A Design Methodology of Optimized Diagnosis Functions for High Lift Actuation Systems

Christian Modest[1], Frank Thielecke[2]

[1,2] *Institute of Aircraft Systems Engineering, Hamburg University of Technology, Germany*
christian.modest@tuhh.de
frank.thielecke@tuhh.de

ABSTRACT

This paper presents a model-based approach to the optimal design of diagnosis system architectures for complex high lift actuation systems. The overall approach consists of two steps. In the first step, safety and reliability related requirements are introduced. These focus on the detectability and isolability of faults. Symptoms are used therefore. These are separated into safety and reliability related symptoms. In the second step, different alternatives to provide the symptoms are drawn and evaluated in order to gain an optimal design solution. A two stage analysis process is used therefore. The first stage focuses on the fulfillment of the safety related requirements whereas the second stage concentrates on the reliability related requirements. All aspects of the analysis are depicted exemplary and formalized theoretically. The results of the application to the high lift actuation system of an Airbus A340-600 aircraft are presented afterwards and discussed in the end.

1. INTRODUCTION

The members of the air transport system compete in a global, steadily growing market. Airlines, manufacturers and maintenance, repair, overhaul (MRO) providers are continuously forced to strengthen their competitive edges and improve business in order to stay profitable. One point they all have in common is to provide a safe operation and a high availability of the aircraft at a minimum of total cost. The development and use of efficient health management technologies can thus be seen as an attribute for the diversification and consolidation of the own position inside a global competitive environment.

New technologies tend to a strong increase in complexity in order to meet diversified customer and environmental requirements. One aspect of current research is to raise the functionality of high lift systems on commercial airplanes in order to improve the overall aircraft performance. A decoupling of surfaces and the use of decentralized drive units state possible concepts. While developing adequate solutions, it has to be considered that new technologies need not only to improve functional aspects of the aircraft but have also to be operated and maintained in a capable manner in order to reap all benefits. The consideration of efficient health management technologies is thus indispensable.

The focus of this paper is on a model-based approach to the optimal design of diagnosis system architectures for complex high lift actuation systems (HLS). Section 2 gives an overview of current HLS and their essential components. General challenges in developing adequate diagnosis systems (DS) and a systematic, model-based approach for the design and test of DS are depicted in Section 3. The first two steps are focused in this paper. These concern the definition of requirements and the conceptual design phase. The importance of these steps and a concept for a systematic, requirement based design procedure are depicted in Section 4.

The development of a diagnosis model is presented in Section 4.1. The identification and formalization of requirements is depicted afterwards. Section 4.2 demonstrates how safety and reliability related requirements are assigned that the design of the DS has to meet. This concerns the detectability and isolability of faults by means of symptoms. In order to identify optimal symptoms, with respect to various objectives, different alternatives for respective monitoring and sensor devices are drawn and implemented in a diagnosis model. A simulation of fault modes is then carried out and cause-effect relationships are gained. These are evaluated in a two stage process that is depicted in Section 4.3. The first stage describes how minimal architectures, that are adequate to fulfill the safety requirements are gained. Afterwards it is addressed in the second stage how additional monitoring devices are chosen in order to meet also reliability related requirements. Both steps are depicted by means of examples and specified theoretically. The results of the application to

Christian Modest et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

the HLS of an Airbus A340-600 aircraft are presented in Section 5. A discussion of related work is given in Section 6. In the end, Section 7 concludes and gives an outlook about open points and future research activities.

2. Fundamentals of High Lift Systems

Commercial airplanes are equipped with high lift devices to augment lift at low speed during takeoff and landing. Today, those systems primarily consist of a mechanical transmission shaft system that transmits mechanical power from a centralized hydraulic power control unit (PCU) to rotary actuators at each wing half. The actuators are located alongside the transmission system and deploy the high lift surfaces synchronously.

Figure 1 depicts a typical high lift actuation system at the trailing edge of a commercial aircraft. This flap system consists of a mechanical shaft system that is powered by a PCU. The mechanical power is transmitted to five down drive stations on each half of the wing via shafts, joints and gearboxes. Each of the drive stations consists of a down-drive gearbox and shaft, an input gear box with torque limiter, a cross shaft and rotary actuator. The inner surface has two drive stations whereas the outer surface is moved by three.

In case of safety-critical failures a wing tip brake (WTB) can hold the system and inhibit movement. In order to monitor for such conditions and control the overall system two Slat-Flap Control Computers (SFCC) are used. The monitoring and control is done using information from discrete and analogue sensors like position pick off units (PPU).

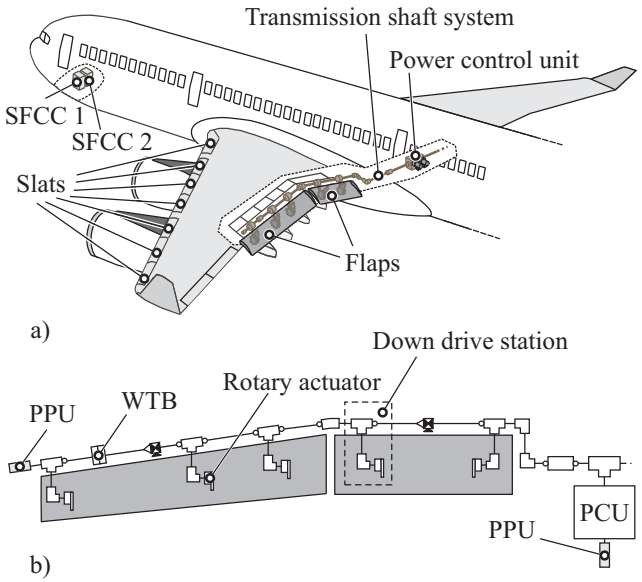


Figure 1. Typical high lift actuation system for a commercial aircraft.

3. Challenges and Model-based Approach

The task of a diagnosis system consists in the detection of abnormal functional conditions (AFC) and the isolation of potential root causes. Based on the criticality of the specific AFC and the underlying fault a decision is made afterwards. This decision can result in an abnormal shut down of the overall system or just in an indication for maintenance. In order to detect and isolate the AFCs and take adequate measures a process chain is used, that consists of different steps. Figure 2 depicts an overview of this chain. In the first step specific features are extracted from measurements on a system level. For the HLS a feature can be a too high position difference between the output of the PCU and one of the PPUs at the transmission ends. The logical combination of different features then leads to the detection of the symptom of an AFC. In case of a high criticality an abnormal shut down of the PCU and the setting of the WTBs results as an action, whereas in other cases a degraded operation is still possible.

The development of the previously mentioned diagnosis functions for HLS is done today primarily empirically or as an after-thought due to in-flight incidences. Considering the complexity of new HLS as presented in (Lulla, 2011) and (Recksiek, 2009) the empiric approach results in a costly and laborious process. Furthermore, gaining optimal functions while considering different design objectives is hardly possible. Thus, the development of diagnosis functions has to be dealt with systematically and traceable already during the HLS development process. A model-based approach for the design and verification of a DS for HLS is proposed therefore and introduced in the following.

The proposed approach is embedded into the general aircraft systems development process. This process is typically divided into several steps that are arranged in the common V-model(Haskins, 2006). The left branch of this model designates the overall system design whereas the right branch marks the overall system test.

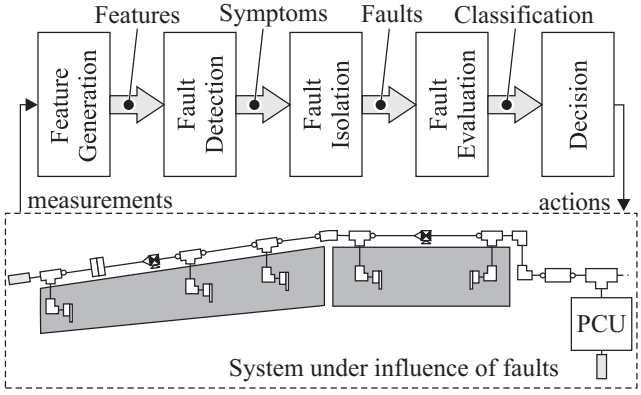


Figure 2. Diagnosis functions for a high lift actuation system.

In the first step of the design phase the system requirements are developed, based on the aircraft level requirements. In order to realize these requirements, the system architecture and all involved components are designed. At the bottom of the model the components are built as hardware. On the right branch tests are performed to verify that the design meets the requirements. This is done first separately for the components and afterwards in an integrated environment for the complete system. If requirements are not met or failures are identified during the test procedures a correction of requirements or redesign could be necessary.

Figure 3 depicts an approach to integrate the development of a diagnosis system into the V-model.

The design and test of the DS should begin early and in parallel to the overall system development process. This should be done in order to avoid failures and unnecessary elements in the functional specifications. The DS should be designed strictly according to requirements. These can be safety, reliability and performance related. For this task models of the different components are used as executable units to validate each design step. At the bottom of the V-model executable code is generated. This can be C-code for monitoring devices and xml-data files for built in test (BIT) specifications.

The model-based test is used for the verification of single and combined applications with respect to the functional requirements. In the first step of the test procedure this is done separately for the components. Examples for this step are the test of stimulated monitoring devices or the test of a knowledge base to verify that BIT requirements are met. In order to test the complete DS in interaction with the physical system a virtual integration approach is used. For this step, models that have already been used for the design and optimization of the system architecture and specific components, are extended in order to be usable for simulations together with the complete software system. This includes the DS and all the control applications.

