GENERATION OF A DATASET FOR SYSTEM IDENTIFICATION WITH VIRTTAC-CASTOR

N. Fezans, C. Deiler, M. S. Roeser
DLR (German Aerospace Center), Institute of Flight Systems
Lilienthalplatz 7, 38108 Braunschweig, Germany

Abstract

VIRTTAC-Castor is the first model of a larger family, VIRTTAC (Virtual Test Aircraft), developed at the DLR Institute of Flight Systems and freely available on GitHub. These models are black boxes in order to put the users in the situation of a flight test engineer who does not have access to the model equations or parameters but can only test the complete aircraft system as a whole. Such models can be used for different purposes and in the GitHub repository the parts that do not depend on the application are separated from the so-called "workbenches", which gather the use-case-specific parts. This paper presents a new workbench for VIRTTAC-Castor, called "GenerateDatasetForSysID", which enables users to generate a dataset that is suited for aircraft system identification purposes. This paper describes the motivations for this workbench and the main capabilities provided. The dataset provided by simply running this workbench with no modification is already well suited for identifying a flight dynamics model of VIRTTAC-Castor. However, the authors consider that system identification is not reduced to the application of parameter estimation algorithms to an exisiting set of experimental data. System identification specialists should be able to define the test points needed to model their system (which in the case of aircraft usually consist of maneuvers, flight points, flight conditions) and therefore the authors encourage the users of this workbench to customize the pre-defined test points or at least to question whether the pre-defined ones are suited for their needs. The aim of this paper is to inform the community of the release of this new workbench and presents its capabilities and foreseen uses.

1 INTRODUCTION

The authors presented in [1–3] a black-box very representative flight dynamics model of a short to medium range airliner, named VIRTTAC-Castor, which is freely provided to the community. It can be downloaded on GitHub[1]. It is released under very permissive licenses in order to restrict as little as possible its use: not a contaminating license, commercial use allowed, etc. The VIRTTAC models are meant to put the users virtually (through simulation) in the position of a flight test engineers who is confronted to a new aircraft and can perform some flight tests with it. The level of representativeness of the VIRTTAC models and their wide availability (Windows 32 and 64 bit, Mac OS Intel 64 bit, and Linux x86-64) should permit to use them for benchmarks e.g. in flight control and in system identification applications. VIRTTAC-Castor a high wing (small anhedral) T-tail airplane configuration powered by two under the wing turbofans, see Fig. 1 and Fig. 2. It can transport 100 passengers on short to medium-haul routes.

Figure 1: Artistic illustration of the VIRTTAC-Castor configuration.

For further details on VIRTTAC-Castor the reader is referred to [1–3], the GitHub repository as well as new papers on VIRTTAC that might have been published since then. In order to ease the use of VIRTTAC models for potentially very different applications, the core functions (i.e. the model itself and the generic / general purpose additions to it) are separated from the use-case-specific parts (code, data, etc.). For each use case a

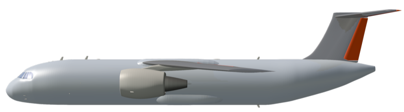
[1] https://github.com/VIRTTAC/VIRTTAC
Contact: VIRTTAC@dlr.de

so-called “workbench” can be added as a subfolder of the “workbenches” folder.

This paper presents a new *workbench* for VIRTTAC-Castor, called “GenerateDatasetForSysID”, with which a dataset for system identification research or benchmarking can be generated. This workbench has been developed with two requirements in mind: 1) it should be able to provide dataset that can already be useful for system identification purposes and 2) remain as simple as possible in order to be a good starting point and example for VIRTTAC-Castor users.

(a) top view

(b) front view



(c) side view

Figure 2: Artistic illustration of the VIRTTAC-Castor configuration, multiview projection.

System identification [4–7] is the scientific discipline and process aiming at finding a suitable system description (usually model) which matches a set of observations made of a real or reference system. The term system identification is related to the theory of dynamic systems and it is often implicitly assumed that the considered system is a dynamic system. In the computer science community such a task is usually considered as a machine learning task, namely a regression problem (i.e. in a family of the supervised learning problems). Whilst this other terminology is correct, the present hype around deep learning techniques tends to give the impression that the machine learning techniques are a newer and better way of doing system identification, which is obviously wrong: both terms are similarly old and simply originate from different scientific communities. Generally speaking, the term machine learning is broader than the term system identification as machine learning includes techniques that are applied for other use-cases. However for the type of applications considered here, the preference for one or the other terminology is usually rather linked either to the scientific community or to the willingness to surf on the current machine learning and artificial intelligence hype. In the following, the authors will only use the term “system identification” but the dataset produced with this workbench can certainly be used by machine learning / computer science researchers.

The paper is structured as follows: Section 2 gives an overview of the various steps used in system identification. Section 3 introduces the two main use-cases that the authors foresee for the herein introduced VIRTTAC-Castor workbench. Then the various steps of the system identification process that are covered by the workbench are detailed: the definition/selection of maneuvers (section 4), the definition of the experimental conditions (section 5), and finally section 6 explains how the virtual flight test data is generated and stored to disk for further use.

2 QUAD-M METHODOLOGY

The goal of system identification is to create a mathematical model based on experimental data and that is representative enough for the intended purpose. The process for creating such a mathematical model commonly consists of the following steps:

1. **Design of experiments:** This step is usually one of the main driver for the “cost” (time, overall duration of the experiments, computational burden, etc.) of the method and for the quality of the final results (precision of the estimated model parameters).

