A MODEL-BASED REQUIREMENT VALIDATION PROCESS FOR HANDLING QUALITIES OF eVTOLs

V. Meidinger, K. Schmiechen, and F. Holzapfel
Technical University of Munich, Garching, 85748, Germany

ABSTRACT

For the development of new electronic Vertical Take-Off and Landing (eVTOL) configurations, missing standards and experience lead to a necessary derivation of new standards and requirements. In our approach, existing standards for rotorcraft are transformed to requirements for the new eVTOL. To ensure a high quality of the transformed requirements, the validation process consisting of requirement derivation, formalization/modeling, testing and validation is integrated in a modern model-based development process supported by the industry tools *MATLAB Simulink* and *Polarion Requirements*. An ideal controller is used to identify the best theoretically attainable system performance and thus to validate handling qualities requirements derived by existing standards. This process promises cost and time benefits due to the model-based requirement validation in early development stages.

1. INTRODUCTION

Especially in aviation, the systems to be developed are becoming more and more complex. To handle this complexity and to prevent design errors being detected only late in the development phase, it is becoming more and more important to have proper requirements early on in the project. This prevents delays in the schedule and exceedance of the project budget [1]. For proper requirements, it is, among other properties, important that they are correct, complete and not in contradiction to each other.

The goal of the process, which is defined in Section 2, is to ensure a high quality of requirements. The textual requirements are derived in the first step and then formalized or modeled to be interpretable by a computer. It depends on the nature of the requirement whether formalization or modeling is more feasible. The formalized or modeled requirements are linked with their textual representations from the first step to enable traceability between the domains. In the third step, they are tested to ensure the above named properties. In the last step of validation, the parameters of the requirements are modified to the values that result in the best attainable performance of the eVTOL.

The first three steps of this process were already performed similarly in [2] not for dynamic behavior requirements but for the discrete behavior of a signal source selection function. For requirement validation, the paper focused on the property proving capabilities of the *Simulink Design Verifier* by *The MathWorks*. Verification and testing in the domain of model based design with the *MATLAB* product family was presented in [3]. In [4], model checking of the *Simulink Design Verifier* was applied for verfication and validation whereas [5] examines the capabilities of the tool from a safety assessment point of view and presents the model based safety assessment tool *ExCuSe*. Previous work on requirements formalization and validation was performed in [1, 6] and requirements modeling and requirements-based test generation was presented in [7, 8].

The industry tools used in this paper are *Polarion Requirements* and *MATLAB* with its extensions *Simulink* and *Simulink Test*. *Polarion Requirements* is a module of the web-based Application Lifecycle Management (ALM) suite by *Siemens Industry Software* and features requirements, change/configuration and branch management for engineering projects. *MATLAB* is a numerical computing environment by *The MathWorks, Inc*. Its extension *Simulink* is a graphical programming tool for modeling, simulating and analyzing dynamic system. *Simulink Test* provides tools for design, management and execution of test cases.

The tool *SimPol* is used to ensure bi-directional traceability between the textual requirements in *Polarion* and the *MATLAB* code and *Simulink* models. *SimPol* has been developed at the Institute of Flight System Dynamics (FSD) of the Technical University of Munich and was first introduced in [2].

The process will be applied in Section 3 for the current eVTOL multi-copter transition vehicle research project at FSD. Since large multi-copter transition vehicle are not yet established at the market, standards and design requirements are rare. Therefore, handling qualities of rotorcraft need to be transformed to eVTOLs and applied to this project without previous experience. An ideal controller that has perfect knowledge of all system states (not using sensors or data fusion) is used to attain the theoretically best possible controller performance. If this controller can fulfill the requirements, it can be assumed that the requirements are valid for the controller architecture embedded within the given configuration and evolved in the next development step.

  doi:10.25967/490030

2. PROCESS DEFINITION

The presented process includes the derivation, formalization as well as the model-based testing and evaluation of requirements. The process is outlined in Figure 1. Firstly, requirements are collected in the form of customer and certification requirements and derived into aircraft level requirements within the *Polarion* environment, while the actual formalization, test definition and requirement evaluation takes place in the *MathWorks* environment. In the following, the single steps of the process are presented.

