Methodology Comparison for Designing a Decision-making Support System

Reitsma, Jelmer P.; Van Paassen, M. M.; Mulder, Max

DOI
10.1016/j.ifacol.2019.12.167

Publication date
2019

Document Version
Final published version

Published in
IFAC-PapersOnline

Citation (APA)
Reitsma, J. P., Van Paassen, M. M., & Mulder, M. (2019). Methodology Comparison for Designing a Decision-making Support System. *IFAC-PapersOnline*, *52*(19), 157-162. https://doi.org/10.1016/j.ifacol.2019.12.167

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-19 (2019) 157–162

Methodology Comparison for Designing a Decision-making Support System

Jelmer P. Reitsma, M.M. (René) van Paassen, Max Mulder *

* *Control & Simulation, Department Control & Operations, Delft University of Technology, Delft, The Netherlands*

Abstract: Designing interfaces for effective decision-making supports for complex, dynamic systems is a challenging task. Besides the already challenging task of determining the visual form, the task of defining the content of these supports can be even more demanding. Especially for an unstable and complex work domain with multiple stakeholders and multiple interrelated systems, e.g., commercial flight operations. Various methodologies for designing such supports have been introduced in the last decades. In this paper two methodologies, Ecological Interface Design (EID) and Applied Cognitive Work Analysis (ACWA) are compared to determine what methodology is best suited for the design of an in-flight decision support system. The methodologies are compared on two aspects, (1) development of the knowledge-based model and (2) the means to translate this model into requirements for the actual representation. The functional abstraction network (FAN), as part of the ACWA, is the preferred knowledge-based modelling method for capturing a complex multi-system work domain, like commercial flight operations. Mainly due to the increased flexibility in modeling and ease of extending the model. The ACWA is also found to the preferable method to translate the functional model into representation requirements due to its structured step-wise and system engineering inspired approach.

Keywords: Human-machine interface, decision support systems, ecological interface design, applied cognitive work analysis

1. INTRODUCTION

Interface design for complex systems has been a topic of considerable research and will likely receive even more attention due to the advances in computational technology. Research on user interface design can be categorized based on the emphasis of the research, with the main three methods known as user-driven, technology-driven or problem-driven (Bennett and Flach, 2011). User and technology driven approaches aim to provide solutions for effective interactions, perception and usability (Norman, 1988). They are particularly focused on the visual form of the presented information and on making sure that the user understands how to interact and use the system. Traditionally, this is called human-machine interaction. This approach is dyadic in that it is focused on the interaction between the human and the computer (Bennett and Flach, 2011).

Although this approach is crucial for effective support, i.e., obtaining the operators goals, one important aspect of the human-machine system is not properly considered in the dyadic approach. Which is the interaction between the system and its environment. Systems are designed to perform work within a particular environment and successful control of a complex system is not merely determined by the human-machine interaction, but is also largely determined by the interaction of the system with the environment (Hollnagel and Woods, 2005) (Bennett and Flach, 2011) (Rasmussen et al., 1994). The operator should know what the system can or cannot do in order to reach his/her goals. In the triadic approach, the interactions between the system and its work environment are systematically studied. The triadic approach is the dyadic approach complemented with an accurate mapping of the interactions between system and environment. The triadic approach is more focused on determining *what* should be presented, rather than *how* it should be presented. However, one should never see the *what* and *how* separate since they both determine the effectiveness of the human machine control system.

The challenge of the triadic approach is to accurately describe the interaction between the system and its environment. Traditionally, interactions are described with *tasks* (Rasmussen et al., 1994). Here, a task is the 'cause' for an effect in the physical world. Describing interactions in terms of tasks can be useful in stable work environments. For these environments stable work procedures or a normative task sequence can be produced. Describing checklists and procedures are a way of describing tasks, which are common practice in aviation. Checklists work well if the conditions are exactly as the predefined conditions, but any deviating from these conditions can lead to a breakdown in performance. Describing interactions with a one-to-one mapping necessitates the formulation of tasks for every specific environmental condition. However, as Burian et al. (2005) describe, it is impossible and unpractical to develop procedures and checklists for every possible situation. In case of an unanticipated event, operators have to redefine their tasks based on features of the work envi-

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.12.167

ronments and the operators' interpretation of the available means to meet the system goals (Rasmussen et al., 1994). And so, they have the risk to be left with limited support during such events, which compromises the performance of the entire system (Vicente and Rasmussen, 1992).