2. **Perform the experiments and measure the outcomes:** During this step the experiment, as designed in the previous step, is performed. This might involve risks for safety and be very costly (e.g. due to flight hours), which is why the experiments must be well designed. The outcomes must be measured and saved. On a real system various physical parameters of interest need to be measured, which often means that the system needs to be instrumented with a series of sensors which are otherwise not necessarily required for operating the system.

3. **Select a mathematical model structure/form:** This step is a main driver for the quality of the model obtained at the end of the system identification regarding the match between the model and the experimental data gathered as well as regarding the

validity of the model for situations that differ from the test points and the used excitation signals. If the model includes too few parameters or the wrong ones, then the match with the experimental data will be poor and so will also be the model validity in general. When a good match on the experimental data is obtained, it is crucial to make sure that the model validity for other situations is also good. A good match on the experimental data but poor validity otherwise is usually indicating that the model was "overfitted", i.e. overly tuned for these particular data. This typical happens when the model used is either unnecessarily complex with respect to the real system behavior (e.g. too many parameters) or when the experimental data used is not sufficient to properly tune the model. This latter case can result from an insufficient amount of data or from an insufficient coverage of the data (either in terms of range of operating conditions or in terms of excitation of the various dynamic modes of the system).

4. **Choose and apply an identification method:** In this step the system identification specialist chooses **a)** a mathematical formulation for expressing the quality of the match / goodness of fit between the experimental data and the model (i.e. typically a cost function to be optimized), **b)** an numerical algorithm which permits to find the model / model parameters giving the best possible match with the experimental data.

In the aircraft system identification community the term "Quad-M principle" is regularly used to refer to these four steps, summarizing each of these steps with a word starting with the letter 'M': **M**aneuver, **M**easurements, **M**odel, and **M**ethod. The herein presented workbench (GenerateDatasetForSysID) only covers the first two steps: Maneuver and Measurements. It allows the generation of a dataset that is representative of the data that would have been collected on a real aircraft at the end of the measurement (second) step. These data are meant to be the starting point for VIRTTAC users wanting to develop, test, or demonstrate system identification methods and models.

3 USE-CASES

Whilst the herein presented workbench, "GenerateDatasetForSysID", can be used in many different ways, two main use-cases are foreseen by the authors and are presented hereafter. This workbench is also meant to be a good and fairly simple example for more complex tasks than just performing a single simulation as in the "OpenLoopManeuver" workbench. The VIRTTAC-Castor model runs typically 20 to 50 times faster than real time, depending on the computer, such that many simulations and also long simulations can be performed in a very reasonable time, which is crucial for some of the foreseen use-cases.

Research on System Identification Methods

From the various steps of the Quad-M methodology (see section 2), the authors expect that the test or application of various system identification methods will be one of the main use for this workbench or of the dataset produced with it. The datasets produced can be used with any identification method, regardless whether these methods are in the time or in frequency domains and whether they consider an "output-error" or an "equation-error" criteria. Of course, if the methods needs very specific inputs signals, as it can be the case for instance with the method presented in [8], the corresponding input signals must be used at the data generation step.

Research on Maneuver Design

Another foreseeable use for VIRTTAC-Castor and more specifically for this workbench is research on methods for designing optimal maneuver / excitation signals for airplane identification. In such work, see for instance [9–17], the general idea is usually to optimize the ratio between the quality of the identified model and the amount of time needed for the experiments (i.e. to perform all required maneuvers). For this, good excitation signals must be designed and, in order to save time, optimized signals often involve simultaneous excitations of many or all actuators / control surfaces.

Marchand [9] has shown that the aerodynamic coefficients used for the aircraft modeling are attributed to a specific frequency range, which should therefore be included in the input and excitation signal. Therefore it is useful to design input signals for system identification that contain the frequency range of the expected eigenfrequencies.

There are several ways to include the desired frequencies in flight test maneuvers. The classical type of maneuvers are multistep inputs attributed to the control surfaces. A widely used input e.g. is the 3-2-1-1 multistep input, originated at the DLR in the late 70's, [10]. This input combines the frequency range expected for the longitudinal motion of the aircraft. Similar inputs, as the 1-2-1 can be applied on the aileron, including the frequency expected for the lateral-directional motion and resulting in a bank to bank maneuver. Doublets can also be applied easily for this purpose on the rudder or on the other control surfaces. Multistep inputs have the great advantage that they can be performed fairly easily by a pilot, i.e. without requiring the use of an automatic control unit. As long as the sequence of step is not too complicated, the pilot can perform the abrupt switches in the amplitude command with no real external help (display, time reference, etc.) precisely enough for the intended purpose. More complex multistep maneuvers (e.g. too long or too complex to be memorized by the pilot) might require the use of an automatic control unit. The main disadvantage of those manually flown maneuvers is that pilots can hardly perform proper inputs on more than one axis at once. Between two maneuvers the aircraft must be stabilized again such that performing a few multiaxis maneuvers also permits to save a significant amount of time compared to performing more single axis maneuvers.

Frequency sweeps are also applied for system identification to include a wide range of input information for the aircraft motion. They are very time consuming and limited by the ability of the pilot. The use of automated frequency sweeps has been common practice for decades for the flight testing of new large airplanes. They are mainly flown to identify flutter and flexibility effects of higher frequency inputs on the aircraft motion.

Multisine inputs for aircraft parameter estimation were proposed by [12] and have been used successfully in numerous studies since then. They can be applied to multiple control surfaces and are therefore mutually orthogonal in time and frequency. The fact that the excitations consists only of pure sine excitations with known frequencies can also be exploited in order to reduce the computational burden with some frequency-domain system identification techniques [8].