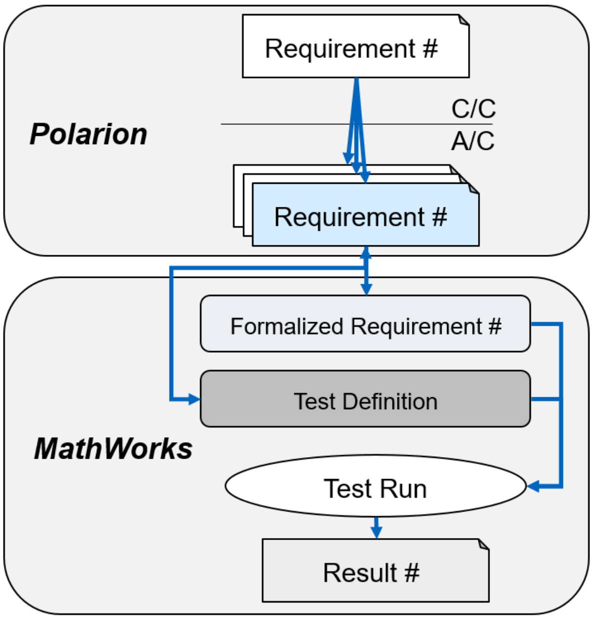

Fig. 1 Process Steps

2.1 Requirement Derivation

The first step is to import the textual requirements to *Polarion*. Since requirements, which are provided by a standard like the ADS-33E-PRF [9], are formulated in natural language, it is necessary to reformulate these requirements in a standardized way. For this purpose, a requirement template is used as shown in Figure 2. Using such a requirement template provides derived requirements in a form such that they can be assessed within the subsequent steps of the presented process.

2.2 Requirement Formalization/Modeling

To evaluate requirements within the *MATLAB/Simulink* development environment, the requirements need to be formalized in terms of being interpretable by *MATLAB/Simulink*. Depending on the requirement properties (e.g. temporal or logic, complexity), different formalization methods have to be considered.

In this paper, the focus shall only be on functional performance requirements. These requirements are mainly formulated for the dynamic system behavior. For this reason, they are formalized by implementing a routine which automatically processes signals which are provided by a simulation to evaluate the requirement criteria. This can be done by implementing a Requirement Assessment Model (RAM) in *Simulink* or by implementing a Requirement Assessment Function (RAF) in *MATLAB*. [2] gives an example of a RAM, while this paper will only consider the formalization via the RAF.

Algorithm 1 shows the generic structure of a requirement implemented as a RAF. The RAF requires a "Test Results Object" provided by the *Simulink Test Manger* as an input. This object contains all information about the conducted test run and model meta-data which is necessary to evaluate the requirement. The RAF algorithm loads the predefined requirement criteria and runs an evaluation routine. This evaluation routine must be adapted for each requirement, but contains parts which are similar for a wide range of requirements and can be reused to implement new requirements. The first evaluation step is to extract the signals from the input object. Based on these signals, certain checks must be performed to ensure that the conducted test matches with the requirement which shall be evaluated (e.g. if the required maneuver was performed). If this is not the case, the RAF will provide an error message and stop the evaluation. The test will then be indicated as failed. After the checks, the routine extracts the requirement metrics from the signals which are then evaluated against the defined requirement criteria. Finally, feedback information is provided as function output by adding the test verdict to the "Test Results Object". This feedback can be graphical and textual.

Input: Test Results Object
Output: Test Results Object
Requirement Criteria;
Evaluation Routine;
- Extract signals from "Test Results Object";
- Run checks on signals;
if *Outside predefined tolerance* **then**
  | RETURN ERROR MSG;
else
  | CONTINUE EVALUATION;
end
- Extract requirement metrics from signals;
- Evaluate metrics against requirement criteria;
- Provide graphical and textual feedback;

Algorithm 1: Structure of a Requirement Assessment Function (RAF)

The formalized requirement (RAF or RAM) can be linked to *Polarion* via *SimPol* to enable the bi-directional traceability as it is required by [11] for the development of airborne software that could lead to a failure condition of the aircraft. Figure 3 shows the concept of the applied tool *SimPol*. It establishes the bi-

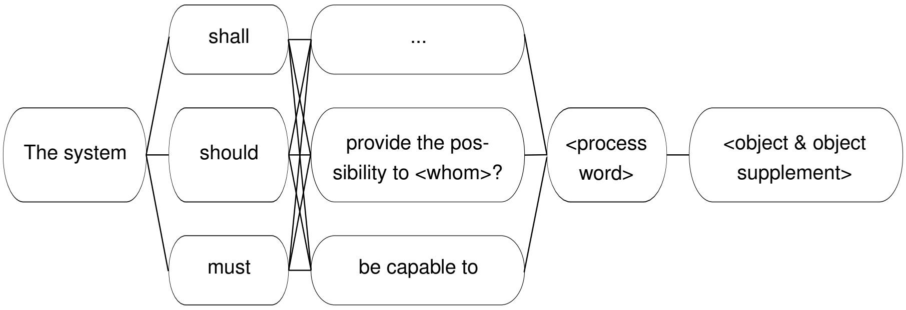

Fig. 2 Requirement template (see [10])

directional link by creating a surrogate workitem within the *Polarion* environment which again is connected to the requirement in the *MathWorks* environment. That way, the requirement artifact is not modified when the link is established. *SimPol* provides a GUI which easily allows the definition and maintenance of the links.

2.3 Requirement Testing

Besides the requirement formalization, test scenarios must be defined. For the current project, the so-called classification tree method as described in [12] is used to determine relevant test conditions as the definition of expressive test conditions is crucial. The test scenarios are used to derive the test data for test case implementation which is done by using the *Simulink Test Manager* of the *Simulink Test* toolbox. Figure 4 shows an example test case which serves as a container for the different elements necessary for the evaluation of a requirement. This includes a formalized requirement below the "Custom Criteria" dropdown menu, a Simulink model below the "System Under Test" (SUT) menu, as well as test data below the "Parameter Overrides" and the "Inputs" menus.

