CAN ARTEFACTS BE ANALYZED AS AN AGENT BY ITSELF – YES OR NO: WHAT DOES HUTCHINS ‘HOW DOES A COCKPIT REMEMBER ITS SPEEDS’ TELL US

Adriaensen, A. [1], Patriarca, R. [2], Smoker, A. [3], and Bergstrom, J. [4]

[1] Lunds Universitet, Division of Risk Management and Societal Safety, Sweden

adriaensen@me.com

[2] Sapienza University of Rome, Department of Mechanical and Aerospace Engineering, Italy

riccardo.patriarca@uniroma1.it

[3] Lunds Universitet, Division of Risk Management and Societal Safety, Sweden

anthony_john.smoker@risk.lth.lu.se

[4] Lunds Universitet, Division of Risk Management and Societal Safety, Sweden

johan.bergstrom@risk.lth.se

Abstract

This paper re-iterates Hutchins’ study ‘How a Cockpit Remembers Its Speeds’ from 1995. This latter emphasized the unit of analysis in exploring the processes and knowledge structures that underpin the activity of a socio-technical system. Hutchins investigated this concept through conceiving the cockpit as a joint cognitive system: how this work-system observed and remembered the speeds by which the aircraft operated, with a focus on turning non-observable properties of system performance into adaptive strategies.

Hutchins’ main conclusion was that “the classical cognitive science approach can be applied with little modification to a unit of analysis that is larger than an individual person” (Hutchins, 1995, p. 2). Starting from this macro-level conception about the joint system, our own study re-establishes that the unit of analysis in FRAM is functional, starting bottom-up by building a model of functions and inter-dependencies among agents, even at different levels of abstraction.

The novelty of this study is that the traditional boundary between medium and agent was abandoned in favor of accepting aircraft systems and artefacts as agents of their own, all entitled to produce functions in the FRAM model and how this effected task and information propagation.

1 INTRODUCTION: FRAM IN A NUTSHELL

FRAM and resilience engineering share several principles, from which the FRAM handbook lists the principle of equivalence, approximate adjustments and emergence. The abbreviation FRAM stands for Functional Resonance Analysis Method and carries two key concepts, functional-based approach and resonance. Resonance is used as an alternative to cause-effect relation and can be used to describe and explain non-linear interactions and emergent rather than resultant outcomes (Hollnagel, 2012). The other term ‘function’ deserves some further explanation. In FRAM, a function is presented by a hexagon and represents the acts or activities – simple or complex – that are needed to produce a certain result (Hollnagel, 2012). This can describe what people, organizations, technical or socio-technical systems do. Each hexagon form has six aspects: Input, Output, Requirements, Resources, Control, and Time. These are connected to each other by relationships. Some aspects can have more than one relationship, but not every single aspect (besides outputs) needs to be connected. The resulting dependencies are subjected to a non-linear analysis, in which it is important to note that the general principle is to analyze the task as it is carried out under normal conditions. FRAM’s point of departure is not to describe incidents, but the work domain. However, anomalies can be analyzed as special instantiations of the model in comparison to the task as normally done. The power of the model lies in the fact that the whole model only consists of hexagons and the way their aspects are interrelated. FRAM is a method-sine-model rather than a model-cum-method (Hollnagel, 2012), thus allowing to study the full potential of resilience engineering, as there is no discrimination between positive and negative outcomes and hierarchical control dependencies are avoided. Every FRAM model is defined by boundaries that are chosen with the scope of the study in mind, but still need to be selected in such a way that the model contains enough functions to explain the task, yet not more than needed. The model does not allow lost functions with no connections, forcing the researcher to identify logic relations and still reach a ‘closed’ model at some point. In the original Hutchins’ study, observational qualities were strong, but a rigid method was missing. The research question of our study is whether the

Hutchins' case can be described more methodically with the help of FRAM vocabulary and modelling. And concerning the FRAM methodology, can artefacts be analyzed as agents of their own? The remainder of the paper is organized as follows. Section 2 details the research method we developed starting from FRAM. Section 3 summarizes the results of the study, applied to the DC-9 cockpit, exploring the work-as-imagined and the work-as-done for speed setting. Section 3 also shows some potential guiding principles for the analysis, i.e. mechanisms. Lastly, the conclusions summarizes the outcomes of the study and the potential for further research.

