CEAS Aeronaut J
DOI 10.1007/s13272-014-0110-7

ORIGINAL PAPER

Data parser approaches for (online) parameter estimation

Christoph Deiler

Received: 4 November 2013 / Revised: 12 March 2014 / Accepted: 14 April 2014

Abstract Finding maneuvers for parameter estimation in flight data records is a laborious task and traditionally performed post-flight on ground. Two different data parser approaches to automatically detect these maneuvers in flight are presented. Both methods search the control input signals for significant changes that correspond to test maneuvers. The first algorithm is based on the signal time derivative of the input signal whereas the second method uses a fast orthogonal wavelet transform. Both algorithms are tested with flight test data recorded with the DLR research aircraft ATTAS. Performance results are compared and potential problems when applying the parsers to other data are discussed. Results indicate that both methods are applicable in an online parameter estimation tool. The intention of the work in this paper is to develop an algorithm with a high level of automation for in-flight use, but both approaches could also be applied to offline flight data mining problems.

List of symbols

a, b, c	Polynomial coefficients
a_m	Approximation coefficient
b	Bias
d_{crit}	Wavelet coefficient threshold
d_m	Wavelet coefficient
f	Signal
f_0	Sampling rate
FOWT	Fast orthogonal wavelet transform
g, h	Wavelet filter functions
i, k, n	Indices
l_{min}	Minimum maneuver length
l_{sep}	Minimum maneuver separation
l_{win}	Wavelet parser window size
m	Number of decompositions
MRA	Multiresolution analysis
n_{seg}	Number of maneuver data segments
OEM	Output error method
Φ	Scaling function
Ψ	Wavelet function
t	Time
Δt	Sampling time
t_{calc}	Parser calculation time
t_{fore}	Segment forerun time
t_{over}	Segment overrun time
Δt_{par}	Parser time delay
u	Control input signal
$\dot{u}_{\mathrm{crit}}$	Control input rate criterion
$\dot{u}_{\mathrm{zero}}$	Control input rate zero threshold
z_{zero}	Response zero threshold

This paper is based on a presentation at the German Aerospace Congress, September 10–12, 2013, Stuttgart, Germany.

C. Deiler (✉)
German Aerospace Center (DLR), Institute of Flight Systems, Lilienthalplatz 7, 38108 Braunschweig, Germany
e-mail: christoph.deiler@dlr.de

1 Introduction

For parameter estimation, which is a common method in the aircraft simulation model development, the right selection of flight test data is essential. The data must contain enough information about the dynamic behavior of an aircraft. Therefore, specific test maneuvers are performed to excite the dynamics of interest. The resulting information should permit to adjust a simulation model in order that it can reproduce the observations correctly. Without these "excitation maneuvers" as part of the system identification process (as defined in [1]) a satisfactory parameter estimation of a dynamic aircraft model would not be possible. The developed models are typically used for flight dynamics and performance investigations, flight control system design, loads assessment or training simulators.

Before the parameter estimation process can start, useful data segments containing the desired information resp. maneuvers in the records have to be found. This task is usually "manually" performed by an engineer. In both cases, online (in-flight) and offline (post-flight) parameter estimation, this task is often very time consuming and sometimes difficult. Four different cases should be distinguished for the data mining task:

Case 1: Low automation level, small data set (e.g. online flight test evaluation with user interaction, or offline evaluation of short flight test campaign).
Case 2: High automation level, small data set (e.g. online flight test evaluation without user interaction, or time-critical offline evaluation of flight test records).
Case 3: Low automation level, large data set (e.g. user driven evaluation of complete flight test campaign with hours of data records).
Case 4: High automation level, large data set (e.g. highly automated evaluation of complete flight test campaign with hours of data records).

The first two cases can be considered as online parameter estimation scenario. In flight test, it is possible to use a system with low automation level (case 1), which requires a high user interaction. In current applications for example, the engineer marks the performed maneuvers in the time histories manually, which is relatively quick and easy. The maneuvers are always monitored in-flight and the results in the aircraft behavior are evaluated, but the additional task of the definition of useful data segments in the time histories could be faced by a highly automated system (data parser), which could relieve the flight test personnel on board (case 2). Additionally, during a flight test not every maneuver is performed for testing purposes. Flight point changes or even level turns to stay in the assigned airspace require valuable flight test time. From time to time, these maneuvers contain valuable information about the dynamic aircraft behavior and could also be used for parameter estimation. A highly automated system would permit detecting these maneuvers as well.

Besides, such a user-independent, highly automated online parameter estimation system can be used in various other applications, such as aircraft health monitoring. Any change of the aircraft's flight performance or dynamic behavior could be an indication of altered aerodynamics or engines, and therefore an economical and safety aspect. This kind of application cannot rely on user intervention, and should therefore be almost fully automated.

In case of online parameter estimation in a flight test, the estimation results shall be obtained within a suitable time frame, for example less than one minute, after the maneuver has been performed. This way, the maneuver can be repeated, if necessary. Due to this time requirement, the data segments containing the maneuvers must be found by a highly automated system (case 2) in only a fractional amount of time, so that there is enough time left for parameter estimation.

Further, a highly automated process could also accelerate the evaluation of the data recorded during one or a few test flights, so that the flight data evaluation is able to cope with tight and challenging flight test campaign schedules (case 2). An (almost) automated flight data preprocessing such as the one proposed hereafter would also provide a valuable support while evaluating a large dataset when the properties that are sought in the data were not already marked or identified. This could typically happen after a particular event occurred during the flight test campaign leading the engineers to check for possible previous occurrences within the already recorded flight data. The proposed solutions help then to convert this typical case 3 problem into a case 4 one.

The herein pursued idea to create a completely highly automated system is driven by the online applications (cases 1 and 2). Even though the proposed approaches have a high potential for cases 3 and 4, which are moreover interesting and of great practical relevance, their use in that context is not considered in this paper.

At DLR a new online parameter estimation tool [2] is currently under development (see Fig. 1) that will provide the necessary parts for an automated operation (case 2). The incoming measurements have to be searched for useful

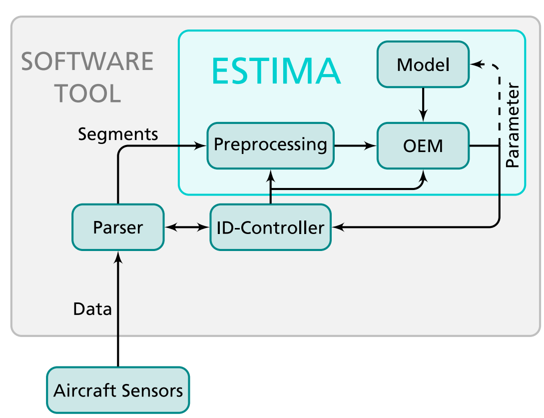

Fig. 1 Scheme of the online parameter estimation tool (cf. [2])

data segments containing excitation maneuvers. The "ID Controller" triggers the parameter estimation, if a new useful data segment is found, and could be used to monitor and to evaluate the results.