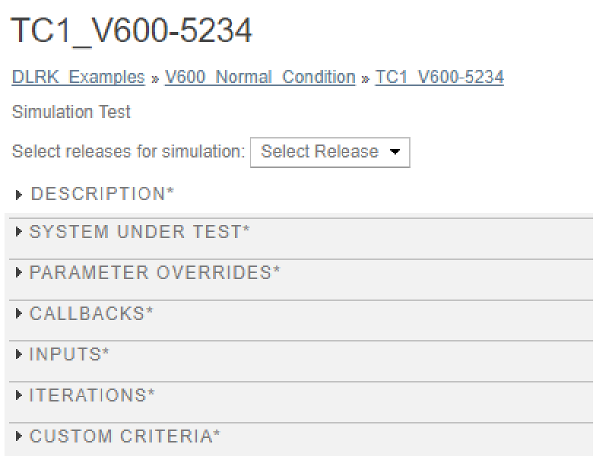

Fig. 4 Simulink Test Manager Test Case Example

Within the context of the current project and the applied model-based development approach, this process is convenient since high fidelity models of the Flight Control System (FCS), the vehicle and the environment are already available and can be directly used as SUT. The procedure of this simulation based testing is shown in Figure 5. First, the specified test data is applied to the simulation, e.g. the specific pitch command inputs. Then, the simulation run is conducted yielding a pitch angle response. This response is then automatically evaluated by the RAF by checking the signal response based on an extracted metric like the maximum pitch angle value against the requirement criteria. A feedback of the requirement verdict (pass or fail) and custom additional information is provided by the RAF.

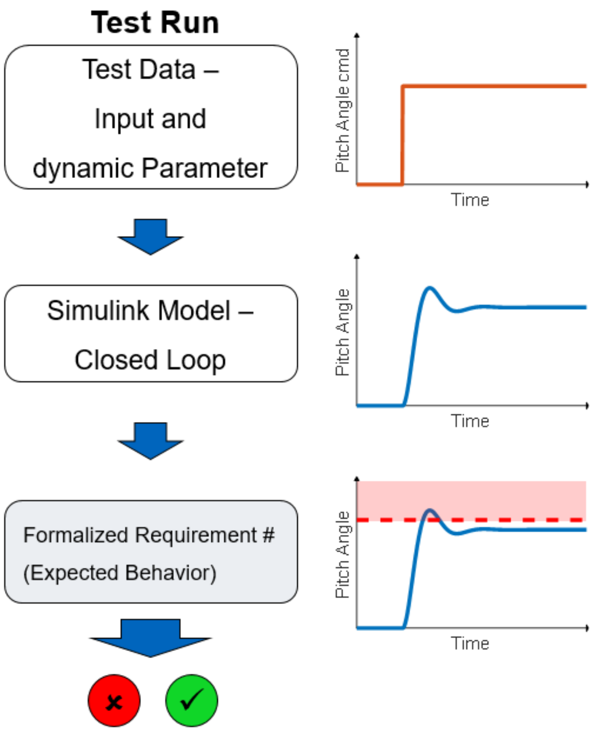

Fig. 5 Simulation based testing procedure

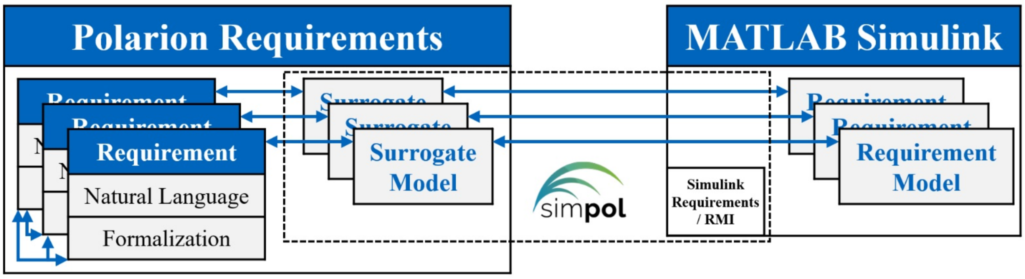

Fig. 3 Linking Requirements with SimPol [2]

2.4 Requirement Validation

The conclusions, based on the outcome of the requirement evaluation using the RAFs and RAMs, depend on the considered SUT. Therefore, the SUT must be chosen carefully.

For the presented process, the focus lies on functional performance requirements. These requirements state metrics for the overall closed-loop system response like (e.g. bank angle responses). For this problem, an Incremental Nonlinear Dynamic Inversion (INDI) controller which has access to all plant states (no sensors and no data fusion) is used. The INDI controller approach allows to specify the desired dynamic behavior directly within the controller structure, which will be the same as later in the real controller. This controller can be considered as the best possible controller which shall be called the "ideal" controller in following. It is combined with a high fidelity model of the eVTOL configuration and the environment. The resulting closed-loop system is used as the SUT in the following and allows to make performance related conclusions based on the results of conducted tests. If a requirement cannot be fulfilled by the system using the ideal controller, the system using a real controller with reduced performance will also not be able to fulfill the requirement.

