ettc 2020

Conference Proceedings
European Test and Telemetry Conference

First virtual Test and Telemetry Conference
23 – 25 June 2020

Proceedings

European Test and Telemetry Conference – ettc2020

The European Test and Telemetry Conference – ettc2020 is organized by the European Society of Telemetry.

This volume covers the proceedings of the European Test and Telemetry Conference – ettc2020, and contains the manuscripts of the lectures.

The authors are responsible for form and content of the papers.

Dear Telemetry Friends,

This year is really challenging for all of us: to preserve people's health against the COVID-19 pandemic, governments worldwide introduced complex restrictions at the beginning of 2020. Some months later, mid of 2020, our personal and professional lives are still highly impacted, although a kind of normalcy is coming back, day after day.

Despite those difficulties, and with the goal to keep on supporting the spirit of exchange of our testing community, the European Society of Telemetry decided to maintain ettc2020 and to make out if it the very first virtual telemetry conference worldwide, free for all visitors. Collaboration and innovation in testing is the motto of this year's conference: COVID-19 slowed our businesses down, but innovation keeps its pace, supported by new forms of cooperation and exchange!

On behalf of the European Society of Telemetry, it is therefore an even greater pleasure for me this year to present you the proceedings of the European Test and Telemetry Conference 2020.

ettc2020 is a world premiere: the first online exhibition and conference for and from our community. The exhibition, made of virtual booths to be visited from your (home) office, gathers more than 20 manufacturers. The conference presents a dense technical program of more than 30 high quality papers, merged in these Conference Proceedings. As always, you can find the latest and most promising methods here but also hardware and software ideas for the telemetry solutions of tomorrow.

Both education and training in the fields of telemetry are among the missions of the European Society of Telemetry. ettc2020 conference participants have the opportunity to attend a free Short Course on digital video, including fundamentals and advanced FTI applications.

The ultimate success of the conference remains entirely dependent upon your continued patronage. So thank you very much for supporting ettc2020 in its new form!

Nobody knows with certainty what the next telemetry conference will look like. However, we all hope to benefit from this years' experience to introduce innovative forms of cooperation in, again, physical events with in-person exchanges.

We are always seeking ways to improve the European Test and Telemetry Conferences. Please contact us with ideas, critiques, suggestions, and visit us on www.telemetry-europe.org.

We are looking forward to meeting you… in real life again very soon!

Renaud Urli
President of the European Society of Telemetry

COMPLETER.NET

WE DEVELOP BUSINESS ...

... BY BUILDING ROADS BETWEEN MANUFACTURES AND USERS.

FLIGHT TEST INSTRUMENTATION

parraid

+ Ground Station Recorder
+ Telemetry Receiver
+ Realtime-Processing Software
+ Post Mission Data Mining Software

CALCULEX®
Electronics, Systems, Software and Support

+ Airborne Flight Test Instrumentation
  Recorders Fixed & Rotary Wing
+ Mission Recorders
+ Multi-purpose Integrated Airborne
  Routers, Processors and Recorders

1. Sensors

Chair: Renaud Urli, Airbus Helicopters, Donauwörth (Germany)

2. Data Acquisition

Chair: Philipp Wager, Airbus Defence & Space, Manching (Germany)

6. Big Data
Chair: Pedro Rubio, Defence & Space, Madrid-Getafe (Spain)

7. Spectrum management & Coding
Chair: Robert Huber (Airbus Defence and Space)

DISTANCE MEASUREMENT FOR

INTRUSIVE OBJECTS ON RAILWAY TRACKS

Jian Song, Junyuan Yang

Guangzhou Zoweetech Ltd.

Guangzhou, Guangdong, PR China

songj@zoweetech.cn

ABSTRACT

It is required to measure the distance between the moving carriage and intrusive foreign objects or the carriage ahead during shunting operations. The traditional solutions for distance measurement include laser and radar distance measurement methods, while more and more studies about the binocular vision system come up. For laser distance measurement methods, one dimensional rangefinder is feasible for the objects in a straight line ahead; however, the scanning laser rangefinder cannot distinguish whether the objects ahead are on the railway track. Limited by the measurement accuracy resulting from camera installation on board, the binocular vision system[1] is not suitable for measuring the distance of intrusive objects on tracks as well. In this paper, a monocular vision system based on special designed image pre-processing algorithm is presented, which will accurately recognize and locate the railway tracks and objects on tracks. Moreover, this system could calculate the distance between the carriage with camera and the objects detected.

1. Introduction

Shunting operation is the center of a railway yard, and is the important component of the whole railway transportation. As shunter driver could not detect carriages or intrusive foreign objects on the tracks while the train is moving backwards, a detection mechanics or system should be applied to monitor the intrusive objects and carriage on the tracks which the train is running, and to measure the distance between for safety guarantee.

The binocular method[1] for foreign object detection uses a binocular camera to measure the track area, and performs stereo matching and three-dimensional reconstruction on the images collected by the left and right cameras to detect invading foreign objects. However, due to the huge amount of calculation in the reconstruction, it is very difficult to apply binocular method. Regarding the foreign object

distinguishing algorithm, an attempt was made[2] to classify the target by extracting target features and combining with SVM classifier. Another study[3] tried to discover intrusive behaviors based on the motion tracking of moving targets. There are other achievements related as well, but all these achievements using fixed cameras are not the best choice for railway shunting.

An solution combining both railway track recognition and invading foreign objects detection is present in this paper. An onboard device with built-in camera is used to recognize the running tracks, shunting carriage and foreign objects, then to measure the distance between.

2. System Framework

The system is composed by four modules, including image acquisition, data transmission, image processing, result display and alert, among which image processing is the focus of this paper. The flow chart of the framework can be shown as figure 1:

Capture image from
camera on train

Data transmission
to cloud server

Image processing

Output result and alert
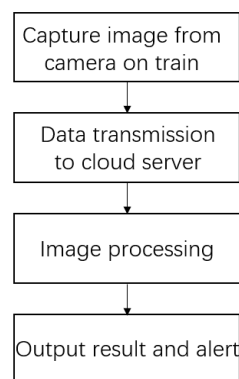
Figure 1: Flow chart of the whole system

There are two parts in image processing module, ranging and recognition. The ranging algorithm based on track detection consists of the following four steps:

1) Optimizing the edge detection algorithm to perform Canny edge detection[4].

2) Sifting to distinguish the edges of the railway track and that of the crosstie/ballast.

3) Distinguishing current running tracks.

4) Distance measurement based on perspective principle.

While the local texture features are extracted by LBP algorithm, and classified by SVM, the foreign objects are identified by their coordinates and the information of the bounding rectangular frame. Then the KCF + DSST algorithm are applied for precisely real time and multi-scale target tracking. The flow chart of image processing can be shown as figure 2:
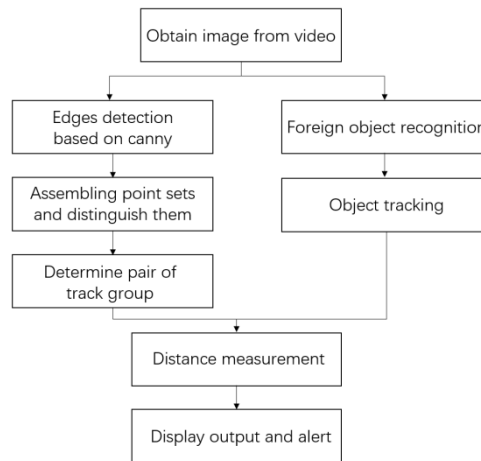
Figure 2: Flow chart of image processing

3. Railway Track Detection Based on Edges Recognition

3.1 Edge Detection Based on Canny Method

Based on Canny edge detection, an optimized edge detection algorithm with Hough line recognition is introduced in this paper to prevent unnecessary interference during image process. The edge detection operator usually distinguishes the edge of the target in the image according to the gradient mutation of the image pixel, which is mainly divided into two detection methods:

1) Based on the first derivative, the target edge detection is performed by calculating the gradient value at the corresponding pixel, such as: Prewitt operator, Roberts operator and Sobel operator.

2) Based on the second derivative, the image edge detection is performed by seeking the zero-crossing points in the second derivative, such as: Canny operator, Laplacian operator, and LOG operator.

The application of Canny edge detection algorithm[4] is divided into 5 steps:

1) Process the original image using Gaussian filter to reduce the invalid edges caused by noise.

2) Convolve the gray image to calculate the direction derivative and gradient.

3) Non-maximum suppression of the amplitude angle of the result.

4) Double threshold on the results after non-maximum suppression, and determine the pixels on the edge by the higher threshold, the lower threshold, and the connectivity of pixel.

5) Connect the edges points of the image after double threshold processing.

3.2 Meanshift Filtering

Light, shadow, crosstie etc. will cause many useless edges and noise in the image, so an appropriate filtering algorithm should be chosen for preprocessing the image. Comparing the Gaussian filtering algorithm with the Meanshift filtering algorithm, Gaussian filtering will blur the edge of the railway track and cripple the far edge detection, while Meanshift filtering can blur the texture information inside the railway tracks, but with little impact on the important track edge information.

MeanShift algorithm is essentially a general clustering algorithm. For a given number of samples, one of the samples is selected, and the sample is used as the center point to define a circular area and determine the center of mass. Then, continue to perform above-mentioned iterative process with this mass center point as the center until it finally converges.

In this paper, this feature of the mean shift algorithm is used to achieve smooth filtering of the image at the color level. This method could neutralize pixel color with similar neighbor pixel color, thereby erasing the texture details inside the railway track.

(a) Original image

(b) Gray image

(c) after Gaussian Filtering

(d) after Meanshift Filtering
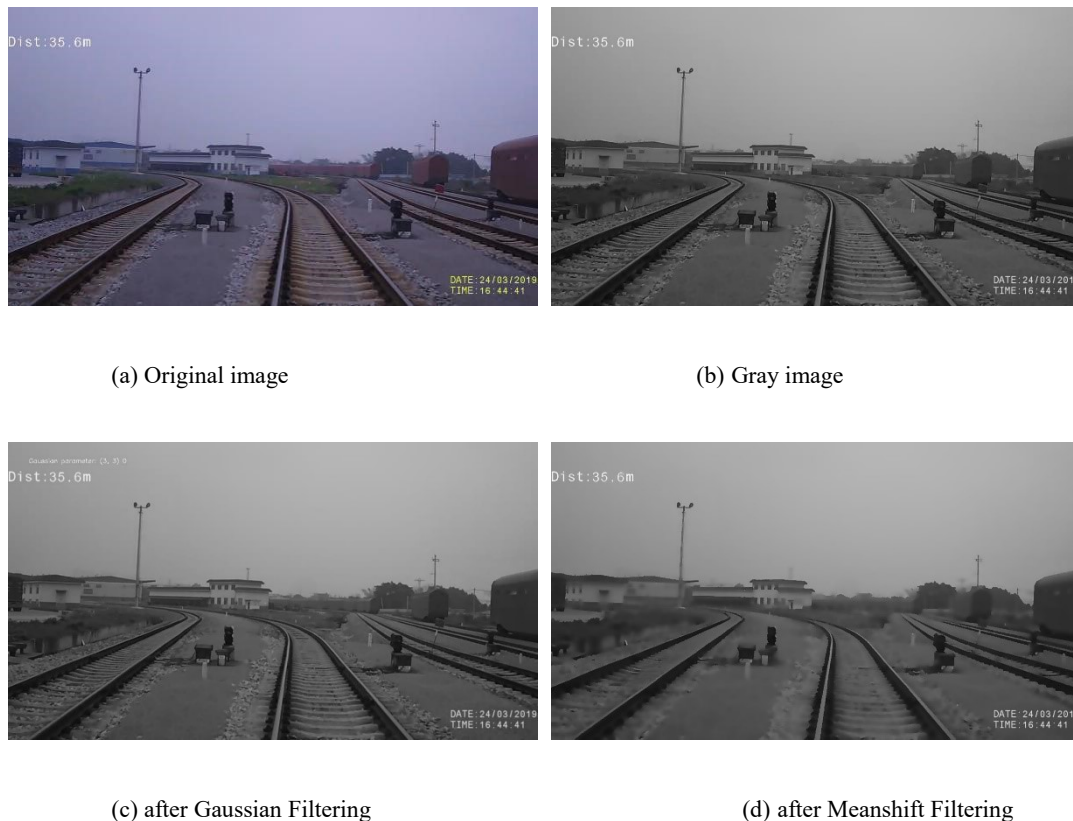
Figure 3. Comparing original image and that after filtering

Figure 3 shows that the Meanshift filtering algorithm can eliminate the texture interference inside the object while ensuring the railway track edges not blurred.

3.3 Edge detection combined with Meanshift and Canny

With excellent characteristics of single-pixel edges and robustness, the edge algorithm based on Canny operator is effective on the edge detection for most images, so Canny operator is widely used in edge detection. However, due to the complex texture information such as light and shadow interference and crosstie/ballast edge noise in the image, there are a lot of useless edges when directly detecting the railway track edge using Canny. For an improvement, an optimized edge detection algorithm that combines Canny edge detection algorithm combined with Meanshift filtering is introduced to retain the railway track edge information to the max and to eliminate the useless edge information of background and other interfering objects. Figure 4 below compared Canny edge detection algorithm and filtering.
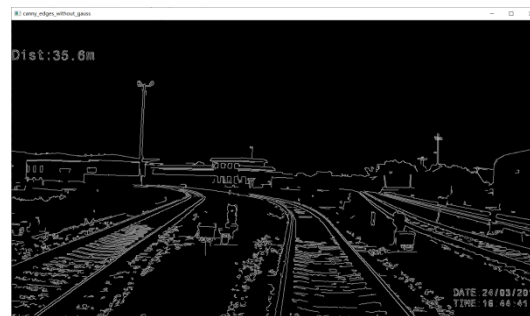
(a) Canny without filtering
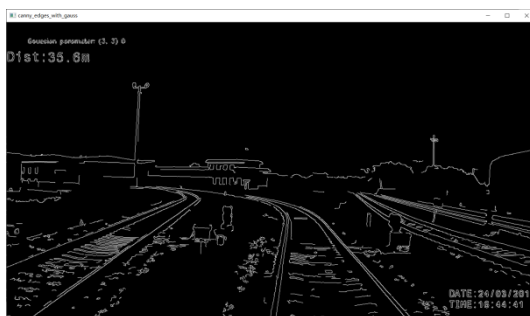
(b) Canny with Gaussian filtering   (c) Canny with Meanshift filtering
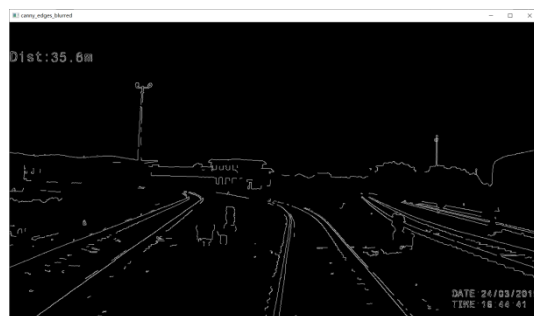
Figure 4: Results of different Canny filtering method

3.4 Point Assembling Algorithm

Since the result from Canny edge detection is a binary image containing edge information, the scattered points cannot reflect the continuity of the track edges, and cannot determine whether a edge point belong to the track or belong to the interfering. An algorithm for point clustering and improves the search method based on the traditional 8-domain search method is proposed in following.

Number 8 field directions in order from 0 to 7 as figure 5:
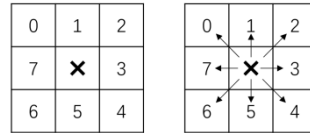
Figure 5: Non-zero center point and next non-zero point

As shown in figure 6, the connection status can be classified into two types: stable connection and virtual connection. The stable connection is in the positive direction of the valued point, that is, a non-zero point located on the 1,3,5,7 direction. The virtual connection is the valued point in the oblique side direction, that is, on the 0,2,4,6 direction lies a non-zero point.
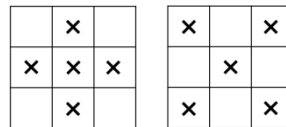
Figure 6: Stable connection and virtual connection

The improved 8-domain search method can reduce the number of times to find the next value point at a certain extent, thereby improving search efficiency, and speeding up the performance of the algorithm. The operation process is shown as follows:

1) Search non-zero points in the image from left to right and from bottom to top. The first point found will be used as the initial point, and the first current point. A new point set class is set up to store the attributes such as the position coordinates of the continuous non-zero points set growing from this initial point.

2) Look for the non-zero points in the stable connection direction of the current point. Store the new non-zero point to the list of non-zero points contained by the point set class and store the current point if there is one, then set the current point value as 0.

3) If there is none, then search the non-zero point in the virtual connection direction. Store the point in the list of non-zero points and set the current point value to 0 if there is.

4) If no non-zero points can be found in the stable connection direction and the virtual connection direction, the current value point is determined as the endpoint.

5) Select the next point from the list of non-zero points as the next current non-zero point.

6) Repeat above steps until all valued point sets in the image are generated.

In a experiment, a total of 17-point sets were found without the optimization process. The running time of the point assembling algorithm is 0.25531 seconds, the total time for processing the single image is 0.41285 seconds.

After optimization by algorithm above, a total of 18-point sets were found. And the time consumed for the point assembling algorithm was 0.16253 seconds, the total time for processing the single image is 0.33530 seconds. The speed of point assembling algorithm is increased to 36.34% while the speed of processing a single image is increased to 18.78%.

The output image of the improved point assembling algorithm can be shown as figure 7. The white line is the searched non-zero point set, and the red points indicate the endpoints.

Figure 7: Left is the Canny edge detection result, right is one of the recognized point set

3.5 Delineating Region of Interest with Probabilistic Hough Transform

In the field of computer vision, it is common to complete straight line detection by Hough transform algorithm[5], which maps the plane of the rectangular coordinate system to the curves of the polar coordinate system and maps the straight line to cross over point of the polar coordinate system. In this paper, the improved Probabilistic Hough Transform method, that is Progressive Probabilistic Hough Transform, prompt by Xu[6] is used, which alleviates the shortcomings of excessive storage occupation in standard Hough Transform.

In order to solve the problem caused by the complicated background, a ROI selection method combined with probabilistic Hough Transform is adopted in this paper to constrain the initial point of the point set. Thereby speeding the point assembling.

Generally, the process of probabilistic Hough Transform can be illustrated as bellow:

1) Input the image after edge detection by optimized Canny Algorithm.

2) Randomly sample the edge points in above image.

3) Transform the selected points by Hough and vote in the accumulator space formed by the distance ρ and the angle θ $(0 \sim \pi)$

4) When the threshold is reached, it is considered that a straight line exists, and move along the straight line direction to find the two endpoints of the straight line.

Select appropriate Hough Transform parameters to obtain a series of straight lines, which are classified as "right-inclined straight line" and "left-inclined straight line" by their slope. These straight

lines in different situations require different methods of constructing ROI.

For a right-inclined straight line, the coordinates of the upper left endpoint [x1-5, y1-5], [x1 + 5, y1-5] form the two endpoints of the parallelogram; the coordinates of the lower right endpoint [x2-5, y2 + 5] , [X2 + 5, y2 + 5] forms the other two endpoints of the parallelogram.

For a left-inclined straight line, the coordinates of the lower left endpoint [x2-5, y2-5], [x2 + 5, y2-5] respectively form the two endpoints of the parallelogram. The coordinates of the lower right endpoint [x1-5, y1 + 5], [x1 + 5, y1 + 5] form the other two endpoints of the parallelogram.

ROI constructing method can be shown as figure 8:
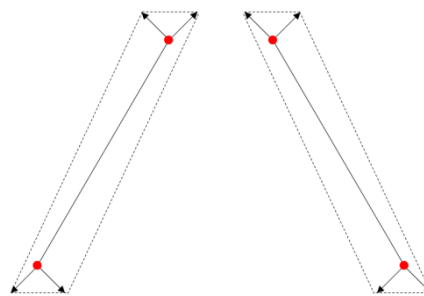
Figure 8: ROI constructed by right-inclined straight line and left-inclined straight line

The parallelograms after expanding all the straight line segments in the probabilistic Hough straight line detection result are represented in an image, named ROI_Mask, as shown in the figure 9:

Figure 9: Region of interest for point assembling algorithm

The initial point of the point assembling algorithm should have the non-zero pixel at the position in the Canny edge detection map, and the non-zero pixel at the corresponding position in the ROI_Mask image. This approach eliminates the set of interference points generated by the edges of the crosstie/ballast, background, etc.

Without ROI delineation in the experiment, the point assembling algorithm needs to search 982 times, and 31 point sets are found. The point assembling had taken 0.24538 seconds, while a single image process costed 0.42791 seconds, and the result of point clustering is shown in the figure 10:
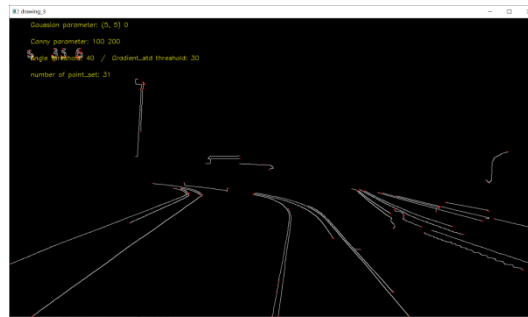
Figure 10: Point assembling result without improvement

If ROI delineation algorithm is applied to limit starting point selection, there are 84 searches in the point assembling, with a total 18 point sets found. The point assembling takes 0.15757 seconds, and the single image process takes 0.33620 seconds. The speed of point clustering is increased by 35.74%, and the speed of single image process is increased by 21.43%.
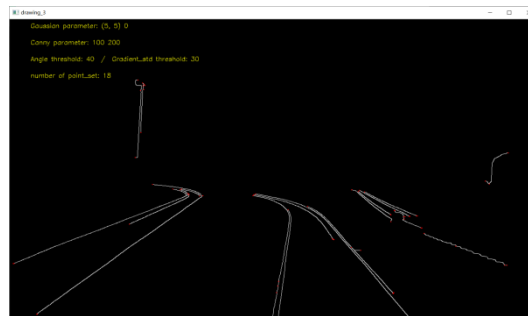
Figure 11: Point assembling result with ROI optimizing

4. Track Pair Determination Algorithm

The assembled point set is obtained after above steps, and points in the set are continuous. However, we still cannot distinguish point sets of the Railway Track from the ballast or other interference term.

In fact, the edge point set attribute can be used to preliminarily separate the track edge from most of the background interference items such as crosstie/ballast edge. In addition, the edge point sets belonging to the same track are merged and smoothed, each track has its unique corresponding edge point set. A score table could be generated by judging whether each two edges constitute a track group, to indicate which two tracks constitute a track group.

4.1 Determine Railway Track Point Set

After a series of point sets obtained from point assembling, we need to distinguish railway track point set from ballast point set by checking the attributes of each point set. Four attributes listed below, derived from the continuity of the edge point set, gradient similarity and other characteristics, are set as the criteria for distinguishing the railway track point set from the interference items.

1) The number of non-zero points in the point set.

2) The height difference between the highest point and the lowest point of the point set.

3) The angle value of line formed by the starting point and the ending point.

4) The standard deviation of the gradient of all points in the point set

Various threshold combinations were tried in the experiment, the threshold of the number of points in the point set is 100, the angle threshold is 40°, and the gradient standard deviation threshold is 20. The flow chart of sifting processing is as figure 12:
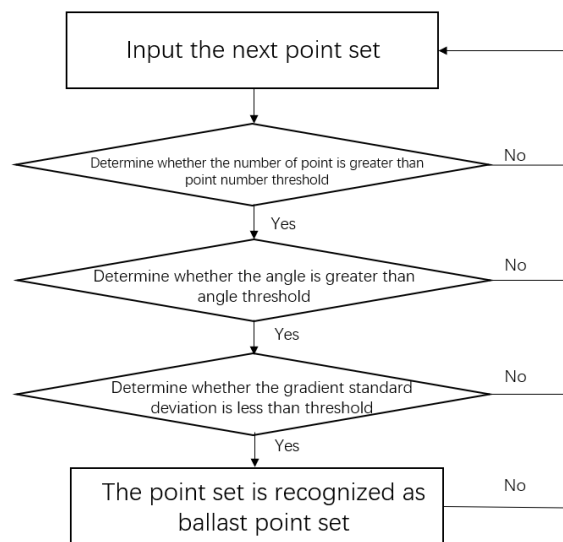
Figure 12: Flow chart for sifting processing

In the case where the edge point set is not sifting, in addition to the needed railway track point set, there are point set of interference objects such as background, as shown in figure 13:

Figure 13: Point set without sifting

After using the sifting algorithm, in figure 14, the railway track point set is retained, and most of the crosstie/ballast edges and background edges are removed:

Figure 14: Point set with railway track threshold sifting

4.2 Determine Crosstie Edge Point Sets

As a pair of tracks lies on a set of , it is possible to identify a pair of tracks by the position of the crossties. Having set some thresholds, 20°for angle, 20 for height, 5 for aspect ratio, 20 for gradient standard deviation, the crosstie edge determining process is shown as figure 15:
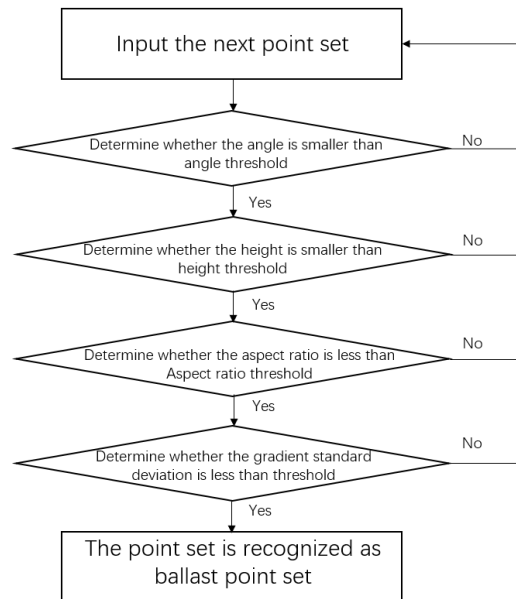
Figure 15: Flow chart for determining crosstie edge point set

The recognition result is shown as figure 16:

Figure 16: crosstie edge point set recognizing result

Corresponding to the original picture, in figure 17, it can be found that there are some background interferences identified as crosstie, but the basis for distinguishing the track pair is the number of crossties in the middle of the two tracks. So a few crossties misrecognition will not affect the result of the track pair recognition.
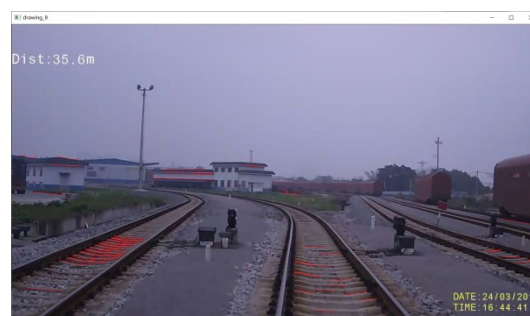
Figure 17: crosstie edge point set located in original image

4.3 Merging and fitting of Point Sets of the Same Railway Track

Observing the result of railway track sifting recognition, it can be seen that the same rail has two left and right edge lines due to the influence of light, and when it is disturbed, one track may be represented as more number of point set. Therefore, an algorithm needs to be designed to merge all edge point sets of the same track and fit a smooth curve to represent the track position.

There are the following steps to merge the point sets of the same track:

1) Sort railway track point set according to the intercept at the bottom of the image.

2) Determine that the set of points with a difference in intercept of less than 20 pixels is the edge of the same track, and merge them by merging the points at the same horizontal height (same y). The x coordinate of merged point is formed according to weighted average using the gradient size.

3) Perform multiple function fitting and smoothing on the points after weighted average.

The result can be shown as Figure 18.
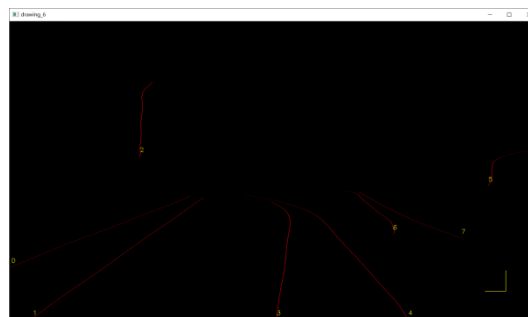
Figure 18: Point set after merged by railway track

Corresponding to the position in the original image (as shown in the Figure 19), it is found that the set of fitting points is roughly located on the track centerline, which can be used as the basis for the next step, distance measurement based on the railway track spacing and perspective principle.
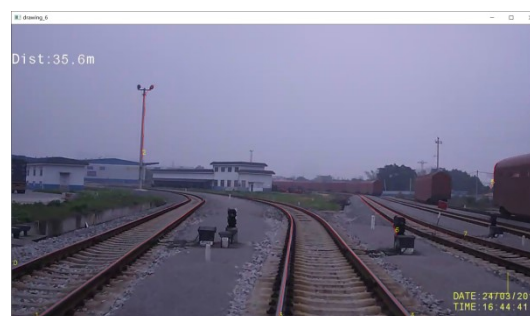
Figure 19: Merging and fitting result shown on original image

4.4 Track Pair Distinguish with Crosstie Edge Information

After obtaining the track coordinates, it is necessary to further determine which two tracks belong to one track pair. It can be found by observing the Canny result that there are crosstie edge in the middle of track pairs. So this paper proposes a track pair determination method taking crosstie edge information into consideration.

First calculate the position of the gravity of each crosstie point set; then randomly select two **point sets i** and **point set j** with similar height as the point set identified as the railway track. Finally, determine the height range for the two point sets, that is, the minimum value of the highest height of the two point sets is selected as the upper limit of the common height range, and the maximum value of the minimum height of the two point sets is selected as the lower limit of the common height, as shown in the figure 20.
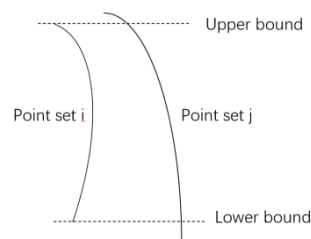
Figure 20: Sketch map for two railway point set and their common height range

Take two points at the same height within the common height range, and calculate their midpoint. Calculate the abscissa difference **x_delta** from the midpoint to the gravity of the crosstie point set with the same ordinate value. That difference, **x_delta**, can be used as the scoring criteria for judging whether **point sets i** and **point set j** belongs to the same track group.

Table 1   Score contribution by different x_delta

Distance	Score
x_delta < 10	50
10 ≤ x_delta < 20	40
20 ≤ x_delta < 30	30
30 ≤ x_delta < 40	20
40 ≤ x_delta < 70	10
70 ≤ x_delta < 100	5

The more the crosstie gravity between **point set i** and **point set j**, the more the crosstie center of gravity will contribute to "score of the same track group" for **point set i** and **point set j**. The track group is determined by the final formed score table. If the scores of the track edge **point set i** and **point set j** are the highest in the row of **point set i**, as well as in column **point set j**, it is determined that the track edge **point set i** and **point set j** are two rails of the same set of tracks .

As shown in Figure 21, there are a large number of crossties between point set 0 and point set 1, it

can be determined that these two point sets form the same track group, and the (0,1) position and (1,0) position in the constructed score table has a score of 735; there are a large number of crossties between point set 3 and point set 4, it can be determined that these two point sets form the same track group, and score in the (3, 4) position and (4, 3) in the constructed score table is 260.

Table 2   Score table constructed for judging track group

Point set i / Point set j	0	1	2	3	4	5	6	7
0	0	735	0	0	0	0	0	0
1	735	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	260	0	0	0
4	0	0	0	260	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
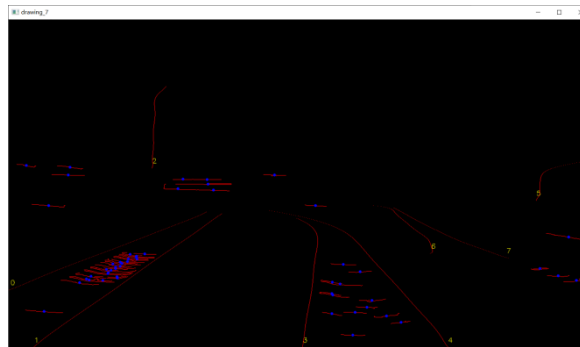
Figure 21: Red lines are the recognized railway track edges and crosstie edges, blue points indicate the position of gravity of crosstie; according to their positional relationship, the track group can be determined

5 Foreign Object Identification and Tracking

5.1 Feature Extraction Based on LBP

LBP, Local Binary Pattern, is an operator for describing local texture features in images and is widely used in face recognition, expression recognition and other fields[7]. In this paper, LBP is used for extracting the feature of railway track, crosstie/ballast and unknown objects (background).

The circular LBP operator can freely adjust the sampling radius and specify the number of sampling points to be used. For example, $LBP_{8,1}$ refers to sampling 8 pixels on a circle with a sampling radius of 1. If the gray value of the sampled pixel is higher than the center gray value, it is marked as 1, otherwise marked as 0, and finally a set of binary sequences would be outputted. Convert this binary sequence to decimal to get the LBP value of the center pixel.

Because the LBP feature calculated by the circular LBP operator used in this paper has rotation

invariance and gray invariance, it can be well adapted to the misjudgment of foreign objects that may be caused by light and shadow interference. After processing the image using the LBP operator, an LBP "code" can be obtained at each pixel, and the extracted LBP feature is still an image. The image is divided into 40 * 40 small blocks, which is convenient for subsequent SVM classification processing. The LBP feature map is shown in the figure 22.
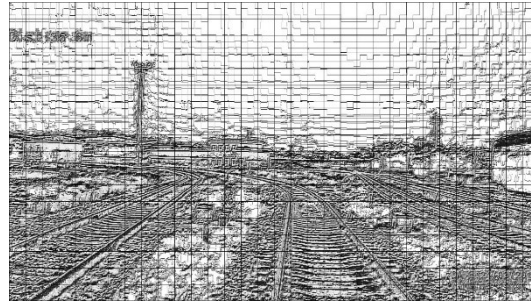
(a) LBP feature for the whole image
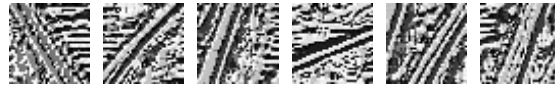
(b) LBP feature for railway track block

(c) LBP feature for crosstie block

(d) LBP feature for unknown objects (background)

Figure 22: LBP feature map for the whole image, railway tracks, crosstie and unknown objects

5.2 Foreign Object Recognition by SVM Classifier

SVM[8] is a classifier that can achieve better based on fewer sample amount. Its basic model is defined as a linear classifier with the largest interval on the feature space. Its learning strategy is to maximize the interval, which can be converted into a convex quadratic programming problem[9]. The SVM algorithm was originally designed as a binary classifier. In this paper, the one-versus-rest SVM is applied, and a multi-classifier is constructed by combining multiple classifiers.

In this paper, the extracted LBP feature parameters are used as a classification basis, by which all the small blocks are manually divided into three categories: rails, crosstie/ballast, and unknowns (background). During AI training, one kind of sample is classified into one class as a positive set, and the remaining as a negative set. Three kinds of samples produce three SVM classifiers. During the test,

the test vectors are tested with three trained SVM classifiers respectively, and three scores are obtained, and the largest one of the three scores is taken as the classification result.

The image is divided into multiple blocks of 40×40 pixels, and each block is classified into a training set and a test set of three kinds of samples. There are 1458 blocks in the track test set and 516 blocks in the test set; 1549 blocks in the crosstie/ballast training set and 605 blocks in the test set; 2998 blocks in the background training set and 2140 blocks in the test set.

The model file **clf.model** obtained from the training set was used to test the recognition accuracy. The accuracy of the track recognition was 0.87, the accuracy of the crosstie/ballast recognition was 0.89, and the accuracy of the background recognition was 0.97.

The result of recognizing the whole image is shown in the Figure 23 below, where label 1 indicates recognition as a track, label 2 indicates recognition as a crosstie/ballast, and label 3 indicates recognition as an unknown object (background).
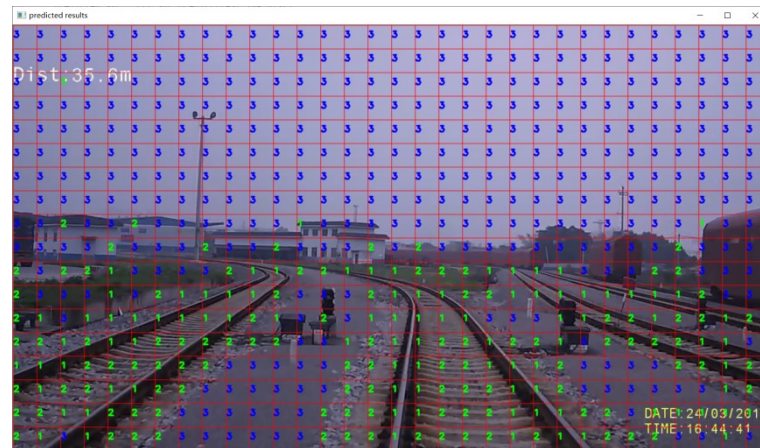
Figure 23: Recognizing the whole image

However, classifying all the small blocks of the full image will cause excessive time consumption. Usually, the detection of a whole image will take about 3 to 4 seconds, which cuts down real-time performance of the algorithm. This paper proposes a warning area division method based on railway track pairing.

First, extract a pair of tracks, as in Figure24 (a), then connect two highest points of the two tracks of the track group, and two lowest points to form a closed loop, as shown in Figure 24 (b).

Second, perform a dilation algorithm for the closed loop in order to expand the single pixel identified by the Canny edge detection to multi-pixel width, as shown in Figure 24 (c). Then look for the convex hull of the expanded graphics, and observe that the convex hull contains the track and ballast area, which will be used as the early warning area, the corresponding area in original image is shown in Figure 24 (d) .

The SVM only processes small blocks in the early warning area. If an unknown object (background) is detected, an alarm would be issued and the coordinate position of the unknown object (background) area is sent to the tracking module for tracking foreign objects.

(a) Railway track pair recognized

(b) Connect to form a closed loop

(c) Processing dilation

(d) Early warning area in original image
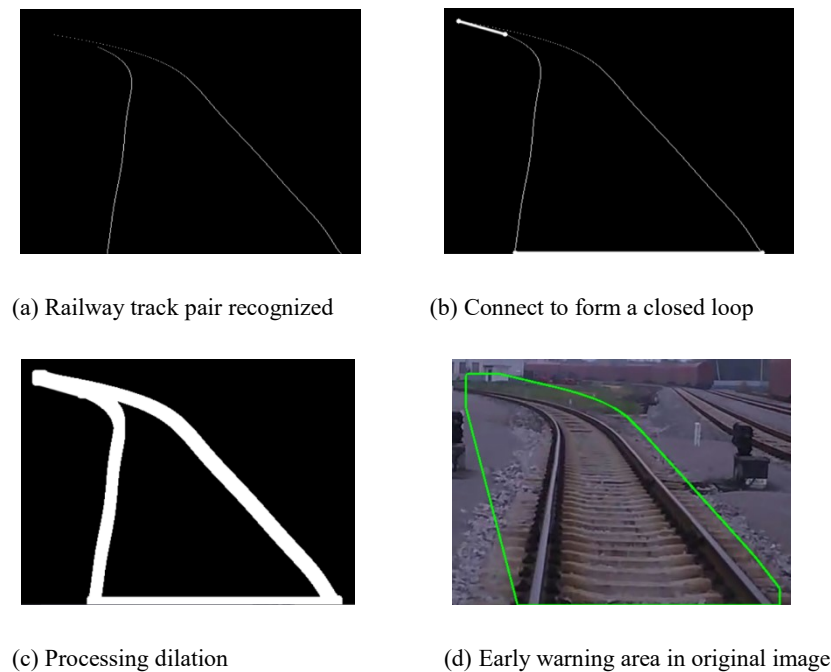
Figure 24: ROI defining process

5.3 Foreign Object Tracking Algorithm Based on KCF-DSST

Since direct foreign object detection algorithm for each frame of image will cause heavy overhead, a foreign object tracking algorithm based on correlation filtering is added to the foreign object detection algorithm. The foreign object intrusion algorithm is processed every 1 second / 15 frames, and the rest frame is using the tracking algorithm that keeps track of foreign objects. This method can ensure the accuracy of the algorithm and the timely detection of foreign body intrusion, and ensure the real-time performance of the algorithm as well.

The tracking module uses the KCF, kernelized correlation filters[10], to track the results of foreign object detection, which better target multi-scale transformation using discriminative scale space tracking method proposed by Martin Danelljan and Fahad Shahbaz Khan in 2016[11].

The algorithm has designed two consistent correlation filters to achieve target tracking and scale transformation, defined as translation filter and scale filter respectively. Translation filter is responsible for positioning the current frame target, and scale filter is used to estimate the current frame target scale. Two filters are relatively independent, so we can select different feature types and feature calculation methods to train and test.

In practical applications, the location and scale information of rectangular frame tracking are selected based on foreign object recognition, and the target position is found through the translation

filter calculation response in next frame. After the scale filter calculation response, the target's next frame scale is found. The result of the filter is used as the updated target position and target scale, and then the updated tracking rectangle is used for training and optimizing the translation filter and scale filter, and the above steps are looped to achieve target tracking. The tracking result can be shown as Figure 25.

The processing speed of the tracking algorithm is about 60fps

Figure 25: Tracking Using KCF-DSST

6 Ranging Algorithm Based on Track Group Detection

The distance measurement based on the perspective principle mainly depends on the relationship between the target distribution in the actual scene and the corresponding target position in the image. The width of the rail in the actual scene is certain, and the width of the rail in the image decreases with the distance increasing due to the perspective principle. The relationship between those two can be used to calculate the actual distance from the camera to any position on the rail.

As shown in Figure 26, in the schematic vertical view of the actual scene, the point O is the camera position on the train, and the two red straight lines represent the railway track which the train is running at; E, F are the two points on the railway track that initially entered the camera screen and actual distance from EF to camera is l_0; the two points A and B are the positions of the left and right vertex that can be seen at the bottom of the image in the actual scene; the blue object in Figure 26 represents the foreign objects appearing on the rail, and the two points G and H are the two points of the rail where the foreign object appears, the distance from the invading foreign object to the camera is l_1; the two points C and D represent the corresponding positions in the actual scene of the most left and right points that can be observed on the horizontal line of G' and H' in the Figure 27. The distance from the connection lines of C and D to the camera is also l_1; the width of the rail adopts the international standard gauge, that is 1435mm.
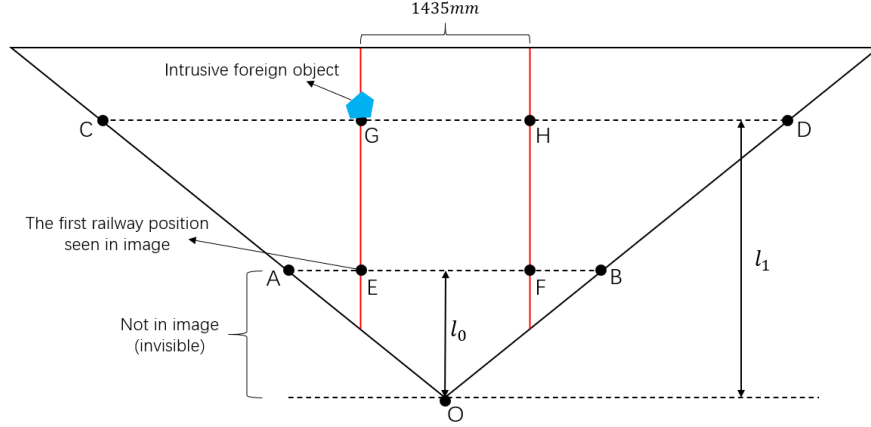
Figure 26: A schematic vertical view of the actual scene, showing the actual positional relationship between the camera, the railway track and intrusive foreign objects

In the image collected by the camera, due to the perspective principle, the width of the railway track (the red curve) continues to shrink in the picture. As shown in Figure 27, suppose the width of the image is W and the image size is 1280×720, so $W = 1280$. The E', F' points in the image represent the intersection of the railway track and the bottom of the image. The A', B' points are the left vertex and the right vertex at the bottom of the image; the intrusive foreign object is found at the G' point in the image, and the H' point is the same height of G' and on another railway track; the position of the two points C', D' in the image is the intersection of the horizontal line G'H' and the left and right borders of the image.

The abscissa difference between E' and F' is set as w_0, and the abscissa difference between G' and H' is set as w_1.

Observing the schematic vertical view of the actual scene, since the triangle OAB and the triangle OCD form a pair of similar triangle, the ratio of the CD side to the AB side is equal to the ratio of the distance l_1 from CD to point O and the distance l_0 from AB to point O, given by Equation 1:

$$\frac{l_{CD}}{l_{AB}} = \frac{l_1}{l_0} \tag{1}$$

The actual length of l_0 is roughly 7.9m, and the distance between the intrusive object and the camera can be obtained by calculating the ratio of l_{CD} with l_{AB}.

The CD line segment in the actual scene is shortened to the C'D' line segment in the image, and the GH line segment is shortened to the G'H' line segment. Considering that the actual scene is shot by the camera, the scaling ratio of objects from the same distance is the same, the equation can be deduced as:

$$\frac{l_{CD}}{W} = \frac{1435mm}{w_1} \tag{2}$$

Similarly, the EF line segment is shortened to the E'F' line segment, and the AB line segment is shortened to the A'B' line segment, and the equation can be written as:

$$\frac{l_{AB}}{W} = \frac{1435mm}{w_0} \tag{3}$$

Combining above three equations, the expression of the distance l_1 of the intruder from the camera can be derived as

$$l_1 = \frac{w_0}{w_1} \times l_0 = \frac{X_{F'} - X_{E'}}{X_{H'} - X_{G'}} \times l_0 \tag{4}$$
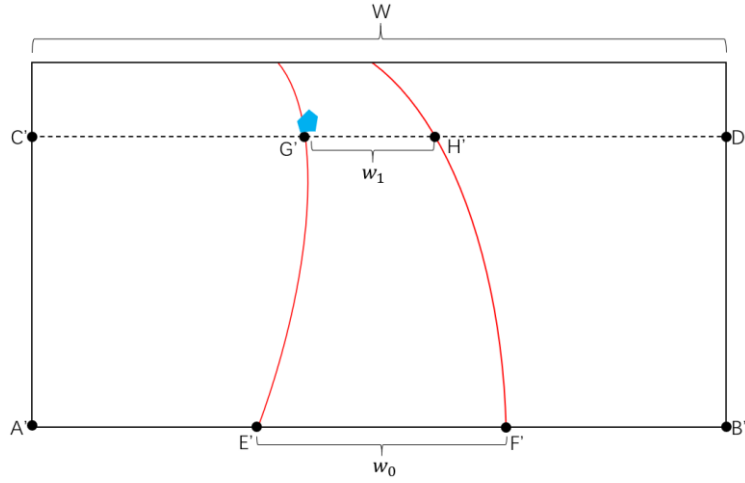
Figure 27: Schematic diagram of camera shooting image

7 Conclusion

In this paper, we designed a system to detect and track the intrusive foreign objects on railway tracks, and a distance measuring method based on railway track group recognition. In order to promote the effectiveness and real-time ability, we optimized the Canny edge detection algorithm, point set assembling algorithm and SVM classification algorithm. Furthermore, we proposed the track group classification algorithm and up to now no such algorithm is in use.

There are some problems to be solved in future:

1) Due to the complexity of the actual track scene, there will often be interference and occlusion, which leads to some reduce of the accuracy rate.

2) The limitation of the detection algorithm makes it hard to distinguish the complex rail sections such as turnouts.

3) Due to the limitation of network bandwidth, it is difficult to transmit high-definition video images for processing, which has a negative impact on long-distance foreign object detection and rail detection

The next research direction is to improve the accuracy of foreign object recognition and optimize the algorithm, improve the overall efficiency of the algorithm, and speed the algorithm to better meet the actual real-time requirements.

Reference

[1] Ohta M. Level crossings obstacle detection system using stereo cameras. Quarterly Report of RTRI (Railway Technical Research Institute), 46(2): 110-117, 2005.

[2] L. Tong, L.-Q. Zhu, Z.-J. Yu, et al. Railway obstacle detection using onboard forward-viewing camera. J Transp Syst Eng Inf Technol, 12(04), pp.79-83, 2012.

[3] Dong, H., Ge, D., Qin, Y. and Jia, L., Research on railway invasion detection technology based on intelligent video analysis. Zhongguo Tiedao Kexue, 31(2), pp.121-125, 2010.

[4] Canny, J., A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, (6), pp.679-698, 1986.

[5] Shapiro, S.D., Feature space transforms for curve detection. Pattern Recognition, 10(3), pp.129-143, 1978.

[6] Xu, L., Oja, E. and Kultanen, P., A new curve detection method: randomized Hough transform (RHT). Pattern recognition letters, 11(5), pp.331-338, 1990.

[7] Ahonen, T., Hadid, A. and Pietikainen, M. Face description with local binary patterns: Application to face recognition. IEEE transactions on pattern analysis and machine intelligence, 28(12), pp.2037-2041, 2006.

[8] Cristianini, N. and Shawe-Taylor, J., An introduction to support vector machines and other kernel-based learning methods. Cambridge university press, 2000.

[9] Zhang, Hongshuai, Zhiyi Qu, Liping Yuan, and Gang Li. "A face recognition method based on LBP

feature for CNN." In 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 544-547. IEEE, 2017.

[10] Henriques, J.F., Caseiro, R., Martins, P. and Batista, J., High-speed tracking with kernelized correlation filters. IEEE transactions on pattern analysis and machine intelligence, 37(3), pp.583-596, 2014.

[11] Danelljan, M., Häger, G., Khan, F.S. and Felsberg, M., Discriminative scale space tracking. IEEE transactions on pattern analysis and machine intelligence, 39(8), pp.1561-1575, 2016.

Evaluation of Energy Harvesting Devices for Industrial Applications

Leander B. Hörmann[1], Thomas Hölzl[1], Christian Kastl[1], Peter Priller[2], Andreas Springer[3]
[1] Linz Center of Mechatronics GmbH, 4040 Linz, Altenberger Straße 69, Austria,
[2] AVL List GmbH, 8020 Graz, Hans-List-Platz 1, Austria,
[3] Institute for Communications Engineering and RF-Systems, Johannes Kepler University Linz,
4040 Linz, Altenberger Straße 69, Austria
Leander.Hoermann@lcm.at

Abstract:

The use of energy harvesting devices (EHDs), such as solar cells and thermoelectric generators, offers a promising opportunity for powering self-sufficient wireless devices. This becomes more important due to the miniaturization of electronic components and the resulting reduction in power consumption of integrated circuits. As a result, wiring and maintenance costs can be reduced or possibly omitted at all. However, the available output power of EHDs is highly dependent on the mounting location as well as on environmental conditions and may vary greatly over time. Therefore, it is meaningful to evaluate the EHDs at the location of use over a certain period of time in order to characterize them in real world scenarios. In this paper, we present a mobile and wireless measurement system for the characterization of EHDs. The measurement system enables the acquisition of different EHDs' characteristics for specific applications under various environmental conditions. This simplifies the selection of a suitable EHD and the design process of an energy management system for a particular application. An evaluation of the implemented measurement system has been carried out by characterizing a specific solar cell at different illumination levels for an industrial application.

Key words: Energy harvesting devices, characterization, wireless measurement system, industrial application, power supply evaluation.

Introduction

In the course of industrial internet-of-things (IIoT), more and more sensors are applied at machines and processes to supervise and optimize them further. However, their wiring is a cost intensive task and reduces flexibility. It could be omitted if self-sufficient wireless sensor nodes are used. The wireless sensor nodes connect with each other to form a so-called wireless sensor network (WSN). Each sensor node measures physical quantities of industrial plants or processes and transmits them wirelessly towards a gateway using a certain communication protocol [1], [2]. The power supply of the wireless sensor nodes can be implemented in two different ways[3]:

First, the energy can be applied during the installation of the wireless sensor node by any kind of battery. Thus, the available energy for operation is limited and the operational time depends mainly on the average power consumption of the wireless sensor node and the capacity of the battery (neglecting any leakage currents and dynamic discharge behavior of the battery). The advantage is, that it allows the estimation of the operational time without any knowledge of the location of installation. For application scenarios with a specified operational time or if the device needs periodic physical maintenance, this solution could be more cost effective than the following solution.

Second, the power needed for supply can be harvested from the environment using so called energy harvesting devices (EHDs). EHDs can convert power of the environment into electrical power, for example solar cells or thermoelectric generators. Therefore, the operational time of the devices can be increased significantly. However, the harvestable electrical power of the environment is highly variable [4] and [5]. Therefore, an energy storage is needed to provide a continuous supply of the wireless sensor node. The energy storage can be implemented using for example rechargeable batteries or supercapacitors. If the capacity is well dimensioned, a perpetual operation is achievable [6].

Wireless sensor nodes supplied by energy harvesting have the advantage of a possible maintenance-free operation. As already said, the available output power of EHDs is highly dependent on the mounting location as well as on environmental conditions and may vary greatly over time. Therefore, it is meaningful to evaluate the EHDs at the location of use over a certain period of time in order to characterize them in real world scenarios. Using this real-world characterization data, an evaluation of continuous power supply can be performed. This paper presents a mobile and wireless measurement system for the characterization of EHDs. The measurement system enables the acquisition of different EHDs' characteristics for specific applications under various environmental conditions. This simplifies the selection of a suitable EHD and the design process of an energy management system for a particular application. The rest of the paper is organized as follows: First, this paper summarizes the industrial application scenario, second, it introduces the measurement system, and third, it discusses the measurement accuracy and shows first measurement results. Finally, it concludes the paper and outlines directions for future work.

Application Scenario

Testing of prototypes is an essential part in development and optimization of automotive powertrain systems. Prototypes are assembled and instrumented as unit-under-test (UUT) in an appropriate laboratory environment, such as a powertrain test bed. This allows observing properties in a realistically simulated environment. Such measurements help to characterize the UUTs to support further development. Wireless sensor nodes offer several advantages: First, a faster and simpler instrumentation due to elimination of cabling. Second, avoidance of plug contact problems and confusion that may occur with conventional instrumentation. And third, improvement of signal quality, because the analog signal is digitized directly at the node and all measuring points are electrically isolated from each other and from the automation system.

Energy supply of the wireless sensor nodes should not cause interruptions to the usual workflow. Any maintenance needs of nodes should be avoided during a measurement campaign (typically a few hours to several weeks). The wireless sensor nodes are therefore typically powered by batteries or energy harvesting systems (EHSs) [3] and thus severely energy constrained. They have either a limited amount of energy (batteries) or are prone to a low available power (EHSs) [7]. To be able to eval-

uate the energy supply guarantee, it is needed to know average power consumption of the wireless sensor nodes and the available power over time. The following chapters presents the proposed measurement system for characterization of EHDs and first evaluation results.

Measurement System for the Characterization of Energy Harvesting Devices

The proposed measurement system is flexible and can be composed of several mobile measurement nodes for the simultaneous characterization of several EHDs at different locations as introduced in [8]. The autonomous measurement nodes consist of the developed measurement hardware and perform the characterization task. The characterization is based on a cyclic acquisition of the current-voltage characteristics of the connected EHDs. The measurement interval and the number of measurement points to be recorded can be configured. The measuring nodes form a mesh network with the base station and additional routing nodes and transmit the characterization data wirelessly to the base station. The thread radio protocol is used for communication. This is an IPv6-based mesh network protocol that is designed for secure, reliable and power-saving wireless communication [9]. OpenThread is used as open source software stack implementing the thread communication protocol.

The measurement hardware is powered by a rechargeable battery to enable mobile use and is designed for a characterization period of up to one year. Fig. 1 shows the simplified block diagram of the measurement hardware with the individual function blocks. The EHD to be characterized and external sensors for the acquisition of relevant environmental parameters are connected to the hardware. The supply circuit is responsible for the power supply of the other circuit components and provides the required voltage levels. The measurement hardware can either be supplied by an internal energy storage for mobile use or an external power supply. The internal energy storage is composed of a lithium-ion battery and a supercapacitor. In addition, it is possible to use the connected EHD to recharge the battery between the cyclic characterization measurements. This may extend the operational time of the measurement hardware significantly depending on available power.

The System-on-Chip (SoC) nRF52840 from Nordic Semiconductor is used as microcontroller and radio transceiver. This controller takes over the control of all hardware components, the power management, and provides the required IEEE802.15.4 radio interface for communication using OpenThread.
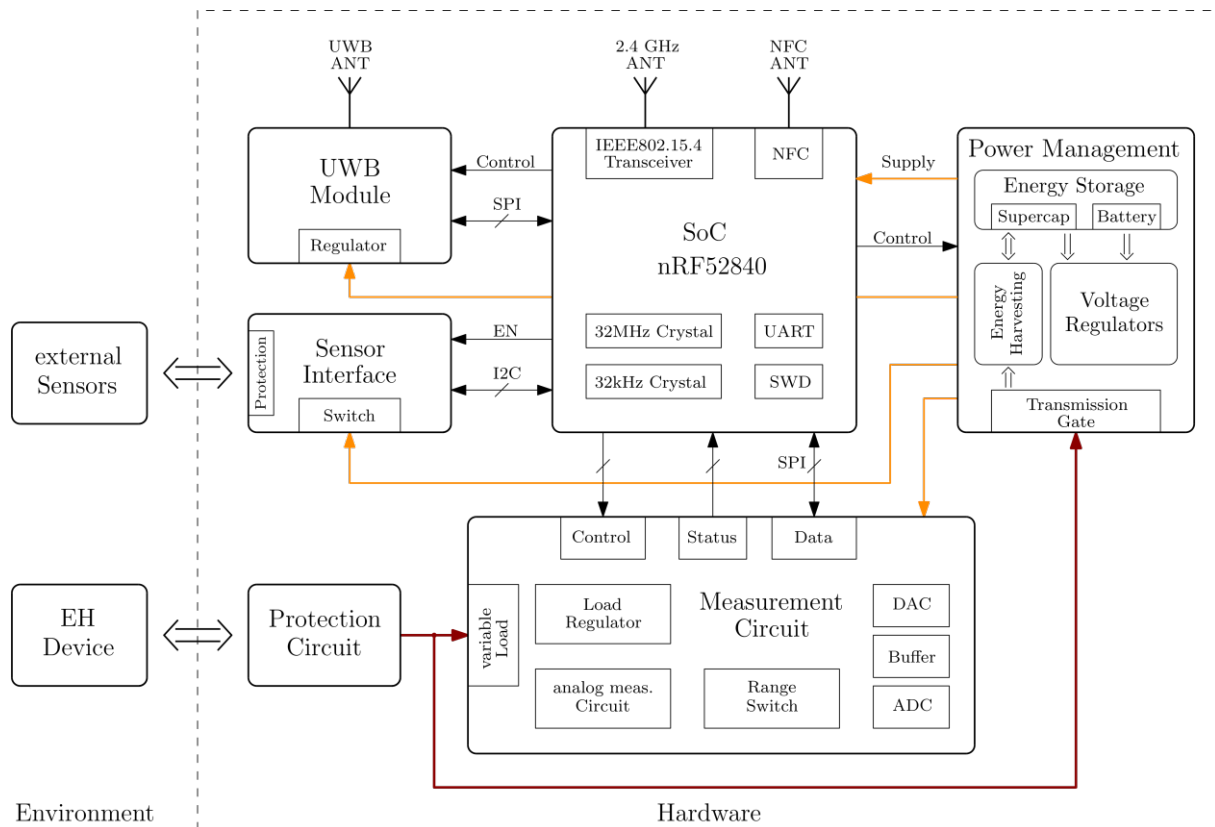
Fig.1: Block diagram of the proposed measurement hardware with connected external sensors and energy harvesting device.

The measurement circuit is responsible for recording the characteristics by measuring the current-voltage characteristic of the EHD. Using a variable load, different operating points including the short-circuit and open circuit case are set, and the resulting output currents and voltages of the source are measured. For a specific operating point, the output voltage of the EHD is controlled by the operating point controller to a desired preset value. The microcontroller uses a digital-to-analog converter (DAC) to provide the preset value. The measurements of the currents and voltages are carried out with an analog measurement circuit and an analog-to-digital converter (ADC). The measurable voltage ranges from below 1 mV to up to 24 V and the measurable current ranges from below 1 µA to up to 1 A. In order to cover the wide measurement ranges with sufficient accuracy, the voltage measurement range is divided into two sub-ranges and the current measurement range into three sub-ranges. The change between the individual current measurement ranges is performed by an analog comparator circuit to ensure fast and automatic switching between the measurement ranges. The voltage measurement range is set by the microcontroller.

External sensors can be connected to the sensor interface for the acquisition of environmental parameters required for characterization, such as temperature or illumination level. A 2.5 V supply voltage is provided for supplying the sensors and an I²C interface for communication. The sensor interface allows the simultaneous connection of several sensors, which enables the acquisition of different environmental parameters.

The ultra-wide-band (UWB) communication module is intended for a localization of the measurement hardware. This allows an automatic capturing of the position of the measurement node and a detection of any undesired or illegitimate movement. During normal operation, this hardware component is deactivated most of the time to keep the average power consumption as low as possible.

Measurement Accuracy

The evaluation of the measurement accuracy and calibration of the measurement hardware has been carried out using the reference instrument SourceMeter® 2602B from Keithley. Two calibration cycles have been performed with a temporal distance of four months. This enables an evaluation of the temporal drift of the measurement circuitry. During the calibra-

Tab.1: Minimum and maximum average measurement deviation of the two voltage measurement ranges of the measurement hardware.

Measurement Range	Measurement deviation between measurement hardware and reference device			
	ΔV_{max} (mV)	ΔV_{min} (mV)	ΔV_{max} (%)	ΔV_{min} (%)
1: 0 – 24 V	0,98	-1,73	0,004	-0,007
2: 0 – 4 V	0,17	-0,60	0,004	-0,015

Tab.2: Minimum and maximum average measurement deviation of the three current measurement ranges of the measurement hardware.

Measurement Range	Measurement deviation between measurement hardware and reference instrument			
	ΔI_{max}	ΔI_{min}	ΔI_{max} (%)	ΔI_{min} (%)
1: 10 mA – 1 A	0,33 mA	-0,38 mA	0,033	-0,038
2: 100 µA – 10 mA	4,66 µA	-5,52 µA	0,047	-0,055
3: 0 – 100 µA	22 nA	-16 nA	0,022	-0,016

tion, different current and voltage values distributed over the individual measurement ranges were applied using the source of the reference instrument. The resulting currents and voltages are measured with the measurement hardware and the measurement unit of the reference instrument. Each measurement range is divided into 101 different measurement points, at which the current or voltage was applied. At each measurement point, 100 sample values are recorded with the measurement hardware and 250 values with the reference instrument. Using this data, a gain-offset calibration of each measurement range of the measurement hardware has been carried out. The mean values of the individual measurement points were used for the calibration and for the evaluation of the measurement deviations. The measurements were performed at room temperature. A statement about the noise performance of the measurement hardware is not possible with the used reference instrument since the noise is mainly determined by the noise behavior of the current or voltage sources of the reference instrument. In the following, the determined measurement accuracies of the measurement hardware are discussed for the supported voltage and current ranges.

Tab. 1 shows the absolute and relative deviations of the calibrated measurement hardware from the reference instrument for the two voltage measurement ranges. The relative deviations are referred to the respective upper range value. The manufacturer of the reference instrument specifies an accuracy of ±(0.015% + 1 mV) for the voltage measurement range up to 6 V and ±(0.015% + 8 mV) for the measurement range up to 40 V [10]. The deviations of the calibrated circuit from the reference instrument are very small and lie within its measurement uncertainties.

The deviations of the three current measurement ranges of the calibrated measurement hardware are shown in Tab. 2. The relative measurement deviations are again related to the respective upper range value. The slightly increased deviations of measurement range 1 (10 mA to 1 A) and range 2 (100 µA to 10 mA) are partly due to the slightly increased noise of the source of the reference instrument in these ranges. Despite this fact, a measurement error of well below ±0.1% is achieved in all three measurement ranges. The manufacturer of the reference instrument specifies an accuracy of ±(0.02% + 25 nA) for the current measurement range up to 100 µA, ±(0.02% + 2.5 µA) for the measurement range up to 10 mA and ±(0.03% + 1.5 mA) for the measurement range up to 1 A [10]. The minimum and maximum average deviations of the measurement hardware and

Tab.3: Nominal performance parameters of the three evaluated solar cells for standard test conditions (STC).

Solar cell type	No. serial cells	Area (cm²)	V_{OC} (V)	I_{SC} (mA)	V_{MPP} (V)	I_{MPP} (mA)	P_{MPP} (mW)
SLMD121H09L	9	13.0	5.67	50.0	4.50	44.6	200.7
SM141K09L	9	14.5	6.22	58.6	5.02	55.1	276.6
Solems07048032	7	15.4	5.00	12.0	3.90	9.9	38.6

the reference instrument of the three current measurement ranges are very small and are in the order of magnitude of the measurement accuracy of the reference instrument.

Evaluation Results of Energy Harvesting Devices

Three different solar cells have been evaluated using the proposed measurement system, which are SLMD121H09L from IXYS Corporation [11], SM141K09L from IXYS Corporation [12], and Solems07048032 from SOLEMS S.A. [13]. Their nominal performance parameters for standard test conditions (STC) are shown in Tab. 3. The STC are 25 °C cell temperature, irradiance of 1000 W/m² and an air mass 1.5 (AM1.5) spectrum. This corresponds to direct sunlight on a clear day for a sun-facing 37°-tilted solar cell and a sun at an angle of 41.81° above the horizon [14]. However, the application scenario targets an indoor application of the solar cells to supply wireless sensor nodes. Thus, the solar cells are evaluated under indoor equivalent illumination conditions which is typically less than 5% of STC (approx. 1000 to 2500 lux compared to 100.000lux).

As reference light source we used an LED light source (Vibesta Capra 12B) with the spectrum shown in Fig. 2 (determined with Hamamatsu C11351-10 spectrometer).
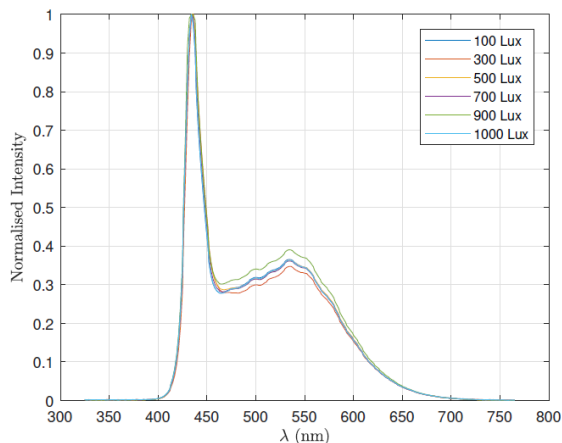
Fig. 2: Normalized spectrum of the reference light source at different illumination levels.

Fig. 3 and Fig. 4 shows the maximum power point (MPP) of the solar cells' power output densities at different illumination levels given in illuminance (Lux) and in irradiance (mW/cm²) respectively. The illuminance in Lux has been measured using the sensor OPT3001 from Texas Instruments [15]. The irradiance in mW/cm² has been measured using the sensor OPT3002 from Texas Instruments [16].
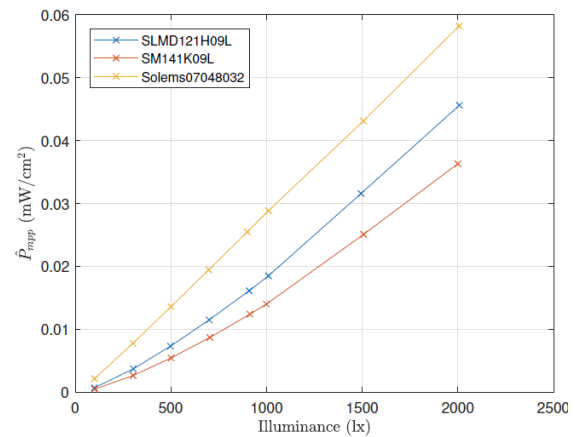
Fig. 3: Output power density of the three different solar cells depending on the illumination in Lux.
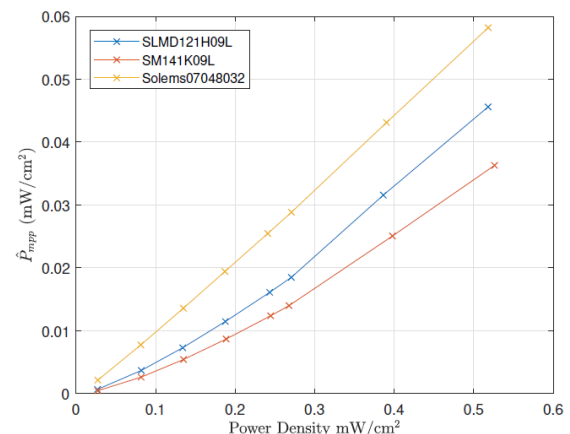
Fig. 4: Output power density of the three different solar cells depending on the irradiance power density in mW/cm².

Although the Solems07048032 has significant lower nominal performance parameter at STC, it performs best compared to the other two solar

cells at low illumination levels. This is because it is developed especially for indoor applications.

Fig. 5 shows the open circuit voltage VOC over different illumination levels. It can be seen that the Solems07048032 has the highest output voltage, which results in its better performance regarding the MPP power output at low illumination levels.
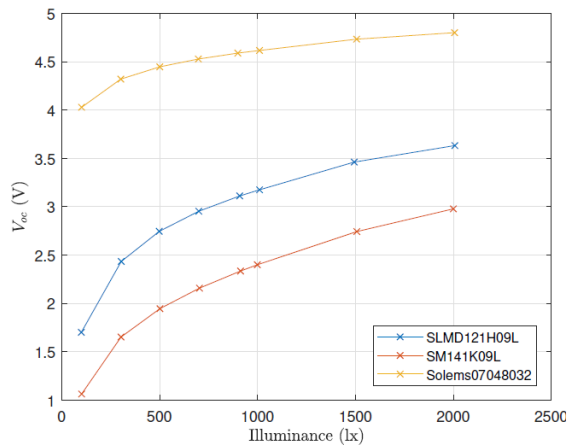
Fig. 5: Open-circuit voltage V_{OC} of the three different solar cells over the illuminance in Lux.

Fig. 6 shows the ratio of the measured MPP voltage V_{MPP} to the measured open circuit voltage V_{OC} at different illumination levels.
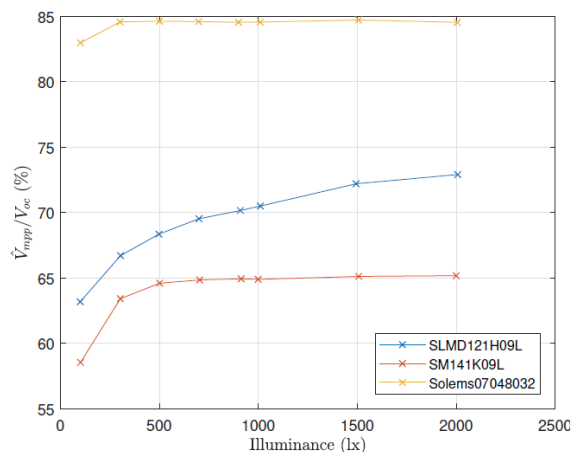
Fig. 6: Ratio of the measured MPP voltage V_{MPP} to the measured open-circuit voltage V_{OC} of the three different solar cells over the illuminance in Lux.

It can be seen that the measured ratios are significantly different to the ratios under STC (SLMD121H09L: 79.4%, SM141K09L: 78%, Solems07048032: 80.7%). These different ratios have a direct influence on the design of the energy harvesting circuitry. Most power point trackers for embedded solutions on the market measure the open circuit voltage V_{OC} and use a fixed ratio to set the MPP. Thus, they are no real MPP trackers, but it is a good and easy way to obtain an MPP approximation. However,

this relies on the voltage ratio of the open circuit voltage V_{OC} to the MPP voltage V_{MPP}. To optimize the MPP tracker's power output, the fixed ratio of V_{MPP} to V_{OC} must be set according to the measurements.

As shown above, the characterization data can be used first, to select the best solar cell for the given use case, and second, to adapt MPP trackers to the targeted solar cell and application scenario.

Conclusion

This paper presents a mobile and wireless measurement system for the characterization of EHDs. The measurement system enables the acquisition of different EHDs' characteristics for specific applications under various environmental conditions. This enhances the development process of energy harvesting wireless devices and enables a prediction of sufficient power supply. The evaluation of three solar cells reveals performance differences of the MPP power output and the ratio of MPP voltage to the open circuit voltage at low illumination levels. The results should be considered during design of energy harvesting systems in order to optimize the power supply.

Future work will target a long-term evaluation of different EHDs at different environmental conditions for industrial applications.

Acknowledgement

References

[1] M. Bal, Industrial applications of collaborative wireless sensor networks: A survey, *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, June 2014, pp. 1463–1468; doi: 10.1109/ISIE.2014.6864830

[2] H.-P. Bernhard, A. Springer, A. Berger, and P. Priller,"Life cycle of wireless sensor nodes in industrial environments," in 13th IEEE Int. Workshop Factory Commun. Sys., Trondheim, Norway, May 2017; doi: 10.1109/WFCS.2017.7991943

[3] L. B. Hörmann, T. Buchegger, and C. Steger, Optimizing the Energy Supply of Autonomous Wireless Sensor Nodes, *Microelectronic Systems Symposium (MESS 2014)*, 2014, pp. 1–6; doi: 10.1109/MESS.2014.7010257

[4] A. Janek, C. Trummer, C. Steger, R. Weiss, J. Preishuber-Pfluegl, and M. Pistauer, Simulation based Verification of Energy Storage Architectures for Higher Class Tags Supported by Energy Harvesting Devices, *10th Euromicro Conference on Digital System Design Architectures, Methods*

and Tools (DSD 2007), Lubeck, 2007, pp. 463-462.; doi: 10.1109/DSD.2007.4341510

[5] L.B. Hörmann, A. Berger, A. Pötsch, P. Priller, and A. Springer, "Estimation of the harvestable power on wireless sensor nodes," IEEE International Workshop on Measurements and Networking (MN 2015), 2015, 1-6; doi: 0.1109/IWMN.2015.7322964

[6] A. Kansal, D. Potter, and M. B. Srivastava, Performance Aware Tasking for Environmentally Powered Sensor Networks, *Proceedings of the joint international conference on Measurement and modeling of computer systems, ser. SIGMETRICS '04/Performance '04*, 2004, pp. 223–234; doi: 10.1145/1005686

[7] L. B. Hörmann, P. M. Glatz, K. B. Hein, M. Steinberger, C. Steger, and R. Weiss, "Towards an On-Site Characterization of Energy Harvesting Devices for Wireless Sensor Networks," in IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops 2012), 2012, pp. 415–418; doi: 10.1109/PerComW.2012.6197523

[8] T. Hölzl, "Drahtloses Messsystem für die Charakterisierung von Energy Harvesting Devices," master thesis, Institute for Communications Engineering and RF-Systems, Johannes Kepler University Linz, Austria, 2019.

[9] Thread Group, "Thread Overview (white paper)," revision 2.0, 2015, https://www.threadgroup.org/Portals/0/documents/support/ThreadOverview_633_2.pdf , accessed April 2020.

[10] Tektronix, "2600B System SourceMeter® SMU Instruments," 2018, 112718.SBG 1KW-60906-0, https://www.tek.com/datasheet/smu-2650-series-high-power-sourcemeter/source-measure-units-smu-instruments, accessed April 2020.

[11] IXYS Corporation, "IXYS SLMD121H09L - IXOLAR(TM) High Efficiency SolarMD Datasheet," Nov. 2016, http://ixapps.ixys.com/DataSheet/SLMD121H09L_Nov16.pdf, accessed April 2020.

[12] IXYS Corporation, "IXYS SM141K09L - IXOLAR(TM) High Efficiency SolarMD Datasheet," Sep. 2018, http://ixapps.ixys.com/DataSheet/SM141K09L.pdf, accessed April 2020.

[13] SOLEMS S.A., "Thin-film solar cells for low and high illumination," Oct. 2017, https://www.solems.com/wp-content/uploads/Solar-cells_en_17.pdf, accessed April 2020.

[14] Sinovoltaics Group, "Standard Test Conditions (STC): definition and problems", 2011, https://sinovoltaics.com/learning-center/quality/standard-test-conditions-stc-definition-and-problems/, accessed April 2020.

[15] Texas Instruments, "OPT3001 Ambient Light Sensor (ALS)", SBOS681C, Nov. 2017, https://www.ti.com/lit/ds/symlink/opt3001.pdf, accessed April 2020.

[16] Texas Instruments, ""OPT3002 Light-to-Digital Sensor, SBOS745A, June2016, https://www.ti.com/lit/ds/symlink/opt3002.pdf, accessed April 2020.

Innovative Inertial Measurement Data Acquisition and Processing for Aircraft Surveying up to Airborne Gravimetry

M. Hoss[1], Dr. rer. nat. A. Dreyer[1], Dr.-Ing. E. L. von Hinueber[1], R. Urli[2], F. Lapeyrade[3]
[1] iMAR Navigation GmbH, Im Reihersbruch 3, D-66386 St. Ingbert; +49 6894 9657 620
[2] Airbus Helicopters Deutschland GmbH, Donauwörth
[3] Airbus Operations S.A.S., Toulouse

Abstract

The paper presents the usage of modern strap-down inertial measurement systems with innovative signal processing within the wide range of applications from highly precise aircraft motion surveying and aircraft testing to airborne gravimetry. A unified hardware and software architecture is shown which allows covering quite different applications like vehicle testing or geodetic surveying with the same platform approach. High dynamics measuring capabilities as well as reliability and accuracy are key requirements for aircraft testing applications, while e.g. quite static flight conditions are the challenge for the missions when determining the Earth's gravity on micro-g level.

Between both applications, which are discussed in the paper, based on practical flight data obtained from a test mission of upgraded CH-53OBS and flight tests for aircraft certification of Airbus A350XWB (and other civil Airbus A/C types like A330neo, etc.), located for most of them over South of Europe, also flight missions from fighter aircraft up to supersonic missile navigation with similar systems of iMAR's iNAT INS/GNSS family are presented as well as the usage of these systems on optical surveying pods for environmental monitoring tasks. Beside these applications, the impact of used inertial sensor technologies like MEMS, fibre-optic or ring-laser gyroscopes is explained.

As a counterpart or in addition to airborne surveying with aircraft installed equipment, in the third section of the paper a new generation of ground-based aircraft trackers of type iIPSC-MSG with high-speed and IR cameras for surveying tasks is presented.

Key words: precise dynamics measurement, unified architecture, strap down, inertial, airborne gravimetry, aircraft motion surveying, Airbus civil and military aircraft, iMAR navigation systems, motion tracking, cinetheodolite

Introduction

Today, a huge amount of different tele-metrical tasks require various levels of inertial precision. These requirements can start at very rough km precision in WGS84 and go down to sub mm motion analysis of customer targets, which might be necessary for further correlation of different measuring sensors like vector magnetrometry or gravimetry for example. iMAR therefore developed the iNAT / iXCOM family as an inter-compatible system network consisting of modular devices and control software as well as open protocols for easy adaption in any customer usage. Devices of several sensor classes, from the highest performant free inertial class (so called strategic class) down to industrial or even consumer performance classes, are available and behave transparent for the customer in the same manner. Additionally, lots of different aiding possibilities are given - in most cases GNSS for instance is used to get direct relation to the global world coordinate system. Pressure sensors, magnetic compass sensors and odometer data for example can also be connected and fusioned by the 42+ state extended Kalman filter in real-time. Additionally, lots of different customer data or events can be synchronized and time stamped, also for later post-processing. This allows the user an easy way to switch between several systems of the same family of iNAT / iPRENA, iSULONA, iATTHEMO or iCORUS regarding his requirements.

Differences between inertial measurement systems

To choose the correct INS, it is important to understand the differences between the different types of inertial sensors. Most common gyroscope sensors at the moment are MEMS based, FOG, RLG and HRG. Accelerometers are mainly MEMS or servo types. Each type of sensor has its benefits and disadvantages, both

technically and commercially. To better understand those differences, they will be compared in the following in a short overview.

Comparison of different types of gyro sensors

First of all, we should have a quick consideration of gyro's relevant parameters. What kind of parameter do they have, what do they mean and what is the impact on the navigation solution?

Angle Random Walk (ARW) is a stochastic noise effect (generated for example due to manufacturing inaccuracies and other physical impacts) and describes the impact of the noise on the measured angle obtained from the gyro output. It is typically characterized under static condition of the gyro.

Drift or day-to-day bias is the offset to be expected on the data of the gyro after calibration. It is characterized over temperature and other specified environmental impacts.

Bias Stability is obtained as a result of Allan Variance Analysis and characterizes the instability of the gyro bias at static condition (motion as well as temperature). It is typically much lower in value than the day-to-day bias and is used e.g. for data fusion parameterization.

Maximum Rate describes the gyros measurement range of angular rate, a high data output rate mostly corresponds with a high sensor bandwidth. Other important parameters are quantization, linearity, scale factor accuracy and stability, but also group delay and many other parameters.

Table 1 shows an overview of an excerpt of typical gyros parameters.

Type: iNAT-	Gyro Type	ARW °/sqrt(h)	Drift °/h	Bias Stability °/h	max. Rate °/s	SR Hz
-RQH (dif. classes)	RLG	0.001-0.0025	0.002 - 0.0036	<0.001	800	300
-RQT (dif. classes)	RLG	0.0012 - 0.005	0.004 - 0.01	<0.001	400	400
-FLSG	FOG	0.0045	0.05	0.01	500	600
-FSSG	FOG	0.15	1	0.1	500	450
-MSLG	MEMS	0.1	5	0.1	500	490
-SLN	MEMS	0.15	3*	0.5	400	500
-SLD	MEMS	0.15	5*	2.5	500	500
-SLC	MEMS	0.26	10*	5	450	500

Table 1: Gyro performance table of different INS (excerpt)

* after algorithm converging under motion with GNSS aiding at const. temperature

Now, if we consider a moving inertial sensor (for simplification just in a simple two dimensional model), we can see that an erroneous angle determination for instance directly would lead to a misinterpretation of position due to

dead-reckoning, i.e. if heading would be assumed to be course-over-ground.
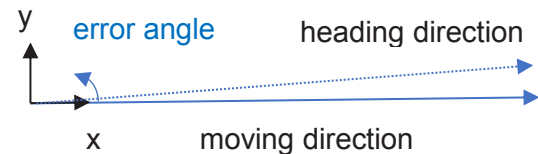
Figure 1: Position error due to dead-reckoning error.