2 RESEARCH METHOD

Our research approach started from the concepts of information flow, transformation of information and distributed access to information in cockpits, as discussed in Hutchins' paper 'How a cockpit remembers its speeds' and Palmer's 'Altitude Deviations: Breakdowns of an Error Tolerant System', co-authored by Hutchins (Hutchins, 1995; Palmer, Hutchins, Ritter, & vanCleemput, 1991). Typical elements of information flows are clearances, instruction read-backs, instrument bug settings, analogic and digital instrument inputs, pilot checks, written checklist information, etc.

We firstly developed an ecologically valid FRAM model, based on work-as-imagined, by consulting company manuals and the expertise of one the researchers who also was an airline pilot, although not on the DC-9 aircraft. Thereafter, the model has been finalized describing work-as-done, thanks to the knowledge of three Subject Matter Experts (SMEs), i.e. former DC-9 pilots, involved in a focus group. The pilots, after a session discussing the theoretical background of FRAM, were interviewed to gain operational knowledge of the work domain. They were free to alter - even delete or add - any function and/or their aspects. Whereas the original Hutchins' paper was based on the MD-80 aircraft, we based our exercise on its predecessor, the DC-9 because it enabled us to work with three former DC-9 pilots from the same airline. This was needed to build an accurate work-as-done model dictated by a single set of company procedures and practices. When building the model, we supported our SMEs with cockpit training posters, cockpit pictures and copies of the actual company manuals.

Generally, FRAM researchers use FRAM Model Visualizer (FMV) to develop a model. However, for complex models characterized by many functions, FMV presents limitations on user-experience. Additionally, FMV is primarily aimed to analyze iterations (walk-throughs) of the model and to generate static relationship overviews, which can become visually overwhelming depending on the amount of functions and aspect. For the purpose of this study, one of the authors developed a VBA code, which interacting with FMV, was able to create a FRAM model, starting from a user-friendly data entry form in Microsoft Excel. Operationally, this means that all functions have been assigned to different levels of resolution (i.e. abstraction), through Excel fields tied to the functions. This complexity-management strategy was based on Rasmussen's Abstraction/Decomposition framework (Rasmussen, 1985), recently adapted in an Abstraction/Agency framework developed to give sense to the functional resonance space (Patriarca, Bergström, & Di Gravio, 2017). Since the abstraction level reflects the "lens" used to describe the work domain, we defined the abstraction layers after gathering data from the focus group. Possible ambiguities were discussed with our pilot informants, by semi-structured interviews.

The novelty of this study is that the traditional boundary between medium and agent was abandoned in favor of accepting aircraft systems and artefacts as agents of their own, all entitled to produce functions in the FRAM model. This allowed to analyze how task-relevant information moves "through the cockpit system by translating the representation of information in one medium to a representation in another" (Palmer et al., 1991, p. 18). Whether this information is than transformed through the process of human cognition, an aircraft system, or even an artefact makes no difference in terms of the model we developed. Although the FRAM handbook itself brings to the attention that functions can also describe technical systems (Hollnagel, Hounsgaard, & Colligan, 2014, p. 23), this paper radicalizes this possibility by defining artefacts and even environmental influences as agents of their own.

3 RESULTS

3.1 Speed setting-as-imagined

The work-as-imagined model in the company manual contains procedures, compact and expanded checklists, calculation tables and diagrams and detailed responses to a wide variety of circumstances. It is interesting that there is no part or a sum of parts of the company manual that fully explains to the pilots, the actual sequence of activities, schedules and events meant for speed control from the aircraft's Top of Descent (TOD) until landing. Even when this information is compiled from different parts of the manual, sometimes located far apart and in mixed forms of text and charts, the dynamic sequence of events involved could not be derived in all its aspects from the written procedures. In the company manual, one example being that there was no reference about how exactly the approach descent checklist is triggered; even if this describes an essential part in the procedure. The lack of a full description demonstrates that aircraft handling is supplemented by a set of commonly shared procedures that are verbally communicated in on-the-job-learning and training, from which follows that a work-as-done model is a more accurate basis for this non-trivial task analysis.