For flight test implementation the "ID Controller" could be replaced by an existing graphical user interface (GUI), which allows the flight test engineer to interact with the tool. The parameter estimation is performed by an adapted version of DLR's software tool ESTIMA, which provides various estimation methods, whose output error method (OEM) implementation in the time domain was used in the context of this work, even though various parameter estimation methods are available. With all other parts of the tool working in a first version, the data parser was the only missing module in the past. The development of two potential parser algorithms is presented in this paper.

A first step in the automated search for maneuvers is to algorithmically rebuild the engineer's procedure: looking for changes in the in- and output signals that correspond to the excitation maneuvers, which is similar to a "template matching" method in pattern recognition [3]. For typical maneuvers like multistep inputs (doublets or 3–2–1–1 signals) or bank-to-bank maneuvers [1], as shown in Fig. 2, changes in the control signals and angular rates are easy to find in time history plots. But it is more difficult to create a simple algorithm, which is able to detect these special maneuvers and skips all other inputs. The first approach (rate criterion parser) presented in this paper searches for significant changes of the input signal time derivatives and compares them with changes in the aircraft's response. If they match the predefined pattern, it is assumed, that the detected changes correspond to the searched maneuvers.

The second wavelet based approach uses a time-frequency signal analysis (wavelet parser). The method looks for signal changes of a certain frequency band, which in the present case consists of filtering out higher frequencies such as measurement noise, while only looking for changes within the frequency bandwidth corresponding to the aircraft's dynamics. Therefore, a correlation of time and frequency analysis is useful to find the desired data segments. Multiresolution analysis (MRA) methods like the fast orthogonal wavelet transform (FOWT) [4] provide these correlations for a measured signal, and a simple implementation using Haar-wavelets is presented in this paper. These MRA methods are widely known in the field of digital image and signal processing and there has also been work

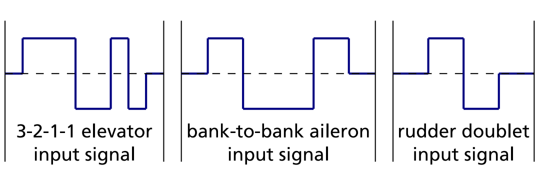

Fig. 2 Example of various parameter estimation maneuver inputs

using MRA for aircraft resp. system identification purposes [5]. In [6] wavelets are used for instance with neural networks to find characteristics in a given signal matching a predefined pattern. The approach developed at DLR is kept much simpler and only uses the wavelet transform with no complex postprocessing. The difference between Fourier transform over a finite time horizon and wavelet transform is given in [4] for a general case. No further evaluation for the herein considered use case was made so far.

In this paper, both proposed algorithms are evaluated using flight test data containing excitation maneuvers that were conducted with the DLR research aircraft VFW 614 ATTAS in 2003.

2 Rate criterion parser

Significant signal changes matching a special pattern are very helpful in finding the desired data segments containing the maneuvers. In case of a flight test this means e.g. to search for significant changes of angular rates due to control surface inputs. One requirement for excitation maneuvers is that the inputs are large enough to cause a significant aircraft reaction. Especially when created synthetically, these inputs also can be considered more abrupt than those during standard maneuvers (e.g. change of flight condition resp. trim point). This can be considered true even with pilots who fly with rather aggressive maneuvers. If for example a highly active flight controller is used in the loop, it would probably be more difficult to deal with, but this case is not considered here.

The previous observation about rate changes leads to the definition of a criterion based on a threshold $\dot{u}_{\mathrm{crit}}$ on the input rate, to discriminate excitation maneuvers from the rest of the flight. When the input rate falls below the defined threshold value, it is assumed that the input is finished and the remaining part of the reaction is further monitored. In addition, a zero threshold for the response (angular rate) z_{zero} is necessary. As long as the response signal exceeds this threshold, the maneuver is considered to be still running and only when the signal falls below z_{zero} after the last input, the end of the maneuver is assumed. The value of z_{zero} has to be chosen high enough to account for sensor errors and excitations due to atmospheric disturbances. The search for multistep inputs as shown in Fig. 2 requires considering the duration of the steps. After a change in the input signal the algorithm must wait for Δt_{par} and check whether another change is following or not. If no further change is detected within Δt_{par}, the detected maneuver start and end time are reported for further processing. A suitable choice of Δt_{par} prevents incomplete detection but still guarantees an acceptable time delay between the maneuver performance and processing.

In an online implementation, the data parser should find the desired data segment shortly after a maneuver is performed to satisfy the above given time requirement. Therefore, every new measured data point should be processed as soon as it is available. The parser algorithm was developed as follows:

1. Parser core:
   (a) Differentiate control input signal.
   (b) Compare absolute value of input signal time derivative $|\dot{u}|$ with predefined threshold $\dot{u}_{\mathrm{crit}}$.
   (c) Set change indication.
2. Monitor response signal and evaluate change indication to set maneuver start and end time.
3. Check detected maneuver data segment for minimum maneuver length $l_{\min}$ and maneuver separation l_{sep}.

In the algorithm, a change indication is used to highlight the segments, where the input has exceeded the threshold $\dot{u}_{\mathrm{crit}}$, and to activate the output monitoring. Moreover, a detected segment is only assumed to be a parameter estimation maneuver, if it exceeds a certain minimum maneuver length $l_{\min}$. This procedure neglects short fast inputs for stabilization and focuses on the desired specific maneuvers and thus reduces false detections. The maneuvers are usually well separated from each other, so a minimum separation l_{sep} is used to further reduce the probability of a false detection.

Calculating the time derivative of a noisy time discrete signal is no easy task. For such a signal, the time derivative calculated from simple difference quotients might contain high peaks due to the overlaying noise. To represent the basic signal's time derivative and not the noise induced changes, another method has to be applied. In the considered datasets no real issue with strong measurement noise was experienced. But if the measurements do not have this high quality, various techniques could then be used to alleviate this problem. Polynomial differentiating approaches as presented in [7] allow to compensate for noise and to calculate an approximation of the time derivative. By considering additional points on each side of the local point k, the signal is smoothed and the noise influence reduced. Using five data points, two in front and behind the point k, a second order polynomial in time

$$f[k] = f(t[k]) = at^2[k] + bt[k] + c, \quad \forall k \in \mathbb{N} \tag{1}$$

has the property, that the time derivative of f at the central point ($t[k] \equiv 0$) equals b. Using a least squares fit, it can be shown according to [7], that the signal's time derivative at $t[k]$ can be approximated by:

$$\dot{f}[k] = \frac{1}{10 \cdot \Delta t}\Big(-2 \cdot f[k-2] - f[k-1] + f[k+1] + 2 \cdot f[k+2]\Big). \tag{2}$$

This is a very simple approach to approximate the time derivative of a noisy, measured signal. By only considering two additional points around k the time derivative value is obtained shortly after the measurement. Especially with the regard to the online implementation, this simple approach was chosen. But it is still a compromise between noise filtering abilities and computational speed, and for signals including a lot of noise, this could not be sufficient. Therefore the implementation of a differentiating low pass filter [8], which includes more points around k and could further reduce the measurement noise, would be necessary.