If a requirement cannot be fulfilled, two views can be considered dependent on certain assumptions. Firstly, the requirement can be assumed as fixed but the eVTOL configuration is not yet finally defined. This might happen in the very beginning of an agile development process of a new eVTOL configuration. This implies that the configuration together with the ideal controller is not able to fulfill the considered requirement and therefore the configuration must be modified. We call this the verification view.

Secondly, the general eVTOL configuration with its kinematic constraints can be assumed as fixed while the requirement is not yet validated. This implies that the requirement itself is not applicable to the considered eVTOL configuration and must be adapted. This happens when requirements are applied to new system configurations which the requirement was not designed for. This is regarded as the validation view and will be the focus in this paper.

3. PROCESS APPLICATION

FSD is currently involved in developing an eVTOL aircraft. The main requirement for this aircraft is the capability to operate in a "Hover" Flight Phase as well as in a "Wing-Borne" Flight Phase. This approach shall enable the aircraft to operate in urban areas as a relief from today's ground based commuting systems [13]. However, this operational concept implies that the so-called "Transition" flight phase between "Hover" and "Wing-Borne " must be covered. Such concepts are discussed in various research projects (e.g. [13–15]). To cover this operational concept, the eVTOL configuration differs significantly from normal aircraft and rotorcraft configurations.

Any development of a rotorcraft or aircraft must fulfill a certain set of requirements to gain initial airworthiness [16]. Thus, a subset of the requirements is dedicated to the FCS [15] on which FSD is focusing within this development project. For this paper and discussion, however, only a subset of the large requirement set for the FCS shall be considered. The focus lies on functional performance requirements, more precise on handling qualities requirements, which are directly influenced by the FCS software.

Due to the new configuration of the eVTOL, there is a lack of applicable standards. To overcome this issue, the approach in this project is, similar to other projects (e.g. [15]), to use the existing standards for aircraft as well as rotorcraft and apply them to the eVTOL. For example, in the context of handling qualities requirements as part of the functional performance requirements, there are military standards (which serve also as a reference for civil projects [17]) for rotorcraft (ADS-33E-PRF) and aircraft (MIL-HDBK-1797 [18]).

However, due to the special configuration of the considered eVTOL (multiple propellers for "Hover" and wings for "Wing-Borne" Flight Phase), the requirements given by the standards might not be valid since they are developed for regular rotocraft and aircraft configurations. This problem shall be considered by the presented process of Section 2. The process application is outlined in the following by applying two sample requirements and discussing the results of the conducted tests.

3.1 Requirement Derivation

The very first step is to collect the requirements that must be considered. In the outlined case, we focus only on the ADS-33E-PRF. Many requirements given in this document are not provided in the desired template form suitable for testing as mentioned in Section 2. For example, the ADS-33E-PRF specifies roll and pitch axis requirements together, while for testing purposes, there should be one requirement for each axis. Therefore, the second step is to reformulate the requirement and to convert it into the desired template form as shown in Figure 2.

To give an example for the first two process steps, two requirements are derived based on the following requirement given by the ADS-33E-PRF [9]:

ID: V600-4620 *For Response-Types designated as Translational Rate Command, the translational rate response to step cockpit pitch (roll) control position or force inputs shall have a qualitative first order appearance, and shall have an equivalent rise time,* $T_{\dot{x}_{eq}}$ $(T_{\dot{y}_{eq}})$*, no less than 2.5 seconds and no greater than 5 seconds. The parameter* $T_{\dot{x}_{eq}}$ $(T_{\dot{y}_{eq}})$ *is defined in Figure 6. For Level 1, the following requirements shall apply:*

- a. *The pitch and roll attitudes shall not exhibit objectionable overshoots in response to a step cockpit controller input.*
- b. *Zero cockpit control force and deflection shall correspond to zero translational rate with respect to fixed objects, or to the landing point on a moving ship.*
- c. *There shall be no noticeable overshoots in the response of translational rate to control inputs. The gradient of translational rate with control input shall be smooth and continuous.*

In addition, for centerstick controllers, the variation in translational rate with control deflection should lie within the limits of Figure 7. For sidestick controllers, the variation in translational rate with control force should lie within the limits of Figure 8.