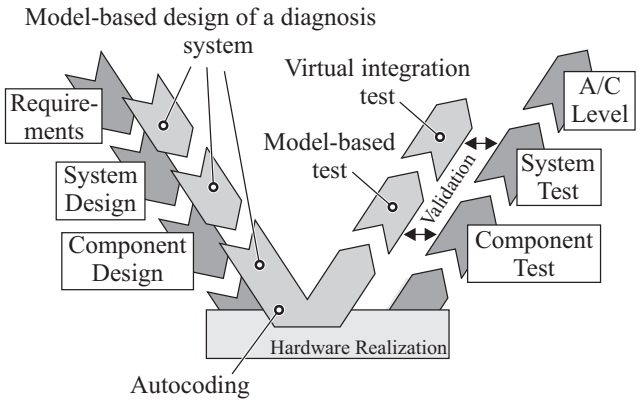


Figure 3. Framework of a model-based development process for diagnosis systems.

The approach of an integrated simulation environment enables the identification of failures, which are caused by the physical interaction of the system components combined with all the software systems.

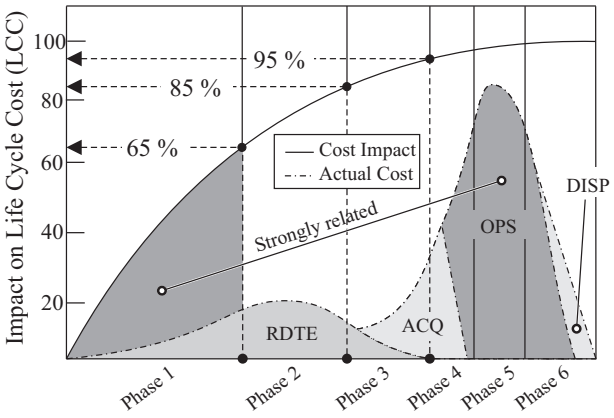
Following the approach for the design and test of the DS a simulation of the complete system is possible before qualified code is generated and implemented on final hardware platforms and further hardware devices are chosen. Thereby an early identification and correction of failures in the functional specifications can be made that should lead to a significant reduction of the overall development costs.

This paper deals with the optimal design of diagnosis system architectures for complex high lift actuation systems. The basic aspects of the integration of all the involved DS instances and the execution of preliminary verification tests are presented in (Modest, Schories, et al., 2011) and (Modest, Grymlas, et al., 2011) for the application to multi functional fuel cell systems.

4. Design of a diagnosis system architecture

The first phases of the overall system development process have an important impact on the product's total life cycle cost (LCC). The LCC includes cost for research, development, test and evaluation (RDTE), acquisition (ACQ), operation and support (OPS), and the final disposal (DISP).

Figure 4 depicts the impact of the different aircraft program phases on the LCC and illustrates where the actual costs occur (Roskam, 2006).



Phase 1 : Planning and Conceptual Design
Phase 2 : Preliminary Design and System Integration
Phase 3 : Detail Design and Development
Phase 4 : Manufacturing and Acquistion
Phase 5 : Operation and Support
Phase 6 : Disposal

Figure 4. Impact of different aircraft program phases on the life cycle cost.

The biggest part of the LCC occurs during the aircraft operation and support whereas the cost for RDTE are comparatively small. The impact of RDTE in general and the planning and conceptual design phase in particular on the cost for OPS are huge though. The first program phase accounts for an impact of 65% on the LCC. This phase has thus to be dealt with in a careful and systematic way in order to prevent high cost that may result from belated, but necessary changes in the aircraft system and the respective DS due to in-flight incidences. Examples are, that additional sensors are needed in order to detect certain failure conditions or to support the system's troubleshooting for the case that faults lead to high system downtimes. In order to prevent such conditions a systematic and traceable design procedure for a DS is necessary.

The main tasks of the DS are the detection of all abnormal functional conditions, which are here related to their safety impact, and the provision of distinct information for an efficient troubleshooting. A general design procedure for a DS that assures that these requirements can be met during the operation is presented in the following sections. The focus is on the conceptual design of a DS architecture.

Figure 5 depicts a general overview of the proposed design procedure. In the first step, requirements that the design has to meet are analyzed and defined. At the current stage these are related to safety and reliability. The first one focuses on the detection of symptoms of safety critical failure conditions whereas the latter one concentrates on the fusion of symptoms in order to identify root causes. An important aspect is thus the definition of safety critical failure conditions on a system level. As a guideline, the SAE ARP 4761(SAE, 1996) is used.

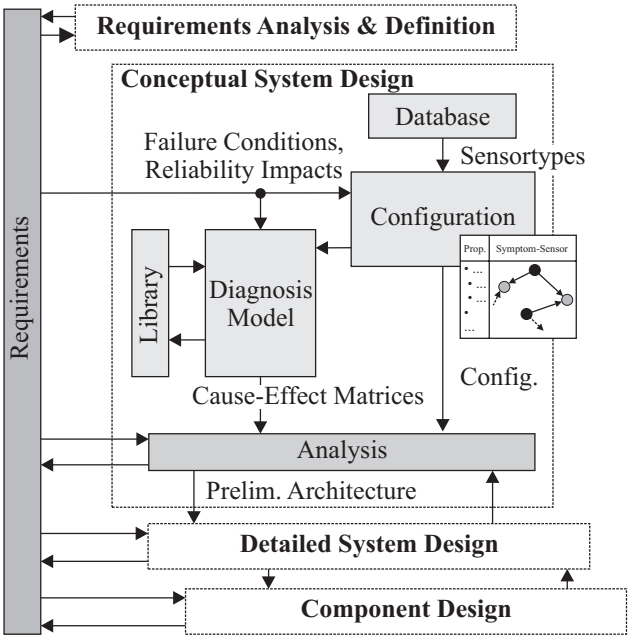


Figure 5. Overview of the design phase of a DS for HLS.

Table 1 depicts three examples of failure conditions on a system level of a typical high lift actuation system. These conditions are numbered and sorted according to their severity. The classification *MIN* stands for minor and *CAT* for catastrophic. In the first case a degraded operation of the system might still be possible whereas in the latter case an abnormal shut down of the overall system has to be commanded. In order to enable the appropriate system reaction these two conditions have to be separated. A prerequisite is, that in the general case the symptoms of all FCs are analyzed properly and adequate monitoring devices for the detection are chosen.

This paper focuses on the optimal design of diagnosis systems. In order to gain these optimal design solutions with respect to various objectives, different alternatives to provide features and detect symptoms for each failure condition have to be considered. In order to clarify this point the FC *Asymmetric Flap Movement* is used as an example. The effect of such a failure condition might be an uncontrollable roll moment on an aircraft level which in turn can lead to a total loss of the aircraft. In order to identify this condition and take adequate measures it has to be analyzed in the first step how respective symptoms look like. One such symptom might be a position difference that exceeds certain limits. This difference can be taken between the left and right hand transmission system or separately between each of the transmission halves and the PCU. A third alternative might consist in taking features from position measurements at each pair of rotary actuators. It is obvious that all the alternatives differ in various objectives. While the first one uses only a minimum of information the other alternatives would need more features and by that cause more effort. However, this simple evaluation holds only for one objective and in general every alternative has to be evaluated in the overall design context. This means with respect to reliability related requirements and other requirements that might be defined in further design steps.

The way, that has been depicted for one failure condition, is now repeated carefully for all the failure conditions that have to be considered. A general overview of the manner in which failure conditions are linked to monitoring devices that again are linked to sensor devices is depicted in Figure 6.

FC Ref.	Title	Class.
001	Flaps operate with reduced rate	MIN
...		
004	Loss of Flap Operation	MIN
...		
011	Asymmetric Flap Movement	CAT
...		

Table 1. Failure condition summary list.

In general, the presented approach leads to design solutions that can effectively fulfill the safety requirement with respect to various criteria. However, the most optimal solution may not be adequate to also fulfill other requirements like the reliability related one. Considering the previous example, this means that only some of the alternatives would allow to isolate potential root causes on different levels of detail. While the first alternative does not allow for any isolation, the second and third one would allow for an isolation on a system or subsystem level. Consequently, the first alternative would have to be extended in order to meet further requirements. Therefore, maintenance conditions (MC) are introduced next.

The MCs are defined to be conditions that are not directly linked to specific FCs. Examples are the conditions "*high position difference on right hand system side*" or "*low load at left hand PCU side*". Both the conditions state possible extensions of the first alternative of the previous example and might be used to meet certain reliability related requirements, like an isolation between left and right hand system side. In general, features that are needed to detect symptoms of MCs can be gained from measurements that are also linked to FCs.

According to the way proposed before, a set of alternatives for the detection of symptoms and the respective sensors for FCs and MCs are defined. This set is then implemented in a diagnosis model. A simulation of all component fault modes is carried out and the respective symptoms in form of monitor flags are stored and transfered into cause-effect matrices. An example is depicted in Figure 7.

