident Investigation Board (CAIB). However, NASA engineers, aware of the uncertainties in the model, believed that the model overestimated damage and therefore may have downplayed the potential risks, according to the report. The CAIB report

stated that the Debris Assessment Team "used a qualitative extrapolation of the test data and engineering judgment to conclude that a foam impact angle up to 21 degrees would not penetrate the RCC [reinforced carbon-carbon]." As stated in the report, "Engineers who attended this [Debris Assessment Team] briefing indicated a belief that management focused on the answer – that analysis proved there was no safety-of-flight issue – rather than concerns about the large uncertainties that may have undermined the analysis that provided that answer." [14]

3. RECOMMENDATIONS

The following are some suggestions to improve analyses, models, and simulations to decrease space system safety risk [15, 16].

Prepare plans for the use of models and simulations. Like any other engineering endeavor, model and simulation use must be planned to be effective. The plans should include all development phases, including acquisition, development, operation, maintenance, and retirement. Plans should define the objectives and requirements for models and simulations, including the acceptance criteria for modeling and simulation products, and the intended use of the models. The plans should include appropriate metrics for determining model validity. Unique computational requirements (e.g., memory, disk capacities, processor, and compilation options) should be defined.

Explicitly state and document assumptions and limitations. Although assumptions are necessary to conduct any analysis, it is important for all members of an organization to understand, and agree to, the assumptions, uncertainties, and limitations of that analysis. Those assumptions should be challenged and tested to assure that they are actually valid and conservative. Of particular concern is when a model is used for designs and operations that lie outside the limitations of that model. Models are usually developed based on data obtained under limiting conditions. If the model is used outside those limitations it may no longer be valid. The limitations on the models must therefore be explicitly stated to understand the risk of extrapolation of the model beyond its intended use.

Define limits on the use of analysis for verification. As part of the planning process, organizations should establish criteria for the proper use of analysis in verification. Verification of complex systems can be expensive and time consuming. Organizations will often make trade-offs between verification by testing and verification by analysis. While these trade-offs are legitimate, analysis should not be used to eliminate testing without a risk assessment. Models and simulations require resources to produce, maintain, and use, and they may not be as cost-effective as organizations believe. In addition, the costs of analysis have to be traded against the cost of redesigning, building, and testing a system that fails to perform as expected due to insufficient modeling and simulation efforts.

Provide documentation and training on the model and simulation usage. It is not enough to develop and certify the model. Organizations must assure that the users and evaluators of those models and simulations are properly trained, and that user documentation exists and is readily available. Documentation should include the basic structure and mathematics of the models, and should define the inputs, limitations, and supporting data for the use of the model. A feedback mechanism should be defined to allow users to report unusual results to model developers or maintenance personnel. Documentation should include guidance to prevent usage of the model and simulation beyond its limits.

Assure that a process exists for review of input data. The model itself should be certified, but the data input to the model must also undergo review. Many incidents have resulted from improper inputs to correct models. Model software should capture and report improper user input where possible, and processes should be implemented to assure the validity of the input and output data. That process should include criteria for proper use of the output. For example, if the model does not give the expected results, a formal mechanism should exist for reporting and analyzing the results to prevent a user from improperly adjusting the inputs to get the desired answer.

Define processes to certify models and simulations for use. Models and simulations should correlate with data from other programs and known standards. If a model or simulation has been previously certified, then it is important to understand the limitations of that previous certification. At a minimum, certification processes should include the following factors:

- *Capability:* what the model or simulation can do in terms of functional representation, behavior, relationships, and interactions - the model should represent all phases and conditions of operation
- *Correctness:* the quality of the code and the appropriateness of the input data
- *Accuracy:* how closely the model or simulation results correspond to actual, measured, observed or demonstrated behavior and interactions of the item being modeled or simulated
- *Usability:* the existence and sufficiency of user-support features (e.g., user manuals, training) that enable the user to properly execute the model or simulation and analyze or employ the results

Assure that model development follows approved standards and practices. Models and simulations are specialized types of software. Therefore, development of models and simulations should follow approved development practices. In other words, model and simulation efforts should follow a standard software development life cycle including planning, requirements development, requirements analysis, design, coding, unit testing, system testing, and acceptance testing. Configuration management and quality assurance should also be part of that development process. And, as stated earlier, models and simulations must be verified and validated through an approved certification process.