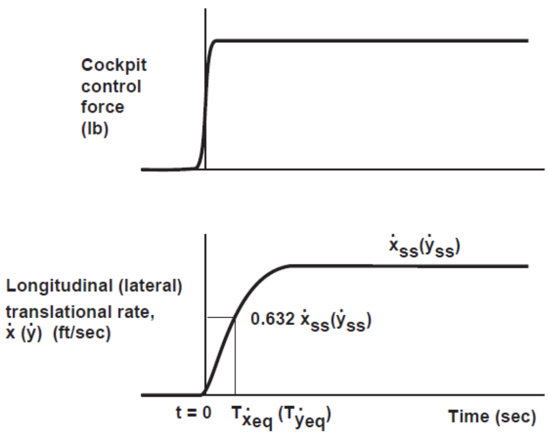

Fig. 6 Definition of equivalent rise time [9]

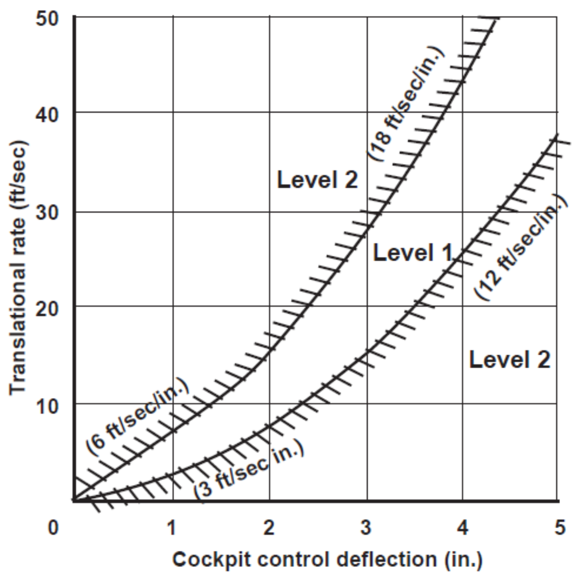

Fig. 7 Control response requirement for center-stick controllers [9]

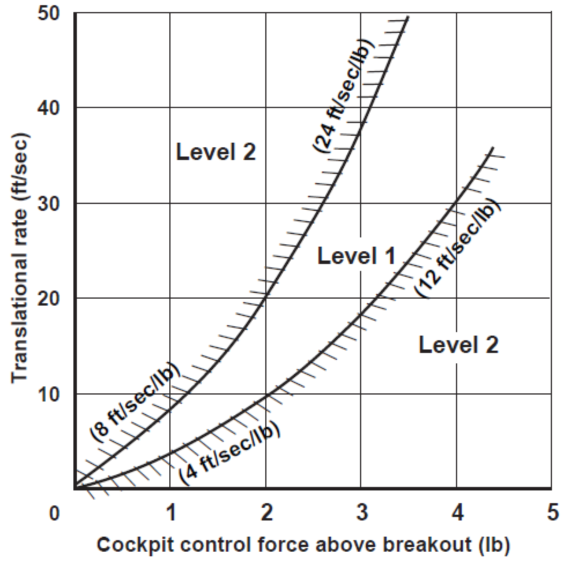

Fig. 8 Control/response requirement for side-stick controllers [9]

Considering the first paragraph of this requirement, the derived requirement is given in the following:

ID: V600-5234 *If there is a step input in lateral direction with maximum possible stick deflection, the A/C shall have a PT1 like response in the lateral velocity in C-frame with a rise time between 2.5 seconds and 5 seconds.*

Based on the bullet point "a." of the main requirement V600-4620, a second derived requirement is given by the following:

ID: V600-5251 *If there is a step input in lateral direction with maximum possible stick deflection, the A/C bank attitude response shall have no overshoot greater than 1 %.*

Requirements V600-5234 and V600-5251 are linked with the parent requirement V600-4620 in *Polarion*. These two requirements are the new basis for further considerations.

Requirement V600-5251 directly shows a first problem in a development process using such standards. Within this process, it is required to formalize the requirement which is based on a qualitative statement "not objectionable". In this case, a 1% overshoot of the steady bank angle is assumed as not objectionable.

3.2 Requirement Formalization/Modeling

The two selected example requirements must be formalized in the next step of the process using RAFs. The pseudo code of the RAF for requirement V600-5234 is given in Algorithm 2, which shows the required input data, the output data and the functional steps to evaluate the requirement. The algorithm uses a specific "Test Results Object" object (*STMCustomCriteria* class) provided by the *Simulink Test Manager* as input data. This object contains information about the simulation, which includes the time series data of the bank angle response and the maneuver commands. Within the algorithm, the requirement criteria is defined. In this case, there is a limit set to the upper and lower bound for the rise time. Then, the evaluation routine takes place. This routine consist of different steps processing the input data, starting with the extraction of the bank angle signal. Then, the bank angle signal is checked to ensure that the maneuver required for the assessment of this particular requirement was correctly performed. Here only one check is given as an example. If the eVTOL shows no steady state behavior for any reason before the maneuver injection, the evaluation is stopped. The bank angle signal is used to calculate the rise time as the metric of this requirement as specified in the ADS-33E-RPF. Finally, the routine evaluates whether the calculated rise time holds the limits as specified by the requirement criteria. Using the *MATLAB* "verify" method of the "Test Results Object" to set the test verdict allows the integration of the results directly into the *Simulink Test Manager* visualization. Finally, this "Test Results Object" object is provided as an output to the *Simulink Test Manager*.