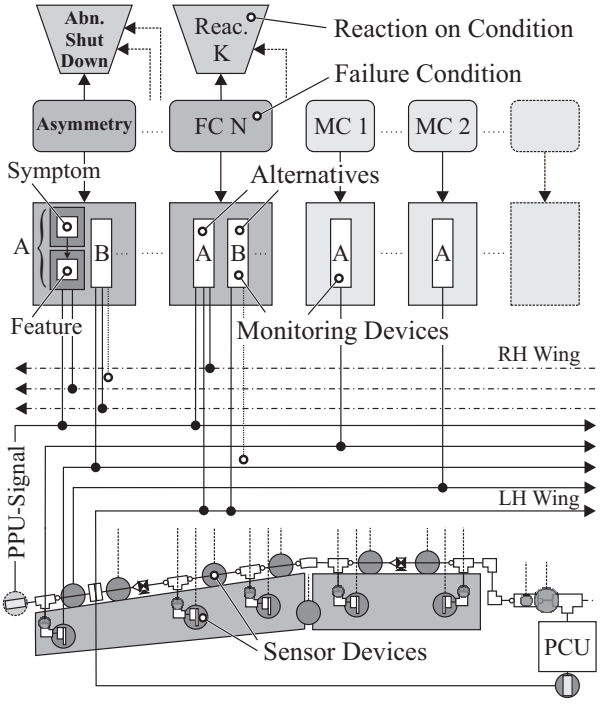


Figure 6. Placement of sensors and concepts for monitoring.

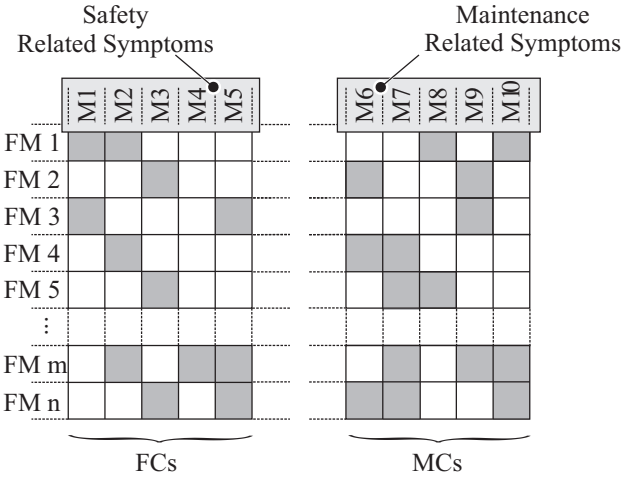


Figure 7. Result of the simulation of dedicated fault modes.

The cause-effect matrices are used as an input for a two stage analysis process. In the first stage, the safety related symptoms are evaluated in order to gain solutions for the fulfillment of a safety requirement. In the second stage, the fulfillment of additional reliability related requirements is checked and if necessary complementary maintenance related symptoms are identified for an extended design solution. In order to identify the overall optimal solution, cost factors for the placement of sensors are considered. The relation between symptoms and sensors is defined as a directed graph, where each sensor node has a cost property. In the end of the overall design procedure a globally optimal solution is gained, that consists in the definition of symptoms, features and measurements.

The entire proceeding is explained in the following. In the first part the development of a diagnosis model is shown. Afterwards, a theoretical definition of the requirements is given and in the third part the analysis process is depicted.

4.1. Development of a Diagnosis Model

The proposed approach uses on an a-causal, component-based, quasi-static model of the high lift actuation system. As a simulation environment the tool RODON is used (Bunus et al., 2009). The overall model is built-up of different layers where each of the layers states one level of hierarchy. This approach is used to effectively handle changes that will naturally occur during the overall system development process and especially during the early phases.

The model's top-layer presents the system level where the interaction between all involved subsystems and the interfaces to other systems, e.g. the electrical power supply, are defined. Subsystems of a typical HLS are the PCU, the right and left hand transmission system and two SFCCs. Each of these subsystems is built-up of components that are defined at

the component level. In case of the SFCC these components are a control and a monitoring part whereas for the transmission system these are shafts, gears and actuators amongst others. Figure 8 depicts an excerpt of the component level of the left-hand transmission system. It is shown the inner flap surface and dedicated mechanical components. The surface is deployed by means of two rotary actuators that in turn are operated by several shafts and gears.

The lowest layer of the model is the constraint level where the specific behavior is defined. A multi-step modeling approach is used therefore. Aspects of that approach have been introduced in (Modest, Schories, et al., 2011). In order to illustrate this approach the example of a gear is used. This gear is located on the lowest component level of the left hand transmission system that has been depicted previously. Figure 9 states the exact position in the second down drive of the inner flap surface. The purpose of the gear is to reduce the transmission's rotational speed. In order to build a model of this component all the interfaces are defined in the first step. A specific domain is used for that. For mechanical components the domain consists of the rotational speed as a potential quantity and the torque as a flow quantity. The gear has one input and one output so that two interfaces are needed. As a result there is a model-shell.

In the second step all parameters are added to the model-shell. In case of the gear these can be the gear ratio, the efficiency factor and the diameters of all cog wheels, depending on the level of detail. Afterwards in the third step all the variables are defined. These can be the rotational speed difference or the power dissipation amongst others. All the variables and the parameters are linked in the fourth step. As it can be seen from Figure 9 the behavioral section is divided into two main parts with two sets of constraints. The first part contains the set of constraints that are supposed to be fix.

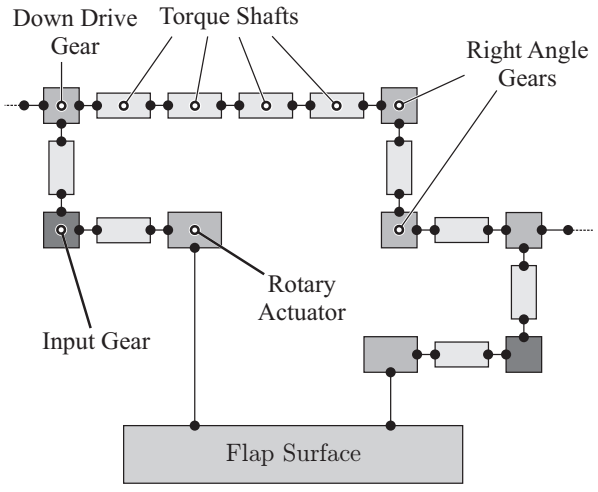


Figure 8. Excerpt from the component level of the transmission model.

The second part contains the set of constraints that are defined to be switchable. An example for a constraint that is supposed to be fix is the rotational speed difference. This is defined as the speed difference Δn between both the speeds n_1 and n_2 at the interfaces under consideration of the gear ratio $i : \Delta n = n_1 - n_2 \cdot i$. The value for the speed difference though is defined as a switchable constraint. In the example there are two of these constraints which relate to two specific fault modes. In general these include the nominal mode. Following the example the speed difference is defined to be zero, $\Delta n = 0$, in the nominal mode whereas in the second mode it is undefined which relates to a rupture. For the torque M_1 and M_2 it holds $M_1 \cdot i + M_2 = 0$ in the nominal and the second mode, whereas in the second mode M_1 is set as a parameter, $M_1 = 0$, so that there is zero torque at both the interfaces. A third mode that is not part of the example is a jam of the gear. The way in which this mode is defined is analog to the example.

According to the way, that has been depicted, all the components of a typical high lift actuation system are modeled. This includes mechanical, hydraulic and electrical components as well as the controller and monitoring part of the software systems. The focus of modeling fault modes though lies on the non software systems. All components are then grouped in a library and used to form a model of the specific system under consideration. In this paper it is the Airbus A340-600 flap actuation system. The architecture is similar to the one that has been depicted in Figure 1.

The validation of the behavior in normal operation according to functional requirements has been done but is not the scope of this paper.

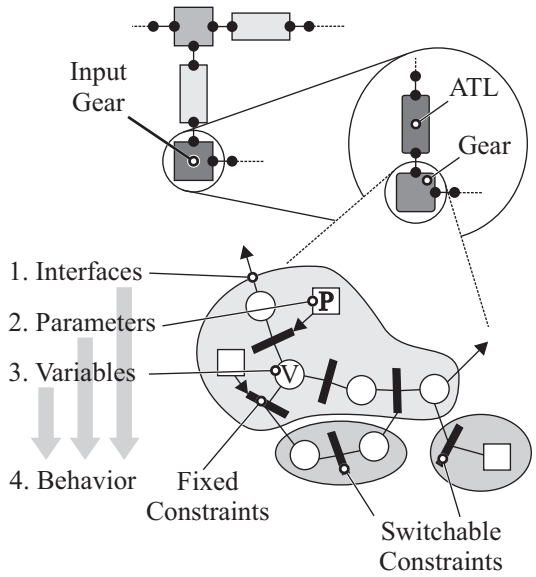


Figure 9. A multi-step modeling approach for a component model applied to a gear.

4.2. Definition of requirements

The planning and conceptual design phase for the development of a DS focuses on the analysis of optimal system architectures with respect to the detection of symptoms of safety critical failure conditions (FC) and the isolation of potential root causes. In the following, two requirements for both aspects are introduced in detail. The first one is related to the symptom-detection of FCs and has previously been mentioned as safety related whereas the second one focuses on the isolation task and has previously been mentioned as reliability related. According to the general proceeding of detection followed by isolation both requirements are introduced consecutively in the following.

In the first step the diagnosis system is in charge of the detection of symptoms of FCs. These are defined in accordance with (Isermann, 2006). Related to that, a safety requirement is defined here, such that all fault modes that map to symptoms of FCs shall be detected using a minimal amount of symptoms with a minimal overlap. This is formalized in the following definition.