This miss-measured angle is caused by the vertical gyro sensor error and even worse, as a gyro does not measure angles but rotation rates, the measured value has to be integrated and the heading error increases over time.

With a constant speed of 50 m/s (typical slow aircraft speed) over ten minutes, the position error as a result of dead-reckoning of three different gyro types is compared vs. each other in the next figure. For simplification, only gyro bias instability effects on heading gyro due to dead reckoning are shown.
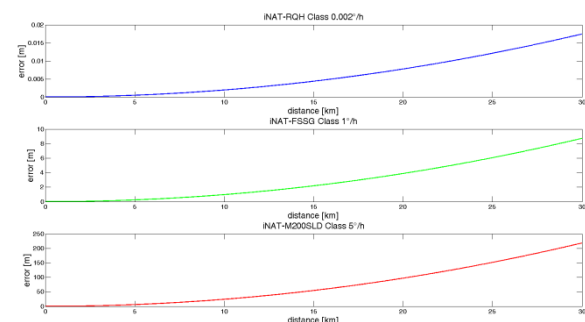
Figure 2: Deflection on a straight line with 50m/s forward motion. Displacement and angle errors over 30 km (600 s) distance due to gyro drift.

This dedicated effect results in a position error for RLGs of smaller then 2 cm, typical fiber optical gyro of around 10 m and an industrial range MEMS Gyro of > 200 m for a distance of 30 km (600 s).
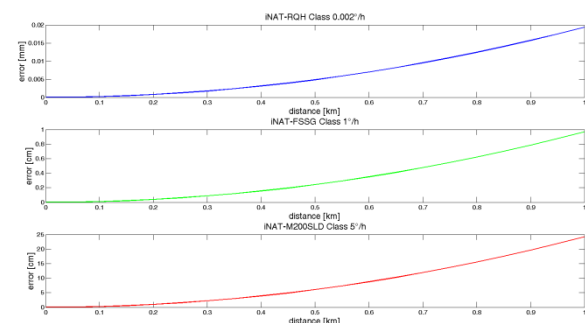
Figure 3: Deflection on a straight line with 50 m/s forward motion. Displacement and angle errors over 1 km (20 s) distance due to vertical gyro drift.

But, considered for a shorter distance of 1 km (20 s), this dedicated sensor error generates an error of smaller then 20 µm by RLG sensor performance, of around 1 cm by typical FOGs and of approximately 25 cm by industrial range MEMS gyros. Of course, real-world deviations are much larger – the here given values shall only show the relation between the different sensor classes.

So, an underrated effect for long term stable free inertial measurements is the gyro bias stability. But what does long term mean? As the effect on position error behaves in an exponential function, the shorter the time period to aid the position measurement the lower the error, but again, the better the gyro stability, the better the solution.

Considering the error for a very small period of time, it is obvious that an angular displacement is very small. In navigation applications, GNSS signals are used as complementary data source to provide position with low data rate, compared to the inertial data, i.e. usually not faster than 10 Hz. Now, even gyroscopes with high drift and instability can be used to interpolate the target position with high data rate between those GNSS based position points with satisfying overall precision, as the increase of the gyro related position error within this period is possibly still far below the error of the GNSS signal itself. It all depends on the requirements of position precision.

Therefore, it is possible to get higher dynamic information (>10 Hz) due to the higher sample rate of the sensors. This is obligatory to do fast platform controlling for example.

Gyro noise ARW describes the expectable data precision after a specified amount of time. It is an indicator of the statistical error of such a data sample depending on the sample rate of course.

The following figure shows typical values of gyro stability and noise density of different gyro types.

In general, it could be mentioned that for standard navigation, surveying and other precision requiring applications, ring laser gyros have today the most stable bias as well as the lowest noise. MEMS based inertial sensors are still far away from such precision, but became much more precise in the last few years. Fiber optical gyros are in the range between RLGs and MEMS gyros MEMS gyros today even supersede smaller FOGs.
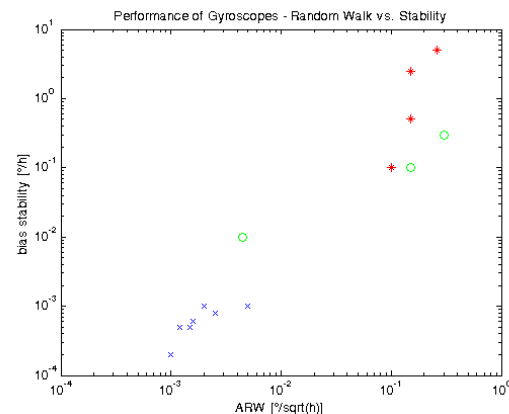
Figure 3: typical Gyro Angle Random Walk vs. stability (examples).

Legend: red +: MEMS, green o: FOG, blue x: RLG

The high reliability, performance and strong robustness of RLGs even in environment with high vibration and strong temperature gradients compared to the behaviour of FOG and MEMS based sensors has also to be considered in sensor selection. HRG sensors still lack a sufficient availability for the markets described in this paper.

The following table shows the typical impacts.

Gyro type	impact with				
	Vibration	Temperaure	Temperature gradient	Magnetism	Acceleration
RLG	low	low	low	medium	low
HRG	low	low	low	low	low
FOG	high	medium	medium..high	medium	low
MEMS	low..high	high	low...medium	low	medium..high

Table 2: Gyro environmental influence dependencies

Another criterion for usage is the quantisation noise for example. RLGs show very high accuracy, but their quantisation is quite high due to the short optical path length. Furthermore, the optical path guiding prism is dithered (mechanical vibration around each sensitive axis with about 500…800 Hz, depending on gyro type) to allow a sufficient separation of the two internal optical waves which are required to detect the angular rate based on the Sagnac effect, even at low angular rate. This dithering leads also to significant vibration on the accelerometers mounted close to the ring laser gyros, and therefore the accelerometers have to fulfil certain requirements.

So, each gyro type has its own advantages and hence the design and selection of appropriate gyro technology within an INS/GNSS system needs extended engineering experience.

Accelerometers

Servo accelerometers are used if navigation grade performance is required (high bias day-to-day performance, low bias drift over time and temperature, accurate scale factor accuracy and linearity, low rectification error etc.), while

open-loop accelerometers are used for more economical challenging applications with lower performance requirements. Both types are available on MEMS basis, while most accurate navigation grade accelerometers today still are manufactured according the "Q-Flex" principle for many decades now. Manufacturing MEMS based accelerometers with similar performance under similar rugged environmental conditions is a technological challenge, suitable and really comparable devices are expected on the market within the next few years.

But, as a simple engineering example, an accelerometer bias of 1 mg would lead only to a short time position error of 0.01 m/s² x 0.5 x 10² s² = 0.5 m within only 10 sec. This explains why a good, navigation grade inertial measurement system usually uses accelerometers with a bias of better than 50 µg.

Influences on navigation

As described above, the precision of inertial measurement depends mainly on the precision of gyros and accelerometers. To select the correct types of sensors, it is important to be aware of the requirements one really needs in the specific application. If we need mainly to bridge short GNSS aiding values (position, velocity), a MEMS based INS might be the sufficient choice, because during such short GNSS outages, IMU temperature stays comparable constant and with sufficient motion, the inertial sensor errors can be estimated quite well. On the other hand, if we have to expect long GNSS drop-outs, for example due to strong manoeuvring with significant roll or pitch, due to jamming or any other reason, or if there is the need of very precise RPY angles or position accuracy at any time (e.g. to aid SAR sensors), RLGs are mostly used. FOG based systems are used for performance requirements between both and if no significant temperature gradients and vibration impacts are to be expected (e.g. in subsea environment).

In the following, some typical applications and usages for different types of INS sensors are presented.

Flight data collection of upgraded CH-53

Airbus Helicopters has been instructed to upgrade the German Forces CH-53 helicopters due to obsolescence of some avionic equipment to a state-of-the-art Helicopter with new flight control and navigation system.

Figure 4: German Forces CH-53 Helicopter.

Airbus Helicopters uses an iMAR iNAT-RQT-4003 RLG based INS/GNSS system to evaluate the flight characteristics of al of their helicopters. In the following an example of testing the CH-53 with high accuracy is presented.

These tests took part in March 2020 near Donauwoerth / Germany. Here, an example of a measured trajectory is given.
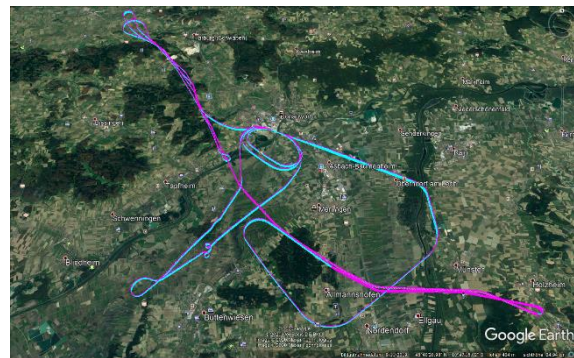
Figure 5: Example trajectories of Airbus Helicopters test flights with CH 53.

Violet: GNSS 1 Hz Data, Blue: INS 10 Hz Data

In deeper analysis of this data, different motion of GNSS antenna versus the inertial system can be found. This is explicable as the leverarm between the inertial sensors of the INS/GNSS system and the GNSS antenna is quite huge. Sideward flight and turns are shown impressively in the next figure.

The violet trajectory shows the motion of the GNSS antenna, the blue one the motion of the INS (which is located several meters away from the antenna).

All these data are available in real-time with their maximum rate of 400 Hz inside the used RLG based iNAT-RQT INS, to feed for example an online flight assistant or to determine limitations of allowed flight parameters for example. Using RTK aiding of GNSS, the 3D trajectory accuracy in real-time is about 2 cm, with an according high accuracy of 3D velocity.
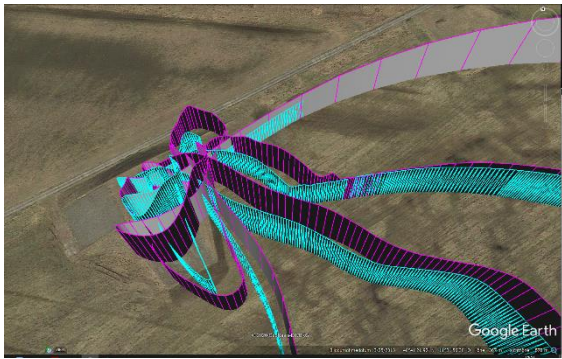
Figure 6: Example trajectories of Airbus Helicopters turn over, rotate and drift manoeuvres (zoom).

Violet: GNSS 1 Hz Data, Blue: INS 10 Hz Data

The INS also collects all relevant flight motion data of the helicopter together with precise time-stamps, which can be used for further post processing and to further improve the measurement performance in forward/backward calculation. As an option the iNAT systems can be used as PTP time server to provide accurate time stamping for all other measuring systems on board.
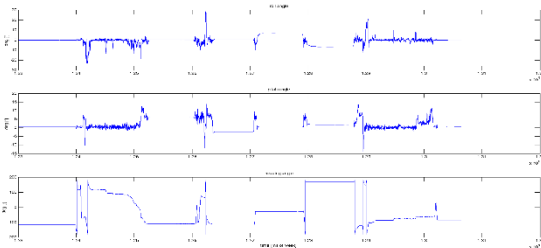
Figure 7: Acquired RPY information of the helicopter.

Additionally, of course, the data can be easily converted and processed with common scientific tools like Matlab etc. to any other format, e.g. to be visualized in GoogleEarth™. The data can be provided in real-time on several interfaces like Ethernet, UART and CAN, where an individual data decimation (in this case to 10 Hz from internal 400 Hz) and an individual data filtering with user applicable response function can be applied by the flight test engineer. Furthermore, each iNAT INS/GNSS system contains a "black-box" which allows to additionally store all INS/GNSS data (raw and processed data) on non-volatile memory.

Due to the open interface architecture of the iNAT systems, also customer specific data interfaces can be provided, like the Airbus specific IENA protocol, which can be linked into the iNAT standard software and allows delivering systems to customers with dedicated features on request to ease the operation by customer's staff.

The following pictures show a typical integration of the INS/GNSS system – all orange painted equipment and components being mounted on top of it shows flight testing equipment (due to confidentiality reasons the integration into the CH-53 cannot be shown in this paper).

Figure 8: Example of iNAT-RQT integration in H145 helicopter at Airbus Helicopters.

Figure 9: H145 with integrated iNAT-RQT during test flight at Airbus Helicopters.

Monitoring flight tests for aircraft certification of Airbus A350XWB

Airbus Operations SAS uses more than half a dozen of iMAR's iNAT-FSLG fibre optic gyro based INS/GNSS systems. They are specifically installed onboard of the development and

certification aircrafts for performing flight tests for dynamics testing as well as for certification purposes in front of the Airworthiness Authorities (e.g. EASA) of either new air-craft types, new functions or new systems.

Figure 10: Airbus A320neo, A330neo and A350XWB

The used DO160G qualified iNAT-FSLG-01 contains accurate fiber optical gyros as well as a surveying class all-frequencies / all-constellations GNSS receiver, a sophisticated integrated 42+ state extended Kalman filter, 32+ GByte non-volatile memory as "black-box" for data collection and a full airborne qualified power supply and EMI/EMC filtering. It also contains a new developed L-band based GNSS correction based on TerraStar, supporting world-wide operation.
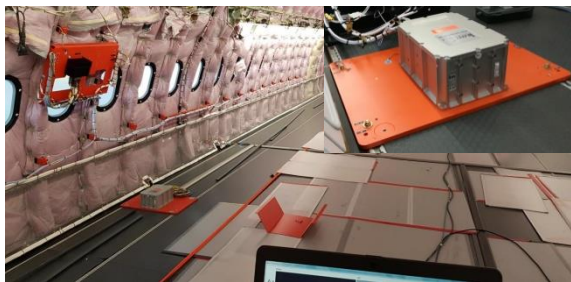
Figure 11: Airbus A330neo single installation

The INS/GNSS system is used mainly for aircraft handling performance testing purposes, both – for real-time guidance used by the flight test engineers and for post-mission data analysis.

iMAR's iNAT-FSLG had been qualified in terms of accuracy towards the A/C certification authorities and performed a demonstration of providing at least the same accuracy or better compared to the previous used traditional INS system.
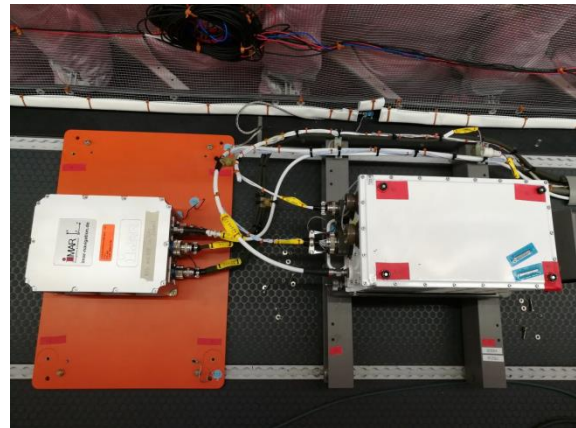
Figure 12: Airbus A330neo back to back installation of light weight iNAT-FSLG (left) and (previous used) standard aviation INS

Since 2018 Airbus Operations SAS uses iMAR's iNAT-FSLG devices in productive operation - on many different types of aircrafts so far.
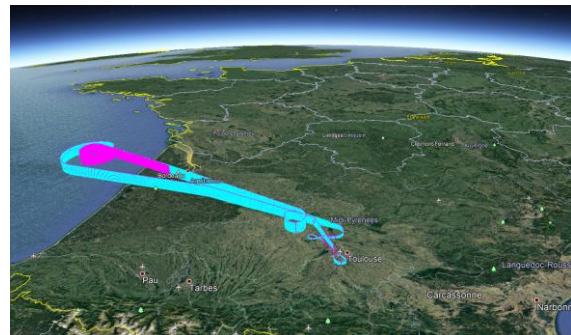
Figure 13: Airbus test flight trajectory collected with iNAT-FSLG in real-time

The following figures show the real-time measured inertial data as well as attitude and velocity. The standard deviations of attitude, heading and velocity show the superior performance of the used sensor system and online hybridization of the INS/GNSS data.

Impressive are the strong pitch angles after take-off during ascending, the significant roll angles during circular flight and the low standard deviations of measured attitude and speed. More detailed data cannot be shown due to confidentiality reasons.

Figure 14: Airbus test flight GNSS satellites used and satellites tracked, correlated to current roll angle
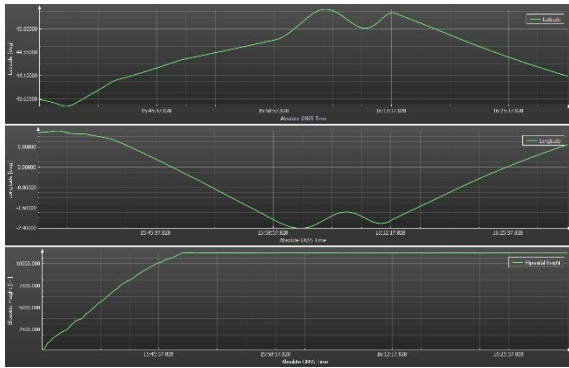
Figure 15: Airbus test flight position and altitude plot (all collected and processed in real-time)
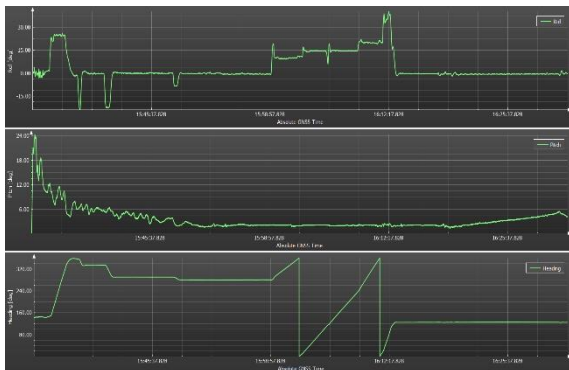
Figure 16: Test flight roll / pitch / heading plot
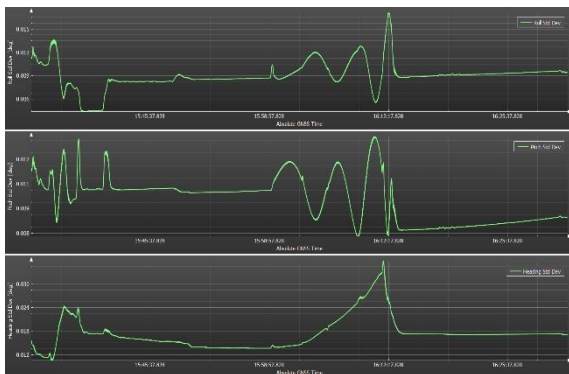
Figure 17: Test flight roll / pitch / heading standard deviations plot
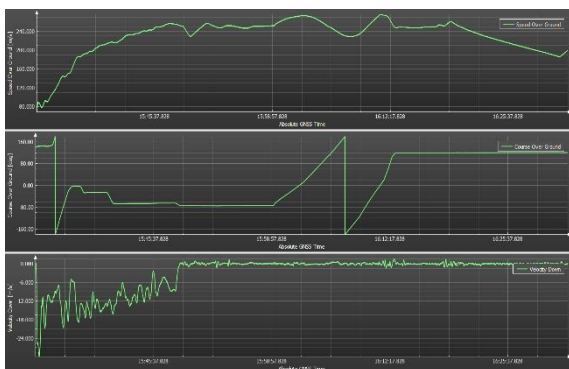
Figure 18: Test flight velocity and course-over-ground plot

Airborne Strap-Down Gravimetry with high performant RLGs

Another very interesting application for use of very high performant inertial measurement systems is the discovery of earth's gravity field disturbances. Even most modern gravity field models like e.g. GRACE gravity model still deal with inaccuracies of several meters as their wavelength resolution is above 100 km. To correct these errors, usually 2 axes gimbaled gravimeters are flown in aircrafts to observe the shorter wavelength areas in between.

The idea behind airborne strap down gravimetry is to use a very accurate GNSS signal, which is referenced on WGS84, and a very accurate inertial sensor signal and to subtract their results from each other, including all kind of coordinate transformations, compensations and so on. In general the result should be the difference between both solutions. This difference is the gravity disturbance in WGS84.
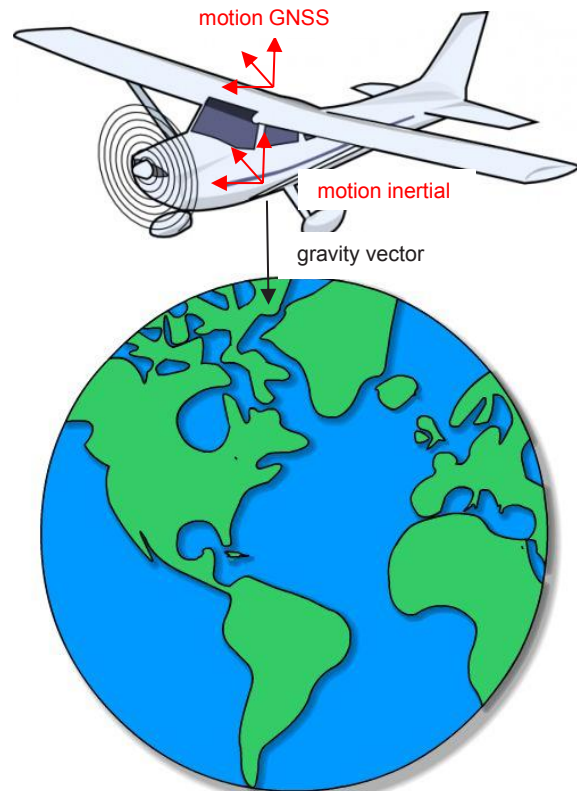
Figure 19: Principle of airborne strap down gravimetry

In simplified words, the GNSS determined motion from the aircraft is removed from the inertial determined motion and only the gravity vector is left. The variation of gravity can be shown in the post processed data.
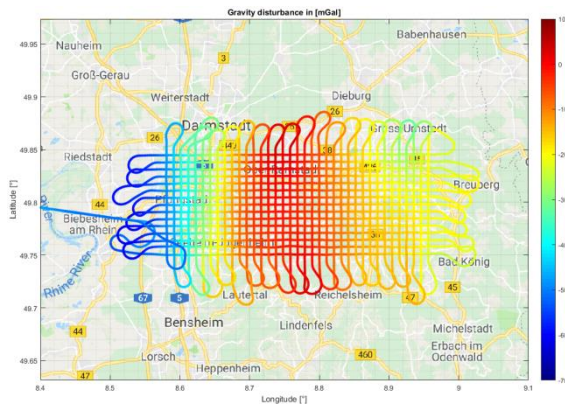
Figure 20: Bouguer anomaly measured south of Darmstadt in iMAR's Odenwald Mission 3/2018

Therefore, the dedicated measurement system iCORUS has been developed considering all significant influences on gyros, accelerometers and timing as mentioned above.

Figure 21: iMAR's iCORUS Gravimeter system

Especially thermal effects are limiting the stability of gyros and accelerometers. Therefore, a highly sophisticated thermal modelling and stabilisation has been developed and applied. Also, most accurate servo accelerometers are used as the distortion of the gravity field is typically in the range of a few mGal only; a value of 1 mGal is the required precision (remember: 1 mGal is equal to 1 µg or 10 µm/s²).

The acceleration stability of this advanced set-up within iCORUS has been verified to be better than 0.1 µg.

As the inertial measurement is to be considered as "free inertial" - at least during the flight lines for surveying - a very accurate gyro design has to be used beside the accelerometers also to minimize position and location errors. For this application, usually gyros with bias significantly better than 0.01 °/h have to be used together with a dedicated sensor modelling and treatment.

Direct trajectory measurement of flying supersonic objects with iNAT

While the previous chapters deal with the surveying of the flight behaviour of helicopters and fixed-wing aircrafts, this chapter deals with the question of how to monitor the trajectory and flight behaviour of very fast, small and agile objects. A good example is the request to survey a supersonic missile on a test range. As the obtained results of the flight campaigns in 2018 and 2019 are confidential, the setup shall still be explained. We call this method the "direct trajectory measurement" setup, while the "indirect" method is explained in the following chapter.

For controlling or monitoring supersonic targets, a stand-alone GNSS solution will not be satisfying due to several reasons:

- The GNSS data rate is comparably small: Even fast GNSS receivers provide not more than 100 Hz data rate

- Taking the GNSS position noise and standard deviation into account as well as the difficulties to obtain continuous GNSS reception on such target, the duration between valid GNSS samples might increase dramatically in real world environment and hence significant information is lost.

- To monitor the behaviour of a supersonic target flight controller or even to feed such controller with relevant data, such data should contain also angular rates and acceleration.

Test flights on a so-called "half scale Pershing" evaluation missile had been performed using iMAR's iNAT-M200/SLN INS/GNSS solution, both for passenger flights as well as for flights where the iNAT-M200 had been used for flight control. The iNAT-M200/SLN is based on MEMS gyroscopes and MEMS accelerometers and an advanced multi-constellation / multi-frequency GNSS receiver of latest generation (2019).

The performance of missile, flight controller and iNAT-M200 had been demonstrated, validated and approved at a speed > 2 Mach. Both the real-time solution of the iNAT as well as all raw data had been transmitted on a radio link to the ground station during the flights. The flight tests confirmed the highly accurate real-time solution of the INS/GNSS system as well as the capability to improve these data in post-processing accordingly.
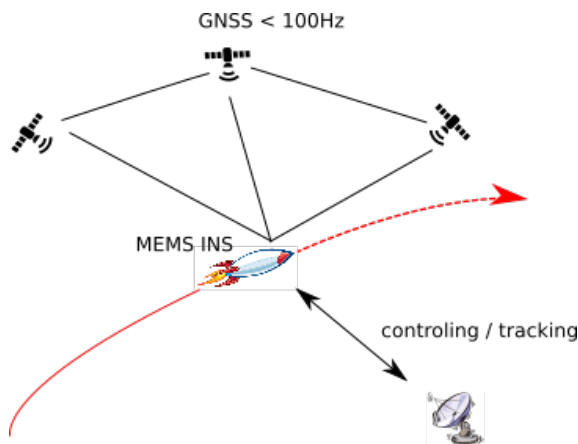
Figure 22: Example of MEMS based INS/GNSS to control or survey trajectories on high speed targets

Besides surveying or control tasks, this setup with the iNAT system is also used to transmit the target's precise location and orientation to a ground station to control the orientation of RF antennas in order to establish a high speed communication link between ground station and target.

Indirect trajectory measurement of flying objects based on optical triangulation

If there is no possibility to apply an INS/GNSS system on the flying target to obtain the its position and orientation directly as in the previous example, a ground based target tracking has to be considered. This method allows to track flying targets due to indirect triangulation.

With a certain amount of optical tracking platforms with known position and orientation it is possible to determine the target's position in any local or world frame coordinate system.
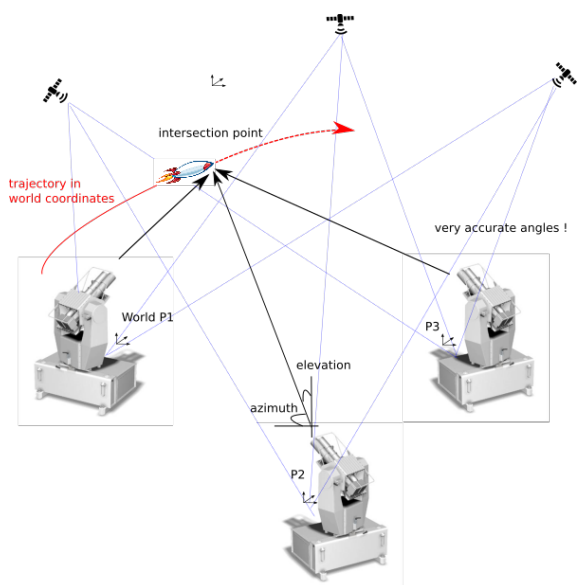
Figure 23: set up of iIPSC-MSG target tracker measuring the trajectory of a moving target

To minimize the statistic errors and to avoid singularities, usually at least a third tracker is used as well. Each optical tracker knows its own position in WGS84 or any other coordinate system from GNSS or geodetic surveying. Due to the integrated INS/GNSS solution and the high resolution encoders, each optical tracker can be operated stationary or applied on a truck or on a naval vessel, and it provides azimuth and elevation of its line-of-sight to the tracked target in real-time. The control centre calculates the three-dimensional trajectory (position and speed) and the estimated future position according to an applicable dynamic motion model of the flying target including its standard deviation.

Several test ranges worldwide are equipped with this equipment. Details are covered by confidentiality of the operators. Published details can be found under the following link:

https://www.imar-navigation.de/downloads/Documentations/iIPSC/20141128_NewDiana_Poster_01.pdf

iMAR solution: A system approach

To provide reliable and accurate inertial measurement systems, a large amount of system and application know-how is necessary.

iMAR's pretention is to give the customer an easy possibility to operate any kind of INS/GNSS solution, optionally extended by magnetometer sensors, air data sensors, odometer etc. with different types and performance classes of sensors within the same toolchain and setup.

To any operator, it does not make any significant difference whether he uses a standard MEMS INS or a high precise RLG INS. All setup, control and data evaluation procedures are similar.

For this purpose iMAR introduced the iXCOM Standard. Any iMAR iNAT device communicates and stores data according to the iXCOM protocol definition. This allows using the same post processing software (iPosCAL), command tools (iXCOM-CMD) and even customer software for any device with the same data interface, independent of the requirements of the application.

iMAR provides off-the-shelf INS as well as customized or OEM versions for an easier integration on customer side:

iNAT / iXCOM – one protocol, one family

As an open standard, iXCOM can also be implemented in customer applications to allow direct integration. Support tools like Python scripts and an SDK are available.
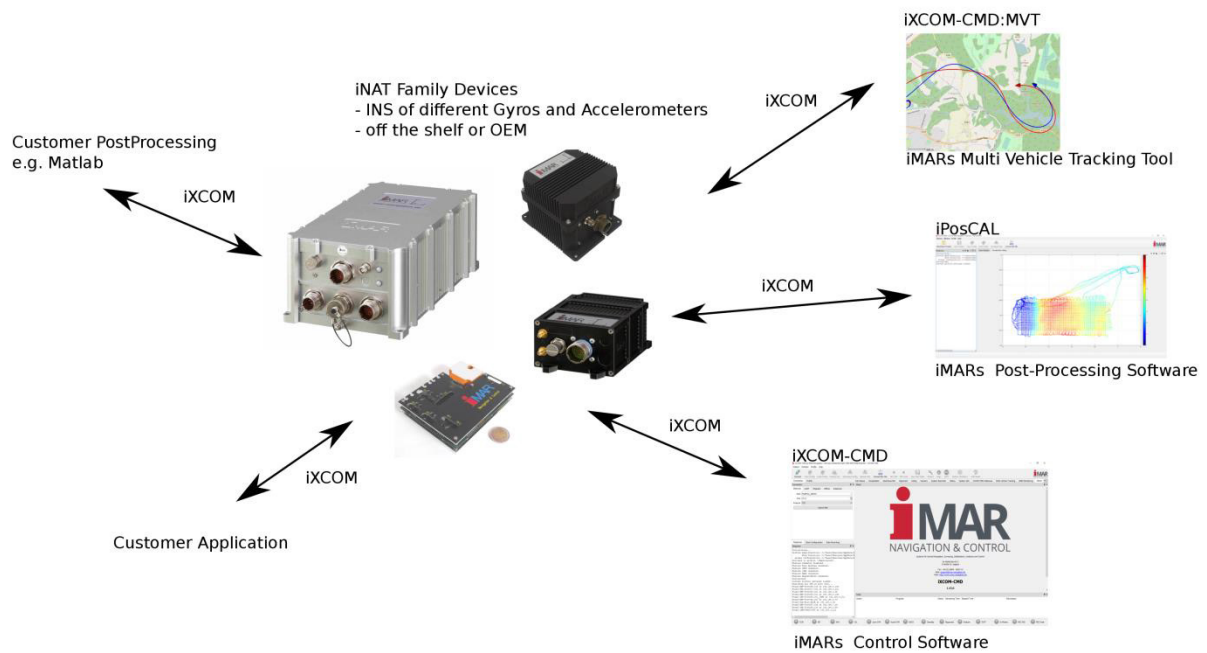
Figure 24: iMAR's iNAT Family Tree with iXCOM interface (excerpt)

Further information can be found under www.imar-navigation.de

Abbreviations

ARW   Angle Random Walk

FOG   Fiber Optical Gyro

GNSS   Global Navigation Satellite System

INS   Inertial Navigation System

MEMS   Micro-Electro-Mechanical Systems

RLG   Ring Laser Gyro

SAW Temperature Sensing for Automotive Test Bench Systems

Mag. Hannes Kurz, Dr. René Fachberger
sensideon GmbH, Ligusterstraße 4, 4600 Wels

Abstract:

Most measurement systems have evolved along with the evaluation and testing of piston engines and are currently being modified to meet the changing expectations with regards to electric engine testing. Hence, some such specific challenges require innovative and new testing and measurement solutions – e.g. Surface Acoustic Wave technology.

Key words: SAW, Heavy Duty Telemetry, engine temperature measurement, wireless sensing

Problem

The electrification of the automotive has raised much uncertainty in an entire industry and still, consumers debate about whether an electric engine can compete with the "good old" combustion engine when it comes down to convenience and performance.

Clearly, a comparison cannot easily be done but it emphasizes the high expectations towards eMobility. To improve the performance of such a powertrain system there are many variables that have to be taken into consideration. One of them is the temperature dynamic of the magnets inside the rotor. To keep track of this indicator, especially during driving operation, wired or optical measurement systems are unsatisfactory. Further, active metrology, especially one that builds up on amplifiers, usually shows limited resistance towards temperatures beyond 160°C, electromagnetic influence, high velocities and oil abrasion. As a result, new approaches come into consideration and are being developed, to speed up and improve the development process of hybrid and full-electric drivetrains. One such approach is based on surface acoustic wave technology.

Technology

Surface Acoustic Wave (SAW), is already known to be suitable for remote temperature measurement under industrial conditions [1]. The sensors are robust, work passively and due to a batteryless function, the sensors can be applied under fairly high temperatures. However, a use in the field of automotive test bench development is rather unconventional.

The system configuration described in this paper is based on the principle, where the material specific change in resonance frequency [2], brought about by a temperature change, is being measured and converted into a temperature value.

System Solution

To represent sufficient added value, one main requirement was to create a design, suitable for both, vehicle and test bench testing and simultaneously to provide a reduced system, that affects the specimen to a minimum.

The applied telemetry system (Fig. 1) uses a modular design consisting of a transponder, a 2,4 GHz antenna and the reader electronics. The transponder consists of a SAW resonator connected to a high temperature stable printed circiut board antenna. All components are tailored flexibly to meet the geometrical requirements of almost any engine design, no matter the rotor diameter, magnet type, rotational speed, oil cooling or electromagnetic interference.
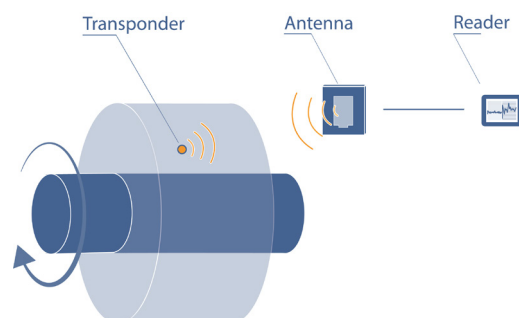
Fig. 1. System configuration SAW telemetry
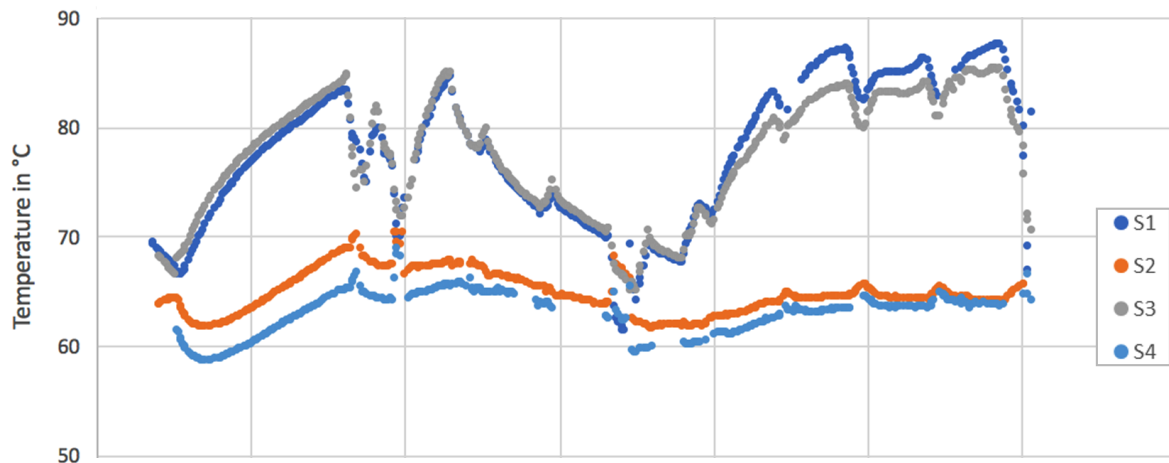
Fig. 2. Temperature plot over time (arb. unit) of a 4 channel SAW telemetry measurement with Stick Sensors S1 to S4 during vehicle prototype tests in Sweden

Due to the minimalistic design, the entire system gets mounted inside the engine housing without influencing oil jets or the performance.

Test Bench vs. In-Car Measurement

The environment of a test bench allows more flexibility when it comes to measurement systems. Cable routings may be more chaotic and it is possible to test single components of a powertrain separately. Telemetry systems can be relatively spacious since it is possible to mount them on shaft extensions for several reasons. When it comes to the testing of encapsulated modules or testing of integrated components especially during vehicle driving operation, to validate the entire system, shaft extensions and complicated wire routings become an issue.

The advantage of a SAW telemetry system is the ability to perform such measurements in any environment with no adaptions necessary. Fig. 2 shows an excerpt of vehicle measurements, recorded during driving performance tests in Sweden winter 2019. The specific challenge was to integrate a telemetry system into a prototype with no software based engine temperature management in order to show the performance potential of a new engine concept at an early stage in the means of a minimum viable product (MVP). Hybrid powertrains have even less available space when the engine is integrated into the gearbox.

SAW Resonator Performance

The measurement result shows a relatively high dynamic of the magnet temperature in certain driving modes. To capture this behavior, the telemetry is dependent on a sensor with an adequate sensor dynamic.

Fig. 3 displays the temperature step response of the stick sensor module used. After applying a temperature shock, T_{90} is reached within 1.65 seconds, representing the time the sensor takes to reach 90% of the equilibrium value.
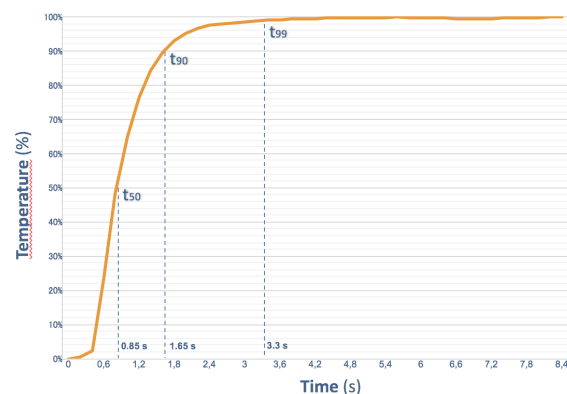
Fig. 3. Step response of Stick Sensor S

Findings

SAW temperature measurement systems are an enlargement of existing telemetry solutions. It addresses the demand to perform sophisticated measurement tasks in the field of automotive electric engine development.
The technology provides reliable and dynamical measurement data and performs such in places, that have not been accessible with conventional measurement technologies so far.

References

[1] R. Fachberger, C. Werner, "Wireless temperature monitoring in an electrolytic galvanizing plant," Proc. IEEE IUS, Prague, (2013)

[2] G. Bruckner, J. Bardong, "Wireless Readout of Multiple SAW Temperature Sensors," Sensors, 19, 3077, (2019)

Dynamic Measurement of Strain and Shape on a Rotating Helicopter Rotor Blade Using Optical Fibre Sensors

Stephen W James[a], Thomas Kissinger[a], Simone Weber[c], Edmond Chehura[a], Kevin Mullaney[a], Huseyin H Pekmezci[a], James H Barrington[a], Stephen E Staines[a], Mudassir Lone[b] and Ralph P Tatam[a]

[a]*Centre for Engineering Photonics and* [b]*Centre for Aeronautics, Cranfield University, MK43 0AL, UK*
[c]*Airbus Helicopters UK Ltd, Kidlington, OX5 1QZ, UK*
t.kissinger@cranfield.ac.uk

Abstract

Fibre Bragg grating (FBG) and interferometric direct fibre-optic shape sensing (DFOSS) instrumentation were deployed on helicopter rotor blades during a full-speed ground run to provide insights into the blade dynamics. Data were streamed wirelessly from the rotor hub-mounted sensor interrogators. Changes in strain and vibration signatures in response to a series of pilot test inputs were successfully identified and high-resolution blade shape change data has been acquired.

Key words: optical fibre sensors, fibre Bragg gratings, shape sensing, modal analysis, interferometry.

Introduction

Helicopter rotor blades incorporate complex light-weight internal structures to facilitate their operation over a broad flight envelope. With the increasing use of numerical model-based design approaches, there is a need for the development of methodologies for the validation in real flight conditions of the complex aeroelastic models. Instrumentation used in ground vibration testing, including accelerometers and strain gauges, are inappropriate for in-flight use due to the added weight and potential influence of the sensors and the cabling on the blades' aerodynamics and structural dynamics. While imaging approaches have been trialed in wind tunnels and on whirl rigs [1], their in-flight use is hampered by the large observation angle and depth of field required to measure along the entire length of the blade, along with issues related to background light and to the fouling of the blade surface. Optical fibre based approaches have the potential to meet both the measurement requirements and the demands of the measurement environment, with recent reports of flight testing of blades instrumented with optical fibre Bragg gratings (FBGs) [2].

FBGs represent a relatively mature technology, and their use in the ground vibration testing of rotor blades was described in [3], where their sensitivity to the flapping and lagging vibration modes of the blade was a result of the local curvature induced strain and was shown to be dependent on the location of the optical fibre relative to the structure's neutral axis, requiring an appropriate distribution of the sensor elements over the surface of the rotor blade. Direct fibre optic shape sensing (DFOSS) [4], based on the fibre segment interferometry approach [5] deployed in the form of three optical fibres arranged around a thin, flexible plastic former and attached to the surface using adhesive tape, was shown to allow the full characterization of the rotor blade's structural dynamics using a single sensor rod aligned along the axis of the blade [3]. Successful characterization of vibration mode shapes has also recently been achieved. Here the two optical fibre based sensing approaches are deployed on two of the four, 5 m long, bearingless main rotor blades of an Airbus Helicopters H135, with the aim of monitoring blade dynamics and operational mode shapes during a series of full-speed ground runs.

The sensor interrogators were mounted in custom designed hub support cap assemblies, with battery-powered wireless telemetry to a ground station located away from the helicopter. The hub support cap assemblies also included a kill functionality that allowed the pilot, or other protection circuits, to rapidly shut down the system for safety reasons. In addition to monitoring during a track and balancing activity, the sensor performance was assessed as the pilot provided a number of control inputs to the blades with the aim of exciting blade dynamics. A schematic of the ground test arrangement is shown in Fig. 1.
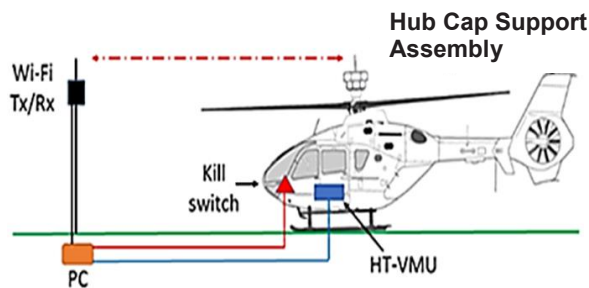
Fig. 1. Ground run configuration. The hub support cap assembly contained the sensor interrogator, battery, and local pc and Wi-Fi router. Data received by the ground station PC was streamed to the vehicle monitoring unit (HT-VMU) over an Ethernet cable

Sensors and Instrumentation

Four FBG sensor arrays, each containing 10 wavelength-division multiplexed FBGs, were fabricated in-house in hydrogen loaded SMF 28 optical fibre, with the sensors distributed along a 2.1 m length of the fibre. Arrays were bonded to the top and bottom surface of two rotor blades using cyanoacrylate adhesive, located 35 mm chordwise from the leading edge, with the final sensor in each array located at approximately 55% of the rotor length. The arrays were interrogated using a 4-channel SmartScan Aero mini interrogator mounted in the hub support cap assembly, running at a data rate of 2.5 kHz.

For DFOSS, arrays of 11 low reflectivity (~10 to 100 ppm) broad bandwidth (FWHM 5nm) FBGs, each of length 250 µm and with adjacent FBGs separated by 190mm, were in 4 optical fibres. These FBGs acted as the reflectors to form an array of low finesse Fabry-Perot cavities, which, together with the Fresnel reflection from the cleaved fibre end, formed the fibre segment interferometers in each fibre. The optical fibres were bonded into slots cut along the 1.2 m length of a D-shaped flexible PLA plastic rod using cyanoacrylate adhesive, with the fibres arranged in a rectangular lattice configuration on the rod. The locations of the reflectors in the fibres along the rod were matched to within 1 mm. The interferometers were interrogated and demodulated using the range resolved interferometry approach [6], which uses the wavelength modulated output from a sinusoidal injection current modulated telecoms DFB laser to produce a carrier signal characteristic of each reflector based on its range. This mechanically robust and simple approach offers high resolution measurement of strain (less than 1 nε/√Hz) over the 33 kHz interferometric bandwidth. The broad optical bandwidth of the low reflectivity FBG reflectors

ensured that a reflection was achieved at the laser wavelength irrespective of the strain or temperature experienced by the FBG. Differential measurement of the strains experienced by the segment interferometers in diametrically opposite fibres allowed the sampling of changes in the angle of the rod at the locations of the reflectors, which, by interpolating between these measurement points and integrating along the path, allowed the shape of the rod, and thus the rotor blade, to be determined as described in more detail in [4]. The angular resolution of the measurement at each reflector location was of order 10 nrad/√Hz. In general, DFOSS does not require strain transfer from the object under test to the fibres, but, in this case, the rod was bonded to the surface of the blade using cyanoacrylate adhesive.

The DFOSS sensing rod was attached near the leading edge of the rotor blade, running parallel to the axis of the blade and parallel to the FBG arrays. The DFOSS rod and the FBGs were covered by blade protection tape, which is routinely used to protect the leading edge of the blade from rain droplet induced erosion. A bespoke DFOSS interrogator was constructed, comprising the DFB laser, a network of couplers to allow each array to be monitored by an individual detector, 4 A-D converters and an FPGA to provide the modulation signal and undertake processing of the phase of the returned signals from each interferometer.

Results

Results obtained from one array of FBG sensors mounted on the upper surface of one of the blades are presented in Fig. 2. The data is down-sampled to a rate of 1Hz to allow the effects of the pilot's input to be seen clearly. As the rotor began to rotate (after approximately 50 s), the fibres initially experienced a compressive load as the initially drooping blades rose up to the horizontal. As the rotation rate and centrifugal forces increased, the FBGs experienced tensile strain. The rotors were held at idling speed for approximately 100 s and then increased to full rotation rate. The pilot then manipulated the controls to induce different blade conditions and repeated these three times before returning to idling speed and then running down. The influence of pilot inputs, the stepped collective input, where the pitch angle of all four blades are changed, and the collective doublet input, are visible in the data, for example at 300s and 380s, respectively. Detailed interpretation of the data is beyond the scope of this paper and will be reported in subsequent publications.
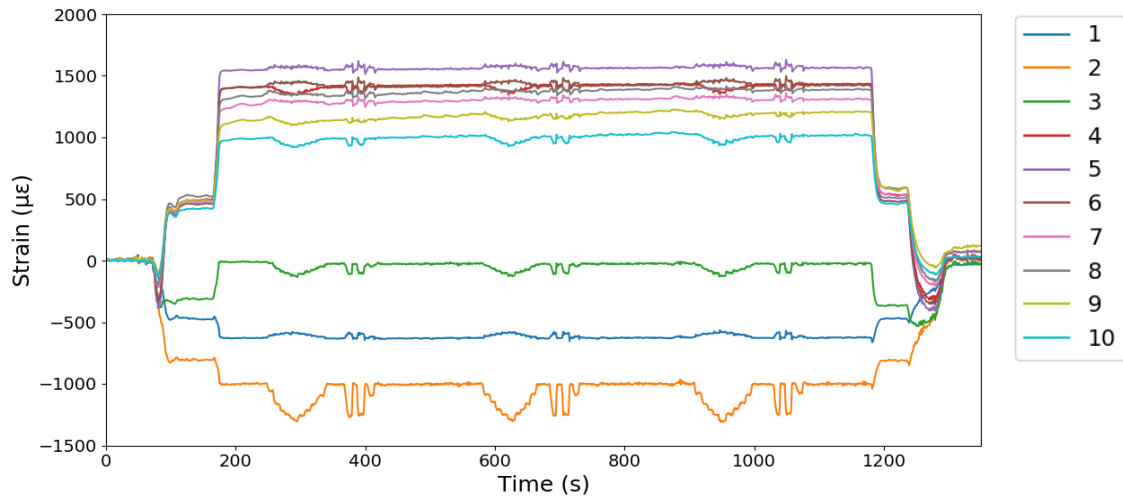
Fig 2. Time-averaged (1s) strain measured by an FBG array bonded to the upper surface of a rotor blade during a ground run on the helicopter. The numbers 1-10 denote the FBGs, with 1 closest to the rotor hub.
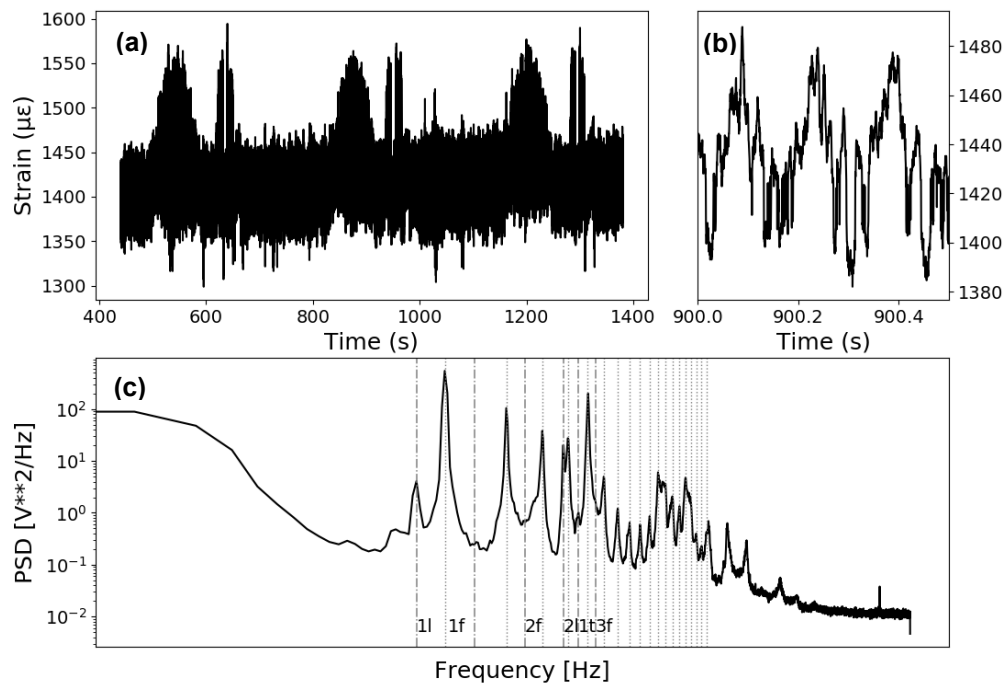
Fig. 3. (a) shows a section of the time series for data recorded from FBG 6, mounted on the upper surface of the blade at 1.76 m from the centre of rotation, while (b) shows a 0.5 s section of the data. (c) shows the power spectral density determined using Welch's method. The dotted lines indicate the rotor frequency harmonics. The dash-dot lines indicate the natural frequencies of the blade: 1l – 1st lagging mode, 1f – 1st flapping mode, 2f – 2nd flapping mode, 2l – 2nd lagging mode, 1t - 1st torsional mode, 3f – 3rd flapping mode.

To illustrate the dynamic information that can be determined, Fig. 3 shows a frequency spectrum calculated from the strain measured by the FBG located 1.76m from the centre of rotation, on the upper surface of the blade. The harmonics of the rotor frequency can be seen, as can the 1st, 2nd and 3rd flapping (out-of-plane) modes, the 1st and 2nd lagging (in-plane, perpendicular to the long axis of the rotor) modes and the 1st torsional mode.

For DFOSS, Figure 4, shows the shape changes determined by the DFOSS sensing system, in the flapping and lagging directions, as the rotor rate was increased from 0 to the idling rate of 5 Hz. For clarity, the shape changes in the vertical direction are referenced to that measured with the blade rotating at approximately 5 Hz. The raising of the blade with increasing rotor rate is clearly observed, the displacement of the blade at the outermost measurement point being 31 mm. Extrapolation

(a)

(b)
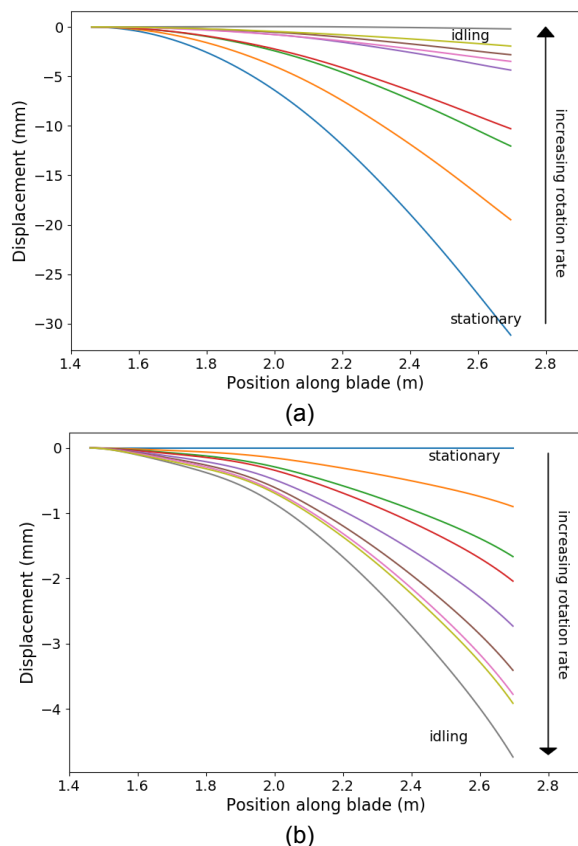
Fig.4. Shape changes measured using the DFOSS system in 2-second intervals as the rotor rate was increased from 0 (blue line) to the idling rate of 5 Hz (brown line). Each measurement presented is separated in time by 2 s. (a) Measurements in the flapping (vertical) direction and (b) measurements in the lagging (horizontal) directions. For ease of interpretation, the measurements in the flapping direction are referenced to the blade shape at the highest rotation rate, while those in the lagging direction are referenced to the shape measured with the blade stationary.

to the tip of the blade reveals a tip displacement of 200-300 mm, in line with expectations. Future work will seek to verify this independently in the rotating frame in addition to existing lab-based displacement verification experiments [5]. In the lagging direction, the bending of the blade away from the direction of rotation is clear, with a small kink in curvature data visible at ~1.9 m. The resolution of the measurement of the displacement along the interrogated 1.2 m long blade section is ~40 μm over the 33 kHz interferometric bandwidth.

Summary

Optical fibre sensing systems have been deployed on a helicopter to measure dynamically the strain and shape of the blade during a series of ground runs. Time averaged strain data allows the effect of pilot inputs to control systems to be observed, and the excitation of vibrational modes of the blade by these inputs is being investigated. For the first time, direct fibre-optic measurement of the shape of the blade in the rotating frame has been demonstrated, the analysis of which will allow the aerostatic interactions to be better understood, providing a new tool for the validation of blade designs and potentially for in-service blade condition monitoring.

References

[1] F. Boden, B. Stasicki, K. Ludwikowski, Optical Rotor-Blade Deformation Measurements using a Rotating Camera, The European Test and Telemetry Conference, 147–154 (2018); doi:10.5162/ettc2018/7.4

[2] L. M, S. Giuseppe, B. Paolo, T. Paolo, C. Franco, P. Emilio, G. Andrea, A. Andrea, A Rugged Fiber Optics Monitoring System for Helicopter Rotor, Blades, 44th European Rotorcraft Forum, 1–16 (2018); http://hdl.handle.net/11311/1065450

[3] T. Kissinger, S. Weber, E. Chehura, J. Barrington, S. Staines, S. W. James, M. Lone, R. P. Tatam, Ground vibration testing of a helicopter rotor blade using optical fibre sensors, Proc. SPIE 11199, 1119902 (2019); doi: 10.1117/12.2538810

[4] T. Kissinger, E. Chehura, S.E. Staines, S.W. James, R.P. Tatam, Dynamic fiber-optic shape sensing using fiber segment interferometry, Journal of Lightwave Technology, 36, 917-925 (2018); doi:10.1109/JLT.2017.2750759

[5] T. Kissinger, R. Correia, T.O.H. Charrett, S.W. James, and R.P. Tatam, "Fiber segment interferometry for dynamic strain measurements", J Lightwave Tech, **34**, 4620-4626 (2016); doi:10.1109/JLT.2016.2530940

[6] T. Kissinger, T.O.H. Charrett, R.P. Tatam, Range-resolved interferometric signal processing using sinusoidal optical frequency modulation, Optics Express 23, 9415-9431 (2015); doi:10.1364/OE.23.009415

Robust and Flexible Flight Test Instrumentation for Wireless and Wired Measurements

D. Foerder, G. Steiner
imc Test & Measurement GmbH, Voltastr. 5, 13355 Berlin, Germany

Abstract:

This presentation will discuss the range of tasks in flight test instrumentation that can be solved with a modular and flexible flight test system. Examples include rotor tests, landing gear tests or complete certification tests, just to name a few. Often, a common challenge is to synchronize and combine the variety of data sources across the airplane or helicopter, like analog sensors, avionics or telemetry for rotating parts. In addition, modern possibilities for handling and processing the live data will be discussed. That includes self-explanatory visualization, real-time processing and if relevant transmission to ground.

Key words: data acquisition system, rotating telemetry, instrumentation, video, flight test

Basic Requirements for FTI

Flight test instrumentation nowadays needs to fulfil a variety of basic requirements. Many of these requirements are common throughout the variety of aircraft. For example, a certification test for a small airplane requires a similar variety of test parameters as one for a large aircraft. Of course there may be differences in the number of sensors, distance between them, available space and permissible weight. In addition, large aircraft allow test engineers to be onboard while small ones rely on telemetry.

Common requirements are such as:

- Acquisition of various sensor types like strain gauges, thermocouples, RTDs, LVDTs, RVDTs, voltages, pressures, vibrations or flow rates

- Integration of avionics or aircraft control systems, commonly via ARINC-429, MIL-STD-1553B or AFDX / ARINC-664

- Synchronization between all FTI subsystems and with absolute time, typically referencing GPS time

- Ruggedness in the mechanical sense as well as against moisture, low temperatures, low pressures and vibration

- Providing as much resolution as possible. Desirable and often possible is 24 bit. Then, saturation of A/D converters can be well avoided, as headroom on input ranges doesn't cost accuracy.

- Ability to handle thousands of parameters and channels

Potential of Flexibility

Flexibility can mean fundamentally different things to different people, especially depending on the industry they work in. Some of the concepts presented here actually do have their origin in automotive R&D or railway certification testing. There, some requirements are in line with flight testing, others are not important while again others are more crucial or simply more common.

All the presented concepts and possibilities have in common that they can make the testing more productive. May that be by providing more information to the engineers, reduce time or reduce the possibility of making mistakes.

Intuitive software GUI

Displayed data is most useful to a viewer if it is intuitive. Then it does not just show numbers or graphs, but it tells the viewer whether the test is going as expected, whether all subsystems are operating properly or to what extend something has improved compared to the previous test.

Everybody however has an own opinion of what is intuitive. One often-appreciated approach is to create a software GUI that looks similar to reality. That can mean that cockpit instruments are simulated with a realistic style and arrangement.
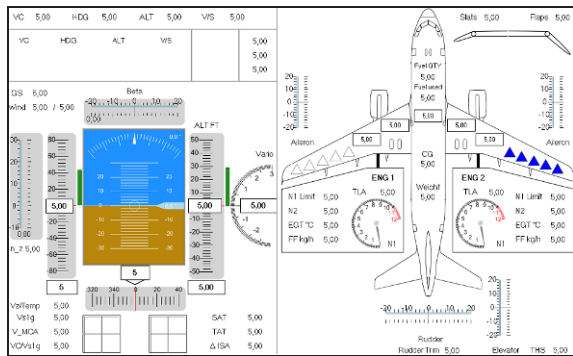
Figure 1 Example software interface designed based on customer ideas

On the other hand, many people appreciate a software GUI that looks similar to what they are used to, which is not necessarily from reality, but from an earlier generation data acquisition system (DAS).

Furthermore, sometimes the software instrument that perfectly illustrates a certain information is not available and difficult to build up by combining standard elements.

So to really achieve something most intuitive, following things have to be available:

- A useful set of combinable software instruments (widgets)

- A possibility to arrange and combine these quickly, i.e. without the need of software development

- A possibility to realize custom widgets for very specific subsystems

None of these points is a problem anymore in modern software. The third point can usually be realized by programming some C# addon or plugin to the test software, such as imc STUDIO. So a software developer is needed, however modern scripting interfaces reduce the needed development time to a few days only.

Multi-monitoring

During many tests, it is crucial that several engineers can access the live test data independently from each other. Then, an engineer might decide he only wants to see 100 out of 10,000 channels. This may not influence the overall acquisition nor what other engineers see.

This structure usually utilizes a central PC handling and storing all data, while offering live data streaming to further PCs. This can usually be scaled up to at least 15 PCs.

On each PC, selected data can be displayed with all the widgets mentioned earlier. Furthermore, engineers can decide to cut certain sec-

tions of data for themselves without influencing the overall data storage.

The number of PCs is flexible and can even change during the test. Usually, these monitoring PCs are unable to control the central data acquisition. They are purely for monitoring.

With such a structure, data is usually stored in parallel in

- the data acquisition systems,

- an on-board storage, e.g. NAS or PC and

- optionally and partly at each monitoring PC.

The concept can be used both with the PCs being in the aircraft or on the ground, receiving telemetry data.
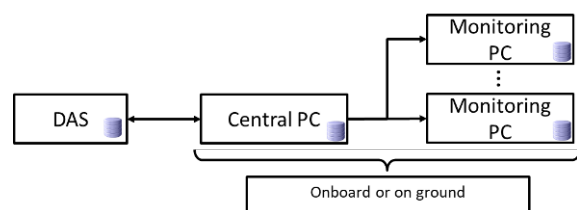
Figure 2 Structure of multi-monitoring PCs

Real-time data analysis

The purpose of data analysis is similar to the one of an intuitive software interface: to turn data and numbers into information and conclusions.

Simple examples are digital filters, slope calculations, integration or statistics. However, also advanced functions are useful like frequency-domain analysis, event detection or comparisons with reference data.

Nowadays all of these can be realized without programming. Instead, formulas are used which are commonly known from post-processing software like imc FAMOS. This enables the test engineers to implement the whole range of analysis themselves on-site without consulting external software experts.

Example formulas might look like:

- Power = Voltage * Current

- Velocity = Integral(Acceleration)

- Res = CrossCorrelation(Ref, In1, 512)

- Spectrum = FFT(Acceleration, 2, 1024)

Usually, in case of standalone applications, the data acquisition system is responsible for carrying out the calculations using FPGAs or DSPs, while any connected PCs or monitoring PCs can add further calculations, or handle those

needing rather massive processing power. The PC-based calculations can also be modified or extended during flight, again without influencing or stopping data acquisition or interfering with other engineers monitoring the test.

3rd-Party Device Integration

Often, the central data acquisition system needs to be combined with further acquisition systems or data sources. Only this combination provides the needed data for complex applications such as certification tests.

Some technology originally developed for automotive R&D can be utilized here. There, it is very common to have various subsystems, controllers and data sources and have a data acquisition combine and synchronize all of these.
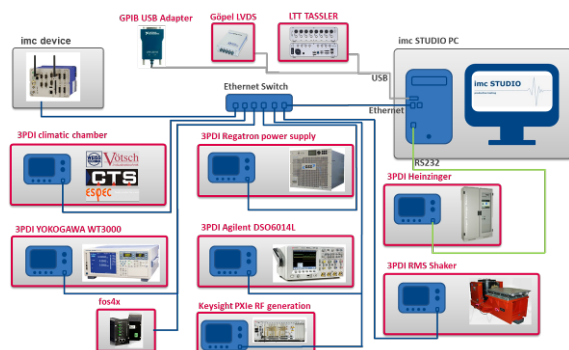
Figure 3 Example structure of combined subsystems in automotive industry

Data acquisition systems traditionally offer interfaces for field-busses like ARINC 429. A similar example is a maintenance port of a proprietary control unit. It can be seen as a non-standardized fieldbus. Such ports are often utilizing Ethernet or serial (RS232, RS422 or RS485) communication.

Figure 4 Example universal serial and Ethernet interface in a data acquisition system

To support such program-specific communication interfaces, modern data acquisition systems offer universal serial and Ethernet interface boards. They come with a basic framework

for communication and can be equipped with custom firmware to support a specific protocol.

Nowadays, it is actually very common of electronic modules to fulfil completely different tasks just with different firmware. Modern universal fieldbus modules open these possibilities to flight testing, as their firmware framework is open source and can be adapted by any developer of embedded systems.

These programmable field-bus interfaces can handle rather simple data sources like readings of pressure sensors via a low-level RS485 port, but also several thousand channels originating from a control unit.

Rotating telemetry

The most typical requirements for rotating telemetry are to capture data from propellers, rotors, turbines or wheels during taxiing or landing.

Figure 5 Example KMT telemetry measuring strain on a helicopter main rotor

These telemetry systems are usually multi-channel and relatively high-speed. An important task of the central data acquisition system is to receive their data. It needs to be synchronized and made available in the same fashion as data from any other subsystem.

That includes giving the engineers all the ways of visualization, data analysis and monitoring that they are used to. Apart from this, a customized field-bus interface can be very helpful. For example, the telemetry system might output 16 channels of strain. Instead of needing 16 analog cables to transmit the data, a single Ethernet cable is sufficient. This reduces the number of cables, weight of cables as well as uncertainty of data by replacing an analog conversion stage with a digital transmission.
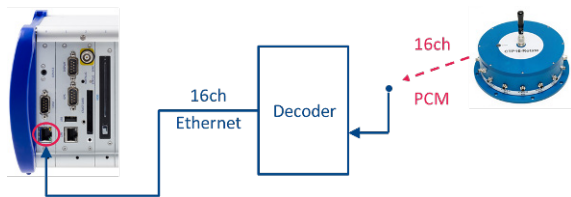
Figure 6 Digital communication between a 16 channel rotating telemetry system and the DAS

Video integration

Acquiring data in flight testing is more and more augmented by capturing videos. That can mean two different things:

- Video camera data
- Video signal data

The first point refers to cameras specifically installed on the aircraft to monitor a certain area. These cameras usually record throughout the complete test, which can create rather large files. It is however also possible to trigger video recording. The trigger can be based on some algorithm to detect meaningful events, or can be manually released. In any case, a pretrigger is also possible. For example, if an even occurs, the video starting 10 s before that can be recorded.

Video signal data in contrast is based on already existing video signals in the aircraft. That usually refers to cockpit displays. Using this approach, test engineers can understand what the pilot saw on his screen and utilize this during data evaluation.

Video signals from any of these sources are usually recorded in highly compressed format, e.g. H.265, sometimes accelerated by dedicated hardware boards.

Modularity and decentralization

Size and weight of FTI is something every flight test instrumentation engineer is interested to reduce as much as possible. In small aircraft, it can even pose serious problems as the tests gain complexity.

One recent helpful possibility to achieve this is to physically adapt the data acquisition system. A good first step is to vary the number of modules depending on the needed inputs. With slot-based devices, this only means weight saving, while there are also devices that reduce in size as modules are removed.

Figure 7 Modules with independent housings that form a variable-size DAS

A large portion of the FTI weight is the wiring. Even with full instrumentation, reducing weight can be achieved by reducing wiring. A typical way to do that is by decentralizing the data acquisition. Naturally, the location of several data acquisition systems can be varied to get closer to the sensor. However, it is also possible to distribute single modules around the aircraft.

Communication between the modules or to the main DAS is often using established field busses such as CAN bus (for low-speed signals like temperature) or EtherCAT bus (for high speed signals). Distances of 100m can easily be reached this way, combining potentially hundreds of signals in just a single cable.
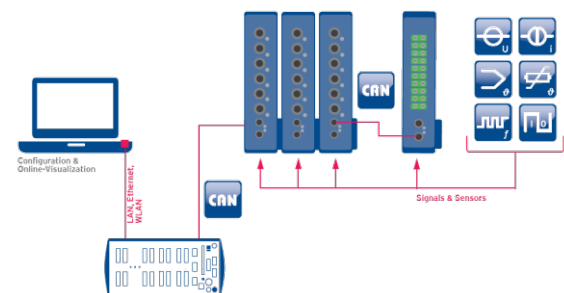
Figure 8 Example of distributed modules with CAN bus as inbetween communication

Conclusion

Flexibility of flight test instrumentation doesn't only mean being able to make changes quickly and easily. Fundamentally, it means that a data acquisition system's functionality can grow with the demands of the test engineers.

Many demands evolve after the first test flight and can actually still be implemented at that point without waiting for specialized software or hardware developers.

Utilizing such a level of flexibility, a data acquisition system cannot only cover a large variety of testing tasks, but can also provide some assurance to answer the unknown needs yet to come during a program.

Using Single Board Computers in FTI console

Óscar Gigato Rodríguez
Airbus, Paseo de John Lennon S/N, Getafe, Madrid (Spain)
oscar.gigato@airbus.com

Abstract:

The born of the first Raspberry PI computer in 2012 created a new market for low cost ARM based single board computer (SBC), initially for teaching purposes; but nowadays, after an amazing increase of computing power while maintaining costs, makes possible their use for many purposes. This incredible evolution allows their use for many tasks, including certain data processing workloads.

A **cheaper**, scalable, **modular** and more **energy efficient** system is proposed for onboard FTI consoles. It provides also an easy to use frontend dynamic web **interface**, usable from a laptop or a tablet when A/C is on ground.

Several services will be provided with this solution, like online/offline monitoring through FxS dataservers, data recording and so on. Anyway, almost every service necessary onboard can be run in these devices, which will be hosted in a distributed environment.

Key words: scalable, modular, energy efficient, web interface, cheaper

Current FTI consoles needs/services

Today's aircraft FTI consoles provide multiple services, such as data recording, monitoring and so on. In Airbus Defence and Space, several console models are designed, most of them taking into account many criteria, like monitoring displays use, data analysis, FTI data recording, either in RAW or processed mode, etc. Cameras control (when present) is also a typical work of these consoles.

Although there are many approaches to provide FTI consoles services, all solutions normally include the use of one or various PCs. These PCs run several services, like data recording, data exploitation utilities, FxS data servers and some adoc processes.

Advantages of changing to Single Board Computers

The two main advantages of using these mini PCs are cost and energy efficiency. Other benefits for using this technology are: heat dissipation, quick replacement and upgrade, modular design approaches, etc.

- Cost: SBCs are really cheap. For example, talking about the successful *Raspberry PI*, a board with 4 GBytes of RAM, 4 ARM Cortex A72 processing cores, WiFi, Bluetooth LE, two USB 3.0 ports, etc, costs less than 100€, taxes included.

- Energy efficiency: Carrying on with the Raspberry PI example, last model (4B) consumes a maximum of 15W, but it normally drains, under load, a bit more than 7W. This includes peripherals, as CPU only drains as much as 7,6W.

- Heat dissipation: Although last RPi version normally need a fan, previous versions like 3B+, or other SBCs do not need it. Only with a heat sink is enough. This is not a really huge advantage, because many low power PCs can be found in the market that do not need active cooling (normally, ultra-low voltage versions).

- Modular design approaches: The little size of these computers allows the use from one to 2 or 4 devices (or even more). As these SBCs have at least 4 processor cores, with 4 devices installed a total of 16 cores are available,

which is indeed a very powerful distributed computer.

Drawbacks of SBCs

Of course, using these devices also have some disadvantages. We are going to enumerate some of them here:

- Compatibility: Although there are SBCs based on x86 architecture, the vast majority of them are based on ARM. This fact has some advantages, as mentioned in the previous paragraph, but existing legacy software has to be recompiled and tested.

- OS availability: As mentioned above, most of SBCs are based on ARM architecture. Although Microsoft is developing an ARM version of Windows 10, is only available on certaing SOCs of ARM (Qualcomm Snapdragon). A speacial version of Windows, Windows IOT, is present in Raspberry PI board. Without graphical envirnment, is suitable for certain projects. When using a x86 platform, OS availability is higher.

- Hardware limitations: Some SBC models are limited to 1 Gbyte of RAM memory. This, that should be enough for a majority of applications run in a FTI consoles, is a limit for certain software.

- CPU raw power: Compared to x86-x64 models, ARM processors are more efficiency oriented.

- Ethernet connections requirement: Each node of this system requires its own network connection. So, if we configure a four node system, we will need four network sockets available.

Use of GPU for parallel computing

A common point for all SBCs is that, based on a SOC (System On a Chip), is the integrated graphics core, with 3D specialized hardware. As this chip is designed for massive parallel operations, and can be used by programming standards like OpenCL for performing this kind of calculations. Some libraries for C++, for example, boost::compute, can bring this power closer to the programmer. Of course, it's necessary to install a driver first.

Although there are other hardware acceleration libraries for GPUs, like nVidia CUDA or ATI Stream, this one (OpenCL) is a multiplatform and open solution, so it availability is more spreaded.
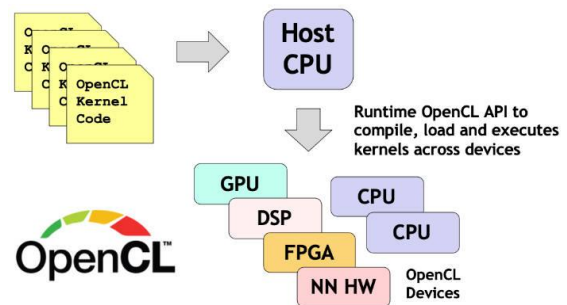
Figure 1: OpenCL task dispatching

This library allows parallel code to be executed in CPU, GPU or other devices, like FPGAs, DSP, etc.

The possibility of using this hardware acceleration is also available in classic PCs, but is mentioned here to realise that there is a lot of power available in these little SBCs. For example, there is an implementation available for RPI 3B+, and another one in development for RPI 4 (GPU of this model is different, and more powerful, than GPU of RPI 3B+).

In this use case, we propose (initially) to use OpenCL for certain heavy calculations, that are some times performed when using calculated parameters. Of course, software has to be developed taking in mind this approach, to make the most of all available execution units, that is, CPUs + GPU.

Modularized system approach

To fit FTI consoles needs, for Airbus Defence and Space, the initial proposal is based on a modularized design. That is, a system built of one, two or four SBCs working together like a unique system. This allows a great amount of scalability and flexibility that eases adaptation to a variable number of scenarios. For example, a basic system of only one SBC is possible for giving basic services, and grow when needs scale up.

As a common denominator, a web server must be present in at least one node, just to host the configuration service.

Advantages of this approach are crystal clear. System can grow according to project needs, and it can also decrease when too much power is not necessary.

The main disadvantage of using this design is the increased complexity, but it is totally affordable and it really worths. This kind of approach can also be adopted when using classical PCs, although this is out of the scope of this paper.

Hardware configuration

Many SBC devices are available in the market, and most of them can fit the requirements for their use in FTI consoles, as today regular PCs are performing these tasks. It is true that there are some devices specially designed for their use in industrial environments, like NxP products. But, as commented above, regular PCs accomplish these tasks until now, so there is no special requirement for industrial SBCs.

Three different devices have been tested by us:

- Raspberry PI 3B+: With Broadcom BCM2837B0, 4x ARMv8 Cortex A53 at 1.4 GHz. 1 GByte of LPDDR2 RAM. Gigabit Ethernet over USB2 (max 300 mbps), WiFI 802.11ac, Bluetooth, microSD of 8 gigabytes. USB 2.0.

Figure 2: A raspberry PI 3B+

- Odroid C2: Amlogic ARMv8 Cortex A-53 quad core CPU at 1.5 GHz, gigabit Ethernet, 2 GBytes DDR3 RAM. eMMC module. USB 2.0.
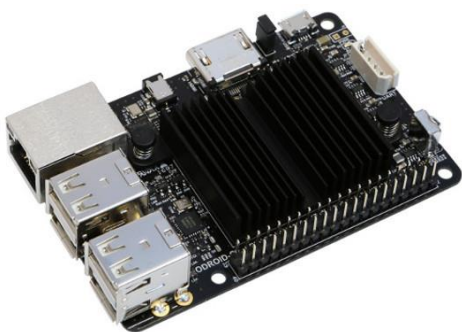
Figure 3: An Odroid C2

- Raspberry PI 4: Broadcom BCM2711, 4x ARMv8 Cortex A72 at 1,5 GHz, 2 GBytes of LPDDR4 RAM, USB 2.0 & 3.0, WiFi 802.11ac, Bluetooth 5.0 BLE, microSD of 8 GBytes.

Figure 4: A raspberry PI 4B

- Raspberry PI Compute Module 3+: Not tested yet, but one of our favorites. This is a special version of the 3B+ model, but compacted to fit in a SO-DIMM module. This allows a custom made designed board to fit one, two on up to four of these modules. Moreover, it comes with a size configurable eMMC of 8, 16 or 32 GBytes. eMMC modules are much faster than microSD, so this option is really very attractive, although board development cost has to be taken into account.

Figure 5: A raspberry pi Compute Module 3+

Software configuration

In this chapter, a description of all environment, including operating system, base and management software,etc.

Operating system

Nowadays, there are a general availability of various flavours of Linux distributions for these machines, although a special version of Microsoft Windows, Windows IoT, is also an option. As we have tested many devices, a description of different choices will be explained here.

- Raspbian: A Linux distribution, now known as "Raspberry Pi OS", is a debian based distribution for Raspberry PI. This is the more extended option for Raspberry PI. Backwards compatible, it runs from original Raspberry PI model

B to latest 4 one. A new 64 bits version is available, but is still beta version.

- Ubuntu 16.04: This version of Linux was installed in Odroid C2. It is also based in Debian. It's a 64 bits version, compiled for ARMv8 architecture (includes many instructions for specific purposes).

- Windows IoT Core: This version of Microsoft Windows was created for Raspberry PI 2, and it has been updated to run in newer versions of this hardware. It is designed for running only one application in the foreground. The main disadvantage of this proposal is that it has a license for commercial purposes.

Once described the three alternatives tested in these devices, let's speak about our conclusions.

- Advantages of choosing Windows IoT Core:

    o Existing WIN32 applications are easy to port.

    o DotNet apps are also portable. In boths cases, not all desktop API is available in Windows IoT Core.

- Disadvantages of Windows IoT Core:

    o Has a cost per license.

    o Only one foreground application is possible.

    o Closed source product.

- Advantages of choosing Linux:

    o It is a full desktop OS. Many applications can be run in foreground.

    o Open Source project with great support. This is especially true in Raspberry PI OS.

    o Hundreds of packets that can be installed for almost any purpose.

    o DotNet Core is available in Linux environmnent, and future developments seem to follow this path.

    o It is a very lightweight Operating System is well configured, ideal for this kind of devices.

- Disadvantages of Linux:

    o Applications need to be port to this operating system.

    o Additionally, when working with ARM architecture, some other kinds of changes have to be performed. For example, ARVv7 architecture does not permit an indirect access to memory.

    o Developing tools are more powerful in Windows environment. For example, Microsoft Visual Studio is a reference for developers. Although last versions are capable of developing and debugging using a Linux host, there is still a gap between the two platforms.

Comparison between Raspbian and Ubuntu has no sense attending to the orientation of this analysis.

Master/Slave

In this system, one node takes on the role of master. It will host the responsibility of managing the processes run in other nodes, and gather for the status of them. It will also host the web server for the managing application (commented ahead).

Services

The common services to be run in these consoles are the following:

- Data recording in RAW format: *dumpcap* (or similar). This is a very light application, whose main requirement is enough bandwidth to storing system. A USB or microSD can be used.

- Data recording in PFF (Parameter Fast File) format: Performed by our *pffrecorder* application, written in C++, requires a high IOPS storing device. Performed in a fast USB3 stick memory, or external USB3 SSD disk.

- *Online* FxS Dataserver: An application that reads IENA packets and serves parameters to any client that asks for them. Written in C++, runs smoothly in a Raspberry PI 3B+, managing a data flow of 60 Mbits per second.

- *Offline* PFF Dataserver: An application that access PFF files (recorded by *pffrecorder*) and serves data to clients. Is very lightweight.

- Other little tools.

Future Services (in development)

- Web server: Hosts the managing/configuration service. Uses dotNet Core technology.

- Web client: Will use *dotNet Blazor* technology. *Blazor* is a new framework from Microsoft that enables the use of C# code directly in a web page. Dynamic web pages can be built easily with this tool, and the use of dotNet Core, an open source implementation, available in many platforms, as x86/x64 Linux, ARM Linux for raspberry PI and others, guarantees a long term support to this technology. Code can be run directly in the browser (with WebAssembly), or in the server.

The optimal use of this infrastructure is using a WiFi connection. The ideal scenario would be turning on the WiFi when the aircraft is on ground. Then, the user will come to the aircraft with a tablet/cellular phone (a laptop would work too, of course). Opening the internet browser and connecting to the server would show the user interface of all the system. Different pages/tabs are planned for this application:

- System status: Will show a brief description of the system: number of nodes, active services, configuration in use, system alerts, etc.

- Services configuration: Will show services available, services configured to run, and associated node to run each service.