The analysis of the company procedures and manuals reveals a sequential process paradigm. This can typically be found in speed reduction schedules from standard approach charts that depict the aircraft in relation to descent profiles or beacons. Another example is the chronological description of standard procedures per flight phase, accompanied by checklists. A further observation is that adaption to a wide variety of both standard and non-standard scenarios, is presented as fragmented information in the form of add-on descriptions or variations to a leading scenario. This decomposition of information contrasts with the dynamic sequence - not always sequential - of actions and events, by the work-as-done model contained in this study. Therefore, the information contained in the manual from both simple and complex procedures requires a cognitive translation into the actual dynamic processes they represent. Even if this company manual is already a few decades old and does not represent today's industry best practices, the decomposition of information remains undoubtedly a major challenge for all training purposes and company manuals. Not to mention that "[t]he work situation in large-scale socio-technical systems is partly intractable", meaning "that the conditions of work are underspecified, in principle as well as in practice" (Hollnagel, 2012).

3.2 Speed setting-as-done

Different datasets can be described, from which Hutchins used the particular example of the task propagation and information flow related to the memorization and utilization of speeds for the aircraft's approach and landing. Four years earlier he illustrated the same principle together with Palmer about altitude, instead of speed specifications:

> "Notice that the altitude specification is in some sense the "same" information whether it is represented in spoken words, as a string of written characters, or in the digits visible in the display window of the altitude alerter. . . .A medium is said to represent some particular piece of information by virtue of having a particular physical state. For example, the altitude-alerter display window represents the altitude thirty-three thousand feet when the digits "33000" appear in order in the window. The crew moves information through the cockpit system by translating the representation of information in one medium to a representation in another" (p.18).

The description above is an insightful illustration how the same information becomes translated in different forms and can subsequently be used as the origin for possible paths that task-relevant information can take in the cockpit system (Palmer et al., 1991, p. 18) Hollnagel and Woods (2006) recognized the Hutchins study as one of three hallmark papers on cognitive task syntheses that later gave rise to frame JCS. Although Hollnagel et al. applaud the incredible quality of observation of those pre-JCS studies, they also note that in each of them any discussion of methods is virtually absent (Hollnagel & Woods, 2006, p. 39). In our current study this problem is resolved, by systemically analyzing each transformation of task-relevant information or a task-related-action with the help of FRAM functions. A complete walkthrough of the information, supported by figures and the full data would be too lengthy for the format constraints of this paper. One can also recognize in Figure 1 that the visualized model is beyond interpretation. Note that even then, the static visualization does not yet tell the full examination narrative. It is a supportive tool that initiates or accompanies the analysis. Therefore, Figure 1 is only provided at a small size-fitting scale to give the reader an understanding of the complexity of all relationships that were required in this specific task. In this data section, we will rather unravel some of the applied mechanisms and apply them to some examples.

3.3 Mechanisms for modelling and analysis

One of the challenges we encountered by studying other literature that made use of FRAM and in making our own FRAM model, is the fact that the method still leaves a certain degree of liberty in defining functions. This might affect the results in terms of reproducibility. We found that: i) the degree of granularity as processed in a function is underspecified, and ii) the choice for agents ranging from human to technical, although well-described by Hollnagel (2012), is rarely applied in other FRAM studies. With the mechanisms described below, we describe some guiding principles for the breakdown of steps when defining one or more FRAM functions, and we advocate radical freedom for the choice of agents that can drive functions.