Definition 4.1 (Safety Requirement) *A safety requirement (SR) is defined such that a set of fault modes that map to symptoms of FCs are detectable with a minimal amount of symptoms that have a minimal overlap:*

$$SR := \{fm \mid \exists\, m \in M : fm \mapsto m \wedge \mathbf{min}(\cap m_i) \forall m_i \in M^*\} \ .$$

The following holds in the definition:

- *m is a symptom that is sensitive to a specific fault mode ,*
- *M is a set of symptoms that are sensitive to any fault mode ,*
- *M^* is a minimal set of symptoms that have a minimal overlap .*

The safety requirement states that all fault modes *fm* that are related to *M* shall be detected using M^*. An example for that is the consideration of two different fault modes. An analysis showed that fault mode one maps to the symptoms A and B whereas fault mode two maps to the symptoms B and C. The safety requirement is fulfilled by two solutions. The first one consists of the symptom B whereas the second one consists of the symptoms A and C. Both solutions use a minimal amount of symptoms to detect all relevant fault modes. The capability to infer the root cause is different though.

In the second step the diagnosis system is in charge of fusing different symptoms to isolate potential root causes. Compared to the definition of the SR it is not required here to isolate between every considered fault mode but according to specified isolability requirements. In order to clearly define this concept some definitions are made in advance. The first one is the definition of isolability items. These can focus on specific fault modes, components and system parts.

Definition 4.2 (Isolability Item) *An isolability item (ii) is a triple (sp, comp, fm) where:*

1. *sp, a system part, is an expression that indicates a location ,*
2. *comp, a component, is an expression that indicates a component ,*
3. *fm, a fault mode, is an expression that indicates a fault mode .*

The set of all isolability items $\cup \{ii_j\}$ is defined to be II. An example for an isolability item is the triple $LH \wedge Shaft001 \wedge Rupture$. In this expression *LH* stands for the flap system's left hand side, *Shaft001* stands for a mechanical component of the transmission system and *Rupture* is a specific fault mode. If it is unambiguous, parts of the formal ii can be omitted if necessary in order to define larger items that cover complete locations or components. An example is the item $PCU \wedge * \wedge *$ that is shortened by PCU. This covers all components and respective fault modes of the system part PCU.

The isolability items are next used to form isolability clusters where each cluster contains one or more items ii.

Definition 4.3 (Isolability Cluster) *An isolability cluster (ic) is a set of isolability items:*

$$ic := \{\cup \{ii_j\},\ ii_j \in II,\ j \leq |II|\} \, .$$

The set of all isolability clusters $\cup \{ic_j\}$ is defined to be *IC*. Extending the previous example of the isolability item, an isolability cluster ic can be the set $\{LH \wedge Shaft001 \wedge Rupture,\ PCU\}$. In combination with other clusters this means that every ii of the ic_j has to be isolated from all other $ii \in ic_k$. The set of clusters is thus used to finally define the isolability requirement.

Definition 4.4 (Isolability Requirement) *An isolability requirement (IR) is a set of disjunct isolability clusters ic such that:*

$$IR := \{ic_1, \ldots, ic_n\} \ , ic_n \in IC,\ 1 \leq n \leq |IC| \ ,$$
$$ic_k \cap ic_m = \emptyset \ , \forall\, ic_k, ic_m \in IR \ , k \neq m \ .$$

An example for an isolability requirement is the set:

$$\{\, \{LH \wedge Shaft001 \wedge Rupture,\ PCU\} \ ,$$
$$\{RH \wedge DownDriveShaft003 \wedge Jam\ \}\, \} \, .$$

The set illustrates the requirement, that all iis of the first cluster have to be isolated from the iis of the second cluster and vice versa in any failure condition. The requirement is fulfilled if this can be shown by analysis. Considering the example of the two fault modes that was introduced previously, the IR can here be defined such that both fault modes are to be isolated from each other. Regarding the solutions for the fulfillment of the SR, under the assumption that only single faults are considered, only the second solution, which are

the symptoms A and C, can fulfill the IR. The first solution, which is symptom B, has to be extended in order to fulfill both the SR and IR. The general procedure to gain optimal solutions that meet both the requirements is presented next.

4.3. Analysis

Design for the fulfillment of safety requirements

The result of the simulation of fault modes of the diagnosis model and the observation of the respective symptoms are two cause-effect matrices. In this section it is presented how the safety related matrix is used to gain solutions for a preliminary system design that meets the safety requirement. In order to illustrate the proceeding an example is used. The theoretical approach is defined afterwards.

Figure 10 depicts an example of a safety related cause-effect matrix. In this example two solutions are highlighted. These lead to sets of symptoms and respective monitors that are adequate to detect all safety critical component faults. The non-detectable faults are analyzed before. If they show a safety impact or have to be considered due to other reasons, more and different alternatives have to be defined and the analysis has to be repeated. Otherwise, these fault modes can be cleared from the list of all the considered fault modes.

The rows of the cause-effect matrix are used to form a *Symptom Set* (S). This is defined as the conjunction of symptoms m that are sensitive to a specific fault mode fm_i:

$$\mathrm{S}(fm_i) := \{m \mid fm_i \mapsto m \wedge m \in M\} \ .$$

The set $\mathrm{S}(fm_i)$ is thus the set of symptoms such that observing any symptom of the set gives detectability of the fault mode fm_i. Figure 10 depicts an example for that. Transferring the first row of the matrix into a symptom set gives $\mathrm{S}(fm_1) = \{M1, M2\}$. This says that observing the symptom $M1$ or the symptom $M2$ gives detectability of fm_1.

In order to provide detectability of all relevant fault modes intersections have to be built that hit each symptom set S at least once. Figure 10 depicts two possible solutions for that.

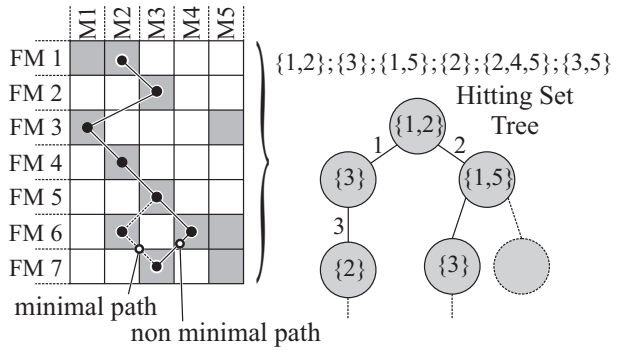


Figure 10. Calculation of minimal hitting sets based on a safety-related cause-effect matrix.

The solutions are different in the way that one of them is minimal whereas the other one includes more symptoms than those that are actually needed for the detectability of all fault modes. Comparing both the solutions the minimal one fulfills the safety requirement whereas the other one does not. In this simple example both solutions were found by manually traversing the rows and columns of the cause-effect matrix. In the general case though this is laborious and hardly possible. In order to find all minimal solutions in a systematic and efficient way the theory of minimal hitting sets (Reiter, 1987)(Greiner et al., 1989) is used. According to that, a minimal hitting set for a multitude of sets is a set that has a non-empty intersection with every set of the multitude of sets. It is thus exactly the set of symptoms M^* that fulfills the safety requirement.

In the depicted example there are six different symptom sets that are needed for the detectability of the specific fault modes. These are $\{M1, M2\}$, $\{M3\}$, $\{M1, M5\}$, $\{M2\}$, $\{M2, M4, M5\}$ and $\{M3, M5\}$. In Figure 10 these are shortened by using their indices. The theory of minimal hitting sets now gives two solutions that fulfill the safety requirement. These solutions are $M_1^* = \{M1, M2, M3\}$ and $M_2^* = \{M2, M3, M5\}$. In this context minimal means, that when removing one symptom there are specific fault modes that are no longer detectable.

The application to the high lift actuation system of finding minimal sets of symptoms M^* that fulfill the safety requirement gives 94 solutions. Details on that are depicted in the following sections. In the next step of this section the example is formalized.

The proposed approach of gaining solutions for the fulfillment of the safety requirement is defined in Algorithm 1. As an input the safety related cause-effect matrix CEM_{FC} is used that states the relation between specific fault modes and their safety critical symptoms. The respective symptom sets S are conjunct in $\mathcal{Z}$ that in turn is used as an input for a minimal hitting set algorithm. As a result all solutions for the safety problem are computed.

Algorithm 1 All minimal Solutions for Safety Problem

```
function DETECTION(CEM_FC)
    Z ← ∅
    for i = 1 → |CEM_FC| do
        s_i = S(CEM_FC(i))          ▷ CEM_FC(i) = fm_i
        Z ← Z ∪ {s_i}               ▷ add s_i to Z
    end for
    M* ← MinHittingSets(Z)
    return M*
end function
```

In order to get the *best* solution out off all solutions, criteria for the evaluation have to be defined. In the first step a cost criteria is considered. Cost can be applied to different aspects of the DS architecture. These can be cost for computation of

symptoms, cost for mounting and weight of wires or cost for hardware. The latter is used at the current stage. At this point hardware is referred to sensors. Every symptom is related to features that are gained using information from sensors. An example for that is the symptom *Asymmetry*. Features that are related to this symptom can be gained using information from two sensors, one at each half of the transmission system. In the simplest case this symptom would thus lead to cost of two unit cost. In the general case though, when calculating cost for symptoms, it has to be considered that in the physical system architecture there are already sensors that are needed for the system control. Using these sensors also for the system diagnosis task thus does not lead to additional cost. An example for that is a symptom that is related to information of only one such sensor. This symptom thus does not lead to any unit cost at the current stage of the proposed approach. Therefore, an important aspect is the relation between symptom and sensor. This relation is configured using a database of available sensors and is formed here as a graph. Figure 11 depicts an excerpt of such a graph that shows the relation between symptoms in form of specific monitors and the information they need in form of sensors.