A new method to design flight test maneuvers for aircraft parameter estimation was proposed by [17]. This method is able to combine several criteria used for perturbation signals for system identification, by the specification of a time-frequency plane and the inverse wavelet transform. Using this method, the engineer can define time localizations in the signal and link it to a frequency band of interest. This permits a time segregation of the signal for each frequency included, without the need for a continuous excitation. This method was also applied for multiaxis maneuvers and showed promising results for an Airbus A320 case study [18]. The idea is to generate rich multiaxis excitation signals as in the case of the multisine excitations but that are however to analyze for the system identification specialist since not all frequency bands are excited at the same time on all axes.

4 DESIGN OF EXPERIMENTS: DEFINITION/SELECTION OF MANEUVERS

When it comes to identifying a dynamical system, it is crucial to generate or collect data that not only show the steady-state behavior of the system but also put its dynamic behavior into relief. For this the experimental data collected via the measurements must include a sufficient level of excitation of the relevant dynamic modes of the system.

In an airplane system identification flight test campaign this is usually done by means of specific maneuvers / sequence of pilot inputs. In fly-by-wire airplanes (which is nowadays the case for most airliners and complex business jets), computerized excitation can be used instead of pilot inputs. The use of computerized excitation permits to use complex excitation signals and also to apply them with a high level of precision and reproducibility on each individual actuator or even on several actuators simultaneously. The optimization of such signals has been also a topic for research for several decades ([11–14, 16, 19–23]). Excitation from external disturbances (e.g. turbulence) can also be useful but are usually, by definition, harder to predict and reproduce. In the herein presented workbench the user can select series of maneuvers to be used to excite the airplane in open loop at various flight points.

4.1 CLASSICAL MANEUVERS AVAILABLE

In the herein presented workbench, a series of typical airplane system identification maneuvers can be used directly. These standard maneuvers are defined in the common part of the VIRTTAC-Castor directory structure (in the directory named “maneuvers”). Users can customize these maneuvers or define entirely new maneuvers to be used. At the time of the writing the existing/pre-defined “maneuvers” are:

- steady flight with no excitation (control surfaces at their respective trimmed positions)
- elevator 3-2-1-1 command
- elevator 1-1-2-3 command
- rudder doublet
- aileron multistep
- flap transition
- thrust pulse
- sequential deployment of all spoilers

For the sake of illustrating the behavior of VIRTTAC-Castor, Fig. 3 shows the response to a 1-1-2-3 elevator command maneuver for a heavy VIRTTAC-Castor (Mass 56,000 kg) at 10 000 ft (standard atmospheric conditions) and a calibrated airspeed of $V_{\mathrm{CAS}} = 280\,\mathrm{kt}$. The 1-1-2-3 maneuver is a variation of the 3-2-1-1 input sequence proposed in [4], where the higher frequencies are at the beginning preventing the aircraft to drift from the trim condition directly at the beginning of the maneuver. On this figure, the left two columns shows the “inputs” of VIRTTAC-Castor (open loop) and the right three columns show most of measurements available. Note that the inputs are here the measured position of the actuators and not the commanded deflection. For system identification purposes, the real deflection of the control surface is often taken as it is what really causes a change in the aerodynamic forces and moments and consequently impacts the aircraft motion.

4.2 OTHER AVAILABLE MANEUVERS

The new VIRTTAC SysID workbench includes a fairly complex multiaxis time-frequency-separated elevators, ailerons, and rudder command excitations. This maneuver is almost the same than the maneuver proposed