A sequence of tasks in a stable environment can be considered as an "open loop" control. Rasmussen et al. (1994) illustrate this principle well with an example of baking bread. A recipe describes how the bread should be baked and once placed in the oven, no corrections are possible. This type of control only reaches its desired output state if the initial conditions are satisfied. Any disturbance can compromise the success of the baking process, e.g., placing of an additional tray on top. On the contrary, if no recipe exists the operator acts differently. The outcome of the process is determined by judgement and experience of the chef. The chef will closely control his/her activities during the process and adjusts these based on the observations of the effects to ensure the intended output. Hence, any disturbance can be counteracted, within the boundaries of the system, to make sure that the end result is as intended. This example illustrates the importance of presenting the interactions between the system and the environment not merely as a recipe, but providing a functional envelope which presents the work domain's constraints (Vicente and Rasmussen, 1992) (Rasmussen, 1985). The constraints of the system should provide transparency on how the system functions and provides the decision-maker with a presentation of the affordances from which he/she can determine the action that results in the intended output. This will allow the operator to anticipate in advance to achieve the desired outcome (Hollnagel and Woods, 2005) (Bennett and Flach, 2011) (Rasmussen et al., 1994).

Now the question remains, how should this presentation of the functional envelope be obtained and presented to allow effective control in events that deviate from the anticipated environmental conditions? In this paper we will evaluate two methods designed for this specific purpose, namely (1) Ecological Interface Design (EID) developed by Vicente and Rasmussen (1992) and (2) Applied Cognitive Work Analysis (ACWA), developed by Elm and Potter (2003).

The comparison will be based on two aspects (1) the development of the knowledge-based model, i.e., how the end-result scores on completeness, the modeling effort and extensibility to allow an incremental design approach, and (2) the ease of translating the knowledge-based model into requirements for the actual representation. The comparison is performed with the objective to select the best-suited method that can be used for designing an in-flight decision support system for the real-world test case of commercial flight operations. But first, an overview of the two methods will be discussed below.

2. ECOLOGICAL INTERFACE DESIGN

The Ecological Interface Design (EID) framework is developed to design and evaluate complex systems and especially for unanticipated events (Vicente and Rasmussen, 1992). The framework is structured around two questions: (1) what is a psychologically relevant way of describing the complexity of the work domain? (2) what is an effective way of communicating this information to the operator?

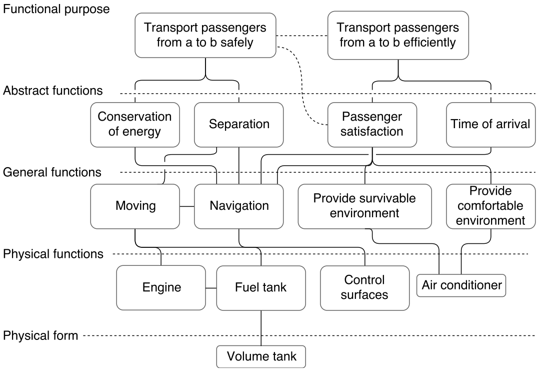

Fig. 1. Example of an abstraction hierarchy (AH) used in the EID

Rasmussen and Vicente's preferred formalism to describe the work domain constraints is the Abstraction Hierarchy (AH), where the Skill, Rule, Knowledge (SRK) taxonomy provides guidelines for the visual form. The term "ecological" is used to refer to an interface that has been designed to reflect the constraints of the work environment in a way that is perceptually available to the people who use it.

Work domain analysis The abstraction hierarchy is used to describe the functional envelope of the system, or in other words, the externalized mental model that the operator should have to successfully control system to reach the intended goal. It is a derivative of a functional-decomposition used in system engineering (Burns and Hajdukiewicz, 2004). A simplified example of the AH is shown in Fig 1. The AH consists of five levels of abstraction:

- Functional purpose level - describes basically what the system is designed to do.
- Abstract function level - describing the underlying values, laws or principles.
- Generalized function level - show how the abstract layer is achieved and is describing the processes involved.
- Physical function level - describes what equipment is involved, e.g., components or systems.
- Physical form level - describes the physical appearance. While often this layer is disregarded or not elaborated in much detail.

An important part of the AH are the means-end links, or the *how/why* links between the levels. These links describe how the layer above is achieved by the layer below (Burns and Hajdukiewicz, 2004). These links support reasoning and options to manipulate the system.