In Fig. 3 the basic idea of the rate criterion parser is shown. For an example signal, which is comparable to one step of a flight test maneuver input signal, the time derivative and its absolute value including the input rate criterion $\dot{u}_{\mathrm{crit}}$ and the zero threshold $\dot{u}_{\mathrm{zero}}$ are plotted.

A moving window of 5 points is used to calculate the signal derivative as part of the parser algorithm. The following configuration parameters are necessary to operate the proposed algorithm:

- Control input rate criterion/threshold $\dot{u}_{\mathrm{crit}}$.
- Control input rate zero threshold $\dot{u}_{\mathrm{zero}}$.
- Response zero threshold z_{zero}.
- Minimum maneuver length $l_{\min}$.
- Minimum maneuver separation l_{sep}.
- Parser time delay Δt_{par}.

Both zero thresholds $\dot{u}_{\mathrm{zero}}$ and z_{zero} are necessary to compensate for small oscillations around the zero value in the flight data. If the input signal rate and response signal have both fallen below $\dot{u}_{\mathrm{zero}}$ resp. z_{zero}, the end of a maneuver is assumed. The zero thresholds should be chosen high enough to account for measurement noise, but small to enough to not corrupt the detection.

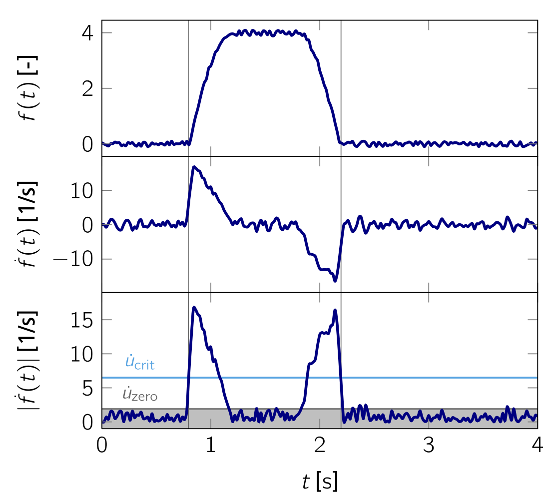

Fig. 3 Example signal and time derivative [see Eq. (2)] with thresholds to illustrate the rate criterion parser decision algorithm

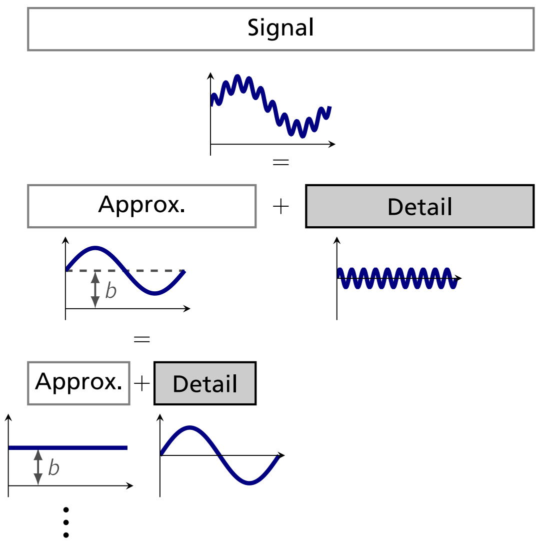

Fig. 4 Signal decomposition with wavelet transform

For an online implementation, the parser must be connected to the parameter estimation tool (Fig. 1). To use the same tool chain as presented in [2] the above defined parser algorithm was programmed in C/C++.

3 Wavelet transform

The development of wavelet transforms was a major step in time-frequency decomposition of continuous and discrete signals (MRA) [4]. Only a few remarks required to understand the proposed parser approach will be provided here and for a deeper insight in wavelet theory and applications the reader is referred to [4].

Fast orthogonal wavelet transform (FOWT) allows decomposing a signal in several successive steps. Each step leads to a decomposition of the signal into

- Its approximation or general trend (lower frequency part).
- Its details (higher frequency part).

The obtained approximation is undersampled (factor 2) between each step, so that the next decomposition generates signal characteristics with a lower frequency than the previous step. The general process is shown in Fig. 4. The decomposition preserves time-wise correlation, which means, that information about the signal characteristics at a certain time is still available. Furthermore, each signal can be recomposed with an inverse process using the obtained approximations and details.

For the work in this paper, the Haar-wavelet [9] was chosen as the wavelet base. No other wavelet types were investigated. The wavelet base influences the possibility of recovering signal properties from the decompositions without recomposing. Basically, the wavelet base function is correlated with the signal at different frequencies and magnitudes (similar to a Fourier transform). The Haar-wavelet, shown in Fig. 5, is one of the simplest wavelet forms and defined through its scaling function $\Phi(t)$ and wavelet function $\Psi(t)$ as

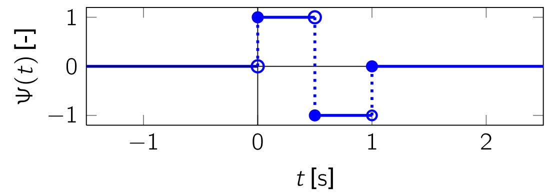

Fig. 5 Haar-wavelet function

$$\Phi(t) = \begin{cases} 1 & \text{if} \quad 0 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

and

$$\Psi(t) = \begin{cases} 1 & \text{if} \quad 0 \leq t < 0.5 \\ -1 & \text{if} \quad 0.5 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The FOWT of a discrete signal f is given by a filter bank with the equations below. In the filter bank, the signal f is assumed to be the top level approximation a_0

$$a_0[n] = f[n], \tag{5}$$

and each new decomposition delivers the coefficients a_m (approximation) and d_m (detail), where $m \in \mathbb{N}$ is the number of decomposition steps.

$$a_{m+1}[n] = \sum_k a_m[k] \cdot h[k - 2n], \tag{6}$$

$$d_{m+1}[n] = \sum_k a_m[k] \cdot g[k - 2n]. \tag{7}$$

Denote that a_m and d_m are the inner product of f with the scaling $\Phi_{m,n}$ and wavelet function $\Psi_{m,n}$ (at level m, with a shift of n periods)

$$\begin{aligned} a_m[n] &= \langle f, \Phi_{m,n} \rangle \\ d_m[n] &= \langle f, \Psi_{m,n} \rangle. \end{aligned} \tag{8}$$

Each new decomposition (see Fig. 4) is calculated by a convolution of the approximation a_m with the wavelet filters g and h, which only needs a few operations. In reverse, the approximation a_m is easily gained from the approximation a_{m+1} and the detail d_{m+1} with the same computational effort.