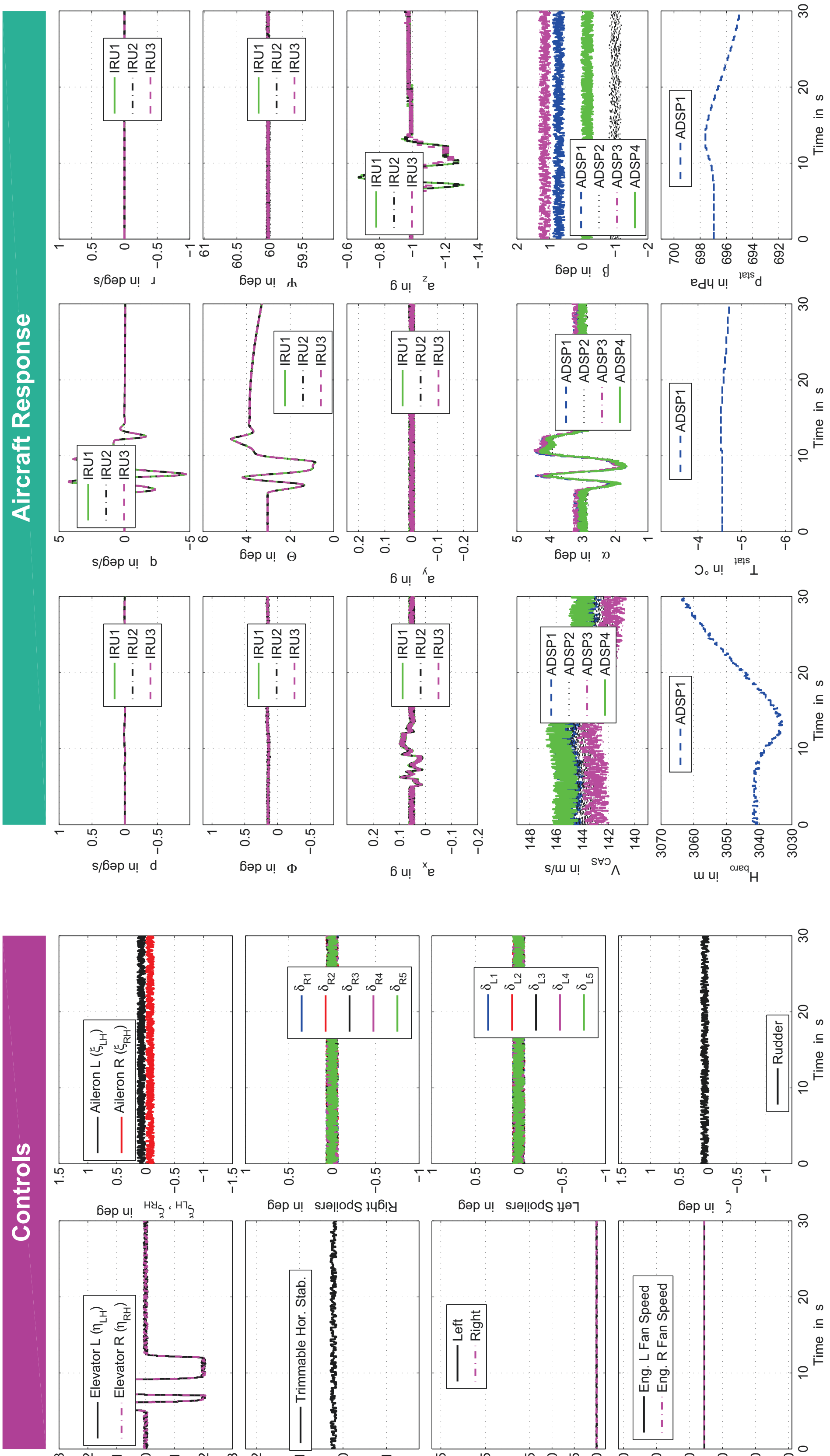


Figure 3: Response of VIRTTAC-Castor to a 1-1-2-3 longitudinal maneuver.

in [17] but without the low frequency pitch excitation through the horizontal stabilizer. A second type of multiaxis maneuver is also included: the time-frequency-separated multistep maneuver on elevators and ailerons that is presented in [24]. The definition of this second type of maneuver is shown in Fig. 4.

These two maneuvers have been added to the set of predefined maneuvers that the users can select from and the results when performed with VIRTTAC-Castor are shown respectively in Fig. 5 and Fig. 6 (same flight point and representation as for the maneuver of Fig. 3).

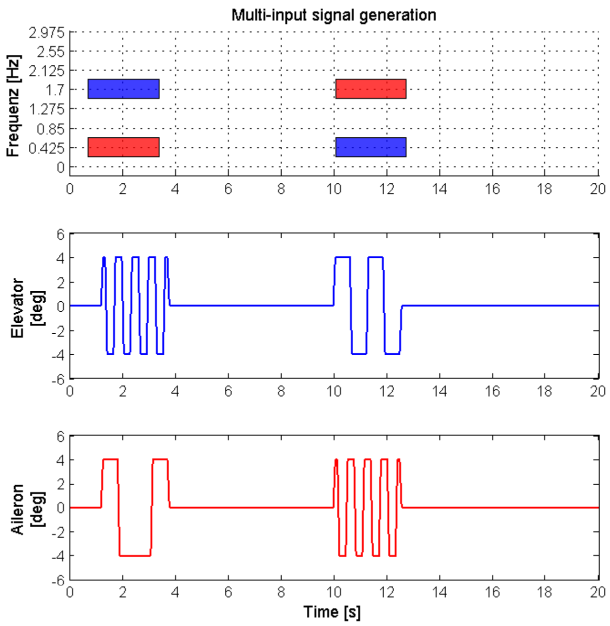


Figure 4: Multi-input signal generation scheme.

Further types of maneuvers will be integrated in the near future, such as multiaxis multisine inputs (see for instance [22, 23, 25, 26]) or the customizable multiaxis time-scale-segregated inputs from [17, 24].

4.3 CUSTOMIZATION/OPTIMIZATION OF MANEUVERS BY THE USERS

Users can define their own maneuvers with no restrictions: highly dynamical or high amplitudes commands might not be followed very well by the actuators as they have physical limits, but there is no further restriction. If some users define excitation signals / maneuvers that might be of interest for others: we, the authors of this paper and current maintainers of VIRTTAC, encourage them to submit these maneuvers for inclusion in the maneuver list. The preferred way for this would be to use a pull-request to the VIRTTAC GitHub repository, but users who do not use GitHub much might also send them to the authors via the contact email address.

5 DESIGN OF EXPERIMENTS: SELECTION OF FLIGHT CONDITIONS

As the previous examples already have implicitly indicated, the definition of an excitation signal alone is not sufficient to define an “experiment”. The conditions under which this experiment shall be performed also need to be specified. This includes (but is not limited to) the altitude, the flight speed and the aircraft configuration (e.g. gear up or down, high lift: clean, flaps 1, flaps 2, etc.).

5.1 FLIGHT POINTS

For each maneuver a flight point can be freely chosen by the users, as long as this flight point is part of the flight envelope and that the aircraft in its current configuration can be trimmed for steady horizontal flight[2] and satisfies the current envelope limits (see [3] and the VIRTTAC-Castor online documentation for details on these limits). These limits are in some regions more strict than the envelope that could be trimmed as many flight test crews (test pilots and test engineers) would not necessarily be comfortable with performing aggressive maneuvers close to the stall speed for 1g flight: V_{s1g}.

In order to show how users can define a series of maneuvers and flight conditions, the basic example provided in the `GenerateDatasetForSysID` workbench includes a series of maneuvers to be performed at different altitudes (10 000, 20 000, and 30 000 ft) and speeds ($V_{CAS} = 220\,\text{kt}$).