Skills, rules, knowledge taxonomy The skills-rules-knowledge taxonomy was introduced to match the presented information to the human information processing capabilities. All three types of human cognitive behavior, i.e., skill-based, rule-based and knowledge based behavior, should be supported by the interface in order not to force the operator in a higher level of cognitive control (Vicente and Rasmussen, 1992).

EID has consistently improved performance compared with the industry state of the art as Vicente (2002) describes. It has its roots in mainly isolated, well structured

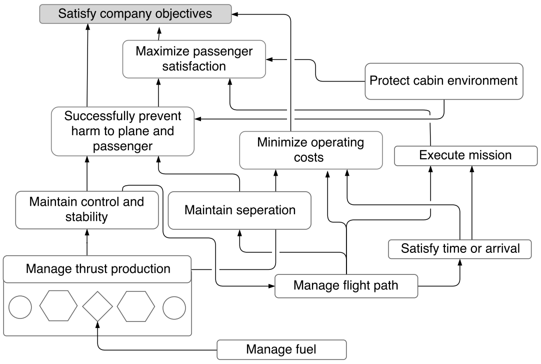

Fig. 2. Example of a functional abstraction network (FAN) as used in the ACWA

process control systems or power generation, but is also tested in the field of aviation, medicine, road transport, command and control and several more (McIlroy and Stanton, 2015; Van Paassen et al., 2018; Borst et al., 2015).

3. APPLIED COGNITIVE WORK ANALYSIS

The Applied Cognitive Work Analysis (ACWA) was introduced by Elm and Potter (2003) and is intended to be used for real-world engineering purposes. It is based on work of Lind, Woods, Hollnagel, Rasmussen and Vicente, but applied to become a more pragmatic process for designing decision support systems. The methodology is developed with the following underlying premises. First, the decision-making support system must embody a ‘knowledge model’ that closely parallels the mental model of expert human operators, just like EID. This knowledge model is composed of functional nodes and relationships intrinsic to the work domain. The basis for this knowledge model is an adaption to the AH, which is used to represent the abstract functional concepts. Second, Potter et al. (2000) observed that many results of cognitive task analysis are weakly linked to the actual design. Hence, ACWA is designed to provide smaller steps to transform the results of the various analysis into traceable design requirements. Hence, the process in ACWA is divided into a sequence of small, logical engineering steps. The five steps in the ACWA are:

Functional Abstraction Network (FAN) The knowledge model as part of the ACWA is an adapted version of the abstraction hierarchy, i.e., Functional Abstraction Network (FAN). This FAN is a function-based multi-level recursive goal-means decomposition and is based on the work of Woods and Hollnagel (1987), the FAN was considered by Vicente and Rasmussen (1992) as an alternative to the AH. A simplified example of a FAN is shown in Fig 2. The basic idea is to understand the goals to be achieved and functional means available for achieving them (Elm and Potter, 2003). Each functional block consists of a goal and a process to achieve the goal. Functional blocks are linked to specific elements in a process. This enables that blocks lower in the FAN are of a lower level of abstraction.

Cognitive Work Requirements (CWR) Once the model of the work domain is created, cognitive demands for that work domain model can be derived. For each functional block in the FAN (each goal, each component of the functional process) cognitive work requirements can be determined. These requirements can be in the form of monitoring, control or assessment activities, which are required by the operator for successful control.

Information / Relationship Requirements (IRR) The next step will be to identify what information is required for each CWR. It reveals what computations and/or transformations are needed. In this step data become contextualized *information*. This process is focused on satisfying the CWRs and is not limited by data availability in the current system. This step might uncover the need for additional transformations or sensors.

Representation Design Requirements (RDR) The next step is to describe the requirements for the intended representation of each set of CWR and IRR witht the intend to make sure that each presentation element effortlessly communicates its information. This explicit description becomes the key artifact between system developers and has the goal to eliminate subjective arguments during the display design phase.

Presentation Design Concepts (PDC) The final step of the ACWA is used to transform the requirement into its visual form (shape, color, layout). The techniques to optimize for perception, usability and effective interactions are used to create the most suitable display. This step is the largest to bridge and requires significant skill and experience. But the end result can always be verified for completeness based on the RDRs.