For the given Haar-wavelet base, the filter function coefficients $g[k]$ and $h[k]$ have only non-zero values for $k = 0$ and $k = 1$. The coefficients are given by [4]

$$g[0] = \frac{1}{\sqrt{2}}, \quad g[1] = -\frac{1}{\sqrt{2}},$$
$$h[0] = \frac{1}{\sqrt{2}}, \quad h[1] = \frac{1}{\sqrt{2}}. \tag{9}$$

The Haar-wavelet base has the advantage that the coefficients of the signal decomposition allow to make a statement about the signal characteristics.

This correlation is shown in the following example. The noisy signal (sample rate 50 Hz, 200 samples) in Fig. 6 can be decomposed using FOWT given by Eq. (5), (6), (7). The detail wavelet coefficients d_1, shown as bars, are directly correlated to the signal noise. Note, that the bars are only a time wise illustration of the coefficients and no reconstructed signal. The oscillations of the example signal are perfectly visible in the coefficients due to the usage of simple Haar-wavelet base. With further decompositions, the large-scale signal changes are visible in the detail wavelet coefficients and the fourth decomposition d_4 give an indication of the basic signal changes. Using these facts, a data parser approach is developed, which automatically detects major signal changes out of the detail coefficients.

4 Wavelet parser

The FOWT used in this approach is no recursive method and thus needs a certain signal length for the decomposition to the desired level m. Larger data sets result in a longer calculation time for the wavelet transform, which could pose a problem concerning fast processing in an online implementation (see Sect. 1). To reduce the calculation time, only a small part of the latest measured data is used, similar to a window with the size l_{win} moving over the signal. As previously shown in Fig. 6, the time resolution of the detail wavelet coefficients gets coarser with more decomposition recursions, which results in wider bars. The window l_{win} should be wide enough for more than one bar. To adapt the windows to the low-pass behavior, they should be consequently scaled by e.g. the subsampling factor 2^m.

To find a maneuver in the time history by searching for a significant input signal change, the detail wavelet coefficients d_m must be processed. As mentioned above, the detail coefficients of the first decompositions contain the higher frequencies, which can be considered as measurement noise. With the further decompositions, the basic signal characteristics appear in the detail coefficients. Therefore, it is necessary but no easy task to find the correct decomposition level for the parser, which allows finding the desired maneuvers, but minimizes the computational effort. If the detail coefficient values d_m exceed d_{crit}, there is a possibility, that the detected signal change corresponds to a excitation maneuver. However, these signal characteristics are not limited to excitation maneuvers but could also result from a trim point change. Thus the detail coefficient is no reliable indicator by itself.

This is the point, where the correlation of time and frequency has a significant effect on the approach. Combining all detected signal changes to a maneuver results in a certain maneuver length. Introducing a minimal maneuver length l_{min}, similar to the approach given in Sect. 2, prevents false detections. If the detected maneuver is shorter than l_{min}, it is neglected, otherwise selected. Another problem is the time between two inputs in one maneuver. It must be considered that for some flight test maneuvers, like a bank-to-bank maneuver, a change in the control signal may not occur for several seconds. Therefore a time delay resp. waiting time Δt_{par} is also proposed for this approach to assure that the maneuver is completed.

Overall, the parser implementation contains the following parts:

1. Parser core:
   (a) Perform wavelet transform and decompose signal m times.
   (b) Compare absolute values $|d_m[n]|$ of detail wavelet coefficients with predefined threshold d_{crit}.
   (c) Set change indication.
2. Evaluate change indication with respect to the location in the timeline to set maneuver start and end time.
3. Check detected maneuver with respect to minimum maneuver length.

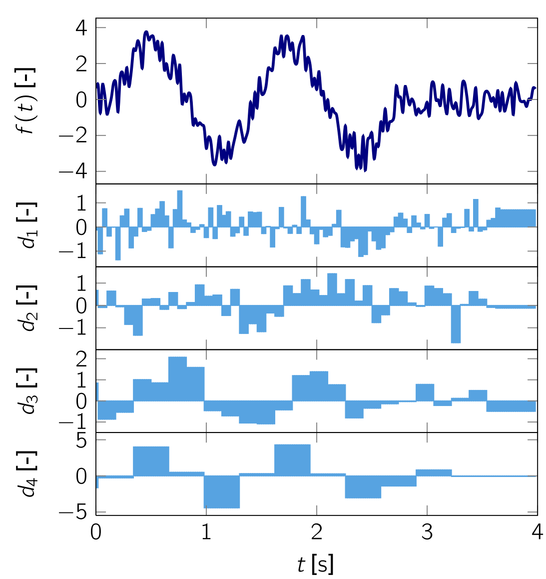

Fig. 6 Example signal and detail coefficients $d_{1\ldots4}$ of the wavelet transform