5.2 AIRCRAFT CONFIGURATION AND WEIGHT & BALANCE

The aircraft weight and balance characteristics can also be specified to some extent. The aircraft empty weight and mass distribution is usually fairly well known (e.g. from weighting the aircraft). The loading of the aircraft (additional equipment, computers, etc.) as well as the additional masses and their distribution of the flight test crew is relatively well known or can even be precisely adjusted before starting the flight. However, due to operational constraints (airspace, air traffic control, other traffic, weather, etc.) the amount of fuel that will be present in the aircraft tanks when a specific maneuver will be executed can only be roughly specified in practice. As a consequence, the users of VIRTTAC-Castor only provide relatively vague desired weight categories at the time of the test. VIRTTAC-Castor then picks randomly a weight and balance configuration that is compatible with the empty aircraft weight and balance characteristics, the loading provided, and adjust the amount of fuel on board within reasonable bounds with the aim

[2] Steady horizontal assumed to be the starting of all maneuvers for now, and it is currently for instance not possible to perform a maneuver during a steady turn. This restriction will be removed very soon in a combined update of the model and of this workbench.

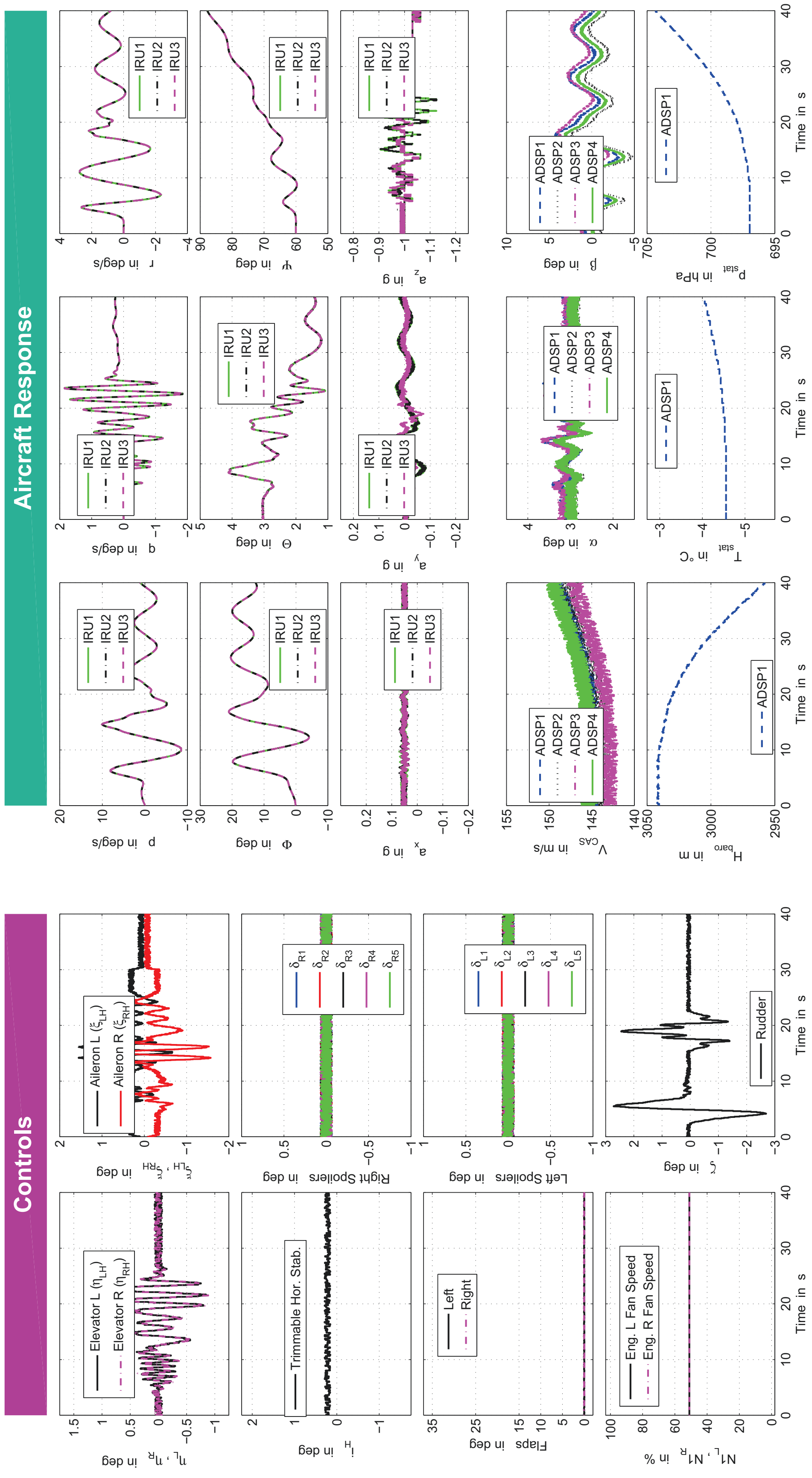


Figure 5: Response of VIRTTAC-Castor to the fairly complex time-frequency-separated multi-axis maneuver (elevators, ailerons, rudder) maneuver derived from the one proposed in [17].

