AI Challenges in Emergency Control Design

Conclusions from well-documented transportation accidents

Avi Harel, Ergolight

ergolight@gmail.com

Abstract

Automation enables facilitating human activity, but humans are still in the loop to cope with the unexpected. When under stress, the human operator might not behave rationally. In emergencies, the system needs to protect itself and society, and to recover from risky situations. The protection and recovery rely on the collaboration between the human operators and automation. AI enables assigning more human tasks to automation. Still, in emergencies, human operators are required to navigate the way to recovery and the system should protect itself by automation. A primary challenge in the design of emergency control is to draw the boundaries of automation and human control and set rules for collaboration. The article discusses a collaboration model based on the qualifications of the parties in a model of controller-service interaction. Humans are best for resilience, for deciding on the reaction based on prediction of the effect of selected reactions. The helm dilemma is about assigning authority in the collaboration. AI is best for preventing risk detection and for protecting in case of human confusion. Automation may override human control by prediction of the system situation in high-risk situations. Risk indicators may be employed for deciding about the transition from human to automated control.

I. Introduction

The industry's need

The history of accidents is saturated with examples of accidents that could have been prevented, had the industry found the ways to protect from human errors and to apply the investigation findings across different domains. Most industry, military, transportation, and medical accidents are commonly

attributed to decision errors made by the human operators. In many cases, the error was due to operating under stress, in an emergency.

Humans are best in reacting to unexpected situations, but they are likely to err in emergencies. A primary challenge in the design of emergency control is to eliminate the possibility of errors. A way to prevent such errors is by AI. The benefit of integrating AI into real projects was demonstrated by the Auto-GCAS solution to the g-LOC challenge.

Definition of an emergency

An emergency may be defined as an exceptional situation in which the controller is under threat. It begins in a change from a normal to a risky situation. It is terminated by resuming the normal situation.

Formally, an emergency may be defined as an event of exceptional situation, such that:

- Recovery is complicated,
- Failure to recover might end up in an accident, and
- The time-to-accident (TTA) is critically short.

An emergency might end up in an unexpected accident, if the operators are not aware of the situation. If the operators are aware of the situation and of the risks, the situation is stressful. The focus of this article is on stressful operation in emergencies.

The model of emergency control is based on a model of safety-oriented operation, employing concepts of operational risks and hazards.

Interaction control

Interaction control involves strategies, tools, and practices designed to handle unusual, unexpected, or crisis scenarios that disrupt normal operations. This might include emergency situations (e.g., natural disasters, security threats, system failures, or market shocks) that require quick and effective response to mitigate damage, restore functionality, or adapt to new circumstances.

Operational integration

System integration is the process of linking various software applications, databases, hardware, and IT environments within an organization. The goal of integration engineering is to ensure that individual parts work together seamlessly to achieve a desired outcome. In the context of system operation, the individual parts are the human operator and the controlled machine. The focus is on validating that the operation is well integrated. In operational integration we focus on the operation, and the goal is to enforce the system to behave as expected.

Interaction errors

In normal operation, all the services are coordinated with the controller. Eventually, the coordination may be expressed in terms of compliance with the scenario that the controller assumes. Apparently, in many accidents, the scenario was defined only implicitly, which means that the system did not have the means to coordinate the services with the controller.

Compromising human errors

Eliminating decision errors is crucial for assuring the system usability, which is essential for assuring safety, productivity, and consumer satisfaction. Prior studies discuss ways to prevent errors by design (e.g. Harel, 2024 A,B,C). These studies proposed a model of decision errors, and a framework for eliminating these errors by design. The model was developed by analysis of published accident investigations across safety-critical domains. It applies to all utility-critical domains. The focus of this article is on emergencies.

Challenges of emergency control

This study explores the role of automation in emergency control by analysis of transportation accidents attributed to operational errors. It demonstrates the potential benefit of integrating AI processes in the design of emergency control. It is based on several Loss Of Control (LOC) case studies attributed to situational confusion. The study addresses the following engineering challenges:

1. A model of emergency control

2. A framework for Integrating AI for effective control
3. Generic rules enabling affordable implementation

Emergency control

The most effective mode of emergency control is by supervision of human operators, while employing automation to support decision-making. Automation should only override human decisions in emergency situations where immediate action is essential to prevent an accident. Only when immediate action is required to avert an accident should the automation override the human decision.

Emergency control involves managing and guiding systems through emergency conditions to maintain stability, safety, and functionality.

Interaction in emergencies may rely on direct mappings from intentions to actions. A way to support the operator's decision making is by providing situational preview to the operators, as well as situational preview of the effect of optional decisions.

The dilemma of emergency control

The dilemma of emergency control revolves around how much autonomy should be given to the automation versus how much control should remain with the human operator, when in emergency operation. The dilemma refers to the tension between the operator and the automation. Sometimes their actions might conflict. The challenge of emergency control is to support operation in stressful conditions.

Automation promises to relieve humans of mundane tasks, but it often turns them into passive monitors, required only in case of failure but inadequately prepared to respond when such situations arise.

Bainbridge (1982) examined human performance in case of automation failure. She found that in such situations people are likely to behave in ways that are optimized for normal conditions, instead of ways optimized for emergencies.