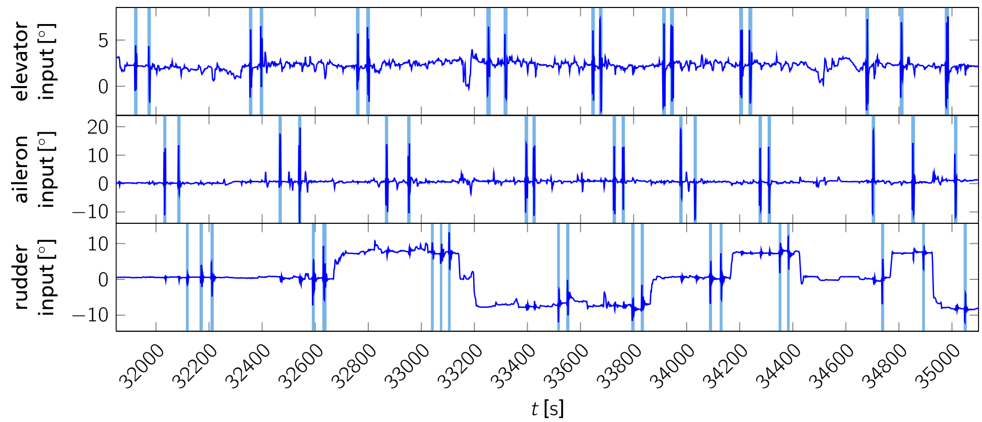

Fig. 7 Primary control inputs from ATTAS flight test data

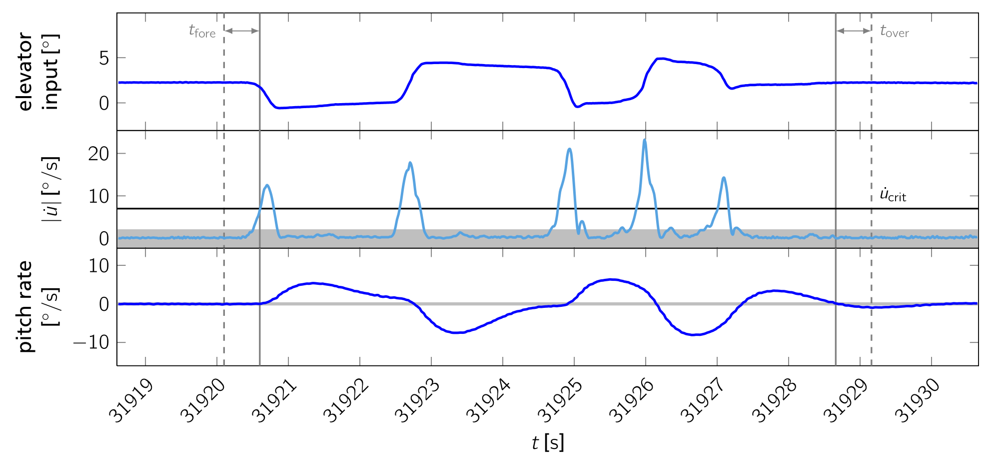

Fig. 8 Rate criterion parser example—detected elevator maneuver

For the wavelet parser, the following parameters have to be specified:

- Decomposition level m.
- Window size l_{win}, default $(2 \cdot 2^m + 2)$ samples.
- Threshold d_{crit}, scaled with the power of 2^m.
- Minimum maneuver length l_{min}.
- Parser time delay Δt_{par}.

The minimum maneuver length l_{min} is here defined as a number of samples, because it was found useful to have a value comparable to l_{win}. Multiplication with Δt results in a value for l_{win} in seconds, as used in the rate criterion algorithm.

The algorithm is also programmed in C/C++, so it can be used as a part of the online parameter estimation tool.

5 Parser test using flight data

Both approaches have been compared using flight test data recorded with the former DLR research aircraft ATTAS (VFW 614), which was retired in 2012. The data was collected during a system identification flight campaign in 2003, and contains 17 elevator 3–2–1–1 inputs, 17 bank-to-bank maneuver aileron inputs and 19 rudder doublet inputs. The time histories of the primary control deflections are shown in Fig. 7, and the segments containing the desired maneuvers are highlighted with vertical bars.

For both parser approaches, a graphical user interface was created to ease the usage during flight. It was assumed that it could be difficult to interact with a command line program during highly dynamic maneuvers and so the choice of the interfaces seemed reasonable. The parser approaches are connected to the online parameter estimation tool and the flight data are loaded into a fast accessible shared memory data store, which creates a setup similar to a flight test.

5.1 Flight data evaluation

First, the rate criterion parser is used to search for data segments. The utilized configuration parameter settings are given in Table 1 for the three primary controls and the corresponding angular response signals. In addition to the above described configuration parameters (see Sect. 2), a maneuver forerun t_{fore} and overrun time t_{over} is defined to

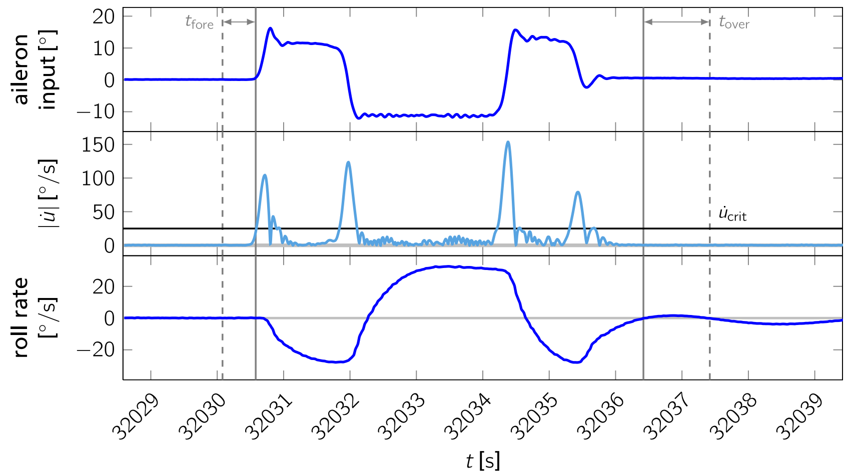

Fig. 9 Rate criterion parser example—detected aileron maneuver

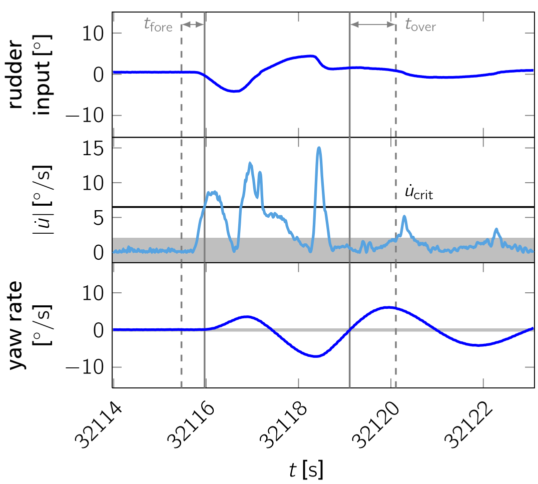

Fig. 10 Rate criterion parser example—detected rudder maneuver

Table 1 Rate criterion parser: configuration parameter setting for example flight data analysis

	Elevator	Aileron	Rudder
$\dot{u}_{crit}$	7.0 °/s	25.0 °/s	6.5 °/s
$\dot{u}_{zero}$	2.0 °/s	2.0 °/s	2.0 °/s
z_{zero}	0.2 °/s	0.2 °/s	0.2 °/s
l_{min}	0.75 s	0.75 s	0.75 s
l_{sep}	0.1 s	0.1 s	0.1 s
Δt_{par}	4.0 s	4.0 s	4.0 s
t_{fore}	0.5 s	0.5 s	0.5 s
t_{over}	0.5 s	1.0 s	1.0

extend the segments. This is needed, because the parser results start at the signal change and for parameter estimation, a short time segment with a steady trim point in advance is useful. If the dynamic response to an input is not completely included in the detected maneuver data segment, the overrun time t_{over} can be used to extend the time segment and to obtain additional information for the estimation.

With these configuration settings the rate criterion parser finds all 53 maneuvers marked in Fig. 7. Moreover, time plots of one resulting segment in each axis are given in Figs. 8, 9, 10. The plots show the control input, the time derivative of the control input and the corresponding angular response. It is visible, that the algorithm can easily find the significant peaks in the control input derivative with the correct configuration. The additional times t_{fore} and t_{over} are marked as the difference between the directly detected maneuver segments (solid lines) and the adapted ones (dashed lines). Moreover, the grey areas in the input rate and response signal time histories are used to highlight the zero thresholds $\dot{u}_{zero}$ and z_{zero}. Depending on the control signal, the rate criterion parser needs about 0.3 s of calculation time on regular desktop computer to process the complete 4.655 s of recorded flight data.