ID: V600-5234
Input: Test Results Object
Output: Test Results Object
Requirement criteria: $2.5\, s\ to\ 5\, s\ rise\ time$;
Evaluation Routine;
- Extract bank angle signal;
- Run check on signal;

if *No steady behavior* $5s$ *before maneuver* **then**
│ RETURN ERROR MSG;
else
│ CONTINUE EVALUATION;
end

- $rise\ time\ =\ steady\ bank\ angle * 0.632$;
- $2.5s\ <\ rise\ time\ <\ 5s$;
- MATLAB verify function to set pass or fail verdict;

Algorithm 2: Pseudo Code V600-5234 requirement

The V600-5251 requirement is formalized using the same approach. The two RAFs can be included within the *Simulink Test Manager* environment to be evaluated automatically after any test case iteration run.

3.3 Requirement Testing

After formalizing the requirement, a test case has to be defined. To derive a test which maintains high quality results, it is challenging to select proper test data. For this paper, as an example, only a normal operation scenario and the default eVTOL configuration is considered (mass, center of gravity, no failure modes, etc.). The requirements V600-5234 and V600-5251 should be evaluated at all points within the hover envelope of the eVTOL where the required lateral sidestep maneuver can be performed. Therefore, in the following only areas within the hover envelope without initial lateral velocity are considered. The resulting subset of the hover envelope, the envelope points (EVP) are defined as following:

- Altitude: 1000 m
- Lateral Velocity: 0 m/s
- Longitudinal Velocity:
  - EVP4 = -2.5 m/s
  - EVP5 = 0 m/s
  - EVP6 = 2.5 m/s

The maneuver, which is necessary to evaluate the requirement, is stated in the original requirement text and is applied by a full stick deflection in the lateral direction. Therefore, the necessary external test input is a step command in the side stick commanding a lateral velocity. This input is the same for the three iterations.

The three EVPs together with the lateral side step maneuver specify the three test case iterations for the defined test cases:

- Iteration 1: Lateral Sidestep Maneuver at EVP4
- Iteration 2: Lateral Sidestep Maneuver at EVP5
- Iteration 3: Lateral Sidestep Maneuver at EVP6

To finalize the test case, the *Simulink* model containing the ideal controller and plant model, which is discussed in Section 2, is used as SUT and must be referenced below the dropdown menu "System Under Test". The requirement for testing is applied as the implemented RAF below the menu "Custom Criteria".

The same test case definition process is done analog for the requirement V600-5251. As soon as the two test cases are defined, the tests can be executed. The *Simulink Test Manager* stores the outcome of the test as shown in Figure 9. This provides a quick overview of the results and directly highlights failed tests.

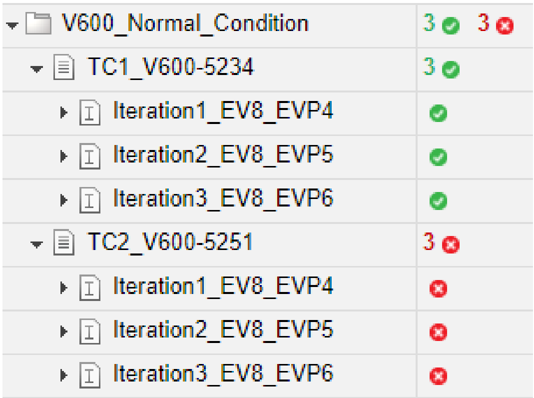

Fig. 9 Test results overview

3.4 Requirement Discussion and Validation

For demonstration purposes, the requirements in this paper were implemented without analysis of the origin of the requirement and the studies which led to the requirement.

Figure 10 shows that requirement V600-5234 is fulfilled at all three considered envelope points.

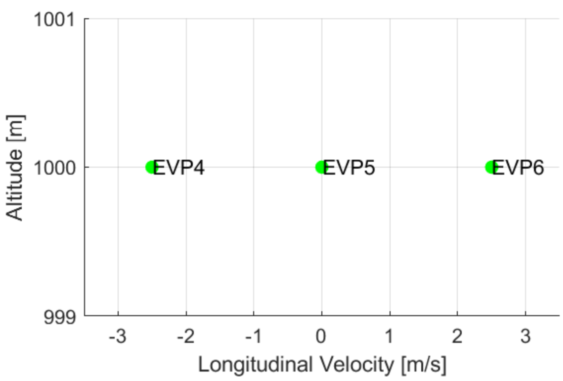

Fig. 10 Results requirement V600-5234

The lateral velocity response of the maneuver at the envelope point 5 (EVP5) is shown in Figure 11. The calculated rise time for this test is 4.48 s and lies within the given bounds. Furthermore, there is no overshoot and an approximately PT1 behavior as stated in the original requirement.