Automation often makes underlying processes invisible to operators, obscuring how outputs are generated. This opacity can make it difficult for

operators to diagnose and troubleshoot issues effectively, as they may not fully understand what the automation is doing at any given moment.

As automated systems handle more of the workload, human operators may become deskilled or less familiar with the nuances of the tasks that were automated. This deskilling becomes a problem when the automation fails, as the human operator might lack the knowledge or skills to step in quickly and accurately.

Ironically, automation systems often require human operators to intervene in precisely the situations where the system fails, but then they do not have the skills required to replace the automation when it fails. However, the case studies here demonstrate that human operators cannot handle unexpected situations if the automation does not provide the information required for decision making.

Automated systems may encourage operators to take risks, assuming the technology will handle any issues. This overconfidence can lead to risky behavior, further compounding problems in emergencies.

Human-machine teaming

In emergency control, we deal with unexpected situations. Currently, automation cannot handle the unexpected. Apparently, this limitation is the single most important reason why we introduce human controllers in the system in the first place. Ironically, however, humans cannot handle unexpected situations under stress: in emergency situations, the human operators react according to their training, which is suites normal operation. The solution to this problem is by human machine collaboration in emergency control. The design challenge is to define the way they should collaborate. Can the machine support human needs to enable effective control?

Traditional machine support is sufficient for some of the tasks; yet other key control tasks require incorporating reasoning in the machine.

Autonomous operation

Self-driving vehicles represent a push toward automation, but human control remains crucial in unpredictable driving conditions. Both the human controller and the automation might make decisions in parallel.

Autonomous systems are designed to operate without human intervention. They are self-operating systems that make decisions and take actions on their own. In a fully autonomous system, the controller is automated, often based on AI. The automated controller is responsible for all aspects of control, decision-making, and adapting to its environment. For example, an autonomous drone would fly itself, make navigation decisions, and avoid obstacles without human control.

Bainbridge's work is most relevant, with the rise of AI and increasingly autonomous systems. It serves as a cautionary reminder that while automation can greatly improve efficiency, it must be designed with an understanding of human psychology and the potential ironies that can arise when human operators and automated systems interact.

The dilemma of control allocation

According to the traditional approach, each of the tasks should be allocated to either the human or the machine, by matching tasks to the strengths of humans and machines.

The challenge is of dynamic allocation of control features to the human operator and to the automation. In a simple model, we focus on a system with a single controller and a single service. The task of emergency control allocation is a special case of dynamic control allocation (DCA), which applies to systems with multiple actuators (such as aircraft, marine vehicles, or robotic systems), in which there are more actuators than control variables (McRuer & Miele, 2009). Accordingly, the dilemma of emergency control is a special case of the DCA challenge.

In emergency situations, we need to choose between manual control or automation. The solution depends on the operational complexity and on the level of emergency.

The complexity factor

The challenge of over-actuation is when the system has more actuators than the degrees of freedom that need to be controlled, providing extra flexibility in how the operation may achieve its objectives. Redundancy allows the system to operate in a more fault-tolerant or efficient manner by choosing the best combination of actuator efforts.

II. Case studies

Traditionally, stakeholders demand investigation of costly events and are not concerned about low-cost events. Therefore, the evidence of the sources of accidents relies on a small part of the exceptions, mostly, those celebrated exceptions resulting in costly accidents. Therefore, the case studies are mostly of well-documented accidents.

The article discusses the operation in emergencies in the following cases:

- The Torrey Canyon supertanker accident in 1967, in which the captain lost control of the vessel
- The Air Peru 603 crash in 1996, due to a maintenance error
- The Asiana Airline 214 accident in 2013, in which the airplane was too short in the exceptionally difficult conditions.
- The AF 296 Loss Of Control (LOC) accident in 1988, in which the autopilot did not obey the pilot command to pull up
- The AF 447 pilot confusion accident in 2009, in which the autopilot was disconnected automatically

These accidents highlight the challenge of emergency control in modern navigation, namely, the control dilemma. The dilemma is about the questions, who should lead the operation in an emergency: the human or the automation. Apparently, the lessons from the individual cases seem to conflict with each other.

Torrey Canyon, 1967

Source: Harel, 2024 D

The Torrey Canyon was a supertanker operated by British Petroleum (BP).

On March 18, 1967, the vessel, transporting 119,000 tons of crude oil, ran aground on the Seven Stones reef between the Isles of Scilly and Land's End in Cornwall, England.

Sequence of Events

March 17, Early Morning:

The ship approaches the English Channel. Captain Pastrengo Rugiati decides to take a shorter route through the channel, close to the Isles of Scilly, to save time. The area has poor visibility due to fog, and strong winds begin to drift the ship off course.

March 18, Morning:

As the Torrey Canyon approaches the dangerous waters near the Scilly Isles, the captain realizes the ship is closer to the reef than anticipated.

Attempting to correct the course, he discovers that the rudder has become unresponsive due to a disconnect from the steering mechanism, exacerbating the situation.

The crew attempts to override the autopilot and control the ship manually, but confusion over the controls and the rudder's failure prevent successful maneuvers.