Secondly the wavelet parser approach is tested with the same flight data. In Table 2, the configuration parameters are given for the three primary control input signals. A decomposition level of three was found to be enough to extract the significant changes in the input signals. The window size l_{win} is set to its above described m-dependent default value, which leaves d_{crit} as the only parameter to be chosen (values given in Table 2). As mentioned in Sect. 4, the detail wavelet coefficient threshold is scaled with the power of 2^m to compensate the increase of d_m with each decomposition level m, which results in the given values for d^8_{crit}. With the given configuration parameters, the wavelet parser also detects all 53 maneuvers. Control input and angular rate response signals of the example data segments are given in Figs. 11, 12, 13 with the corresponding detail wavelet coefficients d_3. Because this approach is currently used without any response monitoring, the detected segments are shorter than the ones shown

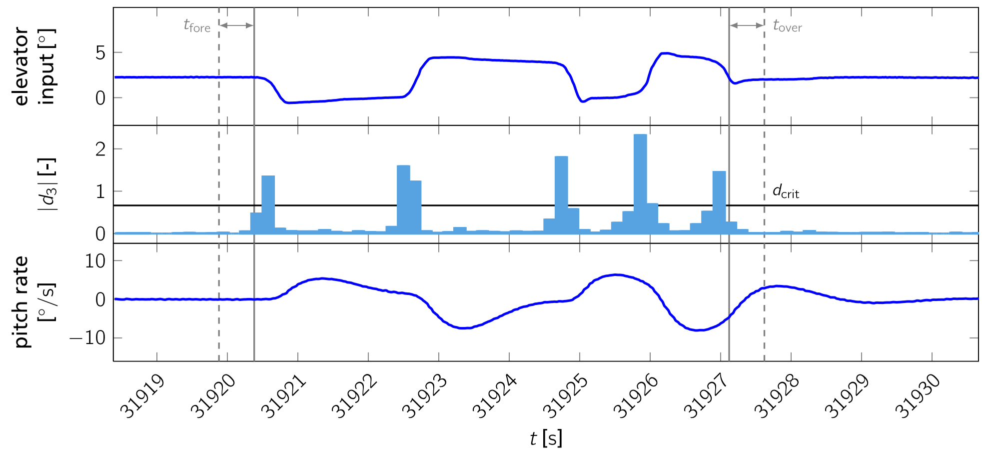

Fig. 11 Wavelet parser example—detected elevator maneuver

Table 2 Wavelet parser: configuration parameter setting for example flight data analysis

	Elevator	Aileron	Rudder
m	3	3	3
d_{crit}	0.95	1.19	0.96
(d^8_{crit})	(0.663)	(4.021)	(0.721)
l_{win}	18	18	18
l_{min}	30	30	30
Δt_{par}	4.0 s	4.0 s	4.0 s
t_{fore}	0.5 s	0.5 s	0.5 s
t_{over}	0.5 s	1.0 s	2.0 s

in Figs. 8, 9, 10. The wavelet parser needs approximately 1.5 s to process the complete data on a state of the art multicore computer for each input signal.

In comparison, both approaches are able to find all the desired data segments, but with different computational effort. The rate criterion parser has a more complex logical built up due to the response signal monitoring, while the wavelet parser is more time consuming. The differences in the found data segments are minor concerning the start and end times. During the test of both approaches, several problems with the parser parameter tuning appeared and must be further discussed.

5.2 Discussion of results

The choice of the configuration parameters is essential for finding the desired maneuver data segments. The appropriate choice for the zero threshold value $\dot{u}_{zero}$ of the rate criterion parser is highly dependent on the measurement noise level. The flight test data used in the tests has a very low noise level. For test data with higher noise levels, the value of $\dot{u}_{zero}$ has probably to be increased, which means, that the difference between the zero threshold $\dot{u}_{zero}$ and criterion threshold $\dot{u}_{crit}$ will possibly vanish, and the parser does not work properly. In an offline implementation, where it is possible to use proper filter algorithms to reduce signal noise, this would not be a problem, but for the online implementation, this has to be kept in mind unless the differentiation method is changed. Furthermore, the response zero threshold z_{zero} depends on the current atmospheric conditions and the aircraft's dynamic behavior, because any additional excitation caused by atmospheric disturbances must be neglected through this parameter.

The rate criterion parser approach delivers larger data segments than the wavelet parser because of the response signal monitoring. To include the aircraft's dynamic response after the control input is faded away was actually only meant to support the detection using the rate criterion. It was assumed that the peaks in the input rate signal (see for example Fig. 8) would not be sufficient to properly detect the maneuvers. But because of the utility of the additional information of the aircraft response to the parameter estimates, the response signal monitoring is more than a simple support to the algorithm.

Concerning the wavelet parser, the decomposition level m is the most critical parameter for a well working system. In combination with the threshold d_{crit}, most problems with a possible false detection can be solved by adjusting these values. But with an increasing decomposition level, the computational effort increases as well because of the higher number of necessary calculations. To tune the parser for a minimization of the time delay between the performance of the maneuver and its detection, this point has to be considered. All other values (cf. Table 2) are necessary for the parser algorithm but have only a minor influence on its behavior. The wavelet parser results are narrow data segments, which end right after the control input signal's last change during a maneuver. The parser only uses the