4. TEST CASE

Both methods will be applied to the test case of in-flight operational decision-making in commercial air transport. In-flight decision-making is a challenging task (Shappell et al., 2007). Flight crews have to monitor, assess the flight progress continuously during a flight and evaluate if the initial flight constraints with which the flight plan was constructed still hold. Unexpected events are happening frequently due to influences of weather, air traffic, aircraft (sub)systems and airport facilities. The flight crew needs to have a clear picture on what affects their operation and determine how to handle the impact.

Analyzing what affects flight operation is a challenging task, because of the many factors and stakeholders. This test case is complex in the sense that it not only embodies the complex system by itself, i.e., aircraft (sub)systems, but also includes the many interactions with (and between) other systems like weather, air traffic, terrain, regulations, passengers and company preferences. These may all need to be included as they all have the potential to affect the operational decision to some extent.

A traditional task-based approach for this “‘unstable”, open, complex environment is not preferred, as discussed before, because it is simply impossible to cover all possible conditions. Therefore, this is an ideal case for supports that can cope with unanticipated events, developed either with the EID or ACWA methodology.

5. RESULTS

Several observations were made about the practical implementation of both methodologies applied to the test case.

5.1 Development of the knowledge-based model

Completeness & Extensibility One observation made is that it is challenging to specify detailed functional purposes in the abstraction hierarchy. Various example AHs have “safety” as a functional purpose *combined* with other functional purposes such as productivity, efficiency, or increase dividend, comfort (Xu, 2007; Bisantz and Burns, 2008; Stanton et al., 2016). But what does safety mean specifically? We can describe it as; not harming anyone or anything involved. However, this goal comes from an even higher mechanism. Besides the ethical considerations, companies don’t what to harm anyone since they will not be allowed by authorities continue operation and no customer will ever buy an unsafe product (depending on the price and demand). This gives us the notion that capturing the functional purpose of a system with safety is not entirely complete. The real purposes are often purely economical and / or political. These layers are represented by Fig 3.

This risk of unspecific goals is also present in the FAN. However, the goals of each functional block are focused on success, which forces the analyst to be more specific on defining what embodies success. The higher level mechanisms, as discussed before, can be added on top with additional function node, with a certain process, until no higher level goals are found. The result is a more complete model. Each stakeholder has different goals and a different perspective of a problem. This means that the functional purpose of the system will be different for various stakeholder. This can lead to a set of inconsistent goals.

A way of capturing these higher level mechanisms in an AH is to introduce an new AH with this specific system. Other potential solutions are to split the AH in several sub-abstraction hierarchies as was presented by (Bisantz and Burns, 2008)(Ho and Burns, 2003). Or to organize the AHs by using levels of sophistication and nested AHs as was introduced by Amelink (2010). This is a relative cumbersome process compared to adding a functional block in the FAN. Furthermore, by adding only a single functional node instead of a full abstraction hierarchy with five layers the increase in size can be minimized.

Ultimately, these unspecific goals come from the need to define a clear system boundary in advance, which is basically the first challenge that one comes across when mapping a complex work domain. Defining the boundary is an crucial step of the work domain modeling. In many examples a single isolated system is the boundary, for example the power plant or the aircraft (Dinadis and Vicente, 1999). This choice simplifies the AH drastically and makes it practical tool. However, a variety of systems are involved with in-flight operational decision-making; i.e., the aircraft itself, the physical world with weather, other traffic, airport infrastructure, passengers, laws and regulations. And whether objects can be controlled by the flight crew have to be included in the boundaries of the system (Burns and Hajdukiewicz, 2004). All these systems will in turn expand the AH significantly.

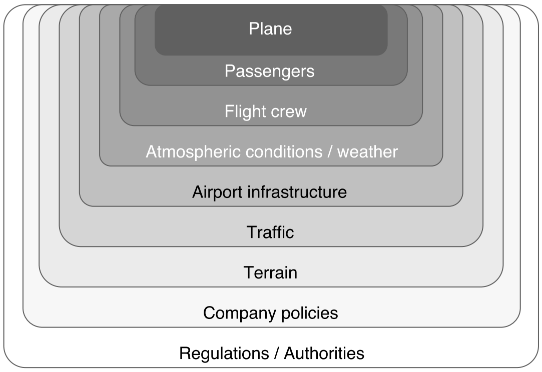

Fig. 3. Various system boundaries for commercial flight operations.