March 18, 08:50 a.m.: The Torrey Canyon runs aground on the Seven Stones Reef between the Isles of Scilly and the Cornish coast.

Immediate Aftermath: The grounding causes extensive damage to the hull, and large quantities of crude oil begin to spill into the sea. Efforts to refloat the vessel are unsuccessful.

Investigation

The investigation into the Torrey Canyon disaster of 1967 revealed multiple factors contributing to the grounding and catastrophic oil spill, highlighting both human and technical failures. The inquiry focused on navigation errors, technological limitations, and decision-making under pressure. Key Findings from the Investigation:

1. Navigation and Course Misjudgment: Captain Pastrengo Rugiati decided to take a shorter route through the English Channel, intending to save time by passing near the Scilly Isles instead of taking a safer, wider berth. Investigators found that this route was risky due to the area's narrowness and hazardous reefs. The investigation suggested that this choice of route, combined with poor visibility from fog, contributed significantly to the disaster
2. Wind Drift and Environmental Conditions: Wind conditions caused the ship to drift off course. During the night, strong winds gradually pushed the vessel closer to the dangerous Seven Stones Reef, but the ship's crew did not detect the drift in time. The investigation noted that the ship's instruments and navigational aids were insufficient for accurate course monitoring in such conditions
3. Mechanical Failure - Rudder and Steering Disconnect: One of the most critical findings was a mechanical failure that disconnected the rudder from the ship's wheel, which made steering impossible. This disconnect occurred as the crew attempted to correct the course. The rudder failure prevented the captain and crew from manually adjusting the ship's direction, trapping the vessel on its path toward the reef. This error was traced back to a flaw in the ship's design and an unfamiliar "control" mode that the crew mistakenly selected
4. Human Factors and Decision-Making: The inquiry highlighted the pressures faced by the captain and crew as they attempted to rectify their course. Rugiati's decision to switch from autopilot to manual control in an unfamiliar navigation mode under stress contributed to the crew's inability to recover from the course deviation. Investigators concluded that both design flaws in the ship's systems and human error were significant factors.
5. Regulatory and Safety Failures: The disaster exposed gaps in international maritime regulations for large oil tankers, which led to calls for stricter safety protocols. The inquiry underscored the need for clearer safety standards and better training for handling advanced navigational systems

The disaster prompted the development of new international safety and environmental protocols, including updates to the International Convention for the Prevention of Pollution from Ships (MARPOL). This case also led to industry-

wide changes in how oil tankers are designed, operated, and monitored to prevent similar accidents in the future.

Root-cause analysis (RCA)

The root causes of the Torrey Canyon disaster include a combination of risky navigational decisions, technical flaws in the steering system, adverse environmental conditions, and failure of the human operator to handle the exceptional situation under stress.

Proactive RCA

Proactive RCA focuses on identifying underlying issues that, if addressed, could have prevented the disaster. The disaster's preventability largely hinged on better navigation protocols, design reliability, and crew preparedness. The findings ultimately influenced future maritime policies, pushing for improved tanker safety standards and international environmental protection:

1. The unintentional shift to the “control” mode instead of the manual navigation mode hints on the need to restrict the operational mode according to the operational scenario.
2. The mode confusion in emergency hints to the need to provide clear indication of the operational mode.

In hindsight

1. Risk-oriented Human-Centered Design (HCD) could have prevented the slip of the control lever to the maintenance-only control mode.
2. Scenario-based design could have enabled detecting the exceptional setting.

These findings suggest the need to prevent changing the operational mode to a state that does not comply with the scenario, to notify on exceptional situations, and to alert on diversion to an exceptional situation.

PL 603, 1996

Source: Harel, 2024 E

Bound for Santiago, Chile, the Boeing 757 experienced severe instrument malfunctions due to an adhesive tape left over its static ports, which were accidentally covered during maintenance. These ports provide critical data like airspeed and altitude; with them blocked, the crew received false readings, causing multiple conflicting cockpit alarms and warnings.

Struggling with disoriented flight data and operating over the ocean at night with no visual references, the pilots attempted to return to the airport but ultimately lost control. The aircraft crashed into the sea approximately 48 nautical miles from Lima, tragically killing all 70 occupants on board, including 61 passengers and 9 crew members

Sequence of Events

1. October 1, 1996: Maintenance and Oversight: Before the flight, the Boeing 757 underwent routine cleaning and maintenance at Lima's airport. During this process, adhesive tape was applied over the static ports to protect them from dust. However, this tape was not removed afterward, which would later prove fatal
1. October 2, 1996, 12:42 a.m.: Aeroperú Flight 603 departs from Lima, heading to Santiago, Chile. Shortly after takeoff, the pilots begin experiencing issues with their instruments, receiving inconsistent altitude and speed readings due to the blocked static ports
2. 12:47 a.m.: Confused by the erratic instrument readings, the pilots contact Lima air traffic control and declare an emergency. They request assistance to return to the airport for an emergency landing. The crew continues to receive multiple conflicting warnings, adding to their disorientation
3. 12:55 a.m.: The crew attempts to follow air traffic controllers' radar vectors back to the airport. However, without reliable altitude or speed readings and in complete darkness over the ocean, the pilots are unable to maintain control.
4. 1:11 a.m.: After approximately 29 minutes of struggling with the faulty readings, the aircraft descends toward the water and crashes into the ocean about 48 nautical miles off the coast. All 70 people on board, including 61 passengers and 9 crew members, lost their lives

Investigation

The Peruvian Commission of Accident Investigations, along with the National Transportation Safety Board (NTSB), determined that the primary cause of the accident was the failure to remove the adhesive tape from the static ports. This oversight caused the instrumentation to malfunction, leading to the pilots' inability to accurately gauge altitude and air speed.