Circles in the graph mark sensors that lead to a specific additional cost whereas the rhomb marks a sensor that leads to no additional costs. In the example the symptom set $M^* = \{M2, M3\}$ is active. Performing now a reachability analysis on the graph leads to a total cost of: $C_{tot.} = C_{S3} + C_{S10}$. At the current stage this will be two unit costs.

Applying the proposed approach of determining cost to all the sets that fulfill the safety requirement, enables to find the most cost effective one. However, it has to be considered that so far cost only includes cost for sensors. In the general case though, the most cost effective symptom set M^* that fulfills the safety requirement will not also fulfill the isolability requirement. How this is checked and how a design is gained that also meets the isolability requirement is depicted in the next section.

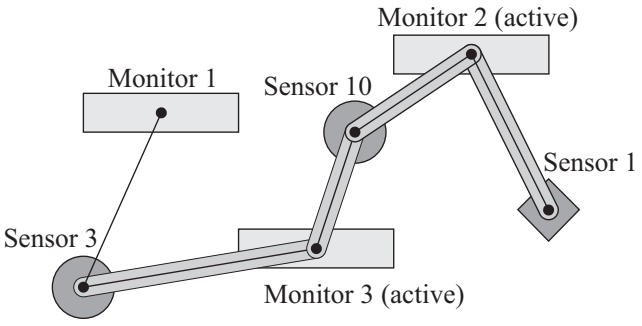


Figure 11. Example of a graph that shows the relation between symptom and sensor.

Extension of the Design for the fulfillment of isolability requirements

In the case that a symptom of a failure condition is detected, possible root causes shall be determined in accordance with specific isolability requirements. These define how strong possible root causes shall be isolated in order to improve the troubleshooting. Examples are system locations, specific components or specific fault modes. Details on this topic have been explained in Section 4.2. In this section an approach is depicted that can be used to gain all minimal solutions for the isolability problem.

In order to make the isolability items ii of different isolability clusters $ic \in IR$ isolable from each other, a prerequisite is, that their respective symptom sets S have to be different in at least one symptom m. Referring to the example of Figure 10 the fault modes fm_1 and fm_2 are isolable as they have non overlapping symptom sets. All their symptoms can be used as *candidates* for the fulfillment of a requirement. On the contrary symptoms that are included in all symptom sets can not be used to meet a specific isolability requirement. This assumption is used to define sets of isolability candidates C:

$$\text{C(a,b)} := \{m \mid m \in (S(a) \cup S(b)) \backslash (S(a) \cap S(b)) \,, \\ a, b \in ic_j, ic_k,\ ic \in IR, j \neq k,\ |a| \leq |ic_j|,\ |b| \leq |ic_k|\} \ .$$

The candidate set $C(a, b)$ is the set of symptoms m of the conjunction of symptom sets $S(a), S(b)$ that are not in the intersection of all symptom sets $S(a), S(b)$. The determination of C is done problem specific. This means that a, b can be isolability items ii of a specific isolability sub-problem, but can also cover complete isolability cluster ic. A necessary prerequisite for the possible fulfillment of a specific requirement is thus that $C(a, b) \neq \emptyset$ for all $a = ic_j, b = ic_k, ic \in IR$. If this is fulfilled, specific candidates for all sub-problems are computed in the next step. In order to find the candidates that lead to an optimal fulfillment of the overall IR problem a solution is presented in the following. The proceeding is illustrated by examples.

In the previous design phase, solutions were gained that are adequate for the fulfillment of the safety requirement. The next step in the current design phase is thus to check if these solutions already provide candidates for the fulfillment of the isolability requirement. In order to illustrate this point the example from Figure 10 is used. The solutions for the SR that were found, are $M_1^* = \{M1, M2, M3\}$ and $M_2^* = \{M2, M3, M5\}$. An isolability requirement can be that IR=$\{FM4, FM5, FM6, FM7\}$. This means that all cluster $ic_j \in IR$, which are in this case the isolability items $FM4, FM5, FM6$ and $FM7$, shall be isolable from each other under any condition. Figure 12 depicts the case that M_1^* is used to check for the fulfillment.

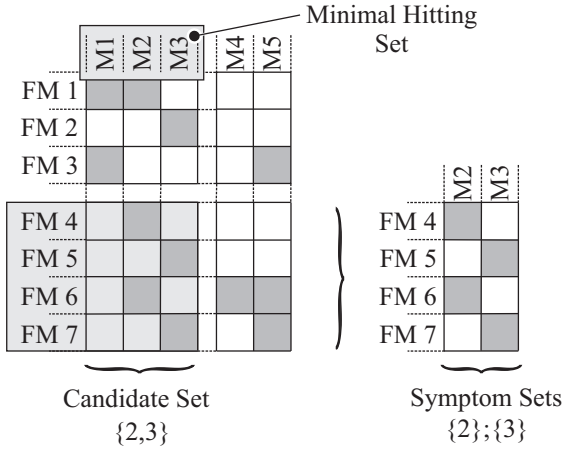


Figure 12. Candidate and symptom sets to check for fulfillment of the isolability requirement IR using M_1^* .

Referring to the previous example, the symptom sets for all isolability items are S(FM4) = S(FM6) = $\{2\}$, and S(FM5) = S(FM7) = $\{3\}$. This illustrates that there are no candidates such that $FM4$ and $FM6$, and $FM5$ and $FM7$ can be isolated from each other. The requirement is thus not fulfilled using the solution M_1^*. The exemplified proceeding to check for the fulfillment of the IR is generalized in Algorithm 2.

Algorithm 2 Check for the fulfillment of the isolability req.

function FULFILLMENT(IR, CEM_{FC}, M_i^*)
  $\mathcal{A} \leftarrow \emptyset$
  for $ic_j,\ ic_k \in IR,\ j \neq k$ **do**
    $(a, b) \leftarrow \mathcal{P}(ic_j, ic_k)$ ▷ $\forall\ a \in ic_j, b \in ic_k$
    $C_{a,b} \leftarrow C(a, b)$ ▷ on CEM_{FC} and $M = M_i^*$
    if $C_{a,b} = \emptyset$ **then** ▷ Req. not fulfilled !
      $\mathcal{A} \leftarrow \mathcal{A} \cup \{(a, b)\}$ ▷ add (a, b) to $\mathcal{A}$
    end if
  end for
  return $\mathcal{A}$
end function

The function FULFILLMENT uses the requirement IR and the safety-related cause-effect matrix CEM_{FC} as an input and checks if each sub-problem, meaning the power set (a, b) of elements ii of the clusters $ic_j,\ ic_k \in IR$, has a non empty candidate set $C_{a,b} = C(a, b)$. An output of the function is the set $\mathcal{A}$. This set is empty in the case that the requirement is fulfilled for the overall problem, otherwise $\mathcal{A}$ includes the sub-problems for which further candidates have to be found.

In the current example the set $\mathcal{A}$ is not empty. This means that the isolability requirement is not fulfilled and further candidates have to be found. Therefore, the maintenance related symptoms are analyzed in the next step. In Figure 13 it is shown that there are four maintenance related symptoms that may lead to a fulfillment of the IR. These are $\{6, 8, 9, 10\}$ and collected in the candidate set C_M. The requirements that were not met by using the candidate set C_S are $(FM4, FM6)$ and $(FM5, FM7)$. Applying now the candidate set C_M, local candidate sets for both the requirements are $C_{FM4,FM6} = \{6, 9, 10\}$ and $C_{FM5,FM7} = \{6, 8, 10\}$. All elements of these local sets lead to a fulfillment of the respective sub-problem. In order to gain solutions that fulfill the overall isolability requirement, combinations of symptoms from both the candidate sets have to be built. For the aim not to find all solutions but the minimal ones, a minimal hitting set algorithm is used. This provides all minimal solutions $\hat{M}_{i,k}$ that lead to a fulfillment of the IR for a specific M_i^*. For the example these solutions are $\hat{M}_{1,1} = \{6\}, \hat{M}_{1,2} = \{10\}$ and $\hat{M}_{1,3} = \{8, 9\}$.

The set $\hat{M}$ includes all solutions $\hat{M}_{i,k}$ for the particular isolability problem. The general proceeding to gain $\hat{M}$ is defined in Algorithm 3. There the function ISOLATION uses the IR, the maintenance-related cause-effect matrix CEM_{MC} and the set $\mathcal{A}$ of unmet sub-problems as an input. The candidates for the fulfillment of the IR are then computed for all elements of $\mathcal{A}$ and collected in the set $\mathcal{B}$. Afterwards a minimal hitting set algorithm uses $\mathcal{B}$ to calculate all minimal solutions $\hat{M}$ that fulfill the IR for the specific M_i^*. The calculation of the candidate set C_M is omitted as it was used only for demonstration.

In order to gain solutions $\tilde{M}_i$ that hold for the overall problem, that is the fulfillment of safety and isolability requirements, the conjunction of both the single solutions M_i^* and $\hat{M}_{i,k}$ has to be built:

$$\tilde{M}_{i,k} = M_i^* \cup \hat{M}_{i,k} \ .$$

For the example, solutions to the overall problem are $\tilde{M}_{1,1} = \{1, 2, 3, 6\}, \tilde{M}_{1,2} = \{1, 2, 3, 10\}$ and $\tilde{M}_{1,3} = \{1, 2, 3, 8, 9\}$.