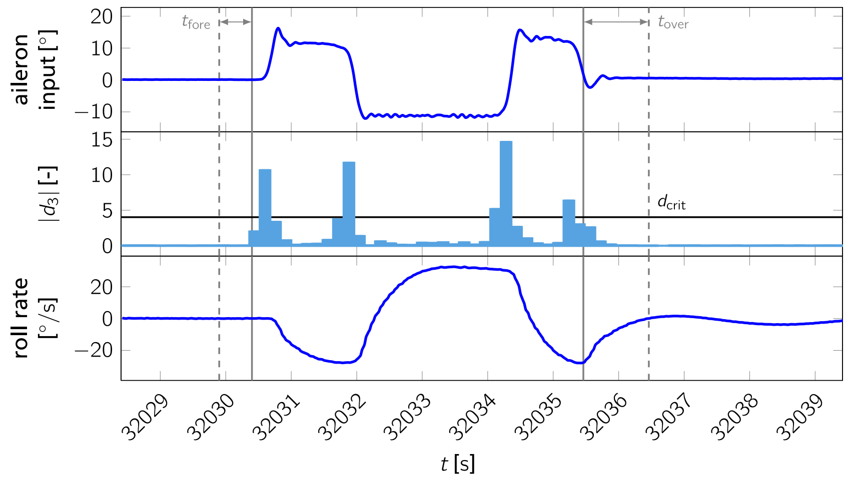

Fig. 12 Wavelet parser example—detected aileron maneuver

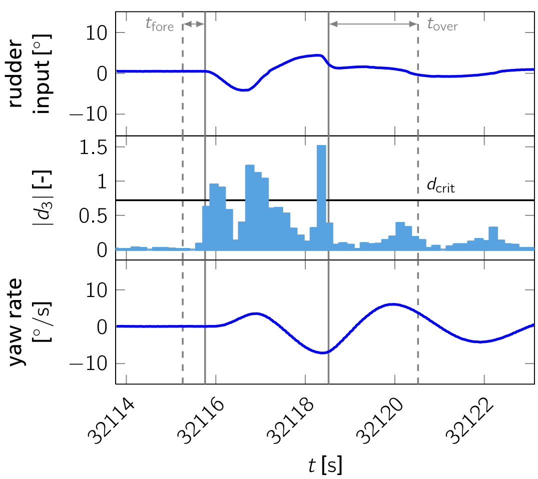

Fig. 13 Wavelet parser example—detected rudder maneuver

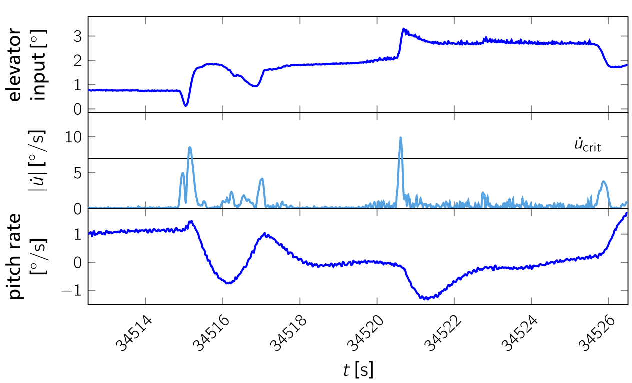

Fig. 14 Rate criterion parser example—discarded elevator maneuver

control inputs and the neglected information about the aircraft's dynamic behavior must be added manually. Therefore t_{over} has to be adapted to the aircraft's characteristics to still consider the response to the input signal change. But this approach seems to be more user-friendly and easier to configure than the rate criterion parser.

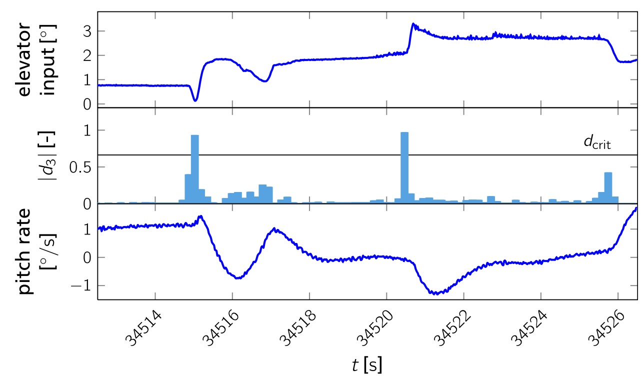

Fig. 15 Wavelet parser example—discarded elevator maneuver

Moreover minimum maneuver length l_{min} and minimum maneuver separation l_{sep} must be tuned for the approaches, so that the found data segments contain only the intended maneuvers and no other inputs. If the maneuver length is chosen too large, some fast maneuvers may remain undetected.

The parser should only detect desired maneuvers and neglect all other inputs. To visualize this behavior for both algorithms, the elevator input and pitch rate response in a turn maneuver is plotted in Figs. 14 and 15. For both algorithms, the threshold $\dot{u}_{\mathrm{crit}}$ resp. d_{crit} is passed and thus the change indication is set. But because no further input follows within Δt_{par}, the minimum maneuver length l_{min} is not reached and the maneuver is therefore disregarded by both algorithms.

To further compare both approaches, a sensitivity study regarding the main threshold parameters $\dot{u}_{\mathrm{crit}}$ resp. d_{crit} was performed. All other parameters were fixed at their values from Table 1 resp. Table 2. Figure 16 shows the results for the rate criterion parser. For a value of $\dot{u}_{\mathrm{crit}}$ between 5.6 and 7.9 °/s all 17 desired maneuvers are correctly detected (blue area, dashed line marks previously used value $\dot{u}_{\mathrm{crit}} = 7.0$ °/s). For lower values of $\dot{u}_{\mathrm{crit}}$ the number of

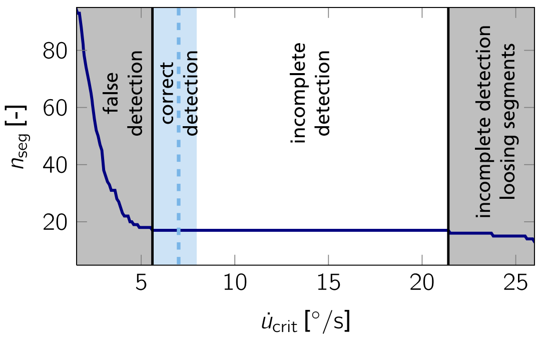

Fig. 16 Sensitivity analysis of rate criterion parser to different threshold values, elevator inputs

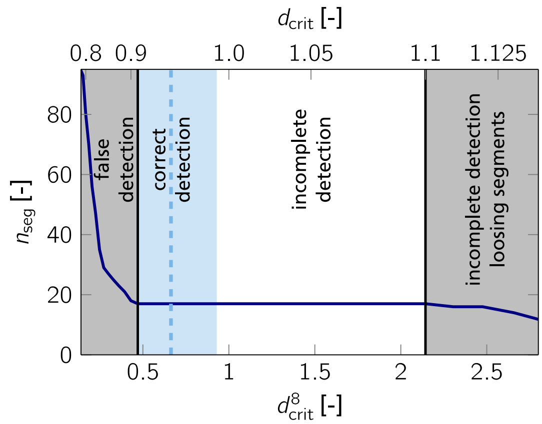

Fig. 17 Sensitivity analysis of wavelet parser to different threshold values, elevator inputs

detected maneuvers n_{seg} increases, meaning that not only excitation maneuvers are detected (left grey area). Even though they were not planned to be detected, some of these maneuvers could still be useful for estimation. For d_{crit} between 7.9 °/s and 21.4 °/s, the parser misses some parts of the maneuvers but they are still found. With a choice of $\dot{u}_{crit} > 21.4$ °/s whole maneuvers will be lost during the detection, which results again in an area of incomplete detection (right grey area).

The large band of the detection of all 17 maneuvers (5.6...24.1 °/s) makes it easy to find a good starting value of $\dot{u}_{crit}$ for the parser tuning. Two possible ways to obtain a correct detection of maneuver data segments are:

1. Manual search for threshold values resulting in a correct detection,
2. Increasing t_{fore} and t_{over} to certainly include all maneuver parts.

Another problem of the rate criterion parser is the fact that the detection behavior and the resulting robustness concerning a threshold value change are directly related to the input rate and thus to the control surface characteristics. Primary aircraft control surfaces have different shapes and are today mostly moved or even augmented by actuators. Further, the aircraft depending motion characteristics influence the control input rate and consequently the parser behavior, which must be compensated by a specific tuning for every aircraft.