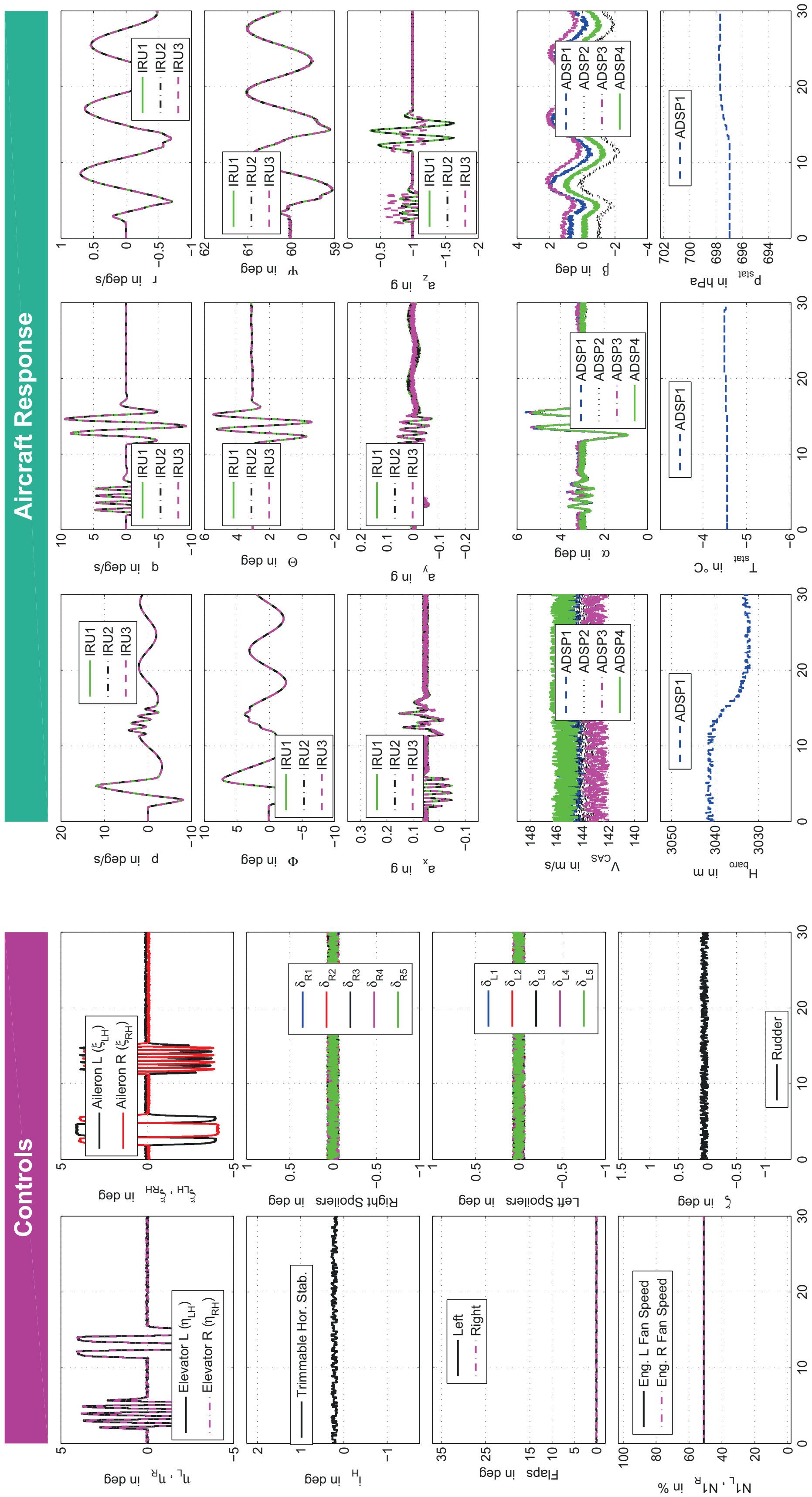


Figure 6: Response of VIRTTAC-Castor to the time-frequency-separated multi-axis maneuver (elevators, ailerons) maneuver proposed in [24].

of satisfying the specified mass category. The "reasonable" bounds mainly consists in ensuring a) that at least a minimum amount of fuel is in the fuel tank and b) that the total mass is not too close or exceeding the maximum take off weight (MTOW). The minimum fuel amount shall ensure that the aircraft can flight back to the home base and could still be then rerouted safely to an alternate airport. Based on the MTOW and the loaded aircraft weight, the maximum amount of fuel at take off can be determined. Additionally, a minimum amount of fuel to reach the test area is required and it is also assumed that statistically not all maneuver can be performed immediately after reaching the test area. All in all, the implemented logic should provide an amount of fuel that is realistic and with the variations as in a real flight test.

5.3 POTENTIAL FUTURE EVOLUTIONS OF THE SPECIFICATION OF THE TEST FLIGHT CONDITIONS

Some tests might require specific conditions which currently cannot always be selected or not easily enough. For instance, the tests might require a specific track or heading (with respect to true or magnetic North). In other cases, it might be required to be flying with no drift angle (either pure head wind or pure tail wind) or with cross wind.

Another example would be to perform a test at a specific density altitude, i.e. accounting for the atmospheric conditions in the area at that particular time such as the difference in temperature with respect to the standard ISA conditions.

Plenty of new options could be integrated regarding the specification of the flight conditions for each test, but most of these options are having lower priorities than many other VIRTTAC-Castor desired features, such that no significant evolution with this regard should be expected in the near future. Missing features regarding the specification of flight conditions can be submitted to the VIRTTAC support email address for future consideration (VIRTTAC@dlr.de).

6 DATASET GENERATION AND STORAGE FOR FUTURE USE

After the user defined the virtual flight test program as a list of maneuvers with the associated test conditions, either by modifying or replacing the provided script `VIRTTAC_InitListOfTestPoints.m`, he/she can use a script called `VIRTTAC_GenerateDataset.m` to perform all these test maneuvers in sequence. For each test point, this script initializes the aircraft at the corresponding conditions (with more or less precision, depending on how easy these conditions can be met in practice) and simulates the maneuver defined by the corresponding input commands. All available sensors provided in output of the simulation model are recorded and the results are saved in individual files: one file per simulated maneuver.