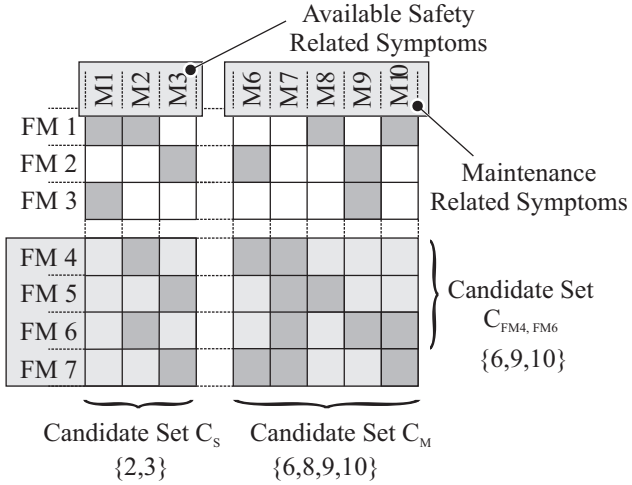


Figure 13. Safety and maintenance related candidate sets for fulfillment of isolability requirement IR using M_1^* .

Algorithm 3 All minimal solutions for the isolability requirement

function ISOLATION($IR, CEM_{MC}, \mathcal{A}$)
  $\hat{M} \leftarrow \emptyset$
  $\mathcal{B} \leftarrow \emptyset$
  for $(a, b) \in \mathcal{A}$ **do**
    $C_{a,b} \leftarrow C(a, b)$ ▷ on CEM_{MC}
    if $C_{a,b} = \emptyset$ **then**
      break ▷ Req. can not be fulfilled
    end if
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{C_{a,b}\}$ ▷ add $C_{a,b}$ to $\mathcal{B}$
  end for
  $\hat{M} \leftarrow$ `MinHittingSets`($\mathcal{B}$)
  return $\hat{M}$
end function

The proposed approach has now to be repeated and to be applied to all the solutions M_i^* in order to gain the global set of all the solutions $\tilde{M}$ that fulfill both the requirements.

In case of the previous example two solutions M_1^* and M_2^* for the fulfillment of the safety requirement were found. A detailed description of all the proposed steps of the analysis process for M_2^* is omitted at this point. However, checking M_2^* for the fulfillment of the IR, four distinct symptom sets and six respective non-empty candidate sets are found. The minimal hitting set M_2^* is thus adequate for the fulfillment of both the safety and the isolability requirement. For this particular case it thus holds that $\tilde{M}_2 = M_2^*$.

In the end of the depicted process four solutions to the overall problem are gained. These are $\tilde{M}_{1,1} = \{1, 2, 3, 6\}, \tilde{M}_{1,2} = \{1, 2, 3, 10\}, \tilde{M}_{1,3} = \{1, 2, 3, 8, 9\}$ and $\tilde{M}_2 = \{2, 3, 5\}$.

The complete proceeding of determining solutions to both the safety and the isolability problem is summed up in the Algorithm 4. There the function `SOLUTIONS` combines the three functions that have been presented in the previous sections.

In order to find the overall optimal solution the relation from Figure 11 combined with a reachability analysis can now be used. This is omitted for the previous example but shown exemplary in the following section for the application of the proposed approach to the high lift actuation system of an Airbus A340-600 aircraft.

5. RESULTS FOR THE AIRBUS A340-600 FLAP SYSTEM

The previous sections gave a general overview of the proceeding and the theoretical backgrounds of a model-based approach to the optimal design of a DS architecture. In this section the results of the application of the approach to the flap system of an Airbus A340-600 aircraft are presented. In the first step, in Subsection 5.1, the solution spaces for the fulfillment of both the requirements and the corresponding efforts are shown. In the second step, in Subsection 5.2, two examples of solutions are depicted in detail. There, an overview of the resulting elements of the DS architectures is given.

Algorithm 4 All minimal solutions that meet both the safety and isolability requirements

function SOLUTIONS(IR, CEM_{FC}, CEM_{MC})
  $\tilde{M} \leftarrow \emptyset$
  $M^* \leftarrow DETECTION(IR, CEM_{FC})$
  for $M_i^* \in M^*$ **do**
    $\hat{M} \leftarrow \emptyset$
    $\mathcal{A} \leftarrow FULFILLMENT(IR, CEM_{FC}, M_i^*)$
    if $\mathcal{A} \neq \emptyset$ **then**
      $\hat{M} \leftarrow ISOLATION(IR, CEM_{MC}, \mathcal{A})$
      for $\hat{M}_k \in \hat{M}$ **do**
        $\tilde{M}_{i,k} \leftarrow M_i^* \cup \hat{M}_k$
        $\tilde{M} \leftarrow \tilde{M} \cup \{\tilde{M}_{i,k}\}$ ▷ add $\tilde{M}_{i,k}$ to $\tilde{M}$
      end for
    else
      $\tilde{M} \leftarrow \tilde{M} \cup \{M_i^*\}$ ▷ add M_i^* to $\tilde{M}$
    end if
  end for
  return $\tilde{M}$
end function

5.1. Analysis

The first stage of the overall design procedure consists in the definition of safety and isolability requirements. Two consecutive steps are then executed to design a DS architecture according to these requirements. A diagnosis model is used to provide safety and maintenance related symptoms. In the first step the safety related symptoms are evaluated in order to fulfill the related requirement. As a result 94 different solutions M_i^* are gained. Strictly speaking M_i^* is a symptom set, as introduced in Subsection 4.2, but in this context it should be seen as a solution, that always consists of a set of specific sensors, a set of features, a set of symptoms and the diagnostic knowledge, that is stored in the cause-effect matrices. Details about the different elements are given in Section 5.2 by means of two specific examples.

In order to evaluate every single solution M_i^* the effort for placement of sensors was introduced as a criteria in Section 4.3. This simple cost function should by no means considered to be complete, but as a first step in a multi criteria decision making process. The effort that is induced by every solution M_i^* is depicted in Figure 14. The solutions are marked there with black dots and are arranged according to their specific index i. In total, there are 11 solutions that cause the same minimal effort of five unit cost whereas two solutions lead to the highest effort which is 15 unit cost. The two solutions 9 and 19 are depicted in more detail in the end of this section.

In the second step of the approach the fulfillment of specific isolability requirements is checked and the design is extended if necessary. As an example, the requirement $IR = \{\text{LH-System}, \text{RH-System}\}$ is chosen. This states that under any condition the root cause of a detected failure shall be isolated at least between the left and right hand system half. The

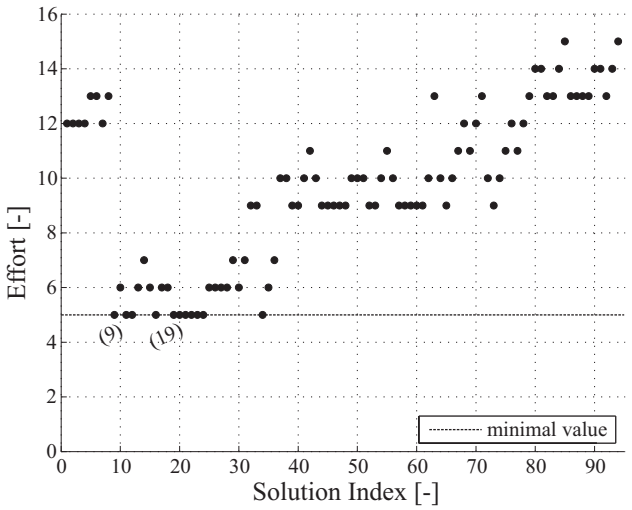


Figure 14. Effort for the fulfillment of the safety requirement.

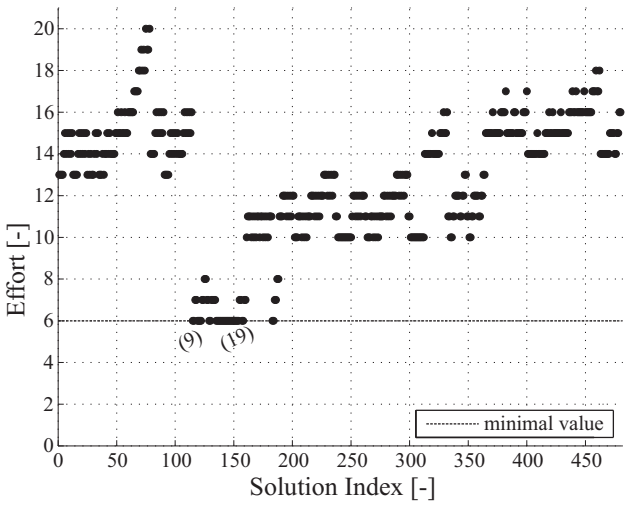


Figure 15. Effort for the fulfillment of both the safety and the isolability requirement.

result of the analysis is, that all of the solutions M_i^* have to be extended by means of additional symptoms $\hat{M}_{i,k}$ in order to meet the IR. In detail this means, that some of the previous solutions can be extended by additional symptoms, that are gained using features from signals that are linked to sensors, which are already part of the architecture due to the safety requirement. This case induces no additional effort therefore. In other cases new, maintenance related only symptoms and sensors are needed, which accordingly rises the effort for the fulfillment of the IR.

In total, there are 480 solutions $\tilde{M}_{i,k} = M_i^* \cup \hat{M}_{i,k}$ that meet both the requirements in a local optimal way with respect to minimality. An overview of the effort for every single solution $\tilde{M}_{i,k}$ is given in Figure 15. Each tuple (i, k) is there represented by a new sequential index i^*. The solutions 9 and 19 from Figure 14 are highlighted for two respective tuples (i, k) using the old index i. Both the solutions had to be extended using an additional sensor in order to meet the IR.