Proactive RCA

The accident demonstrates the need for emergency support, by early detection of exceptional situations.

In hindsight,

These findings suggest the need to prevent changing the operational mode to a state that does not comply with the scenario, to notify on exceptional situations, and to alert on diversion to an exceptional situation.

Asiana Airlines 214 - 2013

This crash was attributed to pilot error in managing descent speed and approach path. The pilots attempted to land a Boeing 777 but were below the standard glide path, ultimately colliding with a seawall just before the runway. A delayed decision to increase engine thrust and attempt a go-around left the aircraft in an unrecoverable state close to the ground. The crash led to three fatalities and numerous injuries but underscored critical issues in pilot training, automation dependency, and communication under stressful conditions

Sequence of Events

Departure and Approach:

1. The Asiana Airlines Flight 214, operated by a Boeing 777-200ER, departed Incheon International Airport in Seoul on July 5, 2013, and was scheduled to land in San Francisco the next day.
2. Three pilots were in the cockpit: the pilot flying (a training captain), a check captain overseeing his training, and a first relief officer. This approach was the training captain’s first landing in a 777 at San Francisco.

Final Approach:

1. As the aircraft approached San Francisco, the Instrument Landing System (ILS) for Runway 28L was out of service due to construction. Pilots used a visual approach, relying on sight and cockpit instruments.
2. During the final 1.5 miles of descent, the aircraft's speed dropped below the target landing speed (137 knots). At this stage, the aircraft was flying too low and slow.
3. The check captain called for an increase in speed, but this action was not executed effectively.

Stall Warning:

1. At around 1.5 seconds before impact, the stick shaker (a stall warning device) activated. By this time, the aircraft was at an altitude of approximately 125 feet and approaching the seawall at the runway's end.
2. The main landing gear and tail of the aircraft struck the seawall just short of the runway, causing the tail section to detach. The impact sent the fuselage spinning off the runway.
3. The plane came to rest at the left side of Runway 28L, and a fire started shortly afterward.

Investigation

The National Transportation Safety Board (NTSB) investigation found several factors contributing to the crash:

- Pilot Error: Mismanagement of the approach speed and glide path, as well as overreliance on the automation systems.
- Crew Training: Inadequate training on flying visual approaches and on handling certain automation systems in the airplane.
- Automation Dependency: Overreliance on autopilot and autothrottle, with an apparent lack of understanding of these systems by the flight crew.

Proactive RCA

The Asiana Airlines 214 crash highlighted the need for improved training in manual flying skills and reliance on visual approaches, especially when advanced automation systems are not available.

In hindsight

This finding suggests the need to assume that operators might err, and therefore we should always thrive to detect and alert on exceptional activity.

AF 296 – 1988

Source: Harel, 2024 F

Air France Flight 296 was an Airbus A320-111 that crashed during a demonstration flight on June 26, 1988, at the Habsheim Air Show in France. The flight was intended to showcase the new Airbus A320, which was equipped with advanced computerized fly-by-wire technology.

Sequence of Events

1. The plane was performing a low pass over the airfield at the air show, intending to fly at a low altitude with gear down and at low speed as part of the demonstration.
2. As the aircraft descended to about 30 feet above ground, the pilots attempted to level off and apply full power, but the aircraft failed to climb in time.
3. The plane hit trees at the edge of the runway and crashed into a forest, bursting into flames shortly after impact.

The incident marked the first crash of an Airbus A320, which was a new aircraft at the time.

Investigation

- The investigation focused on several factors, including pilot actions and the performance of the Airbus A320's fly-by-wire system.
- The pilots claimed that the aircraft's automated systems did not respond to their commands to increase thrust.
- However, the investigation determined that the accident is due to pilot error. The automated systems were found to have functioned as designed, though there was much debate about the role they played in the accident.

Root-cause analysis (RCA)

The error: the primary cause was flying too low and slow, as used in manual control

The risk demonstrated: insufficient trust in the pilot's decision leads to the autopilot overriding the pilot's intention

Proactive RCA

This case study demonstrates common mistakes in the design for handling emergencies.

In hindsight

Unless the system is under an immediate hazard, the system should obey the operators' commands.

AF 447 – 2009

Air France Flight 447 was a scheduled passenger flight from Rio de Janeiro, Brazil, to Paris, France, that tragically crashed into the Atlantic Ocean on June 1, 2009, resulting in the deaths of all 228 people on board.

The aircraft involved was an Airbus A330-203, a long-range, wide-body, twin-engine jet airliner. It was a relatively modern and advanced aircraft at the time of the crash.