The wavelet parser must be tuned concerning the aircraft characteristics as well. The illustration of the parser sensitivity to a threshold value change in Fig. 17 shows a comparable result to Fig. 16. The band of d^8_{crit} (scaled value for $m = 3$) between 0.47 and 0.92 results in a correct detection of all 17 maneuvers (blue area, dashed line marks previously used value $d^8_{crit} = 0.663$). For smaller values, the wavelet parser falsely detects more segments (left grey area), but as mentioned above, they can also be beneficial for an estimation. Increasing d^8_{crit} to values above 2.144 results in the loss of whole segments and a incomplete detection (right grey area). The band of correct detection is large enough to quickly find a suitable value for d_{crit} and with the above given possible ways to overcome the problem of the incomplete detection, all maneuvers should be correctly found in the data.

An increase in the noise level could have deteriorating effects on the parser approaches' robustness. The possibility of "loosing" a maneuver by increasing the criterion to compensate the additional noise level increases as well. If the signal-to-noise ratio is much lower than for the measurements in the flight data used for the herein presented performance tests, this problem would have to be faced and possible solutions like various filtering methods would have to be investigated. But as shown in the time history plots for the rate criterion parser (Figs. 8, 9, 10) and the wavelet parser (Figs. 11, 12, 13), there is still a satisfiable margin between the peaks used for the detection and the "zero values". If these margins get too small other approaches for calculating the signal time derivative like non-recursive differential low pass filters would be necessary.

Nevertheless, tuning the parser for certain maneuvers can be difficult. During the tests, it was found problematic to find a suitable setting for the detection of the rudder doublets. As shown in Figs. 10 and 13, the doublets result in only small peaks of $\dot{u}$ resp. d_3, which are difficult to detect and discriminate from the remaining data. Therefore further investigations on approaches to support the detection of excitation maneuvers with small inputs and to consequently increase the robustness concerning different input magnitudes are necessary.

All in all, the first configuration for every aircraft requires a few tests with existing data records and a known number of maneuvers. Afterwards both approaches should work automatically (case 2). Experiences with the wavelet parser during a flight test campaign with the DLR aircraft ATRA (Airbus A 320) in spring 2013 showed, that if the configuration is set once, the system works quite well.

6 Summary

Two simple data parser approaches are presented in this paper, one based on the control signal time derivative and another based on a control signal wavelet transform. The first proposed approach uses the control input signal and the aircraft's response (angular rate) to find the desired data segments containing the flight test maneuvers. This approach is very fast and can be used as a nearly real time implementation in the online parameter estimation tool. All desired data segments in the example flight test data recording were correctly found with the chosen parameter configuration, which shows the practicability of this parser approach. The additional use of an output signal (angular rate) monitoring supports the algorithm and results in sufficiently large data segments, containing also the aircraft's dynamic response after the control input has ended.

The second proposed algorithm is based on a FOWT of the control input signal. The wavelet transform has low pass characteristics and, depending on the decomposition level, the cut-off frequency is low enough to filter possible measurement noise. This parser is slightly slower than the rate criterion approach, but still fast enough for an online implementation. The configuration effort for the wavelet parser is less than for the rate criterion approach, but the results are still very promising. The parser is implemented in the online parameter estimation tool chain as well, and tests with measured data of a former ATTAS flight test resulted in the detection of all desired maneuvers.

Comparison of both approaches showed a good robustness regarding threshold value changes for a simple test case using the available flight data. Some problems and points to be investigated for the future developments were discussed. Summing up, the following three points characterize the findings of the parser tests:

1. Both approaches work well and all test data segments were found.
2. There is a different computational effort and hence calculation time consumption, but still, both approaches are fast enough for online implementation.
3. With pre-configuration, an easy usage during flight tests is possible.

With these results, both parser approaches show basically the ability to be used in an automated system for online parameter estimation. Moreover, the approaches could not only support the online applications (case 2 in Sect. 1) which they were designed for, the offline data evaluation resp. flight data mining effort could be reduced as well (case 4).

7 Future work

The parser approaches can currently only handle one control input signal at a time. In the further development, the possibility to search in several control input signals will be implemented. Both parsers, but especially the wavelet parser, could benefit from a multi-signal approach, where several in- and output signals are combined to find useful data segments for parameter estimation purposes. Not only angular rates, which are connected directly to the control surface deflection, but also inflow parameters like angle of attack or true airspeed could enhance the system performance. Hence, the structure of algorithm and online tool has to be adapted. These changes would also necessary to address the offline data mining problem (case 4) so that all longitudinal and lateral excitation maneuvers could be found in parallel.

For further investigations, adding noise with different magnitudes to the measured signals will give an indication of the two approaches' robustness. The question about changing the differentiation method must be answered, if the used polynomial differentiating method shows massive disadvantages. Moreover, additional variations of configuration parameters could be helpful to detect possible weaknesses. Also the above mentioned point of the aircraft depending tuning problems have to be further investigated.

Acknowledgments The author would like to especially thank his colleague Dr. Nicolas Fezans for his very useful support during the development of the wavelet parser and the preparation of this paper.

References

1. Jategaonkar, R.V.: Flight Vehicle System Identification: A Time Domain Methodology. Progress in Astronautics and Aeronautics, vol. 216. American Institute of Aeronautics and Astronautics Inc., Virginia (2006)
2. Deiler, C.: An online parameter estimation tool. German Aerospace Congress (Deutscher Luft- und Raumfahrtkongress, DLRK), Berlin, September 10–12 2012. Deutsche Gesellschaft für Luft- und Raumfahrt (DGLR)
3. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
4. Mallat, S.: A Wavelet Tour of Signal Processing: The Sparse Way, 3 edn. Academic Press, Burlington (2008)
5. Naruoka, M.: Real-time wavelet flight data evaluation for system identification of flight characteristics. In: 28th International Congress of the Aeronautical Sciences, Brisbane, AUS, September 23–28 2012. International Council of Aeronautical Sciences (ICAS)
6. Pittner, S., Kamarthi, S.V.: Feature extraction from wavelet coefficients for pattern recognition tasks. IEEE Trans. Pattern Anal. Mach. Intell. **21**(1), 83–88 (1999)

7. Morelli, E.A.: Practical aspect of the equation error method for aircraft parameter estimation. In: AIAA Atmospheric Flight Mechanics Conference and Exhibit, Keystone, Colorado, August 21–24 2006. American Institute of Aeronautics and Astronautics, Inc. (AIAA)
8. Hamming, R.W.: Digital Filters. Prentice-Hall Signal Processing Series, 2 edn. Prentice-Hall Inc., Englewood Cliffs (1983)
9. Haar, A.: Zur Theorie der orthogonalen Funktionensysteme. Math. Ann. **69**(3), 331–371 (1910)