- Configuration database: For checking, loading and applying configurations (FTI parameters specification).

- Profiles/modularization: Each node can assume one or more roles. Depending on system expected load, services will be distributed among existing nodes.

- Software update: This tab will allow an update of software versions running in the system (binaries). It will also show current versions.

- Logs/restart: A tab where system logs can be viewed, downloaded and deleted. It will also show an option for system restart and/or shutdown.

The idea is to simplify the management of all the system in a user friendly application, easy to use and powerful at the same time.
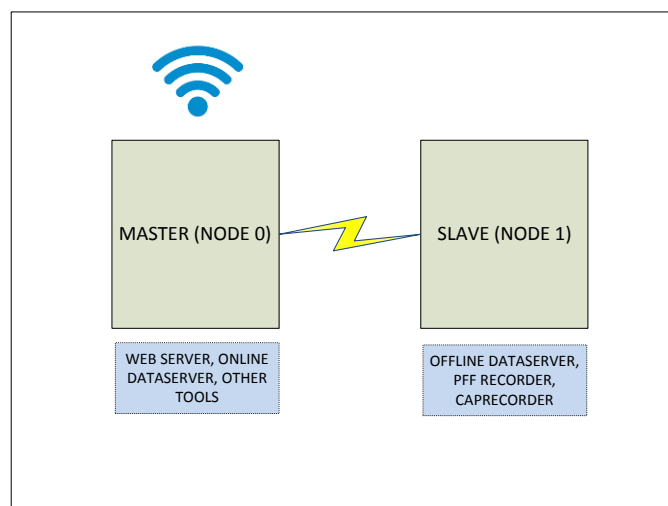
Figure 6: A typical system configuration

Test results

As we have port all base services to linux/ARM environment, we can resume our personal experience with these firsts testing stages.

- Speed: Several improvements have been made to our C++ source tree, in order to be more stable and faster. We have not seen any problem with speed, so our *Online* data server starts in about 9 seconds in a raspberry pi 3, with an A400M configuration (really big, more than 300k parameters). *Odroid C2* is a bit quicker, but barely noticeable. Raspberry PI 4 is much faster,

starting our application in a bit more than 5 seconds. Regarding CPU use, raspberry PI 3 managed with no problem a data flow of almost 60 Mbps.

- Memory utilization: Continuing with our example, *online* data server uses 274 Mbytes of RAM memory with no clients are connected (more than 300k parameters). So, thinking about its use in an onboard console, where clients demand no more than 3000 parameters, one of these SBCs will manage with *no suffering* all clients' requirements. As all these devices have 4 core processors, multithreading applications, like this one, will benefit of this configuration. There is still room for more processes in a raspberry PI 3 running the online data server.

- Multithreading: As commented just above, one of the key factors for the viability of using these devices is the multithreading programming approach. As IPC (Instructions Per Clock) of ARM Cortex A53 is much lower than regular PCs processors, due to many reasons, like in-order design vs out of order one, cache sizes and so on, squeezing each core is possible when multiples threads are running. Our services design are heavily multithreaded, and have been optimized for a slow footprint in both CPU and memory use.

- Energy consumption: This is where these devices bright when compared to x86 alternatives. As ARM architecture is designed primarily for energy efficiency, these processors have a really small footprint in energy requirements. For example, raspberry PI 4, the hungriest device of all tested, demands a *peak power* of less than 8W. This device is really powerful, as it features a 4 core out of order processor, and up to 8 GBytes of RAM (2 GBytes our sample). Two of these devices should be enough to running all of our processes smoothly and would consume a maximum of 30W, including all necessary devices connected to them. Moreover, two raspberry PI 3b+ or two compute modules should be enough too, with a bit less power requirements.

- Access to disk: There is only one application that requires a lot of IOPS to run properly. This is *pffrecorder*, and this is true specially when too many parameters at high bit rate are being recorded

(like A400M). If we are using compute modules, internal eMMC will cover our requirements. Also, with Odroid solution, which uses eMMC, will also be enough. Raspberry PI 4, with their USB3 ports, will also perform well. The only case that would cause any kind of problem is Raspberry PI 3B+, because it only features USB2 ports.

- Upgradeable: Upgrading this kind of design is very easy, as all Raspberry PI have the same size since their first model, Raspberry PI Model B, born in 2012.

Cost analysis

SBC devices are really cheap. You can buy a Raspberry PI 4 with 2 GBytes of RAM for only $35. You can also buy a Compute Module 3+ with 32 GBytes of eMMC for only $40. But, this is not the only cost of the system. For building it, you need to be concerned by:

- Peripherals: microSD or external SSD/USB stick.

- Development costs: As our software was already port to linux, base software adaptation cost has been very low (mainly configuring environment and a few architectural bugs over ARM). Managing service development is pending, but this will benefit both PCs and SBCs architectures.

- Ad hoc hardware design: If Compute Modules 3+ (or more recent models when available) is going to be used, designing, engineering and testing custom boards will be the highest cost. But this development could be very useful, as it would be very compact and expandable.

Taking into account all these aspects, we can estimate the cost of a system based on SBCs:

Two Raspberry PI 4 + two microSD for booting + one external SSD (500 GBytes): aprox (including taxes in Spain): 188€.

Using a PC for these purposes is much more expensive (more than 1000€).
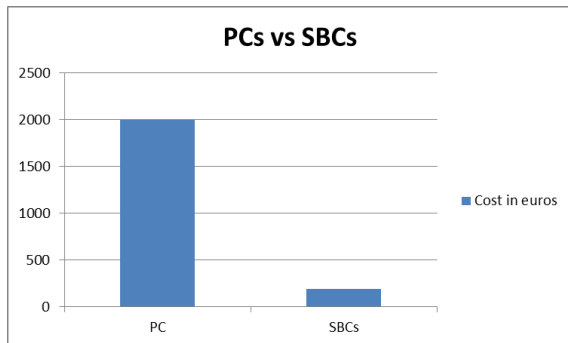
Figure 7: Cost comparison between PCs and SBCs

Conclusions

Single Computer Board devices have been a revulsive for "cheap" computation. Raspberry PI has founded a new category in computer devices. Originally thought as a teaching platform, its evolution has converted this "toy" into a real computer, powerful and cheap enough to handle very hard work. It is very cheap building a multi device system with a great power as a real alternative to classic PCs.

Last version of Raspberry PI can be configured with up to 8 GBytes of RAM, enough for storing medium databases. For our use cases, 2 GBytes is enough for handling with our processes. Even 1 GByte should work.

Using these SBCs can drop the cost of building a console for FTI, at least the PCs for computing tasks. The scalability of these devices is also a plus to have into account.

A new modular electronics approach applied to instrumentation units

Ghislain Guerrero[1], Valentin Chomel[1], Floriane Monteil[2], Olivier Pinto[1]
[1]_Safran Data Systems, 5, avenue des Andes - 91978 Courtaboeuf Cedex, France_
[2]_Safran Data Systems GmbH, Friedrich-Ebert-Str. 51429 Bergisch Gladbach, Germany_
ghislain.guerrero@safrangroup.com

Abstract

One of the main challenges of an instrumentation engineer is to provide accurately the required information while preserving the test vehicle inherent behavior - which, somewhere, breaches the universal Heisenberg uncertainty principle.

This article presents the results of a technology investigation based on an original modular 3D electronic design. This technology aims at minimizing the intrusiveness of an instrumentation system while offering flexibility, scalability, reliability and performances.

Based on a return of experience from the legacy and current instrumentation systems, the study also addresses the daily operational constraints faced by an instrumentation engineer: auto-configuration, plug and play, fast reconfiguration, harsh environmental conditions,…

Key words: FTI - Intrusiveness – SWAP – 3D electronics – IOT

Background

Flight Test Instrumentation is at the heart of several challenges. It requires collecting the most precise and reliable data with minimal influence to the vehicle being tested. It must minimize intrusiveness while maximizing data quality. In addition, it is used at a critical time, just before market introduction and mass production. The financial stakes are high. To allow for fast delivery, it must therefore also be reliable, flexible and quick to set up. This is essential in order to enter the market and refund R&D investments as quickly as possible. This instrumentation must also be scalable to adapt to all cases (from a few measurements to tens of thousands in the same test vehicle among commercial or military aircrafts, fighters, helicopters, space launch vehicles, armaments, maritime vehicles, automotive,…). It must also

be dismantled without leaving signs after tests or a certification campaign. The instrumentation of a production vehicle does not allow neither drilling of mounting holes nor the use of mods but must allow fast and efficient troubleshooting. In real life, the complex process of validation and certification of an aircraft is very precise and perfectly mastered by aircraft manufacturers, but a test campaign is also full of unknowns which force the test engineers to react to unexpected events and therefore the instrumentation team has to adapt to new requirements, as quickly and efficiently as possible.

Legacy instrumentation systems are not natively suited for all these use-cases. They require compromises, modifications, time and work to get results that are not always satisfactory.

Global trend in Flight Test Instrumentation

Instrumenting an aircraft to successfully succeed in getting flight certification is not an easy task. It is a comprehensive area of expertise that has pushed each product generation to adapt, improve and take advantage of new technologies to achieve maximum efficiency in flight test campaigns. However, we must approach these changes and system improvements with the choices inherited from the past in mind.

Flight Test Instrumentation, in its mission to ensure valuable flight test results, was initially led by the safety of data storage and the first needs of specific physical value acquisition. Since then, the main challenges are to get proper data to record, display and transmit to ground stations. The measured data must also be accurately synchronized on a common time base. It enables the engineers to process them at the test vehicle scale and understand complex physical effects that are affecting the behavior of the vehicle.

Historically, instrumentation systems used to be centralized, installed in cabin (e.g. 19-inch

racks) [1]. This architecture centralized the entire amount of data. Such systems ensured accurately timestamped synchronous data acquisitions.

These systems, sole hearts of the instrumentation, then combined all needed operations in one location: they could acquire, process, record, stream out and telemeter from one unit, therefore offering a reliable solution. The different applications requiring very specific features were not left aside: customization was made simple because of the centralization. Although natively designed to fit the FTI requirements, the centralized systems showed limitations. One unit gathering all data means several connections through the vehicle to the data sources: sensors, buses, cameras… and all the constraints that follow. For example, the interferences on cables corrupting data could make transmissions erroneous (bit swapping). In addition to the data corruption risks, the space available in the vehicle is limited, and so is the room left for cables.

The need to reduce the number of flights necessary for a test, to maintain or even improve the level of performance of acquisitions, the reduction of electronics footprint and the emergence of new processing capacities (on board or on the ground) with "BigData" has triggered the possibility of an evolution, going in the direction of decentralization, more data, more flexibility, more precision and multiple optimizations.

Over the past 15 years, instrumentation systems have become more decentralized. The advantages are for example better data accuracy, fewer cables, less weight, less impact on the tested vehicle. New modular equipment has come into operation, able to adapt to many specific applications in aircrafts, rotorcrafts, UAV, but also missiles and launch vehicles.

From sensors to telemetry, new standards, methods and concepts emerged. Network architectures have become more distributed and very efficient using standards such as Ethernet for fast data transfers, without compromising quality, even over long distances. Today, there is no longer a central unit but remote devices smartly spread in the aircraft, located in the few "empty spaces". Reducing the distance from the sensor to the signal conditioning and acquisition system improved the quality of the data. It can be digitalized as soon as acquired, in a much higher quality thanks to improvements regarding bandwidth and storage capacity. With modularity inside high performance small units,

those distributed architectures address a lot of use-case specific requirements.

Wireless protocols can also be used for more flexibility (for the instrumentation of a helicopter rotor for example). [2]

However, those systems are still aggregators of acquisition, processing and communication capacities. They are more distributed than the previous generation but they still have a significant impact on the vehicle linked to their installation and operation. This paper presents the thoughts carried out on the future of instrumentation as well as its results.

The Internet of Things (IoT) Trend

The Internet of Things (IoT) is a system of interdependent devices connected to a network and/or to one another, exchanging data (autonomously in general). Basically, the IoT is a collection of electronic devices that can share information among themselves.

From "ubiquitous computing" in the 1990's to the emergence of Industrial IoT (IIot), the IoT market grew exponentially from zero to several trillion dollars. Electronics have considerably reduced in size (down to being smaller than connectors in some cases), energy consumption has drastically decreased enabling life-time of years on batteries, networks have become wireless and more and more efficient, embedded or cloud computing capabilities followed Moore's law.

The information value loop (Use of sensors to generate information, transfer and collection of this information, discernment of patterns or relationships between phenomena, enabling descriptions, predictions, or prescriptions for action leading to initiating, maintaining, or changing an event or a physical state) [3] has found applications in all areas of industry and life (Examples include smart factories, smart home devices, medical surveillance devices, portable fitness trackers, smart city infrastructure, vehicle telematics, among lots of others) to the point that today there are 5 times more IoT devices in the world than humans (and still a two digit growth).

Limits of the current modular instrumentation electronics

As previously discussed, an instrumentation device needs to be scalable and modular in order to offer the best size, weight and power (SWAP) efficiency. In this respect two main categories of mechanical architecture exist on the market: the racks and the stacks. Both offers advantages but also drawbacks:

- Racks: quick access to each module but fixed size with step effect when switching from a size to another (e.g. from 9 slots to 16 slots). Usually less ruggedized and compact than stacks.

- Stacks: scalable size defined by the exact number of module needed by the user but changing a module requires opening the stack. Most of the time, more ruggedized and more compact than racks

In both cases, the modules used for the racks and the stacks are based on a printed board circuit (PCB), an external connector and one (racks) or two (stacks) internal connectors. Both designs lead to parallelepiped volumes which hit certain limits when downsized:

For interconnection between modules, the design engineer is facing a challenging alternative:

- Apply the downsizing to internal connectors with very tiny, small pitch and low profile connectors, as the one used today in most of our mobile devices, that are proven solution for high speed communication. However, these kinds of connectors have also shown their limits in terms of ruggedization and usually do not cope with:

  - Repetitive assembly and disassembly by the end-user who needs even sometimes to work directly on the field

  - The mechanical harsh environments frequently encountered in flight testing where overall vibrations and transient shocks levels can be very harsh / severe.

- Do not apply the downsizing to the internal connectors and keep legacy flight proven part numbers. In this case the volume ratio between the interconnection items and the electronic parts hosting the features becomes too big

For the external connectors, there are traditionally two kinds of connectors: The ones used to connect the signal/data source (sensors, voltage lines, communication buses, …): the "D-type" and the cylindrical connectors.

The "D type" connectors, with the famous Sub-D type connector, are the natural choice, whatever the stack or rack design. They offer the best profile in terms of pin count when mounted at the extremity of a PCB. The downscaling of devices leads the connector manufacturer to propose higher density Sub-D HD and miniature versions: micro-D and even nano-D connectors. The main drawback of these connectors comes from the fact that, at a certain point, they do not offer anymore the minimal wire gauge #24 usually asked by the instrumentation engineers and also no more crimpable contact and only pigtail type connection that prevents from any last minute modification of the harness which is a key asset in flight test domain.

The cylindrical connectors, as the famous 38999 series, are the reference connectors in aerospace domain to establish connection between devices. However, for modular systems, this kind of connectors mostly requires intermediate backpanels that routes the signals from the flat PCB based modules to the connectors, affecting badly the scalability, the versatility and the overall size of the instrumentation device

Leveraging the 3rd dimension

More and more electronic technology offer the capability to take benefits of the 3rd dimension in order to optimize integration factor, placing more functions in a given space while optimizing the distance between the electronic components. 3D electronic integration can take place at different level:

At PCB level: rigid/flex technology allow to fold in 3 dimensions a flat PCB composed of rigid parts where the components are mounted and flexible ones insuring the interconnections between the various rigid ones. This technology offers the possibility of a secured connection of device components with the assurance of polarity and contact stability, as well as a reduction in plug and connector components [4]. The rigid/flex technology exists at various scales and complexity levels (number and nature of layers stacked together).

Figure 1: example of rigid/flex PCB technology

Component package: System In Package (SIP) consists of a number of dissimilar integrated circuits enclosed in a single highly miniaturized package. The SIP performs all or most of the functions of an electronic system, and, it can

contain several silicon components (bare die or package) and passive components. Safran Data Systems has successfully experienced the 3D integration of the versatile analog frontend of ANA module issued from the XMA Digital Acquisition Unit (DAU) widely used in flight test instrumentation. This 3D integration has been conducted using the 3D-Plus technology, one the key player in innovative 3D electronic packaging [5].
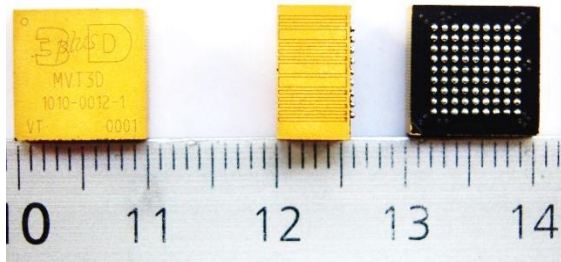
Figure 2: 3D integration of an analog front-end using SIP 3D packaging technology

µMA concept

Inspired by the flight test community feedback and the various trends discussed above, Safran Data Systems has decided to investigate disruptive paths in order to identify and mature the technology bricks needed for a new kind of instrumentation systems. A bank of new ideas have been generated during a set of creativity sessions where design engineers (hardware, software, mechanics, industrialization,…), product owners, sales managers, end-users (test and instrumentation engineers) from the aerospace domain but also from automotive and industry in general, met together to share their return of experience and dream of the future. These sessions were catalyzed by designers and ideation experts in order to stimulate creativity and conceptualize the main ideas.

One axis of the exploration process was about an alternative form factor to the parallelepiped housing. The main drivers of the exploration process, presented hereafter were: ease installation, ease daily operations and minimize intrusiveness.

Ease installation

All the chassis and stacks on the market requires the instrumentation engineer to design specific mounting plate in order to install the devices in the test vehicle. This step is costly, time consuming and affects the reactivity facing changing conditions. Among other ideas identified to address this pain point, the cylindrical shape appeared to be an interesting alternative. Consequently, it has been decided to baseline the concept on a tube housing. It is

a good fit to cylindrical connectors mounted on both ends and allows to use COTS brackets, clamps or straps already used in the vehicle itself (to maintain harnesses and pipes) so already available as in the inventory of the final user. They are affordable, exist in different sizes and materials in order to cope with the local environmental constrains (mainly mechanical and thermal). The Figure 3 shows examples of ideas captured by the designers to illustrate and discuss the various concepts, while the Figure 4 and Figure 5 respectively show the CAD model imagined by the design office and the real device manufactured for the Proof of Concept (PoC) using fast prototyping process.

Figure 3: cylindrical shape concept

Figure 4: 3D CAD model rendering

Figure 5: µMA Proof of Concept (PoC)

Ease daily operations

One of the key drivers was to keep the concept modular to offer the instrumentation engineer all

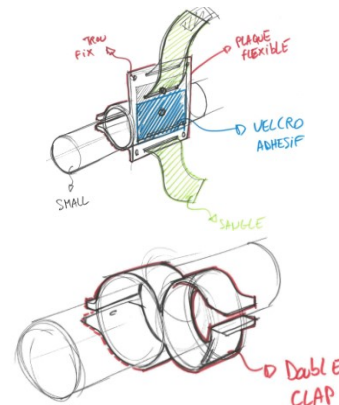the versatility expected to cope with unique needs. In this respect, a "building brick" concept, called "cells" has quickly made a consensus among the contributors. The electronics shall be enough packaged to let the cells being manipulated by the final user without any risk of failure. The end-users promoted the fact that the complete assembly/disassembly of the device shall be performed without the need of any tool in order to ease operations on the field. The statement about internal connector discussed previously motivated the team to try to an innovative way to get rid of traditional internal connectors. In the selected concept the internal connectors are replaced by concentric PCB copper patterns to transfer power supply, data communication and synchronization signals. The efficiency of the electrical connection between the cells is insured by the addition of an interposer with anisotropic conductive properties. This interposer is based on an elastomeric material that insures a certain level of tolerance to mechanical axial or angular misalignment. It also acts as a mechanical decoupling interface between all cells which provides a very good behavior facing high level of vibrations and shocks. The concentric pattern is composed of 8 circles, providing a total axis of freedom in terms of relative angular positioning of two adjacent cells. The position and width of each circle is determined according to the nature of the electric signal to be routed but also takes into account the relative positioning uncertainty and the pitch of the conductive elements of the interposer. For the final user, building his own device should not create any headache: the insertion and extraction of the cells are as easy as opening an aspirin tube to get (or insert) a tablet!
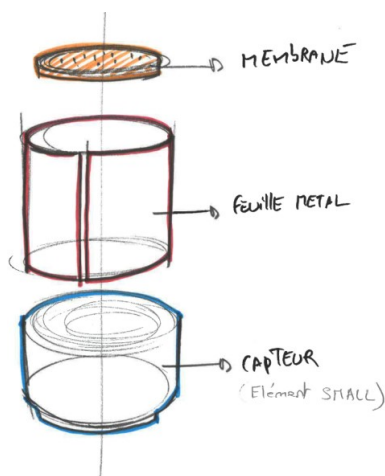
Figure 6: Birth of the cell concept
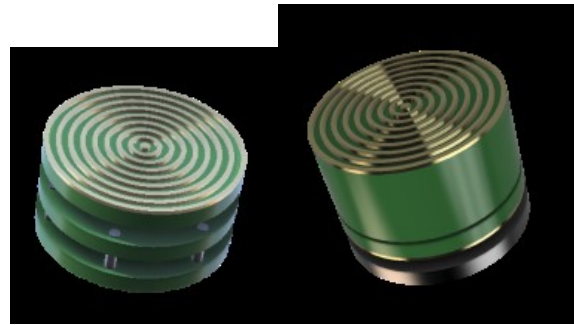
Figure 7: 3D CAD models of the cells

Figure 8: prototype of µMA cell

Minimize intrusiveness

Keeping the device as small as possible was a heavy constrain that everyone kept in mind during all the ideation and design phases. The weight optimization directly inherits from the downsizing effort but also dictated some choices in terms of materials (mechanical housing, potting…). The IIoT trend also shows the way to minimize intrusiveness by paying an accurate attention to the power consumption of every single electronic part involved in the design in order to guarantee full power autonomy with also decent life duration. Wireless and less-wire communication technologies have also been selected at early stages of the concept. At the end, the sweet spot between the downscaling willingness and the kind of features needed to be embedded in the device due to integration constraints, led to an optimal diameter being about 20mm.

From COTS bricks to bespoke solutions

Once the "cell concept" was agreed among the design team, the next step consisted in identifying the main categories of cells that would compose the device. The challenge here was to define the right level of modularity in order to insure that the concept will be versatile enough to cover the wide range of potential applications, while staying simple to use and open to embrace future needs not yet identified. As a result, four main categories of cells have been identified:

Acquisition: these cells host the analog signal conditioning front-end (AFE) and the analog to digital converter (ADC). They are dedicated and therefore optimized for each type of sensors.

Data storage: this optional cell can be added in the device when local storage of the acquired data is required. It is the case when the µMA addresses autonomous data-logger use cases but also when it is inserted in a medium to large scale instrumentation systems, especially wireless based systems, in order to secure or optimize the data transfers. Different kinds of data storage media have been considered, the µSD format, compatible with the overall form factor of the µMA concept has been selected as a first choice for the density and performances offered by the latest versions available on the market. Its wide adoption by the consumer and industrial domain makes the µSD standard a sustainable future proof of choice.

Energy: behind this generic word, various kinds of power sources are anticipated. The first one is obviously the local battery that can be disposable or rechargeable, as small as a button cell, compliant to the standard AA size or even as big as an 18650 form factor if necessary. A local smart power convert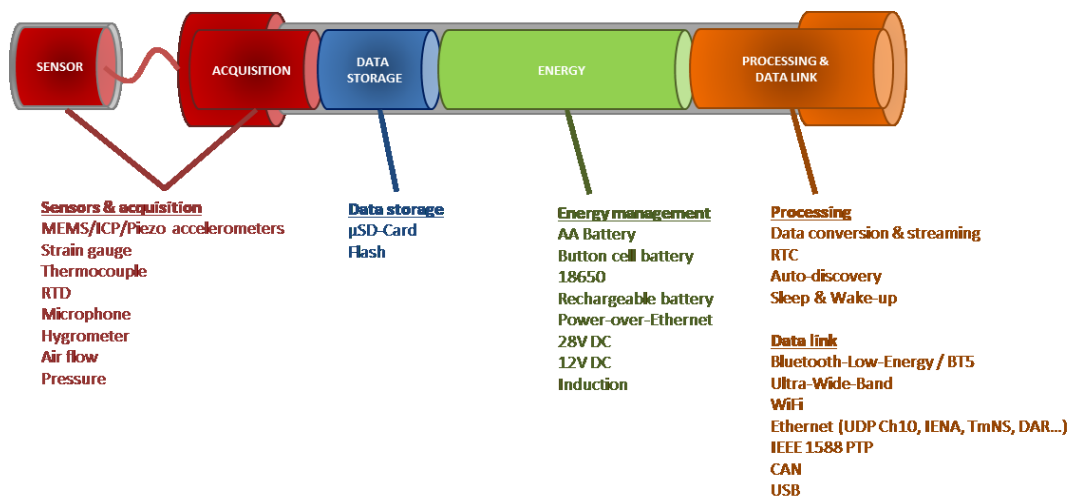er integrated in this cell provides the compatibility between various kinds of battery output voltage and deliver the appropriate constant voltage to the other cells. The power lines between cells

are designed as a bus that distributes the power in both directions and therefore can allow an external source to power the cells and/or recharge the internal battery. In the scope of the less wire approach, Power over Ethernet (PoE) has been identified as a good candidate to merge data communication, synchronization link and power.

Processing & Datalink: initially splitted in separate cells categories, the development of the proof of concept prototypes shown that these two features were intimately linked together. The processing is considered as the brain of the device, coordinating all the features hosted by the different cells so creating the overall function needed by the final user. The communications between the other µMA, the other devices of the FTI network, a mobile device (smartphone, tablet,…) or a PC are managed by the processing and datalink cell. Wired and wireless communications will require dedicated cells as the physical interfaces are specific (RF antenna, Ethernet, CAN,…). Even if the rule is to implement only one processing and datalink cell per µMA, one use case justifies an exception: a wired and a wireless processing and datalink cell can be connected together in order to create a gateway between the wired and the wireless domains (whatever the type of standards: WiFi, BLE, UWB, Ethernet, CAN, USB,…).

Sensors & acquisition
MEMS/ICP/Piezo accelerometers
Strain gauge
Thermocouple
RTD
Microphone
Hygrometer
Air flow
Pressure

Data storage
µSD-Card
Flash

Energy management
AA Battery
Button cell battery
18650
Rechargeable battery
Power-over-Ethernet
28V DC
12V DC
Induction

Processing
Data conversion & streaming
RTC
Auto-discovery
Sleep & Wake-up

Data link
Bluetooth-Low-Energy / BT5
Ultra-Wide-Band
WiFi
Ethernet (UDP Ch10, IENA, TmNS, DAR…)
IEEE 1588 PTP
CAN
USB

Figure 3: µMA cells categories

A fifth category of cell has also been added: **Sensor**. It appears that for some kind of measurements, it was possible to embed the sensing part directly inside a cell, taking benefits of the concept architecture and giving the instrumentation engineer an access to sensors designed to be soldered on a PCB. Ambient air temperature and hygrometry sensors but also MEMS microphones have

been identified among others as good candidates for being directly integrated into a cell. This possibility brings also the need to consider the sensing and acquisition cells to be remotely placed from the main part of the device (hosting the battery, the recording, the processing and datalink cells) in order to ease the installation of the sensing part as close as possible to the area or piece to be instrumented

while digitizing the information as early as possible. In such scenario, the wired connection between the remote part and the main body of the device is composed of the exact same number and type of connection available on the cells themselves.

Figure 9: illustration of the concept of µMA remote sensing cell

Auto-discovery & self-configuration

Compared to other DAUs on the market that cover wide range of features with a limited set of part numbers thanks to a high level of configurability, the µMA concept is based on very specific cells dedicated to a given sensor or function. It is therefore totally possible to leverage the choice made by the final user at the time of assembly to determine the expected behavior. Being able to understand the users' expectations is a way to optimize the setup phase and even to completely bypass it in some cases. This optimization requires to have an accurate knowledge of the various cells connected together using an auto-discovery and identification process at power-up. This analysis role is assigned to Processing and Datalink cell. The user can select the expected behavior to be applied by default (e.g. acquisition of the maximal input range & recording; streaming; sleep mode;…). It is of course possible to apply a legacy configuration process to fine tune every settings when needed.

Network Architecture

From an architectural point of view, it is possible to use the µMA concept independently as a stand-alone device but also connected to others - with or without wires - as light FTI. Extension of an existing FTI (last minute modifications for example) is also a use case.
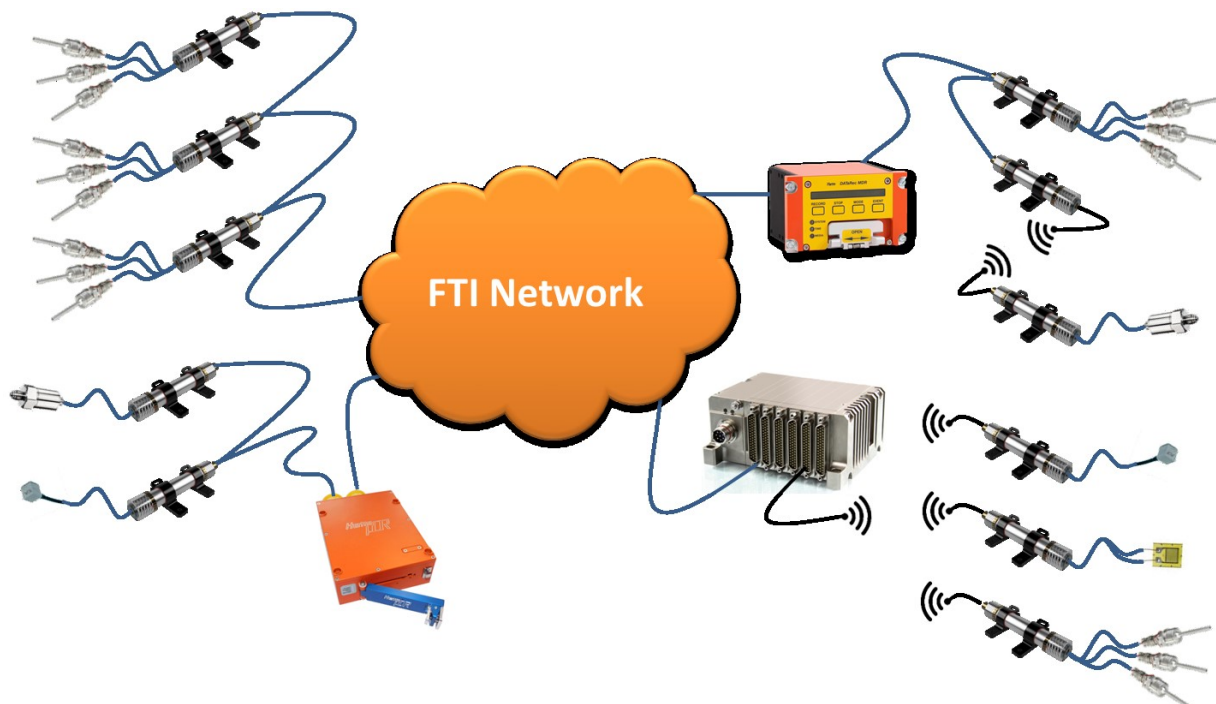
µMA concept assessment

Numerous prototypes have been manufactured in order to evaluate and adapt size, weight, ergonomic aspects, performances, harshness, etc… An agile methodology has been applied to quickly test and validate the different aspects of the concept. Some tests went well on the first run, others failed, and in this case, the design team learned, improved the concept and tried again. The tests started at Safran Data Systems labs, Test & Training Center and then in real aircraft environments.

Among these tests, the behavior while facing high levels of vibrations and shocks of the innovative assembly of the µMA concept has been carefully studied in our Vibration Lab. Various ways of mounting the device (including P-clamps) have also been tested. Thanks to the very light weight of the device and the presence of the elastomeric interface between the cells, all the mechanical test campaign was successful.
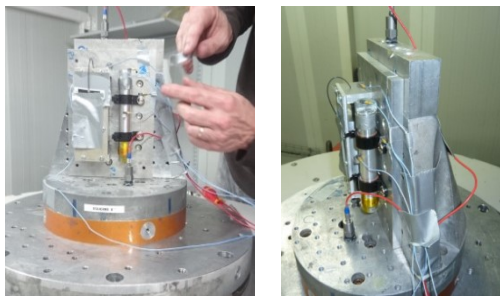
Figure 11: µMA prototype during mechanical test campaign

In terms of Electromagnetic compatibility (EMC) derisking tests have been conducted to evaluate the amount of energy radiated by the device. Thanks to the use of low power components with optimal clock speed the tests were fully passed with radiated levels barely unable to be detected by the measurement tools of the electromagnetic laboratory.

The wireless communication capability of the µMA prototype has been tested with a Bluetooth Low Energy (BLE) implementation. Radio link performance has been tested at various data rates and transmitting power in various environments including in a recent wide body commercial aircraft and also in a twin engine light aircraft as shown on the Figure 12.

All the tests conducted helped to mature the different components of the µMA prototypes and led to the validation of the overall concept.
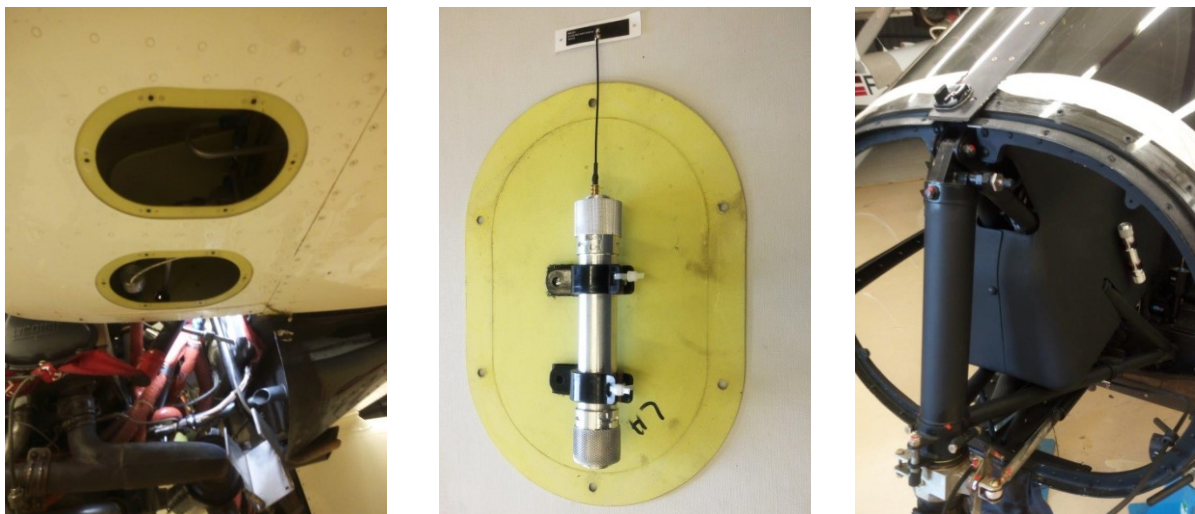
Figure 12: µMA prototypes mounted on a Vulcanair P68 aircraft

Conclusion

Many constraints apply to flight test instrumentation. It must be designed to be integrated into a complex and often tiny structure and to face extreme environments (temperatures, pressures, shocks, vibrations, EMI and EMC, etc.) while targeting to get ever more knowledge about the vehicle. The IoT, which uses the same value information loop as the flight test instrumentation, paves the way to a lot of new technologies (e.g. wireless or low power). At the same time, electronics continue to gain in performance while reducing power requirement and size.

We have shown with this exploration that the flight test instrumentation can become more distributed, less bulky, less heavy, less energy hungry and less wiring consuming, getting closer to the sensor, easier to install and without need of modifications of the articles to be tested, while remaining modular and scalable.

From an architecture point of view, it is possible to use the µMA concept independently but also connected to each other with or without wires,

as a light FTI with a new type of distributed architecture but also as an extension of an existing FTI (last minute modifications for example).

This success is not trivial. It has required capitalizing on decades of experience in innovation and development of instrumentation systems, imagine new solutions and usages with instrumentation players, thinking of systems and architectures in a new way, to develop a new modular approach of 3D electronics, and to investigate low-power, wireless and energy technologies from the world of IoT and adapt them to the environment of test instrumentation on aircrafts, helicopters, armaments, launch vehicles, naval vehicles, automotive or industrial systems.

Our investigations, our tests in laboratory and in real environments, demonstrate that this new approach of the desing (3D modular electronics, wireless, low consumption, distributed architecture, reduced dimensions, compliance to harsh environmental conditions,...) and usage (instrumentation as close as possible to the sensor, no modification of the structures, high speed deployment, immediate reaction capacity, live troubleshooting, auto-configuration, plug and play, rapid reconfiguration,...) creates a revolution in instrumentation, testing and certification activities for the transportation vehicles of today and tomorrow. This survey is only the very promising start of an adventure towards new use-cases still to be invented for this type of modular platforms.

The overall µMA concept presented in this document is patented and remains the full property of Safran Data Systems.

Acronyms

ADC: Analog to Digital Converter

AFE: Analog Front End

BLE: Bluetooth Low Energy

CAD: Computer Aided Design

CAN: Controller Area Network

COTS: Commercial Off The Shelf

DAU: Digital Acquisition Unit

FTI: Flight Test Instrumentation

HD: High Density

IOT: Internet Of Things

IIOT: Industrial IOT

MEMS: Micro Electro-Mechanical System

PC: Personal Computer

PCB: Printed Circuit Board

PoC: Proof of Concept

PoE: Power over Ethernet

RF: Radio Frequency

SIP: System In Package

SWAP: Size Weight And Power

UAV: Unmanned Aerial Vehicle

USB: Universal Serial Bus

UWB: Ultra Wide Band

µSD: micro Secure Digital

All credits: Safran Data Systems

References

[1] X. Delarue, *" Network telemetry: a new digital acquisition system for Airbus A380"*, ITC 2004 proceedings, p645-654

[2] G. Garnier, F. Percie du Sert, *"Implementing wireless solutions for data transmission and synchronization in the aerospace FTI constraints"*, ETTC 2019

[3] M. E. Raynor, M. J. Cotteleer, *"The more things change: Value creation, value capture, and the Internet of Things"* in Deloitte Review, issue 17, 2015.

[4] Multi CB web site as of 05/05/2020 www.multi-circuit-boards.eu/en/products/printed-circuit-boards/rigid-flex-pcb.html

[5] 3D plus Website System In Package https://www.3d-plus.com/product.php?fam=1

A standard proposal for a sensor schema. SensorDefML.

José Antonio González Pastrana, Sergio Sanchez-Montes, Roberto Paredes,
Airbus Defence and Space, Getafe-Spain.
Jose.o.gonzalez@airbus.com, Sergio.m.montes.external@airbus.com,
Roberto.paredes.external@airbus.com

Abstract:

As in other areas of Instrumentation, xml files are used for defining the characteristics of different elements, in this case we are going to focus on instrumented sensors which are not already covered by existing specific standards such as XidML or MDL files. Our intention is a new proposal that could drive the industry to create a new standard that can help with some instrumentation tasks related with these elements, specially software configuration and global configuration management. This would be alike a sensors datasheet but in a xml file.

Key words: MDL [1] , XML [2], Open Standards, XidML [3], XdefML[4].

Introduction

Revising the existing schemas which could represent a sensor we can find one stablished schema called SensorML, but it is used in the same way as XidML or MDL, because it understands a sensor as a system to distribute data instead of representing its internal features and configuration options, but what we want to propose it is to represent the datasheet of a basic sensor like accelerometers, gauges, etc.

Similarly than standards for transducers, known as TEDS (Transducer Electronic Data Sheet), which are focussed just in hardware settings, we need a standard for covering software settings and configuration.

The main objective is to involve sensor manufacturers so maybe they could provide in the future this SensorDefML files the same way they supply their datasheets.

What fields should show SensorDefML?

SensorDefML is a metamodel that allow us to define the characteristics of a sensor. So if we have a magnitude that we want to measure, this it will be our input. We also can have additional inputs to take into account. Generally, the expected output should be a voltage or a current, but could be any other magnitude. This demonstrates that the schema has to be flexible enough to cope all the variety we find in the market. Meanwhile a minimum of characteristics have to be taken into account.:

- Identification.
  - Manufacturer.
  - Model Identification.
  - Part Number.
  - Ordering notes.
  - Local supplier address.
  - Operating principle.
  - Application field.

- Mechanical Characteristics:
  - Weight.
  - Dimensions.
  - Mounting instructions.
  - Drawing files.

- Input/Output (Channel for)
  - Magnitude (Temperature, Pressure, Voltage, Current, etc.), Power
  - Units.
  - Range / Span.
  - Impedance.
  - Bandwidth.

- Grounding.

- Noise.

- Metrologic characteristics:

  - Uncertainty.

  - Accuracy.(Class).

  - Dead zone / band.

  - Sensitivity.

  - Cross channel sensitivity.

  - Repeatability.

  - Hysteresis.

  - Bias.

  - Scale type.

  - Damping

  - Calibration formula.

  - Formula parameters.

  - Errors (sources of error).

- Operating conditions

  - Temperature range.

  - Storage temperature.

  - Pressure. Static/Dynamic.

  - Humidity range.

  - Warm-up time.

  - Vibration range.

  - Maximum acceleration.

  - Impact acceleration.

  - MACH number limit.

  - Type of standard.

  - Accomplished Standards.

All of these characteristics are examples of what has to be taken into account to help us to create a first version of the schema. Obviously a lot of different characteristics can appear later on as more sensor classes start adopting the SensorDefML. For that reason SensorDefML has to be as much flexible and scalable as possible, not only for supporting new and different sensors but also new features or settings without changing the defined internal file structure.

Working with SensorDefML SDK

In order to have an agile way of working and also prevent recurrent errors it is helpful to have a SDK[5] making easier some commonly used methods such as create, read and write SensorDefML files. This SDK should have functions to extract all necessary information in order to fill-in data in the memory data models and/or in databases that the customer could need.

Tools based in SensorDefML

Two tools are planned to use this schema.

One of them is an Inventory Control System (ICS) which manages all the hardware related elements belonging to a system. The other one is a Systems Designer (SD), a tool for designing the complete architecture of a system, taking into account all the different manufacturers and full integrated with the ICS where the hardware items are available to be placed in an architecture.

The ICS needs to persist new sensors, so the best way to do this task is by importing a SensorDefML file which contains all the necessary information to fulfill the tool's requirements, such as quality politics and others.

The SD can take advantage of SensorDefML files to configure the complete measuring chain, check specified ranges and ensure that every sensor fits in the system. The calibration of the sensor can be added afterwards, but most of configuration will be valid, so a great deal of work can be done from the beginning. Also, as the SD is connected to the ICS database you have available not only the sensor models that you can use, but you can have the real serial number you will use from the beginning.

Conclusions

SensorDefML files intends to be a key technology for the ICS, SD and metrology in ADS [6]. This kind of schema could become a standard in the future. As a standard, their definition should be agreed between all stake holders in this field. So all of them are invited to join this initiative.

References

[1]   MDL -http://www.irig106.org/docs/106-
       17/Chapter23.pdf

[2] XML - https://www.w3.org/XML

[3] XidML - http://www.xidml.org

[4] XdefML - http://www.xidml.org

[5] SDK – Software Development Kit

[6] ADS – Airbus Defense and Space

Addressing the Babel's Tower of FTI Standards in a Network Environment

Patrick Quinn

Aerospace Instrumentation
Curtiss-Wright
Unit 5, Richview Office Park, Clonskeagh, Dublin 14, Ireland
pquinn@curtisswright.com

Abstract: This paper discusses the different demands the many industry wide standards place on flight test instrumentation hardware and software in a networked environment and the challenges of supporting all standards from a supplier point of view.

Keywords: Paper format, instruction to authors

1. Introduction

Today's networked flight test instrumentation (FTI) hardware, and supporting software, needs to be a Babel Fish, the universal translation device from Hitchhiker's Guide to the Galaxy, to support the multitude of industry standards.

From Chapter 10, TmNS, iNET-X, IENA and DARv3 transmission protocols to TMATs, MDL, XidML and XML metadata, both the hardware and software are required to speak and understand multiple packet types and file formats.

Each of these formats present their own challenges and have their own advantages and shortcomings, and each present different challenges in a distributed networked architecture.

In an ideal world, connecting networked FTI systems would be a simple – just "plug and play". However, experienced users will tell you that this is just not the case.

This paper discusses some of the challenges imposed on FTI, both at the hardware and software levels, by these various standards and highlights how these may be addressed.

2. Chapter 10 & TMATS

IRIG-106-Ch10 is probably the oldest standard of all the ones discussed in this paper, and probably the one that imposes the largest demands on a modern networked FTI system.

The standard itself was originally a solid state recording standard that evolved from the move away from tape based recorders in the late 1990's. It has evolved over the years to add more data types and time formats.

The Chapter 10 standard is inherently a recording standard. It is not network centric, it does not define how any of the Ethernet protocols, like TCP/UDP/QOS etc., should be leveraged to optimize the FTI network.

It defines a specific data structure which the data must be recorded. The structure of the Chapter 10 recording file is presented in Figure 1.
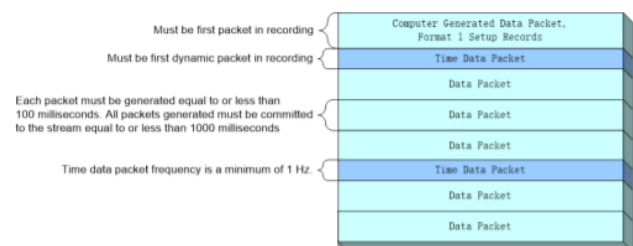

Figure 1: The Chapter 10 format

The first packet in the recording is the Computer Generated Data packet that describes all the subsequent data packets in the recording. This is usually the TMAT's file.

2.1 TMATS

In a networked distributed architecture there are two approaches to using chapter 10 as your data standard:

1. Record all data acquisition units (DAU) to a single system wide chapter 10 recording
2. Record all DAUs to individual chapter 10 files per DAU

For the TMATS file the challenge here, when using approach 1, is primarily a software one. The FTI configuration software must be capable of describing all the data captured by all DAUs into a single coherent TMATS file that describes all packets in the system.

When using approach 2, this is simpler, but the configuration software must generate a TMATS file per DAU and keep them all up to date as the configuration changes.

2.2 Time Packets:

The real challenge at the hardware level when using Chapter 10 in a distributed networked architecture is the time packet.

The first dynamic packet in the recording must be a time packet, this time packet must per periodic and repeat at a minimum of 1 Hz.

Chapter 10 defines 3 formats of time packet, Format 0 is reserved for future use, Format 1 covers time from GPS / relative time counter and Format 2 covers network time.

Format 1 contains a channel specific data word (CSDW) that indicates the IRIG time source, covering everything from free running to PTP locked to external embedded time sources. The resolution of the time stamp in Format 1 is down to milliseconds.

Format 2 also contains a CSDW that indicates the network time format (NTP, PTPV1 or PTPV2) and the validity of the network time. It has a time resolution down to nanoseconds. Format 2 would seem to be the ideal time format to use in networked systems.

However, there is one major missing piece of the puzzle that needs to be addressed for chapter 10 to be truly useable in a distributed networked architecture, this is the question of "Who creates the time packet?" and "How does this time packet keep track the time from all the other DAUs?".

With multiple DAUs, each DAU can produce its own time packet. When these time packets arrive to the recorder the recorder must decide which time packet to use to time stamp the full system recording. There are multiple possible solutions to this:

1. The system could be configured such that only one DAU actually produces a time packet and the recorder uses this time packet for its time source.
   - This has the advantage of being a simple approach, but it does not reflect the status of the data from the other DAUs relative to the time master DAU
   - What happens if Time Master DAU goes out of PTP lock?
   - What happens if Time Master DAU time packet is dropped or delayed?

2. The recorder could produce its own time packet
   - Again a simple approach, but with the same issues

3. The recorder could track the time packets from all DAUs and make a decision on which is the best.
   - Difficult to implement

4. The recorder could record each DAU to its own individual chapter 10 recording
   - Not an efficient use of recording bandwidth

5. The time packet requirement could be dropped by the standard
   - The secondary header time could be used in post flight to align the data

2.3 Chapter 10 Data Types

Another feature that chapter 10 places on FTI hardware, and software, are the excess of different data types and different sub formats of each data type that must be supported for a true Chapter 10 compliant FTI set up. As of the IRIG-106-2019 release, the data types and versions

of each are supported by the chapter 10 standard as noted in Table 1.

Data Type	Formats Defined
Computer Generated Data	Format 0, 1, 2, 3 & 4
PCM Data	Format 0, 1 & 2
Time Data	Format 0, 1 & 2
MIL-STD-1553	Format 0, 1 & 2
Analog Data	Format 0 & 1
Discrete Data	Format 0 & 1
Message Data	Format 0
ARINC-429 Data	Format 0
Video Data	Format 0, 1, 2, 3 & 4
Image Data	Format 0, 1 & 2
UART Data	Format 0
IEEE 1394	Format 0 & 1
Parallel Data	Format 0
Ethernet Data	Format 0 & 1
TSPI/CTS Data	Format 0, 1 & 2
CANBUS Data	Format 0
Fibre channel Data	Format 0 & 1

Table 1: **Chapter 10 data types**

Each of the above formats Chapter 10 also defines a UDP Transfer Header, which has 3 different formats,

- Format 1: Little Endian, 24 bit UDP Sequence number
- Format 2: Big Endian, 24 bit UDP sequence number
- Format 3: Little Endian, 16-32 bit Datagram Sequence number.

Each data type has their own sub set of rules around how the data is packed by the FTI hardware. For example, there are 6 different ways of recording PCM data just in PCM Data format 1 alone.

4. IENA

IENA is the Airbus network packet protocol that originated during the A380 program and has been widely adopted industry wide since then. Originally conceived as a network standard, the IENA protocol is quite simple and clear to understand, and support, from a networked architecture point of view.

Some of the restrictions IENA places on FTI hardware include

- Destination MAC must be Multicast, in the range 01:00:5E:01:01:00 - 01:00:5E:01:01:FF
- Packet fragmentation is allowed.
- Source IP must be in the format 172.28.X.X
- Destination IP must be in the format 235.1.1.X
- Source port must be greater than 50000
- Destination port must equal 51000.
- IENA time stamp is the number µS since the start of the current year.
- There are two status sections of the IENA header:

- Key Status – fixed for any IENA Key
- N2 Status – dynamic for any instance
- The sequence number is 16 bits
- There is an END word at the end of each packet, must be the same for all packets in any configuration

4.1 IENA Packet Types:

IENA defines 5 parameter types, and it is forbidden to use different parameter types in different IENA Keys.

4.1.1 P Type – Positional Parameters

Multiple occurrences of P Type parameters can be placed in one packet, but must follow a repeating pattern, as shown in Figure 1.
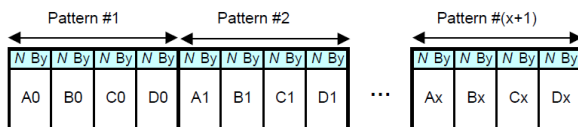

Figure 2: P Type parameters must follow a repeating pattern

One of the restrictions this places on any decom software is that it must know exactly how many parameters are in each packet and the occurrences of each in order to be able to locate all samples of any individual parameter.

4.1.2 D Type – Standard parameters with a delay field

These are groups of a maximum of seven 16 bit data words with an assigned parameter id and a 2 byte delay field. The parameters must be placed in a particular order and the delay field is the delay in µS from the packet time stamp to the acquisition time.
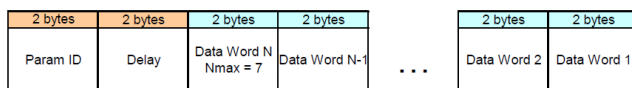

Figure 3: IENA D Type

4.1.3 N Type – Standard parameters without a delay field

Same as above, but without the elapsed time field.
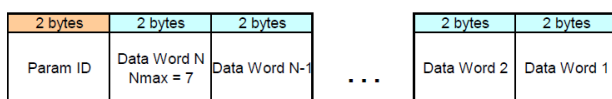

Figure 4: IENA N Type

4.1.4 M Type – Message parameters with a delay field

These are Message parameters whose length and acquisition time of the entire message can be reflected in the IENA packet. The data set can be padded if required and the delay field is the delay in µS from the Packet time stamp to the acquisition time.
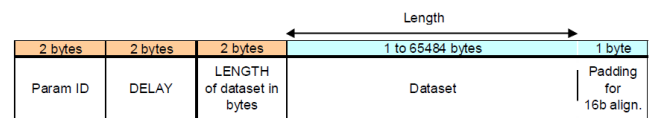

Figure 5: IENA M Type

4.1.5 Q Type – Message parameters without a delay field
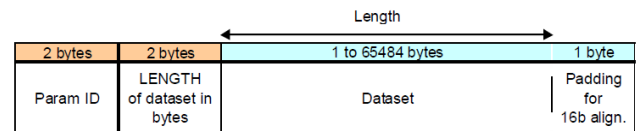
Same as above, without the delay field.

Figure 6: IENA Q Type

4.2 IENA shortcomings:

While IENA is popular there are a couple of shortcomings in the standard that do not take full advantage of todays networked systems capabilities:

1. Time stamp is in µS, in both the packet time stamp and the delay fields
   Greater time resolution is possible with other formats, such as IEEE 1588 PTP which allows time resolutions of nanoseconds to be reflected in the data
2. Parameter placement restrictions do not allow multiple samples of the same parameter to be placed contiguously

5. iNET-X

iNET-X is a network packet protocol developed by Curtiss-Wright's Dublin Office (formerly Acra Control). Again, originally conceived as a network standard, the iNET-X protocol evolved out of the early iNET (TmNS) definition and is also simple and clear to understand and support from a networked architecture point of view.

iNET-X does not support quality of service (QoS) protocols like pause frames, it does not allow packet fragmentation and does not support IGMP or DHCP. All packets are UDP packets, so TCP traffic is not supported. It does support aperiodic traffic transmission and SNMP. File transfer protocols like TFTP are supported, but not FTP.

From a network architecture point of view, iNET-X expects minimal dynamic behaviour from the sources and hardwired traffic routing. This requires the configuration software to be able to map the traffic flow throughout the entire FTI installation.

From a recording point of view iNET-X prefers open standards like PCAP and FAT32 file systems.

Unlike IENA, iNET-X allows:

- Both multicast and unicast destination MACs
- There are no restrictions on source IP, source port, destination IP and destination port

- The iNET-X time stamp takes full advantage of the nanosecond resolution offered by PTP-1588, with 64 bit time stamps counting from 1/1/1970
- There is a 32 bit flags field in the iNET-X header, which is used to dynamically reflect status of the data in the packets
- The sequence number is 32 bits

5.1 iNET-X Packet Types

iNET-X defines 4 packet types: placed, bit-aligned, block aligned, parser aligned and event.

5.1.1 iNET-X placed

This uses fixed, constant length packets, not exceeding 1426 bytes of payload. They must end on a 16 bit boundary. Multiple occurrences of any type of parameters can be placed in one packet, and multiple occurrences of parameters are placed contiguously.
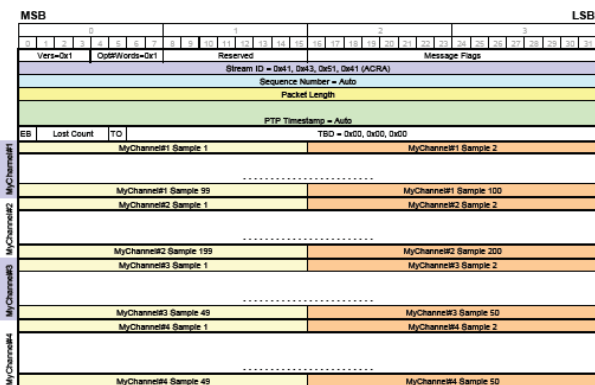

Figure 7: The iNET-X placed packet type

5.1.2 iNET-X bit-aligned

They have variable length packets, used for CVSD audio packets. Max payload length of Nx4 bytes, not exceeding 1426 bytes.
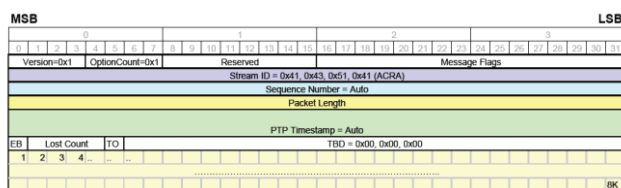

Figure 8: The iNET-X bit-aligned packet type

5.1.3 INET-X block-aligned

These are variable length packets, used for video transport streams. Usually constructed of X blocks per packet, with fixed number of bytes per block.
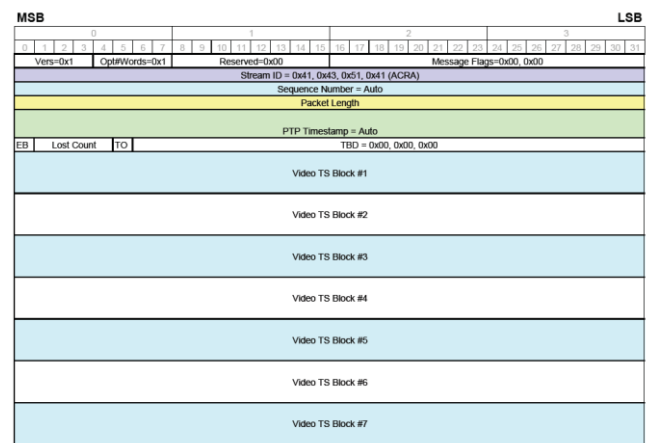
Figure 9: The iNET-X block-aligned packet type

5.1.4 iNET-X Parser-Aligned

These are variable length packets, used for bulk bus data capture. Each parser aligned block is a complete bus message, with elapsed time timestamp relative to the first message in the packet.
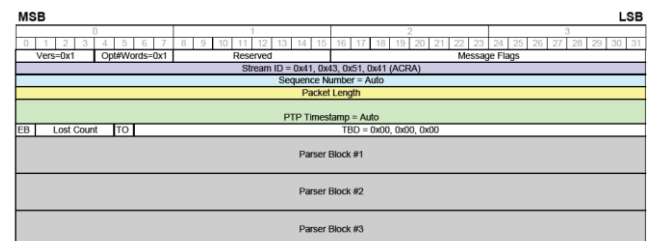
Figure 10: The iNET-X parser aligned packet type

5.1.5 iNET-X event

These are variable length packets, used for event marking. Usually made of a string representation of event code and a time stamp of the event.
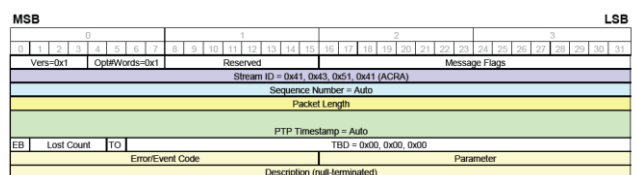
Figure 11: The iNET-X event packet type

6. DARv3

DARv3 evolved out of the DAR standard, originally developed by Boeing and the South-Western Research Institute (SWRI) for the Boeing 787. It was adopted by Curtiss-Wright's Newtown business unit (formerly TTC) and expanded upon to create DARv3. Once again, this protocol started life as a network protocol and is also simple to understand and support from a distributed networked environment.

DARv3 does support QoS protocols like pause frames, it does allow packet fragmentation, it does support IGMP but not DHCP. All packets by default are UDP packets, but TCP traffic is allowed. It does support periodic traffic

transmission and SNMP. File transfer protocols like TFTP and FTP are supported.

From a Network architecture point of view, DARv3 expects some dynamic behaviour from the sources, they must be able to respond to pause frames, and must advertise that fact to downstream devices during link negotiation. This requires the configuration software to be able configure a maximum latency for the packets to manage the traffic flow throughout the entire FTI installation.

DARv3 uses multicast packets and devices that support it use IGMP to join or leave multicast groups. Multicast packets use a default destination port of 50001, but this is configurable.

DARv3 supports fragmentation of jumbo frames, with fragmentation driven by message latency settings. DARv3 is big-endian. DARv3 also defines a network recording standard, the direct structure of which is compatible with the standard used for IRIG-106 Chapter 10, STANAG-4575 Edition 1.

The DARv3 packet format is made up of the DARv3 header followed by 1 or more data segments as shown in Figure 12.
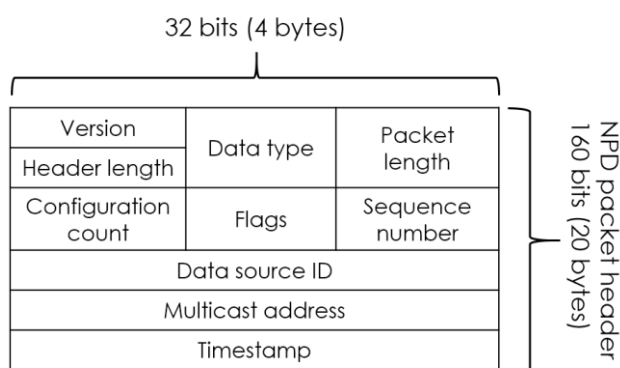

Figure 12: DARv3 packet

Unlike iNET-X, the time stamp, while based on IEEE-1588, is only the 32 bit seconds portion of the full 1588 time. The following data segments also follow a common structure and are all 32 bit aligned.
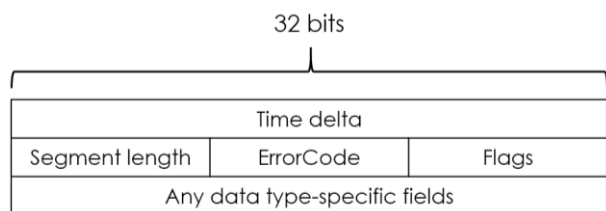

Figure 13: DARv3 time stamp

Here, the time delta is the nanosecond portion of the full 1588 time stamp, relative to the time stamp at the start of the packet.

6.1 DARv3 Packet Types

Similar to Chapter 10, DARv3 has multiple packet types, dependant on the type of data being captured. The data

type being used is indicated in the header of the packet. It is therefore forbidden to mix data types in the same packet. As of the 2016, DARv3 standard the packet types in Table 2 are defined, each with its own set of rules around how that traffic type must be encapsulated.

Name	Protocol
1553	MIL-STD-1553
A429	ARINC 429
ACQ	Analog / digital signal acquisition
A664	ARINC-664
AUD	Audio
ETH	100/1000BT Ethernet
Event	Event packets
FAAD	Fwd Area Air Defence Standard
FC	Fibre Channel
HFCI	IMB protocol
HRV	High Res Jpeg-2000 Video
IP	IP Packets
Link-16	TDL 16
LRV	Low Res Jpeg-2000 Video
MP2	MPEG-2 Video
NDO	Arinc-664 Network Data Objects
PCM	PCM to IP gateway
Raw Bayer	High speed camera images
RDR	Radar
RGB	RGB video
Status	Status packets
TCP	TCP packets
UART	Serial Data
UDP	UDP Packets

Table 2: DARv3's packet types

7. TmNS

The Telemetry Network Standards Definition (TmNS) is the newest of all the standards and one of the most comprehensive in its attempt to define all aspects of the networked environment for FTI. It comprises of eight chapters of the IRIG 106 standard, covering everything from message formats to system configuration and management to RF networks.

TmNS does offer some new capabilities to the FTI environment that are really interesting:

1. Bi directional telemetry
2. PCM back fill
3. Dynamic spectrum sharing
4. Quality of service
5. Fully interconnected systems
6. Over the horizon telemetry

TmNS device support is truly comprehensive in that it covers all of the following:

- 100BT, 100BF, 1000BT, 1000B-SX, 1000B-LX, 10GBT, 10GB-SR, 10GB-LR, 10GB-ER

- Auto negotiation, MAC layer packets, IP protocols, LLC protocols, Spanning Tree protocols, Quality of Service protocols, ARP, IPV6, IPv4, ICMP, IGMP,

TCP, UDP, IP Security protocols like TLS & SSL, both DHCP & Static Ip Address assignment

- TnMS only allows PTP-V2, master, slave & boundary clocks

- 1 PPS outputs and GPS time support

- Encryption

- SNMP – there is a TmNS management information base (MIB)

This does not imply that all devices must support all of the above, but compared to the other standards it includes far more of the possibilities networked systems have to offer.

TmNS attempts to keep the message format simple by trying to define a single message format to describe all traffic types.

The message definition consists of a message header followed by a message payload. The message payload consist of a series of TmNS packages.
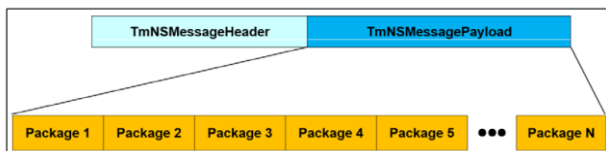

Figure 14: TmNS message format

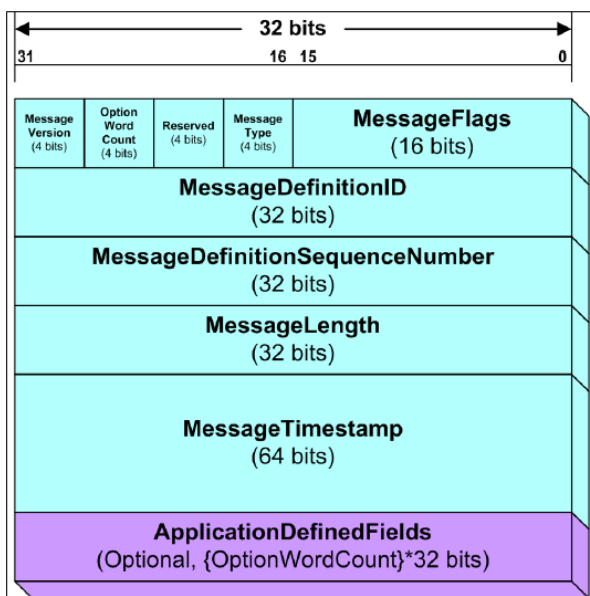The TmNS message header consists of message fields very similar to iNET-X.

Figure 15 TmNS message header

Each payload package consist of a package header and package payload and must be 32 bit aligned.
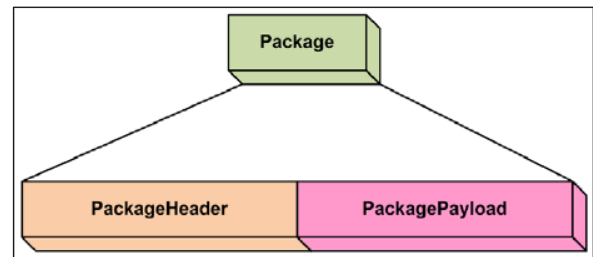
Figure 16: TmNS package

The Package Header contains the fields shown in Figure 17.
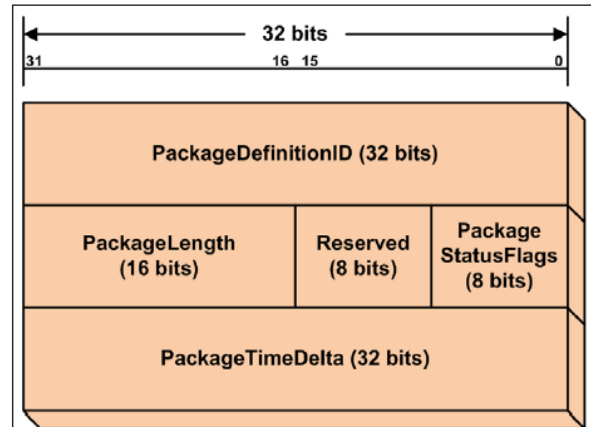
Figure 17: The fields within the package header

While the attempt to use a single package definition is admirable it runs the risk of not being able to describe all data field for all data buses or analog signals inside these restrictions for every vendor. This reduces the possibility of interoperability between multiple vendors.

IRIG-106 chapter 25 requires that TmNS devices support configuration over SNMP, with the use of various public MIBs and a TmNS specific MIB. Again, here it is not clear that all vendor device capabilities can be configured via these MIBs.

TmNS is evolving over time and as adoption increases in the customer base these issues will no doubt be addressed, however, achieving a truly vendor neutral standard via TmNS remains a long way off.

8. Becoming the Babel Fish

From a customer point of view, the benefits of 'plug-and-play' hardware are fairly obvious. The ability to pick and choose between vendors without worrying about integration issues or needing to alter IT infrastructure, has many advantages.

From a vendor point of view, Curtiss-Wright will as always be all things the all people. We will continue to support IENA and INET-X out of Das Studio and DARv3 out of TTCWare. For Chapter 10 our Axon family DAUs will support all time format packets, until the time packet issues in distributed networked systems is addressed by the RCC, all UDP data transfer formats and all data formats, configurable by the user. Both TTCWare and Das Studio will be able to produce TMAT's files at both

the individual DAU level and at the full system level to support both the single recording per system and single recording per DAU approach.

For TmNS Curtiss-Wright will build in the ability to generate MDL files readable by both Das Studio and TTCWare, but will continue to use our existing configuration software to program the hardware until the MDL schema and the TmNS MIB supports a way to configure all of our hardware settings.

How conventional Flight-Test Instrumentation concepts can contribute to the innovative eVTOL aerospace development

Luis Antonio Jacob da Motta
Lilium GmbH, Weßling - Germany
luis.motta@lilium.com

Abstract:

Even in face of all innovative aspects around VTOL aircraft development, specially for all-electric airplanes, the conventional FTI concepts shows up your real value: from Processes Organization until Test Results, FTI can contribute providing solution in many scopes of the Flight-Testing, what includes Requirements Control, Data Acquisition and Special Systems Design, Flight-Test Data Processing and Analysis and Flight Results Reports. In this article, we'll describe how the conventional FTI is being developed and implemented from the scratch in an eVTOL Startup, its development challenges and how to keep improving the instrumentation concepts in front of all new technologies demand for manned and unmanned flights.

Key words: Data Acquisition, Flight-Testing, VTOL, Data Processing, Startup.

Introduction

Keeping in mind we are planning to validate something complete new and challenger, and doesn't matter what you are planning to test in the air, the first and mainly concern we must have "on the table" when designing is SAFETY. And having it as our primary requirement in the new project, now we can focus in everything we must comply with all engineering requirements, combining them between each other, providing us coverage around the second most import aspect in aviation: RELIABILITY and CONTROLABILITY.

In that way the Flight-Test Instrumentation (FTI) has and important involvement and contributes providing all relevant measurements and monitoring systems to the Systems Engineering and Flight-Test Crew during performing flight-tests. The idea is to have the clear aircraft's behavior when performing the tests (Real-Time Data Processing), minimizing risks and maximizing performance, and the possibility to analyze deeply all data and flight conditions after tests (Post-Flight Data Processing).

Even having an innovative scenario in this all-electric aircraft development, the conventional FTI shows how important, reliable and useful it is during the Flight-Test Campaigns (FTC), of course respecting and complying with all electrical related aspects like Electrical Magnetic Interference (EMI) and Unmanned Flight Operation characteristics and restrictions.

FTI Macro Process

Represents a group of best practices sequence of development phases that the instrumentation is being performed at Lilium for a better and faster implementation:

- *Conception:*
  Consists of phase where all preliminary FTI definitions are evaluated;

- *Project:*
  Consists of phase where the FTI System is fully documented, which includes electrical and mechanical installation drawings;

- *Integration:*
  Consists of phase where the proposed FTI System, sensors calibration and aircraft system integration is performed;

- *Software Development:*
  Consists of phase where all necessary software (SW) development for controlling FTI information and Data Processing is performed (continuous development), even along the FTC's;

- *Operation*:
  Consists of phase where the FTI System itself is operated and maintained in the aircraft during the FTC's.

[Conception] FTI Measurements and Special Systems definition

The Flight-Test Instrumentation List (FTIL) is a group of measurements about what the System Engineering is planning to monitor every test. Aspects like numbering and naming standards, type and range of measurement, sampling rate and data-sourcing will be previously defined in the list. Currently Lilium Jet FTIL contains around 10K measurements created (April, 20).

Fig. 1.   Lilium FTIL sample.

Fig. 2.   Lilium Jet FTI Datasources mapping.

As FTI Special Systems we can consider a mandatory group of systems or equipment to perform the tests accordingly like Data Acquisition System (DAS), Radio-Frequency (RF) Telemetry System (Data + Image), Ground Control System (GCS), Pilot and Engineering Displays, etc.. – which provides to the FT Crew and Engineering monitoring and controllability conditions during the tests.

[Conception] FTI Architecture definition

Combining all measurements, special systems and operation requirements we can define the FTI System Architecture, as you can see in the figure below:
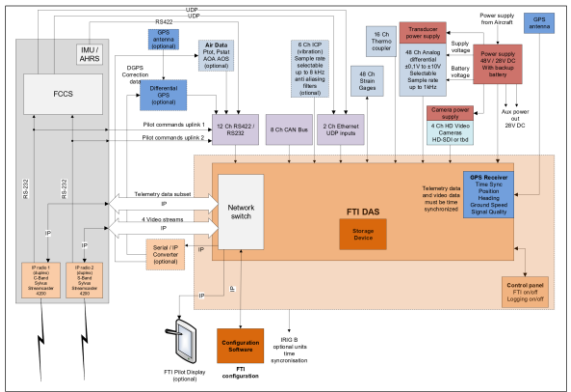
Fig. 3.   Lilium FTI Airborne System Architecture.
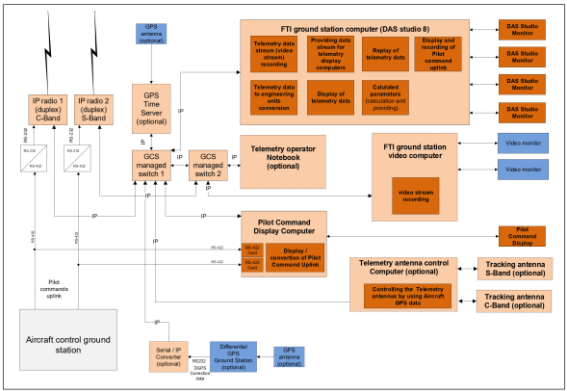
Fig. 4.   Lilium FTI GCS Architecture.

[Project] FTI System installation design

The FTI System installation design includes all electrical and mechanical installation drawings. In special, the FTI Rack installation design is shown below. Inside it, we'll install the DAS, Video and Radio Telemetry Systems, as many other important FTI components:
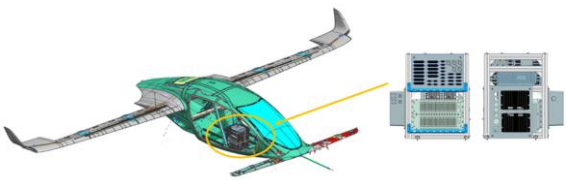
Fig. 5.   Lilium Jet FTI Rack mechanical installation design.

[Project] FTI Sensors installation design

In the same way and standard, all sensors and signal sources are identified in the installation drawings, as you can see below:
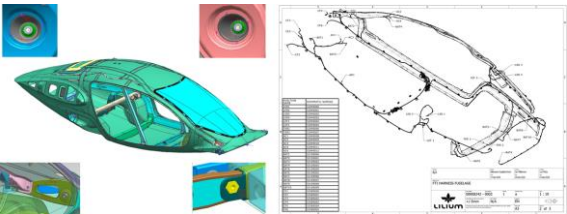
Fig. 6.   Lilium FTI harnesses installation design.
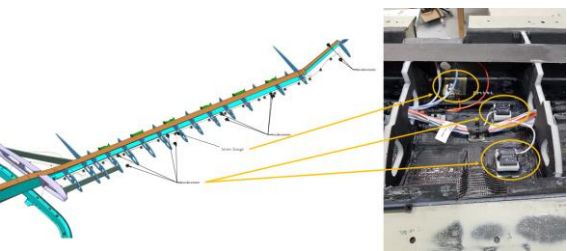
Fig. 7.   Lilium FTI sensors installation design.

As part of the FTI sensors documentation, is also mandatory to prepare the FTI wiring diagrams, where all electrical connections between DAS and Sensors are described, as you can see in the figure below:
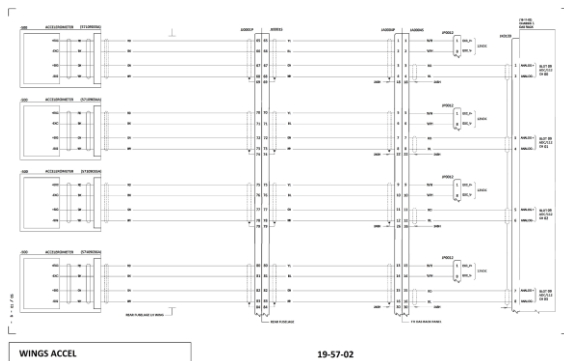
Fig. 8.   Lilium FTI sensor wiring diagram example.

[Integration] FTI Integration, Bench Tests and Sensors Calibration

All Lilium Jet FTI Integration has been performed in the Software/Hardware in the Loop Laboratory (SiL and HiL), simulating flight conditions and all signal types from different data sources into the FTI DAS. In parallel, all Ground Control logical functions from the aircraft side could be verified from the Pilot Control, also available in the room, as you can see in the figure below:

Fig. 9.   Lilium FTI SiL and HiL integration tests.

About FTI Calibration process, we are following and complying with ISO-17025 requirements.

[Integration] Ground Control Station design

The GCS consists in a remote station for the aircraft monitoring (and control, for unmanned flights). From there, Flight-Test Pilots and Engineers can perform different scopes of testing and evaluate the aircraft maneuverability and controllability aspects.

When performing flight-tests, the FTI data is fully recorded inside the aircraft and transmitted from the aircraft to the GCS via RF Telemetry Link; Pilot commands - from the Pilot Control

System (PIS) - are also transmitted via RF link, but in the way-around, from the GCS to the aircraft.

Fig. 10.   Lilium GCS design and preparation.

[Software Development] Lilium Flight-Test Portal development

All information regarding to the FT Process, including FTI, is controlled by the in-house developed FT Portal. It's a web-based application where FTIL, Reports, Test Plans and Results are controlled and registered, and can be accessed by anyone in the company via web browser.

The user-interface is friendly and the system combines web applications with desktop (script-oriented ones) running "behind the scenes", from the user point of view.
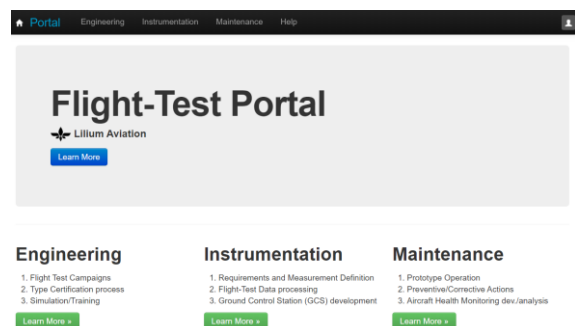
Fig. 11.   Lilium FT Portal main page.

All FT Portal features are connected to the Flight-Test Database (FTDB), developed in-house, as you can see in the figure below.
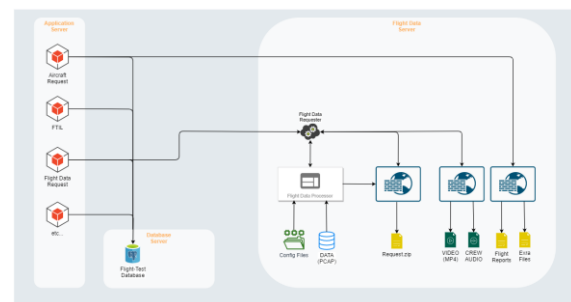
Fig. 12. Lilium FT Portal Architecture.

[Software Development] Flight-Test Results

The Flight Test Results is a group of features that, combined, can provide all information around any test performed by Flight-Test Team to the systems engineering, which includes FT Reports and FTI processed data.
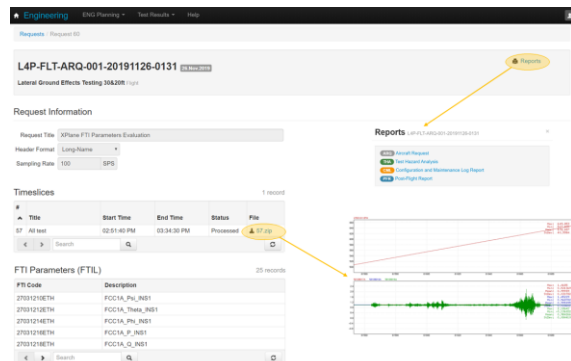
Fig. 13. Lilium FT Results user interface.

[Software Development] Pilot and FTE's Displays development

FT Displays are developed and customized in-house for each FT Campaign, and they are available in the GCS to the Pilot and FTE's every test. Once flying unmanned, they are mandatory.
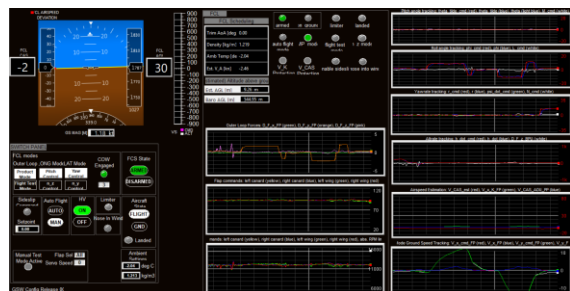
Fig. 14. Lilium Pilot and FTE's Displays (example 1).
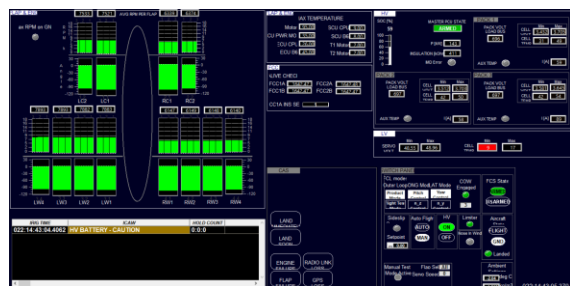
Fig. 15. Lilium Pilot and FTE's Displays (example 2).