Modeling Efforts Another issue with the abstraction hierarchy that has become apparent is the issue of circular means-ends. This can be explained with the following example. If the work domain has the functional purpose of flying from A to B safely and efficiently (see Fig 1). And the abstract layer contains for example; conservation of energy, separation, compliance to regulations, customer satisfaction and time of arrival. Efficiency is achieved for a company, if a customer is satisfied. But how can customer satisfaction be achieved? Only if they arrive on time, safely and comfortably at the destination they bought the ticket for. This suggests that a link exist between a lower function to a higher function and on the same level. We see that we have a problem with the hierarchy, since a hierarchy can only flow down and not up, as being presented with the dashed line in Fig 1. This issue was also noted by Lind (2003). This problem can be resolved by using a network instead of a hierarchy, which is exactly what was done in the FAN. Circular or recursive links are feasible in the FAN.

Further, modelling effort with the abstraction hierarchy is found to be quite high mainly due to the abstract layer. The step from functional purpose to abstract function is often too big of gap, too broad or too high level. Lind (2003) expresses it as follows *”How can abstract functions be seen as causing the purpose or goal to be realized and how can generic functions cause the abstract functions?”*. This layer makes sense if one evaluates or designs a display for systems in which the laws of physics are the start of a design. However, if experts are asked with the ’how’ rule to extract the means to ends, few will mention that they think of thermal laws as a second layer. Many take these abstract function for granted and only mention the generalized functions. The abstraction layer for non-physical systems is difficult to define. Once money, comfort, satisfaction or compliance with regulations are abstract principles this layer becomes difficult to define. Reising et al. (2002) and Naikar (2008) adopted a different set of labels for the various levels to make the abstraction hierarchy suitable for intentional systems. The label that replaces the abstract function, i.e., priority measures, makes the abstraction hierarchy more pragmatic. But the mix between priority measures and laws of physics can be confusing. Also, by putting Newton’s laws or thermal laws the risk is that this layer becomes cryptic and high level with the results that it loses its meaning. Therefore, some heard feedback on the EID methodology is that it is too theoretical or too academical (Bennett and Flach, 2011).

The AH contains five levels, which are sometimes reduced to four depending on the preferences of the analyst (McIlroy and Stanton, 2015). However, this fixed number of levels in the abstraction hierarchy is observed to be restrictive, as was noted by Lind (2003). It becomes difficult to fit all the functions into five layers because of the means-end relations, particularly if multiple systems are integrated. Since functions on one level provide the rationale or reason for the existence of functions on the lower level, the five means-end links may become a limiting factor. This problem doesn't hold for the FAN, because as many functional nodes can be added as desired.

The last observation found on modeling effort is the question of how to model the system in certain states, or phases in time. Distinct periods in a series of events also influences the goals and functions of a operator drastically like the definition of the system boundary. Different goals are apparent in various flight phases. Take for example landing and cruise flight. In cruise flight the landing gear is not used, while for landing this is crucial. Taking the goals of the various flight phases into consideration is important for the entire operation. Some AH modeling approaches describe these phases as functions (Ho and Burns, 2003). While they are in fact periods of times that change specific goals, which cannot are not captured by the functions. While, a specific phase can be captured as a process in the FAN. Providing more flexibility to model these parts.

Clarity Various examples of abstraction hierarchies (Ho and Burns, 2003; Naikar, 2008; Stanton et al., 2016) result into a chaotic diagram with entangled links for relatively small systems. For the application of commercial flight operations, we assume that this issue might worsen due to the size of the analysis. A lot of time would be spent in unraveling the links. With the FAN, the links between the functional nodes and processes are more specific and are placed directly into a certain context, which makes them easier to understand. The entangle of links can be resolved with the different variations of the abstraction hierarchy, but these techniques make it difficult obtain a overview of the links. As can be seen in (Bisantz and Burns (2008) page 58). Therefore, for clarity proposes the FAN is preferred. In which ordering can be done more flexible.

5.2 *Translating into requirements into actual representation*

The above-mentioned observations were about building the knowledge model. However, this model is only useful if it can be transformed into a visual form. Burns and Hajdukiewicz (2004) describe that the first step in transforming the model to a visual form is to extract the variables from the different levels of the model. However, this process is seems relatively unsupported and is based on the analysts intuition. Many of the variables seem to be determined subjectively and are 'nice to have'. No strict guidelines are presented what variables should be included. The risk here is that information is included that is not specific for the system, which can lead to clutter and compromises the effectiveness of the interface.