Sequence of Events

1. Just before the crash, the aircraft encountered turbulence at high altitude.
2. The autopilot disengaged when the pitot tubes froze, and the cockpit became filled with confusing warnings.
3. Despite receiving stall warnings, the pilots applied incorrect inputs, pulling the nose up, which worsened the stall situation.
4. The plane descended for over three minutes before hitting the ocean.

Investigation

1. The initial cause of the accident was attributed to the failure of the aircraft's pitot tubes, which are sensors that measure air speed. These tubes became obstructed by ice crystals, leading to the autopilot disconnecting and unreliable airspeed readings.
2. The crew, facing confusion and conflicting data, responded incorrectly to the situation. The aircraft entered an aerodynamic stall, and despite efforts to recover, the pilots were unable to regain control.
3. The final report by the French aviation authority (BEA) concluded that a combination of technical failure and human error was to blame.

Root-cause analysis (RCA)

The error: The pilots failed to interpret and respond properly to the stall warnings.

The risk demonstrated: pilots are overly reliant on the autopilot, which can lead to skills degradation.

Proactive RCA

This case study demonstrates common mistakes in the support of emergencies.

In hindsight

The design of emergency control should have assumed that the operators might not behave as expected.

III. Cross-event analysis

Control modes: humans vs. automation

In safety-critical applications, automation can be designed with human override functions, ensuring that a human can intervene in case the automation fails or behaves unexpectedly. This principle is applicable to situations such as in the AF 296 accident.

Autonomous systems need ongoing monitoring, validation, and regular updates to ensure they operate safely and stay aligned with human values. A mixed system where humans and automation share control often provides the best balance, leveraging automation's strengths in processing and precision while keeping humans in the loop for critical decision-making.

The control dilemma

The dilemma is about who should lead the operation in an emergency: the human or the automation. Apparently, the lessons from the case studies seem to conflict with each other.

- In the AF 296 example, it was the failure of the automation to react to the human controller
- In the other examples, it was the human failure to perceive the operational situation and to react properly.

Emergency confusion

Confusion in emergencies often disrupt clear thinking and effective decision-making when it is most needed. Confusion can stem from a variety of psychological, environmental, and situational factors. Common practices for coping with confusion include:

- Training and Drills: Regular practice helps individuals respond instinctively and reduce reliance on decision-making under stress.
- Clear Communication: Use straightforward, repeated, and visual messages during emergencies.
- Emergency Plans: Familiarity with evacuation routes and contingency plans can reduce uncertainty.
- Stress Management: Teaching stress-reduction techniques can help maintain calm during crises.
- Empowering Leadership: Designating leaders or roles within groups can foster swift and organized responses.

The focus of this article is on controller-service integrity-oriented operational design, namely, decision support and error-proofing. This challenge was demonstrated in the AF 447 accident, in which the operators did not perceive

the autopilot messages properly, and did not conceive the airplane situation properly thereof.

A model of decision errors

The model of decision errors is based on the cybernetics model of feedback control loops, as described in the STAMP paradigm, and developed in the STPA methodology (Leveson, 2004). The controller may consist of human, automation, and AI elements, and the controller task is to activate one or more processes, encapsulated in services.

Decision errors are due to diversion from feedback control loops, attributed to inadequate feedback from the processes. Accordingly, the engineering challenge is to prevent diversion from the feedback loops, and to support the activity required to resume normal operation.

The helm dilemma

A key topic is the control dilemma: who should take the lead? In the AF 296 accident the automation took the lead, disabling the human operator. On the other hand, in the AF 447 accident the machine pushed the lead to the operators, who failed.

In situations of moderate operational complexity, such as in the AF 296 case, the best choice may be manual operation. On the other hand, when in high alert, such as in the AF 447 case, it may be safer if the automation takes over the human controller, who might be paralyzed in the alarming situation. A possible solution to this problem is to facilitate the control transfer from the operators to the autopilot.

The challenge is to define rules for setting the leader. The rules should be expressed in terms of operational conditions that both the human controller and the automation can verify, and such that they can obey the rules.

Design challenges

In many emergencies, the operators should worry about severe risks of hazards in the context, which is outside of the system operation. To optimize solving problems in the context, the system operation might be seamless, consuming

only a little attention from the operators. Typically, the operational situation is fuzzy, the sources are latent, and the potential options are unknown. The challenge is to manage emergency situations while minimizing harm and mitigate the operational risks typical of emergencies.

Over-reliance on automation could introduce vulnerabilities in critical systems. Therefore, a key challenge of emergency control design is to enable seamless operation when under stress.

The design goals are to shorten the emergency, to eliminate the risks of operating under threat, to prevent operator errors, and to support resilient operation.

IV. The framework: A model of interaction control

The model of emergency control is projected from the general model of interaction control (Harel, 2023). The root cause of an emergency is operating in exceptional situations.

Exceptions may be defined with reference to normal operation: a situation is exceptional if it diverts from the procedure defined for accomplishing an operational task.

Defence strategies

Emergencies often result from operating safety-critical systems in exceptional situations. The common defense strategies are:

- Avoidance: prevent the triggers of exceptional situations
- Resilience: protect the operation in exceptional situations.