Eventually, once all tests of the flight test program have been performed the script stops and the generated dataset can be found on the disk in a subfolder of the workbench folder. Within the workbench folder a folder named `datasets` has been created (if it did not exist yet) and inside of this folder a separated folder is created at each run of the algorithm. The name of that folder is generated automatically based on the current date and time to prevent from potentially overwriting previously generated datasets. For instance if the script is started on Sept. 3rd, 2019 at 19:50:44 (i.e. 44 seconds after 7:50 pm), the simulation results will be placed in the folder `datasets/2019_09_03_19h50m44s`.

The `VIRTTAC_GenerateDataset.m` and `VIRTTAC_InitListOfTestPoints.m` scripts give a simple example of what can be done with this workbench. It performs a series of fairly standard system identification maneuvers at various flight points across the envelope: the same maneuvers for three different altitudes (10 000 ft, 20 000 ft, and 30 000 ft at ISA atmospheric conditions), three different airspeeds (220 kt CAS, 250 kt CAS, and 280 kt CAS) and different weights (see mass cateogries). The maneuvers are repeated once each time: the results will differ slightly, due to randomness in the trimming and in the weight and balance property (the amount of fuel on board is never perfectly the same). The basic script contains 180 simulations, which are typically performed in a few minutes (depending on the computer and on the MATLAB/Simulink version used).

Some scripts enabling to load the generated data are also provided. They permit to load in a common structure the results of several (possibly all) tests at the same time with no risk of name collision. It is also foreseen to be able to filter the data to be loaded based on a user-defined predicate function. This enables, for instance, users to specify very easily that they want to consider only the tests for which the measured angle of attack has exceeded a given value (e.g. $\alpha < 7°$) while the bank angle was small (e.g. $|\Phi| < 10°$). For performance reasons with large datasets it might be useful or even required to store the list of the individual tests to be taken into account between two working sessions. Whilst the authors do recommend intensive VIRTTAC users to use such preselection and indexing of the datasets, no such functionality is provided with this workbench (yet).

7 SUMMARY

This paper presents a new workbench, called "GenerateDatasetForSysID", for VIRTTAC-Castor which enables users to generate, via simulation, a dataset that is representative of the one that would be obtained through flight tests and that is suited for system identification purposes. This paper describes the motivations for this workbench and the main capabilities provided. The dataset provided by simply running this workbench with no modification is already well suited for identifying a flight dynamics model of VIRTTAC-Castor. However, the authors consider that system identification is not reduced to the application of parameter estimation algorithms to an existing set of experimental data. System identification specialists should be able to define the test points needed to model their system (which in the case of aircraft usually consist of maneuvers, flight points, flight conditions) and therefore the authors encourage the users of this workbench to customize the pre-defined test points.

The authors encourage the community to try this new workbench and to send their feedback on the contact email address: VIRTTAC@dlr.de. Direct contributions to the GitHub, for instance for adding some interesting maneuvers that they may have defined/optimized, is very much welcome, as long as the provided code is also released under the very permissive MIT licence.

References

[1] C. Deiler and N. Fezans, "VIRTTAC - a family of virtual test aircraft for use in flight mechanics and GNC benchmarks," Proceedings of the 2019 AIAA Scitech - Intelligent Systems Conference, (San Diego, CA, USA), Jan. 2019. AIAA 2019-0950.

[2] N. Fezans and C. Deiler, "Inside the virtual test aircraft (VIRTTAC) benchmark model: Simulation architecture," *Simulation Notes Europe (SNE)*, vol. 29, no. 1, pp. 1–12, 2019. DOI: 10.11128/sne.29.on.10461.

[3] C. Deiler and N. Fezans, "Virtual flight testing with VIRTTAC-Castor: New capabilities and flight envelope definition," in *Proceedings of the 2019 German Aerospace Congress (DLRK)*, (Darmstadt, Germany), Sept.-Oct. 2019.

[4] R. V. Jategaonkar, *Flight Vehicle System Identification: A Time-Domain Methodology, 2nd Edition*. Progress in Astronautics and Aeronautics, AIAA, 2015. DOI: 10.2514/4.102790, ISBN: 978-1-62410-278-3, eISBN: 978-1-62410-279-0.

[5] M. B. Tischler and R. K. Remplr, *Aircraft and Rotorcraft System Identification, 2nd Edition*. AIAA Education Series, AIAA, 2012. DOI: 10.2514/4.868207, ISBN: 978-1600868207.

[6] E. Morelli and V. Klein, *Aircraft System Identification: Theory and Practice, 2nd Edition*. Sunflyte Enterprises, 2016. ISBN: 978-0997430615.

[7] L. Ljung, *System Identification: Theory for the User, 2nd Edition*. Prentice Hall, 1998. ISBN: 978-0136566953.

[8] E. A. Morelli and J. A. Grauer, "Practical aspects of the frequency domain approach for aircraft system identification," in *Proceedings of the 2018 AIAA Aviation / Atmospheric Flight Mechanics*, (Atlanta, GA, USA), 2018. AIAA 2018-3477.

[9] M. Marchand, "Untersuchung der Bestimmbarkeit von Flugzeugderivativen aus Flugversuchen," Tech. Rep. DLR-IB 154-74/32, DLR Institut für Flugmechanik, 1974.

[10] R. Koehler and K. Wilhelm, "Auslegung von Eingangssignalen für die Kennwertermittlung," Tech. Rep. DLR-IB 154-77/40, DLR Institut für Flugmechanik, 1977.