[Operation] FTI Task Management System

All FTI tasks performed in the aircraft (valid for operation and design phases) must be traceable and logged accordingly: information like task subject, type, assignee and document reference are always important to be registered, and they can be useful for statistics analysis or even in case of incident research.
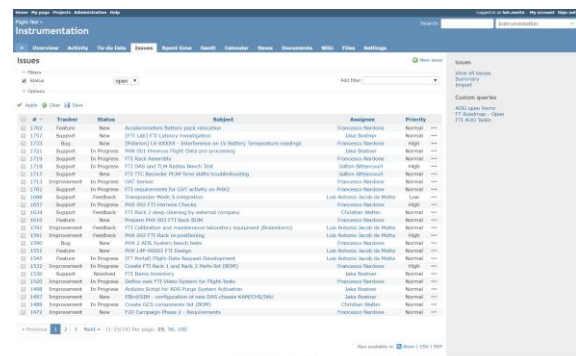
Fig. 16. FTI Tasks Management System using Redmine opensource issue tracker.

[Operation] General considerations for the FTI flight preparation

Test Plan, Risk Assessment, Configuration Control, Briefing, Pre and Post-Flight procedures are some of the mandatory safety and quality steps to be complied with when working in aviation - even more when flying unmanned.

#SafetyFirst

During performing the FTI pre-flight procedure, systems like Data Acquisition, Data Recorder, Telemetry Radio Links and Data Processing Servers are tested and released for flight.

Fig. 17. Lilium Jet Flight-Test field preparation.

After performing the test, all FTI Data is moved to the FTI servers and processed accordingly converting them from a Raw Tormat to Engineering Units (e.g.: from "bits" to "Meters, degree Celsius, PSI, etc..").

The reports from the Flight Crew are also helpful, providing relevant events and details about how was the aircraft's behavior during performing maneuvers, which will support system engineering during the data analysis.

[Operation] FTI Statistics Monitoring System

In order to improve tasks performance and/or evaluate how they are progressing, let's use the numbers in our favor. We can always correlate important Key Performance Indicators (KPI) with FTI planning and actions management.

FTIL growth, Design Changes rate and quantity of tests performed can give us a good expected workload indicator when performing a specific Flight-Test Campaign; or even we can calculate the average working time expended in a regular instrumentation implementation, for example.

Fig. 18. FTI Statistics Dashboard using Grafana opensource interface

Final Considerations

In the end, all conventional instrumentation concepts, combined with aviation best practices and engineering experience can provide a reliable and safer test environment, increasing the quality of data acquired and test results, maximizing the aircraft systems development and mitigating issues during performing the Flight-Test Campaigns.

Machine-Learning-Based Position Error Estimation for Satellite-Based Localization Systems

Tianxiang Lan[1], Andreas Dodinoiu[1], Arne Geffert[1], Uwe Becker[1]

[1] *Institute for Traffic Safety and Automation Engineering, Technische Universität Braunschweig, Braunschweig, Germany*

ORCID: 0000-0002-3943-1401 (Lan), 0000-0002-8183-6816 (Dodinoiu),
0000-0001-5115-5908 (Geffert)
t.lan@tu-braunschweig.de

Abstract

Satellite-based localization systems are nowadays widely deployed in transportation, especially with the progress of global navigation satellite systems (GNSS). However, GNSS signals are easily degraded by the local environment. This compromises the accuracy of the position solution and makes it challenging to implement satellite-based localization systems in the road and railway domain. With the help of Bayes filters (e.g., Kalman and particle filter), the localization accuracy can be improved. However, these filters are constrained by assumptions, and they require accurate modeling of errors for optimal estimation. Under these circumstances, another modeling method is researched in this paper. As machine learning has become more sophisticated over the years, neural networks are now suitable for learning the relation between the position errors and the abundant information from the GNSS receiver without prior knowledge. Therefore, the dilution of precision, the elevation angle and the carrier-to-noise ratio are appropriate indicators for signal degradation. In this paper, it is shown how neural networks are trained to estimate the position error of satellite-based localization systems. For modeling the temporal correlation in position error measurements, the long short-term memory (LSTM) network is applied. Finally, it can be demonstrated that the neural networks are able to learn the trend in position errors.

Key words: satellite-based localization systems, machine learning, land vehicles, LSTM, error estimation

1 Introduction

With the ongoing advancement of global navigation satellite systems (GNSS), satellite-based localization has become more dependable for transportation applications over the last decades. However, due to terrestrial influences, the application of satellite-based localization encounters major challenges in the road and railway domain. The satellite signals are easily deflected, diffracted and obscured by buildings, trees and tunnels on the ground. Commonly arising effects are for example the multipath effect and non-line-of-sight (NLOS) signals. That degradation of satellite signals leads to a reduction in accuracy and other characteristics concerning the measurement quality of the localization systems.

Many approaches are developed to mitigate the degradation of satellite signals. For example, the Bayes estimators such as the Kalman filter and the particle filter are nowadays widely implemented in the GNSS receivers for that purpose. The Kalman filter is however restricted by the assumptions of linearity and normal distribution.

The particle filter is applicable for nonlinear systems and non-Gaussian distributions, but it comes at the expense of high computational complexity. Furthermore, the Bayes estimators require explicit modeling of the effects to be mitigated.

Multipath and NLOS reception are to a certain extent deterministic effects because they mainly exist due to surrounding objects such as buildings. Therefore, an accurate geometric modeling of the degraded signals is feasible. An example would be the ray-tracing method [1]. By analyzing the path of satellite signals, possible deflection, diffraction and obscuration can be found with a 3-D map of the surrounding area. This method is naturally very computationally expensive and its accuracy is highly dependent on the accuracy of the map as well as on the model of signal transmission.

Other methods focus on the possible indicators for the existence of multipath and NLOS in the GNSS receivers. Groves et al. propose a portfolio to mitigate signal degradation under the consideration of multiple possible indicators as well

as antenna techniques [2]. Considered indicators are for instance the carrier-to-noise ratio (C/N0) and the elevation.

Data fusion is also applied to integrate GNSS with dead reckoning methods such as inertial navigation systems (INS) and odometers. However, the two latter ones lack long-term accuracy due to error accumulation over time and hence cannot guarantee accuracy of the integrated system during GNSS outages.

To improve the measurement quality of GNSS, different signal augmentation techniques can be applied, such as real-time kinematics (RTK), Precise Point Positioning (PPP) or a hybridized version of both to combine the benefits of both approaches (PPP-RTK). RTK improves the measurement quality by not only utilizing the time information of the satellite signal, but also by using the information gained by analyzing the characteristics of the carrier phase. This is achieved by using additional information provided by a near base station (30-50 km) or by a network of base stations and advanced algorithms. Thus, RTK can provide an accuracy of up to 1 cm. In contrast to that, PPP only relies on GNSS satellite clock and orbit corrections provided by a global network via satellite or internet link in order to estimate phase characteristics and can achieve an accuracy in the decimeter level. It is a challenging limitation that the algorithms need a different period of time for the position solution to converge (the so-called position fix), i.e., from a few seconds (RTK) up to 30 minutes (PPP). The position fix can be easily lost in environments with significant amounts of signal degradation, drastically reducing the availability of such systems. [3].

However, to the knowledge of the authors, the existing methods can provide a solution to some – but not to all – issues at hand. Over the last decade, machine learning techniques have become a promising solution especially for the cases when no exact models are required for the problems to be studied [4]. In this paper, the possibility of applying machine learning as an estimator for the position error of satellite-based localization systems is explored and discussed.

2 Relevant Work

Similar to other domains, there is a growing interest in applying machine learning to satellite-based localization systems. In this chapter, relevant research is shortly introduced.

Satellite-based localization systems with GNSS and INS are nowadays commonly found. However, an INS suffers from error propagation over time and the position error from the INS can accumulate to a large level in a short period. Ma-

chine learning is applied to learn how GNSS corrects the INS error during its normal operation. The trained algorithm can then be implemented to mitigate the INS error during GNSS outages and to assure accurate localization over a longer time span using a back propagation neural network [5]. Wang et al. further suggest that a hybrid architecture with a radial basis function neural network and an extreme learning machine can improve the accuracy of prediction under dynamic conditions [6].

Machine learning is also adopted to improve the accuracy of map-matching algorithms for satellite-based localization systems. Hashemi and Karimi apply an online feedforward neural network to predict and reduce GNSS errors in order to improve the correct segmentation rate of the map-matching algorithm. The speed and horizontal dilution of precision (HDOP) are adopted as the inputs for the machine learning algorithm under the assumption that the position error is temporally autocorrelated. However, the results prove their assumption wrong. Instead, the neural network provides the best performance when the input is constant. [7].

Machine learning is also applied to aid the detection of multipath and NLOS. Hsu develops a support vector machine (SVM) to distinguish line-of-sight (LOS), multipath and NLOS signals by analyzing features such as the received signal strength, the change rate of received signal strength and the pseudorange residue [8]. He reports a classification accuracy of about 75% for certain features.

Kuratomi conducts similar research compared to this paper. He develops a position error estimator for GNSS using machine-learning-based methods such as decision trees and support vector machines. Common receiver measurements such as C/N0, elevation as well as the information coming from the navigation filter such as the innovation in the Kalman filter are applied as the inputs for the estimator [9]. Furthermore, a camera is used to detect multipath and the results are fed to the machine learning algorithms as additional inputs. Though the final results are promising, he proposes to use deep neural networks to achieve higher estimation accuracy.

3 Fundamentals

In this chapter, the sources and the propagation of GNSS errors are firstly discussed. After that, the used machine learning algorithm as well as the applied dataset are shortly introduced.

3.1 Error Sources of GNSS

The accuracy of GNSS is under the influence of many sources. The major sources are the clock error, the ionospheric and tropospheric errors as

well as multipath and NLOS. Fig. 1 illustrates that the signal paths from the satellites can be extended in the atmosphere as well as due to reflection on objects surrounding the receiver. These error sources directly cause a deviation in the pseudorange measurement corresponding to every satellite. The receiver position is then estimated by least square estimation or by the Kalman filter based on the pseudorange measurements. The relation between the pseudorange measurement of the ith satellite and the receiver position $[x_r, y_r, z_r]$ to be estimated can be expressed as

$$\rho_i = (x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2 + \delta\rho_i \quad (1)$$

where $[x_i, y_i, z_i]$ is the satellite coordinate and $\delta\rho_i$ is the sum of pseudorange errors. It can be easily noticed that the pseudorange measurement is the sum of the geometric distance between receiver and satellite and the range errors. Moreover, the relation between pseudorange and position is nonlinear. The propagation of the pseudorange error to various localization solutions such as position, velocity and time (PVT) information is dependent on the signal geometry, which can be generally understood as the distribution of all used satellites in 3-D space. The signal geometry can be quantified by the dilution of precision (DOP) [10]. The aforementioned HDOP is a variant of the DOP and is dedicated to horizontal planes, i.e., to describe the influence of the satellite geometry on the accuracy of the horizontal position solution. A high HDOP value indicates that the pseudorange errors propagate to the horizontal plane to a great extent.
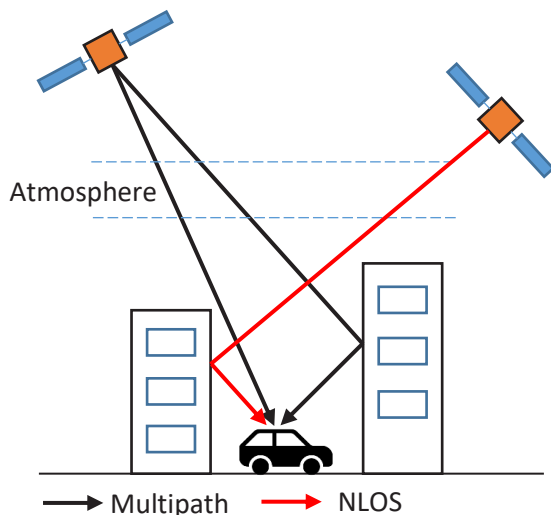

Fig. 1. Illustration of major error sources for GNSS

Several measurements of the receiver can be linked to the error sources. The elevation is the angle between the signal (from satellite to receiver) and the horizontal plane. Generally, signals from satellites of low elevation are more influenced by atmospheric effects since their paths in the atmosphere are longer than the paths of signals from high-elevation satellites. Moreover, low-elevation signals are more likely to be degraded by multipath and NLOS since the signals are more probable to be blocked and reflected [10]. C/N0, the carrier-to-noise ratio or signal-to-noise ratio, is considered to be relevant to the quality of the received signals. In urban areas, reflections happen frequently and C/N0 is observed to be lower. Hence, C/N0 is regarded as indicator for multipath and NLOS [11].

Many models have been developed to analyze the impact of error sources on pseudoranges. For ionospheric and tropospheric effects, there are sophisticated models that are especially applied to receivers of high quality. These models are complex and require current data for a given region from base stations.

For multipath and NLOS, the aforementioned ray-tracing method has already been implemented for simulation purposes in commercial software. However, the required 3-D map is not available everywhere due to high data volumes as well as due to the high complexity in generating the map. If the map is not up-to-date, it would also lead to an accuracy degradation of the ray-tracing method. Moreover, running ray-tracing in real time on vehicles would lead to high expenses on the computation hardware. Furthermore, multipath and NLOS do not simply lead to the geometric elongation of signal paths, but the resulting errors are also dependent on the signal processing inside the receiver [12].

A model to correctly describe the error propagation from diverse error sources via pseudorange measurement to the position errors is extremely difficult to establish. Even if it can be realized, it would be excessively complex with the classic modeling techniques based on linear or nonlinear equations and various assumptions on the underlying statistical distributions.

3.2 Machine Learning

The advantage of machine learning is that it does not require explicit modeling and that it can be trained to discover the model by learning the linkage between data. Due to the difficulties that classic approaches have in finding a model for the position errors of satellite-based localization systems, machine learning could be suitable to find this model based on the collected data in the field.

Generally, there are three main kinds of learning tasks with machine learning: supervised learning, unsupervised learning and reinforcement

learning. Supervised learning is targeted to learn the mapping function between the inputs and the outputs based on the input-output pairs in the training set and it is commonly applied to solve, among others, regression problems. Apart from that, unsupervised learning is often adopted to study the distribution of data and it is widely used for clustering and generating data. Reinforcement learning, however, has the feature that it can update the model during deployment, whereas for supervised and unsupervised learning, the model is determined during the training phase and remains unchanged during deployment. Since the target of this paper is to find the relation between several indicators and the position errors based on the collected data – without further deployment of the model in the field –, supervised learning is suitable.

Many models can be applied to perform a machine learning task, such as the feedforward neural network by Karimi [7]. This type of neural network is normally applied to link the relation between inputs and outputs of the same time epoch, i.e., it only considers the static behavior. However, the position errors of satellite-based localization are timely correlated, i.e., the position error at time epoch t_i can also be under the influence of error sources of previous time epochs [13]. The recurrent neural network (RNN) is specially designed for processing sequential data [14] and is therefore suitable to model time series. As shown in Fig. 2, an RNN layer with multiple cells can process a sequence. The cells not only learn the inputs, but also pass information further to the next cell.
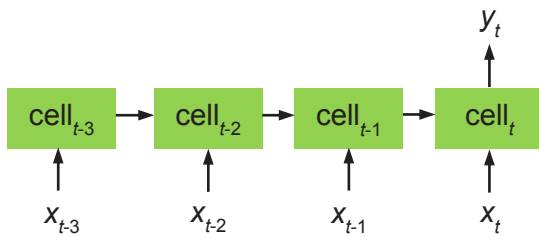

Fig. 2. Illustration of a typical many-to-one mode of RNN

The term RNN covers a class of recurrent neural networks. The most widely applied network in that class is the LSTM [15]. It is suitable for capturing both the long-term and the short-term pattern in the sequence. An LSTM network consists of input i_t, forget f_t and output gates o_t. LSTM networks use the cell state vector c_t to receive information from previous steps in the processed sequence and to pass the information further to the following steps in that sequence. Mathematically, an LSTM cell can be described using following equations

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{5}$$

$$h_t = o_t \odot \sigma(c_t) \tag{6}$$

where σ represents the sigmoid activation function and $\odot$ denotes the element-wise multiplication. The major improvement of LSTM against original RNN is that the vanishing and the exploding of gradients is avoided using the forget gate [15]. LSTM networks can be applied for machine translation [16], also for the prediction of time series [17].

3.3 Dataset

The data applied in this paper were collected in Braunschweig, Germany, on several days in August 2014. For the data collection, an automotive-grade u-blox receiver and an OxTS RT 3002 integrated system were used as the common receiver on the vehicle and as reference respectively. The test vehicle followed the designed trajectory for 20 times in total. Data collected during 5 of those 20 runs are used as dataset since they were all collected under the simplest receiver settings without additional correction services. Those settings truly represent the quality of an automotive-grade GNSS receiver. The trajectory consists of environments including urban areas, multiple bridges, forest, suburban area and highway (Fig. 3). Both the u-blox receiver and the reference system recorded data at 5 Hz frequency.

With the aim to identify suitable inputs for the machine-learning-based model, the available quantities provided by the receivers are analyzed hereinafter. Apart from the PVT information, the u-blox receiver can provide different metrics of dilution of precision, C/N0, elevation and azimuth of the used satellites. Utilizing all available C/N0 and elevation angles of the used satellites as inputs for the neural network would render the dataset too complex. Hence, those measurements are reduced to their mean values. For C/N0, another parameter is considered, i.e., the percentage of satellites with low C/N0. This is done under the assumption that if most used satellite signals are degraded by errors and indicated by lower C/N0, the estimation algorithm in the receiver can neither identify nor exclude the erroneous measurements. Thus, large deviations in the position measurements can be expected. Here, the threshold for low C/N0 is defined at 20 dB. Moreover, the model requires reference values of position errors to evaluate the
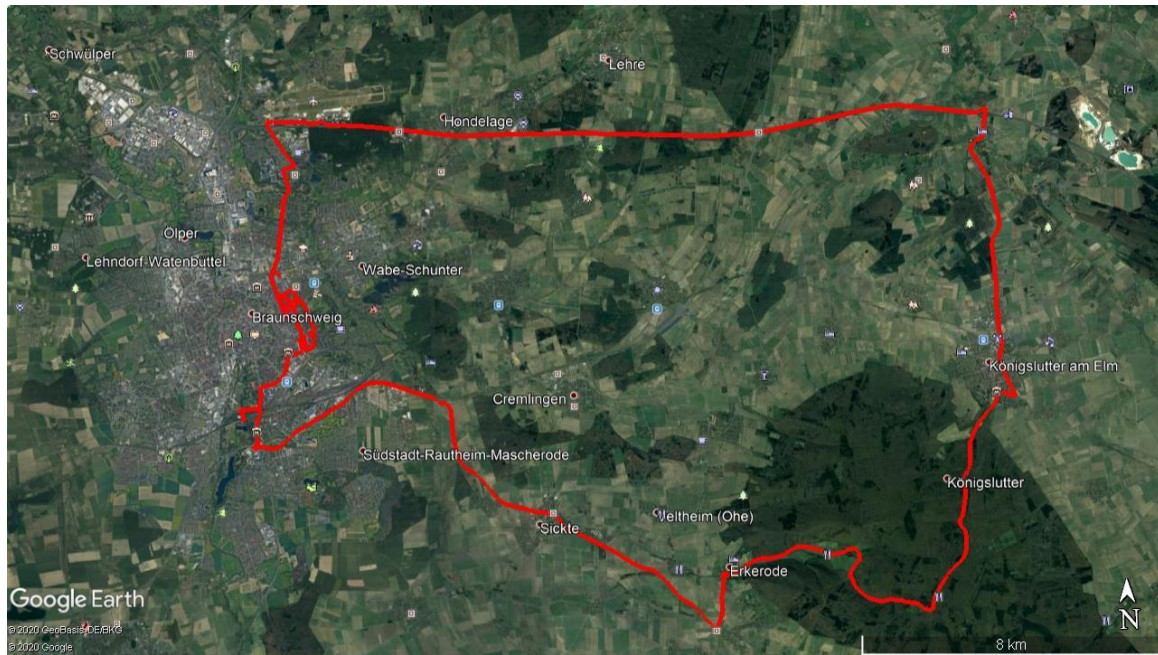
Fig. 3. Trajectory for data collection [18]
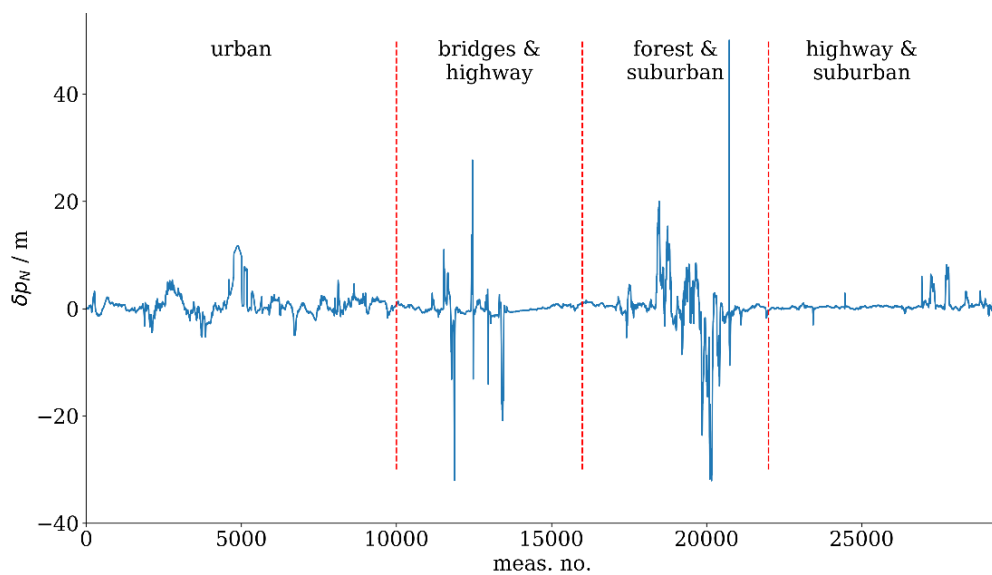
Fig. 4. Position error in north direction collected in one run

estimation. The reference values of the position errors are calculated as the difference between the positions measured by the u-blox receiver and the reference receiver.

It can also be observed in Fig. 4 that the data for diverse environments differ from each other. Using a single model to learn all the features throughout the run would be particularly challenging. Therefore, the dataset is segmented on the basis of the aforementioned environments.

3.4 Model

Based on the understanding of position errors in the fields of satellite-based localization and ma-

chine learning, a neural network with LSTM layers is applied in this paper to model the position errors. The model consists not only of LSTM layers, but also of a fully connected linear layer. The LSTM layers are used to discover the underlying temporal correlation and the fully connected layer links that correlation to the output. Moreover, a dropout layer is inserted between the LSTM and the fully connected layers to improve regularization in order to avoid overfitting. The architecture of the model is depicted in Fig. 5.

Specifically, the following measurements are adopted as the inputs for the neural network:

- number of used satellites
- HDOP

- mean value of C/N0 of used satellites
- mean value of elevation angle of used satellites
- percentage of used satellites with low C/N0 (< 20 dB)
- speed over ground.
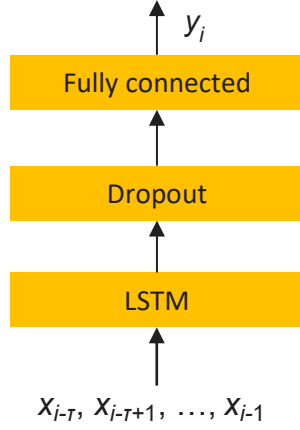
Fig. 5. Architecture of the applied neural network

Two models with the same architecture as depicted in Fig. 5 are developed for two kinds of outputs:

- Model 1: the position errors in north and east direction [δp_N, δp_E] in local navigation coordinates are the outputs:

- Model 2: the horizontal position error δp_H, calculated by following equation, is the output:

$$\delta p_H = \sqrt{(\delta p_N)^2 + (\delta p_E)^2}. \qquad (7)$$

Mathematically, the models can be formulated as:

$$y_i = f(x_{i-\tau}, x_{i-\tau+1}, \ldots, x_{i-1}) \qquad (8)$$

where

- y_i is the position error(s)
- $x_{i-\tau}$ is the input(s) at $t_{i-\tau}$
- τ is the time lag in the processed sequence.

The target is to minimize the difference between the output estimated by the neural network $\hat{y}_i$ and the reference values of position errors y_i. The mean squared error (MSE) is used as the loss function. Mathematically, the MSE loss is expressed as

$$L = \frac{1}{N} \sum_{N}^{n} (\hat{y} - y)^2 \qquad (9)$$

where N is the batch size.

4   Experiments and Results Analysis

For the training of the models, the dataset for the environment "urban area" is used. The sizes of training, validation, and test sets are respectively 36000, 4500, 4500. Here the size refers to the number of timestamps with multiple quantities measured per timestamp. The data are first normalized to improve the convergence speed and to avoid instability in the training. Furthermore, the dataset is sliced in a window form for the training of the LSTM network. This process is depicted in Fig. 6. As it can be seen, the window takes every time a subsequence in length of τ, which corresponds to the input array mentioned in eq. 8. After that, the window increments one step further down in the input sequence.
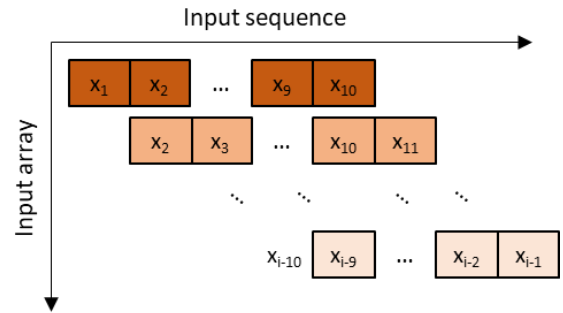
Fig. 6. Window method for dataset preparation with time lag equal to 10 measurement points

The models are implemented with the help of the machine learning library PyTorch [19] in Python. The Adam [20] and the rmsprop [21] optimizer have been tested. For the applied dataset, the rmsprop optimizer provides better results.

For model 1, it can be observed as in Fig. 7 that the estimated value of the model can generally follow the trend of the reference value. However, the model encounters difficulties in capturing the short-term features such as the peaks with extreme values. Moreover, the estimated values show a high frequency fluctuation. After further parameter tuning, it is noticed that a stacked LSTM layer or an LSTM layer with more hidden units cannot improve the drawbacks of the model. For the test set, the MSE loss for model 1 is 14.00 m^2.

Model 2, however, presents different results. It can be noticed in Fig. 8 that model 2 can capture the short-term peak at the beginning of the dataset. It can also trace the trend of the reference value. The MSE loss for the test set is 2.85 m^2. From the 2-D density plot in Fig. 9, it can also be noticed that most estimations lie near to their corresponding reference values. This result shows that model 2 could potentially be used to
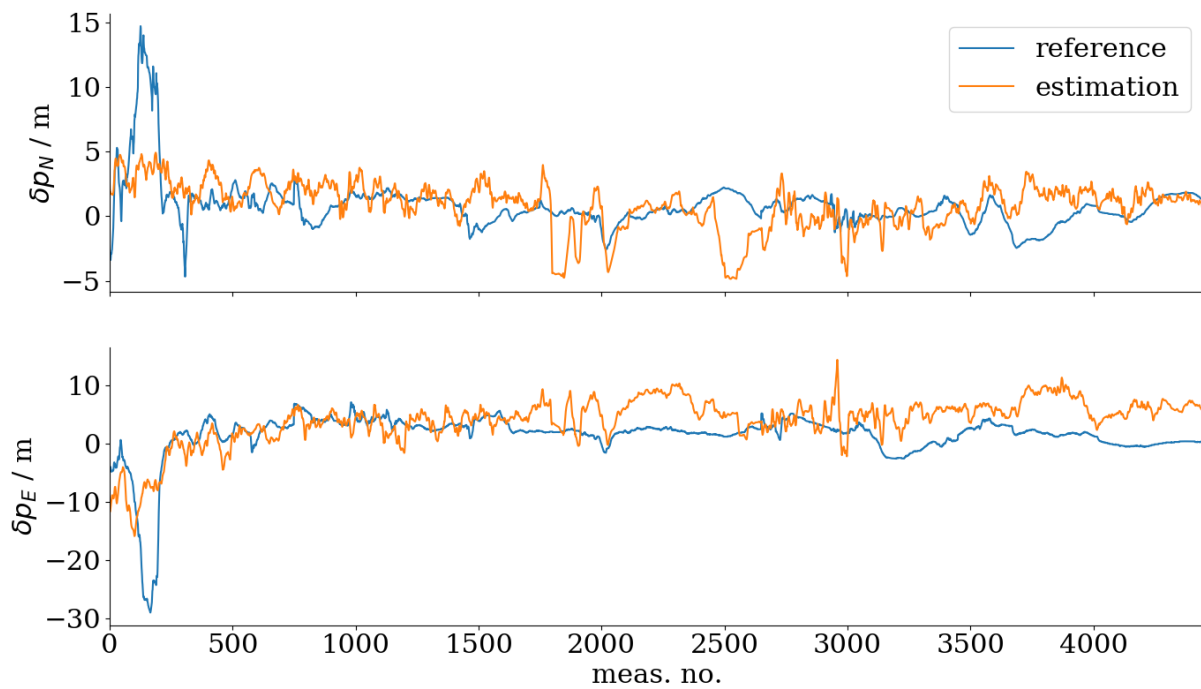
Fig. 7. Comparison of estimated and reference position error for model 1
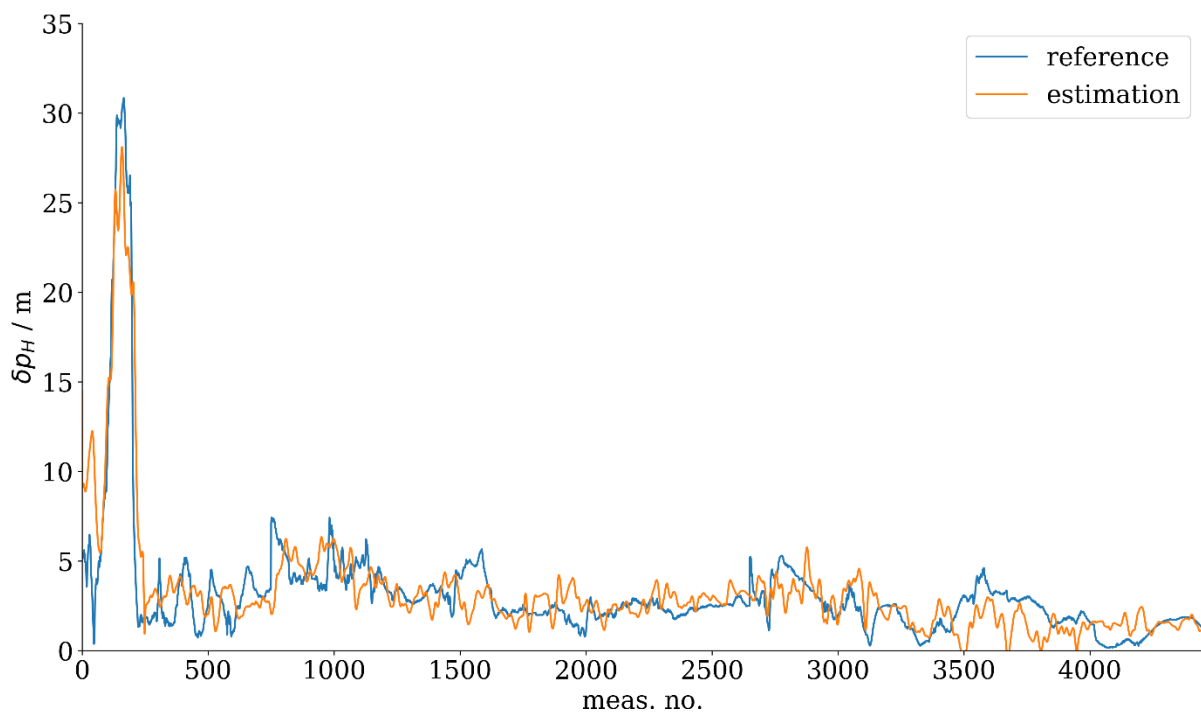
Fig. 8. Comparison of estimated and reference position error for model 2

calculate the horizontal protection level which is used for the autonomous integrity monitoring of GNSS-based localization systems.

Possible reasons for the existing problems are summarized as follows:

- The used inputs are only indirect indicators for the error sources. More inputs are possibly required to capture the transient arising peaks;

- Position errors contain both the deterministic part, which can be traced back to the inputs, and the stochastic part, which cannot be linked with the inputs. Thus, the model would not be able to learn the stochastic elements in the position errors;

- Training and test demonstrate slightly different patterns. Despite the effort to improve regularization, the model is still

overfitted to the training set. This resides in the nature of machine learning that it is data-dependent.

Fig. 9. Comparison of estimated and reference position error for model 2. The color scale represents the probability of occurrence.

Nonetheless, the models show the ability to predict the trend, especially the long-term bias is well learned.

5   Conclusion and Outlook

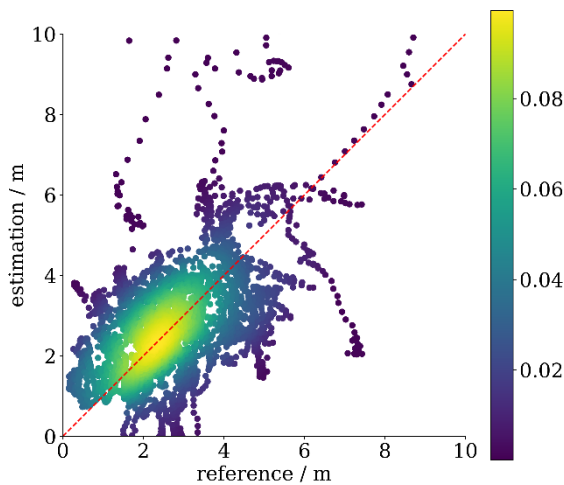A machine-learning-based position error estimation for satellite-based localization systems is developed with the aim to explore an alternative to current common methods such as Bayesian estimators. The analysis shows that an explicit modeling from the error sources to the position domain is difficult to establish. Especially in the localization environment of land vehicles, local effects like NLOS complicate the modeling. In contrast, machine learning has the advantage that it can be trained to learn the effects of the measurement environment based on data.

Neural networks with an LSTM layer are applied for position error estimation based on the assumption that position errors are temporally correlated. Several common outputs of automotive-grade GNSS receivers are used as the inputs for the neural network to estimate position errors. Through simulation, it is shown that the trained networks have the ability to learn the long-term trend in the position error datasets. However, they do not satisfactorily accomplish to fit the short-term noise.

Possible improvements could be made by developing an alternative approach to model stochastics, i.e., a decomposition of position errors into deterministic and stochastic parts. Apart from that, modifications to the neural networks, such as a more complex architecture, are worth considering for research in the future as well.

References

[1] M. Obst, S. Bauer, and G. Wanielik, Urban multipath detection and mitigation with dynamic 3D maps for reliable land vehicle localization, in *IEEE/ION Position, Location and Navigation Symposium (PLANS), 2012: April 23-26, 2012, Myrtle Beach Marriott Resort & Spa, Myrtle Beach, South Carolina, Myrtle Beach, SC, USA,* 685–691 (2012); doi: 10.1109/PLANS.2012.6236944

[2] P. D. Groves, Z. Jiang, M. Rudi, and P. Strode, A Portfolio Approach to NLOS and Multipath Mitigation in Dense Urban Areas, in *ION GNSS 2013* (2013)

[3] European Global Navigation Satellite Systems Agency, PPP-RTK market and technology report (2019). Accessed: May 10, 2020. [Online]. Available: https://www.gsa.europa.eu/sites/default/files/calls_for_proposals/rd.03_-_ppp-rtk_market_and_technology_report.pdf

[4] G. Rebala, A. Ravi, and S. Churiwala, *An introduction to machine learning*. Cham: Springer (2019); doi: 10.1007/978-3-030-15729-6

[5] Q. Zhang and B. Li, A low-cost GPS/INS integration based on UKF and BP neural network, *2014 Fifth International Conference on Intelligent Control and Information Processing (ICICIP): 18 - 20 Aug. 2014, Dalian, Liaoning, China, Dalian, China,* 100–107 (2014); doi: 10.1109/ICICIP.2014.7010322

[6] D. Wang, X. Xu, and Y. Zhu, A Novel Hybrid of a Fading Filter and an Extreme Learning Machine for GPS/INS during GPS Outages, *Sensors (Basel, Switzerland)*, vol. 18, no. 11 (2018); doi: 10.3390/s18113863

[7] M. Hashemi and H. A. Karimi, A Machine Learning Approach to Improve the Accuracy of GPS-Based Map-Matching Algorithms (Invited Paper), in *2016 IEEE 17th International Conference on Information Reuse and Integration: IRI 2016: 28-30 July 2016, Pittsburgh, Pennsylvania, United States: proceedings, Pittsburgh, PA, USA,* 77–86 (2016); doi: 10.1109/IRI.2016.18

[8] L.-T. Hsu, GNSS multipath detection using a machine learning approach, in *IEEE ITSC 2017: 20th International Conference on Intelligent Transportation Systems: Mielparque Yokohama in Yokohama, Kanagawa, Japan, October 16-19, 2017, Yokohama,* 1–6 (2017); doi: 10.1109/ITSC.2017.8317700

[9] A. Kuratomi, GNSS Position Error Estimated by Machine Learning Techniques with Environmental Information Input, KTH Royal Institute of Technology, Sweden, 2019. Accessed: Apr. 26 2020. [Online]. Available: http://kth.diva-portal.org/smash/get/diva2:1362035/FULLTEXT01.pdf

[10] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Boston: Artech House (2013)

[11] N. Viandier, D. F. Nahimana, J. Marais, and E. Duflos, GNSS Performance Enhancement in Urban Environment Based on Pseudo-Range Error Model, in *IEEE/ION Position, Location and Navigation Symposium, 2008: Monterey, CA, 5 - 8 May 2008, Monterey, CA,*377–382 (2008); doi: 10.1109/PLANS.2008.4570093

[12] C. Hegarty, E. D. Kaplan, M. Uijt de Haag, and R. Cosentino, GNSS Errors, in *The GNSS technology and applications series, Understanding GPS/GNSS: Principles and Applications / Elliott D. Kaplan, Christopher J. Hegarty*, editors, E. D. Kaplan and C. Hegarty, Eds., 3rd ed., Boston: Artech House, 619–659 (2017)

[13] M. Olynik, M. G. Petovello, M. E. Cannon, and G. Lachapelle, Temporal Variability of GPS Error Sources and Their Effect on Relative Positioning Accuracy, in *Proceedings of the 2002 National Technical Meeting of The Institute of Navigation*, 877–888 (2002)

[14] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, (2016); [Online]. Available: http://www.deeplearningbook.org/

[15] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural computation*, vol. 9, 1735–1780 (1997)

[16] I. Sutskever, O. Vinyals, and Q. V. Le, Sequence to Sequence Learning with Neural Networks, in *Advances in Neural Information Processing Systems* 27, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds.: Curran Associates, Inc., 3104–3112 (2014); [Online]. Available: http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf

[17] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, Deep Learning with Long Short-Term Memory for Time Series Prediction, *IEEE Commun. Mag.*, vol. 57, no. 6, 114–119 (2019); doi: 10.1109/MCOM.2019.1800155

[18] Google Earth 7.3.2.5776, Braunschweig and surrounding area, Germany: 52°14'44.70" N, 10°39'25.73", elevation 103 m: 2020 GeoBasis-DE/BKG; 2020 Google. Accessed: April 27 2020

[19] A. Paszke et al., Automatic differentiation in PyTorch (2017)

[20] D. P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization (2014); [Online]. Available: http://arxiv.org/pdf/1412.6980v9

[21] G. Hinton, N. Srivastava, and K. Swersky, Neural networks for machine learning lecture 6a overview of mini-batch gradient descent

Intelligent SpaceWire Router with Multilevel Priority Arbitration and Multicast Packet Transmission

G. Skvarč Božič[1], M. Plattner[1]
[1] *Max Planck Institute for extraterrestrial Physics, Giessenbachstr. 1, 85748 Garching, Germany*
gaskvarc@mpe.mpg.de

Abstract:

SpaceWire (SpW) is a major communication standard used to connect systems in spacecrafts. It supports point-to-point and network connections. For the Wide-Field-Imager payload onboard the ESA ATHENA satellite, the next generation European x-ray observatory, we have designed an intelligent SpW router. It supports dynamic multilevel priority arbitration based on a maximum finder circuit and multicast packet transmission. The SpW router was implemented as a VHDL component to have a flexible design and a possibility to include it in other projects. The implemented SpW router has an RMAP supported configuration port for remotely configuring the router.

A test setup was designed to characterize and test the SpW router functionality. A routing delay and a packet router latency were determined. Simultaneous packet traffic from different sources with the same and different priority levels was emulated to demonstrate the arbitration functionality. Based on performed tests, the implemented router achieved a satisfying performance with the capability of handling transmission rates up to 200 Mbps. Moreover, the designed router increases payload system functionality, eases the design of more complicated networks, and enables a new way to design redundant systems with its multicast transmission support.

Keywords: SpaceWire, SpaceWire router, multilevel priority arbitration, multicast packet transmission, VHDL

Introduction

When designing a complex system, there is always a need for transferring data from one subsystem to another. Therefore, like for any other application, there are defined communication protocols for satellite onboard data-handling systems. Among those is SpaceWire (SpW), a communication protocol supporting point-to-point and network connections. It is used by several agencies, namely by ESA, NASA, JAXA, and Roscosmos for current and future space missions.

In the scope of ESA's ATHENA mission, the Wide Field Imager (WFI) instrument uses SpaceWire as the onboard data-handling system depicted in Fig. 1. Five Detector Electronics (DE) are connected through a SpW router (SWR) to a Central Processing Module (CPM) in the Instrument Control and Power Distribution Unit (ICPU). The CPM compresses the data from the detector electronics and sends it to the mass memory of the spacecraft (S/C). It is responsible for interpreting the telecom-

mands from the spacecraft with which it controls the ICPU subsystems and detector electronics.

Despite several SpW router devices available on the market from which two are directly supported by ESA, a ten port SpW router AT7910E (Atmel) and an eighteen port SpW router GR718 (Cobham). Both support two-level, high, and low, priority arbitration. And a few commercial products such as Flexible SpaceWire router with 2 to 32 ports from 4Links, 4-port SpaceWire router UT2000SpW4RTR with round-robin output arbitration from Cobham, SpaceWire router from NEC Japan, and an open-source six-port SpaceWire router IP core with round-robin arbitration from Shimafuji Electric, inc. We saw the need for the improved capability of a SpW router as there is no multicast transmission available in the mentioned devices.

The ICPU remote and main unit are designed in cold redundant, single-point failure configuration. Cross-strapping between the units is pos-

sible and optional. Nevertheless, as an example use case for the SpW router, the multicast feature is expected to simplify the design of cross-strapping between nominal and redundant systems depicted in Fig. 1. Offloading the required logic from DEs to the SpW router. For example, instead of sending two identical packets from DE, only one multicast packet is needed. Also, implementing a multilevel priority arbitration adds a degree of flexibility where each application protocol can be assigned a different priority level. Moreover, priority levels can be changed during operation if needed.
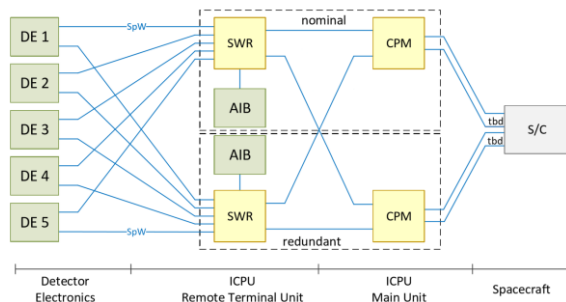

Fig.1.WFI SpW data-handling system.

SpaceWire standard

The SpaceWire standard is a standard for high-speed links and networks for use onboard spacecraft, easing the interconnection of sensors, mass-memories, processing units, and downlink telemetry sub-systems. SpaceWire was developed under ESA in the late '90s and formally standardized by European Cooperation for Space Standardization (ECSS) to provide space users with directly applicable specifications. SpW standard is very similar to Myrinet, a local area network (LAN) for computer clusters, especially regarding data and control characters. Both use 9-bit data characters and several control charters to control the flow of data through a link [1].

SpW, as a communication standard, has a defined protocol stack. The SpW protocol stack is composed of a Network layer, a Data Link layer, an Encoding layer, and a Physical layer. Compared to the well-known OSI model, the Network and Physical layer of the SpW protocol stack match the OSI model, whereas the Data Link and Encoding Layer both fall under the Data Link layer in the OSI model as can be seen in the Fig. 2.

The point of interest for this paper is the Network layer on which the SpW router operates. Other lower protocol layers are fulfilled by the SpW Coder/Decoder (CODEC) in router ports. No furthers explanation for these layers is provided since this is out of the scope of this pa-

per. The focus is on the router architecture and the control logic associated with it.
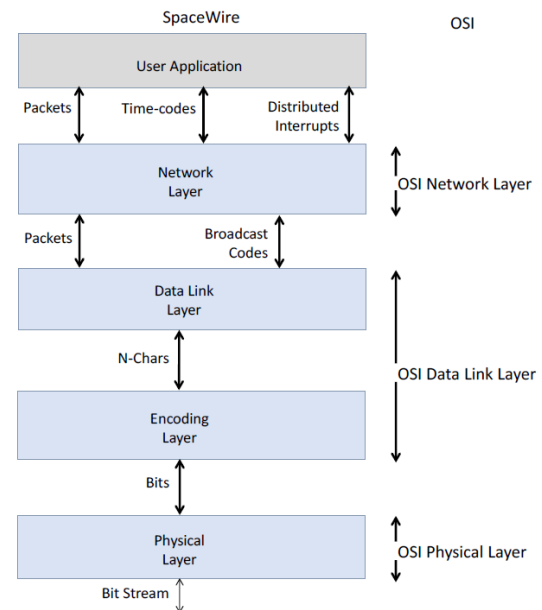
Fig.2.SpaceWire protocol stack compared to the OSI model [2].

Devices on a Network layer operate on packets. SpW packet structure depicted in Fig. 3 consists of a header, cargo, and end of packet marker (EOP).

Fig.3.SpaceWire packet structure.

The header of a SpW packet holds the address of the destination, and it is used to route a packet through a SpW network correctly. The Cargo section of the packet contains actual data which can be encapsulated in any user-defined protocol. Each packet is ended with an end of packet marker, which notifies a network device about the end of transmission. Moreover, it provides information about whether a packet transmission completed without any errors.

In theory, the SpW network is considered a switch-based network like Autonet, Myrinet, and ServerNet. Switch-based networks are composed of links and a set of switches, which together usually form an irregular network topology. Each switch has a set of ports where a subset of these ports is connected to other switches, a subset to network nodes, and some are left open. The network supports bi-directional full-duplex links with support for multiple links between two switches. However, the only guarantee of such networks is that the network is connected. They provide the flexibility that is required in LANs and the necessary scalability for designing scalable systems. However, their

irregular topology results in complicated routing. Therefore, in switch-based networks, no routing algorithm can be employed on a router level. There are two possible routing techniques. One of them is source routing. A source provides the packet's destination in terms of a path that needs to be taken through a network to reach its destination. And distributed routing where a routing table, which contains the routing information, is added to each switch in the network. Before any packets are transmitted, a network mapping algorithm needs to be executed to populate the routing tables.

Based on the above-described routing techniques, there are two primary addressing mechanisms in the SpW standard. One is called path addressing, and the other one logical addressing. Path addressing, as the name suggests, specifies the whole path that the packet needs to traverse to reach its destination. Therefore, the header of a path addressed packet consists of multiple characters where each character specifies one turning point in the network. Logical addressing, on the other hand, requires only one character which uniquely identifies a destination. However, for logical addressing to work routers in the network need a routing table where each routing table entry maps a corresponding logical address to one of the output ports or multiple output ports in case of multicast.

The purpose of the router is to connect multiple nodes and to route packets using the wormhole switching technique from any input port to any output port based on information within the header. A packet can be forwarded through two or more ports if a multicasting feature is provided. The number of external ports in the router is limited to 31. Whenever two or more arriving packets have the same output port as the destination, arbitration is needed to resolve the output port contention. Together with low-error rate, low footprint, low-cost, low-latency, full-duplex, point-to-point links, they form a SpW network. Providing high-speed (2 Mbps to 200 Mbps), bi-directional, full-duplex connection between two network nodes.

Based on SpW standard specifications, a SpW router shall comprise of[2]:

- One or more ports that interface to the SpW network.

- A switch matrix which connects an input port to an output port.

- A routing table which, together with the leading byte of a packet (header), determines through which port a packet is forwarded.

- A configuration node that is accessible through the port 0 (configuration port) and enables configuration of the router and port parameters.

The SpW standard includes some additional features which are not listed here since they were not implemented in the current version of the presented SpW router. Mainly to simplify the design and to shift the focus on to the control logic.

A configuration node of a router was mentioned. There is a defined communication protocol for accessing the memory of a remote network node called Remote Memory Access Protocol (RMAP). It can be used for configuring and updating the router and port parameters from anywhere within the SpW network [3].

SpaceWire Router Architecture

SpW router architecture presented in this paper is based on a generic router model depicted in Fig. 4, router architectures of currently available devices mentioned in the introduction, and Network-On-Chip routers presented in [4, 5]. Key components of a router are [6]:

- **FIFO buffers**. They are used for storing messages in transit. They are present in input and output ports. Size of which depends on the employed switching technique.

- **A switch**. Physically connects input ports to output ports. Fully connected in high-speed implementations.

- **Routing and arbitration unit**. Responsible for executing a routing algorithm, forwarding incoming messages to the right output port, setting the switch, resolving conflict regarding simultaneous access of an output port. Also known as control logic and switch allocator.

- **Link Controller (LC)**. It controls the flow of messages over a physical link.
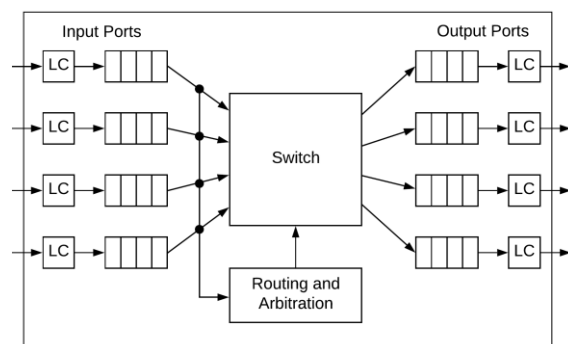

Fig.4.Generic router model [6].

The SpW router uses a wormhole switching technique as it enables the use of small input and output buffers. Considering the router is implemented as a VHDL IP-core, small buffers are desired. One consequence of the wormhole switching technique is that packets can span through multiple routers at once, which can lead to blocked paths and, in the worst case, a deadlock. Certain preventive measures can be implemented on the router level, such as packet timeout. However, a degree of caution should be taken when designing a network, including wormhole-based routers.

Considering specifications from the SpW standard, a few additional components were added to the generic router model to form the complete SpW router architecture depicted in Fig. 5.
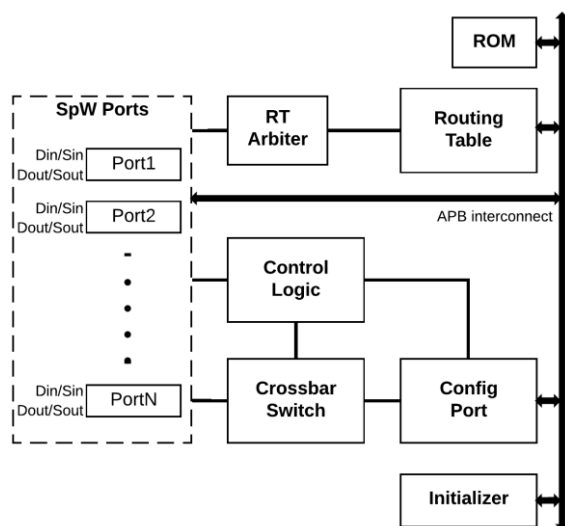

Fig. 5. SpW router architecture.

One desired feature is the possibility of remotely configuring the SpW router. For accomplishing this, a configuration port or node, which is accessible via port 0, is added. The purpose of the configuration node is to create an interface between SpW and internal interconnect. Its task is to decode and encode RMAP packets. When an RMAP command is decoded, it is translated into corresponding internal interconnect command. If decoded RMAP command requires a response (e.g., read command), then data from the internal bus is encoded into the RMAP response packet. The core logic of the RMAP decoder was taken from an open-source router implementation from Shimafuji Electric Inc. [7]. It was adapted for the custom control logic, different internal interconnect, and the RMAP packet Cycle Redundancy check (CRC) calculation was changed from a look-up table implementation to on the fly calculation.

The configuration port is connected to all router register via the Advanced Peripheral Bus

(APB). All registers and the routing table are memory mapped to comply with the RMAP standard. APB was selected as it is simple to implement, and there is no need for a higher performance bus since only register read/write operations are performed. The data width of the interconnect matches the width of the registers. Therefore, only one bus access is needed when manipulating the register data. It is true that the routing table, which covers a larger memory space, is connected to the same bus. However, this does not justify the use of a more complex bus as the routing table is in usual cases accessed only by its entry fields, which also match the APB data width.

All registers are implemented with fabric D flip-flops; therefore, the content of registers is lost after power loss. The same goes for the routing table, which is based on the inferred SRAM cells of the target FPGA device. As an option, if the target device has an included Read-Only Memory (ROM), the router can be initialized to a default state after a power cycle or reset of the device. Initialization is done by the initializer block, which reads data from the ROM and writes it to an appropriate register or routing table entry. If ROM is not present on the target device, the SpW router must be remotely initialized using RMAP before sending any normal SpW packets.

The routing table fulfills the requirement for SpW logical addressing. Moreover, it holds the information about the priority level of a logical address and some additional control flags. At this point, it is composed out of three data banks of size 256x32-bit with the possibility of expending the number of banks. Each bank represents a different routing table entry field. A destination field which is used in normal logical addressing meaning the logical address is mapped to one output port. A multicast set field that maps one logical address to multiple output ports when multicast packet transmission is enabled by the multicast enable flag. A control field where the 8-bit encoded priority level of a logical address, logical address deletion flag, and group adaptive routing enable flag are stored. The first 32 routing table entries are reserved for direct external port addressing when the path addressing is used. With this, there are 223 available logical addresses left plus the reserved logical address 0xFF. Logical addresses directly map to the addresses of the routing table entries; therefore, no additional address decoding is needed when accessing the routing table. The header of an incoming SpW packet is directly used as an address applied to the routing table. Routing table entry fields are 32 bits wide, which matches the SpW

standard limit of 31 external ports plus the port 0. This way, each bit position corresponds to the external port number with the same number as the bit position.

The routing table is a common resource for all input ports. Therefore, all requests are processed by an arbitration unit, which resolves the contention for the routing table access. Since the priority level of a packet is unknown before accessing the routing table, it makes sense to employ a fair arbitration such as round-robin.

SpW ports are the link controllers of the SpW router. Each SpW port is composed out of a SpW CODEC, which is responsible for maintaining an active link between two network devices. And a packet processor that is responsible for extracting the routing information from an incoming packet, communicating with control logic, and controlling the flow of incoming packets.

All SpW ports and port 0 are connected through a fully connected non-blocking crossbar switch. Full connectivity allows routing packets from any input port to any output port. Non-blocking characteristic allows multiple connections at once as long as an output port is not the same destination for multiple input ports. When there are no active requests for an output port, it is disconnected from all input ports.

Crossbar switch allocation is controlled by the control logic. Control logic is divided into two subcomponents, an arbitration unit that handles requests from input ports. And a switch allocator that generates correct control signals for the crossbar switch based on the given grant signals from the arbitration unit. Each output port is associated with its priority arbitration and switch allocator logic.

Arbitration in the control logic is designed as a two-stage arbiter inspired by the work presented in [5]. It follows a return-to-zero four-phase handshake protocol for the request and grant signal pair. There is no dedicated release signal; the de-asserted request signal releases the output port. The control logic itself is designed as a three-stage pipeline (see Fig. 6). In the first pipeline stage (Stage0), the request and priority signals are captured. In the second pipeline stage (Stage1), the priority signals are masked with the request vector to process only priority levels of active requests. The highest priority is determined with a k-bit maximum/minimum finder circuit based on Array-Based Topology (AbT), a fast and efficient circuit topology of a max/min finder circuit [8]. When only one high priority request is found, the arbitration process is complete, and the one-hot encoded position of this request (Addr OH) is passed to the grant signal generator in the next pipeline stage (Stage2). After that, the grant signal is passed to the output grant signal on the next rising clock edge.

A problem occurs when there are multiple requests with the same highest priority (among current active requests). If that is the case, requests with the same highest priority are isolated and passed to the second arbitration stage in the third pipeline stage (Stage2) of the control logic. Since these requests have the same priority, a fair round-robin arbiter determines which of those receives a grant signal. At the end of the pipeline, a multiplexer selects the appropriate grant vector based on whether there were multiple requests with the same highest priority or not.

Fig. 7 illustrates the above-described arbitration process in case of packets with matching priority. The example is divided into four frames. In the first frame, tow packets are waiting to be forwarded through the same output port o2.
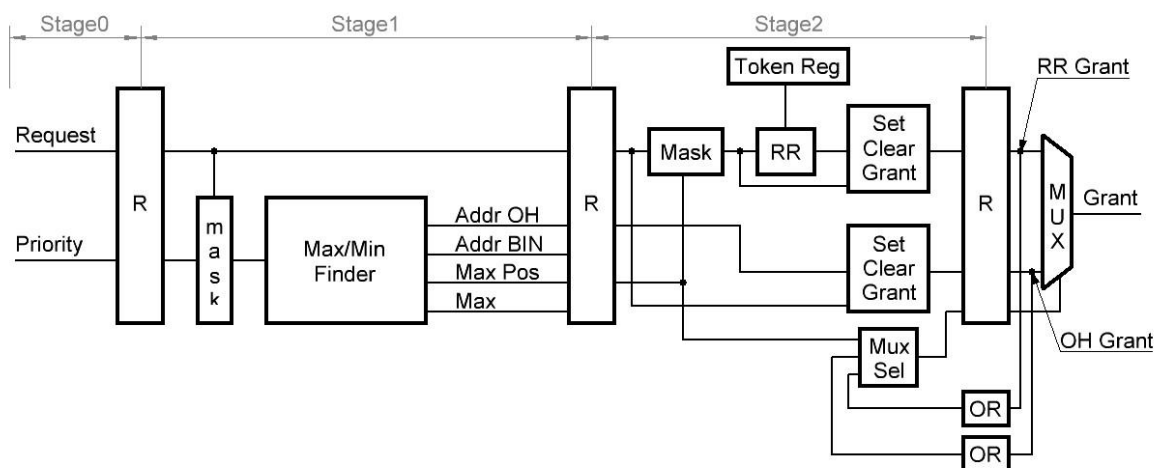

Fig.6.Control logic pipeline [9].

Meaning round-robin arbitration takes place. The last input port to have access to the output port o2 was the input port i4. Therefore, in the second frame, the input port i1 is selected, as it is the first one in line based on the previous access. Meanwhile, during the transmission, another packet with the same address arrives at the input port i2. In the third frame, input port i2 is selected even though the packet arrived later that the one on input port i3. Because of the round-robin arbitration, the port i2 is higher in line then the port i3. Finally, in the fourth frame, the input port i3 is granted access to the output port o2.
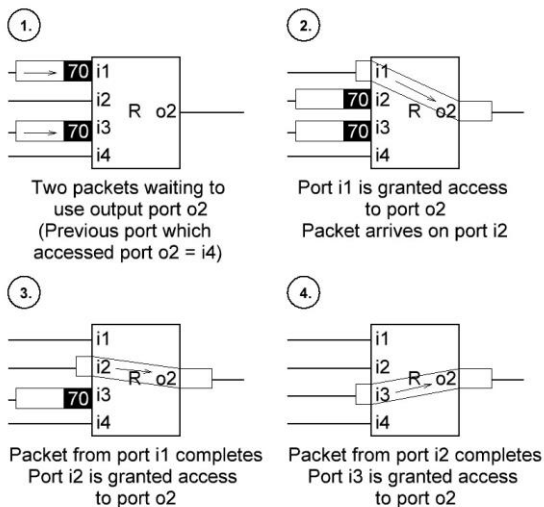

Fig.7.Arbitration of packets with matching priority [9].

Fig. 8 illustrates another arbitration example this time with packets with different priority levels. The example is divided into six frames where packets with logical address 35 have higher priority than packets with logical address 70. The first frame in this example is the same as in the previous example. However, in the second frame, during transmission of a packet from the input port i1, a packet with higher priority arrives at the input port i4. Therefore, in the third frame, input port i4 is selected as the next port to access the output port o2 after the completed transmission of a previous packet. During the transmission of the packet from the input port i4, another lower priority packet arrives at the input port i1. In the fourth frame, no high priority packets are present at input ports. Previous input port with access to the output port o2 when a packet with address 70 was present, was the input port i1. Meaning, the first input port in line is port i2. However, there is no packet waiting on this port. The first place is passed to input port i3, which is selected, and the packet is forwarded to the output port o2. In the fifth frame, after the previous packet has completed transmission, the input port i1 is selected. The packet from the input port i1 is

forwarded to the output port o2. In the sixth frame, the last packet has completed transmission.
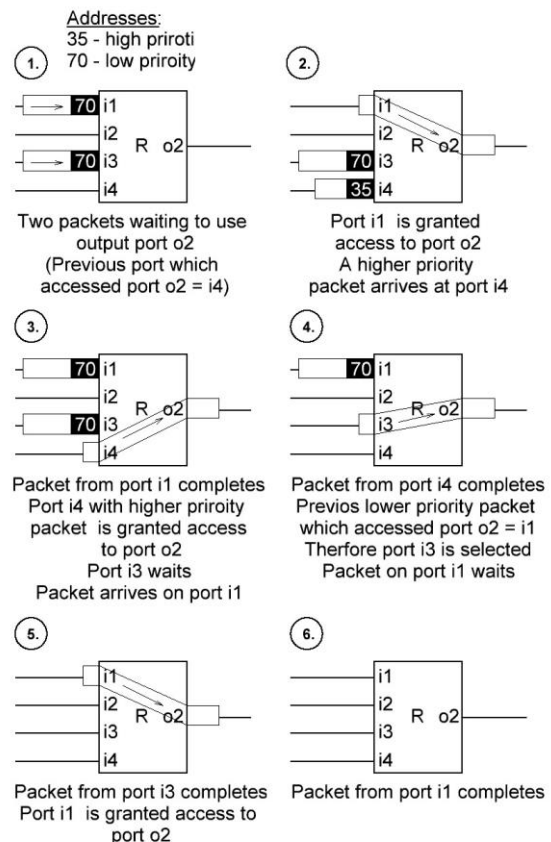
Fig.8.Arbitration of multiple packets with different priorities [9].

Multicast packet transmission is implemented in a way that the packet is forwarded through the selected output ports only when all of them can receive a new packet. Otherwise, the packet has to wait for this condition to be true.

Fig. 9 illustrates how the control logic handles multicast packet transmission. The example is divided into four frames where the multicast is enabled for the logical address 70. Indicated by a set Bit 0 in the multicast-set. Set bits on the other positions in the multicast-set denote the output ports through which the multicast packet should be forwarded. In the first frame, a multicast packet arrives at the input port i1. However, it is not immediately forwarded because one of the output ports specified by the multicast-set is busy. In the second frame, the input port i1 waits for the availability of all ports within the multicast-set. In the third frame, the packet that previously occupied the output port o2 has completed transmission. Now the multicast packet can be forwarded through the targeted output ports, o2 and o4. In the fourth frame, the multicast packet completes its transmission.

Fig.9.Multicast packet transmission [9].

Results

The SpW router IP-core was tested on a Micro-semi RTG4 FPGA. It was configured as a four-port router with an internal clock frequency set to 50 MHz.

Besides an adequate control logic within the router, there are two additional important characteristic parameters of a router. A router delay and router propagation delay or packet router latency. The latter is, of course, dependent on the packet length. The router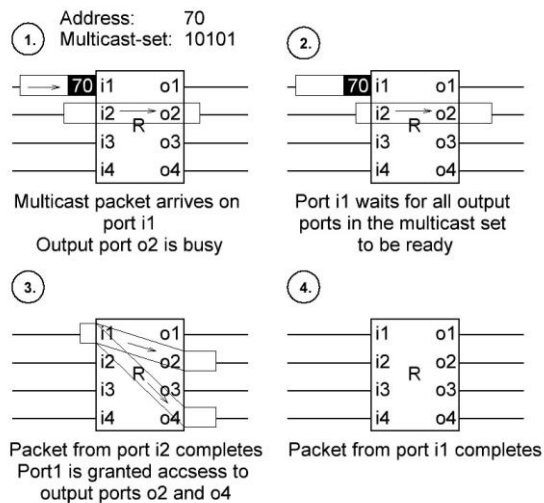 delay indicates how long does the router need for processing an incoming packet and determining its destination. Whereas, the packet router latency determines the time a packet needs to traverse through the router.

Test setup for measuring the packet router latency included NI PXI system with RMAP SpaceWire expansion card from STAR-Dundee and an RTG4 development kit with FMC SpW/SpFi card also from STAR-Dundee. The PXI system was running a LabVIEW program for two source nodes, logging, and a destination node. Both sources were configured to sand the same size packets to the same destination to demonstrate the influence of an output port contention alongside the packet router latency.

Preliminary tests were done on a Windows machine where the execution of tasks cannot be precisely controlled. It turned out that for traffic with a packet length of 2kB, 3kB, and 4kB, the packets were transmitted just right that when a packet from the first source completed transmission, a packet from the second source arrived at the router. Therefore, it did not experience any additional latency, except its own. In other cases, the measurements were as expected, as can be seen in Fig. 10. In continuous

traffic, for this scenario, sources interchangeably wait on each other.
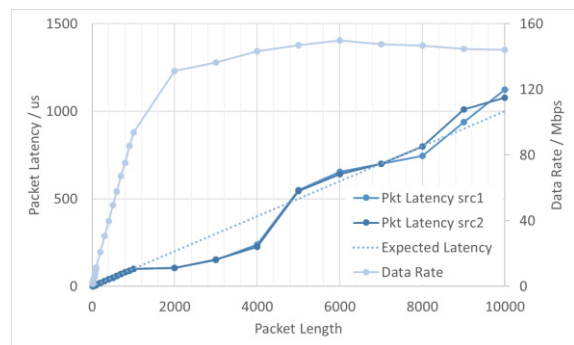
Fig.10.Packet router latency [9].

The router delay was determined by measuring the time difference between two trigger signals, which correspond to the events when the start of a packet was received at an input port and later at an output port. The router delay is different for each addressing mechanism. Fig. 11 depicts the measurement for packets using path addressing. The router delay is equal to 7 clock cycles, which results in a router delay of 140 ns at a 50 MHz clock.
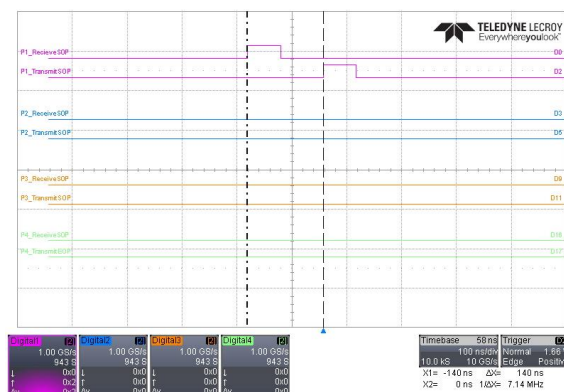
Fig.11.Router delay in case of path addressing [9].

Fig. 12 depicts the measurement for packets using logical addressing. In this case, the router delay is larger. As expected, since there are more processing steps in the packet processor, besides that, access to the routing table takes additional clock cycles. Altogether it takes 12 clock cycles to process a logically addressed packet, which results in a router delay of 240 ns at a 50 MHz clock. In case when multiple logically addressed packets arrive at the same time. The first packet is delayed for 12 clock cycles. However, subsequent packets from other input ports are only delayed for an additional 6 cycles, which is the time needed for the routing table access and an issued grant signal from the control logic. It is not the full 12 clock cycles because packets are already partly processed, and the input ports are just waiting for the routing table access.
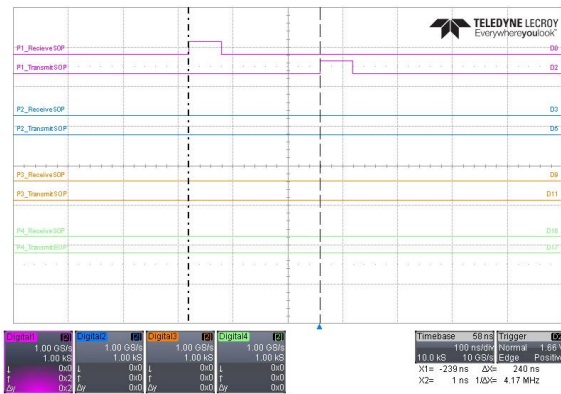
Fig.12. Router delay in case of logical addressing [9].

Fig. 13 and Fig. 14 give an overview of individual SpW router subcomponents resource usage compared to the resources used by the entire SpW router. The most significant component regarding the usage of Look Up Tables (LUTs) is the control logic (CL), followed by SpW ports (SpWP), and the configuration port (CP) (see Fig. 13). The size of the other components is negligible.

Fig.13. SpW router components LUT usage.

The most significant component regarding the usage of D Flip-Flops (DFFs) are the SpW ports, followed by the control logic, configuration port, routing table arbiter (RTA), and routing table (RT) (see Fig. 14). The RTG4 FPGA has 151824 LUTs and DFFs. Overall resource usage for a four-port SpW router was 13642 LUTs (8,99 %) and 5173 DFFs (3,41 %).
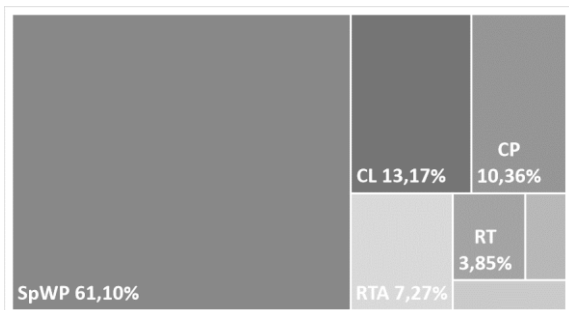
Fig.14. SpW router components DFF usage.

A couple of tests were performed to demonstrate the functionality of the control logic. The test setup was the same as for the packet router latency test. Though, for this test, all four

SpW router ports were utilized. Fig. 15. depicts a snapshot of packet traffic where all four sources were transmitting packets with the same priority to the same output port. Packets follow the round-robin patter as it expected for output port contention among packets with the same priority.
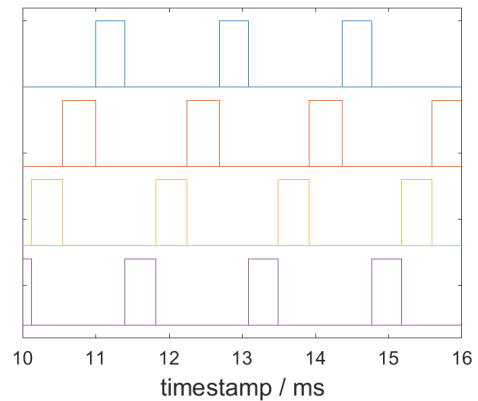
Fig.15. Snapshot of packet traffic, all packets have the same priority.

In another test, the last source was transmitting packets with a higher priority than all other sources, which were transmitting packets with the same lower priority. However, still through the same output port in order to demonstrate that a higher priority packet is granted access as soon as the previous transmitting packet completes its transmission. Snapshot of such traffic is depicted in Fig. 16. Black vertical lines indicate when a high priority packet arrived. The high priority packet is granted access to the output port as soon as the currently transmitting packet completes its transmission. Upper three sources in Fig. 16 still follow the round-robin pattern.
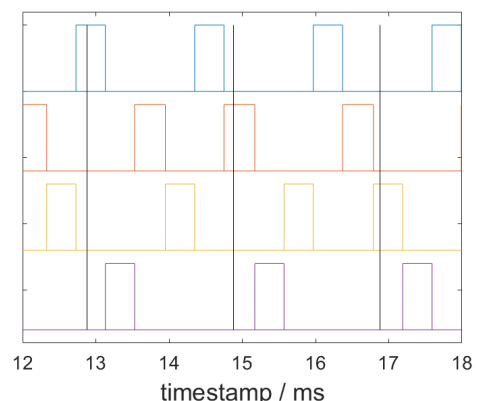
Fig.16. Snapshot of packet traffic, packets from the last source have a higher priority than the others.

Conclusion

Due to missing multicast and multilevel priority features in currently available devices, a design of an intelligent SpW router is investigated.

Essential aspects of the SpaceWire standard, which are relevant for the SpW router operation, are presented in the scope of this paper.

Description of the implemented SpW router IP-core is provided in the form of sperate components, determining their purpose and functionality. The focus is on the control logic as it provides the brains of the router and it is implicitly responsible for the flow of packets through the router.

Though, one crucial aspect of the SpW router, or rather the SpW CODEC, which can pose a serious implementation challenge, is not described. It is a SpW clock recovery circuit in the SpW CODEC [10].

Preliminary tests show that the SpW router meets all defined design goals. Two important characteristic parameters are defined, a router delay, and packet router latency. The control logic arbitration unit follows the implemented algorithm. Though, it is hard to showcase the multicast traffic with the current test setup.

New test setup and procedures are in work for evaluating the individual SpW router performance as well as performance on a system (network) level. Including deterministic sources, better network traffic logging, more extensive network, better visualization, and more.

References

[1] N. J. Boden, D. Cohen, R. E. Felderman, A. E. Kulawik, C. L. Seitz, J. N. Seizovic, and W.-K. Su, "Myrinet: A Gigabit-per-Second Local Area Network", IEEE Micro, vol. 15, pp. 29–36, (1995); doi: 10.1109/40.342015

[2] ECSS-E-ST-50-12C-Rev1 SpaceWire - Links, Nodes, Routers and Networks, ESA Requirements and Standards Division Std., Rev. 1.0, (May 2019)

[3] ECSS-E-ST-50-52C SpaceWire - Remote Memory Access Protocol, ESA Requirements and Standards Division Std., (Februray 2010)

[4] B. Chemli and A. Zitouni, "A Turn Model Based Router Design for 3D Network on Chip," p. 8, (2014); doi: 10.5829/idosi.wasj.2014.32.08.1254

[5] M. Dridi, S. Rubini, M. Lallali, M. J. S. Florez, F. Singhoff, and J.-P. Diguet, "DAS: An Efficient NoC Router for Mixed-Criticality Real-Time Systems," in 2017 IEEE International Conference on Computer Design (ICCD). Boston, MA: IEEE, (November 2017), pp. 229-232; doi: 10.1109/ICCD.2017.42

[6] J. Duato, S. Yalamanchili, and L. M. Ni, Interconnection Networks: An Engineering Approach. San Francisco, CA: Morgan Kaufmann, (2003); ISBN: 1-55860-852-4

[7] SpaceWireRouterIP 6PortVersion. shimafujigit. Accessed 28.8.2019. [Online]. Available: https://github.com/shimafujigit/ SpaceWireRouterIP 6PortVersion, (May 2019)

[8] B. Yuce, H. F. Ugurdag, S. Gö¨ren, and G. Du¨ndar, "Fast and Efficient Circuit Topologies for Finding the Maximum of n k-Bit Numbers," IEEE Transactions on Computers, vol. 63, no. 8, pp. 1868–1881, (August 2014); doi: 10.1109/TC.2014.2315634

[9] G. Skvarč Božič, "Intelligent SpaceWire Router Design", Master's thesis, (2019)

[10] Implementing SpaceWire Clock and Data Recovery in RTG4 FPGAs Application Note, 3rd ed., Microsemi, (March 2018)

Real-Time Data Visualization in Telemetry Systems

M. Ufelek[1], O. Yoruk[2], M.E. Cakici[3]
Turkish Aerospace Inc., Fethiye mah. Havacilik Blv. No:16 Kahramankazan/Ankara, Turkey
[1]mustafa.ufelek@tai.com.tr, [2]oyoruk@tai.com.tr,[3]muhammedemir.cakici@tai.com.tr

Abstract

It's highly critical to acquire, store, evaluate and analyze accurate real-time telemetry data during a test flight. Flight Test Management Centers are designed and used for this purpose. However, the capacity of Flight Test Management Centers are limited and majority of design engineers could not attend the test flight. The need of an application that lets design engineers to monitor the ongoing flight test from their workspace has occurred. For that reason, a software has been developed for real-time data monitoring and extracting.

On kick off of this project, two main objectives are aimed. The first one was to develop a smart system, which helps design engineers to monitor critical flight parameters in real-time. The second one was to prevent human faults with pre-defined caution/error indicators and make predictions by using data-driven algorithms.

The first phase of the project has determined as to gather data from flowing telemetry stream simultaneously in real-time and to store critical parameters in a time series database. The second phase has determined as to visualize data in comprehensible and user-friendly dashboards by selecting proper widgets. The last phase includes developing data models by using machine learning algorithms to make predictions regarding to previously stored flight data.

Key words: Data Acquisition, Real-Time Data Visualization, Data Analysis, Flight Test Database, Big Data

The Idea Behind

As Mustafa Kemal Ataturk told 100 years ago, 'The Future is in the skies'. Aviation and Aerospace Industry has boomed in the last 100 years and it is still growing day by day. Requirements are changing and companies are designing new products to meet that requirements. Time is more important than anything else today. No matter how good the design is, the product could not succeed if it has not been served to the market on time. For that reason, the design should be transformed into the product very quickly.Testing activities are vital in aviation and aerospace industry, significant time is allocated to flight testing activities in the product development process. These flight tests should be accomplished very carefully, in a very short time period. There is a need to collect as much data as possible from the experimental aircraft and analyze this data accurately. Rapid data analysis allows project Flight Test Team to be ready for the next flight test. Flight Test Management Centers are designed to visualize real-time data received from the aircraft and manage the flight test according to this data. However, the capacities of flight test management centers are limited and designers who are not directly related with the ongoing flight test does not attend to the test. In many tests, design engineers also need to monitor real-time data; so the data should be available under authentication control outside of the test management centers. First aim is to extend the capacity with an in-house web-based software that allows engineers to monitor the data directly from their workspaces. Second aim is to store real-time data that is collected during the flight in a time-series database and use the stored data for further analy-

sis. Thus, while the data is visualized in real time, a flight test database will be created.

Telemetry data includes parameters that are critical for analyzing the design. The data must be stored in a database for design engineers, product teams and flight test engineers. The Flight test team should be able query this data with different criterias. For example, comparison of values for the same parameter on test flights in various weather conditions or test points is very important.

After obtaining a sufficient amount of data, predictions can be made by using machine learning algorithms. As the size of the flight test database grows, the data model built on it will work with higher accuracy. Therefore, storing and labeling parameter data in each new flight test is critical.

Fig. 1. A Flight Test Management Center, Turkish Aerospace Industries Inc.

Current Design of Telemetry Network

The flight test management center owns a seperated network called 'Telemetry Network'. Desktop applications (such as IADS) runs on client computers which are called test monitoring consoles. Flight test team can monitor the

flight ontest monitoring consoles. Every test monitoring console runs a customized flight test monitoring display which has designed uniquely according to the flight test team members requests.
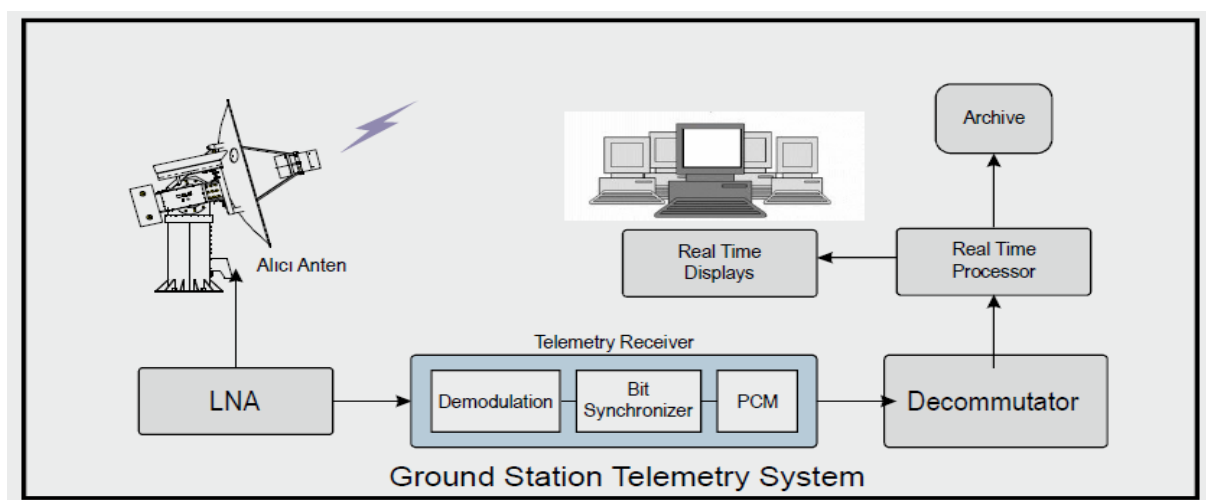
Fig. 2. Telemetry System Design

Extended Design

Turkish Aerospace has an internal network of approximately 10000 clients. This network is isolated from the Internet and telemetry network. Employees use computers on this network to carry out their jobs. Therefore, they need to access telemetry data through this network. The design which has shown on Fig. 2 has extended as shown in Fig. 3. Conversion of signals from telemetry is done by Decommutator. A software that fetchs data after this

transformation and inserts into database, has been developed. As shown in the Fig. 3, this software receives parameter data from the source in the telemetry network and inserts into a database in the company network where the employees' workspaces are located. The database is a time-series database and data is stored with its sampling rate. There is an analytics platform running in the server. This analytical platform provides real-time visualization of data recorded in the database.
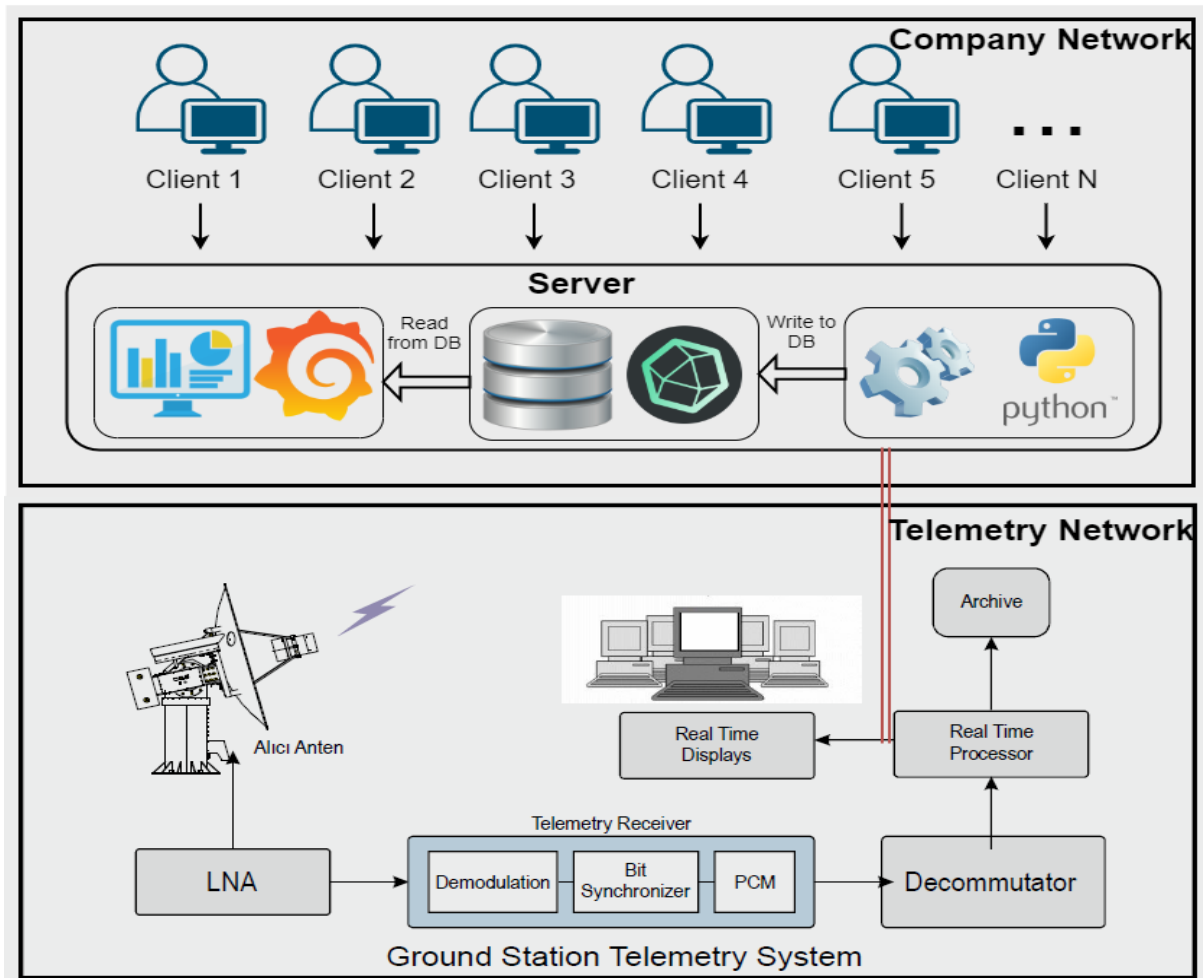
Fig. 3. Access data through Company Network

Phases of Solution and Techologies Used

Three stages had to be applied to implement the solution. In order to implement these processes, technologies commonly used in software world such as Python, Grafana, InfluxDB and Docker have been selected. Docker has been used to run container structures. Python programming language has been used to develop software that fetches real-time telemetry data and inserts into the database. Influxdb has been used to store data, and Grafana has been used to visualize stored data.