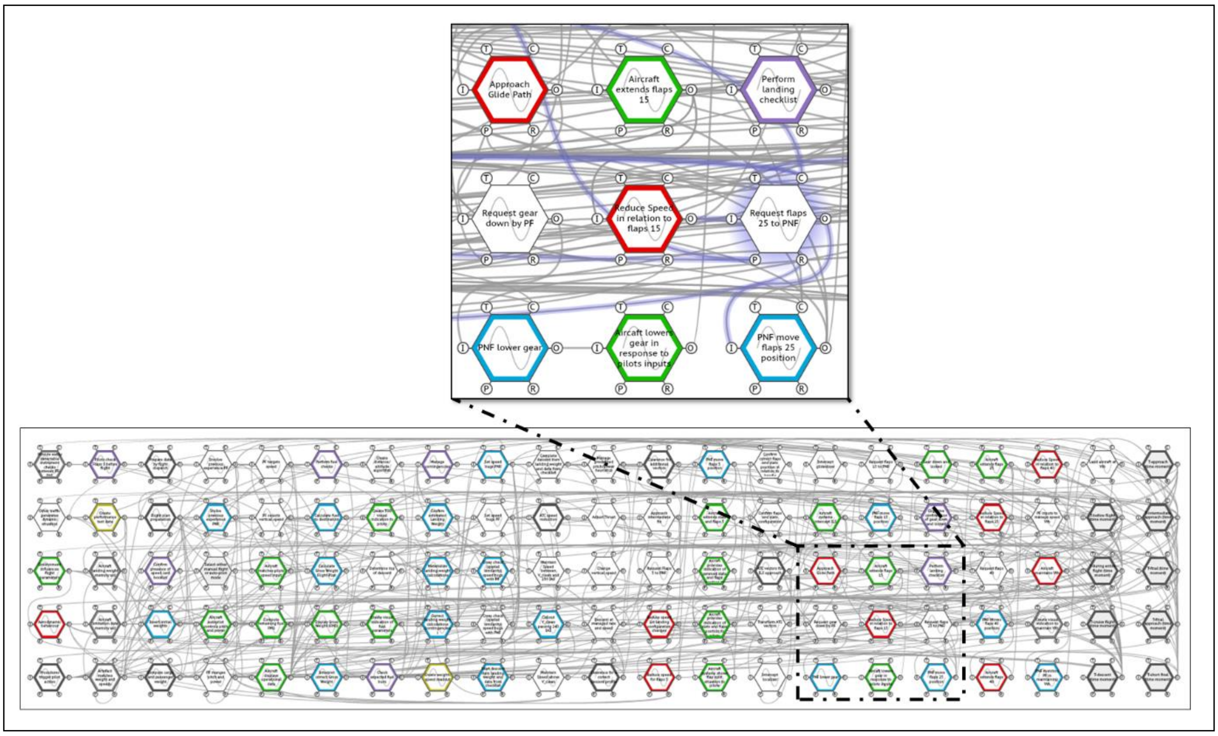

Figure 1. *The FRAM model for speed-setting in a DC-9 cockpit. Note how the graphical representation becomes overwhelming due to the huge number of functions and tight interactions among them. Note that each colour represents a different agent, according to a JCS perspective.*

Mechanism #1: a breakdown of functions (the different steps in the task analysis) in its smallest significant parts.

For example, one could typically define in FRAM that the function <Set Flaps 25>, requires Input <Flaps position smaller than 25> and produces Output <Flaps 25 position>. However, it can also be described by describing several smaller functions. In our model, the initial general function has been broken down into the following steps, beginning with a function called <Request Flaps 25 to PNF>, requiring Input: <Reducing speed from Vp15 to Vp25 and below> and producing Output <Call out "Flaps 25">, by agent Pilot Flying (PF), i.e. PF slows down the aircraft and immediately requests the new flap setting to his Pilot Non Flying (PNF), followed by Function <PNF moves handle to flaps 25>, requiring Input <Call out "Flaps 25"> and producing Output <Flaps 25 handle position> and <Call out "Flaps 25 selected">, by agent Pilot Non Flying (PNF). Thereafter the agent <aircraft systems>, reacts with the function <Aircraft extends flaps 25>, requiring Input <Flaps 25 handle position> and producing Output <Flaps 25 extended left side> and <Flaps 25 extended right side>. This sequence of three functions equals one single function at the beginning of our example. Note that in our model the task has suddenly travelled through 3 agents, Pilot Flying, Pilot Non-flying and Aircraft systems, instead of one.

Mechanism #2: look for the function that requires the least processing effort (cognitive or computational).

So far, we have described that the aircraft has extended its flaps at the left and the right side. However, we have not described in which way the pilots receive feedback from the aircraft systems to tell them that the flaps, which to them are out of sight, actually are in the desired position. Therefore, the pilots need to check that the flaps indicator (e.g. Flaps 15 instrument indication pointer) corresponds to the correct handle position (e.g. Flaps 15 dent position). Although this single confirmation appears sufficient, chances are high that a physically blocked flap also causes an asymmetry. In this case, the pilots would immediately see that the two pointers on the flap indication instrument would not be aligned. Verifying the spatial similarity of two overlapping pointers on a single dial is translated in the verification that <Flap pointers left-right show no split>, a typical example of something that hardly requires any cognitive computation. 'Pattern matching' of two aligned pointers is cognitively less demanding and less prone to error, compared to checking a flap handle position (at a head-down visual position), with 2 pointers on the flap indication instrument (at a heads-up visual position). But at the same time checking the handle position is more thorough, because it includes the scenario where 2 flaps would have failed symmetrically. Note that it is therefore not strictly necessary to create the <Flap pointers left-right show no split> function, because it is superfluous to the confirmation of checking the flap pointer indications with the respective handle position and it would not appear in a work-as-conceived model. However, this two-step efficiency-thorough checking is how cognition really functions in this case, according our SMEs, and therefore we must add one step to the work-as-done model. FRAM does not tell us to add this step, and chances are likely that our SMEs would even overlook this step if not confronted with the right interview questions. One could say that it was Hutchins that unveiled how cognition is often not the "reconstruction of some internal representation", but rather a "combination of recognition, recall, pattern matching, cross modality consistency checking" (1995, p.18). Therefore, Hutchins in return learned us how to better define FRAM functions that involve cognitive processes.