4. SUMMARY

Analyses, models, and simulations are important tools in the development of space systems. These tools are necessary for designing space systems, identifying potential problems, providing assessments of risk, and verifying requirements and risk reduction approaches. However, all analyses contain uncertainties and are based on assumptions, and the risk of using those tools may not be fully understood. Failure to account for such factors can lead to an underestimation of risk, and may lead to accidents, as illustrated by the examples presented here. To improve the safety of complex space systems it is imperative that models be validated, that analyses be independently verified, and that assumptions, limits, and uncertainties be explicitly stated, challenged, and tested. Above all, the use of models and simulations requires humility - analysts must recognize that these tools are mere representations of reality and will likely not completely reflect the actual system.

5. REFERENCES

1. U.S. Chemical Safety and Hazard Investigation Board (2007). "Runaway Chemical Reaction and Vapor Cloud Explosion, Synthron, LLC, January 31, 2006," Report No. 2006-04-I-NC.

2. U.S. Mine Safety and Health Administration (2008). "Report of Investigation: Underground Coal Mine Fatal Underground Coal Burst Accidents, August 6 and 16, 2007, Crandall Canyon Mine, Genwal Resources Inc, Huntington, Emery County, Utah," ID No. 42-01715.

3. Goodman, J. (2007). "Lessons Learned from Seven Space Shuttle Missions," NASA/CR-2007-213-697.

4. Leveson, N.G. (2004). "The Role of Software in Recent Aerospace Accidents." *Proceedings of the 19th International System Safety Conference*.

5. U.S. Chemical Safety and Hazard Investigation Board (2009). "Allied Terminals, Inc. – Catastrophic Tank Collapse, Allied Terminals, Inc., Chesapeake, Virginia, November 12, 2008," Report 2009-03-I-VA.

6. Transportation Safety Board of Canada (2007). "Capsizing, Barge *OTM 3072* off Bas-Caraquet, New Brunswick, 29 October 2006," Report No. M06M0110.

7. Netherlands Aviation Safety Board (1994). "Aircraft Accident Report: El Al Flight 1862, Boeing 747-258F-4X-AXG, Bijlmermeer, Amsterdam, October 4, 1992," Report 92-11.

8. Dutch Safety Board (2006). "Sagging of a Barge: Sagging and partial sinking of a barge on 5 July 2004 in the Middensluis lock of Ijmuiden."

9. National Aeronautics and Space Administration (2003). *Report of Findings, X-43A Mishap, by the X-43A Mishap Investigation Board, Volume I.*

10. Matins, R., and Delatte, N.J. (2001). "Another Look at the Hartford Civic Center Coliseum Collapse," *Journal of Performance of Constructed Facilities*, 15(1) 31-36.

11. U.S. National Transportation Safety Board (2010). "Crash During Takeoff of Carson Helicopters, Inc., Firefighting Helicopter Under Contract to the U.S. Forest Service, Sikorsky S-61N, N612AZ, Near Weaverville, California, August 5, 2008," NTSB/AAR-10/06.

12. U.S. National Transportation Safety Board (2006). "Capsizing of New York State-Certificated Vessel *Ethan Allen*, Lake George, New York, October 2, 2005," NTSB/MAR-06/03.

13. U.S. General Accounting Office (1992). "Patriot Missile Defense Software Problem Led to System Failure at Dhahran, Saudi Arabia," GAO/IMTEC-92-26.

14. Government Printing Office (2003). *Columbia Accident Investigation Board, Report Volume 1*.

15. National Aeronautics and Space Administration (2008). "Standard for Modeling and Simulation," NASA-STD-7009.

16. Hardy, T.L. (2012). *Software and System Safety: Accidents, Incidents, and Lessons Learned*, AuthorHouse.