A result of the analysis is, that all of the optimal solutions found in the first step remain optimal also in the second step. In addition there are five more solutions that cause the same minimal effort of six unit cost. A reason for that is, as mentioned previously, that for some solutions from step one there was no need for additional sensors in order to meet the IR, but only for additional symptoms gained from data from already available sensors. Due to that, some solutions from step one did not cause a rise in effort during the analysis in step two. Summarizing the results, in the end of the analysis process, there are 16 solutions that meet both the requirements in a global optimal way with respect to minimality and the single cost function that has been considered.

In order to choose one solution that has to be detailed in further steps, more criteria for the evaluation have to be introduced. Possible extensions are the consideration of an isolability performance metric and the use of more realistic cost factors amongst others.

Apart from detailing of the cost function, a further important aspect is, that only one design case, that means one operating condition, under a certain side condition has been considered so far. In the current case the analysis was based on an extension of the system against high air loads. Further cases that have to be analyzed are different operating speeds and other load conditions, so that in the end an overall optimal solution is gained that holds for all conditions considered.

5.2. Discussion

The solutions 9 and 19 that were marked in both the previous figures belong to the set of global optimal solutions. In the following, both of them are depicted in detail and differences are discussed.

An overview of the resulting preliminary design of a DS architecture according to solution 9 is given in Figure 16. On the lowest layer of this architecture there are seven sensors. These are three position sensors, three proxy sensors and one load sensor. The sensors a and c are placed on the left and right hand system side, whereas all the other sensors are only placed once. At this point of the design procedure these sensors are basically reduced to the physical quantity they measure, so that aspects of redundancy have not been considered yet which may demand additional sensors in future. Although there are currently seven sensors, the effort for the architecture consists of six unit cost. This is based on the fact, that the position sensor located at the PCU is also used for the system control and by that comes for free for the diagnostic tasks.

The sensor signals are evaluated by six monitoring devices.

Five of them are related to the safety task and one is only needed for the fulfillment of the isolability requirement. This device is marked as maintenance related only. All of the devices are built up like it is shown for the device A. In the first step different signals are used to generate features, that in the second step are compared to a threshold in order to detect symptoms. The example of device A is related to the failure condition $Asymmetry$. In this case, position measurements from both ends of the mechanical transmission system are used to calculate the absolute difference between both the position signals. If this feature exceeds a certain threshold the symptom of the failure condition is detected. In order to take adequate measures and to generate maintenance messages all of the detected symptoms are correlated afterwards. In order to determine potential root causes the diagnostic knowledge is used. At this point of the preliminary design phase it is stored in the cause-effect matrices. Due to the isolability requirement, the root cause can then be isolated at least between the left and right hand system side by using the knowledge.

The resulting architecture of the DS according to solution 9 is very close to the Airbus A340-600 flap system's current DS's architecture. The focus of the current architecture is on the detection of FCs though. Due to confidentiality reasons no details can be stated about that at this point. In the proposed new concept though, less monitoring devices are needed in order to meet the stated safety requirement. Furthermore one additional, maintenance related only, device is added in order to fulfill the isolability requirement. Due to the non-safety critical functionality of this device a low development assurance level can be applied in the next design steps. It has to be considered though, that a loss of the respective function could lead to more cost and effort for the troubleshooting. The consideration of this aspect and the detailing of all the elements of the architecture will come in further design steps. Next, a second alternative of a preliminary architecture of a DS is presented.

Figure 17 depicts the resulting architecture according to solution 19. In this case, there are again used information provided by seven sensors. These are position and proxy sensors and one load sensor. The effort consists of six unit cost, due to the fact, that again the position sensor, that is located at the PCU, comes for free. All of the sensor signals are evaluated by six safety related monitoring devices and again one additional, maintenance related only, device.

Comparing both the architectures they show clear similarities. Both are using only information that are provided by position, proxy and load sensors. They differ in the way, the sensors are located and features are extracted from the measurements.

The architecture according to solution 9 uses information that are provided by position sensors a, which are located at the outermost ends of the mechanical transmission system.

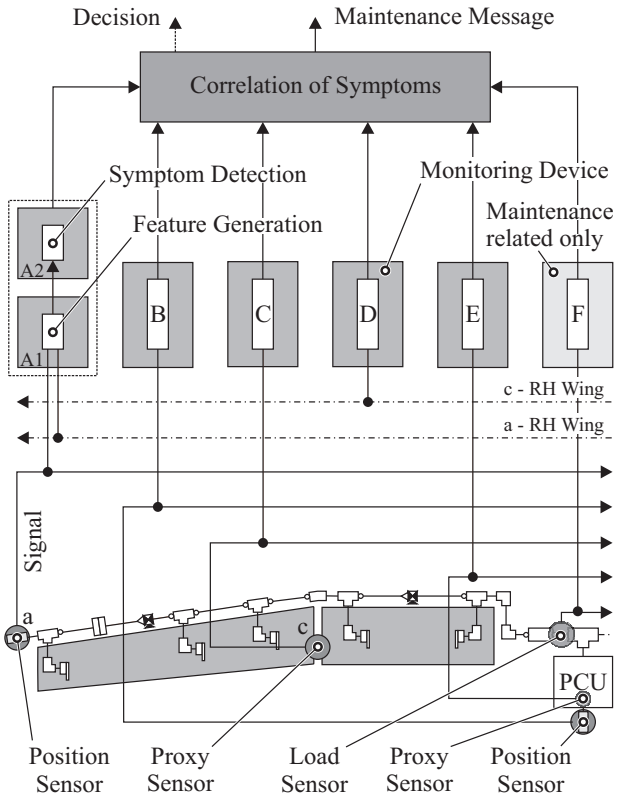


Figure 16. Detailed overview of the resulting architecture according to solution 9.

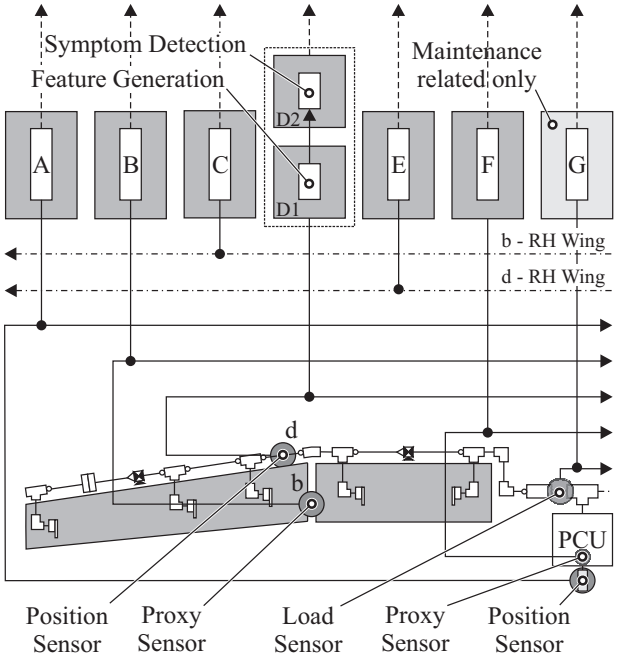


Figure 17. Detailed overview of the resulting architecture for solution 19.

The monitoring device A is in charge of extracting a feature from these information in order to detect the symptom of an $Asymmetry$ failure condition. In the architecture according to solution 19 the sensors a are missing. Instead, two sensors d are used, that are located on the transmission system between the in and outboard flap of each system side. In detail, the sensors d provide position measurements. The monitoring devices D and E use these measurements to calculate a speed signal that is used as a feature in order to detect a symptom of an $Overspeed$ failure condition.

The differences in both the architectures result from the fact, that the failure conditions $Asymmetry$ and $Overspeed$ overlap. The safety requirement was defined such that the overlapping should always be as minimal as possible, so that the two alternatives result as solutions. If this holds for all the conditions that have to be considered, has to be checked in the extended design procedure, as mentioned previously.

In the current case only unit cost factors for the placement of sensors have been considered in order to evaluate all the solutions. Extending this approach by taking the weight for wiring into account, solution 19 seems to be a little advantageous. On the other hand though, the computational effort rises, due to the fact that more monitoring devices are needed. Therefore, it is obvious that further criteria have to be identified and considered in order to determine the overall optimal solution.

6. RELATED WORK

The development of diagnostic systems and in general of prognostic and health management systems has been studied by various researchers under different aspects. In the following, selected examples of works are presented that are related to the topics of this paper.

In the work of (Kurtoglu, Johnson, Barszcz, Johnson, & Robinson, 2008) a design methodology for the development of system health management is introduced. This is called the Functional Fault Analysis(FFA) and is based on a functional model of the system to be analyzed. Outputs of the FFA are timing analyses for fault-effect propagation, ambiguity statistics for fault isolation and the model itself for online integration. The definition of effect nodes and test points shows similarities to failure and maintenance conditions from our paper, but no clear policies about the optimal selection of the nodes and no integrated requirement based process are shown.

Another approach that combines an extended tabular FMECA and a functional block diagram to a functional graphical health management model is presented in (Kacprzynski, Roemer, Hess, & Bladen, 2001) and (Kacprzynski, Roemer, & Hess, 2002). The model and its elements contain various attributes, that are used as input to a fitness function for a genetic optimization procedure. Output of the analysis is a health management configuration consisting of sensors and algorithms that has the highest system reliability to cost/benefit ratio. Not only diagnostic but also prognostic aspects are addressed. Temporal information, that are manually inserted in the process of (Kurtoglu et al., 2008), are not considered completely, but by means of propagation probabilities and response models(Kacprzynski et al., 2002). While in our approach, for each task of the design procedure, a requirement is introduced, in order to keep the process traceable, and to extend the model complexity only if needed, the design procedure of (Kacprzynski et al., 2001) is basically done using only one model and one single iteration step.