[11] E. Plaetschke and G. Schulz, "Practical input signal design," *AGARD Lecture Series No. 104 - Parameter Identification - Paper 3*, 1979.

[12] E. A. Morelli and V. Klein, "Optimal input design for aircraft parameter estimation using dynamic programming principles," *AIAA Atmospheric Flight Mechanics Conference*, 1990. DOI: 10.2514/6.1990-2801.

[13] K. Godfrey, *Perturbation signals for system identification*. Prentice Hall international series in acoustics, speech, and signal processing, Prentice Hall, 1993. ISBN: 978-0136564140.

[14] C. Seren, F. Bommier, L. Verdier, A. Bucharles, and D. Alazard, "Optimal experiment and input design for flight test protocol optimization," in *Proceedings of the 2006 AIAA Atmospheric Flight Mechanics Conference and Exhibit*, (Keystone, CO, USA), Aug. 2006. AIAA 2006-6280.

[15] C. Seren, *Optimisation du protocole d'essais pour l'identification de la dynamique du vol*. PhD thesis, École Nationale Supérieure de l'Aéronautique et de l'Espace - ENSAE/Supaéro, Toulouse, France, 2007.

[16] P. Lichota, K. Sibilski, and P. Ohme, "D-optimal simultaneous multistep excitations for aircraft parameter estimation," *Journal of Aircraft*, vol. 54, pp. 747–758, Feb. 2017. DOI: 10.2514/1.C033794.

[17] M. S. Roeser, "Multi-axis maneuver design for aircraft parameter estimation," in *Proceedings of the 2018 German Aerospace Congress (DLRK)*, (Friedrichshafen, Germany), Sept. 2018.

[18] DLR, "Project VicToria - Virtual Aircraft Technology Integration Platform." https://www.dlr.de/as/en/desktopdefault.aspx/tabid-11460/20078_read-47033/, Aug. 2018.

[19] W. Wells and S. Ramachandran, “Multiple control input design for identification of light aircraft,” *IEEE Transactions on Automatic Control*, vol. 22, no. 6, pp. 985–987, 1977. DOI: 10.1109/TAC.1977.1101653.

[20] M. Marchand, “Untersuchung der Bestimmbarkeit der flugmechanischen Derivative des CCV-Versuchsträgers F-104 G,” Tech. Rep. DLR-IB 154-77/12, DLR Institut für Flugmechanik, 1977.

[21] E. A. Morelli and V. Klein, “Optimal input design for aircraft parameter estimation using dynamic programming principles,” in *Proceedings of the 17th Atmospheric Flight Mechanics Conference*, (Portland, OR, USA), August 20th-22th 1990. AIAA-90-2801.

[22] E. A. Morelli, “Multiple input design for real-time parameter estimation in the frequency domain,” *IFAC Proceedings*, vol. 36, no. 16, pp. 639 – 644, 2003. 13th IFAC Symposium on System Identification (SYSID 2003), Rotterdam, The Netherlands, 27-29 August, 2003.

[23] E. A. Morelli, “Flight test maneuvers for efficient aerodynamic modeling,” *Journal of Aircraft*, vol. 49, pp. 1857–1867, Nov. 2012. DOI: 10.2514/1.C031699.

[24] M. S. Roeser, “Flight test and identification from automatic inputs for parameter estimation on a glider,” in *Proceedings of the 2019 German Aerospace Congress (DLRK)*, (Darmstadt, Germany), Sept.-Oct. 2019.

[25] J. Grauer and E. Morelli, “Real-time frequency response estimation using multi-sine inputs and recursive Fourier transform,” in *Proceedings of the 2012 AIAA Atmospheric Flight Mechanics Conference*, Aug. 2012. AIAA 2012-4409.

[26] P. Lichota, J. Szulczyk, M. B. Tischler, and T. Berger, “Frequency responses identification from multi-axis multisine manoeuvre,” in *Proceedings of the 2019 AIAA SciTech / Atmospheric Flight Mechanics Conference*, Jan. 2019. AIAA 2019-1322.

A CONDITIONS OF USE

The following information is provided with the aim of allowing the readers of the paper to know the terms of the used licenses without having to download VIRTTAC first. Since these licenses might evolve over time and differ from file to file, the license information provided with each VIRTTAC model applies: the following information is of indicative nature only.

Source files
As of now, anyone can download and use the VIRTTAC models. Any part of VIRTTAC provided in source form (e.g. MATLAB .m files or Simulink models) is subject to the very permissive MIT license:

Other files: executable, binaries, binary data
All non-disclosed code and data files (e.g. executable, dynamic/static libraries, binary files, etc.) are licensed under the Creative Common Attribution-NoDerivs 4.0 Generic license (CC-BY-ND 4.0). A human-readable summary is provided hereafter: please refer to the official license text for the legally binding text. Note that any attempt to disassemble the binary code, binary data, executable, or dynamic/static libraries provided is hereby considered as a derivative and is consequently hereby prohibited, even if not shared. A normal use of VIRTTAC for its intended purpose does not require such operations and therefore users will normally not be affected by this restriction.

B VIRTTAC DOWNLOAD AND CONTACT INFORMATION

To provide the models of the VIRTTAC family to the community, a GitHub repository was created. This repository is located at https://github.com/VIRTTAC/VIRTTAC and will be updated if necessary due to new model developments of aircraft within the VIRTTAC family.

For any questions on VIRTTAC-Castor, the VIRTTAC family or for general support concerning the VIRTTAC simulation, please use the following VIRTTAC email address:

VIRTTAC@dlr.de.