Fig. 4 Technologies used

Phase 1: Data Acquisition and Storage

The first phase was to develop an application that reads telemetry data in real time and stores into a time-series database. In telemetry network IADS CDS (Cache Data Server) is used to serve data to client desktops. In addition to this, IADS provides an API to get this data programmatically. This interface has been used to read real-time data. Through this interface, it is possible to get not only parameter data but also many additional data such as test point logs, event logs, parameter list, parameter calibration definitions, and mission attributes tables.

The software has been developed with the Python programming language. It is a desktop application that runs in a server in the telemetry network. Since the data rate is very high, the software has been developed to operate at maximum speed and it works with a large number of threads to carry out the operations. It is a desktop application without any visual components and works as a service. Parameter names can be configurable for different type of aircrafts. Information about which parameters will be written to the database in real-time is recorded in a configuration file and the software performs data reading and writing operations in accordance with this information.

For the selection of the database, the most important criteria was the speed of writing and reading. Planned writing and reading speed was designed to be one tenth of a second. This means that all selected parameters are both be written and read within 100ms. Even delays in the level of seconds can cause data to be misinterpreted. Since the data is processed on time based, this database must be a time-series. One of the most used time-series databases in the software community, Influxdb was chosen to meet that requirement. In addition to parameter data, the database contains metadata about the flights. Test point and mission attribute data are also stored in a relation in the database. Storing these data together makes it easier for flight test engineers to make detailed analyzes with accumulated data. At the end of the test flights, a telemetry-specific flight test database has been created and this database is accesible for all users to run complex queries.
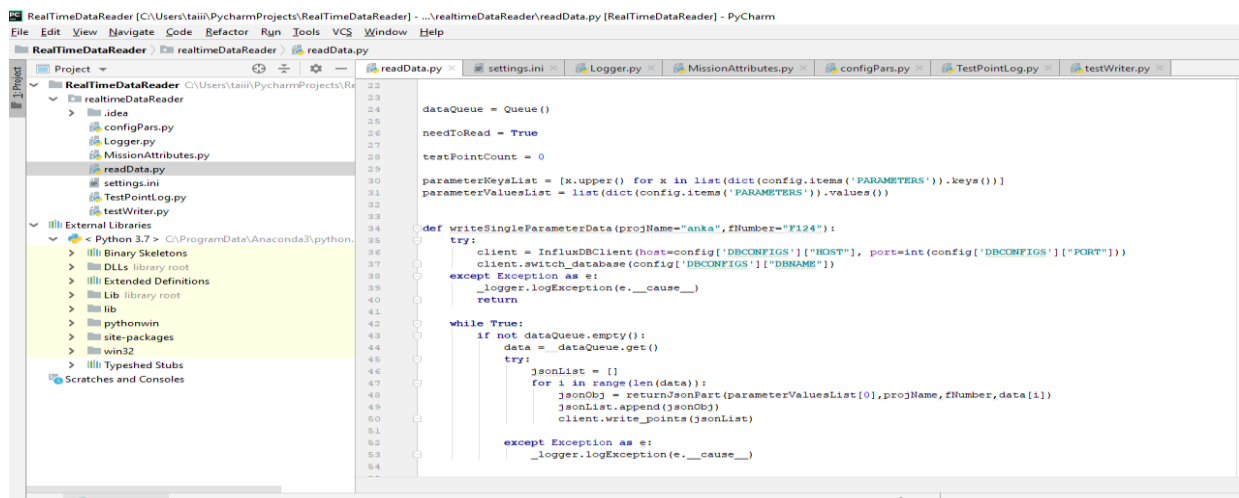

Fig. 5. Real-time Data Reader and Writer Application

Fig. 6. Sample Parameter Data

Phase 2 : Visualization

Visualization of the data via using proper widgets is as important as acquiring it. Stored data should be converted to visuals at high speed and this visuals should be easy to understand and interpret. Considering the speed of writing data to the database, it should be displayed on the dashboard with a delay of up to one second. Since the data is stored in a time-series database, the visualization technology has to support this type of database.

As the data visualization application will be the gateway to data access and will operate on the same network with all internal users, authorization control is very important. Only authorized personnel should be able to access the dashboards to be developed. Therefore, a solution with a strong security structure and in accordance with the authorization policy used in the network should be chosen. It should also be customizable and extensible. Although there are some differences according to the technology used, generally there are standard visual sets. These visual sets may not meet the requirement when the working area is very specific. Since aviation and aerospace is a very specific field, it cannot be expected that the products on the market to contain the visual sets used in this field. Therefore, it should be open to custom development. Grafana is the right choice from this point of view. As many custom plugins can be found and used, it is not difficult to develop a new custom plugin. As a matter of

fact, some of the visuals used in the dashboards are customized for custom use within the company. The infrastructures used in this phase were preferred to be open source. Community support is very important on these open source platforms. This criterion was also taken into consideration during the selection.

Fig. 7. Artificial Horizon-Community Plugin extended
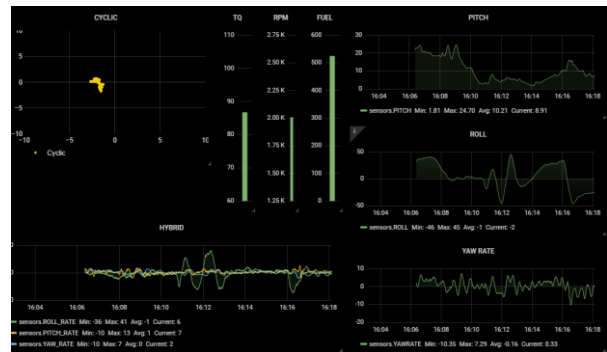
Fig.8 Sample Visuals from Flight Test Dashboard
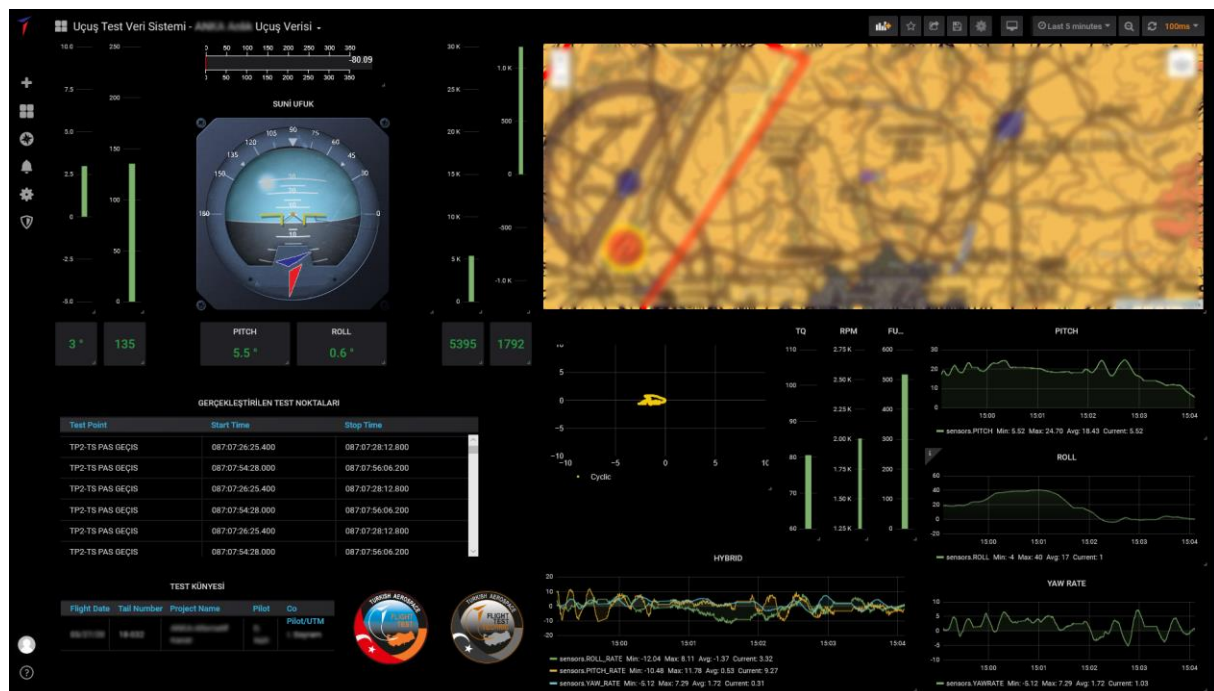
Fig 9. Flight Test Dashboard

Phase 3: Post-Flight Further Analysis

It was mentioned that data is being stored into a time-series database while visualization process. Thus, a database where post-flight analysis can be done after the flight was obtained. Storing flight data with this method helps to analyze the behavior of one or more parameters between different flights, test points and conditions such weather, aircraft configuration etc. Furthermore, important flight test data that is acquired can be transformed into more valuable way by using machine learning methods. One of the most important objectives of flight tests is to detect abnormalities on the aircraft. These abnormalities are very prone to faults when attempted to be detected by the human eye. Data models fed with sufficient data are very successful in detecting these abnormalities. Machine learning techniques are used to detect these abnormalities. Both of the two most important types of machine learning methods, supervised and unsupervised learning, can be used.

Supervised Machine Learning

In case the inputs and the outputs resulting from these inputs are known, it is called supervised machine learning. So there is a function like **f (x) = Y.**

Here, x is input and Y is output. The aim is to predict Y value based on previous data for new x values. The data should be tagged here. For example, the engine failure that occurs in a certain combination of torque, engine speed and oil temperature values should be labeled. Thus, the data model will evaluate these parameters and predict whether an engine failure will occur.

Unsupervised Machine Learning

Unsupervised machine learning is one of the widely used methods. It allows for anomaly detection by clustering method in cases where Y result is unknown corresponding to x inputs. What is done here is to group each parameter by evaluating the values of previous data, even if Y is unknown, and to identify values outside this set.

Hundreds and even thousands of parameters are collected at high frequencies according to the measurement type during a test flight. It is almost impossible for a human to evaluate such this amount of data instantly and make right decisions. At this point, where human abilities are limited, using computer-aided methods will increase the safety and effectiveness of test flight. In order to use these methods, a flight test database is required first. As mentioned in the first part of the article, while the data is visualized, it is also recorded into the database. By

using this database, the data model is trained which values are normal under different conditions for each parameter. To give a simple example, when the engine speed is 3000 rpm and it is known that the oil temperature should be between 80 to 90 degrees celsius, it can be considered as an anomaly when this temperature rises above 90 degrees. Although this is easily detectable by a human and can be resolved with a simple warning mechanism, more complex techniques are needed considering the mentioned data. In this example, an inference was made by evaluating only two parameter and one rule. Although it varies depending on the type of aircraft, thousands of parameters from hundreds of sensors in an aircraft can be collected at the frequency of a few kilo hertz. In a dataset of this size, it is impossible to evaluate all parameters that may be related to each other instantly. Moreover, while the relationship between engine speed and oil temperature is strictly known in the given example, thousands of parameters in the same dataset could be related to each other. Therefore, a solution has been developed to detect abnormalities.
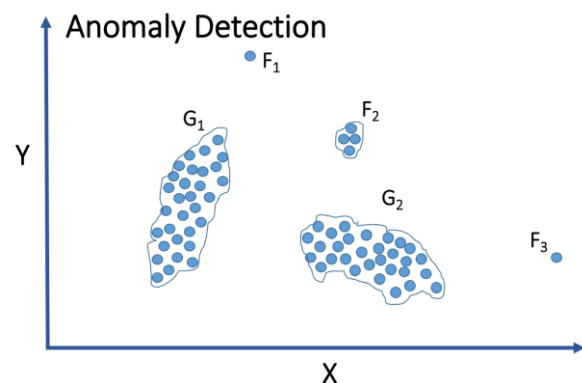
Fig 10. Anomaly Detection (Clustering)

Conclusion

The visualization and evaluation of the data collected in flight tests with analytical systems has a great importance for the schedule, cost and success of the project. With the completion of this software, it is predicted that the flight test process carried out within the company will be accelerated.

Simulation tools to assess the impact of Wind turbines on RADAR and other electromagnetic signals

Karen von Hünerbein[1], Nicolas Douchin[2], Werner Lange[1]
1. Lange Electronic GmbH, Rudolf-Diesel-Str. 29A, D-82216 Gernlinden
kvh@lange-electronic.com
2. Oktal-Synthetic Environment, 11 Avenue du Lac,
31320 Vigoulet-Auzil, Frankreich
nicolas.douchin@oktal-se.fr

Abstract

In recent years the amount of wind energy power plants has risen dramatically, driven by the urgent need to increase the percentage of energy production from renewable, eco-friendly sources. Thus, more and more on-shore and off-shore wind turbine fields have been installed. While this increase is very desirable from an ecological climate perspective there are also side-effects as noise levels, a threat to birds' lives and impact on the propagation of electromagnetic waves, such as RADAR signals, air traffic navigation signals, and frequencies used for data transmissions, e.g. from UAVs (Unmanned Aerial Vehicles) to the ground and back. Air traffic control RADARs, for example, need to be able to detect and track all aircraft near the airport, especially those flying at low altitude.

Detrimental effects include shadowing effects and scattering of signals, as well as increase in EM noise due to scattering, leading to degradation and even loss of the wanted signal, handicapping the data transmission between aircraft and ground stations and detection of flying objects by ground-based RADAR.
In order to be able to assess the wind turbines' impact and degradation of the signals, sophisticated computer simulation tools have been developed [1] which perform physically correct calculations of signal propagation, including geometrical and physical optics [12], with relation to a high resolution 3D environment accurately representing both the terrain including all man-made objects such as the wind turbines and the moving land and air vehicles.
In this presentation we will explain the concept and the background of one such simulation tool and present images and videos of UAV with and without interference caused by wind turbines. We will also show examples of the impact on wind turbines on Signals at 433 and 915 MHz, as used by UAVs for telemetry.

Keywords

Simulation, RADAR signals, telemetry signals, shadowing, interference, security

Introduction

In order to produce climate friendly electricity, more and more wind energy power plants and wind farms have been built in recent years on-shore and off-shore. German energy produced has increased from 105 TWh (Tera Watt hours) in 2017 to 131.8 TWh in 2019, with 29.456 installed wind turbines on-shore, in Germany alone [2]. Wind turbines represent large structures, with static and moving parts, and varying heights, some of them as much as 160-200 m high. Masts consist of steel, and sometimes of concrete. Hubs and the blades consist of glass fibre reinforced carbon and or metal.

In Germany, many wind farms, which are fields of multiple wind turbines, are distributed across the country side, which need a minimum distance of 1-15 km to the next airport according to German regulations, a distance, which is different in different states of Germany. No new wind turbines and farms can be built in a radius of 15 km around a D/VOR navigation station [7]. All new wind turbines and farms need an official permission by the German aviation authority (Luftfahrtbundesamt), before installation, to avoid dangerous Electromagnetic interference and shadowing, as well as mechanical interference with low flying aircraft during take-off and landing. Currently there are around 2.100 wind energy power installations near airports [8].

There is are huge amount of evidence that wind turbines and wind farms can interfere with RADAR signals of ground based RADAR, e.g. for airport surveillance RADARs and air traffic control. Effects on RADAR signals include shadowing and masking, backward clutter, meaning unwanted return signals from wind turbines, distortion and fading of signals. [3, 4, 5, 6]. Shadowing means that the wind turbine is large enough to block the return signal from an aircraft. Masking means that the wind turbines act as obstacles. Backward clutter is caused by the reflections from the different parts of the wind turbines and adds enough noise to the return signals to make detection of the wanted target signal difficult or impossible. There are similar effects on a variety of aviation navigation signals (VOR, DME, TACAN), communication signals (VHF, UHF) and airport landing systems, such as ILS and MLS [5].

The effects are more pronounced for low-flying aircraft, e.g. during landing approach or take off, especially near airports impairing the return of the RADAR signals of ground based airport surveillance RADARs [4]. As these effects and unwanted additional signals can endanger normal flight operations near and at the airport, there are regulations in different countries, prescribing minimal distance of wind farms to the airport.

Civil UAVs generally fly at lower altitudes and rely on remote control by ground based pilot (uplink) and transfer data back to user on the ground (downlink) on standard telemetry bands such as 395 MHz, 420-450 MHz or 902 – 928 MHz. In these cases, wind turbines can also block or partly block line of sight between the transmitter and the receiver, and add noise and unwanted additional signals to the transmission.

Simulating the effects on the signal propagation of RADAR, telemetry and other EM signals prior to building and installing new wind farm and/or new RADAR and navigation sites, can help to avoid interference and technical problems in the field. The goal is to assess the impact of Wind Turbines on signal propagation of RADAR signals and other RF signals, before installation and operation of new wind turbines and wind farms close to airports, aviation navigation aids and all sorts of RADARs. Another goal is to improve signal detection and processing algorithms both in RADARs and in on-board RF receivers or UAVs and aircraft for telemetry and communication signals, and thus save time and cost in the end.

In order to achieve these goals, a computation tool with graphical output has been developed. It combines the methods of 3D geospecific terrain modelling, simulation of the propagation of electromagnetic signals with raytracing and RADAR modeling [1,12]. In this paper we will introduce this tool, and show examples of simulations for signals of a ground based RADAR for UAVs and

airplanes. We will also show examples of the impact on wind turbines on Signals at 433 and 915 MHz, as used by UAVs for telemetry.

Signal Propagation Modeling Tool

The computation tool consists of two parts: a signal propagation module and a RADAR modeling module. The RADAR modeling is usually performed by the RADAR manufacturer and is included into the tool as a plugin. The signal propagation module is publicly available and combines the methods of 3D geo-specific terrain modeling, 3D wind turbine and wind farm modeling and the simulation of the propagation of electromagnetic signals with raytracing and physical optics. The tool was developed by Oktal-SE in cooperation with French research institutes and is called SE-Workbench-RF.

In the first step a 3D environment is modeled by importing satellite imagery, digital elevation models and different static and moving objects, such as buildings, trees, UAV or ground vehicles, and the RADAR emitter itself. The 3D model consists of millions of polygons and is made to be as representative of the geographical landscape as possible. It can be very large, e.g. 130 * 130 km, and takes into account the curvature of the Earth resulting in a correct RADAR horizon. Each of the polygons is classified by assigning a physical material to it, with the electromagnetic properties from the SE-WORKBENCH materials database, e.g. reflectivity and back scattering coefficient.

EM signal propagation is calculated in a deterministic way with the following methods:
- geometrical optics
- physical optics
- edge diffraction ECM
- monostatic RCS (Radar Cross Section) computation
- bistatic RCS computation for each reception point

The solution considered here has been developed jointly by OKTAL-SE and ONERA and is dedicated to the simulation of raw data of large-scale environments including complex targets for several kinds of RADAR.

State of the art

Computations are based on an electromagnetic simulation code based on ray tracing and asymptotic methods [9]. These methods are less physically rigorous than "exact" methods that are strictly based on the resolution of Maxwell equations. However, asymptotic methods enable to handle complex scenes that are very large compared to the wavelength with enough accuracy. In such situations, asymptotic methods provide main contributions of the electromagnetic field which are correctly represented.

In our solution, rays are traced from transmitters towards reception points. These rays are grouped four by four in beams. Rays are traced from transmitters through a grid (figure 1). The intersections of theses beams are computed. There are two types of interactions (figure 2):
- Geometrical Optics (GO), when the beam is reflected by a metallic or dielectric surface
- Physical Optics (PO), when part of the beam energy is scattered towards the reception points at each interaction.
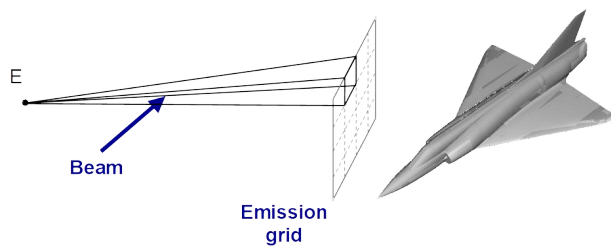
Figure 1. SE-RAY-EM emission grid
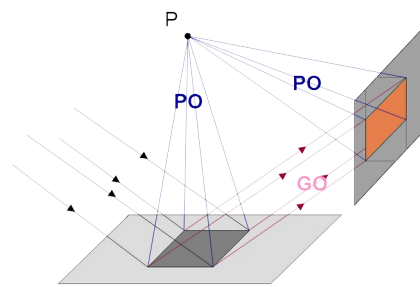
Figure 2. Principle of beam interactions

The ray tracing based on shooting and bouncing ray technique is used to identify the EM contributors in the scene. In the very simple case illustrated by the figure here after (fig. 3) composed by two flat polygons and a unique ray tube, two EM contributors are identified. Then the amplitude and phase of the EM fields scattered to the receiving point by each EM contributor are computed using PO.
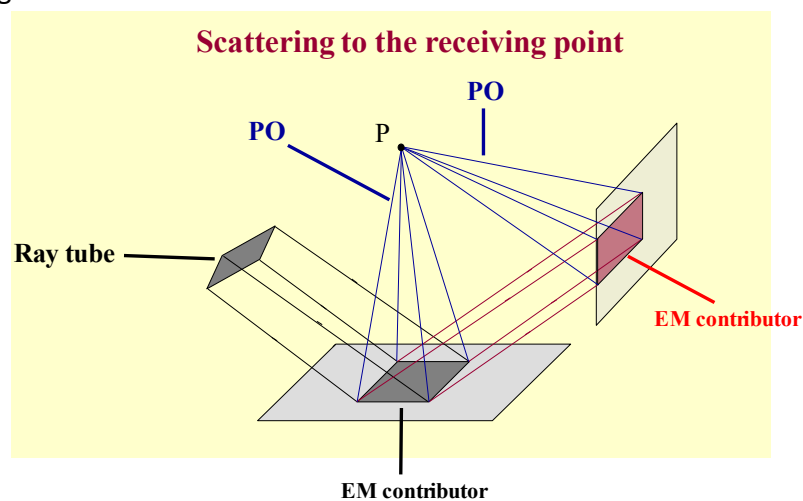
Figure 3. Concept of EM contributor

Validation is achieved through comparison of our solution with other codes based on different methods: typically, an ONERA code based on the Method of Moments to solve the integral equation, called MAXWELL3D [10]. Validation is also based on comparison between simulation and measurements, in cooperation with Fraunhofer FHR for example, as illustrated hereafter (fig. 4):
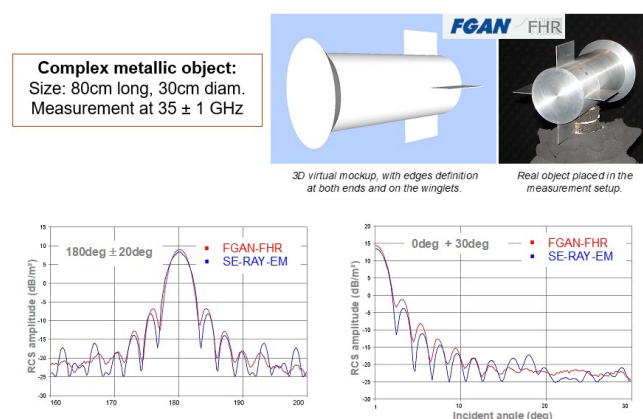
Figure 4. Example of validation test

Reduction of the computation time through GPGPU implementation

Nowadays, Graphic Processors Unit (GPU) have proven to be very efficient for optimizing General Purpose (GP) computations, and particularly ray tracing applications [11]. However, the transition to GPU is not straightforward and several issues have to be taken into account.

Double versus Float

Since GPU are designed to be very efficient for single precision floating point (float) computation, our GPGPU implementation only uses double when it is absolutely necessary, mostly in the computations that involve the phase of the electromagnetic signal.

Cone tracing

Instead of tracing individual rays, we use cones to trace beams. This way of anti-aliasing is more reliable and the initial resolution can be really coarse. Another advantage of using cones is that they can be processed independently of their neighbors, which is very interesting for parallelization.

Multiple frequency cases

Multi-frequencies computations consist in computing several frequencies at the same time. Frequencies are independent in terms of electromagnetic computations, but share all the geometric computations. In our GPGPU version, frequency computations are postponed as long as possible. The anti-aliasing process generates all geometrical contributors. These contributors are then used multiple times for applying EM models, one time per frequency.

Validation of the GPU implementation

The GPU implementation is tested versus the CPU one in terms of both accuracy and performances.

The first test consists in comparing qualitatively the results between the new implementation and both the standard version of SE-RAY-EM and the reference from the MAXWELL3D software of ONERA. The test consists in the computation of the Radar Cross Section (RCS) computation of an aircraft for a frequency of 600 MHz (as represented in figure 5). Figure 6 shows that globally results computed by the three codes are very similar.
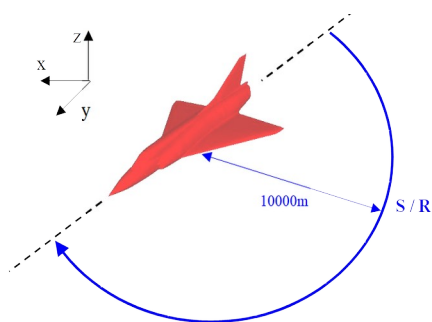
Figure 5. Aicraft RCS computation setup
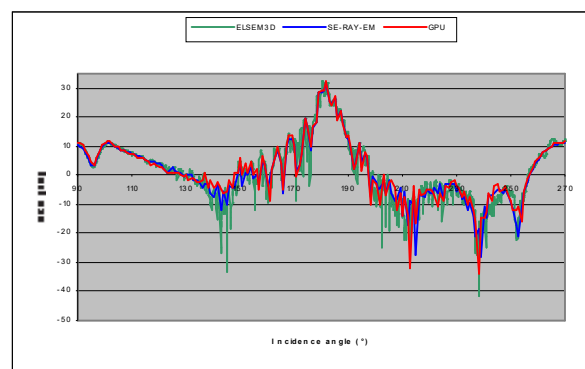
Figure 6. RCS of an aircraft computed with ELSEM3D (green), SE-RAY-EM (blue) and GPU version (red)

The GPU implementation is 10 to 100 times faster than the standard SE RAY-EM version, depending on the capacities of the graphic board as illustrated in [12].

Physical modelling of the environment

Any entity of the scene (terrain, wind turbines, mobile entities …) is considered through a geometrical model (e.g. external surface of a vehicle) and then one or several physical materials are assigned to this geometrical model.

In the simulation presented hereafter, the wind turbines (WTs) are supposed to be entirely metallic which is a worst-case compared to the real life. It means that the physical material assigned to the WTs is a Pefect Electric Conductor (PEC) material.

Concerning the terrain, texture classification is used to efficiently assign physical materials to the 3D model. Textures derived from measured aerial or space photos are mapped on geometries to improve the quality of the appearance of the 3D environment, but also to assign physical materials to the terrain by associating different materials to different colours identified in the image: this is what we call texture classification.

Scenario edition

Entities of the environment are gathered to compose a virtual scene, shooting conditions are defined, trajectories of mobile entities are created, temporal behaviours and events are handled and the scenario edition tool also enables to interactively visualize the scenario while running.

Scene rendering

For each sensor defined in the scenario, the physical signal received by the sensor is computed at a given date or over a time interval using realistic rendering (priority put on the precision of the computed signal). The scene rendering consists in computing the EM signal in range gates or as a function of angle of arrival or speed (Doppler shift).

The main advantage of using so complex 3d databases in RADAR simulation is that the real direct interaction of the incident EM wave and the 3d scene can be simply taken into account. This enables one to treat in the same run the target and the background, which is very interesting to naturally deal with the complex interaction of the target with its background.

Special focus on the modelling of WT

The WT 3D model is decomposed in several parts which are:
- The mast in blue (see the picture hereafter of the WT 3D model, figure 8)
- The fixed part of the hub in yellow
- The rotating part of the hub in green
- The blades in red.
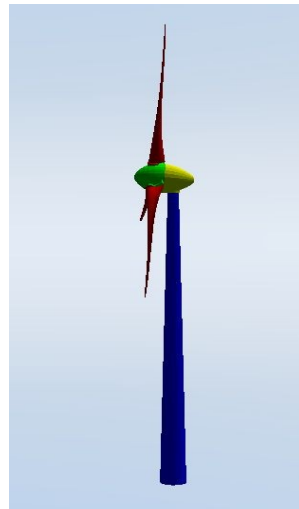
Figure 8. Decomposition of the 3D model of the WT

This decomposition allows one to define rotation for orienting the plane of the blades facing the wind and the make the blades rotating during the simulation. For defining the rotation of the blades Python scripting is used as illustrated on figure 9 hereafter.
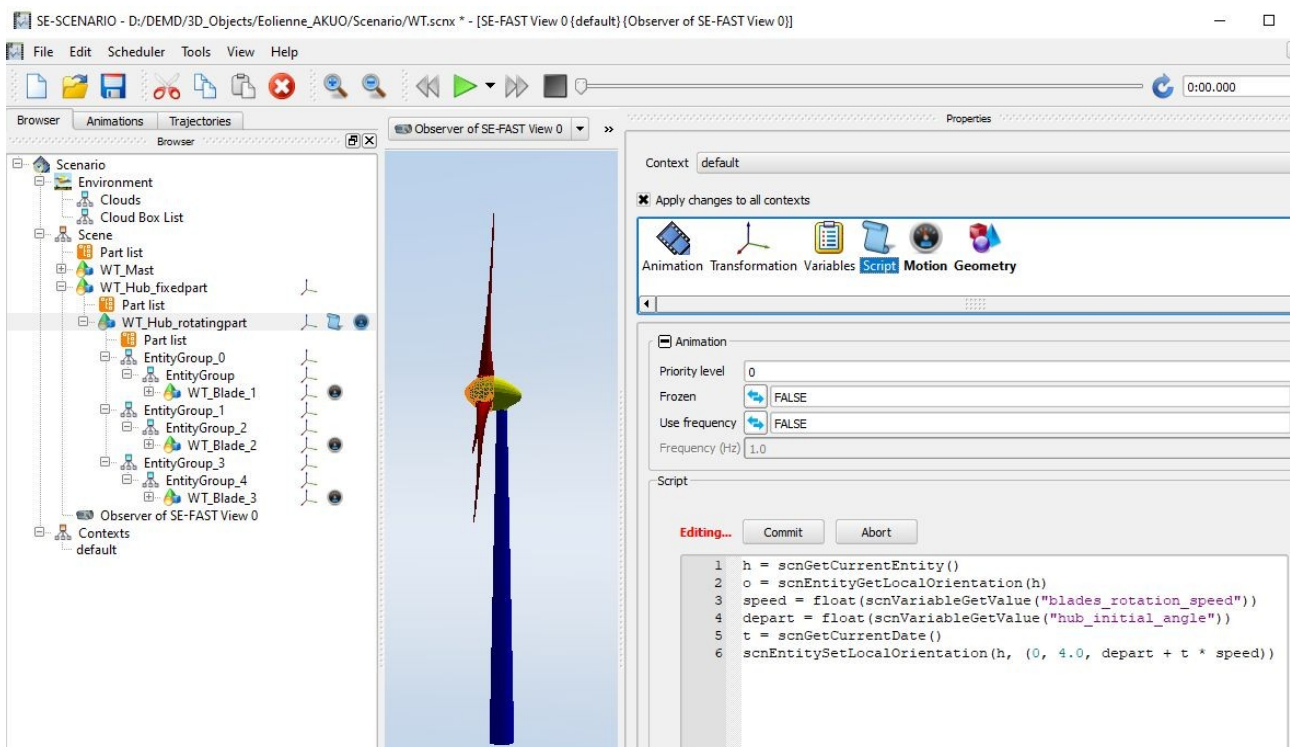
Figure 9. Python script used to define the rotation of the blades

Wind directions considered in the simulations

In the simulation results presented hereafter, two wind directions have been considered:
- Wind blowing from the West (WD = -30°)
- Wind blowing from the South (WD = -120°).

The orientation of the 12 WTs of the wind farm considered in this work is adapted accordingly as illustrated hereafter (figures 10 and 11):

Figure 10. Wind blowing from the West

Figure 11. Wind blowing from the South

Scenario description

The scenario considered in this work is composed by (see figure 12 hereafter):
- the environment (hilly terrain) close to Lyon Saint-Exupery airport
- A wind farm made of 12 WTs put on a hill
- A target which is a drone (quadrocopter) flying along a trajectory at 25m/s speed
- A RADAR (in blue color) installed in the vicinity of the airport, dedicated to air surveillance.
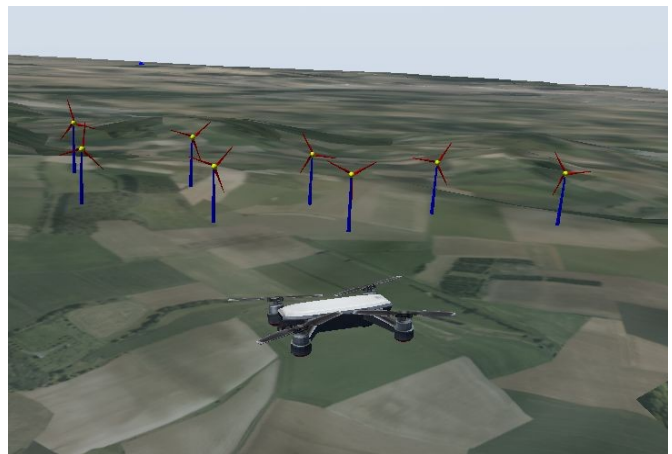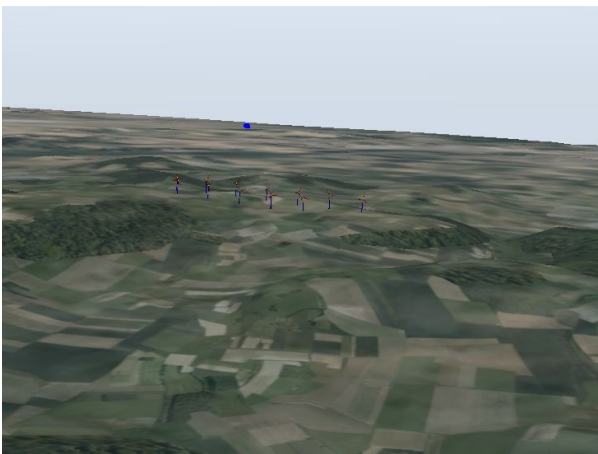
Figure 12. Views of the scenario (the RADAR corresponds to the small blue box in the background)

Time-Doppler computation in monostatic configuration

The results presented hereafter (Figures 13 to 16) were derived from the computation of the RADAR echoes generated by the whole dynamic scene considering the Doppler shift due to the rotation of the blades of the WTs and the flying target, as function of time. The time sampling is 100Hz and the radar carrier frequency is 5.6GHz.
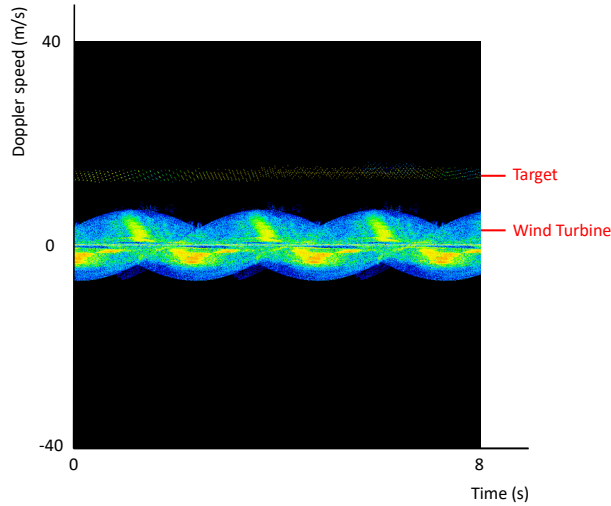
Figure 13. One WT and WD = -30° (from West)
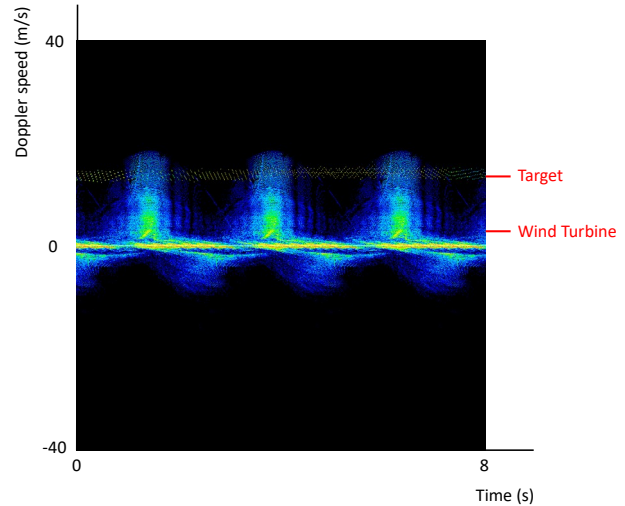
Figure 14. One WT and WD = -120° (from South)
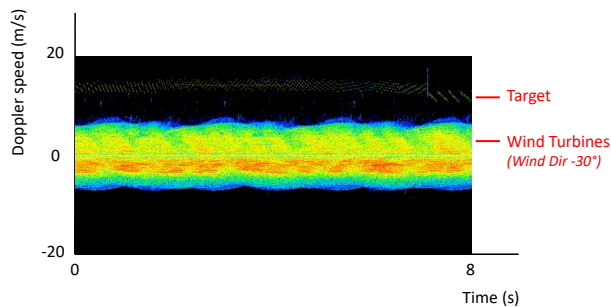
Figure 15. 12 WTs and WD = -30° (from West)
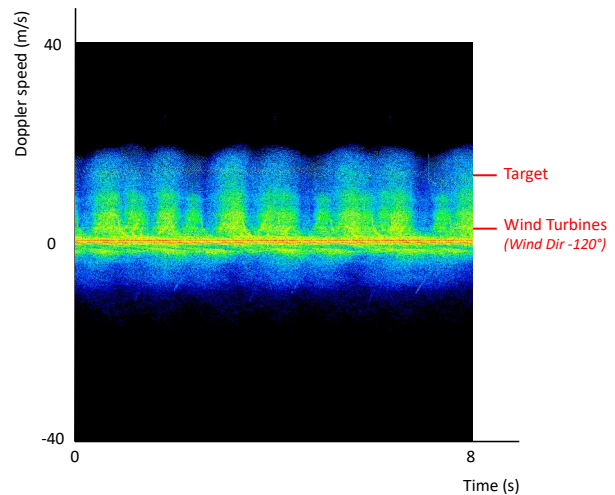
Figure 16. 12 WTs and WD = -120° (from South)

It can be observed that the RADAR echoes generated by the WTs is stronger when the blades facing the radar, meaning when wind is blowing from the West. But the Doppler shift due to the rotation of the blades is higher when the blades plane is parallel to the RADAR Line-Of-Sight, meaning when wind blowing from the South. In this situation, it is trickier to detect the target since its Doppler shift is the same range as the Doppler shift of the rotating blades.

It can also be observed that the Doppler signature of the wind farm is much more complex than the one of a single wind turbine.

Range-Doppler computation in monostatic configuration

Results presented hereafter (Figures 17 and 18) were computed in the same conditions as before.
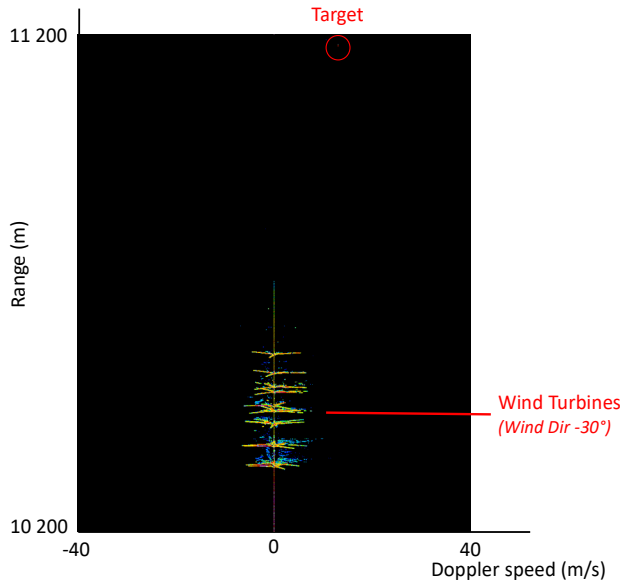
Figure 17. 12 WTs and WD = -30° (from West)
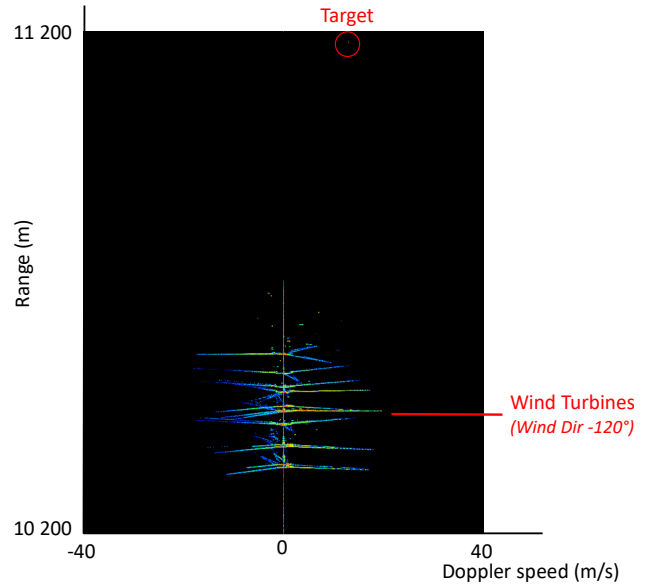
Figure 18. 12 WTs and WD = -120° (from West)

Once again, it can be observed that the Doppler shift of the WTs is larger when wind blowing from the South, meaning the blades plane parallel to the radar LOS.

In this Range-Doppler analysis detection of the target is easier in both wind conditions since its range is quite different from the WTs range domain.

Received power computation in bistatic configuration

The results presented hereafter (Figure 21 to 23) were obtained in bistatic configuration, meaning the receiver is put on the drone target. Two different altitudes were considered for the drone target. At the lowest altitude the drone may be masked by the WTs which is not the case at the highest altitude (see figure 19 and 20 hereafter).

Figure 19. Drone flying at low altitude (100m)

Figure 20. Drone flying at higher altitude (200m)

To obtain the results presented hereafter, the so-called "Forward Scattering" model was used. It is based on the Equivalence Principle which consists in computing the currents induced by the incident wave on all illuminated surfaces. Then we compute the EM fields (scattered fields) radiated by those currents to the receiving point without any occlusion test. The final step is a coherent summation with those scattered EM fields with the fields generated by the wave radiated by the radar directly to the receiving point without any interaction. This approach implemented in our solution allows to compute the EM field in the shadow region of an obstacle. It has been validated with ONERA by comparison with MAXWELL 3D code based on Moment Method.
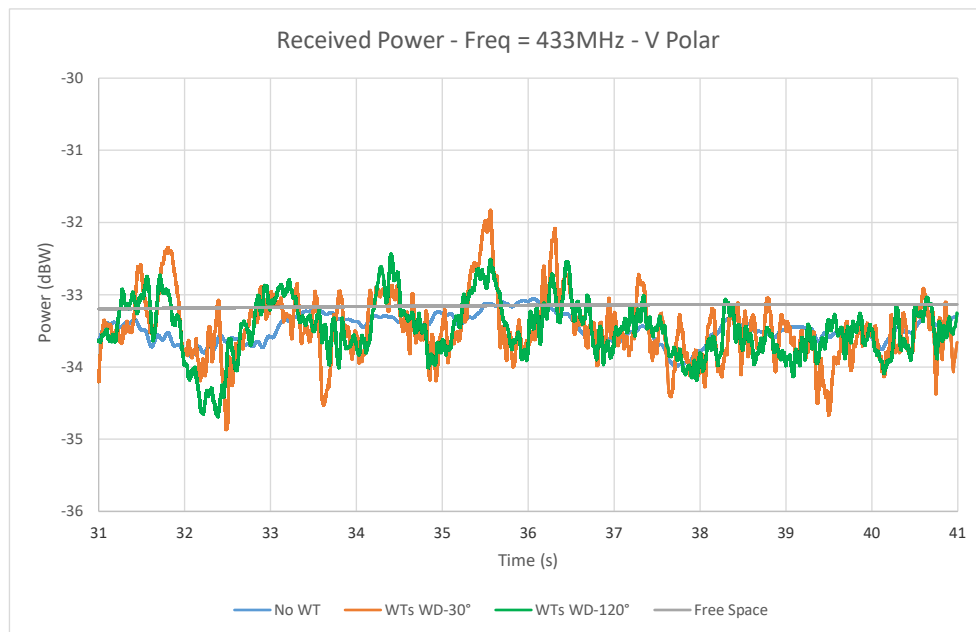
Figure 21. Scintillations at 433MHz – Drone flying altitude = 200m
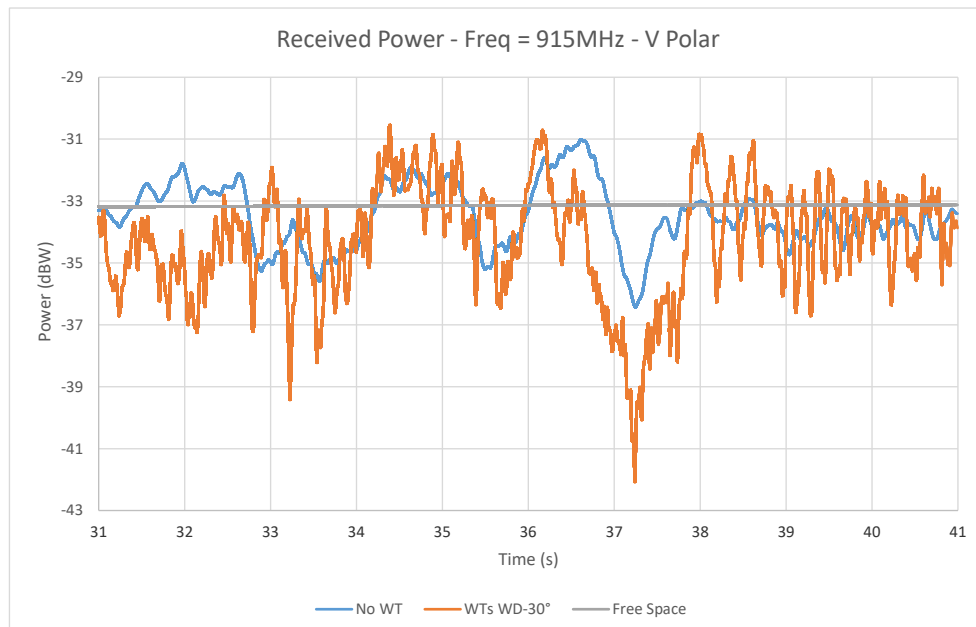
Figure 22. Scintillations and Shadowing at 915MHz – Drone flying altitude = 100m

Shadowing can be easily observed at time of the scenario just after 37s.
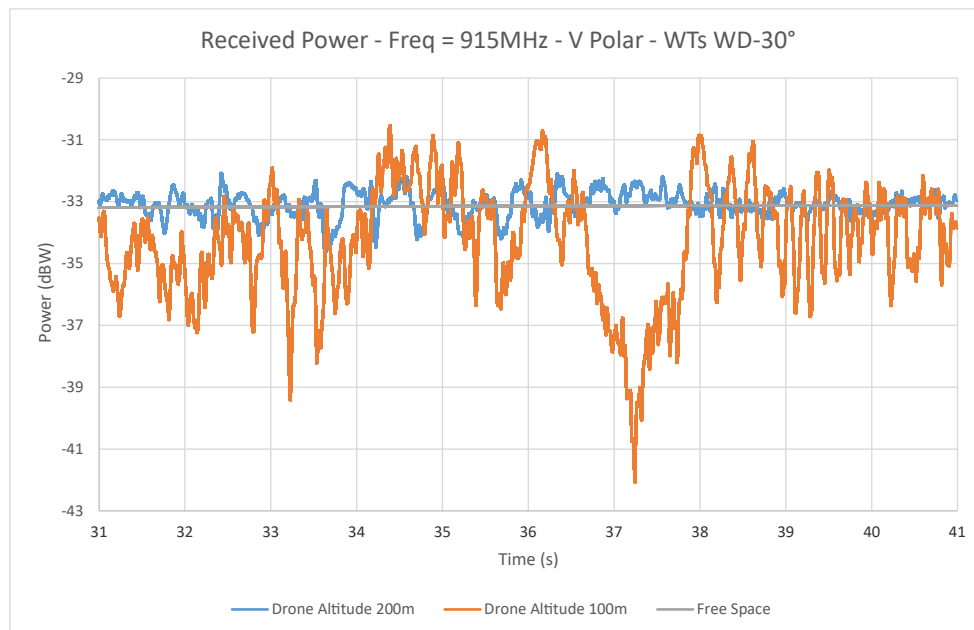
Figure 23. Scintillations and Shadowing at 915MHz

Scintillation is larger when the drone is flying at low altitude. Shadowing observed in low altitude flying situation doesn't occur when the drone is flying at an altitude higher than the wind farm as viewed from the radar (see figure 20).

Conclusions

High fidelity RADAR simulation is considered more and more as an alternative and complementary methods to real field measurements. Here we have shown a new simulation tool for efficiently simulating the signal propagation of RADAR signals in the vicinity of wind farms with a high computation speed. However, this tool needs to be further validated in different types of environment with different sensor configurations. Some complex simulations require further modeling, e.g. vegetation. The quality of the simulation depends heavily on the high fidelity of the input data, such as accurate and high fidelity synthetic 3D modeling, physical characterization of materials, and determining their EM parameters correctly, and modeling of the atmosphere close to reality.

Validation of the presented solution has been conducted in cooperation with ONERA in the frame of a project sponsored by the DGA (French MoD). Fitting between simulation and measurements is quite good but results are confidential and can't be presented in this paper.

References

[1] N. Douchin, C. Ruiz (2019) „Performant and High Fidelity Solution for Complex and Large Radar Scene Simulation Including Wind Turbines", EMWT 2019, Toulouse

[2] https://www.wind-energie.de/themen/zahlen-und-fakten/

[3] A. Brown, C. Home, L. Danoon, R. Palama (2018), University of Manchester, Measurement and Modelling of RADAR signatures of large wind turbine using multiple sensors. 2018 IEEE Radar Conference (RadarConf18), DOI: 10.1109/RADAR.2018.8378767.

[4] J. Bredemeier, T. Schrade, M. Mihalachi, D. Ulm et al (2017) Effects of Wind Turnbines on RADAR Signals. EMWT 2017 Braunschweig 6.-7.12.2017

[5] G. Greving et al., NAVCOM Consult (2016) Wind Turbines and Systems – Simulation Results and Experience 2016, EMWT 2016, Dec 06-07, 2016 at Fraunhofer FHR in Wachtberg

[6] C. A. Jackson, (2007) "Windfarm characteristics and their effect on radar systems," 2007 IET International Conference on Radar Systems, pp. 1-6, 15-18 October, Edinburgh, UK.

[7]https://www.dfs.de/dfs_homepage/de/Flugsicherung/Umwelt/Windkraft/FAQs%20zu%20Windkraft/

[8]https://www.mdr.de/nachrichten/politik/inland/windkraft-fuenfzighertz-will-ausbau-flughaefen-offshore-100.html

[9] H.J. Mametsa, S. Laybros, T. Volpert, P.F. Combes, P.N. N'Guyen, P. Pitot, (2004) " FERMAT: A high frequency EM scattering code from complex scenes including objects and environment", Progress in Electromagnetic Research Symposium (PIERS 04), Pisa, Italy 28-31 March 2004.

[10] P. Soudais, H. Steve, and F. Dubois, (1999) "Scattering from several test-objects computed by 3-D hybrid IE/PDE methods", Antennas and Propagation, IEEE Transactions on, Vol. 47, No. 4, pp. 646-653, 1999.

[11] Hasselmann, K. and al., (1973) "Measurements of windwave growth and swell decay", Ergänzungsheft zur Deutschen Hydrographischen Zeitschrift, 1973

[12] N. Douchin, C. Ruiz, J. Israel, HJ. Mametsa, (2019) "SE-Workbench-RF: Performant and High-Fidelity Raw Data Generation for Various Radar Applications", Proceedings of IRS2019 Conference, 26-28 June 2019, Ulm, Germany

The Latest in Telemetry Data Visualization Solutions

David N. Kortick

Product Development Manager

AstroNova Inc.

600 East Greenwich Ave.

West Warwick, RI 02893 USA

Abstract/Summary

The importance of visualizing time-critical data in aerospace telemetry applications cannot be overstated. The latest data visualization solutions offer the ability for the telemetry facility to utilize strip-chart display software, real-time chart printers or combination systems for range safety, flutter testing, quick-look and other mission critical applications. These visualization and recording systems are designed to support both existing and new ground stations by offering analog as well as digital (network) interfaces. Sharing a common user experience (UX) design, today's telemetry data visualization solutions provide the immediate data display required by flight test engineers, range safety personnel and others. Advances in recording, display, data acquisition and security technologies are all part of the architecture of these recording systems, which are easily adapted to the needs of each user.

Keywords

visual display, data recording, recorder, chart recorder

Topics

Data Acquisition & Sensors, Recording Technology

Introduction

Data visualization has been a crucial component of telemetry systems for many decades. From early strip chart recorders with their unique ability to graphically represent continuous real-time data to the latest in both software and hardware-based recording systems, data visualization is used in many applications including flight testing, rocket and missile launch monitoring, range safety and satellite testing. The modern telemetry recording systems have constantly evolved by utilizing the latest technologies to fulfill the needs of the users.

A Brief History of Telemetry Recording Systems

Telemetry data was originally recorded on strip chart recorders. These systems were originally galvanometer-based pen-writing instruments. The typical recorder drew eight waveforms on eight separate pre-printed grids of 40 mm width. The point-of-writing mechanism changed over many years, from free-flowing capillary ink styli to pressurized ink, to heated elements on thermally sensitive paper. While these recorders served the telemetry community very well, the drawbacks to these methods include the mechanical limitations of the galvanometer for high frequencies, the high maintenance costs of these systems, the untidiness of ink-based recorders and the lack of communication for command and control.

Telemetry recorder technology evolved to include a light-beam oscillograph which marked a photo-sensitive paper. While this offered higher bandwidth signal recording, the photosensitive paper was expensive and did not archive well.   The next technology evolution was the electrostatic array recorder which was the first introduction of digital recording in a previously analog field and offered advancements such as overlapping channels, alphanumeric annotation, grids, graphics, and host control capabilities. One of the shortcomings of the electrostatic array recorder was a digitization effect due to a recording technique that limited the output resolution.

The thermal array recorder followed shortly after the electrostatic array recorder. This printing method, almost exclusively the only type still used for telemetry today, has many benefits including high resolution (300 dpi) printing, high frequency response, flexible chart formats, and alphanumeric annotation. Full host control was a major advancement for thermal array recorders, finally closing the loop between telemetry processor and recording system. Over the many years since the introduction of the first thermal array recorder, features such as video display, large memory capacities and digital signal processing were added to these instruments. High-speed digital data inputs became widely used with the thermal array recorders, eliminating the high costs and constant need for calibration of digital-to-analog converters (DAC) at telemetry installations.

Telemetry Recording Today

The data visualization provided by the telemetry recorder is still very relevant today. The instruments have become much more than basic print recorders and have now advanced into true multimedia recording systems. The additions of the large touch-panel display, intuitive graphical user interfaces (GUIs), large capacity storage media and networking are capabilities found in the modern telemetry recorders at many telemetry facilities. The latest of these systems employ standard data protocols and include advancements in real-time data visualization, recording and networking.

The trend with most telemetry facilities today is toward software-based solutions for the real-time visualization of the telemetry data. However, the ability to visualize the telemetry data in a format that is useful to the engineer also calls for a hardcopy output for many applications. What is important is that the user experience (UX) is the same for these visualization systems. As paper is no longer the primary storage medium, the telemetry recorder plays a smaller, yet no less useful role. Continuous paper output still offers many unique benefits to the user in applications such as quick-look and range safety. The physical paper recording is a reliable and tangible medium that can be quickly viewed, marked-up, and "flipped through" to see important data or trending patterns. All of the data needed to make a determination is located on the paper. This includes the waveform data itself, timing marks, IRIG timecode, text annotation, event marks and even highlights of areas deemed important by the engineer. While the strip chart printout itself can be an important part of the record, there are many technologies utilized in the recorder that augment this fundamental capability.

Waveform Display and Video

The most recent designs of the telemetry recorder utilize a high-resolution touchpanel display that gives the user the ability to visualize real-time waveforms, events, grids and timing information in a traditional telemetry waterfall format. These displays can be totally customized in terms of waveforms sizes and placement, colors, grid definition, timing marks and annotation. In addition to the real-time waveforms, other types of graphs such as XY plots and meter displays can be selected by the user. Figure 1 shows a typical display formatted for 8 channels.

These displays also offer an intuitive and familiar GUIs for local command and control of the recording system. Control panels can be standardized or customized by the facility for specific tests and missions. The control panel buttons seen at the top of Figure 1 are a good example.
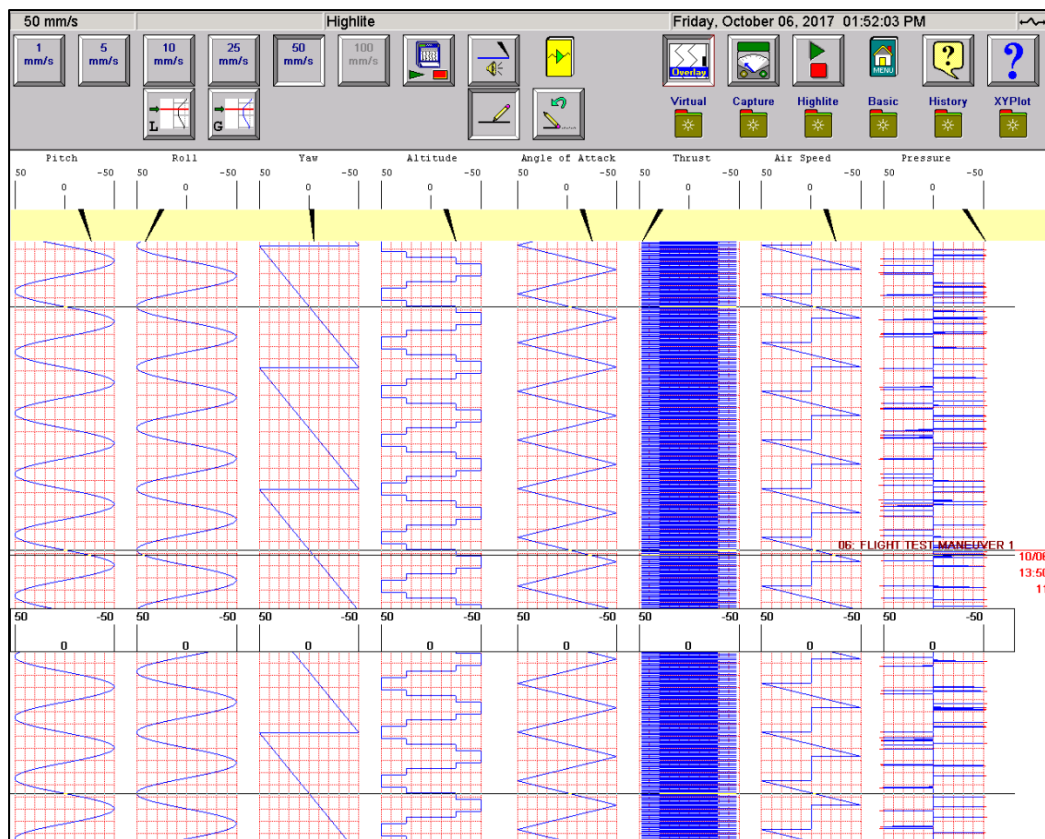
Figure 1: Screenshot of the recorder display

Another benefit of these instruments is the ability to have a video output available to allow the display to be sent to another monitor, auxiliary feed to a control room, or to a video mixing system. The latest designs utilize the High Definition Multimedia Interface (HDMI) for video output, due to the industry acceptance and standardization of this interface.

Data Acquisition and Storage

One of the most important aspects of the next-generation telemetry recorder is the ability to offer both digital and paper storage mediums, giving the engineer or analyst both options.   In fact, the data acquisition capabilities of these systems have been greatly improved based on the requirements of the telemetry community. Sample rates of up to 200kHz per channel are now standard for installations where analog signal inputs are required. Enterprise-grade solid-state drives (SSD), with their consistent performance and reliability, have replaced magnetic hard drives in most of the newest installations. These drives are typically 3D NAND TLC flash memory, offering a good balance between performance and cost.

A typical use case for the data acquisition capability is for an application that requires an entire mission to be recorded locally on solid-state removable media, but the engineer or analyst may

want to have a hardcopy printout of a few important areas of interest during the mission. The modern telemetry strip chart recorder makes this possible and offers maximum flexibility.

Networking

The ability to network multiple telemetry recorders is a requirement for almost all telemetry facilities. Many of these have central locations where one engineer or system operator can control the various recorders that are used in the facility. The newest systems offer Gigabit Ethernet (1000BASE-T) ports for interfacing to the equipment. This connectivity provides command and control, file transfer capability, web interfaces, digital data interfaces and even peer-to-peer communication. Figure 2 shows a typical block diagram for this interface.
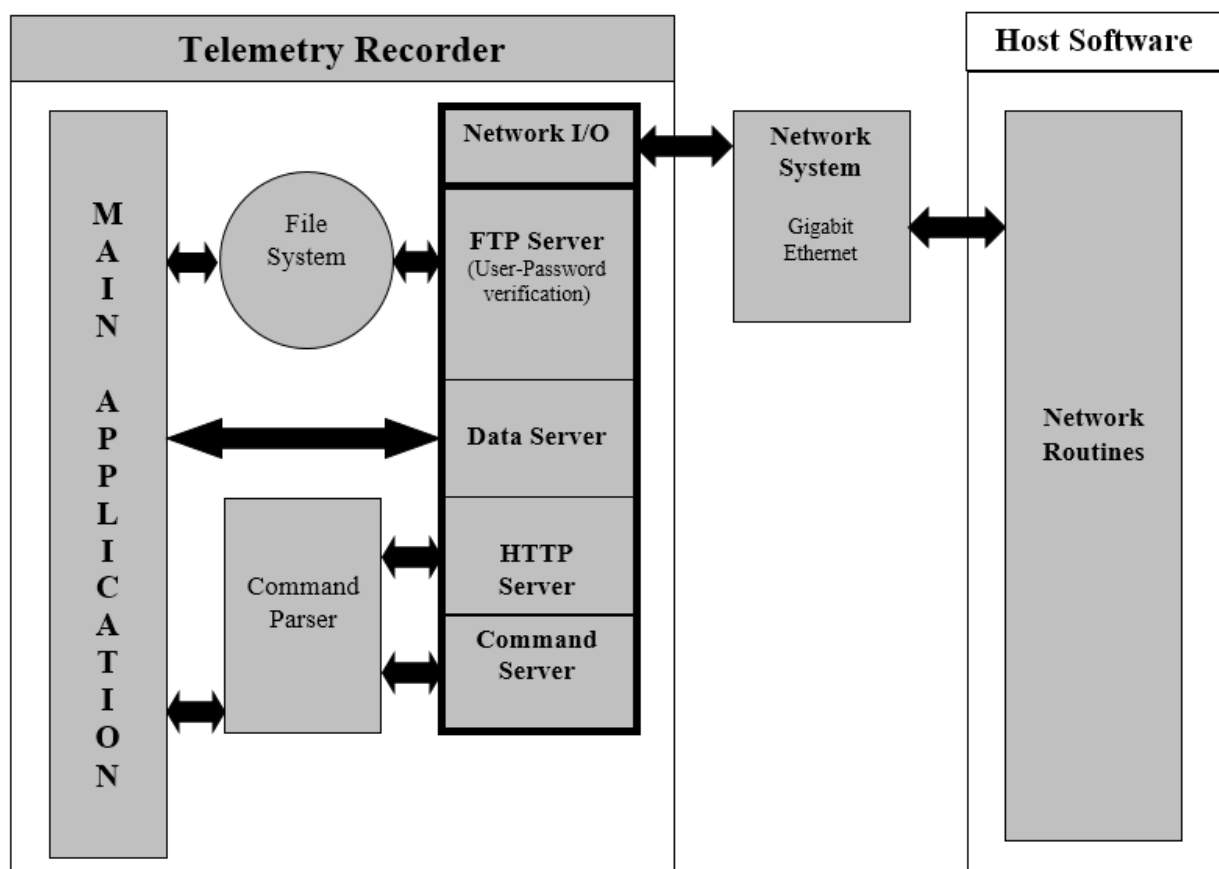
Figure 2: Block diagram of the network interfaces for the telemetry strip chart recorder

Modern telemetry recorders utilize both TCP and UPD protocols, depending on the need of the specific telemetry facility. FTP and HTTP protocols are also used in some facilities for file transfer and support of web-based client software.

The digital data interface over Ethernet is one of the most commonly used capabilities of these recording systems, eliminating the need for DACs to provide analog data to the recorder. A higher-level data protocol is used to transmit data to the recorder. An example is an Ethernet digital data protocol which is optimized for strip chart recorder data with the use of the min/max method and requires the host to send line segments based on the minimum and maximum waveform values for a slice of time, or ΔT. The value of ΔT depends on the chart speed of the recorder. The data packets containing the line segments also include timing marks, IRIG time stamps and grid information that provide the host with advanced control over the visualization and printing of the data. Other benefits of a digital data interface include the reduced bandwidth requirements on the network and the advantage of glitch capture.

With glitch capture, the primary sample rate at the host can be much higher than the min/max rate at which the packets are sent to the strip chart recorder. Therefore, a high-speed event can be seen as a minimum or maximum even at the relatively low rate that data is being sent across the network. In Figure 3, a waveform spike with frequencies of 1, 10, 100, and 1000Hz is shown with the full amplitude information as recorded.
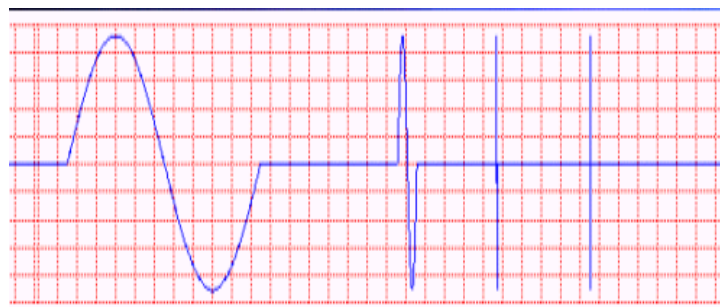
Figure 3: A sinusoidal waveform with glitch capture

Security

Security is an important consideration for any telemetry equipment today due to the proliferation of both physical and cyber threats. Malicious code, viruses and state-of-the-art intrusion methods are all concerns for the telemetry community. The next-generation telemetry recorders play a role in mitigating these threats by the use of the advanced technologies and operating systems in their designs.

From a physical standpoint, the recording system utilizes lockable and removable drive bays for the system and data acquisition media. This security is not only utilized for protection against malicious intent, but also to have the ability to use different media for distinct classified projects.

The Windows 10 operating system is now used for the display and GUI functions of the recorder. This operating system, accepted by the DoD, offers the ability for Information Assurance (IA) compliance with the configuration standards of the Security Technical Implementation Guide (STIG).

Technology-Based Data Visualization Solutions

One of the challenges of telemetry facilities that use strip chart recorders is that many of these instruments are aging and utilize obsolete technology. These systems are used in a mission critical environment where failure is costly. Perhaps the most noteworthy development is the interest of many telemetry facilities in replacing these older systems with a COTS solution of customizable data visualization and printing methods that offer the same UX. This architecture is viable because of the ability to separate the waveform display capability from the printing capability in the latest recorder designs. Furthermore, the digital data protocol allows for a seamless integration between the source of the data and the endpoints for visualization or recording of the time-critical data. These endpoints are data visualization software stations, continuous network printers and strip chart recorders.

Figure 4 shows an example configuration where a telemetry processor is sending packets via the digital data protocol over Ethernet to these solutions. If only data visualization and local digital storage is required, then the visual display software station is utilized. If only strip chart recording is needed, then the continuous network printer can be used. If a full visualization, data recording and strip chart solution is required then the telemetry strip chart recorder is used. All three solutions share a common command and control structure as well as support the digital data protocol.
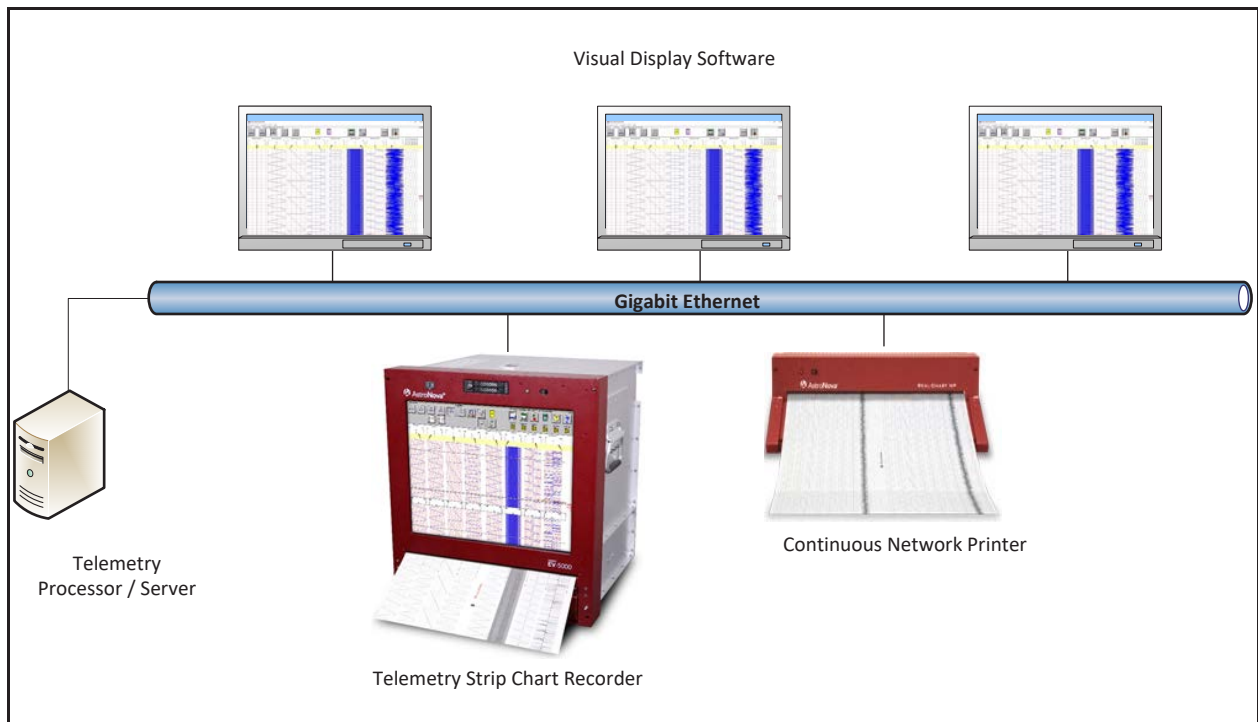
Figure 4: An example of a custom data visualization and recording configuration

The benefit to this architecture is that the telemetry facility can adapt to the needs of their customers. This flexible configuration can support different missions requiring different numbers of recorded parameters. An example is having many engineers or analysts viewing the data in real-time using the visual display software and having the ability to send data to the continuous network printer upon demand. Another example could be the addition of IRIG 106 Chapter 10 recording and playback capability.

Conclusion

The telemetry recording system is still a useful and viable part of the instrumentation for telemetry facilities throughout the world. While the need for a continuous printed record of telemetry data has been reduced due to other recording media, the requirements for some applications still call for the printed record. Today's next-generation systems not only fulfill that requirement but also offer the users and facilities much greater flexibility and capability in terms of data visualization and data acquisition. By offering a combination of software-based and hardware-based solutions, these telemetry recording systems are flexible in their configuration and can be adapted to any telemetry facility. The latest technologies and protocols used in today's designs also ensure that these systems meet current and future mission requirements.

EYE-GAZE INTERFACE TO OPERATE AIRCRAFT DISPLAYS

LRD Murthy[1], MD Babu[2], V Yelleti[3], KPS Saluja[1], P Biswas[1]
[1] Indian Institute of Science, Bangalore 560012, India,
[2]Aircraft and Systems Testing Establishment, Indian Air Force, Bangalore, India
[3] Indian Institute of Information Technology, Kalyani, India
dillibabu_mohan@yahoo.com, pradipta@iisc.ac.in

Abstract:

Eye gaze trackers have been studied widely for their utility in aviation domain since long time. So far, numerous studies have been conducted in the direction of gaze-controlled interfaces for electronic displays in flights and Head mounted display systems under simulated conditions. In this paper, we present our study on usage of eye gaze trackers in real flight conditions and their failure modes under such usage conditions and illumination. We show that the commercially available off the shelf (COTS) eye gaze trackers with state-of-the-art accuracy fails to provide gaze estimates beyond certain level of illumination on the eyes. We also show that the limited available tracking range of eye gaze trackers limit them to provide gaze estimates even during the pilots' natural operating behavior. Further, we present three approaches of developing eye gaze trackers which are designed to use webcam instead of infrared illumination and are aimed to be functional at high illumination conditions. We present our intelligent tracker, developed using OpenFace framework, provides comparable results to COTS eye tracker in terms of interaction speed for both indoor and outdoor conditions.

Key words: Eye gaze tracking, Eye gaze estimation, Gaze controlled interface, Human Machine Interaction, Cockpit Display

1. Introduction

Eye gaze tracking is the process of estimating where a person is looking at. Eye gaze tracking technology is used to understand eye gaze scanning process, visual search and reading behavior from late 18th century. With availability of portable infrared based eye trackers, researchers also explored controlling digital user interfaces just by looking at it. While the technique sounds intuitive, underlying technical limitations like innate eye movement process and the constraints on direct manipulation of graphical user interfaces so far restricted eye gaze controlled interfaces to limited applications for people with severe disability [3] and as a binary input channel (on/off) for a few smartphone video viewing applications.

In recent time, a set of new applications were explored for eye gaze controlled interfaces where the operators' situation impedes him/her to operate traditional physical or touch screen user interfaces. Examples of such situations include undertaking secondary tasks in automotive [2, 8] and mission tasks inside combat aircrafts [10, 11].

Eye gaze tracking technology is widely explored in aviation domain for pilot training, understating pilots' scanning behavior, optimizing cockpit layout and recently to estimate pilots' cognitive workload [6, 9]. Although a commercial product is not yet available, different defense manufacturers are already investigating to use eye gaze-controlled interfaces inside cockpit. While most of the research on eye gaze-controlled interfaces for military aviation concentrate on Head Mounted Display Systems (HMDS) [4, 11], this paper explores use of eye gaze controlled interface for Head Down Display. The paper initially explored a state-of-the-art wearable eye tracking device in a combat aircraft undertaking representative combat maneuvers and then proposed and evaluated a set of algorithms for developing screen mounted eye gaze tracker for operating head down displays. These algorithms can also be used in transport and passenger aircrafts.

The paper is organized as follows. The next section summarizes our earlier published research on eye gaze-controlled interfaces followed by the study on actual aircraft. Section 4 presents different eye gaze tracking algorithms followed by concluding remarks.

2. Earlier Research

So far, we set up and undertook numerous investigations to study the utility of eye gaze tracking in aviation field. Our studies are set in

different conditions like a fixed base flight simulator, a virtual reality (VR) based simulators. We also performed user studies in actual transport and military aircrafts. Our studies focused on evaluating pilot's interaction experience for mission control tasks using gaze-controlled interfaces in simulators and actual flights.

Initially, we set up a flight simulator and evaluated gaze-controlled interface for military fast jets [10]. Our studies found that the gaze-controlled interface statistically significantly increased the speed of interaction for secondary mission control tasks compared to touchscreen and joystick-based control systems. Subsequently, we tested the gaze-controlled system inside a transport aircraft both on ground and in-flight with military pilots. We found that the pilots could undertake representative pointing and selection tasks in less than 2 secs on average.

We also integrated the COTS eye tracker to a flight simulator to evaluate the feasibility of its functions as a part of HMDS and evaluated the latency incurred to point at an aerial target. In parallel, we measured pilots' various ocular parameters like fixation rate, pupil dilation and saccadic intrusion in standard and VR based flight simulators at various G levels in different combat aircrafts. We compared these metrics in various flight phases and during air to ground attack maneuvers [1].

In addition to utilizing COTS based eye trackers for our user studies, we studied the accuracy of gaze control in various acceleration due to gravity (G load) conditions under various flight maneuvers. We used a COTS eye-gaze tracker in an advanced jet trainer aircraft and we found that the error is less than 5° of visual angle up to +3G although it is less accurate at -1G and +5G [7]. In line with our error measurements of COTS eye gaze trackers under various flight conditions, in the next section, we present our analysis on various failure modes of COTS eye gaze trackers in actual flight conditions.

3. Testing COTS tracker in Combat Aircraft

We have recorded data from two flights using the COTS eye tracker (Tobii Pro Glasses 2) which uses infra-red (IR) illumination-based eye gaze estimation principles [12]. The duration of the first flight is 55 minutes 58 seconds *(Flight 1)* and another flight's duration is 56 minutes *(Flight 2)*, the flight profiles are furnished in Table 1 below. The eye tracker contains a front-facing scene camera which records the first

Table 1 Flight Profiles

Sl No	Objective	Profile
Flight #1	Maneuvering flight with head mounted eye tracker on Pilot in Command	Take-off – climb – level flight to Local Flying Area – Constant G (3G and 5G) level turns both sides each – Vertical loop – Barrel Roll – Air to Ground dive attack training missions – Descent – ILS Approach and landing
Flight #2	Non - Maneuvering flight with head mounted eye tracker on Pilot in Command	Take-off – climb – level flight to Local Flying Area – Straight and Level cruise with gentle level turns – Descent – ILS Approach and landing

person view of the pilot. It also contains four eye-cameras, two cameras per each eye, to record the eye movements. The eye tracker estimates gaze points at a frequency of 100 Hz. The frame rate of scene camera is 25.01 frames/second at 1920 x 1080 resolution and that of each eye camera is around 50 frames/second with a resolution of 240x240. Each gaze point is recorded with a dedicated identifier, called "gidx". We initially used *Tobii Pro Lab* tool to analyze the recorded gaze samples and observed that both flight recordings contain gaze samples only for around 50% of the duration. We investigated this loss of data samples during the flight using the raw data provided by manufacturer in json format and by correlating the raw data with the eye images.

The raw data obtained in json format contains various other information recorded during the flight like gyroscope and accelerometer data. We discarded the irrelevant information and retained the data points required for our investigation of lost gaze points.

At first, we synchronized the raw data stream and the eye camera stream in time scale since eye camera stream starts off with an offset from raw data. This is achieved using the Position Time Stamps (PTS) provided in both data streams. We also find that the different frequencies of these two streams is a challenge for data synchronization. Hence, we considered the time duration between two successive frames of eye camera stream and consider all the corresponding gaze data points recorded during that time window. Thus, the latter frame and these data points together form one pair of synchronized data points. Each time windowed raw data may contain multiple gaze points. Every gaze point with its "gidx" contains a status code,

's' which indicates the error associated to that datapoint, if any. The status code 0 indicates no error and any non-zero value of s indicates an error associated is with the data point. We observed that all the gaze points with a non-zero status code are recorded as zeros for both x and y directions [0.0, 0.0]. The gaze points are provided in normalized values, hence the minimum gaze point is [0.0, 0.0] and the maximum is [1.0,1.0].