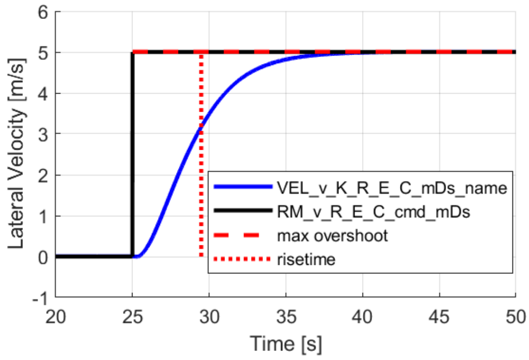

Fig. 11 Lateral velocity response

However, Figure 12 shows requirement V600-5251 which fails at all three envelope points. Figure 13 illustrates the results at envelope point 5 (EVP5). As shown, the attitude overshoot exceeds the requirement criteria of 1% of the steady bank angle of about 1.1°.

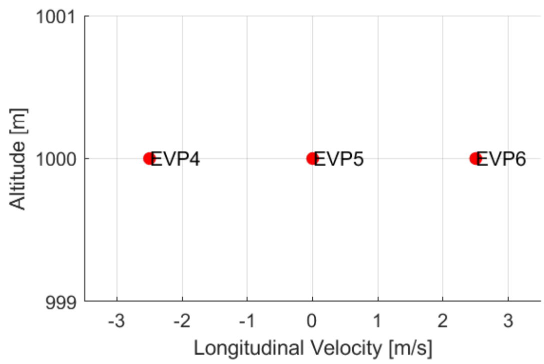

Fig. 12 Results requirement V600-5251

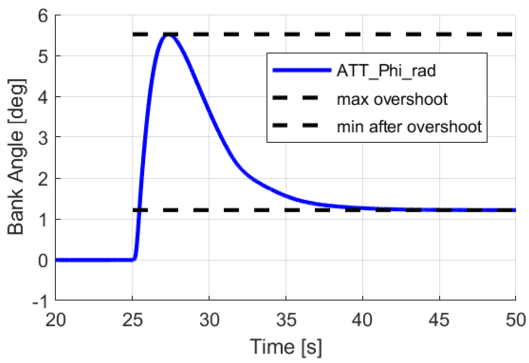

Fig. 13 Bank angle overshoot

In this case, using the validation view, the result indicates that requirement V600-5251 might not be valid for the considered eVTOL configuration. In such a case, the next proposed process step is to analyse the failed test of the requirement to adapt or reject the requirement. Within this paper, we assume that the implemented RAF was done correctly and can be considered as valid. Furthermore, the SUT shall be assumed as valid such that the model represents the physics correctly as well as the controller specification is suitable for this problem.

Based on these assumptions, the analysis requires a closer investigation of the derived requirement V600-5251, the parent requirement V600-4620 and the test results. Exemplarily, this is done in the following.

By having a closer look on the origin of requirement V600-4620 given in ADS-33E-PRF, the detected issue is obvious. The considered multi-copter eVTOL configuration in the hover flight phase is similar to a consumer drone configuration. Such a configuration must use the bank angle to establish a force component in lateral direction. This lateral force component leads to a lateral acceleration and over time to a lateral velocity. Additionally, the steady bank angle during lateral velocities is an equilibrium between the aerodynamic drag and the required lateral force component to maintain the desired lateral velocity.

The translational rate command requirement of ADS-33E-PRF (V600-4620) takes this effect into account based on a wide range of experimental work conducted with rotorcraft. This requirement is a trade-off between precise lateral velocity control (upper limit rise time) and reduction of aprubt bank angle changes (lower limit rise time). Furthermore, pilots do not like the attitude changes comming with the attitude based translational rate command system. [15]

As shown in Figure 13, the lateral acceleration, as defined by rise time requirement V600-5234, implies a large bank angle. However, the steady bank angle is small due to a small lateral aerodynamic drag on this particular eVTOL configuration. This leads to the attitude overshoot and to the violation of requirement V600-5251. Taking the eVTOL configuration into focus, a reason can be the fixed motor mountings which do not allow a decoupling of the rotor plane from the vehicle cell. Taking the derived requirement into focus, it might not be applicable to specify the term "not objectionable" as 1% of the steady state bank angel.

Considerations like the presented ones and additional analysis must be taken into account to adapt or reject the invalid requirements. As soon as the set of requirements is updated, a new development iteration and evaluation of the requirements can be launched.

4. CONCLUSION

The presented process provides a method to validate functional performance requirements like handling qualities requirements based on a formalization of these requirements and a model of the system. As shown, the benefit of this process is the capability to automatically evaluate requirements by *MATLAB* and to identify requirements which cannot be fulfilled. Based on this information, an investigation must be launched to identify reasons for failing the conducted tests. By drawing conclusions of these investigations, the requirements can be improved.

This process can be applied in fields like the eVTOL development where a lack of standards and experience exists, to quickly use existing requirements of aerospace standards for rotorcrafts and aircrafts. The process highlights requirements which might not be applicable to the considered configuration. These requirements can then be refined and adapted to the actual configuration step by step as part of the development work. Furthermore, errors during the requirement derivation and implementation as well as contradictory requirements can be identified and solved. This process finally leads to a set of qualitative improved and valid requirements for the subsequent development.