ACWA has the advantage that for each functional node, cognitive work requirements have to be determined. These are determined by analyzing what cognitive work an operator has to perform to reach the goal in that specific block, e.g., monitoring, transforming data, making decisions. The benefit of this approach is that the determination of variables remains linked with the process and functional nodes to reach the purpose of the system. Furthermore, the sequential steps, IRR, RDR and PDC are all traceable and based on the FAN knowledge model. This with the goal to eliminate unnecessary information and clutter. Although, the SRK taxonomy is mentioned as a stand-alone tool and integrated part of EID, few designs are using it in the design of the displays McIlroy and Stanton (2015). This might be the result of how it is formulated in the methodology. There is no step-wise method to determine on how this SRK taxonomy is implemented, and many see the SRK as a recommendation or good practice.

6. DISCUSSION

EID is presented above as a structured process to go from a model to a visual representation. However, concerns exist on this process since much of the end-result depends on the practitioners experience with the methodology. Also, the abstraction hierarchy shows some issues for multiple stakeholder, complex socio-technical systems.

The EID process seems to be semi-structured and more focused on obtaining the knowledge model than on transforming the findings into a visual form. The SRK taxonomy states that the model should be reflected in the interface, but a structured process on how to do this is not present. Secondly, the abstraction hierarchy is a theoretical useful tool. However, in practice it seems to be too restrictive for larger systems than isolated process-control systems due to the fixed five levels, difficulties in defining the abstract function level and circular means-ends relationships.

ACWA provides a structured step-wise approach to transform the knowledge model into an interface. Furthermore, the FAN is less restrictive in modeling the work domain. This seems to be beneficial since the fixed layers and the circular means-end issues can be resolved. This will lead to more complete models, that can easily expanded. Hence, it is suitable for incremental design approaches.

Despite the benefits of ACWA, only a few publications with examples (Elm and Potter, 2003) (Potter et al., 2003) were published since its introduction in 2003. Several books (Bisantz and Burns, 2008) (Hollnagel, 2008) (Stanton et al., 2005) publish the method. Elm and Potter (2003) state that ACWA has been applied across a wide range of domains. However, proof that this methodology actually works is unfortunately not present. However, the methodology is based on experimentally proven work of Rasmussen, Hollnagel, Woods and Lind and is derived from well-known system engineering practices.

7. CONCLUSIONS

Operational decision making in commercial air transport has specific characteristics: an unstable, open environment with many stakeholders, various flight phases, and different types of constraints, i.e., intentional and causal. This work domain was found to be easier captured by the Functional Abstraction Network as part of the ACWA. Relative to EID, it features a less restrictive knowledge model analysis and a step-wise process to derive representation requirements. This has the advantage to eliminate subjective arguments about what should be presented and what not. ACWA will still lead to an ecological interface, since it is based on a knowledge model that represents the abstract, functional envelope of a work domain. Therefore, although both methods are quite similar, we found ACWA to be a more applied, more pragmatic and structured methodology to analyze and transform a model into an effective visual representation for real world applications.

REFERENCES

Amelink, M.H.J. (2010). *Ecological Automation Design: Extending Work Domain Analysis.* Ph.D. thesis, Faculty of Aerospace Engineering, Delft University of Technology. doi:10.1111/j.1539-6924.2011.01605.x.

Bennett, K.B. and Flach, J.M. (2011). *Display and Interface Design: Subtle Science, Exact Art.* CRC Press, Inc., Boca Raton, FL, USA, 1st edition.

Bisantz, A.M. and Burns, C.M. (2008). *Applications of Cognitive Work Analysis.* doi:10.1201/9781420063059.

Borst, C., Flach, J.M., and Ellerbroek, J. (2015). Beyond Ecological Interface Design: Lessons from concerns and misconceptions. *IEEE Transactions on Human-Machine Systems*, 45(2), 164–175. doi: 10.1109/THMS.2014.2364984.

Burian, B.K., Barshi, I., and Dismukes, R.K. (2005). The Challenge of Aviation Emergency and Abnormal Situations. *NASA Technical Memorandum*, (June), 1–21.

Burns, C.M. and Hajdukiewicz, J. (2004). *Ecological Interface Design.* CRC Press.