Avoidance

The focus here is on avoiding emergencies and supporting the human operator when under stress. However, this same model may also describe emergency operation of Systems of Systems (SOS), in which one of the subsystems controls the behavior of the other subsystems.

Resilience

The challenge is to impose safe operation in exceptional situations. The risk is of errors due to operator confusion.

Operational envelopes

To tackle the control allocation dilemma, recovery may consist of two coordination envelopes:

1. Controller envelope – controller-driven coordination
2. Protection envelope - service-driven coordination

The recovery begins when in the controller-driven envelope. While in this envelope, the controller is in charge of the recovery. Then, if the controller fails to handle the situation, the service needs to take the lead, overriding the controller's commands.

Coordination in the controller envelope

During the controller-driven coordination, the controllers may realize that they cannot handle the situation. The recovery in the AF 447 case failed because the autopilot enforced manual operation, which the pilot could not handle. In this case, they should be able to enforce switching from manual to automation.

The coordination in the controller envelope develops in three stages:

1. During the first stage, the service handles regular activities, as in normal operation.
2. During the second stage, the operation is under hazard. The primary design challenge is that the operators are aware of operating under hazard.
3. In the third stage, which is optional, the service is operated in safe mode. The design challenge is to prevent escalation.

Imposing clear communication and coordination between the controllers and services is vital for effective decision-making. This can include things like alert systems or interfaces that provide controllers with sufficient understanding of the service's behavior and rationale.

Coordination in the protection envelope

In transition from the controller envelope to the protection envelope, the control is transferred from the human operator to the automation. The service needs to take the lead, overriding the controller's commands.

During the third stage, the controller may realize that the automation is wrong, which might end up in an accident. To tackle this risk, the design should provide a means for the controller to overcome the automation. This means should enable fast switching from automation back to manual. However, to avoid unintentional switching to manual operation, the switching should not be easy. The design should include special means for preventing errors, such as by enforcing using both hands for the exceptional switching.

Control tasks

The model assumes eight tasks in emergency situations:

1. Detecting a risk
2. Hazard identification
3. Informing the human operator(s) about the hazard
4. Assessing the hazard risks
5. Proposing optional reactions
6. Evaluation of the optional reactions
7. Selecting the best option
8. Executing the selected option

In each task, the goal is to maximize efficiency and safety. These goals suggest a need for decision support, and for enforcing operation by rules.

V. Essentials of collaboration design

Situational complexity

Often, the response of the services to a command received from the controller depends on the service situation. Typically, it depends on the scenario. These dependencies are error prone. An event of activating a command in a wrong situation is called a situational error.

Situational complexity is a term used to indicate the likelihood of encountering a situational error. Scenario-based modeling enables linear situational complexity by assigning service situations to scenarios.

Control design

To ensure proper controller-service coordination, the system design should be based on utility-oriented rules, describing normal coordination and synchronization. To facilitate the rule definition, the controller operation should be defined in terms of scenarios, which must be defined explicitly, and implemented in the system. Situational rules should specify the normal situations, and the indications of exceptions. Activity rules should specify the response to scenario transition, the implied changes in the service modes, the sync time-out, and the reaction to exceptional activity.

Following the classical theory of decision making, special rules for enforcing access to utility-critical features, may apply to prevent type I (alpha) decision errors, and other special rules for disabling error-prone controls may apply to prevent type II (beta) decision errors.

Service design

To enable proper AI decision making, and to facilitate human decision making, the services should provide the controller with information about their situations, and feedforward about potential changes. The situations may be represented faithfully by risk indicators, based on statistical analysis of measurements of continuous service variables, such as performance and process time. The feedforward information should include predictions of future situations, obtained by trend analysis, and of potential service responses to optional controller decisions. A means to provide such predictions is by simulation, which may be performed by behavioral twins of the interacting processes, which the controller can either embed or activate.

Decision support in emergency

Deciding when and how control should be transferred between controllers and services in stressful environments is crucial. Sudden or ill-timed handovers might lead to accidents. The recovery in the AF 296 case failed because the autopilot took the lead too early, preventing the pilot from handling the

situation. The recovery in the AF 447 case failed because the autopilot transferred the lead to the pilot, although the pilot was not capable of coping with the unexpected situation.

Emergency control may be effective if it relies on available information about the risks. This kind of information may be based on indicators about the levels of risks during the development of stressful situations.

Risk indicators

A risk indicator is a system variable associated with a certain risk. For example, consider a system designed to operate at a temperature range between 10^0C and 40^0C. Suppose that operating at a temperature higher than 80^0C is dangerous, and operating at a temperature lower than -30^0C might result in damage to the system. To avoid the risks, the service may alert the operators about approaching extreme values. In this example, the temperature is a risk indicator, and the protection envelope provided with this indicator is the temperature range [-30^0C, 80^0C]

Emergency-oriented interaction design

A key challenge of machine-driven orchestration is to prevent operational errors. Most errors involve making wrong decisions. Therefore, the challenge is to support the operator's decision making. This requires providing situational preview to the operators, as well as situational preview of the effect of optional decisions.

Decision errors might end up in diversion from normal to exceptional situations. In case of diversion, the system needs to recover and subsequently resumes normal operation. These activities are complicated, and therefore they are error prone. The challenge is to prevent diversion and to support recovery by design.