Looking ahead to next steps of a standard V-model development process, the implemented RAFs and test cases can be used for verification of the developed system against the requirements (verification view compare Section 1). Since the RAFs only depend on signal time series, they can also be used to support virtual integration as well as the real integration process afterwards (e.g. HIL tests and flight test data evaluation). In addition, the requirement tracking capability can be maintained during the whole process based on the established SimPol link.

The presented process using *Polarion*, *SimPol*, RAFs, and RAMs within the *MathWorks* environment allows to evaluate requirements early in the development process and to maintain a set of high quality requirements early on in the development of new eVTOL configurations.

References

[1] Cecconi, M., and Tronci, E., “Requirements formalization and validation for a telecommunication equipment protection switcher,” *Proceedings, Fifth IEEE International Symposium on High Assurance Systems Engineering (HASE 2000)*, IEEE Computer Society, Los Alamitos, Calif, 2000, pp. 169–176. doi:10.1109/HASE.2000.895456.

[2] Schmiechen, K., Hochstrasser, M., Rhein, J., Schropp, C., and Holzapfel, F., “Traceable and Model-Based Requirements Derivation, Simulation, and Validation Using MATLAB Simulink and Polarion Requirements,” *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 01072019, p. 01222. doi:10.2514/6.2019-0500.

[3] Abraham, J., Urban, P., and Canny, P., “Verification and Testing of Embedded Software with Model Based Design,” *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 01072019. doi:10.2514/6.2019-1479.

[4] Kügler, M. E., Rhein, J., and Holzapfel, F., "Timely Flare Initiation during Automatic Landing – A Model Checking Application Example," *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 01072019, p. 43. doi:10.2514/6.2019-0505.

[5] Kügler, M. E., Rhein, J., and Holzapfel, F., "Validation of decision logic of an autoland system for a UAV using model-based safety-assessment techniques," *CEAS Aeronautical Journal*, Vol. 40, No. 6, 2019, p. 43. doi:10.1007/s13272-019-00401-5.

[6] Cimatti, A., Roveri, M., Susi, A., and Tonetta, S., "Formalization and Validation of Safety-Critical Requirements," *Electronic Proceedings in Theoretical Computer Science*, Vol. 20, 2010, pp. 68–75. doi:10.4204/EPTCS.20.7, URL http://arxiv.org/pdf/1003.1741v2.

[7] Lee, C.-C., and Friedman, J., "Requirements Modeling and Automated Requirements-Based Test Generation," *SAE International Journal of Aerospace*, Vol. 6, No. 2, 2013, pp. 607–615. doi:10.4271/2013-01-2237.

[8] Murphy, B., Wakefield, A., and Friedman, J., "Best Practices for Verification, Validation, and Test in Model-Based Design," SAE International400 Commonwealth Drive, Warrendale, PA, United States, 2008. doi:10.4271/2008-01-1469.

[9] USAAMC, Aviation Engineering, "Handling Qualities Requirements for Military Rotorcraft, Performance Specification," , March 2000.

[10] Eigner, M., Roubanov, D., and Zafirov, R., *Modellbasierte virtuelle Produktentwicklung*, Springer Vieweg, Berlin, 2014. doi:10.1007/978-3-662-43816-9, URL http://gbv.eblib.com/patron/FullRecord.aspx?p=1966086.

[11] RTCA, "DO-178C: Software Considerations in Airborne Systems and Equipment Certification," , 2011.

[12] Mirko Conrad, *Modell-basierter Test eingebetteter Software im Automobil: Auswahl und Beschreibung von Testszenarien*, Deutscher Universitäts-Verlag, 2004.

[13] Perfect, P., Jump, M., and White, M. D., "Towards the development of a flight training programme for future personal aerial vehicle users," *CEAS Aeronautical Journal*, Vol. 8, No. 3, 2017, pp. 541–560. doi:10.1007/s13272-017-0255-2, URL https://doi.org/10.1007/s13272-017-0255-2.

[14] Perfect, P., Jump, M., and White, M., "Development of handling qualities requirements for a personal aerial vehicle," *38th European Rotorcraft Forum 2012, ERF 2012*, Vol. 2, 2012, pp. 758–775.

[15] Malpica, C., Theodore, C. R., Lawrence, B., and Blanken, C. L., "Handling Qualities of Large Rotorcraft in Hover and Low Speed," NASA Ames Research Center, 2015.

[16] Gratton, G., *Initial airworthiness: Determining the acceptability of new airborne systems*, 2nd ed., Springer, Cham, 2018.

[17] Bittner, W., *Flugmechanik der Hubschrauber: Technologie, das flugdynamische System Hubschrauber, Flugstabilitäten, Steuerbarkeit*, 4th ed., VDI-Buch, Springer Vieweg, Berlin, 2014. doi:10.1007/978-3-642-54286-2, URL http://gbv.eblib.com/patron/FullRecord.aspx?p=1967803.

[18] US Air Force, "MIL-HDBK-1797: Flying Qualities of Piloted Aircraft: Handbook," , 1997.