Dinadis, N. and Vicente, K.J. (1999). Designing Functional Visualizations for Aircraft Systems Status Displays. *International Journal of Aviation Psychology*, 9(3), 203–223. doi:10.1207/s15327108ijap0903.

Elm, W.C. and Potter, S.S. (2003). Applied Cognitive Work Analysis: a Pragmatic Methodology for Designing Revolutionary Cognitive Affordances. *... Cognitive Task Design*, 1–40.

Ho, D. and Burns, C.M. (2003). Ecological Interface Design in Aviation Domain: Work Domain Analysis of Automated Collision Detection and Avoidance. *Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting*, 119–123.

Hollnagel, E. (2008). *Handbook of Cognitive Task Design.* doi:10.1201/9781410607775.

Hollnagel, E. and Woods, D.D. (2005). *Joint Cognitive Systems. Foundations of Cognitive Systems Engineering.* doi:10.1177/106480460701500208.

Lind, M. (2003). Making Sense of the Abstraction Hierarchy in the Power Plant Domain. *Cognition, Technology & Work*, 5(2), 67–81. doi:10.1007/s10111-002-0109-4.

McIlroy, R.C. and Stanton, N.A. (2015). Ecological Interface Design Two Decades On: Whatever Happened to the SRK Taxonomy? 45(2), 145–163.

Naikar, N. (2008). *Work Domain Analysis: Concepts, Guidelines, and Cases.* CRC Press.

Norman, D.A. (1988). *The Design of Everyday Things*, volume 16. doi:10.1002/hfm.20127.

Potter, S.S., Gualtieri, J.W., and Elm, W.C. (2003). Case studies: Applied Cognitive Work Analysis in the Design of Innovative Decision Support. *Handbook of Cognitive Task Design*, 653 - 678(September 2014).

Potter, S.S., Roth, E.M., Woods, D.D., and Elm, W.C. (2000). Bootstrapping multiple converging cognitive task analysis techniques for system design. *Cognitive Task Analysis*, (February 2017), 317–340.

Rasmussen, J. (1985). The Role of Hierarchical Knowledge Eepresentation in Decisionmaking and System Management. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(2), 234–243. doi: 10.1109/TSMC.1985.6313353.

Rasmussen, J., Pejtersen, A., and Goodstein, L. (1994). Cognitive Systems Engineering. *"Ergonomics: The history and scope of human factors"*, (October).

Reising, D.V.C., Jones, B.G., Sanderson, P.M., and Moray, N. (2002). List of Publications & Invited Conference Contributions.

Shappell, S., Detwiler, C., Holcomb, K., Hackworth, C., Boquet, A., and Wiegmann, D.A. (2007). Human Error and Commercial Aviation Accidents: an Analysis Using the Human Factors Analysis and Classification System. *Human factors*, 49(2), 227–242. doi: 10.1518/001872007X312469.

Stanton, N.A., Harris, D., and Starr, A. (2016). The future flight deck: Modelling dual, single and distributed crewing options. *Applied Ergonomics*, 53, 331–342. doi:10.1016/j.apergo.2015.06.019. URL `http://dx.doi.org/10.1016/j.apergo.2015.06.019`.

Stanton, N.A., Hedge, A., Brookhuis, K., Salas, E., and Hendrick, H. (2005). *Handbook of Human Factors and Ergonomics Methods.* doi:10.1201/9780203489925.

Van Paassen, M.M., Borst, C., Ellerbroek, J., Mulder, M., and Flach, J.M. (2018). Ecological Interface Design for Vehicle Locomotion Control. *IEEE Transactions on Human-Machine Systems*, 48(5), 541–555. doi: 10.1109/THMS.2018.2860601.

Vicente, K.J. (2002). Ecological Interface Design: Progress and Challenges. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(1), 62–78. doi:10.1518/0018720024494829.

Vicente, K.J. and Rasmussen, J. (1992). Ecological Interface Design: Theoretical Foundations. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(4), 589–606. doi:10.1109/21.156574.

Woods, D.D. and Hollnagel, E. (1987). Mapping Cognitive Demands in Complex Problem-Solving Worlds. *International Journal of Man-Machine Studies*, 26(2), 257–275. doi:10.1016/S0020-7373(87)80095-0.

Xu, W. (2007). Identifying Problems and Generating Recommendations for Enhancing Complex Systems: Applying the Abstraction Hierarchy Framework as an Analytical Tool. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49(6), 975–994. doi: 10.1518/001872007X249857.