We segmented the synchronized data points into two categories. The first category *category1* contains eye stream frames whose corresponding gaze points have zero status code. The second category *category2* contains those eye frames with all corresponding gaze points with non-zero status codes. There are frames whose data points have only a subset of gaze points contains zero status code. We did not consider these frames in our analysis as it brings uncertainty on eye image tagging.

For *Flight 1*, we observed that out of 167,647 frames, only 57,111 frames fall under *category1* and 69,732 frames fall under *category2*. For *Flight 2*, we observed that out of 167,567 frames, only 81,911 frames fall under *category1* and 51,402 frames fall under *category2*.

Summarizing, 41.6% of the frames does not have any gaze points recorded during *Flight 1* and for *Flight 2*, this stands at 30.7%. Further, if we just look at unsynchronized raw data, both flights recorded more than 51% of the gaze samples are error-prone.

We visually inspected these flight recordings and we hypothesize two reasons for this data loss.

1. Higher levels of illumination on eyes may affect the eye tracker resulting in no gaze estimation.
2. Limited field of view (FoV), especially in the vertical direction, renders the eye tracker with no gaze estimates when user looks beyond the tracking range.

We validated our hypothesis 1 using the eye images in the above mentioned two categories. Since the recorded video stream is an IR video, we converted all eye images into grayscale and computed average of all the pixel values for each image present in both categories. Figure 1 represents the histogram of image intensities for *category1* and *category2* for Flight 1. Figure 2 represents the same for Flight 2. Figure 1a indicates that 93% of the images under *category1* have an average intensity less than 131. But, *category2* contains 42% with average
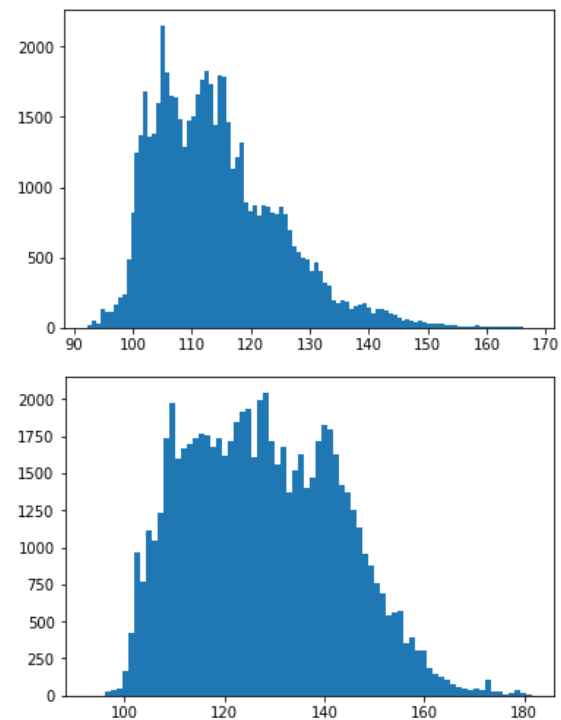
Figure 1. Histogram of image intensities for Flight 1 1a. *Category1*(Top) 1b. *Category2* (Bottom)

intensity higher than 131. Further, this can also be observed in Flight 2 case, shown in Figure 2. Around 42% in *category2* have higher intensity than 150, while *category1* contains 94% of the images with intensity less than 150. This indicates that images with higher illumination, precisely above 131 in Flight 1 and above 150 in flight 2 have low probability to obtain accurate gaze estimates.

While this evidence supports our hypotheses 1 partially, we observed that there is overlap in the left and right histograms plotted in Fig 1 and Fig 2. Hence, we could not identify a clear average image intensity threshold in order to identify all the failure modes of eye gaze estimation.

We further investigated the data points in *category2* to understand the 58% of the datapoints which have lower image intensities than above mentioned thresholds for each flight using our hypotheses 2. Since we observed that the gaze estimates are lost for a sequence of eye image frames, we clustered the datapoints in *category2* based on their "gidx"s. If a sequence of datapoints under *category2* are having successive gidx's, then all those points are considered as a single cluster. Hence, each cluster can contain one datapoint or several datapoints. Extending our hypotheses 2, we assumed that the pilot must be looking at a
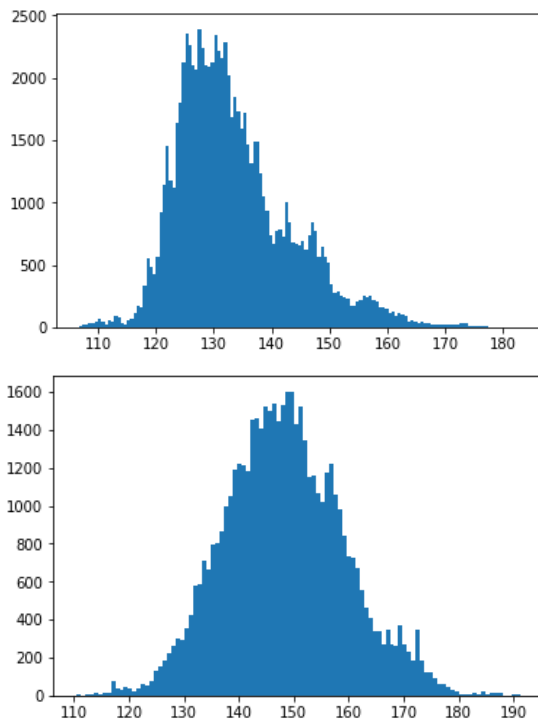
Figure 2. Histogram of image intensities for Flight 2 2a. *Category1*(Top) 2b. *Category2* (Bottom)

position closer to the extreme tracking positions (beyond which eye tracker cannot track), just before or after the eye tracker fails to provide gaze estimates. During our visual inspection of first person video recorded using eye tracker, we observed that the pilot looks down for various activities like looking at the information displayed in the Multi-functional displays (MFD)s while keeping the head faced horizontal (perpendicular to the vertical axis of the aircraft). During such scenarios, we observed that gaze points were not recorded.

Hence, we analyzed three preceding or subsequent datapoints adjacent to each cluster, which we refer to as boundary datapoints. We looked for boundary datapoints with gaze values beyond 0.8 and less than 0.2 (in both x and y). If any of the boundary datapoints satisfy above criterion, then we may infer that the loss of gaze points is due to the pilot looking beyond the tracking range of the eye tracker.

For flight 1, we obtained 12178 clusters for 69,732 datapoints. For these clusters, 11,865 (97.43%) clusters have boundary points that satisfy the above criterion. To understand image intensities for these datapoints, we plotted a histogram of the image intensities for the datapoints whose boundary points satisfy above criterion. We observed that these image intensities lie in the range of [96,145]. This is clearly in the overlap range identified between Figure 1a and Figure 1b.

Similarly, for flight 2, we obtained 8646 clusters for 51,402 datapoints. For these clusters, 8408 (97.24%) clusters have boundary points that satisfy the above criterion. Interestingly here as well, we observed that the histogram of image intensities for the above points lie in the range of [117,164], which is the range of overlap identified in Figure 2a and Figure 2b.

Thus we infer that, this eye gaze tracker could not identify beyond certain illumination level or if the user is looking beyond its tracking range. We should note that the pilot is performing his assigned tasks during the flight and maintained his natural behaviour. This indicates that the tracking range offered by this eye tracker is not sufficient for military aviation environments.

Hence, using our two hypotheses and the raw data, we studied the failure modes of eye gaze tracker in aviation environment. We further add that, while commercial off-the shelf eye trackers may be used in real aviation environments, researchers and practitioners should keep in mind about both the horizontal and vertical tracking range of the eye tracker and it's robustness to external illumination as there is a high chance that the illumination varies rapidly at high altitudes in high speed maneuvers.

4. Developing Eye Gaze Trackers

In this section, we have described three different eye tracking systems and compared them through user studies Initially we described the different algorithms used for estimating gaze followed by two user studies.

HoG based Gaze Tracking System

We used a pre-trained facial landmark detector with iBUG 300-W dataset [iBUG 2019], which works on classic Histogram of Oriented Gradients (HoG) feature combined with a linear classifier to detect facial landmarks [Rosebrock 2019]. In comparison, Haar cascades are a fast way to detect an object but often detect more false positives compared to HoG and linear classifier [Dalal 2005]. HoG features are capable of capturing the face or object outline/shape better than Haar features. On the other hand, simple Haar-like features can detect regions brighter or darker than their immediate surrounding region better. In short, HoG features can describe shape better than Haar features and Haar features can describe shading better than HoG features. In this case the shape is more important as we need the landmarks of the face hence HoG features produced a better result. After detecting eye region, we detected

pupil location by selecting the smallest rectangle possible in the eye region where the pupil can exist. After retrieving pupil locations, we calculated the Eye Aspect Ratio (EAR). We have noted that the eye aspect ratio changes with respect to the distance between the user and the camera. We have modified the EAR calculation formula by using the distance between the two eyes as denominator.

Webgazer.js

We implemented a second system using webgazer.js [Webgazer 2020; Agrawal 2019] to compare performance of the proposed system. Webgazer.js runs entirely in the client browser. It requires a bounding box that includes the pixels from the webcam video feed that corresponds to the detected eyes of the user. This system uses three external libraries (clmtracker, js_objectdetect and tracking.js) to detect face and eyes. It has methods for controlling the operation which allows us to start and stop it. We have taken the mean of last thirty points from webgazer.js for better target prediction and accuracy of system. We also calculated the mean value during this time to predict the gaze location on a webpage.

Intelligent System

We have developed a gaze block estimator which maps user's eye movements to 9 screen blocks using OpenFace [Baltrusaitis 2018] toolkit. Since the OpenFace (figure 3) was reported to have an error of 6^0 for gaze point estimation, we designed a calibration routine which uses the gaze vector data from Open-Face and maps user's eye movements to screen blocks, instead of screen pixels. We have divided the screen into 9 blocks of equal area. We designed a smooth pursuit based calibration routine where a marker traverse across all these 9 blocks and user was asked to follow the marker's movement. The corresponding gaze vectors from OpenFace were recorded and stored with the respective block number as the label. Once the marker completes its path, a neural network is trained to map these gaze vectors to 9 blocks of the screen. For this classification task, we used a 2 hidden layer network with 256 and 128 neurons respectively with cross-entropy loss function and with Adam optimizer. We used the 70% of the data we recorded during calibration for training, 15% for validation and the rest for testing. On a i7 processor computer, we observed that each epoch takes around 0.8 seconds and we trained the network till the test accuracy reaches 90%.
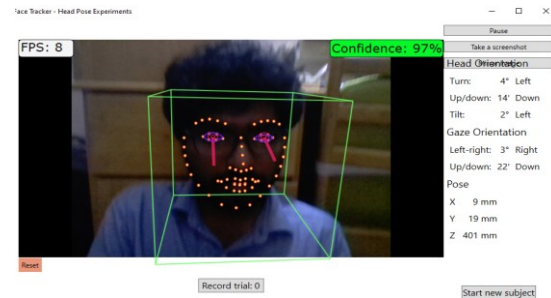
Figure 3. Screenshot from OpenFace Face Tracker

User Study

We undertook the following user study to compare different eye tracker implementations in different lighting conditions and compared them with a COTS screen mounted eye gaze tracker.

Participants: We collected data from 9 participants (8 male, 1 female). All participants were recruited from our university. They do not have any visual or motor impairment.

Material: The user trial was conducted on a Microsoft surface pro tablet powered by dual-core processor and it comes with 8 GB RAM and running Microsoft Windows 10 operating system. The surface has a 5 MP camera, which was used to estimate gaze direction.

Design: We wanted to use the eye tracker to operate a graphical user interface with limited number of screen elements, hence instead of traditional precision and accuracy measurement, we calculated the pointing and selection times for a set of fixed positions in screen.

We created a user application in which we divided the screen into nine blocks and one of the blocks gets randomly highlighted with blue colour as shown in figure 4a. If the user clicks on the blue block, it turns green as shown in figure 4b and a different block was highlighted. If the user is unable to click on the highlighted block within 10 seconds, it turns randomly some other block to blue. Using this interface, we calculated the response time by measuring the time difference between appearance of a highlighted block and its selection. Users selected target using the left mouse button.

The trial was performed twice - once in laboratory with lux meter reading 180-200 lux and the other in outdoor condition with lux meter reading between 1800-3000 lux.

The trial consisted of four eye tracking implementations

- HoG based bespoke webcam based gaze tracker
- Webgazer.js based webcam based gaze tracker

- OpenFace based intelligent webcam based gaze tracker
- Tobii PCeye mini eye-gaze tracker (referred hereafter as COTS tracker)

a) Randomly highlighted block in blue
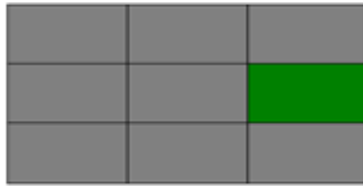
b) After click, highlighted block turns green

Figure 4. Pointing Task application

For all trial conditions, we conducted trial on the same user application discussed before. The order of conditions was randomized to minimize practice or learning effect. Each participant undertook all trial conditions.

Results: We recorded 322 pointing tasks inside room and 270 pointing tasks outside. We measured the time difference between onset of a target and its correct selection. We removed outliers by identifying points greater than outer fence. We removed only one point from the selection times recorded from the bespoke eye gaze tracker while 12 data points were found values higher than outer fence for the COTS eye gaze tracker.

Figure 5 presents the average selection times and standard deviation. Participants took lowest time to select target using the COTS tracker. We undertook a 2 × 2 unbalance factorial regression based ANOVA (type of eye gaze trackers × lighting conditions) on the response times.
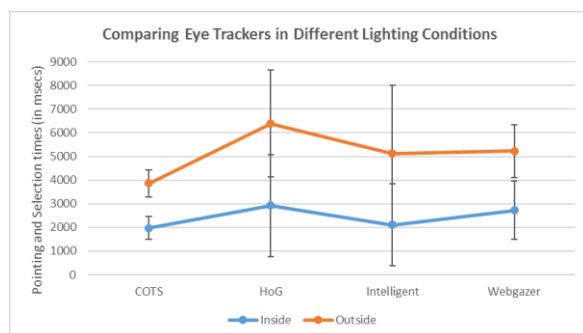
Figure 5. Comparing Average Response Time among different eye trackers

We found

- significant main effect of type of eye gaze tracker $F_{(3,567)}=15.44$, $p<0.01$
- significant main effect of lighting condition $F_{(1,567)}=4.05$, $p<0.05$
- significant interaction effect of type of eye gaze tracker and lighting condition $F_{(3,574)}=3.45$, $p<0.05$

Then we undertook two one-way ANOVAs for each lighting condition and found significant main effect of eye gaze tracker implementations.

- Inside room $F_{(3,318)}=8.43$, $p<0.01$
- Outside room $F_{(3,266)}=11.31$, $p<0.01$

Finally, a pair of unequal variance t-tests did not find any significant difference between COTS tracker and our intelligent eye gaze tracker implementation inside room at $p<0.05$ although the difference in response times between the intelligent system and COTS tracker was significant at outside condition at $p<0.05$.

Discussion: Our initial approach for eye gaze estimation used feature based approach. We used a method that extracts Histogram of Oriented Gradients (HoG) features combined with a linear SVM to detect eye landmarks. These landmarks were used to compute Eye Aspect Ratio (EAR) feature to estimate the gaze block on the screen. Even though HoG based landmark detection had been used earlier widely, we observed that it occasionally failed to detect landmarks of our users and affected the gaze estimation accuracy. Variations in illumination and appearance of facial features like beard or spectacles could affect tracking accuracy based on pre-selected facial features.

The second approach, Webgazer.js proposes to map the pixel data of eye images directly to gaze locations rather than to rely on handcrafted features from eye images. They used a 6 x 10 eye image patch for each eye and converted them to 120-dimensional feature vector. This vector is used as an input for a regression model to map to gaze points on screen. This approach also relies on multiple eye landmark detection algorithms which suffers similar limitations as HoG based algorithms. Further, this approach requires users to click at least 40-50 locations on screen for calibration purpose before it can make predictions which requires significant time.

OpenFace uses a state-of-the-art deep learning approach for landmark detection and gaze estimation. It uses Constrained Local Neural Field (CLNF) for eye landmark detection and tracking. Unlike HoG and Linear SVM approach,

which was based on handcrafted features and trained on a relatively smaller dataset, OpenFace uses larger dataset and deep learning approach to learn the estimation of 3D gaze vector from the eye images. Even though OpenFace is not very accurate in predicting gaze points on screen, it does not suffer from illumination and appearance to detect eye landmarks. Further, OpenFace was implemented in C++ which makes real-time gaze estimation possible even on CPUs. In addition to these, the reported state-of-the-art cross-validation accuracy prompted us to test for a gaze block detection application.

It may be noted that the response times were lowest for the COTS tracker compared to webcam-based approaches. The COTS tracker used on board ASIC chip to run image processing algorithms, which has lower latency than general purpose processor used in webcam-based eye trackers [5]. It may be noted that still the difference in response times was not significant inside room between COTS and intelligent tracker. Our future work is investigating to further reduce the latency and make eye tracking work in bright lighting condition.

5. Conclusions

This paper presents a case study of testing and development of bespoke eye gaze tracker for operating Head Down Displays in an aircraft cockpit. Our study showed that present COTS eye gaze trackers are not yet ready to be integrated to combat aircraft in terms of tracking eye gaze at different lighting conditions and vertical field of view. We presented a set of algorithms that can be configured for operating multi-function displays inside cockpit and a particular intelligent algorithm using the OpenFace framework worked better than classical computer vision based algorithms.

References

[1] Babu MD, JeevithaShree DV, G. Prabhakar and P. Biswas (2019), Using Eye Gaze Tracker to Automatically Estimate Pilots' Cognitive Load, 50th International Symposium of the Society for Flight Test Engineer (SFTE).

[2] Biswas P. (2016), Using Eye Gaze Controlled Interfaces in Automotive Environments, Springer, 2016.

[3] Biswas, P. and Langdon, P. (2013), A new interaction technique involving eye gaze tracker and scanning system, ACM International Conference Proceeding Series pp. 67-70.

[4] De Reus A.J.C. , Zon R. and Ouwerkerk R.2012. Exploring the use of an eye tracker in a helmet mounted display, Avionics Europe Conference & Exhibition, Munich, Germany, March 21-22, 2012.

[5] Duchowlowski A. T.(2007), Eye Tracking Methodology. Springer, 2007

[6] Fitts, P.M.; Jones, R.E.; Milton, J.L. Eye movements of aircraft pilots during instrument-landing approaches. Aeronaut. Eng. Rev. 1950, 9, 24–29.

[7] Murthy LRD, Mukhopadhyay A., Yelleti V, Arjun S, Thomas P, Babu MD, Saluja KPS, JeevithaShree DV and Biswas P, Evaluating Accuracy of Eye Gaze Controlled Interface in Military Aviation Environment, IEEE Aerospace 2020

[8] Prabhakar G, Ramakrishnan A, Murthy LRD, Sharma VK, Madan M , Deshmukh S and P Biswas, Interactive Gaze & Finger controlled HUD for Cars, Journal of Multimodal User Interface, Springer, 2020

[9] Rudi, D., Kiefer, P., Giannopoulos, I., & Raubal, M. (2019). Gaze-based interactions in the cockpit of the future: a survey. Journal on Multimodal User Interfaces, 1-24.

[10] Shree, D.V., Murthy, L.R.D., Saluja, K.S. and Biswas, P., 2018. Operating Different Displays in Military Fast Jets Using Eye Gaze Tracker. Journal of Aviation Technology and Engineering, 8(1), p.31.

[11] Thomas, P., Biswas, P. and Langdon, P. (2015), State-of-the-art and future concepts for interaction in aircraft cockpits, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9176, pp. 538-549.

[12] Tobii Pro Glasses 2 Product Description. 2019. Retrieved 28th May 2019 from https://www.tobiipro. com/siteassets/tobii-pro/product-descriptions/tobii-pro-glasses-2-product-description.pdf/?v=1.95

Vistas architecture implementation in a multi-system integration bench

Santiago Rafael López Gordo[1], Emilio García García[1], Jose Luis Galindo Sanz[1]
[1] *Airbus Defence and Space, S.A.U. Avda. de Aragón, 404. 28022 Madrid, Spain*
santiago.r.lopez@airbus.com

Abstract:

Future developments require an approach able to deal with virtual, hybrid and real equipment integration testing. Benches will be designed as live entities that can evolve during the different phases of the project, evolving from virtual integration to real equipment integration using the same bench.

For this purpose, VISTAS EUROCAE (ED-247) standard approach is going to be demonstrated on an existing integration bench (traditional approach).

VISTAS standard virtualize avionic signals -discrete, analog- and avionic buses -A429, CAN, AFDX …- on a standard Ethernet Bus. This standard includes also the possibility of failure injection. Benches architectures based on VISTAS provide more flexibility and enable important wiring reductions.

Key words: Virtual Testing, V-Model, VISTAS, Virtualization

Introduction. The V-Model for aircraft development

In the V-model (see Fig. 1), two sides can be distinguished. The left one, devoted to Design, and the Right one at which integration activities take place. The major issue related to this approach is that is necessary to wait until the system under test is designed, and the aircraft equipments manufactured, considering hardware and software, to perform the integration activities.

With the Virtual Testing approach, integration activities located at the right side of the V are moved to the left side, by using simulations, in other words: real aircraft equipment (HW/SW) are substituted by models (called Virtual Components) that are representative simulations of the equipment/system/system of systems under test.

The integration of these Virtual Components produces Virtual Test Benches.

So, Virtual Testing comprises the development and integration of Virtual Components and the related testing activity.

The message is clear, the fact of performing integration at the left side of the V model, by using models, allow detecting integration related failures at earlier stages of development, with the immediate effect related to time and cost. It has been demonstrated that the time and cost to fix these failures at earlier stages of

development, is lower if compared to the traditional approach.

At the end test with real equipment –traditional approach- are necessary, but considering Virtual Testing, the number of issues detected are lower as the maturity at system level is higher.

VISTAS added value

VISTAS, which is an Standard for Virtual Interoperable Simulation for Tests of Aircraft Systems, allows the development of Virtual / Hybrid / Real benches with the possibility of performing integration at the earlier stages of the V model, with all the advantages described before.

Although the concept is not technologically a novelty, the importance of VISTAS is that it is a standard itself. Contributors come from different areas such as aircraft manufacturers, aircraft equipment manufacturers and Test System Support companies. All of them are working together and defining the standard. The revision ED-247A of it has been published by EUROCAE in March 2020.

VISTAS standard includes Virtual Components definition and protocols for interfaces standardization between test benches and Virtual components. A proper VISTAS implementation in a bench can provide the following advantages:

Cost reduction:
- By allowing the integration at earlier stages through Virtual Components, discovering system errors and SW errors at the left part

of the V, which is traduced in a lower cost of fixing them if compared with the traditional approach of integration at the right side of the V model.

- By reusing SW libraries of virtual test HW.

- By reducing the cost of bench development, based on an approach of VISTAS implementation at bench architecture by using the so called VISTAS I/O bridges or VISTAS I/O boxes.

Product Quality:
- By improving the regression testing. In the regression testing, automation plays an important role, and combined with a VISTAS implementation at which virtual / hybrid and real scenarios can take place the results is a very flexible and automated environment which allows to perform regression testing with a high degree of coverage, which impacts at the end in the robustness and maturity of the system under test prior to be installed at aircraft.

- Providing a quick verification of system design changes, before HW/SW implementation at real equipment, by minimizing the number of iterations at aircraft equipment level (Ex. only SW/HW changes are implemented at real aircraft equipment when they have demonstrated functionality in a virtual way).

Another advantage is less development time, based on the possibility of the duplication of the virtual environments as they are SW based, and of course by performing integration testing at earlier stage without HW availability. Even it can be considered to distribute geographically development and testing.

VISTAS Architecture

VISTAS architecture is defined by two buses (see Fig. 2):

The first one is the ED-247 Data Bus, intended for transporting all functional data, which are:

- avionics and aircraft system exchanges

- non-avionics data for environment exchanges

In this bus communication lines (avionic buses) are represented in a virtual way, and are used for connecting virtual components (simulations) and/or bridge components in the case of hybrid and/or real equipment benches.

A second bus is a Command and Control bus for operating ED-247 components in terms of configuration, state machine, network modification and health monitoring.

In a virtual configuration, the different Virtual Components are integrated through these two buses.

In a hybrid configuration, real equipment can be connected to VISTAS architecture by using an I/O box or a BRIDGE box. This box interacts with the real aircraft equipment and with the ED-247 data bus and Command and Control bus, and it is bi-directional. This element is composed by an I/O stage which is connected to the interface of the real aircraft equipment, and a HW/SW which is able to convert real aircraft buses and signals into VISTAS protocol virtualized signals and vice-versa. The key point in the development of these I/O boxes is the capability of performing the conversion from/to VISTAS Data Bus and real Avionic Buses and signals with the minimum delay.

In addition, some aircraft equipment manufacturers can consider prototypes with VISTAS interface for development purposes, so the usage of the I/O box can be avoided.

Finally, and moving towards and scenario with real aircraft equipment, the interaction of all the equipments can be achieved through VISTAS, by substituting real aircraft wiring by interconnection through VISTAS protocol. This approach enables the virtualization not only in terms of wiring, but also the virtualization of failure injection modules and patch panel modules which require a large number of wires, manufacturing time and space in the bench. The key point in this approach is the level of representativeness comparing to real aircraft wiring and the delays inherent to a conversion from the real aircraft equipment interface to VISTAS architecture, which maybe today is not enough for obtaining credits for certification in all of the cases.
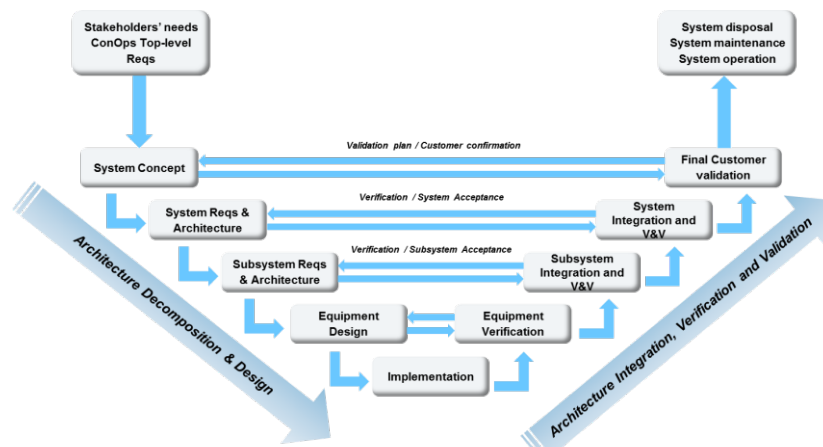
Fig. 1. V-model.
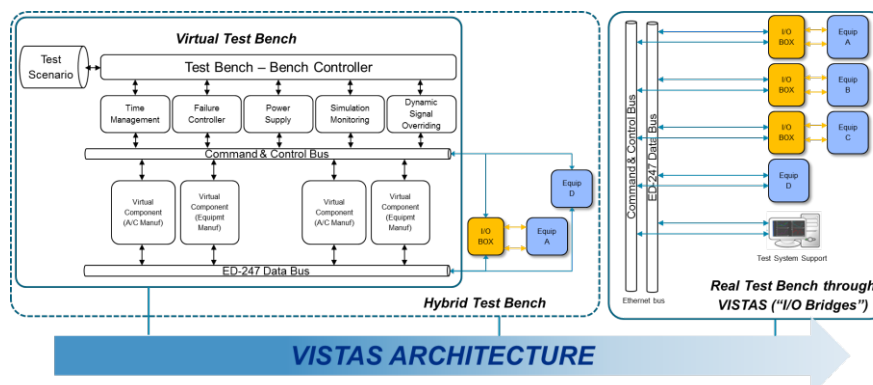
Fig. 2. VISTAS Architecture.

Implementation of VISTAS concept in a real integrated test bench

In this chapter it is described the implementation of VISTAS concept in a real integrated test bench, which is Cleansky 2 integrated bench.

Cleansky 2 is the largest European research program, developing innovative, cutting-edge technologies, aimed at reducing CO2, gas emissions…

Inside the scope of this program ADS is participating at different fields (airframe, regional…) promoting technologies in order to achieve the emissions reduction objectives.

One of them is related the use of Electromechanical actuators, with the intention of substituting hydraulic by electro mechanic actuators. For this purpose, an integrated rig has been developed based on the existing C295 platform. At this rig several systems are integrated such as: electrical, hydraulic, flight control system, cockpit, displays…

A project inside Cleansky 2 program is related to the implementation of VISTAS in the CS2 integrated rig. A call for proposal has been launched for this purpose.

Objectives

The objectives of this work are the following:

- To create a real demonstrator for VISTAS concept based on existing test means, and combining hybrid and real bench environments.

- To implement VISTAS protocol in a real bench with different signals and avionic buses to be virtualized such as A429, CAN, analog, discrete, serial, RVDT interfaces and including in addition simulation models.

- To assess the differences between real integration and virtual integration through VISTAS, specifically addressing the impact produced by the I/O or bridge boxes during conversion between real and virtualized environment.

- To develop the VISTAS capability into current Test System Supports available at the bench. In Cleansky 2 integrated bench are two different test systems support, one is an "In-house" solution called SEAS and the other is a COTS solution provided by National Instruments.

- To establish a VISTAS based framework that allows to connect CS2 integrated bench with other existing benches at different locations (ex. avionics benches)

- To foster the OPEN architecture concept and connect different equipments/tools not present in the current bench that provides VISTAS interfacing functionality (UEI, AI-DASS, SANDRA…)

Cleansky 2 integrated rig bench architecture

The Cleansky 2 integrated rig is composed by the following test means (see Fig. 3):

- Electrical rig, at which generation, conversion and distribution is integrated and tested shaping the electrical power generation system of the aircraft.

- Flight Control Computer bench, at which the two Flight Control computers are integrated and tested.

- Flight Test Instrumentation bench, at which the Data Acquisition System installed at real aircraft is integrated at rig environment.

- Flap Test Bench, at which High Lift Control Unit computer is integrated and tested.

- Cockpit and displays bench, used for cockpit systems integration: trim actuator, springs, displays; and for handling qualities evaluation, flight control laws evaluation and emergency interim procedures evaluation.

- Surfaces benches (for primary and secondary surfaces) at which integration between computer and actuator is achieved.

- Hydraulic rig for hydraulic system integration.

All of these benches can be operated in stand-alone mode, with the aircraft environment external to the equipment or system under test completely simulated, or in integrated mode, by interconnecting the benches through real aircraft wiring. In addition, an additional level of integration can be reached by interconnecting the benches in terms of electrical and hydraulic power.

Using VISTAS approach all the benches are interconnected using Ethernet with VISTAS protocol implemented (see Fig. 4). The I/O boxes are based on the current Test System Support HW, as indicated before combining both "in-house" and COTS solutions, so it has not been planned to develop specifically any HW devoted to VISTAS.

The "in-house" is based on SEAS and the COTS is based on National Instruments Compact Rio and PXI (this last for electrical rig).

The SW deployed on all these platforms is modified with the implementation of the VISTAS protocol for interconnection of the different benches.
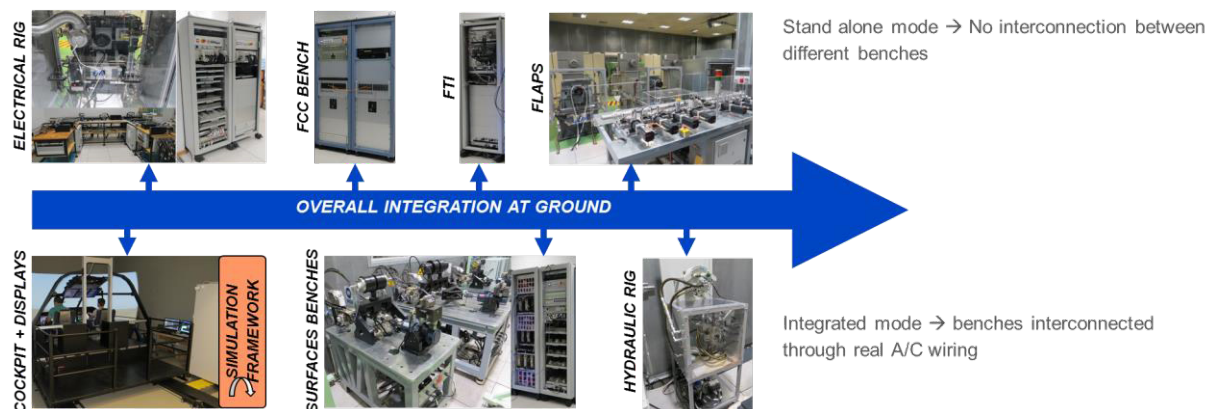
Fig. 3.   Cleansky 2 Integrated Rig.
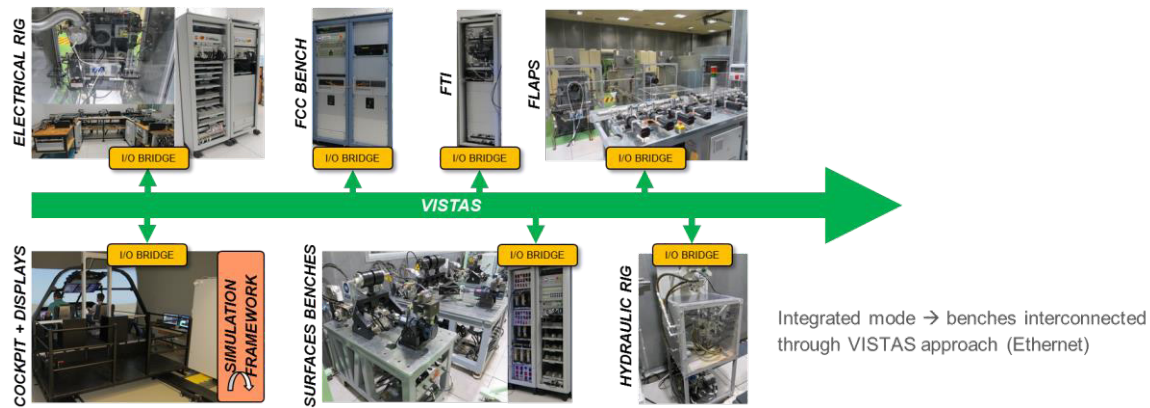
Fig. 4. Cleansky 2 Integrated Rig with VISTAS protocol implemented.

SW components diagram

Every bench which is part of CS2 integrated rig has its own Test Support System, providing a distributed Test Support System when all the benches are working in integrated mode. The implementation of VISTAS is based on the development of a dynamic link library which is able to convert the Test Support System present at every bench into a I/O Bridge.

The objective of VISTAS implementation is to keep physically all the benches in stand/alone configuration, in other words, without considering all the interconnections between benches at signal level, and substitute this by an interconnection through VISTAS protocol.

The dynamic link library with some functions has been developed in C++. The dynamic link library which has been developed is platform agnostic, as it can works with the Windows platform present at "in-house" test support systems such as SEAS, or it can works also with Linux platform present at National Instruments COTS platform.

The working principle is the following (see Fig. 5):

- The I/O Bridge interacts with the real aircraft equipment through signal/bus input output stage.

- The I/O Bridge is provided with the ECIC file, at which the VISTAS configuration is stated. It is necessary to remark that for complex benches this ECIC file (one per I/O Bridge) may be provided with the aid of a Resolver SW application. In our case it has be done manually without resolver application. In the ECIC file is included the following information: IP, ports, channel assignation, packetization strategy, timestamps, and header… all the data necessary to virtualize the signals in an optimized way.

- Using the different functions of the dll the frames are generated and passed through Ethernet over VISTAS protocol.

Of course, this working principle is bidirectional.
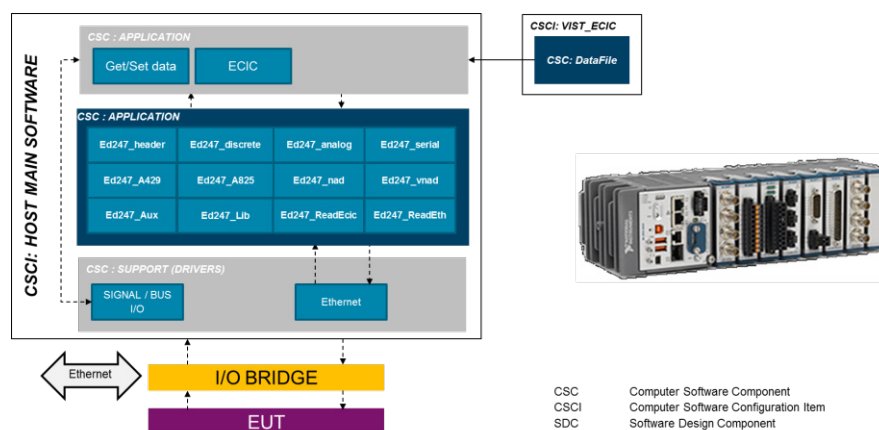
Fig. 5. SW components diagram.

Status of the activities

The current status regarding VISTAS implementation at CS2 Integrated Rig is the following:

- The development of the ED247 library is done for the avionic buses used at CS2 integrated rig, which are: A429, CAN BUS, Analog, Discrete, and RVDT.

- Generation of configuration files is done in a manual way.

- VISTAS performance testing activities are taking place at CS2 integrated lab and other available benches

- Implementation on existing HW is achieved at National Instruments cRIO platform, but it is still pending other platforms present at CS2 integrated bench such as National Instruments PXI and PCI platform at SEAS "in-house" test system support

- And finally, the pending activities are Monitoring/recording SW, development of resolver SW and bench manager SW is pending

Performance tests

A429 Virtualization through VISTAS has been tested at CS2 Integrated lab with different configurations as indicated in the table (see Fig. 6), by the interconnection through VISTAS of different aircraft equipments.

Current National Instruments compact RIO present at the bench has been used as VISTAS I/O bridges.

No packetization strategy has been considered, which means one UDP packet per A429 label.

In all the cases the latency average value is very low if it is compared by the update rate of the aircraft equipment A429 buses, which in principle enables the multisystem integration, without significant delays that can affect to the behavior of the system/multisystem under test.

It has been observed different latencies that are explained by the different model of switch used for the test.

Test case	Direction	Nº of buses	Speed (Kbps)	Labels	Rate (ms)	Latency (us)	Switch
BCU/HERCU	A	2	12,5	4	16	750	LinkSys SD205
				6	128	750	
				2	1024	750	
	B	4	100	9	16	750	
				2	1024	750	
ACE/ACE	A	2	100	4	8	500	CISCO 2950
	B	2	100	4	8	500	
FCC/ACE	A	1	100	3	16	500	CISCO 2950
			100	1	32	500	
TEFU/PU	A	1	100	14	250	500	CISCO 2950

Fig. 6.   A429 performance tests.
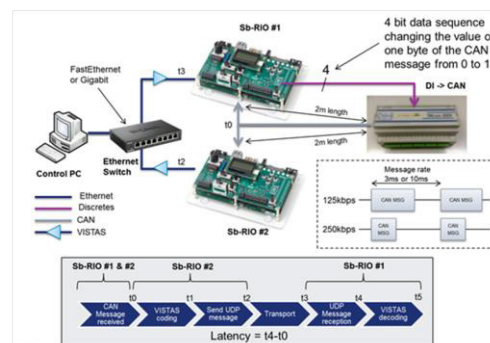
Fig. 7.   SW components diagram.

CAN Virtualization through VISTAS has been tested, using a National Instruments SB-RIO demonstration kit as VISTAS I/O bridge, for hosting the VISTAS SW (see Fig. 7).

Different CAN BUS speeds, from 125Kbps to 1000kbps, and different message rates (3msec and 10msec) have been considered to set up the test case list. For each test case 10 tests have been considered with duration of 10 seconds.

In all the cases the latency average value is constant and independent on the CAN bus speed.

Conclusion

And finally and as a conclusion, the key messages based on our expectations related to integration with VISTAS are summarized:

- It has been presented the VISTAS concept as a future concept for bench architecture, as a tool for virtual testing

- It has been provided an overview of the VISTAS implementation at CS2 integrated Rig, and its current status, focusing on the performance tests performed for A429 and CAN buses

- It can be evaluated the necessity of real aircraft wiring for integration. If finally is necessary, at least, VISTAS integration provides the possibility of delaying real aircraft wiring manufacturing at rig until a reasonable level of maturity has been reached through VISTAS integration, which will contribute to cost reduction as the number of modifications and/or reworks over real aircraft wiring will be very limited. If finally real aircraft wiring is not necessary, the next step is to check if certification credits can be obtained from VISTAS approach that is not 100% representative of aircraft, but provides integration capabilities.

- It can be assessed if the signal exchange between different elements is representative using VISTAS, because at the end, the savings in terms of wirings, patch-panels, and fault insertion HW devices… are based on bidirectional conversions from acquired/issued signals to VISTAS protocol, and the conversion itself provides some delay.

References

[1] EUROCAE ED-247A: Technical Standard of Virtual Interoperable Simulation for Tests of Aircraft Systems in Virtual or Hybrid Bench (March 2020)

A New Flexible Photogrammetry Instrumentation for Estimating Wing Deformation in Airbus

Quentin Demoulin[1,2], François Lefebvre-Albaret[1], Adrian Basarab[3], Denis Kouamé[3],
and Jean-Yves Tourneret[2]
[1] Airbus, 316 route de Bayonne, 31060 Toulouse, France,
[2] IRIT/ENSEEIHT/Tésa, University of Toulouse, 31071 Toulouse, France
[3] IRIT UMR CNRS 5505, University of Toulouse, Université Paul Sabatier, CNRS, France
email: quentin.demoulin@airbus.com

Abstract:

As part of aircraft certification and optimization, wing bending and twist measurements are performed under various load cases (aircraft weight, speed, angle of attack, etc.) to validate and improve wing deformation models. Since these measurements are acquired during flight, their analysis requires to face strong environmental constraints. Indeed, the highly varying luminosity conditions, the presence of possible reflections or shadows, the vibrations and the deformations of the entire aircraft, are strong constraints that need to be considered carefully.

Current approaches applied in Airbus are based on inertial measurement units installed inside the wing, or on photogrammetry-based solutions using calibrated sensors and retro-reflective targets located on the wings. These methods are not only highly intrusive, but also require time-consuming installation, calibration phases and dedicated flights to produce only sparse measurements. Moreover, the use of reflective targets on the wing has an impact on the wing aerodynamic, which should be avoided.

In this paper, we investigate a new method for estimating wing deformations. This method adapts a photogrammetry approach classically used for reconstructing buildings or art structures to the aircraft environment. To this aim, we propose to use synchronous videos from high resolution cameras, which can be easily installed on the aircraft windows and on the vertical stabilizer. Appropriate features are extracted from the images acquired by these cameras, related to wing joints or reference points located on the aircraft wing. The system uses these features to autonomously recalibrate itself at each frame and provide a dense 3D reconstruction of the wing in the aircraft reference coordinate system. Some experiments conducted on real data acquired on Airbus aircrafts show that the proposed estimation method provide promising results.

Key words: Photogrammetry, Wing Deformation, Flight Tests, A350

1. Introduction

Theoretical models of aircraft deformations are developed to predict wing fatigue and behavior in all flight conditions, simulating the wing deformations for varying scenarii depending for example on aircraft weight, speed, or angle of attack. These models are built on the Finite Element Method (FEM) reflecting the mechanical properties of the aircraft. However, theoretical models may slightly differ from the real aircraft behavior. Therefore, in aircraft certification procedures, it is necessary to measure the 3D deformations of wings to assess and further correct the models. This paper considers the specific case of wing deformations, which is important for many practical applications.

First of all, one could instrument the wing of interest to acquire the measurements. However, placing sensors inside the wing requires difficult installation, and provides only few measurement points. Another solution would be to install sensors on the wing surface, but this operation should be conducted carefully, by avoiding to introduce disturbances in the wing aerodynamic properties. For these reasons, we consider in this study a non-contact 3D measurement method, based on multiple view photogrammetry form video cameras. It is worth mentioning that LIDAR (Light Detection and Ranging) reconstruction is also a potential good candidate for the problem addressed in this work.

The use of cameras for wing deformation estimation faces several constraints. On the one hand, camera positioning is restricted presently since the only available positions are on aircraft

windows and on the aircraft rear stabilizer. Moreover, due to the geometry of the aircraft, cameras on windows with grazing views of the wing, should be close enough to guarantee a good overlap between images, which is not ideal for photogrammetry triangulation. Indeed, the optimal distance between cameras should be close to the distance between each camera and the observed object. Finally, we must also take into account that cameras will slightly move during the flight as the whole aircraft is deforming.

On the other hand, our system should be robust to varying environmental conditions. Indeed, luminosity variations during the flight, potential shadows or reflections on the wing surface strongly impact the 3D reconstruction method. For instance, Fig. 1 illustrates the impact of luminosity variations on some real images, showing the variability of the observed images.

This paper is structured as follows. Section 1 presents the current methods applied in Airbus to provide the wing deformation measurements. The proposed new photogrammetry approach is introduced in Section 3. Results obtained on real Aircraft measurements are reported in Section 4. Conclusion and perspectives are finally reported in Section 5.
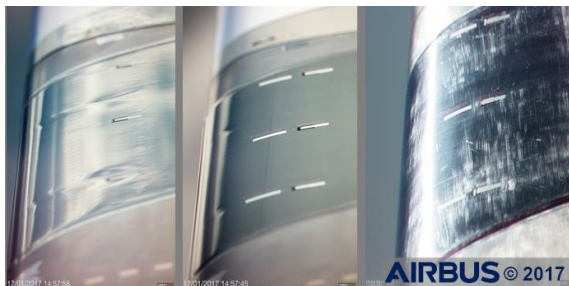
Fig. 1. Several views of the same wing leading edge under various illuminations.

2. Current Reconstruction Method

To estimate wing twist and bending, recent aircraft used in Airbus flight tests consider inertial motion units (IMU) installed inside the wing. These systems can deliver real time and accurate measurement of the wing bending and twist with respect to a reference inertial unit at the center of the aircraft. However, IMUs are not always available, and they only produce sparse measurements. Indeed, only a few IMUs can be installed inside the wing as they require difficult installation. Therefore, for some specific needs, Airbus also uses a photogrammetry approach.

2.1. Photogrammetry

The photogrammetry system used by Airbus to estimate wing deformation is based on retro-reflective targets stuck orthogonally to the wing

surface, observed by cameras installed on windows. The estimation approach is similar to the studies presented [1] and [2] (see Fig. 2 for illustration). To limit distortions, standard windows in front of the cameras are replaced by metrological ones. During flight, cameras are triggered synchronously with flashes, illuminating the targets to easily and accurately detect targets. Finally, the accurate position of targets and cameras is initialized through a long calibration phase, and cameras are attached such that that their locations remain fixed with respect to the aircraft.
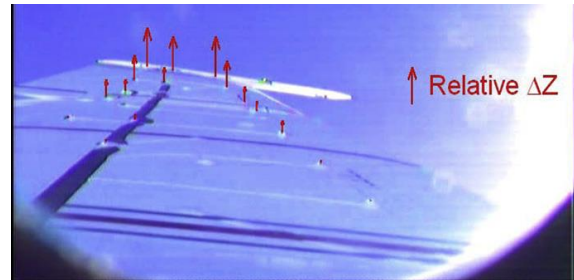
Fig. 2. Example of an aircraft wing imaged using a photogrammetry system (from [2]).

Though this system provides very satisfying results during tests, it has several drawbacks. First, the use of protuberant targets on the wing induces aerodynamic disturbances, which can alter the aerodynamic properties of the wing and reduce the flight domain of the aircraft, thus enforcing the need for dedicated fights. Secondly, to reduce the camera motions during flight, the mechanical fixations of the cameras are intrusive and not easy to install. Another drawback is the time consuming calibration step required for the system, which could be lightened. Finally, the method provides only sparse measurements associated with target locations, and the necessary time to recharge our flashes does not allow us to record several images per second.

2.2. Other Studies

Other investigations in the domain of contactless 3D reconstruction were perform in Airbus. In this field, we can mention the IPCT (image pattern correlation technique) which was tested on aircraft A380 [15] and A350. Using pairs of cameras, this technique combines image correlation and photogrammetry algorithms to recover the 3D shape and deformations of a surface. To capture image correlations, the method relies on the presence of image texture. Thus stickers of random pattern are stuck on the wing surface (see Fig. 4). Deflections of the surface are calculated using a simplified deformation model of the wing as function of a normal load under static conditions.

Another explored method is based on deflectometry, which was tested in the A340 BLADE (Breakthrough Laminar Aircraft Demonstrator in Europe) project [3].The principle of this approach is to consider the reflection of a known pattern on a surface, and recover the local surface deformations from the reflected pattern deformations. To use this approach, strong assumptions on surface properties must be made (geometry, reflecting properties), which is only feasible within very specific applications. In the frame of the BLADE project, a pattern was painted on the wing edge, reflected on the wing, and reflections were observed by cameras installed at the wing tip (see Fig. 3). To ensure good reflection of the pattern, a specific paint was used on the wing surface.

Fig. 4. Example of pattern stripe installed for IPCT tests on A380 in 2009.

Note that both methods gave interesting results, but the required installations are intrusive and onerous, and hence should be avoided.

3. Proposed Photogrammetry Approach

In this Section, we describe a new photogrammetry approach that cope with the principle drawbacks of the previous methods. This method operates a set of cameras installed at window positions and another camera located on the vertical stabilizer. The wing deformations are then estimated by recovering the 3D coordinates of points located on the wing surface, using a generic photogrammetry method adapted to the specific case of flight tests.

3.1. Photogrammetry Principle

The basic concept of photogrammetry is to recover the 3D position of points from their observations on images. To better understand its concept, some mathematical principles are recalled.

We observe a set of 3D points $(X_i)_{=1,...,N}$, with N the number of points, using a set of M cameras (depicted in Fig. 5). Each camera has its own coordinate system with respect to that of the aircraft. The 3D points are expressed in the aircraft coordinate system. To transfer the points from
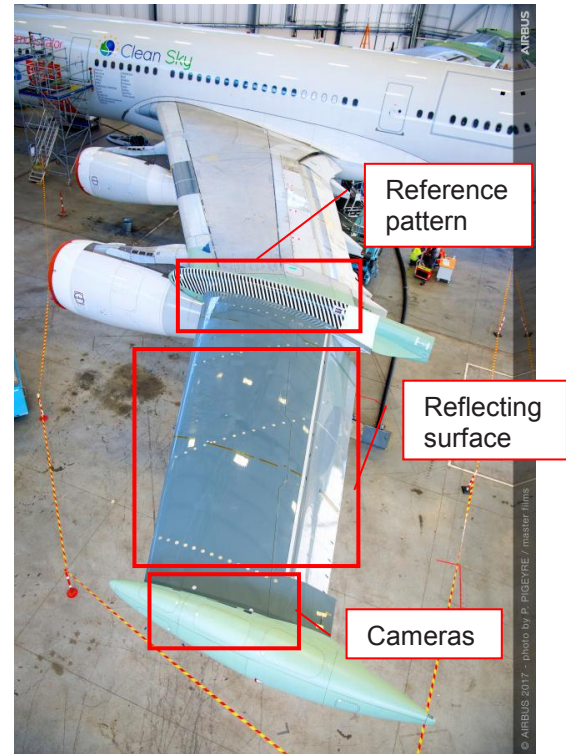
Fig. 3. BLADE wing with deflectometry installation.

this system to the camera one, we use the camera transfer matrices $(P^j)_{j=1,...,M}$, which represent the extrinsic parameters of the M camera (rotations and translations). Then 3D points are projected from camera to image coordinates using the matrices $(K^j)_{j=1,...,M}$, representing the camera intrinsic parameters (focal, sensor center, pixel size). These matrices can be estimated from a calibration phase. The resulting 2D points can then be expressed as a function of the 3D points and camera matrices:

$$x_i^j = f(X_i, P^j, K^j)$$

where x_i^j and K^j are known quantities. To recover the 3D points, several solutions can be considered.

First we could choose for P^j some known constants, which is done in the classical approach (Section 2.1), or introduce constraints on the 3D point coordinates (for instance, we could expect points to lie on non-deformable solids). However, in this study, we seek to keep our system as flexible as possible. Consequently, we use a more generic approach, based on bundle adjustment [4]. To this aim, we work in the camera coordinate system, and simplify this system to ensure that matrices P^j only represent rotations and translation from camera to camera. Then, we find simultaneously P^j and X_i that minimize the reprojection error, i.e., the distance between

the estimated point projection $\hat{x}_i^j$ and its observation x_i^j on the image. This leads to the following minimization problem

$$\min_{X_i, P^j} \sum_{i,j} [x_i^j - \hat{x}_i^j(X_i, P^j)]^2. \qquad (1)$$

Note that bundle adjustment is not convex and thus relies on a correct initialization. Moreover it requires a set of 2D observations and does not directly outputs the 3D coordinates of estimated points in the aircraft coordinate system. These issues have to be accounted in the method, and solutions are given through the processing chain.
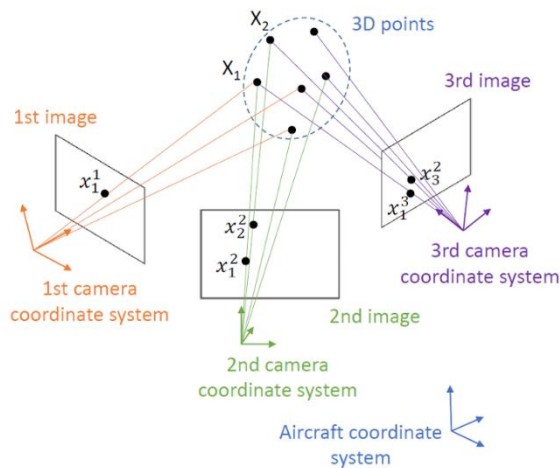
Fig. 5. Diagram of the multiple view photogrammetry for 3 cameras.

3.2. Algorithmic Chain

The algorithm chain is summarized as follows:

i. Camera and scene calibration

ii. Image acquisition and undistortion

iii. Observation detection

iv. Bundle Adjustment

v. Registration

vi. Parameters update

Step (i) is performed before flight and step (ii) to (v) are repeated for each new camera acquisition.

Step (i): The camera calibration is performed using pictures of a chessboard from different angles and distances, and using the OpenCV calibration functions [5], which outputs the intrinsic parameters of the cameras and distortion coefficients. The scene is then calibrated to initialize the positions of the 3D points and cameras. This step is conveniently performed using pictures from a drone and the photogrammetry software Metashape [6] (illustrated in Fig. 6). Note that a

drone can take pictures of the full scene from a wide range of angles in less than 30 minutes.
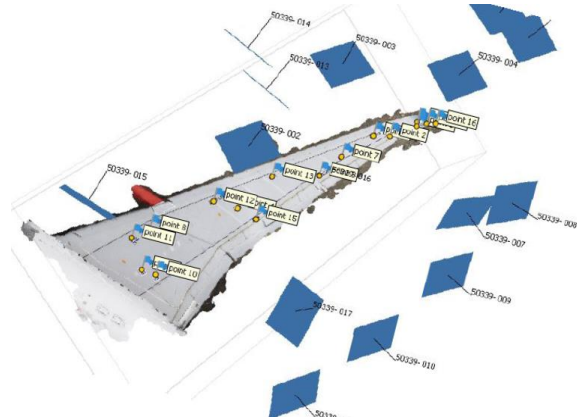
Fig. 6. Initial 3D reconstruction using the Metashape software and images from a drone. Images are displayed in blue at their point of view.

Step (ii): For each new image acquisition, we use the OpenCV undistortion function with the distortion parameters estimated in Step (i).

Step (iii): We propose three different ways to detect the 2D observation of the 3D points. First, we can use the natural features existing on the wing surface (corners, joints, writing, etc.). We find points with strong information using the ORB detection algorithm [7] and describe these points using a combination of ORB, SIFT [8][7] and DAISY [9] descriptors. Descriptors basically attribute a signature to a point based on the intensity values of it surrounding pixels. Descriptors are then compared between images to find matching points (see Fig. 7), which is performed in our case using the FLANN algorithm [10]. To increase the number of detected points, we use the guided matching algorithm described [11].
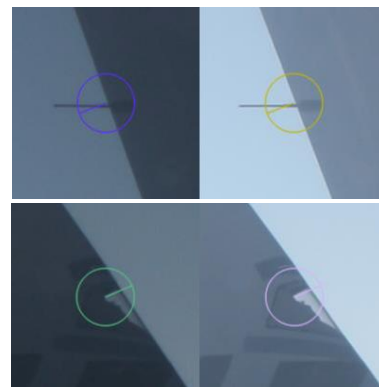
Fig. 7. Example of natural features of the wing detected and matched on a pair of images. Left and right images correspond to two distinct cameras.

The second approach to generate observations consists in exploiting the presence of black lanes

along the wings of most aircraft. To do so, we rectify images using camera geometric properties estimated from a first set of corresponding points [14]. Image rectification allows the horizontally alignment of the corresponding points in a pair of images (see Fig. 8). Once images have been rectified, matching points can be found by searching lanes in image lines, which can be done using sub-pixel line detection methods [12][13].
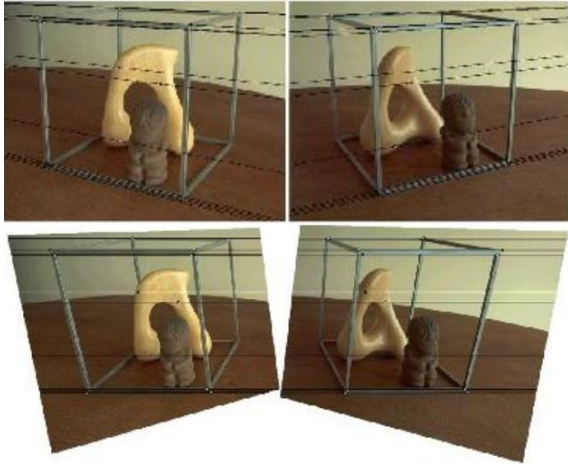
Fig. 8. Example of stereo image rectification (from [14] Zhang). Top – original pair of images, bottom – rectified images.

The third approach, when possible, is to use markers stuck on the wing surface. Markers allow an easy and accurate localization of the observations. A more detailed description of the markers is given in Section 4.1.

Step (iv): The initialization from step (i) and the set of 2D observation from step (iii) are used as input to the bundle adjustment, in order to minimize the reprojection error (1). The algorithm outputs the optimum 3D points X_i and the camera matrices P^j in the coordinate system of the cameras.

Step (v): The registration phase is performed using the rear camera. This camera can observe reference points close to the aircraft fuselage (see Fig. 9), which are considered as quasi-static during the flight. By detecting these points on the rear view image, we can recover the transfer matrix from the rear camera to the aircraft coordinate system. Since the rear camera is also part of bundle adjustment in Step (iv), we also know the transfer matrix from this camera to the global camera coordinate system. Finally these matrices are combined to transfer the 3D points and cameras to the aircraft coordinate system.

Step (vi): The initial parameters are updated, taking the output of Step (v) as knew initial positions of the algorithm.
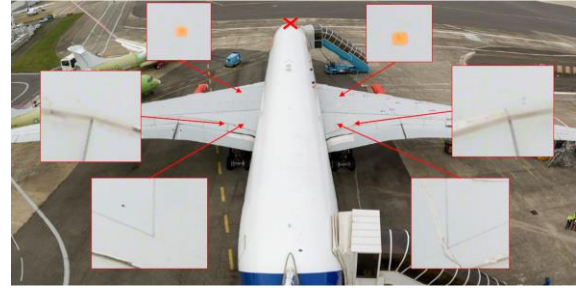

Fig. 9. Rear camera view with some reference points zoomed.

3.3. Constrained Bundle Adjustment

The method described in the previous section enables the 3D shape of the wing to be reconstructed, for any camera and marker motion, using no prior knowledge about wing deformations.

In this section, we go further and improve the performance of the method by introducing prior knowledge about the aircraft mechanical limits. We take benefit from the FEM of the wings to define limits corresponding to the ultimate loads. Assuming that the wing would break in such extreme cases, we consider that they are not possible during the tests. Thus, we construct a set of constraints that we inject in the classical bundle adjustment, such as local bending or elongation constraints (see Fig. 10). The resulting constrained bundle adjustment is constructed by adding a penalty term to the classical reprojection error.
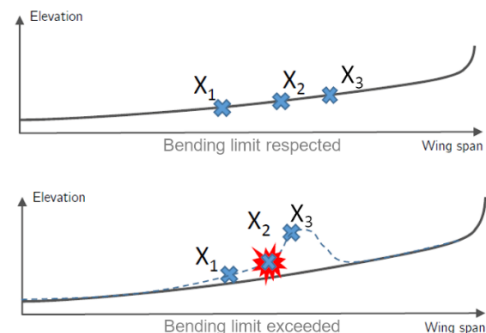
Fig. 10. Illustration of the local bending constraint: Top - constraints is respected, bottom - constraints is exceeded, and thus will be activated in the constrained bundle adjustment.

4. Experimentations and Results

This section first describes the system installation for flight and presents some results obtained from tests conducted on ground.

4.1. Preparation for Flight Tests

Our aim is to operate and validate our system on an A350-1000, whose wing is equipped with an IMU that will give a ground truth.

The test was conducted using ruggedized cameras, offering a 31MP sensor with global shutter, while preserving small dimensions (about 60x60x60 mm). Metrological windows were used instead of classical ones to prevent optical distortions. The shape of these camera, combined with our method, which is adapted to camera motions, allows an easy installation of the cameras on the aircraft window using simple 3D printed fixations. Cameras are placed as depicted in Fig. 11. Cameras 1 and 4 look at the wing end half, camera 2 and 3 look at the engine to tip section, and the rear camera 5 looks at the fuselage and the wing beginning.
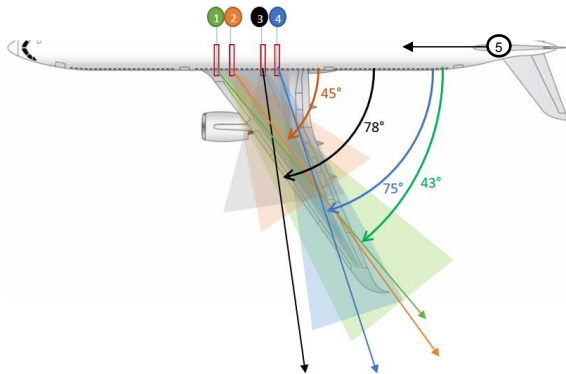
Fig. 11. Positions of the 5 cameras for the tests.

For the acquisition and data storage, dedicated computers were prepared with suited coaxlink cards and synchronization cards. During tests, data are recorded directly on the computer using the StreamPix software.

On the wing side, for robust and accurate detection, we use quadrant markers printed on stickers using matte ink. These markers can support flight conditions and the matte format allows correct detection for any illumination. Elongated stickers of various sizes were used depending on their positions onto the wing to ensure a similar aspect ratio when seen by the cameras (see Fig. 12 and Fig. 15). The detection can reach sub-pixel accuracy [16].

This setup was used in different experiments conducted on ground to evaluate the relevance of the proposed system.
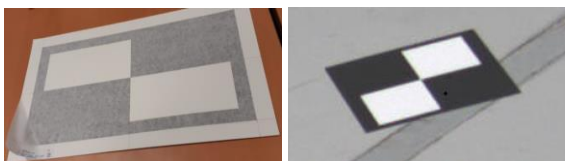
Fig. 12. Stickers of quadrant markers. Left – close view before installation, right - aspect of the target from camera view.

4.2. Test on Ground

The new photogrammetry system was evaluated on ground on an A350-900. The devices were installed in order to reproduce in-flight installation. We installed our markers on the wing surface, plus tape graduations on the two black lanes of the wing, at about every 30cm (shown in Fig. 13), and four 4K cameras on an aircraft window. The cameras were placed about one meter above the windows to produce similar views that would be obtained at a window position with the wing bent (see Fig. 14). Examples of the acquired views are displayed in Fig. 15. Note that the wing and the cameras were not moving during this test. However, several pictures of the wing were acquired during three days to see the impact of the luminosity on the reconstruction. As expected, we monitored illumination variations and moving reflections on the wing, as visible in Fig. 15 for camera 3. This induced the failure of correct detection of the graduation located at the wing end. However, quadrant markers remained always detectable.
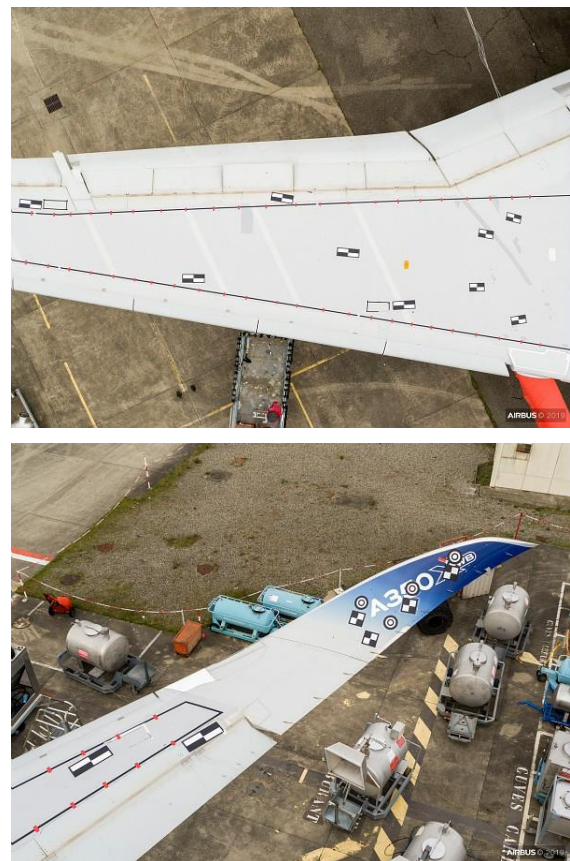
Fig. 13. Markers and graduations installed on the A350-900 wing.

To initialize the algorithm presented in Section 3.2, we triangulated the 3D positions of reference points on the wing, using images taken from a drone (see Fig. 6). We shook manually

the wing at its tip to produce vibrations confirming the behavior of the system for small deformations. The amplitude of the vibrations was estimated to about 5cm using a scale board.

The proposed algorithm was implemented in Python and tested on different images. Quadrant markers where detected and their 3D coordinates were estimated and registered in the Aircraft coordinate system. Fig. 16 shows the results of the algorithm superimposed on the reference 3D shape of the wing (on ground). A ground truth was not available for this test and therefore we cannot determine the accuracy of the reconstruction. However, the results seem to correctly coincide with the 3D reference shape.

The study of mechanical limits applied as constraints on the bundle adjustment has also shown promising results. Fig. 17 displays some reconstruction results for markers with and without a volume constraint. This constraint states that points in the aircraft coordinate system can move only in a cylinder with a 5cm radius, oriented towards the z axis. Using the volume constraint allows us to obtain estimated displacements of the wing tip close to 5cm, which is coherent with our observation using the scale board.

5. Conclusion

This paper presented a new photogrammetry method for estimating a wing deformation. This method bypasses some limitations of previous approaches, offering easy installation, more flexibility, dynamic mapping, with no dedicated test, and delivering more accurate estimations of the wing deformations. The proposed method is a flexible adaptation of the classical multiple view photogrammetry for the specific case of 3D deformation estimation. Our algorithm goes even further by introducing mechanical limits in the process to improve the reconstruction accuracy. The system has been prepared for flight tests and tested on ground, where experimentations delivered promising results. The next step will be to evaluate the system in flight with countermeasure. Future work will focus on the detailed assessment of the metrological accuracy of the system as a real sensor.
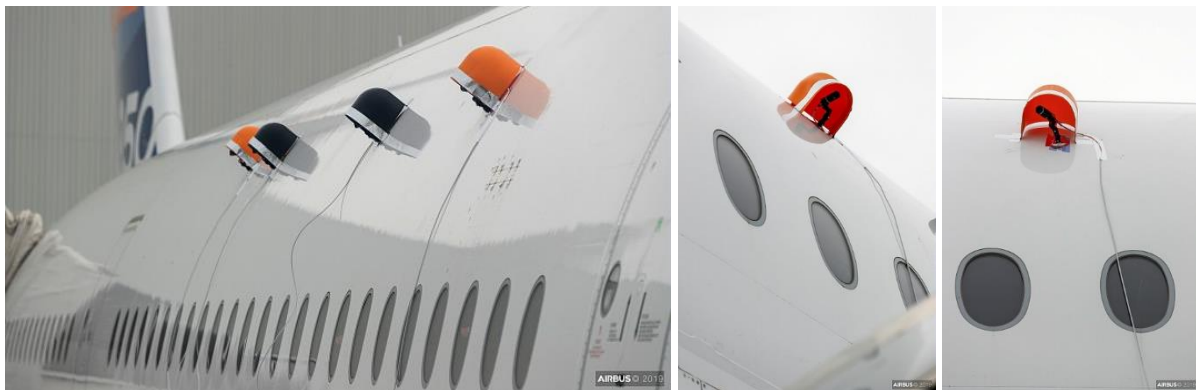
Fig. 14. Camera installation above windows of an A350-900.

Fig. 15. Example of views from the camera during the test on ground. Left - camera 1, right- camera 3.
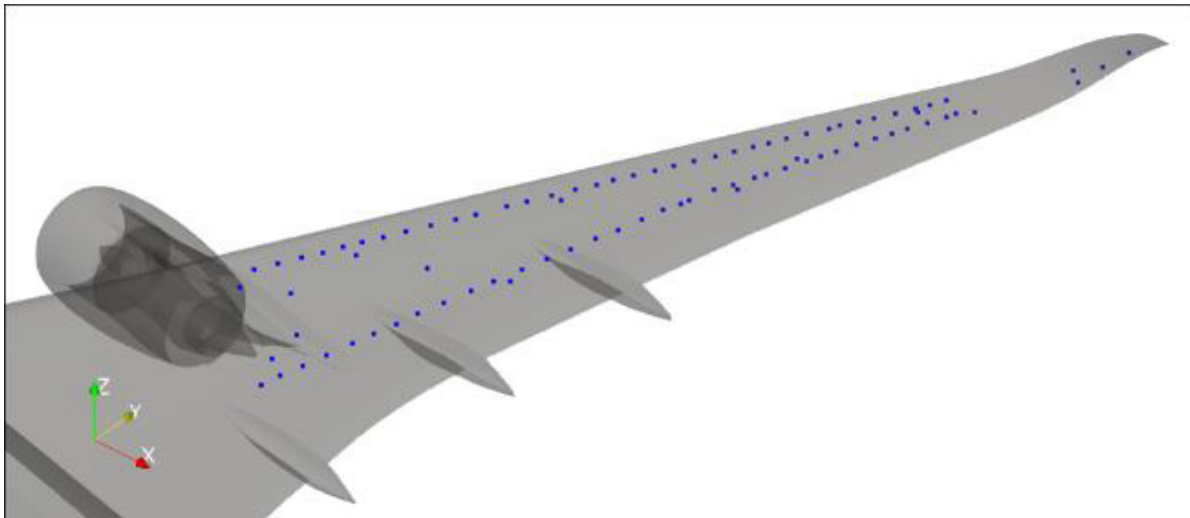
Fig. 16. Results obtained with the proposed algorithm. Blue dots indicate the 3D reconstructions of graduations and markers located on the A350-900 wing (that have been superimposed on the 3D shape reference of the wing).
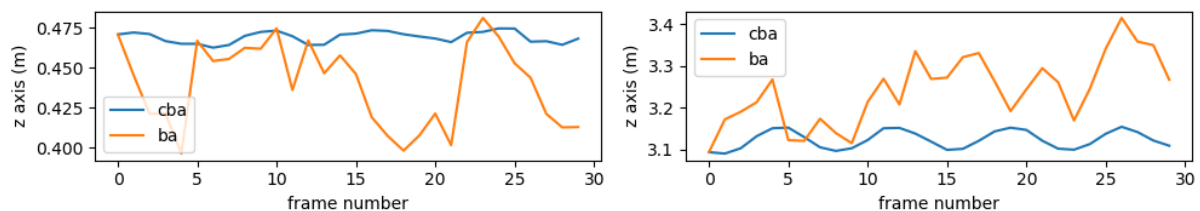
Fig. 17. Estimated bending of two markers versus time while the wing was shaken. The result of the algorithm without constraint is displayed in blue (ba) and the result with a displacement constraint is displayed in orange (cba). Left - marker at the middle of the wing, right – marker on the wing tip.

References

[1] D. Barrows, "Videogrammetric model deformation measurement technique for wind tunnel applications," in *Proc. 45th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nev., USA, Jan. 2007; doi: 10.2514/6.2007-1163

[2] T. Liu, A. W. Burner, T. W. Jones, and D. A. Barrows, "Photogrammetric techniques for aerospace applications," *Progress in Aerospace Sciences*, vol. 54, pp. 1-58, Oct. 2012; doi: 10.1016/j.paerosci.2012.03.002

[3] "Airbus BLADE" laminar flow wing demonstrator makes first flight" (Press release). Airbus, 26 Sep 2017. Available at: https://www.airbus.com/newsroom/press-releases/en/2017/09/airbus_-_blade_-laminar-flow-wing-demonstrator-makes-first-fligh.html

[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2004; isbn: 978-0-511-18618-9

[5] Intel, *The OpenCV Reference Manual*, 2.4.13.7 edition, Dec. 2019; See http://opencv.org/

[6] Agisoft LLC, St. Petersburg, Russia, *Agisoft Metashape User Manual: Professional Edition*, Version 1.6, 2020, available at: https://www.agisoft.com/downloads/user-manuals/.

[7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Computer Vision*, Barcelona, Spain, Nov. 2011; doi: 10.1109/iccv.2011.6126544

[8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91-110, Nov. 2004; doi: 10.1023/b:visi.0000029664.99615.94

[9] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *Trans. Pattern Anal. Machine Intell.*, vol. 32, pp. 815-830, May. 2010; doi: 10.1109/tpami.2009.77

[10] M. Muja and D. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proc. Int. Conf. Comput. Vis. Theory and Applications (VISAPP)*, Lisboa, Portugal, Jan. 2009; doi: 10.5220/0001787803310340

[11] R. Shah, V. Srivastava, and P.J. Narayanan, "Geometry-aware feature matching for structure from motion applications," in *Proc. Conf. on Applications of Computer Vision*, Waikoloa, HI, USA, Jan. 2015; doi: 10.1109/wacv.2015.44

[12] R. G. Gioi, J. Jakubowicz, J. M. Morel, and G. Randall, "LSD: a line segment detector*," Image*

Processing on Line, vol. 2, pp. 35-55, Mar. 2012; doi: 10.5201/ipol.2012.gjmr-lsd

[13] R. G. Gioi and G. Randall, "A sub-pixel edge detector: an implementation of the canny/devernay algorithm," *Image Processing on Line*, vol. 7, pp. 347-372, Nov. 2017; doi: 10.5201/ipol.2017.216

[14] C. Loop and Z. Zhang, "Computing rectifying homographies for stereo vision," in *Proc. Computer Society Conf. on Computer Vision and Pattern Recognition*, Fort Collins, Co., USA, Jun. 1999; doi: 10.1109/cvpr.1999.786928

[15] F. Boden, N. Lawson, H. W. Jentink, and J. Kompenhans, *Advanced In-Flight Measurement Techniques*, Springer Berlin Heidelberg, 2013; isbn: 978-3-642-34738-2

[16] A. M. G. Tommaselli and A. Berveglieri, "Measuring photogrammetric control targets in low contrast images," *Bulletin of Geodetic Sciences*, vol. 24, pp. 171-185, Jun. 2018; Doi: 10.1590/s1982-21702018000200012

Displays Validation Based on Image Processing

Israel López Herreros
Airbus Defence and Space, John Lennon S/N 28906 Getafe-Madrid, Spain
israel.lopez@airbus.com

Abstract:

Aircraft manufacturers integrate a lot of equipment from different suppliers and on some occasion this equipment are like black boxes, the only easy way to check the data is through special monitoring displays supplied with the equipment. On other occasions a real aircraft display needs to be validated. These displays can be very complex, big amount of symbols and very dynamics. The validation of these displays can be a very tedious work if it is done by hand.

This document presents a methodology to automate the validation of the aircraft displays based on image processing to extract numbers, texts and symbols. These displays are not completely standards in the market and they have not been created taking into account these image processing techniques, so it is almost impossible to develop a generic tool. A specific tool will be developed for each display but they will share common stages from the specific pipeline.

There is a lot of literature about image processing but what it is really important is the definition of the pipeline in order to have a robust and efficient data acquisition system based on image processing. This document will give an overview of the different stages and some pipelines applied in real use cases in Flight Test Airbus Defence and Space.

Key words: Image processing, computer vision, displays validation.

1. Acronyms, Abbreviations and Symbols

AC: Aircraft

CSV: Comma-Separated Values

2. Introduction

Aircraft manufacturers integrate a lot of equipment from different suppliers and on some occasion this equipment are like black boxes, the only easy way to check the data is through special monitoring displays supplied with the equipment. On other occasions a real aircraft display needs to be validated. These displays can be very complex, big amount of symbols and very dynamics. The validation of these displays can be a very tedious work if it is done by hand.

This document presents a methodology to automate the validation of the aircraft displays based on image processing to extract numbers, texts and symbols (see Fig. 1).
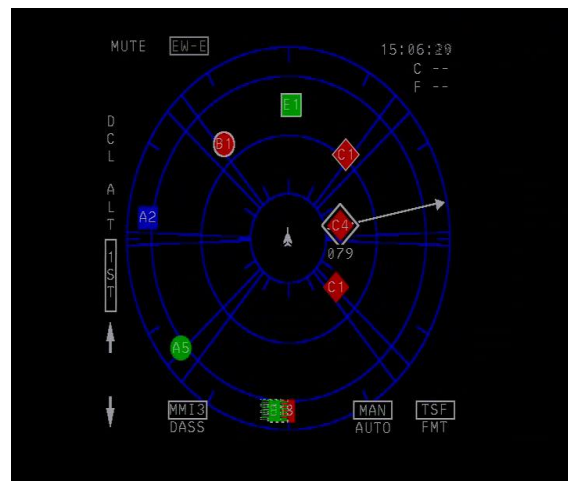
Fig. 1. Image of the aircraft display used for the prototype.

3. The Challenge

The methodology presented in this document will demonstrate the capacity to extract the screen position, color, text and shape of the symbols presented in a real AC display.

This display is not completely standard and it has not been created taking into account the

image processing techniques applied at a later phase.

Relative to this data acquisition phase several difficulties have been found and they will be described below.

The symbols have not a unique identifier. It makes it difficult to track two symbols with the same identifier in a similar screen position or when the symbols disappear for a while and they appear again in different positions (see Fig. 2)

Fig. 2. Non-unique identifier.

There are rendering problems in some frames. The origin of this problem is unknown but some hypothesis could be related to the graphics hardware or the codecs used for the recording video (see Fig. 3).

Fig. 3. Image with rendering problems.

There are different types of symbols overlapping that difficult the symbol recognition: Arrows that cross a symbol and several symbols overlapped (see Fig 4).

Fig. 4. Image with overlapping issues.

The blue color is reused for static background and some symbols. The colors in general are not completely pure, they have an interval of colors, i.e. when the blue symbols are analyzed different blue colors are found. These issues make the color recognition phase difficult (see Fig. 5).

Fig. 5. Blue color is reused in the display.

The objective is to minimize false positive with the introduction of high restrictions to avoid recognize positive symbols when they are not true, e.g. recognize C1 when it is not really C1.

Other objective is to reduce false negatives using the accuracy level, e.g. if a false negative symbol as BB is recognized it may be rejected because its accuracy is low.

4.   Prototype

To demonstrated the viability of this project a low cost prototype to extract data using image processing techniques from videos has been developed.

Searching in Google it can be seen that there are a lot of strategies to recognize text and numbers. Different algorithm evaluations based on Neural Network as the Tesseract [1] ocr library has been tested but all of them were very general and the accuracy wasn't enough for the requirements.

The philosophy applied in this prototype is to develop a specific tool, versatile, with well differentiated phases in order to have a general

strategy as much as possible but without compromising the accuracy.

OpenCV [2] has been selected as the computer vision library. The library is cross-platform and free for use under the open-source BSD license.

Python [3] has been chosen for rapid prototyping, departmental experience and for its good integration with OpenCV.

The pipeline chosen to solve the data extraction is described below:

First phase: Segmentation by color. The original image and the chosen color are the inputs of this phase. Due to the fact that the colors are not pure an interval has to be taking into account. The output of this phase is the image filtered by the chosen color (see Fig. 6).
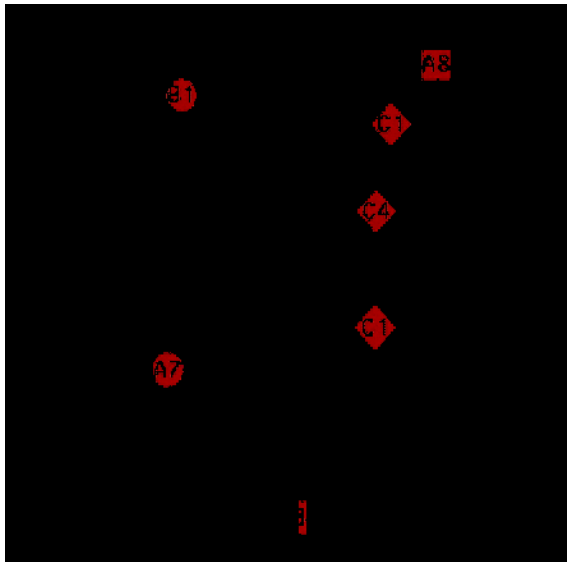
Fig. 6. Filtered image using red colors.