Integration design focuses on enforcing seamless operation. The design goals comprise enforcing:

- Coordination between the system components
- Detection and reporting on operational hazards
- Robustness and recovery from hazards.

Unexpected exceptions may be detected by risk indicators, which are limits of system variables. Similar limits may serve for notifying the operators about approaching the safety envelope.

The safety limits

The safety limits define the conditions for the transition from controller-driven to service-driven coordination.

The system in the example above may be required to enforce safe-mode operation when the temperature is higher than 70^0C or lower than -20^0C. These limits define a safety envelope, denoting high risk in the range of [70^0C, 80^0C] and a low risk at the range of [-30^0C, -20^0C].

Predicted time to accident (PTTA)

The human operator may override the automation as long as the time to accident (TTA) is sufficient, Automated control may be required to override human control only when the TTA is too short.

The predicted TTA is a measure used to decide if and when the control should shift from the controller to the service. It may be calculated by risk indicators.

Elementary TTA prediction

Suppose that the system is facing a hazard, identified by a risk indicator. The service may predict the TTA by estimating the rate of temperature change based on trend analysis of measurements. In the example above, suppose that the rate of temperature change is 2^0C/minute, and the current temperature is 35^0C, then the predicted TTA is (80^0C – 35^0C)/2^0C/minute = 22.5 minutes. During the remaining 22.5 minutes, the controller and the service should coordinate in order to detect and eliminate the hazard, in order to prevent the accident.

Multi-sensor TTA prediction

Consider a system designed to operate at a temperature range between 10^0C and 40^0C and pressure range between 10 and 20 psi. Suppose that the system is facing a hazard, identified by two risk indicators: temperature and pressure.

Typically, the TTA predictions of the two sensors may not be the same. How should we combine the two predictions?

To be on the safe side, we may take the conservative approach, and choose the minimal prediction, to ensure that the last, automated stage will not be too late.

VI. Enforcing situation awareness

Situation indication

To enable recovery, the controller should be aware of the operational situation. The operation design should provide a means to indicate the hazards and the PTTA. The means may include visual indications of the hazard and the PTTA.

Attention-oriented alarm design

In normal operation, the indication should enable, but not enforce awareness of the situation, so that the controller may focus on the primary, attention consuming tasks. The challenge is to reduce attention demands. Only when the PTTA is significantly short, should the service enforce awareness of the hazard, typically, by audio signals.

Warning limits

The warning limits define the transition from the first to the second recovery stage of the controller envelope.

The system in the example above may be designed to warn when the temperature is higher than 60^0C or lower than -10^0C. These limits define a warning envelop, denoting the range for warning about high risk in the range of [60^0C, 80^0C] and about a low risk at the range of [-30^0C, -10^0C].

Warning notification

A general non-intrusive warning notification is a subtle alert meant to inform users about operating in an exceptional situation. Attributes of the warning notifications:

1. Position: Appears such that it does not cover primary content.

2. Color: Uses a calm, warning color like yellow or light orange to draw attention without creating urgency (e.g., red is often reserved for critical warnings).
3. Symbol: used for fast recognition of the hazard, and to indicate the nature of the warning at a glance.
4. Message: Contains a concise message, expandable to a screen with explanations on the source for the warning and tips for how to react.
5. Actionable: May include an "Action" control to react.

Alerting on hazards

Alerting on diversion from normal operation. Hazards are critical for risk management and public safety. It involves detecting potential threats or dangerous situations and communicating timely warnings to the relevant individuals or groups. Key components of a hazard alerting system include:

1. Detection: Using sensors, monitoring systems, or intelligence to detect hazards.
2. Risk Assessment: Analyzing the severity and likelihood of the hazard affecting the population. This helps determine the urgency and scope of the alert, as well as the attributes of the alert message.
3. Communication Channels: Once a hazard is detected, alerts must be communicated quickly and efficiently.
4. Message Clarity: Effective alerts must be clear, concise, and actionable.
5. Duration: The message remains visible until acknowledgement.
6. Real-Time Updates: As conditions change, continuous updates are essential. This allows operators to stay informed and adjust their response if necessary.
7. Integration with Emergency Services: Alerts should trigger a coordinated response from emergency services (fire, medical, police, etc.) to manage the situation effectively.

VII. Integrating AI into interaction control

AI processes may provide both the automation and the human operators with the reasoning required for protecting each of the emergency tasks above.

Avoiding the emergency

When the human operator is at the helm, AI can help with decision making, by evaluating the risk indicators and the optional operator reactions to the situation, and by guiding them in the situation awareness and choosing the safest option.

Resilience

When operating in exceptional situations, AI may help protect from escalation, managing safe-mode operation, and enforcing safe termination, such as by escape (e.g. Auto-GCAS) or shut down.

Task that AI can support

Here are key AI activities in the tasks of the control model:

1. Detecting a risk: automation support, by risk indicators
2. Hazard identification: automated troubleshooting
3. Informing: human situation awareness, based on HCD principles
4. Assessing the hazard risks: providing preview information, by trend analysis
5. Proposing optional reactions (option generation)
6. Evaluation of the optional reactions (preview, by simulation)
7. Selecting the best option (Avoid potential mode error)
8. Executing the selected option (Setting proper conditions).