The papers mentioned and other similar ones focus on the development of functional, qualitative models that provide information about fault propagation and serve as basis for different analyses. Input to most of them is a manually created FMECA. Matters of optimality are only dealt with in a few and integrated processes for the design and test are addressed only marginally. The focus of our paper was to define the framework of a traceable process, based on physical models, where the design and test of a diagnosis system is done in consecutive steps according to posed requirements. A set of two requirements was introduced. This however should not considered to be complete but to be the basic starting point. The further discussion on related work will therefore focus on papers, that have problem formulations with similarities to our paper.

In the work of (Scandura, 2005) an overview of a general framework for the development of integrated vehicle health management systems is introduced and the importance of the combined consideration of a philosophy, a methodology and a continuous process is emphasized. Policies like a fault detection and isolation philosophy, the optimal sensor quantity and placement guidelines are mentioned, but no approaches to design a system accordingly are presented.

A detailed description of an approach that is used by Boeing for the development of model-driven integrated support architectures is given in (Ofsthun & Wilmering, 2004). There, a process is defined, that is centered around requirement based design and test. The framework of the process is close to our approach, whereas the realization is different. While our approach uses a performance model, that tries to capture the physics of operation under all normal and failure conditions, their starting point is a qualitative model of the system. The result of their process is a directed timed failure propagation graph that is formalized as platform executable code. In contrast, our approach focuses on the configuration of a generic diagnostic engine and the generation of diagnostic rules as presented in (Modest, Schories, et al., 2011).

7. CONCLUSION

This paper addressed a model-based development approach for diagnosis systems of high lift actuation systems. The focus was on the definition of requirements and the optimal conceptual design of a diagnosis system architecture. A diagnosis model was developed therefore in the first step. This provided safety and reliability related cause-effect matrices. These were used as an input for a two step design process. The first step was about the design for fulfillment of safety-related requirements, whereas the second step focused on the extended design to meet reliability-related requirements. A set of minimal solutions that met both the requirements was found in the end. In order to determine the optimal solution a cost criteria was introduced, so that optimality was defined in terms of minimality and a single criteria for effort. As a result of the overall proceeding a preliminary specification of an optimal design for a diagnosis system architecture was gained. This specification showed the type and location of sensors and respective monitoring devices by means of signals, features and symptoms. All steps of the process were illustrated by examples and defined afterwards. The results of the application to the high lift actuation system of an Airbus A340-600 aircraft were presented and discussed in the end. In the following, an outlook about open points and future research activities is given.

The current approach considered only one design case for the HLS operation and only unit costs as criteria for effort. Future work focuses therefore on the consideration of different design cases and more advanced evaluation criteria in order to find the overall optimal solution. A performance criteria of isolability might be one point for extension as well as the consideration of weight factors for wiring. Furthermore the other parts of the proposed development approach have to be worked out. The next steps are the detailed system design as well as the component design. This includes further studies on monitoring, sensor devices and BIT functionalities. Aspects that have to be addressed are the definition of the kind of sensor, meaning hardware or a virtual sensor, and the evaluation of different strategies to the BIT, amongst others. Strongly related to that is the diagnosis model. Up to now, a quasi static model has been used. The reason to start with such a model was based on the concept to increase the model's complexity according to the posed requirements from stage to stage and only if needed. The current model can thus be seen as a first stage model. It has now to be investigated what the additional value of second stage models, i.e. dynamic models, offers and how to link both the stages. Important aspects that have to be answered by that are the support for the detailed design phases and the consideration of temporal aspects of the symptoms.

A complexity analysis for the minimal hitting set algorithm was not in the scope of this paper, but is an important aspect that has already been dealt with and will be discussed in a further paper together with the extended design procedure.

ACKNOWLEDGMENT

The authors thank the Airbus Operations GmbH for sponsoring and supporting their work.

NOMENCLATURE

ACQ	**A**cquisition
BIT	**B**uilt **I**n **T**est
CEM	**C**ause **E**ffect **M**atrix
$DISP$	**D**isposal
DOC	**D**irect **O**perating **C**ost
FC	**F**ailure **C**ondition
IR	**I**solability **R**equirement
LCC	**L**ife **C**ycle **C**ost
HLS	**H**igh **L**ift Actuation **S**ystem
HMS	**H**ealth **M**anagement **S**ystem
MC	**M**aintenance **C**ondition
MRO	**M**aintenance **R**epair **O**verhaul
OPS	**O**peration and **S**upport
PCU	**P**ower **C**ontrol **U**nit
$RDTE$	**R**esearch **D**evelopment **T**est **E**valuation
$SFCC$	**S**lat **F**lap **C**ontrol **C**omputer
SR	**S**afety **R**equirement
WTB	**W**ing **T**ip **B**rake

fm	Fault Mode
ic	Isolability Cluster
ii	Isolability Item
m	Symptom
M	Set of all Symptoms
S	Set of Symptoms for specific Fault Mode

REFERENCES

Bunus, P., Isaksson, O., Frey, B., & Münker, B. (2009). Model-Based Diagnostics Techniques for Avionics Applications with Rodon. In O. von Estorff & F. Thielecke (Eds.), *Proceedings of the 2nd International Workshop on Aircraft System Technologies.* Shaker.

Greiner, R., Smith, B. A., & Wilkerson, R. W. (1989). A Correction to the Algorithm in Reiter's Theory of Diagnosis. In Elsevier Science Publishers Ltd. (Ed.), *Artificial Intelligence* (Vol. 41, pp. 79–88). Essex, UK: Elsevier Science Publishers Ltd.

Haskins, C. (Ed.). (2006). *Systems Engineering Handbook - A Guide For System Life Cycle Processes And Activities* (3rd ed.). INCOSE - International Council On Systems Enginnering.

Isermann, R. (2006). *Fault-Diagnosis Systems*. Springer.

Kacprzynski, G. J., Roemer, M. J., & Hess, A. J. (2002). Health Management System Design: Development,

Simulation and Cost/Benefit Optimization. In *2002 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3065-3072).

Kacprzynski, G. J., Roemer, M. J., Hess, A. J., & Bladen, K. R. (2001). Extending FMECA-Health Management Design Optimization for Aerospace Applications. In *2001 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3105-3112).

Kurtoglu, T., Johnson, S. B., Barszcz, E., Johnson, J. R., & Robinson, P. I. (2008, october). Integrating System Health Management into the Early Design of Aerospace Systems using Functional Fault Analysis. In *2008 International Conference on Prognostics and Health Management Proceedings* (p. 1-11).

Lulla, C. (2011). Functional Flexibility of the A350XWB High Lift System. In Deutsche Gesellschaft für Luft- und Raumfahrt (DGLR) (Ed.), *Tagungsband DLRK 2011* (pp. 385–392).

Modest, C., Grymlas, J., Schories, K., Lüdders, H. P., & Thielecke, F. (2011). Model-Based Development of Control and Diagnosis Concepts for Multifunctional Fuel Cell Systems. In *9th European Workshop on Advanced Control and Diagnosis, ACD 2011*. Budapest.

Modest, C., Schories, K., Lüdders, H. P., & Thielecke, F. (2011). A Model-Based Development Approach for a Diagnostic System for a Multifunctional Fuel Cell System. In Society of Automotive Engineers (Ed.), *SAE International Journal Of Aerospace* (Vol. 4, p. 1324-1333). Warrendale, PA: SAE International.

Ofsthun, S. C., & Wilmering, T. J. (2004, March). Model-Driven Development of Integrated Health Management Architectures. In *2004 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3692-3705).

Recksiek, M. (2009). Advanced High Lift System Architecture With Distributed Electrical Flap Actuation. In O. v. Estorff & F. Thielecke (Eds.), *Proceedings of the 2nd International Workshop on Aircraft System Technologies* (pp. 49–59). Shaker.

Reiter, R. (1987). A Theory of Diagnosis from First Principles. In Elsevier Science Publishers Ltd. (Ed.), *Artificial Intelligence* (Vol. 32, pp. 57–95). Essex, UK: Elsevier Science Publishers Ltd.

Roskam, J. (2006). *Airplane Cost Estimation : Design, Development, Manufacturing and Operating* (3rd ed.). Lawrence, Canada: DARcorporation.

SAE (Ed.). (1996). *Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment* (No. 4761).

Scandura, J., Philip A. (2005, October). Integrated Vehicle Health Management as a System Engineering Discipline. In *Digital Avionics Systems Conference, 2005. DASC 2005. the 24th* (Vol. 2).

BIOGRAPHIES

Christian Modest is a research assistant at the Institute of Aircraft Systems Engineering of the Hamburg University of Technology. He received the German academic title Dipl.-Ing. in mechanical engineering from the Hamburg University of Technology in 2010. His research interests include diagnostics, health management systems and aircraft systems engineering.

Frank Thielecke is the head of the Institute of Aircraft Systems Engineering of the Hamburg University of Technology. For 10 years he was in charge of the Systems Automation Department at the Institute of Flight Systems of the German Center for Aeronautics and Space (DLR). He has a long-standing experience in fields of model-driven systems engineering, flight systems design, reliability analysis as well as diagnostics and testing.