Second phase: Thresholding. The filtered image and the original image are the inputs of this phase. Two binary images [4] are generated, one for contours recognition from the filtered image by color (see Fig. 7.). This binary image is enhanced with a morphology close step [5] (see Fig. 8).
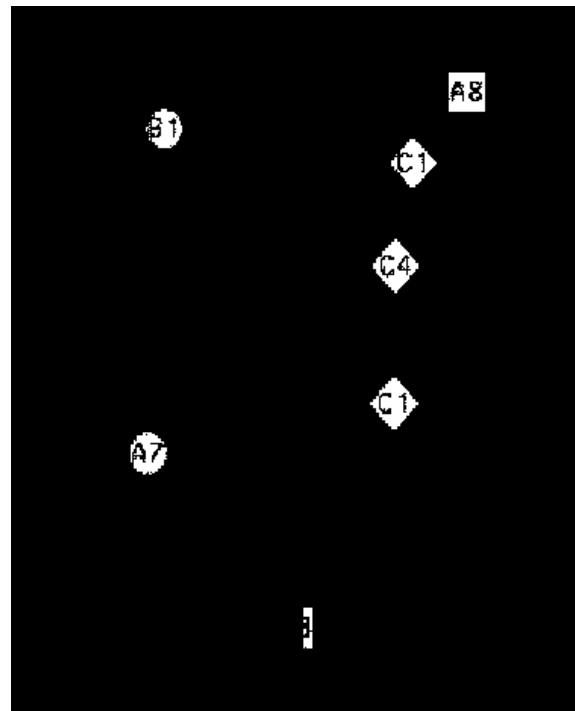
Fig. 7. Binary image for contours recognition.
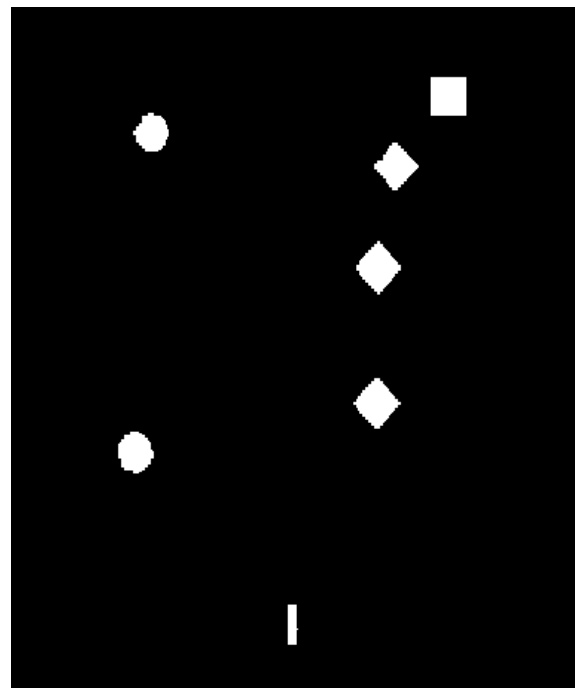
Fig. 8. Morphology close image for contours recognition.

A second binary image is generated from the original image for text and number recognition (see Fig. 9).
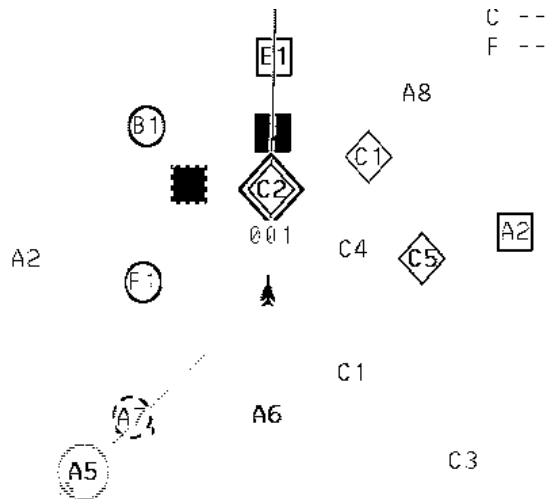
Fig. 9. Binary image for text/number recognition.

Third phase: Contours recognition. The morphology close image is the input of this phase. The contours found are filtered by the area calculated from the moments [6] of these contours. The contours are smoothed out with the convex hull [7] (see Fig. 10).
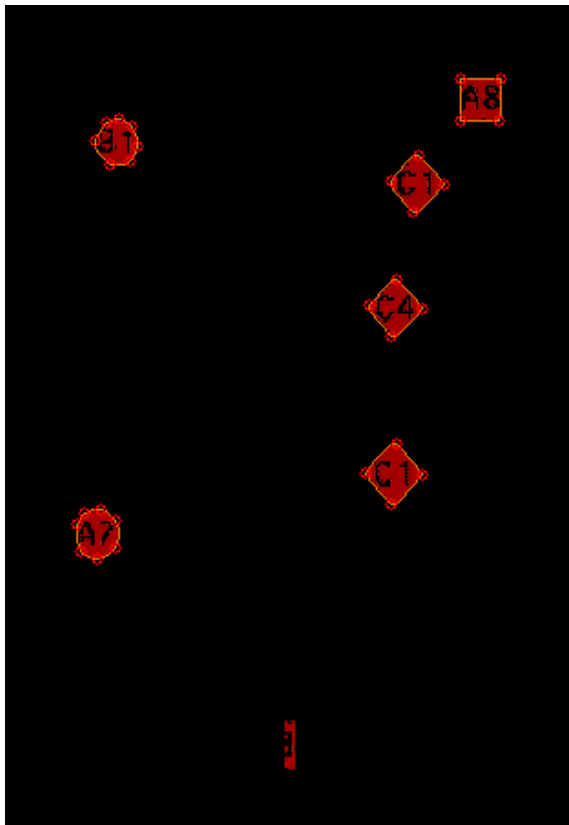
Fig. 10. Image with recognized contours.

Fourth phase: Shape recognition. The contours are the inputs of this phase. These contours are analyzed using geometry properties to identify if the contours correspond to circles, squares or diamonds. The outputs of this phase are the different shapes with the corresponding bounding boxes (see Fig. 11).
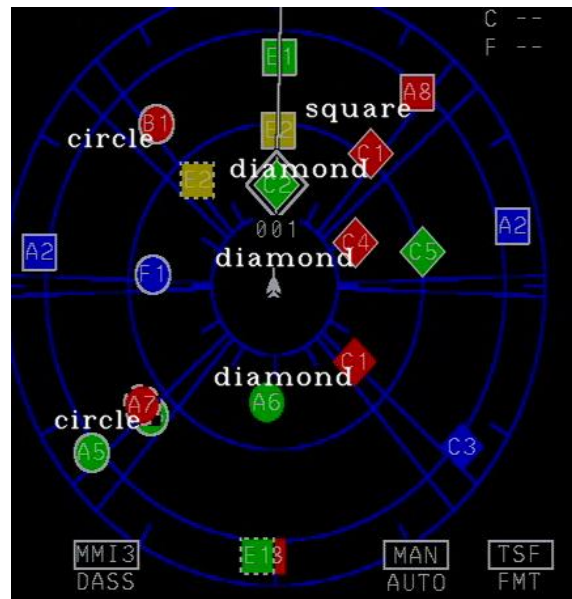
Fig. 11. Image with recognized shapes.

Fifth phase: Text/number recognition. The binary image (see Fig.9) and the bounding boxes are the inputs of this phase. A training dataset of all characters from the binary images is used in this phase (see Fig. 12).

Fig. 12. Example of training dataset.

Each digit from the boundary box is isolated based on geometry properties. For each digit a correlation algorithm is used against the dataset to find the characters that match with best accuracy. The outputs of this phase are the different recognized texts with their corresponding accuracies (see Fig. 13).
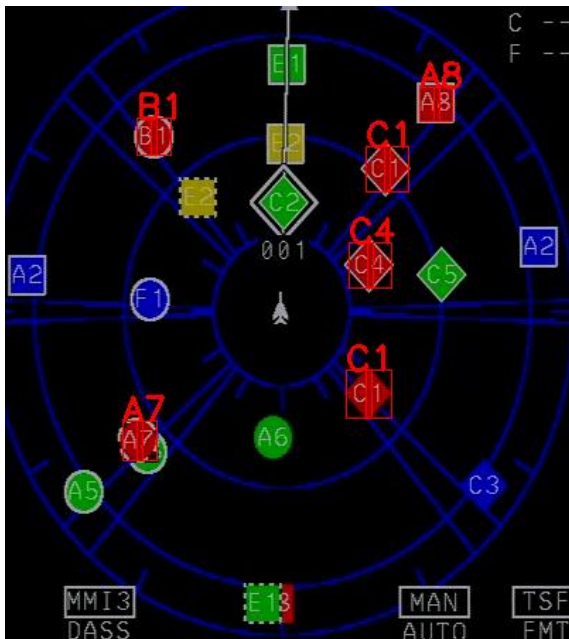
Fig. 13. Recognized texts in the image.

Sixth phase: Update symbol information in the corresponding series. The actual and the historical information are used to better identify the right series to add the new symbol information. The non-unique identifier makes this operation more difficult and some assumptions have to be taken. In our case a maximum displacement based on system information of 25 pixels is used to consider that the symbol belongs to the series. If the displacement is bigger than 25 pixels or it disappears during one frame a new series is created with this symbol information.

5. Results

For each color and text symbol a csv file is generated with the following time series information: id image, accuracy, x screen position, y screen position, shape and color.

A time series is drawn with each component position. Each color represents a different series due to the restrictions explained in chapter 4, sixth phase (see Fig. 14).

Reviewing the time series is easy to detect system malfunctions. A specific event detector could be implemented to check these kinds of malfunctions.

A time series drawing can show different symbols at the same time due to the non-unique identifier (see. Fig. 15).
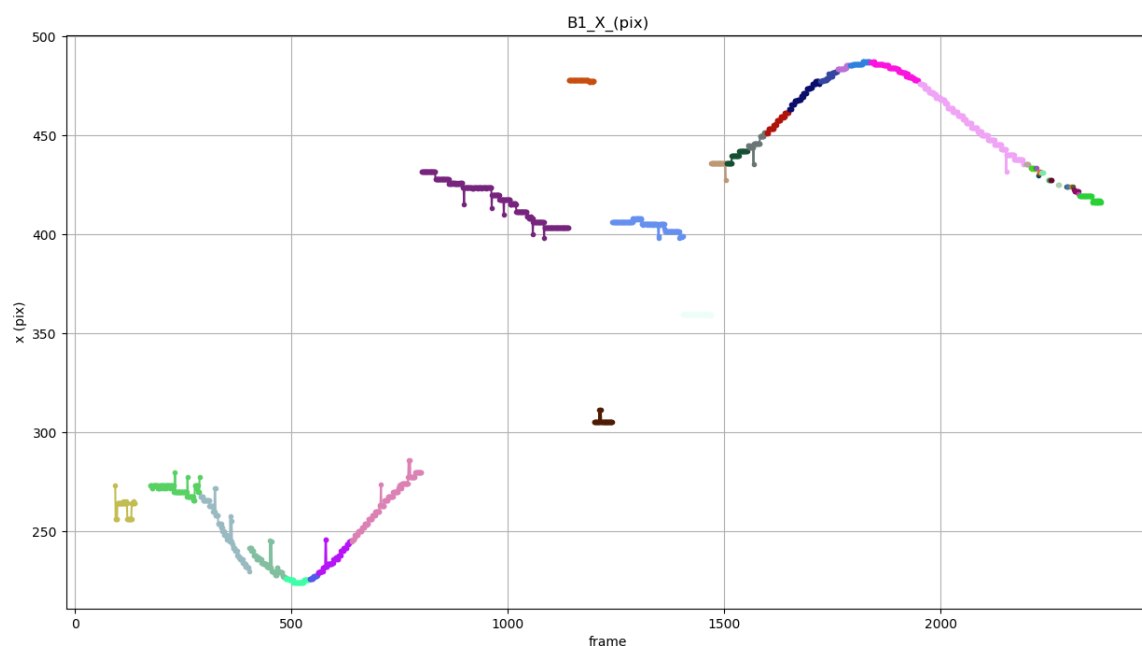
Fig. 14. Red B1 symbol x screen position.
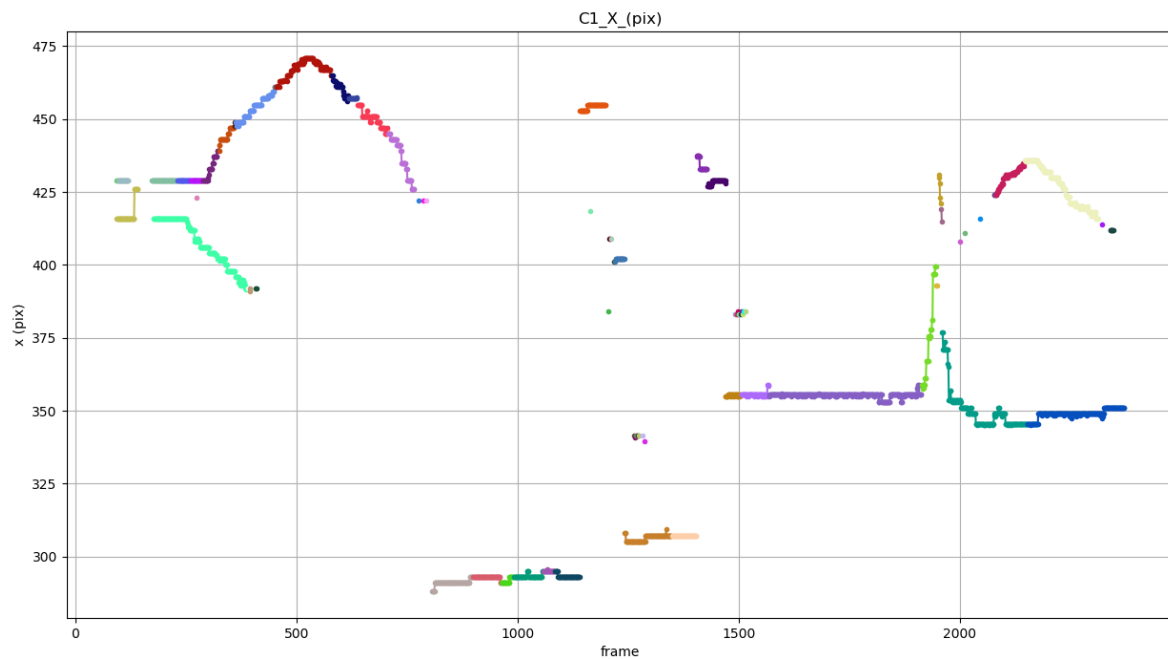
Fig. 15. Red C1 symbol x screen position.

A time series with the symbol accuracy is drawn. Each color represents a different series due to the restrictions explained in chapter 4, sixth phase (see Fig. 16). This data is very interesting to filter symbol data with low accuracy. False positive symbols identifications will disappear in this way, i.e. below 0.7 B1 symbols will be filtered.

A final figure with the overall symbol accuracy is drawn (see Fig. 17). This boxplot is a method for graphically drawing groups of numerical data through their quartiles.

The accuracy by symbol is represented. It is a very useful figure to improve the accuracy data of the overall recognition. In our video example the only true symbols are B1, A7, C1, A8 and C4, so if the data is filtered by an accuracy level of 0.6 all false positives and false negatives are rejected.
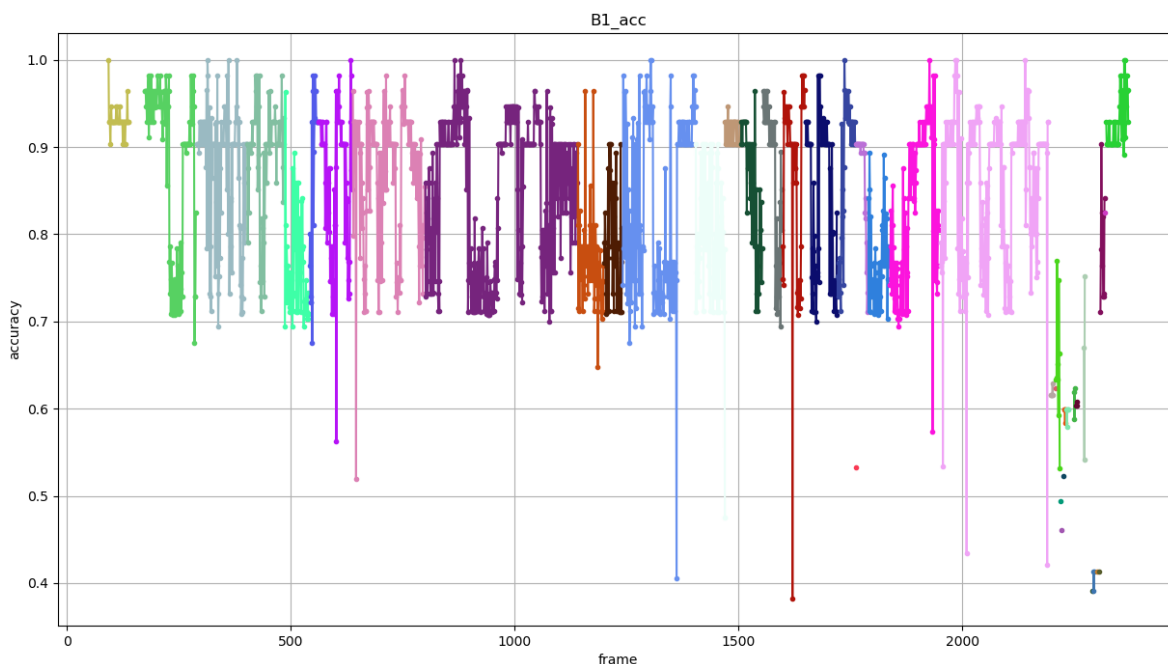
Fig. 16. Red B1 symbol accuracy.
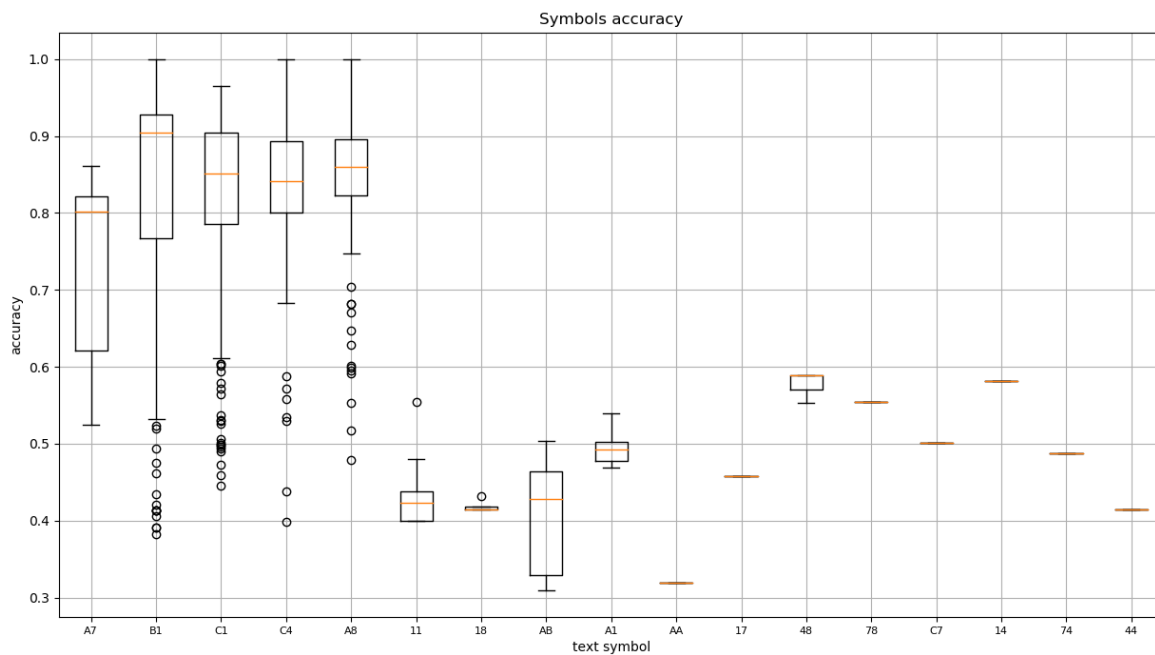
Fig. 17. Overall symbol accuracy.

6. Recommendations

In order to facilitate these kinds of displays validations several recommendations could be taking into account.

Visual data acquisition is a complex problem, therefore it is important to have several testing videos to increase robustness

It could be very useful to design displays trying to reduce all the challenges seen in chapter 3.

7. Conclusions

Aircraft manufacturers integrate a lot of equipment from different suppliers and on some occasion this equipment are like black boxes, the only easy way to check the data is through special monitoring displays supplied with the equipment. On other occasions a real aircraft display needs to be validated. These displays can be very complex, big amount of symbols and very dynamics. The validation of these displays can be a very tedious work if it is done by hand.

The methodology presented in this document demonstrates the capacity to extract the screen position, color, text and shape of the symbols presented in a real AC display with great accuracy. This pipeline is in some way specific to the display but it shares common phases to other AC displays.

This methodology provides very useful information in order to validate the system behind the display or to validate the display itself.

It can be concluded that displays validation based on image processing is a valid methodology.

References

[1] Tesseract

https://github.com/tesseract-ocr/tesseract

[2] OpenCV

https://en.wikipedia.org/wiki/OpenCV

[3] Python

https://en.wikipedia.org/wiki/Python_(programming_language)

[4] Thresholding

https://docs.opencv.org/master/d7/d4d/tutorial_py_thresholding.html

[5] Morphological Transformations

https://docs.opencv.org/trunk/d9/d61/tutorial_py_morphological_ops.html

[6] Image Moments

https://docs.opencv.org/3.4/d0/d49/tutorial_moments.html

[7] Convex Hull

https://docs.opencv.org/3.4/d7/d1d/tutorial_hull.html

Airborne FTI Camera Performance

Russ Moore[1], Bruno Marchevsky[2]

1: Curtiss-Wright, 15 Terry Drive, Newtown, PA, 18940, USA
2: Z3 Technology, 100 N. 8th St, Suite 250 Lincoln, NE, 68508-1369, USA

Abstract: Flight test instrumentation (FTI) have long sought to benefit from the bandwidth, scalability, and low cost of Ethernet networks and Internet Protocol (IP) based systems. Merging video and imaging cameras in the next generation FTI networks will require new methods and protocols to enhance data movement and recording and for time coherency of data. This paper will address the system and performance of the video encoding and decoding components, and various network performance issues, and the vision for deploying camera and FTI equipment in the same Ethernet network.

Keywords: IP cameras, H.264 AVC and H.265 HEVC, FTI displays, telemetry

1. Introduction

An internet-protocol camera, commonly referred to as an IP camera, is a digital video camera much like a high-definition (HD) camera or even a webcam. It transmits and receives data over a network or utilizing internet protocols to communicate video and audio information. Unlike an ordinary webcam it is a standalone unit with its own IP address that requires nothing more than a network connection in order to transfer images. The IP camera connects to a network in exactly the same way as any other standard network device such as a laptop or printer or even FTI Data Acquisition System (DAS) devices.

There is increasing demand for high-quality HD video for airborne applications such as flight test instrumentation (FTI). Ideally, such new camera solutions can reduce the weight and difficulty of installing wiring, and enable data to be coherently combined with image data. Ethernet cameras can address these needs with built-in compression and IP data format encoding with multiple output video streams. Additionally, as Ethernet-based networks have become an attractive choice for FTI applications, we see increased requirements for integrating Ethernet-based cameras alongside FTI data acquisition equipment, network recorders, and telemetry systems as this removes duplication of wiring and devices.

Using an Ethernet camera that supports onboard compression enables video compression to be removed from the DAS, or it can eliminate a dedicated unit. The camera can be connected via an Ethernet switch directly into the system, like any other data acquisition unit. Even better, because there is no need for dedicated hardware compression, SWaP is minimized and installation wiring greatly simplified.

The images captured by an IP camera may be viewed from anywhere in the network, whether via pc, laptop or video display, telemetered to remote displays or dedicated video analysis equipment and network recorders. In many cases, as well as being able to view video footage and listen to audio streaming, the camera may also be controlled remotely.

The key considerations when designing and operating the FTI devices on an aircraft or vehicle are image quality, bandwidth and latency. This paper investigates and evaluates these factors.

2. The IP video ecosystem components for FTI

The key considerations when designing and operating the FTI devices on an aircraft or vehicle are image quality, bandwidth and latency. This paper investigates and evaluates these factors.

The internet was not originally designed to deliver video and synchronized audio. HTTP live streaming protocols emerged as a way to deliver video/audio files over the internet by breaking the stream into a sequence of small http-based video/audio chunks. By using http transactions, live streaming can transverse firewalls and use standard web servers to deliver video streams and scale when many cameras are streaming to end-points.

The IP cameras used for FTI missions need to be designed for rugged environment usage with aircraft power converters able to withstand short power outages and aircraft power levels, ruggedized enclosures able to withstand high vibration and shock environments as well as mitigation for EMI and EMC effects.

The base ecosystem consists of devices that provide data as use-cases, such as

- Acquire (data, video)
- Record
- Display
- Telemetering
- Processing

These are all supported by the network "fabric" which provides data movement and communication activities, as shown in Figure 1.
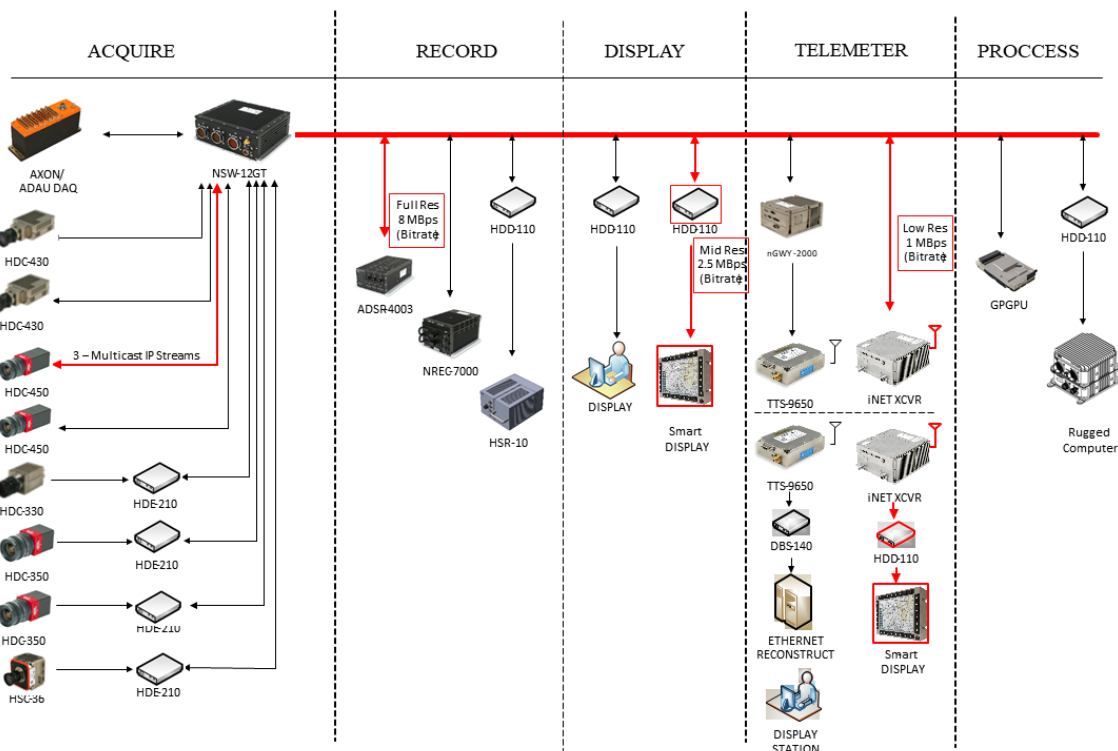
Figure 1: A network fabric supports all the use cases for the video data

3. Data acquisition in IP cameras for FTI Applications

Capturing and recording data from cameras in flight test applications is just as important as capturing video. Data is made available in multiple formats and transmitted in multiple interfaces, but the most common formats by far are carried over a standard serial port (RS232, RS485 or RS422) or as embedded data packets in the blanking time of SDI video. Single, slow events are many times represented by contact closures or their digital equivalent, a "GPIO" (general purpose input output).

Modern IP cameras tailored for flight test can acquire these data items embedded in the video stream, from a serial connection or from a GPIO. Because the data collection and the video capture of the camera are integrated and being performed in the video capture device, the IP camera system is able to closely correlate the time of arrival of these events with the time of each frame of the video, resulting in a more tightly integrated system with better performance and lower cost than a similar system built up from multiple boxes.

4. Synchronization of the Video and Data Streams

Compressing the video dramatically reduces its bandwidth requirements and is a necessary step for storing the video stream, or enabling its transmission over a communication method which provides less bandwidth than the full uncompressed video (e.g. an IP network). Algorithms such as JPEG, H.264 and H.265 among others "encode" the images or video according to the rules of that standard to reduce the number of bits required to represent the image.

Video compression is a complex operation which can sometimes introduce large amounts of jitter. In order for the receiver of the stream to be able to pace the frames correctly into the video compressor, the encoder includes a time stamp with each video frame which identifies the exact time when that frame is to be displayed. This same mechanism is used to time stamp the data acquired prior to bundling it with the video content for transmission. The decoder at the receiving side is then able to put together the sequence of events with their correct timing.

4.1. Synchronizing the Video, Audio and Data Streams with real Time

As mentioned, compressed video from the IP camera naturally carries timing information. It is possible, and often desirable, to correlate the time stamps from the video frames or metadata packets with a real-time clock. In order to do so, some IP cameras with IP-encoders, can use the time protocol IEEE-1588 PTP to synchronize the video acquisition with the time in the real world. Once the relationship is established the video and data time stamps can be related to the world clock.

Depending on the streaming protocol desired, the timestamps themselves can be made to carry real-time, as opposed to a time relative of when streaming started or power was turned on, or additional metadata. This can be added to the stream making the correlation between the camera hardware time and the world time reference.

The accuracy of the relationship depends on how the real-time is acquired and how good the time reference is. For

example, using PTP is inherently more precise than using the network time protocol (NTP). But neither would be accurate if the network server that provides the time reference to the IP camera itself has the wrong time or is set up incorrectly.

In practice time stamps can be correlated to precisions between 100's of nanoseconds and tens of milliseconds, which is perfectly adequate for the vast majority of applications.

4.2 Synchronization of streams between multiple IP cameras

It is possible (and fairly common) to synchronize and correlate in real-time the data output of multiple IP cameras, metadata streams and all.

The above mechanisms of providing a method to relate one camera's timestamps to a clock reference on the outside world also provides the way for multiple streams to relate to each other, provided that all IP cameras involved are synchronized to the same time standard reference. Once that reference is established, it becomes possible to collect time-correlated data from multiple IP cameras in multiple locations connected by just the IP network. Streams synchronized this way can be recorded for future use or displayed in real time, or both, all along retaining the timing relationship between the various video frames and data items.

5. Quality metrics and considerations for IP video and telemetry

5.1 Video image quality vs network bandwidth

Understanding the basics about network bandwidth availability and demands is critical to planning, designing and deploying a network-based video systems with IP cameras in FTI. To determine bandwidth availability, you first need to determine what locations you are communicating between.

The most important factor in determining how much bandwidth is available is whether or not you are sending video within a vehicle, or small space, or telemetering the video from an aircraft to ground locations. Within a small space such as a vehicle, the bandwidth availability could be large, say 70 Mbps to 700 Mbps. Sending video off-vehicle to ground stations or in a space application use, the bandwidth could be quite small, e.g. 250 kbps to 4 or 5Mbps.

5.2 Image quality vs compression loss

Video compression is necessary so that the amount of information created fits the allowed amount of capacity in the channel. Video information is highly compressible because it contains lots and lots of redundant, repetitive or not very important information and encoding can therefore eliminate the need to transmit or store such information. The encoding process is a collection of methods that seek to remove the "extra" information step by step, in an assembly line kind of way.

One example on how compression is achieved is the "color space" used to transmit the visual information. We all know that red, green and blue are the primary colors because they match the color receptor cells in our eyes. The light sensors in our retinas are much more sensitive to the overall light levels than to color, so the luminance information of an image is more important for our brain than the color information.

This causes the coloring book effect – a child can splash color in a coloring book, and as long as the black and white countour lines are visible our brains see a good picture. This effect is applied by video compression, where the algorithm is allowed to make the color information "fuzzier" in order to lower its bandwidth requirement.

To take advantage of this effect, the captured image which has red, green and blue components or RGB, are converted to a color space called YUV. The "Y" stands for luminance, or brightness, while U and V carry the color. Pretty much all image compression schemes use the coloring book effect and just discard a lot of the color information. H.264 and H.265 for example, discard a whopping 75% of the color information.

If we started with, let's say, four pixels, each with R, G and B values, H.264/H.265 will convert to the YUV space and keep four values of Y (that is the black and white, important part) and keep only one value of U and V. So instead of carrying twelve pieces of data (RGB x4) now it carries only six (YYYY UV). This alone is already a 50% reduction in the amount of information. The side effect is that for all intents and purposes these four pixels are painted with the same color at the receiver, when the YUV data is converted back to the RGB color space.

The interesting point to make from the above example is that information was lost. In the color space conversion the loss of information is perceptually minimal – the nature of our vision is such that we don't really see the negative effects of losing that data (color blurriness). The data loss, however unimportant, is real.

Other obvious ways to remove redundancy is by recognizing that video is by nature repetitive. Why transmit pixels – even in YUV space – if these same pixels were already transmitted before? So for example, if the video stream is that of a clock on a wall where nothing else moves, why keeping re-transmitting pixels of the wall around the clock? Instead, it should be sufficient to just send the pixels that changed when the arms of the clock moved.

Indeed, motion video compressors do just that. By storing the previous image on a local buffer, the encoder can use the stored frame as a reference to compare the new image with and only encode the parts that are different. Modern encoders go a lot further than just requiring images to be different. H.264/H.265 have methods to detect that objects moved within the frame, and instead of sending all the pixels of the affected area it can just send the information that "this object moved from here to there".

While all these schemes can achieve impressive compression ratios, there is only so much truly redundant information. At some point it becomes necessary to

remove visual information from the picture in order to achieve the amount of compression required to fit in the channel. In the clock example above, if we only transmit the pixels that have the clock hands we could lose the change in shading on the wall itself due to the change of illumination as the Sun goes up and down.

Perfect fidelity would require us to transmit all the pixels all the time. "Good enough" requires us to update these pixels only every so often as the motion is quite slow. The important thing is that with encoding the fading brightness of the wall in the evening will be in steps, as pixels are refreshed by the encoder. We can choose to update these pixels anywhere from very often to almost never and in the process create a trade-off between more bits to create a better image quality, or fewer bits to fit in a "smaller" channel or record longer durations.

5.3 Choosing the best compression parameters

Video compression is a complex process with many different "tools" looking for possible reduction in bits. Each of these steps/tools are minutely specified by the encoder standard, and many have parameters that can be adjusted at the encoder, just as above we can imagine a control parameter which would specify "how much different is different enough" to trigger transmitting that pixel. Tools exist that determine how big some area of pixels should be for subsequent operations, how far should the encoder look for objects that moved, how to look for camera movement (whole image shifting), how many and which frames to retain to use as reference for future comparison, how to pack the bits resulting of the encoding, etc. There are lots of different buttons and levers that affect the trade-off of quality vs. amount of bits.

In a compressed video system, such as the IP camera, the quality of the image is directly related to the amount of bits used to transmit it. More recent encoders are "better" because they are more efficient in the sense that they can create images that we perceive at the same quality level as other encoders, only using less bits to do so. This is accomplished by better, deeper analysis of the video given to the encoder, which of course tends to be computationally more intensive. As an example, H.265 coding is 25% to 50% more efficient than H.264, meaning that it requires 25% to 50% less bits than H.264 to achieve the same level of image quality. On the flip side the H.265 encoder uses about 10x the computing power required by H.264.

The grand takeaways are that it is better to use H.265 encoding technology if is available, as it is going to provide better image quality given the amount of bits that can be had. Also, experiment with encoding parameters. They are so complex and there are so many interactions between them that it is best to just play around and get familiar with them. The trade-offs can be significant and many times the trade-offs are not intuitive.

For example it is common that for smaller bit rates a smaller resolution stream looks better than one with higher resolution. For example at 250 Kbps a video stream at 320x240, 15fps will usually look better than a 720p, 60 fps stream (unless there is no motion).

When in doubt, ask the IP camera vendor for help. The IP camera designers will be familiar with the behavior of their camera (they are all different) in many different applications, and will be able to provide at least pointers for optimal configuration.

6. Video latency

Latency is defined as the amount of time between the instant an image frame is captured by a camera and the instant that frame is displayed. Low latency is a design goal for any system where there is real-time interaction with the video content is required; such as pilot, crew video interaction or through robotic and computer image processing applications.

There are several stages of processing required to make the pixels captured by a camera visible on a video display. The delays contributed by each of these processing steps—as well as the time required for transmitting the compressed video stream—together produce the total delay, which is sometimes called end-to-end latency.

The following aims to define and explain the basics of video latency, and discuss how one of the biggest impacts in reducing latency comes from choosing the right video encoding.

6.1 Acceptable latency

There are several stages of processing required to make the pixels captured by a camera visible on a video display. The delays contributed by each of these processing steps, as well as the time required for transmitting the compressed video stream, together produce the total delay.

But the biggest contributors to video latency are the processing stages that require temporal storage of data, i.e., short-term buffering in some form of memory.

There is no universal absolute measure of video latency. Instead, what is considered acceptable varies by use-case, or application, of the video. When the application is for human interaction, humans can react to stimulus input at roughly 5 Hz (human reaction times are noted in Table 1).

Table 1: Human / Operator Average Reaction Time

Stimulus	Ave. Time
Video / Visual	250 ms
Audio / Sound	170 ms
Touch / Pressure	150 ms

Video latency under 200-250 ms is generally acceptable while under 175 ms is considered low latency. In an application where a machine will interact with video stimulus, latency requirements are usually much lower; under 50 ms and many times under 30 ms.

6.2 Examining latency within systems

From camera to network to telemetry to video displays and recorders, most design goals look to achieve the

optimum balance of encoding speed, network transmission speed, telemetry bandwidth and video quality.

A downside to achieving low latency is sacrificing video image quality. Image sensor speed or image frames per second can affect latency since smaller frames-per-second add to latency. Testing also has shown that using the real-time protocol (RTP) and the real-time streaming protocol (RTSP) lower overhead time and helps lower latency performance. GOP size (group of pictures i.e. how many images are bundled together as intra and progressive coded pictures) is also important to latency; larger GOP size improves latency.

There are typically five main sources of latency in an Ethernet network IP camera video system:

- Image sensor/source
- Video encoder and compression
- Network and/or telemetry transport
- Video decoder
- Video display / renderer

The image source latency can come from image sensor data architecture and sensor data-clock and data serializer-deserializer, raw (Color Bayer Pattern) data processing and data to RBG conversion and, finally, color detect and segmentation. This is shown in Figure 2.
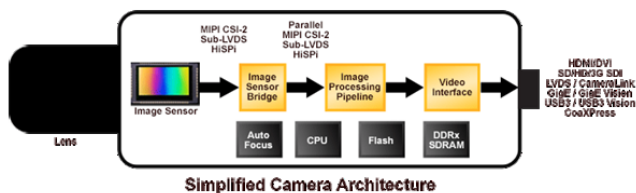
Figure 2: Simplified camera architecture

The video encoder and compression engine introduces latency as the encoder frame capture and process engine can scale, filter and translate image data. Many times a technique to lower latency will be to disable the upscaling process and to set image sensor resolution appropriately. **Fehler! Verweisquelle konnte nicht gefunden werden.** below provides latency times from an FTI IP camera using various compression techniques.

Table 2: Codec and latency table

Codec	Resolution	Latency
H265	1080p60	37-70 ms
H264	1080p60	35-60 ms
MJPEG	1080p60	37-50 ms
H265	720p60	30-50 ms
H264	720p60	30-50 ms
MJPEG	720p60	29-45 ms

Latency from network transport architecture illustrates how many times that latency is dependent on the network transport type. In testing it is shown that using RTP and RTSP protocols allows most network architectures to

achieve the lowest possible latency results. This is primarily due to the low overhead of the protocol.

Networks can be as simple as star configured Ethernet or more complicated with Ethernet, PCM conversion, RF telemetry radio communications, PCM-to-Ethernet reconstruction, etc. Network throughput issues many times cause buffer data to stack up on the encoder and even momentary degradations can cause large amounts of added latency to video.

Next, decoder latency issues can be the greatest source of video latency. Here, timing is the issue. The decoder needs to tie the encoder video time to the display timing. The buffer level scheme should ensure the display never runs out of buffer to ensure smooth video playback. Hardware, FPGA-based, decoders are better at synchronizing the encoder-display timing. **Fehler! Verweisquelle konnte nicht gefunden werden.** shows encoder and decoder effects on frames and decoder network cache.

Table 3: Encoder - decoder effects

Encoder FPS	Decoder Network Cache	Frames in Decoder Buffer (For Smooth Playback)
60	80 ms	5
30	180 ms	5+
15	400 ms	6
7	900 ms	6+
5	1200 ms	6+

Finally, the latency at the video display or renderer needs to be considered. The display latency can vary widely with display design architectures and display settings and especially higher resolution settings can add latency. PC and gaming settings can provide lowest latencies. Measured display latency from 33 ms to 120 ms are typical. See figure 5 below for measured and theoretical latency.
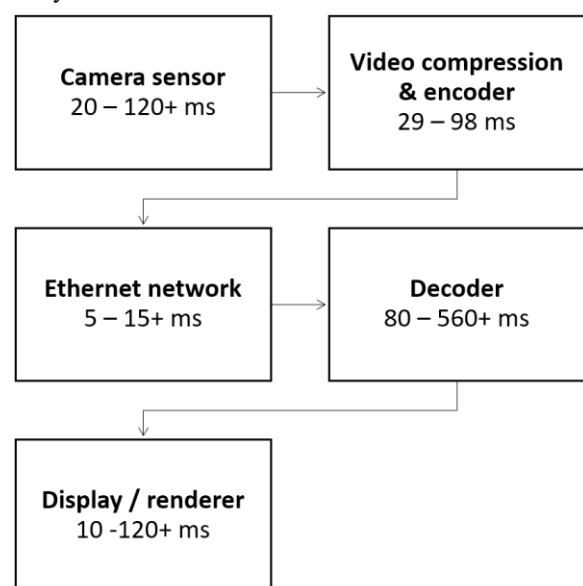
Figure 3: Measured and theoretical video latency

7. Conclusions

In the FTI implementation of IP camera networks, bandwidth dedicated to video is critical. This includes the information communicated from acquisition to recording, visual display and further processing.

The CODEC used greatly influences bandwidth and video quality. As stated earlier, the grand takeaways are that it is better to use H.265 encoding technology if is available, as it is going to provide better image quality given the amount of bitrate. At the video display, decoding functions tightly tied to display operations will also enhance latency timing.

Constructing the FTI network "fabric", i.e. managed network switches, is key. Elements such as IEEE-1588 PTP time grandmasters, and video and data acquisition equipment able to time synchronize and time-stamp the acquisition of data, and simultaneously send it to consumptions point, such as recorders and displays, will provide time coherency for data reduction activities.

Finally, while there are several elements that all contribute to latency in a video system, including image source, encoder, the network fabric and the display, the most impact is often attributed to the decoder. This latency can best be reduced by using hardware such as an FPGA decoder that is tightly linked to the display.

7. Glossary

DAS: Data acquisition system
EMC: Electromagnetic compatibility
EMI: Electro magnetic interference
FPGA: Field programmable gate array
FTI: Flight test instrumentation
GOP: group of pictures
GPIO: general purpose input output
HD: High definition
IP: Internet Protocol
NTP: network time protocol
PCM: Pulse code modulation
PTP: Precision time protocol
RF: Radio frequency
RGB: Red green blue
RTP: real-time protocol
RTSP: real-time streaming protocol
SDI: Serial digital interface
YUV: Luminance-Bandwidth-Chrominance

Beholder
Automated Data Validation in Flight Test

Miguel Arevalo Nogales
Airbus Defence and Space
miguel.arevalo@airbus.com

Abstract:

Current aircraft development mandate for very complex data acquisition systems, the amount of parameters being stored has grown over the years from a few thousands to almost half a million. Data validation and sanitization has become a priority, especially given current (unattended) data analysis techniques. These two problems mandate for an **automated verification and validation** system, capable of detecting anomalies in real time [1].

Airbus Defence and Space has developed a custom solution to this problem, it is known as **Beholder**. This software leverages user knowledge in the form of rules, making them into a validation piece of code which is automatically loaded and executed over a data flow.

Airbus **Daemons** builds up on current Data Server technology to support an automated process execution system. The combination of both, Daemons and Beholder, provide Flight Test with an automated, unmanned data validation procedure, improving data quality and enhancing response times to erroneous sensors or recordings.

Key words: Automation, validation, rule, sensor, acquisition, testing.

Introduction

Aircraft's data analysis can take place in several phases, from system development to in service maintenance. We can, broadly speaking, group these phases into:

- Development
- Certification
- Delivery
- In service

These phases require different types of data acquisition systems, ranging from very intrusive at the very beginning to almost non-existing in the "in service" phase.

Development phase is, probably, the most intense phase regarding data acquisition. Design offices have to test their theoretical models in real world conditions, system integration has to be tested and design problems may arise. The consequence is data acquisition is really intense, not only internal aircraft data is recorded, but custom instrumentation is installed and prone to changes, having to deal with changing needs. All this facts make data validation especially important during this phase.

Certification can also require a complex data acquisition system to provide proper evidence to authorities, guaranteeing not only the aircraft meets specifications, but also and most importantly flight safety.

Both development and certification have to deal with non-aircraft native sensors, added on purpose to provide further data to analysts. These sensors must be treated with special care since, by their nature (exposition, installation limits, etc.), they are susceptible to failures.

Delivery and in service phases normally rely on simpler acquisition systems, usually no instrumentation is present, and analysts rely on bus recorded data.

All this data is later on used for data analysis, be it a design change needed to fulfil specifications, some kind of evidence needed to certify the aircraft or fleet analysis for predictive maintenance. Of course if data quality is no good all these analysis will be impacted.

Data acquisition plays a major role in aircraft development but, how can we guarantee data quality? Manual validation is costly and error prone, especially in modern developments due to the sheer amount of acquired data; **some way of automatic data validation is needed.**

Solution Overview

Beholder is Airbus Defence and Space solution for rule analysis and anomaly detection.

In an ideal world every system's expert would check for data quality not only after the test but before and during the test itself, meaning that if the data is not up to the quality levels needed, the test could be stopped, resulting in improved safety conditions and reducing expenses in unproductive tests. Due to current acquisition systems complexity and aircraft testing pace, where several flights are performed in a single day, this is utopic.

Beholder answers this problem by leveraging expert knowledge, in the form of validation rules, and produces a simple to read log where every rule occurrence during the flight is recorded.

This methodology guarantees experts are not encumbered by their computing skills and can focus on defining a good-enough rule set that grants recorded data is up to required quality levels for later analysis.

Beholder makes test validation an autonomous task, meaning the expert does not need to visualize or manually analyse the test recording to validate data, but this process has to be executed manually whenever a new test is recorded.

Data recording in Flight Test normally happens on-board during the flight, meaning the whole data set is not available until the aircraft lands and the hard drives are processed. This task is time consuming and can take place during night time; the experts may not be there to validate their data when the data is available.

Airbus Defence and Space has developed a Daemon Automation System intended to execute tasks whenever a new test shows in the servers.

This system provides several advantages to manual systems:

- Tasks can be executed without human interaction, they can be executed during night hours and their results are normally available next day in the morning.

- No human interaction diminishes failure risk and improves timing by not requiring long working hours or turns.

- Validation sharing, which is sometimes overseen if user does not see immediate profit from it, is improved as it is a natural product of standardised rule sets and validation procedures.

Beholder

The rationale behind Beholder is that an expert can write a rule set that defines the data quality level for the operation of the system, detecting anomalies, strange working conditions, and situations where limits or thresholds surpassed.

Manually detecting these conditions is possible, but it is not a desirable approach since it is tedious, error prone and time consuming.

Beholder is based around the following concepts:

- A rule is a logic condition which can be evaluated by a common compiler into a Boolean value.

- Rules are external to Beholder; no rule logic is stored inside the application.

- Each rule can be triggered in every instant.

- All rules are evaluated at the same pace; this eases rule & parameter synchronization. This is what commonly known as CVT: Current Value Table.

- Rules can be grouped in sets; a set defines a system's behaviour.

- Beholder's output is a time line describing when rule's conditions are met: the rule is true.

As a software application Beholder's main characteristics can be summed up as follows:

- Multiplatform: developed in Java, Open-JDK 1.8 [2] [3] compliant code.

- Based on Flight Test Multiplatform Analysis Software (FT+) and Dataserver technology.

- UI and BATCH processing modes. Normally the data validation process takes place in an automatic, unmanned way, but a UI offers the possibility to run Beholder on purpose over a set of tests.

- Dataserver is used as the data providing technology, meaning IRF, PFF, or CDF files are readily accessible.

- Since it is Data Server based Client – Server technology offers several benefits including: scalability, heterogeneous systems, etc.

- Automatic code generation. Beholder does not store rules inside its code, they are defined externally to the application. In order to get the best performance user rules are automatically transformed into Java code, which is compiled and linked dynamically in execution time.

- All Java [2] syntax functionalities are allowed in rule definition: simple mathematical functions, more complex functions as defined by Math class, etc.

- Internal Beholder functions. In addition to Java functions Beholder provides a set of custom, internal functions to be used in rules:

- Timers: a timer is a special function intended to measure the amount of time a rule has been in certain state.

- Buffering functions: some functions need a time slice to be usable, relying on past data to provide a meaningful value. Examples of this kind of functions are maximum, minimum o mean values.

- Macros. There are certain macro values that are usable within Beholder rules, for example: current time, analysis starting or ending times, etc.

- External mathematical functions: the idea of these functions is extend Beholder functionalities to Machine Learning systems, including classification algorithms, etc.

From a design [4] [9] [7] [8] point of view Beholder is divided in several modules:

1. Rule analysis and transcription

2. Beholder core

3. Loggers

4. Interface

Prior to rule execution Beholder needs to transcript natural language syntax into a computer intelligible language, in this case Java code.
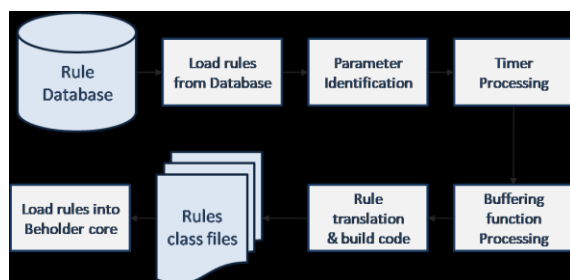
Schema 1: Beholder rule preparation workflow.

The process follows the following steps:

1. Rules are loaded from a user selected database

2. Parameters are identified and replaced with a placeholder. This placeholder is a link (pointer) to a memory location that is going to hold requested data, this data changes every sample.

3. Timers are identified and extracted, for every timer a new condition and rule is created. Original timers are replaced, like parameters before them, by a pointer to a memory location; this location will hold the timer value after the rule has been evaluated.

4. Buffering functions are identified and replaced by pointers, like parameters and timers before them.

5. Rules are translated into plain Java language, every rule is written as a different class, compiled, linked and loaded dynamically [5] [6] into Beholder core.

Beholder core is the processing nucleus of the program, all other parts attach to it, preparing rules, providing data or consuming rule events.
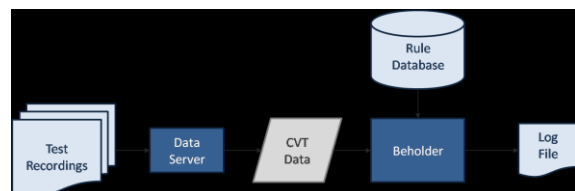
Schema 2: Beholder core diagram. Once rules are loaded into the core they are evaluated using Dataserver provided CVT data.

Beholder is based in a processing pipeline; this pipeline is repeated for every sample read from Dataserver (CVT data in Schema 2).

1. Data is read from the Dataserver, one DBT at a time. This DBT contains all parameter information for a current time.

2. Data is copied into a static data array. This is needed to ensure all rules get to access their parameter information properly.

3. Elaborated functions are executed using this static data array.

4. Timers rules are evaluated and the static times array updated with their values

5. Rules are evaluated, in this step all variable values (buffered function values, timers, etc. are known).

6. An event is thrown for every rule state change. These events contain relevant information such as time, change status (true to false or false to true), etc.
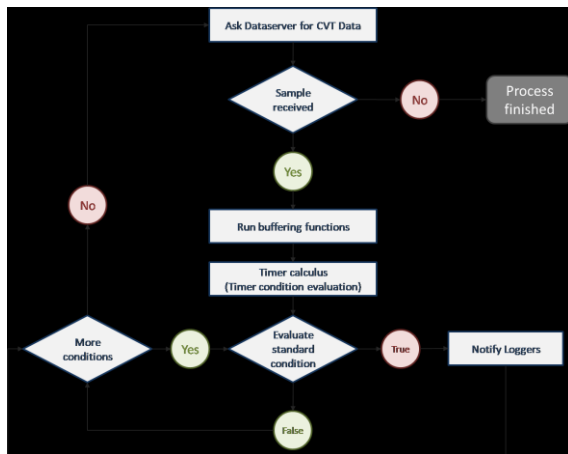
Schema 3: Internal Beholder core workflow.

Events are processed asynchronously by different classes, external to the calculus core; some of these classes provide a graphical representation of the events whereas others write rule events in different formats (log files, databases, etc.).

Some examples of loggers include, but are not limited to:

- Database logger, where each event is inserted into a results table.

- Simple text logger, output is a plain text file.

- Summary logger, where rules are grouped by type.

Beholder user interface is a detachable module intended to offer flexibility in test selection and process execution. From a programmatically point of view the design follows a MVC paradigm, separating control agents from graphical representations and model structures.
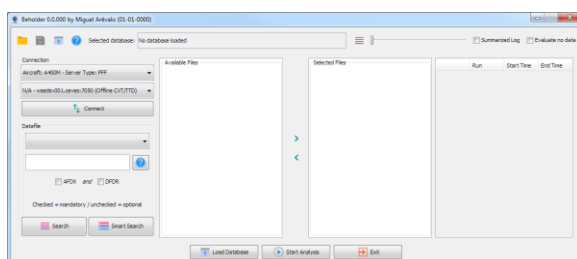
Image 1: Main graphical interface.

Interface is divided in three rows:

- Top row holds an icon bar for opening and closing sessions, selection the amount of threads used for analysis or customizing outputs (logger types).

- Middle row is dedicated to test selection.

- Bottom row has three buttons:
    o Load database
    o Start analysis
    o Exit application

As can be seen middle row holds most of the options, the idea is for the user to proceed from left to right, top to bottom selecting options:

- Dataserver

- Aircraft model and serial number

- Data file filter

- Acquisition system type (DFDR and or AFDX)

Once this information is filled in user can look up for files matching this criteria. Files are presented in the available file list, and can be transferred into the selected file list.

For each file selected user is offered the option to specify a custom time slice or set of time slices. User is offered a default time slice covering the whole test.

Once all this information is selected the analysis is ready to proceed, user must only provide a database containing a rule set.

When in UI mode there are several loggers active intended to provide enhanced visual information to user, whereas in batch mode graphical loggers are reduced to save up resources and status information is shown in a command window.

In order to properly display rules and events a dedicated graphical logger is shown, the Detected Events window (see image 3).

Image 2: Detected events window.

This logger displays a table with a single row for every condition met, this table shows rule ID, aircraft information, starting and ending times and some rule information such as rule group, description or severity level.

While Beholder is running Analysis status windows displays information about the whole process, seen as a data processing task. Conditions are show when they start or end, but the main task of the window is to provide information about the data transfer status.

Image 3: Analysis status dialog

Relevant data retrieval information such as aircraft, data transfer status and connection settings are shown in this logger.

Real Time Validation

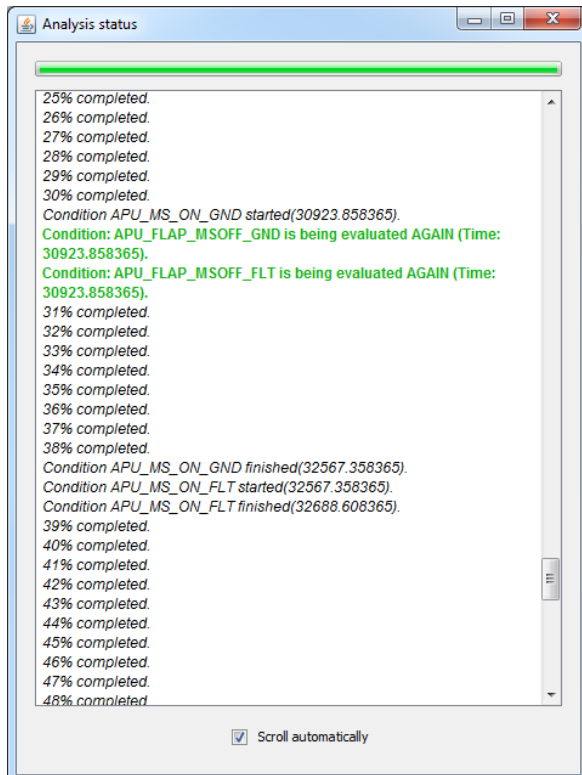Data corruption can sometimes happen during testing, due to the large amount of systems and parameter sometimes it is not feasible to check all these data during testing phase, moreover, even in the event of being able to do so, doing it manually is error prone and stressing.

Beholder can act as a real time validation system.

Under this working scenario Beholder works exactly in the same way as described before, but there are differences in parameter inputs and capabilities.

- Beholder limits itself in the amount of rules being analysable. This is a consequence of having to run in real time, if rules cannot be analysed in a timely manner it is better to split the ruleset into smaller sets and execute multiple Beholder instances.

- Data is not provided by an offline Dataserver but an IENA packet one. This type of Dataserver is a real time based server offering the same access protocol to real time telemetry data. The fact that data access is homogeneous between real time and files greatly simplifies the application.

- Due to real time Dataserver way of working there is no need to select an aircraft or data file, the Dataserver already knows which aircraft, MSN and test number it is receiving data from.

- A simplified UI is used, only detected events are offered and a telemetry status monitor is provided. This interface is controlled by the RTMS monitoring system, ensuring it is auto relaunched if it fails.

Real time autonomous validation is a great addition to any telemetry system; typically this was done with ad-hoc applications meaning small changes in sensors or parameters impacted monitoring software.

Beholder splits parameters from its validation system and the software used for it, improving validation opportunities.

A relevant real time monitoring use scenario, that showcases the advantages of this separation, is adding new rules in real time while the test is being performed. In this use case user can define a new rule, add it to Beholder's database and restart the application. Beholder will analyse the rule and compile it into working code, add it to the rule pool and continue the monitoring process.

In order to improve performance real time mode introduces a key characteristic over standard Beholder, compiled rules are kept from one execution to another, meaning that if there are no rule changes between executions Beholder does not need to recompile the whole ruleset. This optimization is only used in real time because it may lead to lazy condition checking problems; on the other hand it improves application start time, which is necessary if the application is to be managed by RTMS (Real Time Monitoring System).

Daemon System

Daemon system is an external application used to automatically run analysis tasks whenever a new test is detected in Flight Test data servers.

Daemon concept is similar to the one used in computer science, where a program, the daemon, is always running in the background. This program is normally in sleeping state, not consuming resources, but it awakes periodically and runs a predefined task.
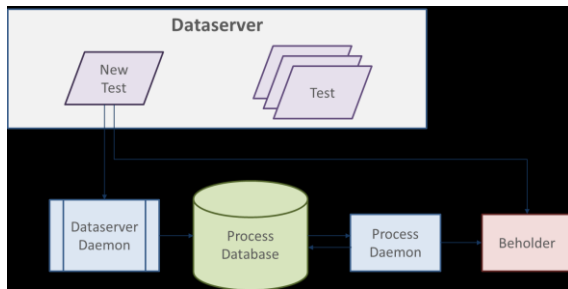
Schema 4: Daemon system

The rationale behind Flight Test Daemon System is that whenever a new flight is uploaded into the server a daemon, the Dataserver Daemon, will find it and upload its meta-information into a database, the Process Database. This information describes a test univocally:

- Aircraft model

- Manufacturing serial number

- Test identification: test type plus number.

Other daemons, known as Process Daemons, consume this information; they add their own information to the database and launch a user defined analysis process.

This process can be configured when creating the daemon, in the case we are analysing it is a Data Acquisition Validation Process which is uses Beholder.

The information added to the database is a register, with the same meta-information added by the Dataserver Daemon, adding some additional data:

- Process Id.

- Processing machine.

- Processing date.

- Process status.

When a new process is launched it updates its status to ONGOING, if it fails it will automatically change to ERROR and when it finishes it is updated to FINISHED.

There are control mechanisms in place to ensure daemons are alive, checking daemon status and working times. These guarantees there are no zombie processes and all tasks are executed appropriately.

When a task is updated to an error status other daemons running in different machines to the one that executed this process previously can re-execute the process, this leads to execution redundancy.

One important fact about daemons is that multiple daemons can be configured to take care of the same task type. We may have multiple Validation Daemons so when several flights are uploaded into the Dataserver at the same time they can be processed concurrently by different machines. This kind of redundancy ensures that if a machine is down for any reason others can take its place and keep with the data analysis tasks.

Airbus Daemons have the following characteristics:

- Multiplatform, developed in Java using OpenJDK 1.8 [3] compliance.

- Centralized database based [10]. This ensures multiple daemons, multiple machine synchronization.

- UI is web based.

- Daemons can be tied to a user group, meaning users can be notified whenever a daemon is down or a new test has been analysed.

Daemons interface has been developed using HTML technology. This web interface communicates with daemon database providing information and control over existing daemons, showing their status (sleeping, working or down).

Image 4: Daemon view

Interface offers not only daemon status but also process information. From the interface any user can order a test reprocessing, check its status or see when it was processed and what machine performed the analysis.

Image 5: Process view

Parameter Validation

Previous chapters showcased different aspects of Beholder, but they did not delve deep into how it can be used as a parameter validation tool and what are the benefits associated to its usage.

First thing to do is define what we should understand by parameter validation. Sometimes, especially when dealing with complex system, we incur in a misconception, mixing system validation with parameter validation, it is important to make a difference between these two concepts.

- Parameter validation only focuses in parameter value, checking if it is within limits and its value is reasonable considering contour conditions.

- System validation checks if the parameter is right considering system's current state.

As can be seen sometimes the difference can be very small, since a parameter out of bounds may not mean our sensor is misbehaving but the system providing erroneous data.

There are several approaches regarding parameter validation:

- Threshold rules. In this case the user defines an upper and lower limit for the parameter. If the parameter exceeds these limits then a warning is issued.

  A especial case are defined state rules, in this case a parameter can only take a small set of predefined values, for example Open-Close o True-False.

- Slope rules. These rules are based not in parameter's value, but how the parameter changes from one instant to another.

  Beholder offers several possibilities in this situation; since we are computing a derivative we can do so taking into account a predefined number of points, this buffer can be selected by user when creating the rule. We can also select the maximum or minimum values for a given time buffer, or even the mean.

- Cross system validation. It is common in aeronautical environments to have different systems provide different parameters for the same measurement. A very useful rule for parameter validation is to compare the difference between parameters or the delay between signals corresponding to the same parameter.

- Data alive check. Some straightforward validation consists on checking data moves and reacts to stimulus.

Once rules have been defined the instrumentation validation phase starts. During this phase systems are started, whenever possible, and data is recorded into small test files, these files are processed and added to Flight Test Dataservers.

As seen in previous chapters Daemons detect these tests and run specific rule detection tasks on them, looking for sensor misbehaviours, parameter exceedances, etc.

The result of this analysis is automatically uploaded to web servers and communicated to system owners and flight test instrumentation personnel, when something strange is found these groups can take proper action to fix the issue or investigate it even further.

Image 6: Rule database example

Conclusion

Current aircrafts mandate for a large amount of FTI sensors, sometimes more than on thousand. These sensors are added to an already large parameter dataset provided by the aircraft.

Validation of all these parameters is a huge task for which an automated system is necessary, manual validation does not guarantee good enough quality standards.

Making an automated system, open for end users usage, like design offices, not only improves Flight Test data validation, but helps users to get involved in their system's acquisition task.

Since data is automatically processed large amounts of test can be used for validation, meaning a new realm of validation techniques opens to end user: Big Data and machine learning. Using machine learning algorithms for data validation can help to identify upcoming problems, meaning FTI can make predictive maintenance to the acquisition system.

An additional benefit of automated data validation is react time improvement. Current technologies such as 3D printing or Data Science applied to system analysis allow for fast paced development, sometimes referred to as fast prototyping. This methodology mandates for improvements in FTI installation and validation, due to the prototype nature of the modifications being installed in the aircraft.

References

[1] Avner Engel. *Verification, Validation, and Testing of Engineered Systems (Wiley Series in Systems Engineering and Management Book 73)*. Wiley (2010)

[2] Oracle. "Java SE". *Oracle Technology Network*. (18 December 2014).

[3] Alex Kasko. *OpenJDK Cookbook.* Packt Publishing (2015).

[4] Robert Lafore. *Data Structures and Algorithms in Java*. SAMS (2003)

[5] Cay S. Horstmann. *Core Java, Volume II - Advanced Features*. Prentice Hall (2017)

[6] Glen McCluskey. Using Java Reflection. Oracle. https://www.oracle.com/technical-resources/articles/java/javareflection.html (January 1998)

[7] Trygve Reenskaug. *Working with objects. The OOram Software Engineering Method*. Manning/Prentice Hall. (1996)

[8] Gamma, Erich et al. *Design Patterns*. Addison-Wesley (1995)

[9] Martin Kleppmann. *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*. O'Really Media. (2017)

[10] Michael J. Hernandez. *Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design*. Addison-Wesley (2007)

BMAD. Automatic Detection of Flight Manoeuvres using wavelets

Francisca Coll Herrero
Airbus Defence and Space
Francisca.Coll@airbus.com

Abstract:

BMAD is a big data oriented project which aims to automatize most of the flight test analysis process.
The idea is to classify and identify automatically all the manoeuvres of ADS (Airbus Defence and Space) aircrafts fleet.
Up to now manoeuvres are selected interactively using plotting tools and in some cases these time slices are not stored in a data base.
The project consists in the automatic identification of maneuvers, the storage of their representative data in a database, and the exploitation of the data in a web service.
This paper will describe the use of wavelets for the detection of typical flight quality manoeuvres applied to real data corresponding to a combat aircraft.

Key words: BMAD: Bigdata Manoeuvre Automatic Detection, SQL: Structured Query Language, NoSQL: Not only SQL, Ftnet: Flight Test Net, CSV: Comma separate values, HDF5 : Hierarchical Data Format, JSON : JavaScript Object Notation.

Introduction

BMAD starts from the need of automatizing Flight Test data analysis chain.

This process covers: data acquisition and processing, time slice detection and manoeuvre classification, manoeuver's calculus and data storage, flight test report

generation and data exploitation: web services & dashboards.

The first step is to classify and identify automatically all the manoeuvres of ADS (Airbus Defence and Space) aircrafts fleet.

Each manoeuvre corresponds to a time slice and it is identified by an initial and final time. There are also some calculations associated to each manoeuvres like maximum values, minimum values, mean, standard deviation, even complex calculus like parameter identification .In most cases, the manoeuvres are selected by the analyst using interactive plotting tools.

Some manoeuvres like take off, landing or stabilizations can be easily identified using simple rules, for these cases we have an in-house developed tool called Event Detector. Event Detector is a program written in Java in which you can define with a rule the initial time and the final time of a manoeuvre.

But for other kind of manoeuvres it is necessary to use other techniques of identification using complex patterns.

Different technologies have been tested like neural network and wavelets the latter obtaining better results.

Once the manoeuvres are identified, in-house software, called JProcessor, is used for automatically performing desired data reduction.

The project consists in the automatic identification of manoeuvres, storing the times slices, calculations, and even the complete manoeuvre in a NoSQL (not only SQL) data base and the exploitation the data in a web service.

This paper will focus on the use of wavelets for the detection of typical Flight Test manoeuvres following these steps:

a) Define the pattern signal corresponding to the manoeuvre.
b) Obtain the best mother wavelet adapted to this pattern.
c) Find the wavelet in the time series by translation and dilations of the mother wavelet using the Scalogram.

BMAD workflow

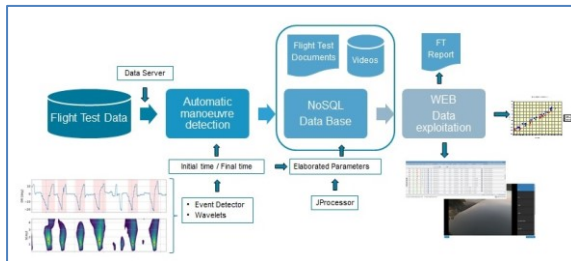In the Figure 1 it is shown the diagram of the BMAD project:

Figure 1.BMAD workflow

a) Access to the flight test data using data server.
b) Automatic manoeuvre determination using Event Detector and/or Wavelets.
c) Automatic elaborated parameter calculations (JProcessor).
d) Store desired information in a NoSQL data base.
e) The server includes in addition to NoSQL data base, all flight information: documents, videos and relational data base.
f) Finally, via web service (FTNet), you can access to all flights information, status of the program, plotting, streaming video service and even an automatic generation of Flight Test Reports.

Data Reduction, Parallel Processing

BMAD adopts many Big Data concepts during the data reduction phase:

a) Analysis focuses on multiples manoeuvres, different flights and several prototypes.
b) Paralleling processing is used for improved performance, relying on MapReduce concept.
c) Increasing the number of machines improves the speed of the analysis process. Tasks are assigned using a task scheduler (daemon) which controls the work flow of the phase.
d) Queuing is used to minimize idle times and improve data reduction efficiency.

Data reduction and calculus are performed by JProcessor. This application is an in-house developed tool coded in Java. This software is plug-in based, meaning new calculus functions can be added without modifying its structure.

JProcessor reads data, builds a data processing tree and calls elaboration functions. Finally, when all data is computed, it can write the results to several formats: databases (SQL & NoSQL), data files (CSV, CDF, HDF5), JSON etc.

Data Exploitation

Once the process of automatic manoeuvre detection, involved calculus and the storage in the data base is completed, BMAD process includes data exploitation via web –based services.

The web service will contain:

a) General Flight information: date, flight objective, aircraft configuration….
b) Manoeuvres information: initial time, final time, calculus results …
c) Status of the program : test point key status …
d) Plotting cross-plots, comparing different flights….
e) Streaming video service.
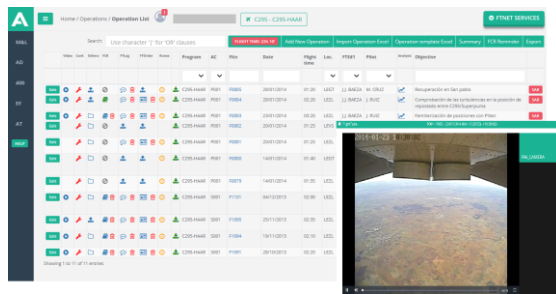f) Automatic generation of Flight Test Reports.

Figure 2. Web service

Figure 4. Mother wavelet ψ

Scaling ψ by a positive quantity s, and translating it by u ∈ R, we define a family of functions obtained from "mother" wavelet :$\psi_{u,s}$,

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right), \quad u \in \mathbb{R}, s > 0 \qquad (1)$$

Given f, the continuous wavelet transform (CWT) of f at time u and scale s is defined as:

$$Wf(u,s) = \int_{-\infty}^{+\infty} f(t)\psi_{u,s}^*(t)dt \qquad (2)$$

Analysis Methodology, Why Wavelets

The objective is the automatic pattern detection in time series.

Typical algorithms used are:

a) Short time Fourier Transform, this analysis has some limitations because it only works in frequency domain.
b) Neural networks, they need a lot of time for the training and results are not good enough.
c) Continuous wavelet transforms. Is similar to the Fourier analysis but it works not only in frequency domain but also in time domain. **Fehler! Verweisquelle konnte nicht gefunden werden.**.

Wavelet analysis expands functions in terms of translations and dilation of a wavelet function called mother wavelet.

A wavelet is a function $\psi \in L^2$ (R) with zero average (i.e.$\int_R \psi = 0$), normalized (i.e. $\|\psi\| = 1$), and centred in t = 0.

Fourier versus wavelets
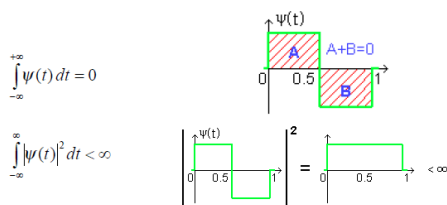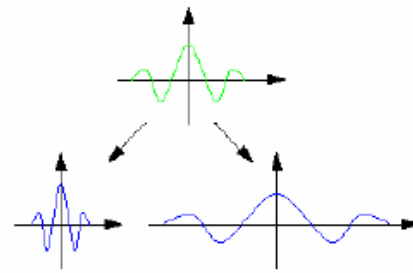
Fourier analysis has some limitations because only works in frequency domain. Sometime later, the notion of scale analysis by creating structures that vary in scale was developed

The most interesting difference between these two kinds of transformations is that individual wavelet functions are localized in space. Fourier sine and cosine functions are not. [5]

A wavelet is a mathematical function useful in digital signal processing and image compression. [2],[5]
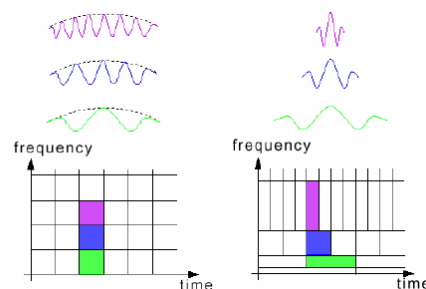

Figure 3. Haar wavelet

Figure 5. Time-Frequency plane :Fourier transform

Construction of Wavelets Approaching a Pattern

The technique described in this paper is based on the correlation of each manoeuvre with a pattern.
The first step is to construct wavelets adaptable, as well as possible, in the least squares sense, to the specific pattern.
But it is no using the typical wavelet library like Haar , b-spline,daubechies ,coiflet or Mexican hat , it is adapting each pattern to a sum of linear combinations of orthogonal polynomial [6].

Take the family of N functions F $\in L^2$ (a,b)

$$F = \{\rho_i\}_{i=1}^N \qquad (3)$$

We want to construct the approximation of the pattern=f in the least square sense using a linear combination of the F:

$$\psi = \sum_{i=1}^N \alpha_i \rho_i \qquad (4)$$

with the condition that:

$$\sum_{i=1}^N \alpha_i R_i = 0 \qquad (5)$$

Where:

$$R_i = \int_a^b \rho_i \ (t)dt \qquad (6)$$

The vector $\{\alpha_i\}_{i=1}^N$ and the function ψ are obtained by solving the following linear system where:

$$\begin{pmatrix} G & M^t \\ M & 0 \end{pmatrix}\begin{pmatrix} \alpha \\ \lambda \end{pmatrix} = \begin{pmatrix} B \\ 0 \end{pmatrix} \qquad (7)$$

Where :

$$G_{ij} = \int_a^b \rho_i \ (t)\rho_j(t)dt \qquad (8)$$

$$M_{i=} R_i \qquad (9)$$

$$B_i = \int_a^b f \ (t)\rho_i(t)dt \qquad (10)$$

In numerical calculations, we have a finite sets of values of the function in an interval [a,b]:

$$\{(t_k, y_k)\}_{k=1\ldots N} \text{ where } a \le t_k \le b$$

and $f(t_k) = y_k$

The problem consists in to find

$$\alpha = \{\alpha_i\}_{i=1}^N \text{ and } \psi = \sum_{i=1}^N \alpha_i \rho_i$$

Such that minimize in least square sense the expression:

$$\sum_1^N [\psi(t_k)\text{-}y_k]^2 \qquad (11)$$

With the constraint:

$$\int_a^b \psi(t)dt = 0 \qquad (12)$$

The vector α is obtained by solving the equation (7).

Manoeuvre Detection using Wavelets

The steps used to identify the manoeuvres using wavelets are the followings:
a) Select a pattern of the specific manoeuvre.
b) Obtain the corresponding mother wavelet adapted to this pattern.
c) Find the pattern in the time series by translation and dilations of the mother wavelet function using Scalogram. Ref.3.

The following Figures 4, 5, 6 represent manoeuvre identifications steps applied in a theoretical signal composed by sines:
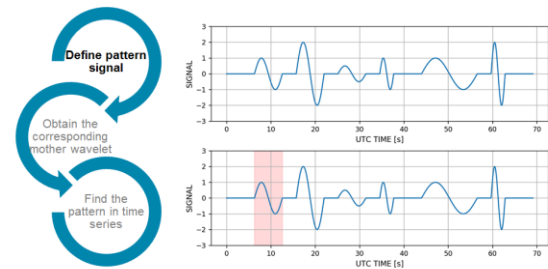

Figure 6.Define pattern signal: SETP1

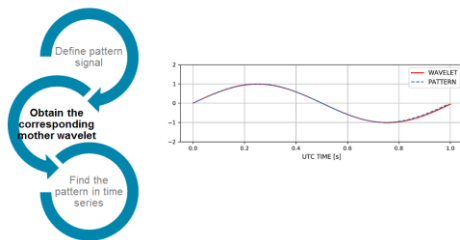In this case the mother wavelet is the sine function:
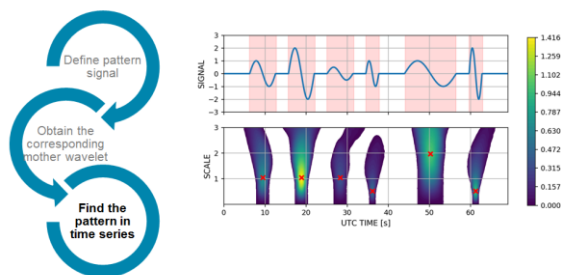

Figure 7.Mother wavelet : STEP2

Figure 8.Find pattern in time :STEP 3

Examples

This section shows cases of the usage of wavelets for the identification of manoeuvres corresponding to a Combat aircraft.

a) Rapid Roll

The following figures correspond to Flight Quality test on a Combat aircraft which contains a series of Rapid Roll (lateral directional manoeuvre).

- The first step is to select the pattern on the STK_PARAM parameter:

Figure 9.Define pattern

- The second step is to obtain the corresponding mother wavelet:
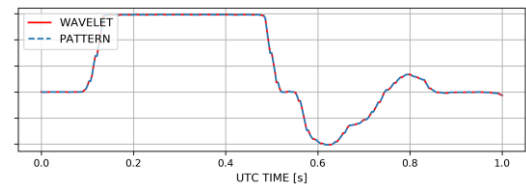
Figure 10.Mother wavelet

- The third step is to find the pattern in to the time series.

Figure 11.Find pattern in time

b) Lateral directional triplets

The following example will demonstrate that it is possible to use this technique to identified complex manoeuvres like lateral directional triplets that is compound by three simple doublets. The input of this manoeuvre is generated mathematically and applied directly to the Flight Control System (FCS Bias input).

First it will use the FBI input to define the pattern and second the Rudder parameter, the times slices of each identified manoeuvres will be the same in both cases.

- The first step is to select the pattern on the FBI_PARAM parameter:

Figure 12.Define pattern

- The second step is to obtain the corresponding mother wavelet:
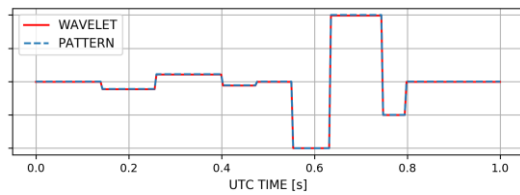
Figure 13.Mother wavelet

- The third step is to find the pattern in to the time series.
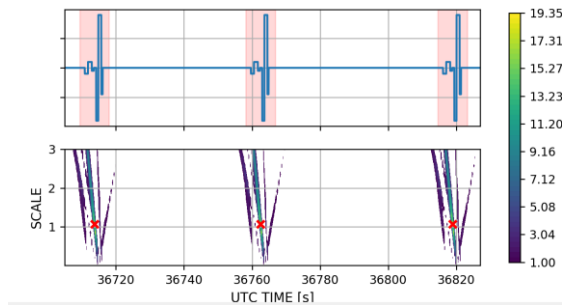
Figure 14.Find the pattern in time

The following example the Rudder parameter is used as a pattern:

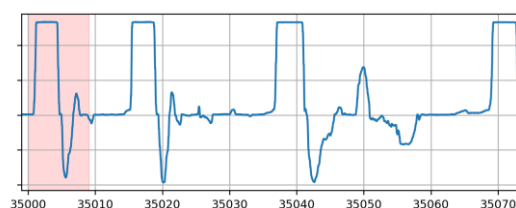- The first step is to select the pattern on the DR parameter:
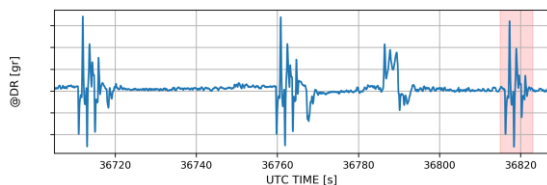
Figure 15.Define the pattern

- The second step is to obtain the corresponding mother wavelet:
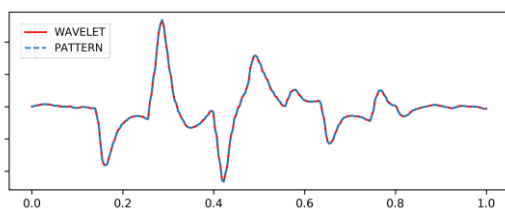
Figure 16.mother wavelet

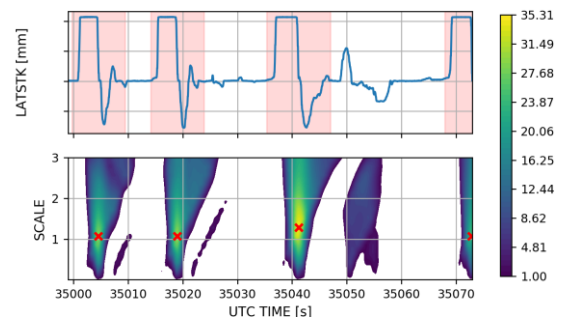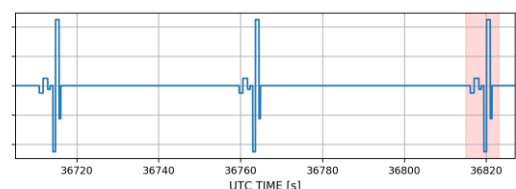- The third step is to find the pattern in to the time series.
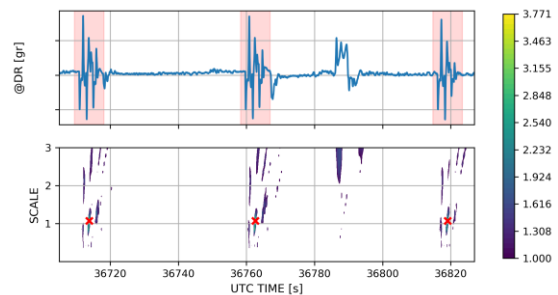
Figure 17.Find the pattern in time

Find Pattern Tool

The tool to find patterns using wavelets is an in-house software written in Python.