VIII. Engineering

The implementation of the collaboration is within the scope of integration engineering.

The key to the collaboration design is the need to define the rules for constraining the operation, to eliminate emergency risks. Normal operation may be specified in terms of project specific operational rules. The implementation of these rules must be affordable in terms of budget and time to market, implying that the rules must be easy to define and implement. This is possible by adding a layer of models and rules about the system situations

and activities, on top of the V model. This extension is affordable if the rules are defined by standard templates, based on generic models.

To enable feedback from failure, it is essential that the system activity is traced during the operation. The system should be provided with probes, consisting of sensors and software, for capturing exceptions during integration testing, and post-deployment incidents of near miss. Testability features should include means for faking hazards during the integration testing.

Affordability

The challenge is to facilitate system integration, to allow all projects to incorporate the design guidelines. To facilitate the design and testing, common operational rules may be expressed as generic mini models (GMM; Harel, 2021), which may be customized to different kinds of emergencies. Here is a list of GMMs applicable to emergency control:

- Implementing scenario-based operation
- Diversion detection: automated detection and alerting
- Hazard detection, rebounding, and alerting,
- Decision support: situation preview, option evaluation, implemented by behavioral twins. These are digital twins optimized to support human activities.

Emergency-oriented interaction testing

To avoid surprise, we often apply integration testing, in which we capture and become aware of design limitations and mistakes. Integration testing focuses on proactive detection of unexpected exceptional activity. Testability infrastructure includes consols enabling to enforce exception, triggers and diversion to the exceptions, and to examine and evaluate the system reaction to the exceptions. The testing goal is to verify that

- The only exceptions detected in the testing are those that are unavoidable
- The system rebounds from all reboundable exceptions
- The operation does not escalate from those exceptions that are not reboundable
- The operation may continue safely during the exceptions

- The system may recover and resume normal operation

Model-based integration testing should be based on special testability features, enabling tracking and recording normal activity, faking and simulating triggers and exceptions, and software probes for comparing the results with the recorded expectations.

Following the testing, in hindsight, we are often required to go back and change the design, to prevent the problems captured and identified during the integration testing.

Optionally, a human operator may be present as a backup, to cope with situations unforeseen at design time.

IX. Conclusions

The future is likely to lie in hybrid systems where humans and machines work together. Automation can handle repetitive, data-driven tasks while humans focus on complex decisions. This combination leverages the strengths of both to improve efficiency and safety, while maintaining flexibility.

The timing of transition from controller control to service control is projected from the safety limits specified in the requirements documents.

To avoid surprise, prior to the transition, the service may provide warnings about crossing other limits, defined also in the requirements documents.

The scope of emergencies

The extended model applies to any engineered system. The scope of hazard to which the emergency control applies may extend from aviation hazards to anything from natural disasters (earthquakes, floods, wildfires) to technological failures, industrial accidents, or even health emergencies (like disease outbreaks).

Vision

The quality of operation relies on rule definition. To prevent human errors, we need to develop affordable methods to oppose accountability biasing, to constrain operation by the rules, and to detect and alert about exceptions.

For cross-domain learning, we need to create a cross-domain ontology, comprising standards that formulate the generic rules, and to enforce employing them by regulation.

The generic rules developed in prior studies may be customized and applied to various kinds of interactive systems, in various domains, thus enabling reducing the costs of eliminating operational risks.

References

Bainbridge, L. (1982). IFAC Proceedings Volumes, Volume 15, Issue 6, September 1982, Pages 129-135, https://doi.org/10.1016/S1474-6670(17)62897-0

Harel, A. (2021) - Model-based Human Interaction Design. Accepted to the 5th International Conference on Human Interaction & Emerging Technologies (IHIET 2022) Paris, August, Reprint. DOI: 10.13140/RG.2.2.22631.16807.

Harel, A. (2023) Essentials of Integration Engineering, Preprint, DOI: 10.13140/RG.2.2.19607.55201

Harel, A. (2024 A) - Sensor Integration Verification, DOI: 10.13140/RG.2.2.27400.23044

Harel, A. (2024 B) - Enforcing feature availability, DOI: 10.13140/RG.2.2.31237.36326

Harel, A. (2024 C) - Configuration verification, DOI: 10.13140/RG.2.2.32397.35040

Harel, A. (2024 D)- Enforcing feature availability: the Torrey Canyon case study, DOI: 10.13140/RG.2.2.25365.33766

Harel, A. (2024 E) - Enforcing feature availability: the PL 603 case study, DOI: 10.13140/RG.2.2.34802.52165

Harel, A. (2024 F) - Enforcing feature availability: the AF 296 case study, DOI: 10.13140/RG.2.2.19703.02725

Leveson, N. G. (2004). A new accident model for engineering safer systems. *Safety Science*, 42(4), 237–270. https://doi.org/10.1016/S0925-7535(03)00047-X

McRuer, D. T., & Miele, A. (2009). Dynamic control allocation for aircraft: A review. *Journal of Guidance, Control, and Dynamics, 32(2)*, 573-585. DOI: https://doi.org/10.2514/1.35231