Mechanism #3: the principle of agent-neutrality.

Now we go back one step back in the model to the end of example #1. The function <Aircraft provides indication of flap position to pilots>, receives Inputs were already produced at the end of #1 with <Flaps 25 extended left side> and <Flaps 25 extended right side>, as outputs of the preceding function. This function in turn produces the Outputs <Pointer indication of slats and flaps left side> and <Pointer indication of slats and flaps right side>. This is followed by the Function <Aircraft systems shows flap split situation to pilots>, with the Input <Pointer indication of slats and flaps left side> and <Pointer indication of slats and flaps right side> and the output <Flap pointers left-right show no split> or in case of a contingency <Flap pointers left-right show split>. Only thereafter it becomes one out of several Controls that the pilots (PF) receive as the Function <Confirm the flaps and slats configuration>. Note that by abandoning the traditional boundary between medium and agent, there is little cognitive difference in labelling the aircraft systems' function which <*shows* flap split situation to pilots> on the one hand, and the function of the human pilot agent who <*confirms* the flaps and slats configuration>, by means of the Control, i.e. <Left-and right-hand side pointers are aligned>. This is not to say that there is no minimal cognitive processing involved in pointer checking, but the model treats human and technical processes equally.

Mechanism #4: consider parallel sequences of functions for (apparently) symmetrical tasks.

We often created parallel sequences of tasks for PF and PNF, even if both pilots performed identical tasks. When a checklist says that speed bug settings need to be cross-checked, one could define a single function for that cross check, but due to the trajectory of information and/or tasks, we found that such symmetries were often discontinued a few functions away, downstream or upstream. In our model, PF and PNF for example, received separate cross-check functions for this reason. <Cross check speed bug settings between PF – PNF> is an equal but distinct activity for both pilots. However, downstream the PNF starts using these bugs as a retained memory aid for monitoring his PF, which then ceases to be a reciprocal function. Upstream such crosschecks are linked to checklist actions which are not symmetrical. This happens by the very nature of challenge and response type checklists. However, even in this study, some functions handled by both pilots were still allocated to a single function, and as a consequence, a single agent <joint pilot team>. The motivation could be that they happened at an abstraction level from which no physical functions were defined. This is for example true for contingency management, which this model did not further investigate beyond defining such contingencies.

3.4 Propagation of tasks

Central in this study are the speeds used for controlling and landing the aircraft and how they are memorized. "In order to maintain safe flight at slower speeds, the crew must extend the slats and flaps to produce the appropriate wing configurations at the right speeds" (Hutchins, 1995, p. 4). The appropriate speeds differ with varying aircraft's weight and will be calculated just before or after the descent point. In the MD-80 example from the Hutchins study, the transformation from the landing weight into the accompanying speeds is accomplished by a very effective artefact, a booklet of speed cards. "The booklet contains a page for each weight interval (usually in 2,000 pound increments) with the appropriate speeds permanently printed on the card" (1995, p.5). The DC-9 example used in our study, had a nearly identical booklet. Turning the booklet on the correct page simply means that the pilots have the correct landing speeds displayed. The booklet is then positioned on a prominent place in the cockpit. The speeds from the booklet serve to set the speed bugs, another simple, but effective memory aid. The speed bugs, are moved by a rotating knob and physically positioned around the analogue Air Speed Indicator (ASI) dial. Both pilots have their own ASI, each with their own set of speed bugs.

In Hutchins view, this makes the system redundant in memory and processing and provides more redundant checks. "The interaction of the representations in the different media gives the overall system the properties it has" (1995, p. 19). The propagation of tasks can be seen as a set of transformations of information flows and the pilots' reactions to these changes. Speed bugs become involved in a distribution of cognitive labor across social space and "the cognitive work of reading the airspeed indicator, and monitoring the other instruments on the final approach can be divided among the pilots" (1995, p. 19). Speed bugs do not help pilots remember speeds, rather they are part of the process by which the cockpit system remembers speeds. Also, “speed bugs permit a shift in the distribution of cognitive effort across time. They enable the crew to calculate correspondences between speeds and configurations during a low workload phase of flight, and save the results of that computation for later use” (1995, p. 19).

A central question is if a FRAM model can handle a similar propagation of tasks. The answer is that a FRAM model can certainly be read as a propagation of tasks, although the relations must not necessarily be read as sequential steps. The FRAM model allows to depict non-linearity when the output of functions leads to several Preconditions or Controls for impending functions. For example, in the case of a Precondition, this does not necessarily define a tight sequence, but it creates a condition for certain functions that cannot be started before other critical functions are fulfilled.

Furthermore, when Hutchins’ less systematic, but more descriptive approach would be translated into FRAM, his transformations of information would be mere Inputs and Outputs, whereas (e.g.) one could describe the crosschecks between pilots as Controls. FRAM vocabulary has several other aspects and thereby has a richer potential for describing dependencies and exploring the work domain. Even a pilot crosscheck, intuitively thought as a Control, might be a Precondition for another function and thereby have the control to start or delay a specific task. This gives FRAM, in contrast to the Hutchins, the opportunity to describe non-linear relations.

4 CONCLUSION

Hutchins described 3 accounts to “examine the activities in the cockpit that are involved with the generation and maintenance of representations of the maneuvering and reference speeds” (1995, p. 5): 1) a procedural description of memory for speeds; 2) a cognitive description by means of representations and processes outside the pilots and; 3) the same by representations and processes inside the pilots. He dismissed all three in favor of a system-level cognitive view that “directs our attention beyond the cognitive properties of individuals to the properties of external representations and to the interactions between internal and external representations” (Hutchins, 1995, p. 20). This study not only agrees with this, but included these findings in its methodology. We dismissed a procedural description by starting from a work-as-done model and we were mindful not to use FRAM functions that would represent possible opaque cognitive processes. In fact, FRAM allowed us to treat the transformation of energy or information by human agents, aircraft systems or artefacts as agent-neutral transformations and thereby avoided the use of cognition in its classic sense. By using FRAM as a model or even a language to treat humans and machines as equivalent producers of functions, the joint performance of the system can be described as the net result of the resonances between functions. FRAM proved to be a tool that allowed to study JCS more systematically. However, this implies that some pre-concepts, which we described as a set of mechanisms to define FRAM functions are aligned with JCS. At this point, FRAM itself does not tell us something about those choices and this study benefited from studying the JCS concepts. The results of this study should offer future researchers a chance to determine a level of granularity about the unit of analysis, in relation to the question which agents are entitled to create functions. The model developed in this study, might be helpful to gain in-depth knowledge of the work domain, with potential implications for both organizational learning and safety management, using the cited mechanisms.

REFERENCES

Hollnagel, E. (2012). *FRAM: The Functional Resonance Analysis Method: Modelling Complex Socio-technical Systems*: Ashgate Publishing, Limited.

Hollnagel, E., Hounsgaard, J., & Colligan, L. (2014). *FRAM – the Functional Resonance Analysis Method – a handbook for the practical use of the method*: Centre for Quality, Region of Southern Denmark.

Hollnagel, E., & Woods, D. D. (2006). *Joint Cognitive Systems: Patterns in Cognitive Systems Engineering*. Boca Raton; London; New York: Taylor & Francis.

Hutchins, E. (1995). How a cockpit remembers its speeds. *Cognitive Science, 19*(3), 265-288.

Palmer, E. A., Hutchins, E. L., Ritter, R. D., & vanCleemput, I. (1991). Altitude Deviations: Breakdowns of an Error Tolerant System: National Aeronautics Space Administration and U.S. Department of Transportation Federal Aviation Administration.

Patriarca, R., Bergström, J., & Di Gravio, G. (2017). Defining the functional resonance analysis space: Combining Abstraction Hierarchy and FRAM. *Reliability Engineering & System Safety, 165*, 34-46.

Rasmussen, J. (1985). The Role of Hierarchical Knowledge Representation in Decisionmaking and System Management. *IEEE Transactions On Systems, Man, And Cybernetics, 15*(2), 234-243.