First a library of mother wavelets corresponding to each manoeuvre is generated.

This library is composed by synthetic and real manoeuvres.

Once the library is created, the steps for the automatic manoeuvre detection are the following:

a) Select the fight to find the manoeuvres.
b) Select the initial and final time (by default is applied to the full flight
c) Select the list of flight parameters used to search the pattern.
d) Select the wavelet from de library and the correlation and scale margins.
e) Use the wavelet to find Find all the manoeuvres selected in the flight.
f) Store the time slices corresponding to each manoeuvre in the data base.

The following figure shows the interface of the tool.
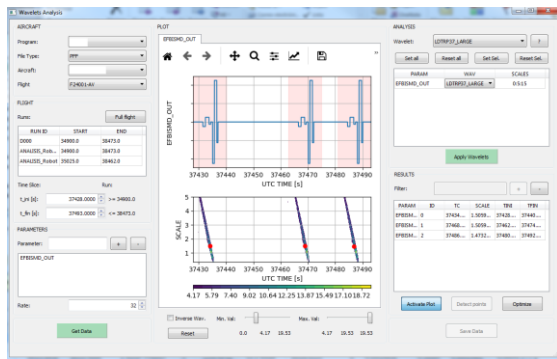
Figure 18. Find Pattern Tool

Conclusions

Nowadays, the Flight Test analysis process is a mostly manual procedure.

The main idea of BMAD is to improve analysis chain by:

a) Minimizing hand-made processes, prone to human error (machine learning techniques).

b) Automatic error detection through Big Data analysis procedures.

c) Reducing the time need to generate reports and documents.

d) Improving data access throw digitalization and webs tools.

References

[1] Paola Andrea Quiñones Roa. "TRANSFORMADA WAVELET, LOCALIZACIÓNTIEMPO FRECUENCIA Y ANÁLISIS DE CIRCUITOS RLC ", Universidad Distrital Francisco Jose de Caldas. Facultad tecnológica.

[2] Liesner Acevedo Martinez. "Computación paralela de la transformada Wavelet; Aplicaciones de la transformada wavelete al álgebra lineal numérica ». Departamento de Sistemas informáticos .Universidad Politécnica de Valencia.Mayo 2009

[3] V.J. Bolos,R. Benitez ."The wavelet scalogram in the study of times series". XXIII Congreso de Ecuaciones Diferenciales y Aplicaciones .XIII Congreso de Matematica Aplicada. 2013.

[4] Noelia Rillo Pastor. "Introducción a la teoría de wavelets" .Departamento de matemática aplicada y análisis , universidad de Barcelona .2005

[5] Amara Graps. "An indroduction to Wavelets".

[6] Wavelets and theirs applications. Michel Misiti, Yves Misiti, Georges Oppenheim and Jean-Michel Poggi

Machine Learning Assistant for Aircraft Acceptance

Rodrigo López Parra
Airbus, Paseo de John Lennon S/N, Getafe, Madrid (Spain)
rodrigo.l.lopez@airbus.com

Abstract:

The task of testing an aircraft to be considered "ready for delivery" is a task that implies a lot of work, involvement of pilots, flight engineers and it demands significant costs. Optimize the time of acceptance flights is critical to save a lot of time and money. Currently the flight test engineers are responsible for supervising and planning all the maneuvers that are required to verify that all systems work properly.

The objective is to create a machine learning system which will analyze all the actions carried out in the aircraft during acceptance flights, and it will learn when to perform them, taking into account the maximum information available.This machine learning system must be able to perform the task currently performed by the flight engineer and lead the pilots in the maneuvers to be performed, also checking if they have been performed correctly.

Moreover, controlling thousands of variables that exist in an aircraft is only suitable for an artificial intelligence entity, which might detect possible problems and warn them with enough time to avoid major problems.This system could become an intelligent assistant essential in any environment.

Key words: machine learning, artificial intelligent, big data, deep learning, predictions

Introduction

New technologies are evolving at a very fast speed, as well as the programming paradigms that are progressing to achieve increasingly complex and smart systems.

Tab. 1: Programming paradigms

A programming language is a problem-solving tool	
Imperative Style Program	algorithms + data
Functional Style Program	functions * functions
Logic programming	facts + rules
Object-oriented Style	objects + messages

These have been the programming paradigms more important in the last decades.

However, we now have huge amounts of data, of which, new tools are needed that allow us to analyze hundreds or thousands of different variables at once, obtaining new results and knowledge.

Just as in his day, the pioneer Alan Turing created a machine to be able to evaluate the mathematical calculations much more quickly than any team of human beings, today we need machines that autonomously analyze all the millions of data we have in different contexts, and are able to detect anomalies where we do not see them, and even predict and recommend actions as any human being would do.

This is where the new tool development environments come into play to create automatic learning machines.

Machine learning applications make use of patterns in the data to make predictions rather than needing to be explicitly programmed.

Actual System, ASSPA

Based on the acceptance programme for A400M aircraft manufactured on the final assembly line in Seville, Spain. There are currently more than 100 aircraft delivered to customers, which have had to go through a series of tests and flights to check every single part and system of the aircraft.
There is an acceptance manual that collects all the information with the results obtained in each of the tests and manoeuvres performed on each aircraft. This is the main basis for the customer to accept the aircraft.

Currently, we work with an application that uses the latest programming technologies called ASSPA (Automatic aSSistant Production Aircraft), which allows us to monitor and collect the necessary data from each maneuver performed.

We can see in *Figure 1* the standard profile of an acceptance flight for an A400M aircraft. This profile covers each of the maneuvers that pilots must perform at each altitude (flight level FL).
And in *Figure 2* we can see a data collection screen for the specific maneuver of Trim at FL 3XX (3XX00 feet)

Fig 2. Check Point of AC Trim at FL3XX

Fig 1. Standard Flight Profile for A400M Aircraft

This software helps flight engineers to collect data and manage the maneuvers that must be done at any moment. Collecting the data manually is practically impossible and each hour of flight is quite costly. In fact, this tool has become essential for aircraft acceptance tests. Great efforts have been made to automate the activation and data collection of certain manoeuvres without human intervention. For this purpose, certain variables such as flight phase, speed, altitude, engine power, data sequences, etc. have been taken into account with good results. Although taking into account all the casuistic that can occur in an aircraft, it is tremendously complex.

Remark
On the other hand, the aircraft's acquisition system is able to collect and record more than 250,000 different parameters during flights that can last up to 8 hours. Of these parameters, the ASSPA system uses only about 3000 parameters. That is, only 1.2% of the available data is used, and the current system can be considered quite advanced. A team of approximately 20-30 people is involved.

Who can extract the information we do not see from the 98.8% of data that is currently not used?

Machine Learning

What is a Machine Learning?.

"*A learning machine is one that learns by itself to find patterns without being specifically programmed by a human being.*"

This technology is at the heart of artificial intelligence and big data management.

Machine learning works by feeding an "algorithm or model" with input data that collects observations from the past, and builds a model to predict and classify new observations not known by the algorithm, imitating a human cognitive process.
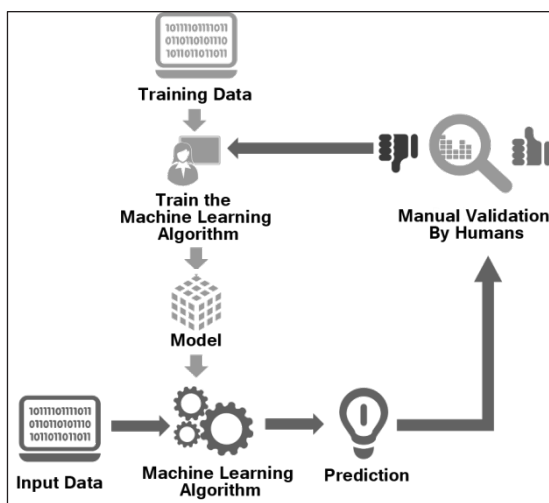

Ilustración 3. Machine Learning

Initially we do not know what we are looking for, we do not know what there is in that huge amount of data, we are not able to analyze parameter by parameter the 250,000 parameters and much less find the correlations between them if any. We need "something" to dive into all that data and look for "things" by itself that we don't have programmed.

A learning machine consists of three main parts:
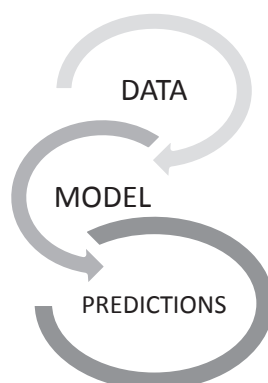
We have the data, although we could need to process it as input to the model.
The model is the "algorithms" that need to be trained with the data in order to make predictions.
And the predictions are the conclusions that the machine is able to detect by itself in the data.

There are four main types of learning algorithms in Machine Learning: supervised, unsupervised, reinforcement and deep learning.

Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output. $Y = f(X)$

The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.

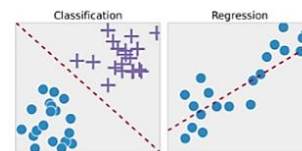For instance, regression and sales forecasting.

Ilustración 4. Classification, Regression

Unsupervised learning is where you only have input data (X) and no corresponding output variables.

The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

For instance, Clustering, Splitting, ...

Ilustración 5. I can see a pattern

Reinforcement
Identify which actions a software agent should choose in a given environment in order to maximize some notion of "reward" or "penalty"

Ilustración 6. Reinforcement learning

All these learning systems, allow us to classify, predict, form the way to act, and are very powerful tools. But we're looking for something even more advanced.

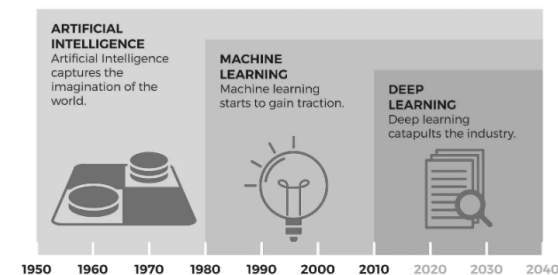## Machine Learning - Deep Learning

Ilustración 7. Evolution of artificial intelligence

In the figure we can see the evolution of AI (artificial intelligence) in the different decades.

While machine learning works with regression algorithms or decision trees, deep learning uses neural networks that function much like the biological neural connections in our brain.

It is extremely beneficial to data scientists too who are tasked with collecting, analyzing and interpreting large amounts of data.

The possibilities of creating learning machines for different purposes are endless. And it also depends on where our own imagination is able to reach. And it is in this field where we are interested.

Anomaly detection

We are at the beginning of our study to in a first step be able to detect anomalies in any equipment of the aircraft. Even be able to detect that an equipment is about to end its life time, before it fails completely.

Example of X-Y anomaly detection.

In case of two-dimensional data (X and Y), it becomes quite easy to visually identify anomalies through data points located outside the typical distribution.
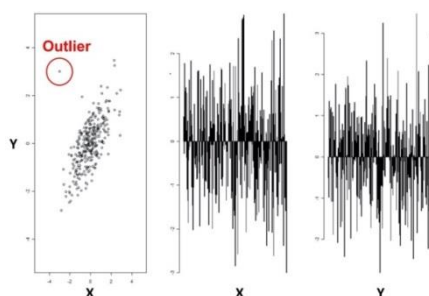

Ilustración 8. Sample anomaly X-Y

However, looking at the figures to the right, it is not possible to identify the outlier directly from investigating one variable at the time: It is the combination of the X and Y variable that allows us to easily identify the anomaly. This complicates the matter substantially when we scale up from two variables to 10–100s of variables, which is often the case in practical applications of anomaly detection and in our case.

Here is where we begin to see the need to create more complex learning machines, able to contrast and monitor hundreds or thousands of variables at once and find correlations between them.

Neural Networks

Essentially, neural networks are composed of layers of computational units called neurons, with connections in different layers. These networks transform data until they can classify it as an output. Each neuron multiplies an initial value by some weight, sums results with other values coming into the same neuron, adjusts the resulting number by the neuron's bias, and then normalizes the output with an activation function.
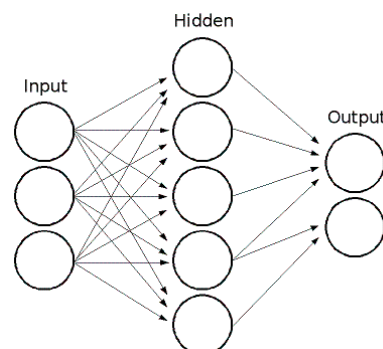

Ilustración 9. Neural Network

A key feature of neural networks is an iterative learning process in which records (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all, cases are presented, the process is often repeated. During this learning phase, the network trains by adjusting the weights to predict the correct class label of input samples.

Artificial Neural Network - Autoencoder

This is a popular type of neural network with which we have started to create our first projects because could be fit with our goals.

An autoencoder is a type of artificial neural network used to learn efficient data codings in an unsupervised manner. The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for dimensionality reduction. Along with the reduction side, a reconstructing side is learnt, where the autoencoder tries to generate from the reduced encoding a representation as close as possible to its original input.

Architecturally, the simplest form of an autoencoder is a feedforward, non-recurrent neural network very similar to the many single layer perceptrons which makes a multilayer perceptron (MLP) — having an input layer, an output layer and one or more hidden layers connecting them — but with the output layer having the same number of nodes as the input layer, and with the purpose of reconstructing its own inputs.
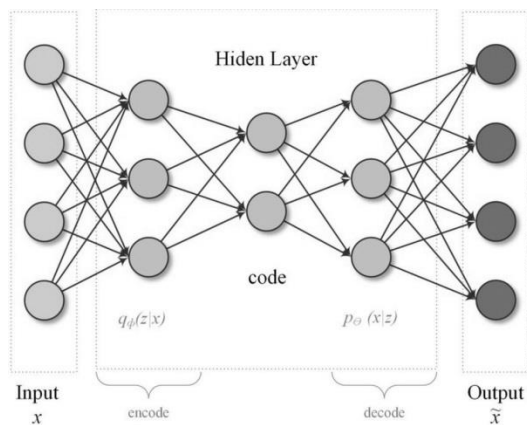

Ilustración 10. Autoencoder Network

<u>In the context of anomaly detection and condition monitoring</u>, the basic idea is to use the autoencoder network to "compress" the sensor readings to a lower-dimensional representation, which captures the correlations and interactions between the various variables.

The autoencoder network is then trained on data representing the "normal" operating state, with the goal of first compressing and then reconstructing the input variables. During the dimensionality reduction, the network learns the interactions between the various variables and should be able to re-construct them back to the original variables at the output.

The main idea is that as the monitored equipment degrades, this should affect the interaction between the variables (e.g. changes in temperatures, pressures, vibrations, etc.). As this happens, one will start to see an increased error in the networks re-construction of the input variables. By monitoring the re-construction error, one can thus get an indication of the "health" of the monitored equipment, as this error will increase as the equipment degrades.

Machine Learning for Aircraft Acceptance

For our specific study we have data from new aircrafts, which we can choose those that we know have a perfect behavior of all systems, and that will be used to "train" our machine learning, to make it learn how is the correct behavior of each system.

Once our machine learning has been "trained" with this data, it can be used to check the correct functioning of the new aircraft that leave the factory, and to check the correct functioning during the whole life cycle of the aircraft.

We would have the perfect real-time guardian of the integrity of our entire aircraft.

And once this has been achieved, we will move on to a systems that are able of telling pilots when and how to act, without human intervention. **The Machine Learning Assistant**.

References

[1] Vegard Flovik, Article, How to use machine learning for anomaly detection and condition monitoring. https://towardsdatascience.com/ (2018)

[2] Jay Shah, Article, Neural network learning. https://blog.statsbot.co (2017)

eDrive-ready telemetry systems for rotating applications

Gerhard Spitz, Technical Director

AXON Systems GmbH

Wildmoos 5, 82266 Inning am Ammersee, Germany

g.spitz@axon-systems.com

Abstract:

Telemetry systems for rotating installations providing contactless data transmission is state of the art in many branches of mechanical industry. The measurement of physical quantities on rotating vehicle and machine components is an important component in the development of modern drives. Inductive power supply in combination with RF data transmission ensures contact-free and therefore wear-free operation. In addition to the anyway challenging environment, modern drive concepts requiring latest telemetry systems to be adapted to some new challenges, especially strong EMC impacts in the surrounding of the electric engine.
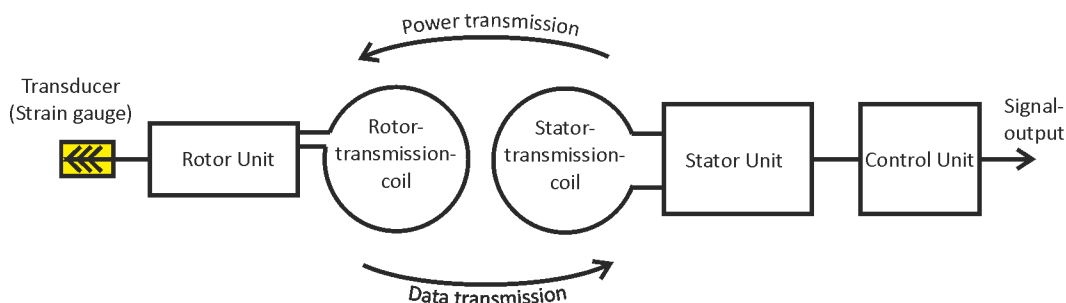
Telemetry systems for rotating installations consisting of:

Rotor Unit

     Sensor supply

     Signal acquisition and processing

     RF Data transmission

Stator Unit

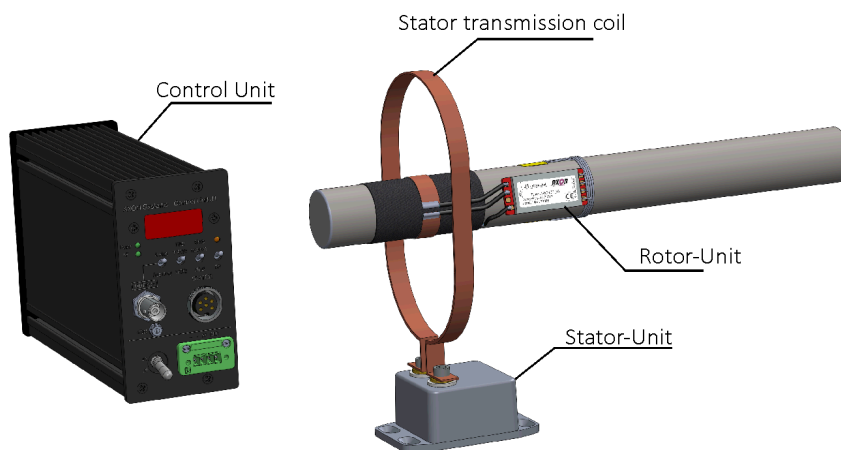   Inductive power supply

     RF Data reception

Control Unit

Central processing unit

RF reception, decoding

Signal playback

Inductive power control

A classic powertrain, containing a motor, a gearbox, propellershaft and 2 or even 4 half shafts can require many different torque-, force- and temperature-measurements on its rotating parts. Certainly, the most common telemetry measurement for automotive applications is measuring torque on drive shafts during test drives and long-term testing. But also, the gearbox input, crankshaft or cardan shafts necessitating torque measurements. Temperature measurements for example can be necessary on brake discs, clutch- and differential plates or even single gearings.

Highest possible efficient drivetrains are going to be increasingly important, what also makes the constructions complexity enormous increasing.

Particularities on eDrive applications:

High packaging density

High operating temperature

High RPM

High EMC load

Electric engines in Automotive applications coming with extremely high power, controlled by steep edges which leads to immense EMC load on the drivetrain. Shafts might even work as "transmitting antennas" for the disturbances. As for example torque or temperature measurements are based on transducer signals in the range of μV, it is obvious that these environmental conditions can lead to signal spikes or even total data loss.

As packing density in mechanical technology is increasing, installation space for additional measuring technology or telemetry systems is continually shrinking. Telemetry needs to be both, small and flexible in its installation to use remaining installation space in an optimal way.

Meet the requirements of modern eDrive applications

Rotor Units of latest telemetry systems needs to be equipped with an extreme powerful RF transmitter by meeting the RTT&T-standards coming with the CE-certification at the same time. Also, the Control Unit is coming with a new, extreme sharp filter for best possible data reception.

Stator-Units with option "E" additionally are equipped with an EMC suppression right on the receiving antenna which allows an early as possible filtering of any disturbing fields. The inductive power transmission of the Stator Unit is not affected and remains extremely powerful. Besides EMC-interferences also the size of the telemetry components needs to be considered. The arrangement of Rotor Unit, Stator Unit, antenna and transducer needs to be possible in different ways to use the space in an optimal way. Rotor Units are available in different designs in order to meet the requirements of any individual application. Electronics are available in rectangular shape with aluminium housing, round in carbon housing or flexible without housing to be "wrapped" around the shaft.

The Control Unit works as the central processing unit, controlling the inductive power supply and the data reception. It allows to be different Stator Units connected as well as shaft diameters between 5cm to 2 meters.

As modern drivetrains can develop a lot of torque, particularly while misuse-testing half shafts can break. To avoid losing a telemetry installation if a shaft breaks, the torque measurement can be carried out on gearbox output flanges as these components are even more robust against torque peaks. A broken half shaft can be easily replaced, and the torque measuring device remains undamaged.

Gearbox output flanges as well as shaft installations with a measuring range of just a few Nm as well as even MNm – all can be operated with the same system without the need to change any adjustments.

To arrange a torque measuring installation, no mechanical manipulation on the test object is needed. Strain gauges, telemetry, antenna and wiring are bonded to the shaft by using advanced adhesive technologies.

The same applies for temperature measurements. Full flexibility allows measurements in electric engines, clutch plates, joints, gearings and many more. Latest systems can measure up to 1.250°C. Those systems are especially designed for measuring temperatures on brake discs, such as necessary on dynamometer test benches. When used in big test benches, EMC will be an issue as the test stands are usually equipped with large frequency converters which can affect electrical installations due to EMC impacts. Temperature telemetries with "option E" allow safe and reliable measurements in this difficult environment.

Automated closed-loop testing of cyber-physical systems

Gonçalo Valadas (Principal Engineer), _Nuno Bustorff Silva (Business Development Manager)_

Critical Software S.A., Pq. Ind. de Taveiro, Lt 49, PT-3045-504 Coimbra, Portugal
goncalo.valadas@criticalsoftware.com, _nuno.bustorff@criticalsoftware.com_

Abstract: Modern systems envisages fusion of technologies across the physical, digital and biological worlds, an incoming era of cyber-physical systems powered by artificial intelligence (AI), the Internet of Things (IoT), autonomous vehicles and big data. More than any other industry, Aerospace & Defence innovation and production demand precision and high-quality products. Aerospace & Defence Industry must adopt features of modern product development like Agile methodologies, DevOps, and tests automatization, to accelerate time to market and business value.

The increasing complexity and validation costs raises an unaffordable trend, particularly in the evolution from hardware-based systems to modern software-based systems, where the costs to develop, integrate, and maintain software continues to grow at an unpredictable rate.

With the objective of providing an infrastructure for testing high integrity cyber-physical systems in closed-loop, Critical Software has designed and developed an end-to-end process, integrating Agile methodologies, client toolchain and an automation testing platform. This integrated solution enables real-time control and joint and flexible integration of simulation models and physical components or subsystems to build a closed-loop test environment adaptable to any development environment.

This paper presents Critical Software's System Validation approach to reduce both the time and cost of testing high-integrity and complex systems for the Aerospace & Defence Industry.

Keywords: Defence, Aerospace, Agile, DevOps, Testing Platform, Closed-Loop Testing, Regression Testing, Security Testing, Cyber-Physical Systems, Complex Systems, Validation

1. Introduction

Developments in Artificial Intelligence (AI), the Internet of Things (IoT), autonomous vehicles and big data enabled the creation of new aerospace and defence systems that historically were mainly hardware-based but are gradually becoming more software-defined [1]. Software defined means that items or functions that were mainly physical have now become virtual, controlled and automated by software [2]. These systems integrating computation, networking and physical processes are called cyber-physical systems. Examples of this systems are smart measuring systems, robotics, wireless sensor networks among others.

This evolution was triggered by the need of OEMs to have faster access to the market, boost efficiency and produce economy of scale. The usage of virtualization, cloud computing and artificial intelligence made possible new technology developments and complex networks generating huge amounts of data to be analysed and measured [3] [4].

At the same time, the effort and time needed to validate these systems also grow at an exponential rate because of the increasing configurability of these systems, the number of external interfaces, and the need for safety-related certification.

Within this scenario of constant evolution and adaptation, it is impossible to continue to execute these developments in a classical waterfall lifecycle. Verification and Validation activities must be performed from the beginning of development and an iterative approach that enables the system to be developed and validated incrementally is required. The new Agile approach seems to solve all these problems, but a formal process must be defined to guarantee that the System Under Test is able to accomplish not only the functional requirements but, sometimes even

more importantly, meets all non-functional requirements when complete [5].

Validation activities shall be performed over and over again, regressively, validating disruptions, so an automated approach is at once possible and necessary. Although it is understandable and easier when it comes to test software, it becomes more complex when the testing environment also incorporates hardware.

Beside the automation, another important thing is to have all these activities integrated into the customer's build and deploy environment. Establishing a continuous integration environment improves reliability and robustness of the validation process, therefore also improving the reliability and robustness of the system under validation as well as its overall quality.

To overcome these challenges, Critical Software defined a process to validate high integrity systems, addressing the validation of software and hardware-in-the-loop and resulting in important cost and time savings, as presented in the following sections.

2. Integrated Validation Process

Critical Software's integrated validation process enables the specification, implementation and execution of the necessary test cases to validate the System Under Test. It is composed of four main activities as outlined in the figure below.
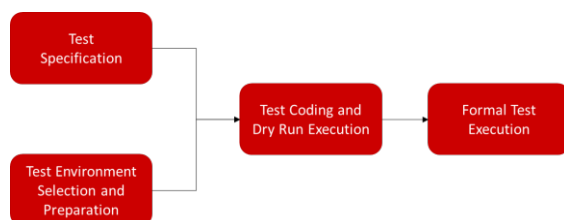

Figure 1: Process for system validation

2.1 Test Specification

In order to properly establish a test specification, the first step should be to gain knowledge about the system. This is done through performing interviews with the development team, reading the system requirements specification (SSS), and reviewing the existing interface control documents (ICDs).

The validation and integration strategies are defined at this stage. The testing methods and activities to be performed are identified, including the different types of tests that the validation process will have to execute (e.g., security, performance, usability, other functional or non-functional tests). The criticality of requirements and system components, software or hardware, are thoroughly detailed. Certification against safety standards, e.g. IEC 61508 or its industry specific derivatives, require that these activities are documented to provide evidence to the certification authority.

Then, the test plan to validate the system is created, which covers all these definitions. Because the test outputs will be validated against expected values, in this phase the applicable thresholds used in resulting assessment of non-deterministic systems are also defined, as these systems are not usually precise but will fit into a range of acceptable outputs.

Mapping to the Agile world, this could be the Product Vision and Backlog creation phase. The test sets are the epics and the test cases are the user stories described with the acceptance criteria.

2.2 Test Environment Selection and Preparation

The environment selection and preparation are important in establishing the basis for test automation. The identification of the target system, both physical and logical interfaces, are the most relevant aspect in environment preparation.

To perform the automation of integration and system tests, Critical Software have developed a System Validation Facility platform, an asset providing the basis for the development of automated test sets that will exercise the system and capture their outputs to validate them against expected values.

The system validation facility is an automated execution environment responsible for test execution, including the simulation of the necessary input conditions and the evaluation of response values. It also compares expected and actual test case output values, validating

them according to the acceptance criteria defined for that test case [6].

Another important feature of the system validation facility is the ability to perform fault injection at hardware level. The ability to inject faults at hardware level enables the tester to simulate internal faults on the System Under Test and evaluate the reaction to errors inserted in different parts of the system, such as the processor registers, the memory, or the application code. Faults are injected with minimum interference with the target system workload, making it almost non-intrusive. This enables a very robust test capability of the safety and reliability mechanisms which are not easily testable.

In the Agile process this phase matches sprint 0, where the implementation sprints are prepared, and the backlog is prioritised.

The environment preparation also includes the setup of a continuous testing environment where automatic build, test and static analysis is performed when a change occurs in the version control system repository. At this point, three major verifications are being performed: (a) that the build system is correctly generating the target system; (b) the target system passes all defined test suites; and (c) the system code complies with the defined code quality rule set.

2.3 Test Coding and Dry Run

Test coding activity is where test cases defined in the specification are implemented. This is a manual coding activity where developers write the code that executes steps defined in the test case. The usage of Artificial Intelligence techniques in this process to automatically generate test case code would be beneficial as systems become more complex. Input vectors that stimulate the implemented test cases will be defined separately in order to change these inputs without having to rebuild the test cases in question.
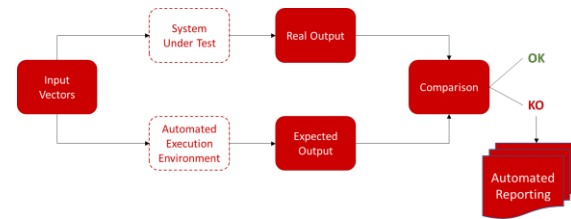

Figure 2: Test Execution Scenario

The first item to validate is the correct implementation of system interfaces. These will transport the inputs of the test cases to the System Under Test and return the relevant outputs.

During this activity, dry runs of the test suites shall be performed to guarantee that tests execute as specified and, if not, are able to debug and correct the test. At this time, the first issues are raised allowing developers to fix arising problems before the formal test runs.

The usage of an Agile methodology to implement the tests and verify their correctness enables incremental validation of the test implementation and the System Under Test, identifying the first issues to be raised during these first dry runs.

Due to automation, these test suites can be run when required to perform *ad hoc* test runs or execute regression testing integrated into the continuous integration environment.

When the test coding is finished and all tests are behaving as specified, the automation testing environment creation process has been completed.

2.4 Formal Test Execution

The formal test runs are executed according to plan, running the released tests against the released system.

These activities are fully automatic and consist in the execution of all tests which are to be formally used for the certification record. They automatically produce a report with the validation evidence at the end of the process.

When the entire system is finished, a final formal test is executed to produce the documentation used for certification purposes. At this point, the development must be frozen and the version to be formally tested should be released.

After this, the System Under Test is ready for certification, but the automated environment created can be continuously used to support future enhancements of the system. If those enhancements change the expected behaviours or outputs, the automated execution environment must be adapted accordingly to support the new behaviour.

3. Conclusion

This paper presented an integrated process allowing OEMs to enhance their product development by automating the validation process and to reduce the total cost of ownership of critical systems through the automation of several Verification and Validation activities.

The usage of the System Validation Facility enables the automation of the tests using hardware-in-the-loop. The automation of test suites reduces the time and effort taken to realise test campaigns and eases the process of producing certification records and validation evidences. The ability to perform regression testing gives engineering teams the time to focus on high value tasks instead of spending time re-running test suites manually.

Fault injection at hardware level enables the tester to simulate internal faults on the System Under Test and enable the validation of hardware built-in tests as well as the reaction to errors inserted in different parts of the system such as the processor registers, the memory, or the application code.

The usage of a SCRUM approach in developing the automated execution environment is beneficial as it means change can be managed as an integral part of the process and also shortens feedback loop.

In future work, the Critical Software's System Validation Facility aims to use Artificial Intelligence to generate test cases that will simplify the test specification activity and use Artificial Intelligence to develop test data for use as input vectors in test coding and dry run activities.

Based on the experience of some of our clients, this process reduced the TCO of critical systems by up to 50%.

References

[1] Grant J. McDonald, Yvon Audette (2018), 'Making the move to Industry 4.0', published in Canadian Defence Review Magazine, February 2018.

[2] Sadiku, M & Nelatury, Sudarshan & Musa, S. (2017). 'Software defined everything', 4. 48-50.

[3] Daliri, A, Das, R, Orifici, A, Marzocca, P and Cole, I (2019), 'Virtual Design, Optimisation and Testing (VDOT) framework for innovative sustainment', in Proceedings of the 18th Australian International Aerospace Congress (AIAC18: 2019), Melbourne, Australia, 24-26 February 2019, pp. 122-128.

[4] Kim, Beom-Su & Kim, Ki-Il & Shah, Babar & Chow, Francis & Kim, Kyong Hoon. (2019). 'Wireless Sensor Networks for Big Data Systems'. Sensors. 19. 1565. 10.3390/s19071565.

[5] Valadas, G. (2019), 'White Paper - Using Agile to Develop High Integrity Systems' published by Critical Software, October 2019.

[6] A. Chaves, R. Maia, C. Belchior, R. Araújo and G. Gouveia (2018), 'KhronoSim: A Platform for Complex Systems Simulation and Testing' 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), Turin, 2018, pp. 131-138.

Job Process Architecture Based On PostgreSQL Database

Gonzalez-Martin Moises
Roses-Sanchez Diego
Garcia-Garcia Jorge (ALTRAN)
Flight Test – Airbus Defence and Space
moises.gonzalez@airbus.com
diego.roses@airbus.com
jorge.garciagarcia@altran.com

Abstract: The purpose of this paper is to show an integrated Job Process architecture that allows executing heavy jobs that can be parallelized at process level without the need to use complete Big Data Architecture. This solution can be seamlessly integrated into any Data Centre, be it in a Linux or Windows environment. PostgreSQL is shown as the system to coordinate the workflow of execution of these processes.

Key words: Job, Process, PostgreSQL[1], Parallelization, Batch processing

Introduction

The flight test world has to deal with large amounts of information coming from test activity recordings, which needs to be analysed in different ways. Many of the analysis involve heavy computational processes that take long periods of time, and are repeated many times with different sets of data. When there are a high number of test activities that need to be processed, it can take many days to get the results.

There are many Big Data and cloud computing platforms like Hadoop YARN[2] or Microsoft Azure Batch[3] that could handle these processes, but there could be some downsides: complex, expensive and cloud dependent that may not be allowed when it comes to military data.

The Job Process architecture proposed enables the parallelization of data analysis processes on a virtualized computing platform to accelerate results and report generation.

Job Process Architecture

To provide a simple solution, we propose to create a task queue in a database that will run on a set of virtual machines connected to the "local" network.
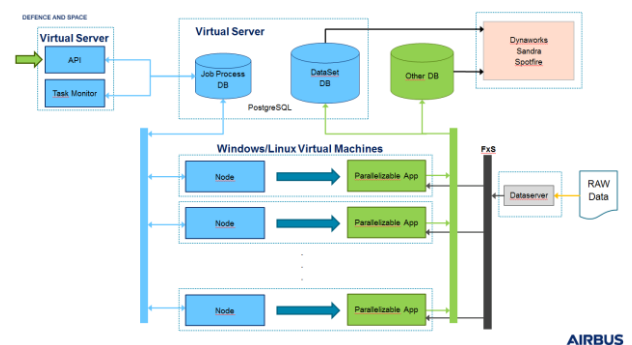

Figure 1. Job Process Architecture

Here, we can identify some definitions:

- **JOB**: A group of tasks that could have a specific order of execution.

- **TASK**: The minimal piece of analysis, performed by a console executable process.

- **NODE**: A Windows/Linux computer connected to the same network, running a "Job Process" Windows Service or Linux Daemon for task queue execution.

- **Parallelizable App**: Any app accessible from the "local" network such as Matlab or Python processes that can be run in console mode, receives an argument in JSON[4] for the inputs, and generates results unattended.

- **FxS Dataserver**: Provides access to test activity recordings.

- **DataSet DB**: The system provides a database to store task results.

The core of this architecture is the service/daemon running on the Nodes. It will periodically query the database for available task that comply with a set of rules. Once a task is found, it will be locked down to avoid double execution and if lock is granted, the executable process associated to the task will be run in the node. Once the task is completed, its status is updated on the database and the service will continue looking for new tasks.

Database structure

Figure 2 shows the tables used to handle the Jobs scheduling and processing.
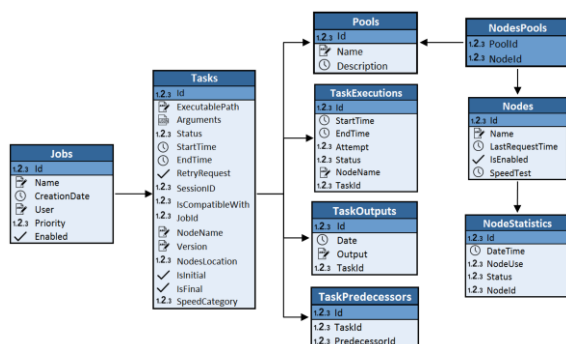
Figure 2 Job Process Database diagram

JOBS

Defines groups of tasks requested by a User with a specific priority level.

TASKS

Each of the processes that comprise the Job, with information to run the process and properties that deterime when and where this task will be executed.

POOLS

It allows the reservation of a set of computers for the execution of specific tasks, generally for any type of analysis that should have priority over any other task.

TASKEXECUTIONS

Used as a locking mechanism to prevent multiple nodes executing the same task. An index on TaskId, Status and Attempt fields, ensures that only one Node will be able to lock the task for execution.
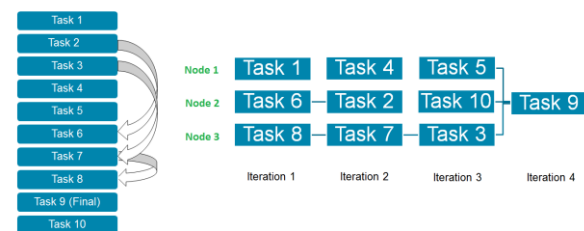
TASKOUTPUTS

Stores the standard output of the task process if any. It could be useful to trace execution errors.

TASKPREDECESSORS

Defines the order of execution of tasks for a given Job.

Figure 3. Task Precedence example

Figure 3 illustrates an example of the order of task execution on 3 nodes with predecessor tasks defined. This is what happens in the first iteration:

Task1 has no predecessors so it will be the first executed task by node 1.

Node 2 will try to execute Task 2, but it needs that Task 6 be executed first so, it will run Task 6.

Node 3 will check Task 3, which needs Task 7 to be executed first, and it also needs Task 8 to be executed first, so it will run Task 8.

The same process continues in the next iterations, if the Task has a predecessor, it will try to run that task instead. If a task is set as Final, it won't run until all other tasks are completed. Also, if there is an Initial task, no task will be ran until that task execution has finished.

NODES

The Windows service / Daemon installed on the Nodes stores information about the characteristics and status of the Host in this table. It is used by the Job Monitor Service to detect blocked tasks or nodes.

NODESTATISTICS

Used by Job Monitor Service to store information about task execution performance to generate statistical reports.

API

The System exposes an API that allows users to develop tools to manage and request Job execution.

It also allows gathering information about the system and the status of jobs, tasks and nodes.

JOBPROCESS Service / Daemon

This is the core of the architecture. It is a service that can be installed on any computer connected to the local network. It turns it into a Node that will start looking for tasks to run, making system scalability really simple.
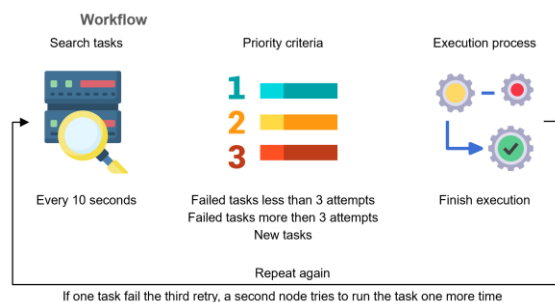
Figure 4 Service/Daemon workflow

The service queries the database every 10 seconds looking for a task suitable to execute.

FAIL CHECKS

Since the tasks that make up the jobs are external processes to the system architecture, we must bear in mind that these tasks may not work correctly, and some node may be blocked for uncontrolled reasons, so the system must be prepared to handle these situations.

Every task has 3 attempts to complete on the node that has picked it. If it continues to fail, a different node will try to run it once. This is to discard the issue is related to the node. The TaskOutputs table contains information about the task execution and could help to investigate issues.

When using the system DataSet database to store the results, if a task fails to execute, the Job Monitor service reverts the results to clear any incomplete data.

JOB MONITOR Service

This service runs in the computer where the API is hosted and it is responsible of system health monitoring.

Its main functions are:

1. Gather Node task execution statistics every minute.

2. Monitor and reset tasks running on nodes that has been unresponsive for more than 40 seconds, and rollback partial results.

3. Restart blocked nodes (available for VMWare virtual machines).

Conclusion

The Job Process System Architecture can be easily implemented to allow scheduling heavy computational tasks such as flight data analysis. Also, it can be used to run any kind of task that involves an unattended console App with JSON inputs.

Nodes can be created either with physical or virtual devices increasing the parallelization capacity to reduce Job execution times. Installation of the JobProcess service/daemon turns any computer connected to the system network into a node that will execute tasks.

Providing "intelligence" to nodes instead of having a task dispatcher makes the system simpler and easier to scale, and having a distributed architecture, a critical point of failure is avoided.

References

[1] https://www.postgresql.org/

[2] https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html

[3] http://azure.microsoft.com/en-gb/services/batch/

[4] https://www.json.org/json-en.html

BSS with Heterogeneous Sources

Vincent MARIE
NEXEYA Test Range Product Manager
Dedicated to the memory of Julien MALIVERNEY

Abstract

Telemetry ground stations have to ensure missions with always growing complexity (reception from various data provider, increase of data load, distribution on several sites) making them more and more expensive to maintain. Their refurbishment is usually carried out by successive iterations in order to reduce costs and secure the integration of new hardware and software bricks. One major consequence is the appearance of heterogeneous data formats representing the same telemetry data (CH10 packed, unpacked, PCM Data + Clock, raw PCM over IP, TMoIP, Hardware proprietary format, etc.). This article describes architectures and methods we implemented in order to assure the most efficient BSS / BSC processing of those heterogeneous data formats with various data quality factors (as DQM, DQE), time desynchronization of the sources, allowing to provide, as a selection or as a combination of telemetry data, the best telemetry stream to the end user.

Key words: Telemetry, Best Source Selector, Best Source Combination, Ethernet, Time Synchronization, Software Frame Sync

Introduction

For more than a decade, telemetry data formats have evolved as much on board as on ground installation. New data encapsulation formats have emerged (CH10, CH7, TMoIP, IENA, AFDX, Hardware proprietary format, etc.); transmission protocols have also evolved to come closer to standard network formats (TCP / IP, UDP / IP). Telemetry systems are complex installations which must be able to adapt to these changes without compromising their general architecture.

Telemetry Range with distributed architecture are connected through Ethernet network and Analog signals have been reduced to the minimum length. Ethernet assets are obvious: Standard cable connectors, worldwide protocol for data transfer from & to anywhere. That has been a great opportunity to reduce the cost of the telemetry installation with distributed architecture.

Large Telemetry Ranges with several remote sites have to proceed to Best Source Selection (or Best Source Combination) to ensure to get in real time the best telemetry stream without interruption and at any position of the target to track. To do this, it is required to concentrate all telemetry streams to the Telemetry Data Centre and this whatever the data format proposed by the hardware. Nowadays, installations evolve from Analog signal transmission to Ethernet transmission through mux / demux equipment. However, these evolutions can be expensive, they must therefore be able to be carried out iteratively, to reduce the cost but also to ensure a simpler validation of the modifications as well as to preserve the operational continuity of the installations. Nevertheless, BSS/BSC has still to be done from any Antenna on the range. Currently, Most of the Telemetry Ground Station equipment (receivers, decom stations) provide Ethernet outputs. However, the Ethernet protocol is not always following a standard and can be based on a proprietary Ethernet format.

BSS/BSC, which has to manage all these streams (Ethernet, Data+Clock, etc.), needs first to standardize the inputs in a common data format before proceeding to the Best Source Selection or Best Source Combination Algorithm.

This paper presents:

- The most common PCM formats and quality factors

- The different packet transmission methods

- The BSS/BSC architecture that NEXEYA experimented

- Some practical cases

A conclusion based on the case study results

Most common PCM format and quality factors

PCM CH4 Streams Description:
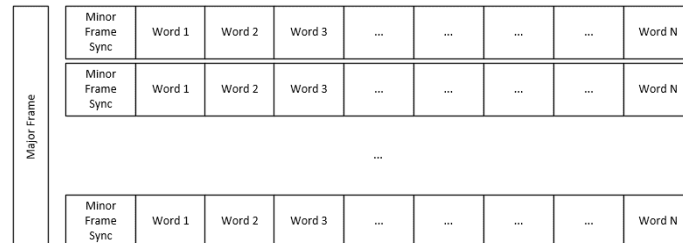
PCM CH4 are defined as follow:

Figure 1: PCM CH4 format

PCM Packet Timestamp

PCM packets can be dated in several ways:

- Packets are dated when the frame is transmitted. For example, each short cycle contains reserved words containing its timestamp

- Packet are timestamped by reception hardware. The reception equipment adds a proprietary dating header. This format may differ from one supplier to another.

Quality Factors

Generally, raw PCM frames do not contain any quality factor. It is however possible to calculate, using to the content of the minor frame, some quality factors using the following information:

- Number of bits in error on the synchronization word

- SFID value

- Parity errors on PCM words

- Calculation and verification of a CRC present in the cycle

- …

CH10 PCM Streams Description:

PCM Packet in CH10 streams are defined as follow:

Figure 2: PCM Packet in CH10 format

Three different transmission modes are available in CH10 mode

- Pack

- Unpacked

- Throughput

PCM Packet Timestamp

The PCM packets are timestamped using a 48 bits counter (100 ns precision)

The PCM packets are timestamped using the optional header "Intra Packet Time Stamp". If the header is missing, the packet is timestamped using the last received "Time Packet". If both packets are present, a correlation can be made to ensure data synchronization.

Quality Factors

Generally, PCM packets in CH10 streams do not contain any quality factor. It is however possible to calculate, using to the content of the minor frame, some quality factors using the following information:

- Number of bits in error on the synchronization word

- SFID value

- Parity errors on PCM words

- Calculation and verification of a CRC present in the cycle

- …

CH7 PCM Streams Description:

CH7 is an encapsulation of CH10 streams and proprietary network streams. The PCM data is generally sent in CH10 packets

TMoIP (IRIG 218-10) Streams Description:

TMoIP (IRIG 218-10) are defined as follow:

TMoIP Control Word	Raw Packet Payload

Figure 3: TMoIP Packet Format

Packet Timestamp

The TMoIP (IRIG 218-10) packet format does not contain any clock information that would be useful in reconstructing the PCM output at the receiving device. The standard makes a reference to using the IETF RFC 1889 Real Time Protocol (RTP) to provide clock recovery support for TMoIP (IRIG 218-10). The required RTP header fields are shown in the following figure and the IRIG 218-10 packet is placed in the payload of the RTP packet.

	0							1								2								3							W
00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
Version	P	X		CC				M			PT										Sequence Number											1
Timestamp																																2
SSCR Identifier																																3
IRIG 218-10 Header																																4
Payload Data																																

Figure 4 : RTP Header for IRIG 218-10 timestamp

Quality Factors

The TMoIP (IRIG 218-10) implements Data Quality Encapsulation (DQE) standard and Data Quality Metrics (DQM) standard as described in IRIG 106 Chapter 2 Appendix G9.

1	2	3	4	3	4	
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	
16 bits		12 bits	4 bits	16 bits		1024 −16384 Bits
Synchro Word		Reserved	VER	DQM		Data Payload

Figure 5: DQE format

The data payload is prefixed by a header including a synchronization word (0xFAC4), a version and the Data Quality Metrics (DQM).

TMoIP (IRIG 218-20) Streams Description:

TMoIP (IRIG 218-20) are defined as follow:

	0							1								2								3							
00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sequence Number																TSR	RES				FRG	MFSS		mFSS		PLD		VER			
Timestamp (seconds)																															
Timestamp (nanoseconds)																															RES

Figure 6 : IRIG 218-20 Header Format

Packet Timestamp

TMoIP (IRIG 218-20) are timestamped using a 64-bit timestamp in PTP format:

- 32-bit seconds field

- 30-bit nanoseconds field

- Timestamps are relative to 00:00 01 Jan 1970

Quality Factors

The TMoIP (IRIG 218-10) implements DQE standard and DQM standard as described in IRIG 106 Chapter 2 Appendix G9.

1	2	3	4	3	4	
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	
16 bits	12 bits	4 bits	16 bits			1024 –16384 Bits
Synchro Word	Reserved	VER	DQM			Data Payload

Figure 7: DQE format

The data payload is prefixed by a header including a synchronization word (0xFAC4), a version and the Data Quality Metrics (DQM).

PCM Ethernet Streams Description:

Ethernet is a wide world with open protocol and data format. This is the same with Ethernet PCM streams, different data normalization are available. This section introduces the Major cases.

Here is the standard format of a PCM Ethernet Message:

Figure 8: Example of Ethernet UDP message

IP/UDP header will not be explained here; only payload has interest. It is generally formatted according to manufacturer's Telemetry receiver, or following a standard (UDP Ch10, Ch7, iNet, PCM over Ethernet…).

Generally, the UDP payload consists in the following data:
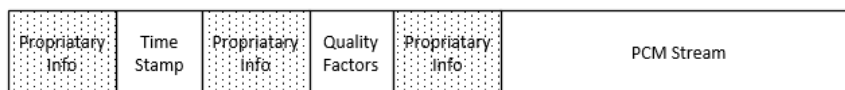
Figure 9: Example of UDP/IP payload

Common PCM Information:

The packet Timestamp

The quality factors (if provided)

The PCM stream

Packet Timestamp

Packet Timestamp is mandatory as Ethernet is non-deterministic protocol. Therefore, The Ethernet source of the PCM stream needs to have timestamped the packet before sending it on the Network.

Quality Factors

Quality factors also depends on the Ethernet Source. These information depend on the ability of the receiver. Generally, we have to deal with AGC, Eb/N0, and Lock Status.

Finally, Each Manufacturer sends the PCM data stream in a dedicated format with its own positioning & its own format (integer, float, double format…). It is then necessary to apply the right process to extract all the required information.

PCM Packet Data Transmission:

Three possibilities of PCM stream transmission are possible:

- Unpacked

- Packed

- Throughput

Unpacked mode is a synchronized data flow with PCM word set on a 16-bit or 32-bit word container with additional padding. Therefore, to manage this stream it is necessary to know the word container size and the alignment (right or left).
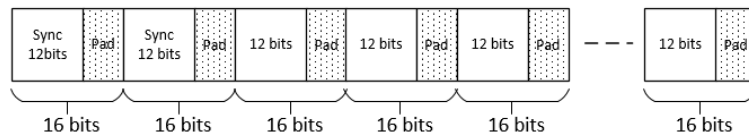
Figure 10: PCM Unpacked format

PCM word on 16-bit word container

Packed mode is a synchronized minor frame put on a 16-bit or 32-bit payload size. So Additional bits are added at the end to complete the minor frame on a 16-bit or 32-bit full size frame.
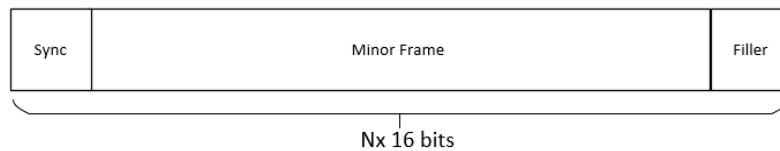
Figure 11 PCM Packet format

Throughput mode is used by telemetry receivers, which are not composed of a Frame Sync. So digital data are provided directly from the Bitsync Output and the stream is a raw data stream which could still be encrypted and not synchronized.
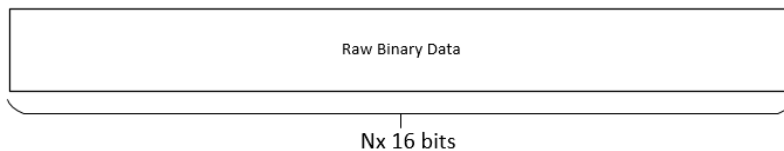
Figure 12: PCM Throughput format

These three formats can be possible in the Ethernet output stream of a TM station and in order to compare the contents, it is first required to standardize the format before proceeding to BSS.

BSS / BSC Architecture

With these PCM stream formats, it is now possible to determine the different required tasks to define a BSS / BSC with an architecture able to manage these heterogeneous streams. Regarding the previous chapter an architecture based on three layers is adequate:
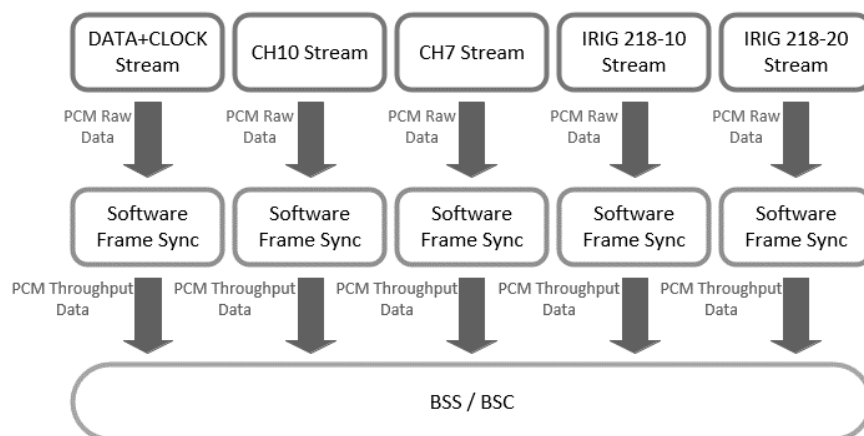
Figure 13: BSS / BSC Architecture

Proprietary Protocol layer is in charge of extracting the required information for the BSS from specific data format:

- Timestamp
- PCM stream
- Quality factor (if provided)

This layer will be specific depending on the provider (manufacturer) of the telemetry source. There will be as much Proprietary Protocol layer module as manufacturers.

Software Frame Sync layer will have three goals:

- Proceed to the frame sync on throughput data

- Reformat extracted data (unpacked, packed, throughput) for suitable comparison in BSS

- Compute the quality factors based on the minor frame data (parity, synchronization word error bits, etc.).

Using the Software Frame Sync will sort the data in a common format to make data easy to compare for BSS.

Parameters input of this frame sync are

- The PCM format (FS size, MF size, mf size, word size etc…)

- The Word mapping (Word container size: 16 or 32 bits, MSB or LSB…)

- PCM payload buffer

- Timestamp of the buffer

- Format of the quality factors provided (DQM, EB/N0, Parity, Synchronization word bit errors, etc.)

From these inputs, the soft frame sync will provide a timestamped and qualified standardized output.

The interest of the Software Frame Sync is that any kind of data format can be managed: Throughput data, PCM words from 3 bits to 32 bits, swapped or not, MSB or LSB…

Finally, the output is always in the same format: MSB PCM throughput data.
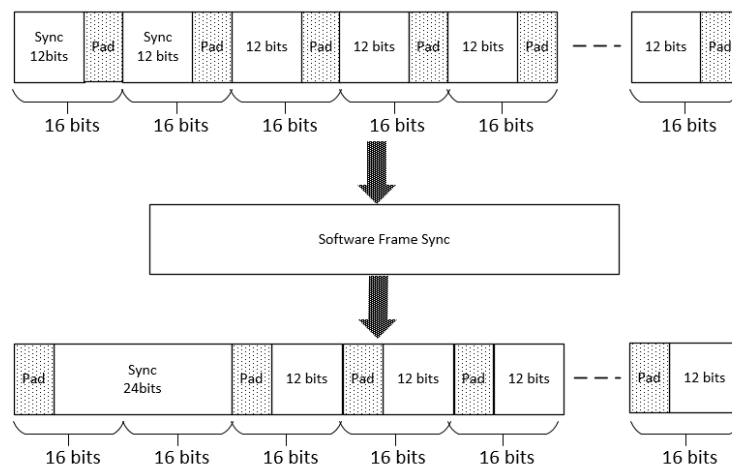
Figure 14: Reformatting data example

Figure above shows the reformatting of a 12 bits PCM word packed stream. The output holds the same data but in a different way that will be comparable with the other sources.

Once this part is complete, the BSS will always work on standardized data streams.

BSS can then be applied on all the sources received, as they are now comparable. Additional Quality factors extracted from the proprietary layer can help to the decision following the criteria and the weighting associated to these factors. Each minor frame is timestamped and with SFID information, it is possible to time align the different sources and proceed to BSS algorithm.

Figure below shows an example of a PCM stream comparison from three heterogeneous sources.
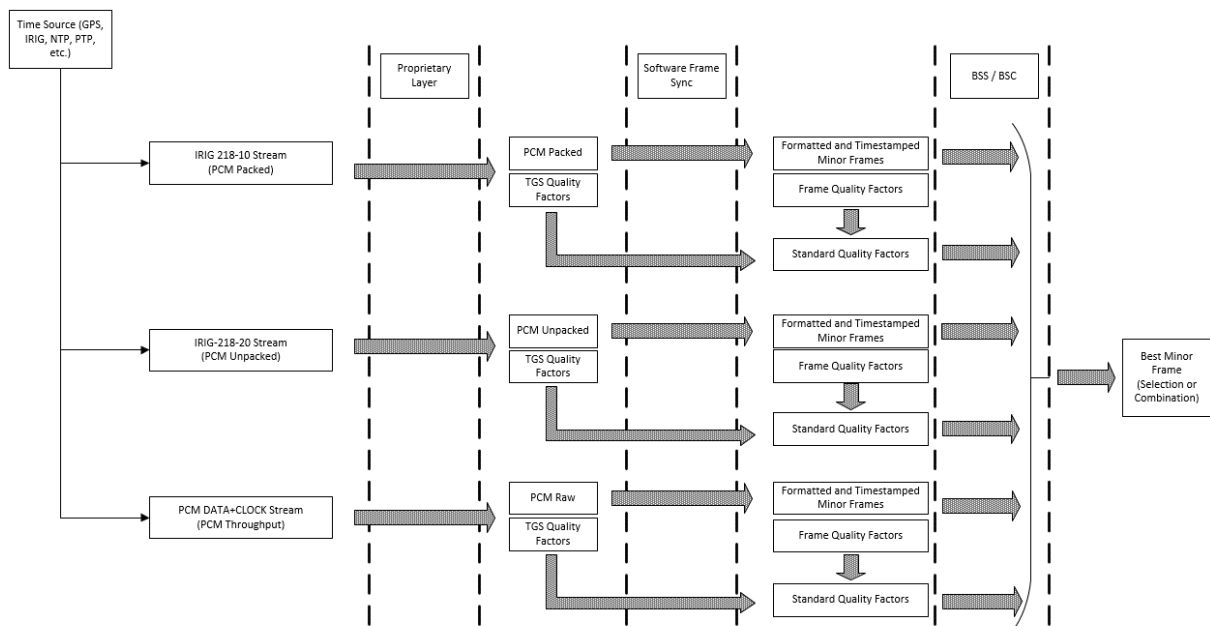
Figure 15: PCM Stream comparison/combination

In this example, the three sources are carrying the same PCM stream but in heterogeneous format. Proprietary layer extracts the PCM stream information and the TGS quality factors. Software Frame Sync formats the PCM streams in comparable data frames, extract frame quality factors so it is possible to compare it in the BSS. All the minor frames are timestamped and can be compared bit to bit with the standardized weighted quality factors.

Packet Synchronization

The main challenge we had to deal with was the packet synchronization: Ethernet protocols are buffered and non-deterministic. The propagation time of the telemetry signals is dependent on the varying distance between the mobile and the antenna, thus causing varying delays. If there is no information available on the PCM data to timestamp and evaluate this delay, BSS/BSC architecture could produce invalid results if the delay is greater than a long cycle duration.
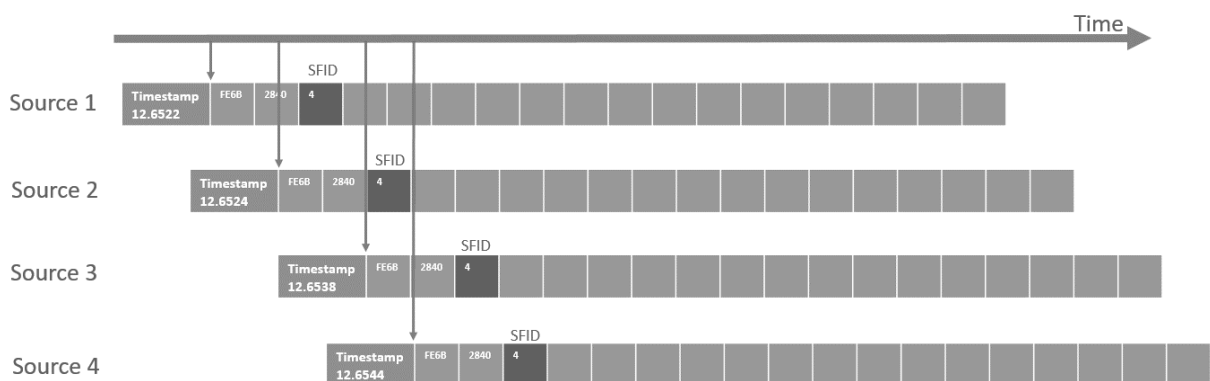
Figure 16: Example of receiving time desynchronization

Most of telemetry systems timestamp the data at receiving time, PCM frames must include some identification parameters (SFID, counters, etc.) to be able to synchronize and compare the short cycles. Figure below presents an example of synchronization using timestamps and SFID values.
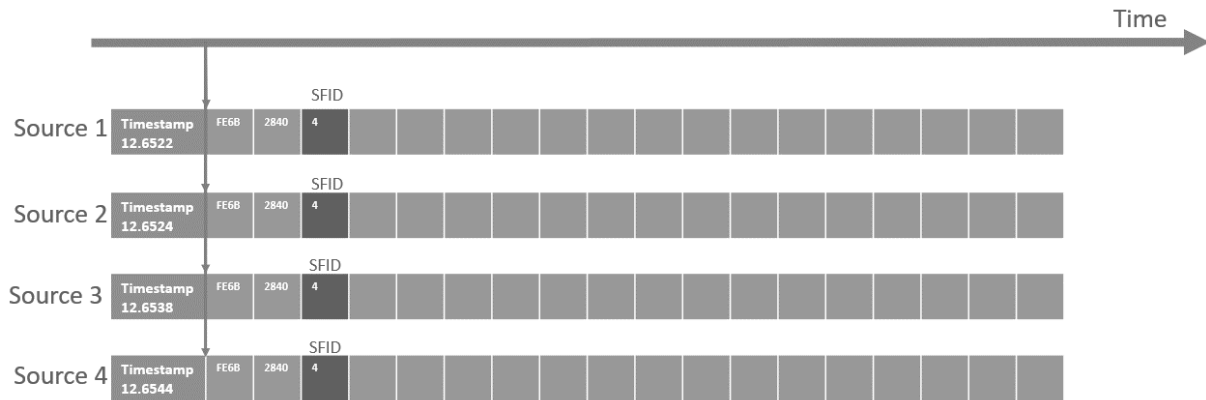
Figure 17: Example of synchronization algorithm using timestamps and SFID values

Practical case

To validate this model, several architectures have been tested with good results.

The Three layers BSS have been implemented in MAGALI Telemetry Software used as the telemetry data software.

The first practical case is based on Telemetry receivers sending proprietary Ethernet Output in throughput mode which are connected to MAGALI decom station. Each Telemetry Receiver sends three Ethernet Outputs (LHCP / RHCP / Combined) at 30 Mbps to the MAGALI telemetry software.

The decom station had to produce a QLM (Quick Look Message) stream. This is an Ethernet message (50Hz) composed of selected Telemetry parameters (filtered from ICD).
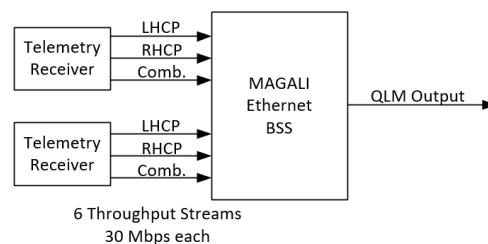
Figure 18: 30Mbps streams / QLM Output

The Telemetry Stream is a 30Mbps PCM-FM with 50 µs minor frames.

The tests have been done on Core i7-6700@3.4GHz CPU. CPU Load was around 20% for the six streams with a 130 Mbps Ethernet Load. Delay measurement was around 60ms with a deviation of 30 ms.

Tests have shown that this solution ensures signal continuity over long distances by defining the best time to switch from one source to another automatically.

The second practical case is based on three heterogeneous sources as shown in figure below. The system had to generate the best Ch10 unpacked telemetry stream. The sources came from several installation on the range.

- A new receiver able to provide TM throughput signals in Ch10 format

- A MAGALI TGS front end which provides Frame synchronized data blocks

- A PCM source which has been upgraded few years ago with a PCM to Ethernet Module to broadcast telemetry over Ethernet. This source was a redundant source in case of failure as there is no other quality factor than Frame sync Lock.
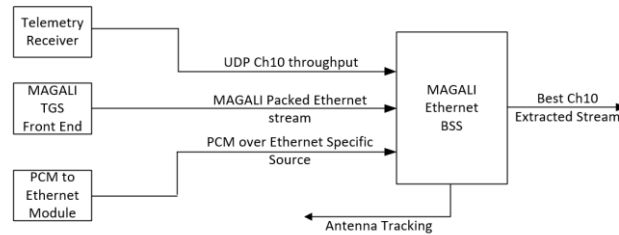
Figure 19: Heterogeneous sources

The bit rate in this case does not exceed 10Mbits but the interesting aspect is the heterogeneous sources as PCM inputs. Even if the complete installation has not been upgraded with the same hardware interfaces, the Ethernet BSS is able to manage the different kind of inputs. Therefore, end customer did not have to buy other hardware to manage the different inputs. Time delay has not been measured for this project, nevertheless the system was performant and delay was not exceeding 200ms to provide the Ch10 stream in real time.

The best source selection algorithm used in this case (with 3 systems) allows to correct the frames and obtain 42% of corrected bits compared to a single-stream system (20 % in term of corrected frames).

The third practical case is based on four heterogeneous sources as shown in figure below. The sources came from: two receivers sending TMoIP (IRIG 218-10) streams and a MAGALI TGS front end which provides 2 frames of synchronized data blocks.
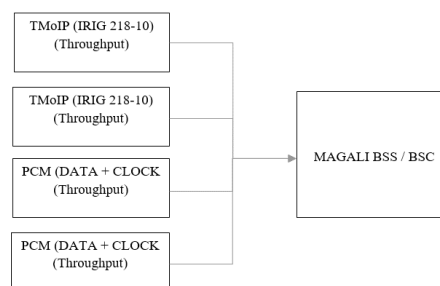
Figure 20: TMoIP and PCM DATA+CLOCK

The bit rate in this case is 20Mbits but the interesting aspect is the heterogeneous sources as PCM inputs and the heterogeneous format of quality factors (DQM for TMoIP, and frame quality factors for PCM DATA+CLOCK). The MAGALI system had to compute standardized quality values to permit the BSS/BSC module to perform selection and/or combination.

The combination algorithm used in this case (with 4 systems) allows to correct the frames and obtain 50% of corrected bits compared to a single-stream system (30 % in term of corrected frames).

Conclusion

The BSS/BSC architecture is a good answer to:

- The prolific Ethernet PCM format that all the manufacturers can propose.

- The management of heterogeneous sources in case of a partial update of an existing installation.

This solution is a low cost investment as it is only software based. Moreover, the three layers architecture provides a flexible model in case of new format management, only the first layer has to be upgraded with a new plugin in order to manage the new format. Software Frame Sync & BSS Layer remain the same once the information are provided.

The BSS/BSC algorithms allows to correct the frames and obtain from 42% to 50 % of corrected bits compared to a single-stream system (20 % to 30 % in term of corrected frames).

The constraint of this architecture is that the packets are timestamped using different formats and packets can be sent with a variable delay (as an example, Ethernet is a non-deterministic protocol). The BSS/BSC architecture must ensure that:

- The packets must be timestamped using a standard format.

- The packet synchronization: the BSS must compare (and then combine or select) full synchronized packets using timestamps and counters extracted from the frames (SFID for example).

Another constraint of this architecture is in case of encrypted data, throughput data cannot be compared with synchronized data. It is necessary to decrypt all the data before using this method or to compare only encrypted throughput data based on a random sync pattern.

If all the streams are encrypted using the same algorithm, another solution would be to transmit these streams to the BSS / BSC module. In this case, a bit correlation could be applied in order to synchronize the frames. This mechanism, which consumes a lot of processor time, could be transferred to external processors (GPU for example)

References:

[1]   ITC_2004_04-09-01: Telemetry Best Source Selection at White Sands Missile Range

[2]   ITC_2004_04-23-01: Performance evaluation of Ethernet LAN based distributed telemetry data network and its extension using router and bridge

[3]   ITC_2012_12-05-02: IRIG 106 Chapter 10 vs. iNET Packetization: Data Storage and Retrieval

[4]   ITC_2012_12-24-01: PCM to Ethernet: A Hybrid System Used to Certify the Next Generation of Data Transfer Technology

[5]   ITC_2018_18-09-03: BSS with Heterogeneous Ethernet Sources

---

[i] In memory of Julien MALIVERNEY (1978-2019), rest in peace my friend.

SOQPSK-TG STC : a new decoding scheme for a higher bit rate and a better link budget

Alain Thomas, Julien Renaudin, Gregory Blanc, Rami Othman
[1] *Safran Data Systems 5 Avenue des Andes, 91940 Les Ulis France,*
alain-dominique.thomas@safrangroup.com

Abstract

To protect telemetry from antenna masking during flight tests, two antennae must be used on-board. To avoid phase opposition of the two transmitted signals at reception, frequency diversity has been used for decades. However with spectrum auctions this technique is no longer compatible with the requirement of higher bit rate. To overcome this issue, well-known modulation format like COFDM may be used at the expense of power efficiency, with few constraints on antennae position. IRIG-106 standard proposed also SOQPSK-TG STC which is efficient both in power and in spectrum at the expense of the antennae position. To demodulate this format a dual-trellis solution has been proposed based on XTCQM representation [1] with results very close to SISO as far as distance between antennae does not exceed half bit delay. Therefore this decoding scheme is a hard decision scheme and consequently is less power efficient when a FEC such as LDPC is used. In this paper a new decoding scheme will be presented with soft decision outputs to increase power efficiency, and with a better tolerance to differential delay.

Key words: Flight test, spatial diversity, Space time coding, telemetry.

Introduction

Transmitting real-time telemetry with low latency on aeronautical channel is very demanding. The data link has to be error free on at least four different channel models from high Doppler spread take-off fading channel to low SNR far flight channel [1]. Spectrum auctions constrain even more the data link pushing it to C band and requiring spectral efficiency to transmit up to 20Mbps to meet market demand. A first waveform based on COFDM meeting all these requirements has been proposed [1][2] to transmit telemetry from commercial aircraft. However for small aircraft or for carriers with little power available, the amplifier back-off required by COFDM could be incompatible with mission requirements. For such missions SOQPSK-TG combined with equalization techniques can meet power and spectral efficiency on channel with moderate frequency selectivity [3]. However under certain aircraft to ground station geometries, antenna masking occurs, leading to link outage. A MISO hard bit scheme Alamouti like has been proposed [4] to mitigate the shadowing and the associated differential delay based on XTCQM decomposition of STC-SOQPSK-TG waveform. In this paper a new low resource decoder with soft output and its performances will be presented.

IRIG-106 STC-SOQPSK-TG

IRIG-106 has standardized constant envelope waveforms with ever increasing spectral efficiency enabling use of power amplifier at saturation

TABLE A-3.	CHARACTERISTICS OF VARIOUS MODULATION METHODS.			
Characteristic	**PCM/FM with single symbol detection**	**PCM/FM with multi-symbol detection**	**FQPSK-B, FQPSK-JR, SOQPSK-TG**	**ARTM CPM**
Occupied Bandwidth	1.16 bit rate	1.16 bit rate	0.78 bit rate	0.56 bit rate
Sensitivity (E_b/N_0 for BEP=1e-5)	11.8-15+ dB	9.5 dB	11.8-12.2 dB	12.5 dB
Synchronization time	100 to 10,000 bits	250 bits	5,000 to 30,000 bits	30,000 to 150,000 bits
Synchronization threshold level (E_b/N_0)	3 to 4 dB	2 dB	4.5 to 5 dB	8.5 dB
Phase noise susceptibility*	2	1	3	4
Co-channel interference susceptibility*	2	1	3	4
* 1=Best, 2=Second Best, 3=Third Best, 4=Worst				

Fig.1 IRIG-106 waveforms comparison

However to prevent data link from on-board antenna masking outage, dual antennae (MISO) transmission is required to ensure error-free telemetry transmission, and SOQPSK-TG features an interesting trade-off between spectral efficiency with an occupied bandwidth of 0.78*bit rate and power efficiency thanks to its 4dB synchronization level and its performance when coupled to an ARJA LDPC FEC [5]. Combining SOQPSK-TG and MISO space diversity is however not straightforward as modu-

lation is not linear and antenna spacing on-board induced a differential delay at reception which is not compatible with Alamouti scheme. In [6] an Alamouti like scheme has been proposed, and a SOQPSK-TG space time decoder based on XTCQM representation given. This scheme has been standardized by IRIG-106 group [7] enabling telemetry transmission with a high quality of service whatever the aircraft to ground station geometry.
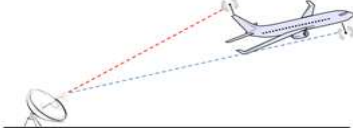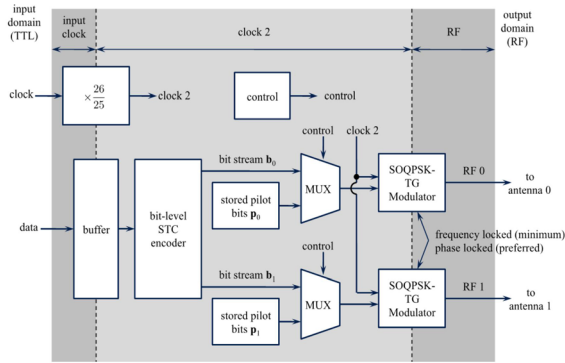

Fig.2a Miso scheme with differential delay

Fig.2b STC on-board modulator

Fig.2c Bit level STC encoder

As can been noticed on fig.3 the tolerated differential delay is strictly limited to one bit, and a noticeable BER degradation occurs at half a bit period. Taken into account plane geometry and constraints on antennae location, a direct consequence of this limitation is a limitation of the telemetry rate which cannot be higher than

$$R\text{max} < \frac{c}{2Dant}$$

(1)

Where Dant is the distance between antennae, considering differential delay due to on-board cables has been perfectly compensated.
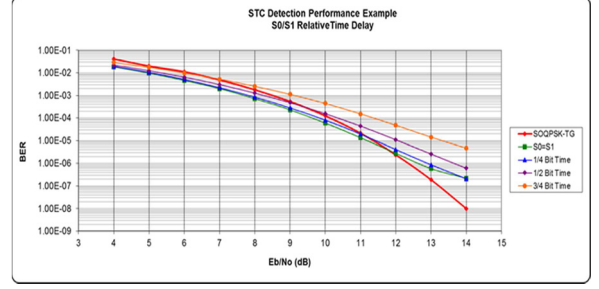
Fig.3 STC detection performance vs differential time delay

Consequently, the ability to decrease degradation related to differential delay could be a way to relax constraints on antennae position or to increase telemetry rate with the same position constraints. Another constraint of the differential delay on the XTCQM LS decoder [4][6] is it requires to implement two trellis, which increases its complexity.

STC PAM LS decoder

In [9] a new PAM decomposition was proposed by Othman, STC signal can then be written as

$$s_p(t) \approx \sum_i \rho_{0,2i}^p w_0(t - 2iT) - \rho_{1,2i+1}^p w_1(t - 2iT - T) + \left(\sum_i \rho_{0,2i+1}^p w_0(t - 2iT - T) - \rho_{0,2i}^p w_1(t - 2iT)\right)$$

(2)

Where :

$p \in \{0,1\}$ is the on-board antenna number

$$\rho_{0,i}^p = \begin{cases} \left(2b_i^{(p)} - 1\right) \text{ i even} \\ j\left(2b_i^{(p)} - 1\right) \text{i odd} \end{cases}$$

$$\rho_{1,i}^p = \begin{cases} -j\left(2b_{i-2}^{(p)} - 1\right)\left(2b_{i-1}^{(p)} - 1\right)\left(2b_i^{(p)} - 1\right) \text{ i even} \\ -\left(2b_{i-2}^{(p)} - 1\right)\left(2b_{i-1}^{(p)} - 1\right)\left(2b_i^{(p)} - 1\right) \text{ i odd} \end{cases}$$

$b_i^{(0)}$ and $b_i^{(1)}$ are ith bits feeding respectively channel 0 and 1 encoded accordingly to fig2.C

$w_0(t)$ and $w_1(t)$ are shaping pulse given in [9] and fig.4
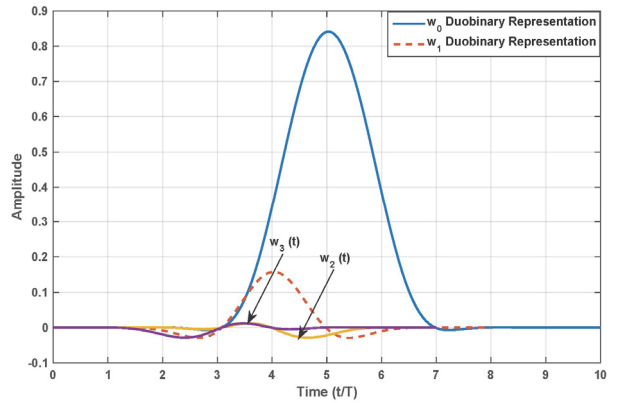
Fig.4 DBD Pulse shape

On ground antenna I the signal received is

$$r_I(t) = \left[h_{0,I} s_0(t - \Delta t_{0,I}) + h_{1,I} s_1(t - \Delta t_{1,I}) \right] e^{j2\pi\Delta f_I t} + z_I(t)$$

(3)

Where

$h_{p,I}$ is the LOS channel complex gain between antenna p and antenna I

$\Delta t_{p,I}$ is propagation delay between antenna p and antenna I and $\Delta \tau_I = \Delta t_{1,I} - \Delta t_{0,I}$

Δf_I frequency offset, $z_I(t)$ additive noise

In Miso scheme I=0, considering perfect carrier synchronization and combining (2) and (3) received signal after ADC could be written

$$r_0(n)$$
$$\approx h_0 \underbrace{\left[\sum_i \rho_{0,i}^0 w_0(nT' - iT) + \sum_i \rho_{1,i}^0 w_1(nT' - iT) \right]}_{s_0(nT')}$$
$$+ h_1 \underbrace{\left[\sum_i \rho_{0,i}^1 w_0(nT' - iT - \Delta\tau) + \sum_i \rho_{1,i}^1 w_1(nT' - iT - \Delta\tau) \right]}_{s_1(nT' - \Delta\tau)}$$
$$+ z(nT')$$

(4)

Where T' is sampling time. As shown on fig.4 shaping pulse lasts more than T, consequently intersymbol interference remains and has to be filtered. Optimal detection filter is discussed in [10], after filtering and sampling of two TX channels samples can be written using (4) accordingly to Fig.5:

$$y(4k) \approx h_0 \sum_{i=-1}^{1} \rho_{0,4k-i}^0 \widetilde{w}_0(iT) + h_0 \rho_{1,4k}^0 \widetilde{w}_1(0)$$
$$+ h_1 \sum_{i=-1}^{1} \rho_{0,4k-i}^1 \widetilde{w}_0(iT - \Delta\varepsilon T)$$
$$+ h_1 \rho_{1,4k}^1 \widetilde{w}_1(-\Delta\varepsilon T) + \tilde{n}(4kT)$$

(5)

$$y_{\Delta\tau}(4k) \approx h_0 \sum_{i=-1}^{1} \rho_{0,4k-i}^0 \widetilde{w}_0(iT + \Delta\varepsilon T)$$
$$+ h_0 \rho_{1,4k}^0 \widetilde{w}_1(\Delta\varepsilon T)$$
$$+ h_1 \sum_{i=-1}^{1} \rho_{0,4k-i}^1 \widetilde{w}_0(iT)$$
$$+ h_1 \rho_{1,4k}^1 \widetilde{w}_1(0) + \tilde{n}(4kT + \Delta\varepsilon T)$$

(6)

Where $\widetilde{w}_i$ is filtered pulse shape, $\tilde{n}$ is z filtered, and $\Delta\varepsilon$ is nearest integer from $\Delta\tau/T$
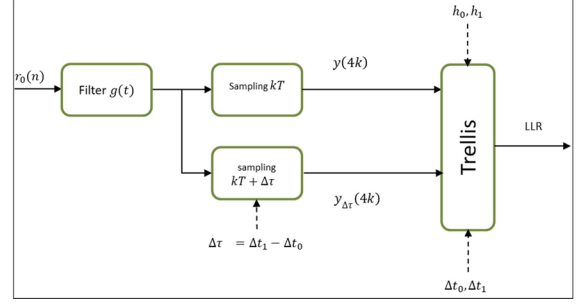
Fig.5 STC SOQPSK-TG decoder

Then the following likelihood function is considered

$$\Lambda(\underline{S}) = \sum_{n=0}^{(N-1)/4} \left(\sum_{m=-1}^{2} \left[\left| B_{m,n}^{(0)} \right|^2 + \left| B_{m,n}^{(\Delta\tau)} \right|^2 \right] \right)$$

(7)

Where

$$B_{m,n}^{(0)} = y(4n + m) - h_0 \left(\sum_{i=-1}^{1} \rho_{0,4n+m-i}^0 \widetilde{w}_0(iT) + \rho_{1,4n+m}^0 \widetilde{w}_1(0) \right) - h_1 \left(\sum_{i=-1}^{1} \rho_{0,4n+m-i}^1 \widetilde{w}_0(iT - \Delta\varepsilon T) + \rho_{1,4n+m}^1 \widetilde{w}_1(-\Delta\varepsilon T) \right)$$

(8)

$$B_{m,n}^{(\Delta\tau)} = y_{\Delta\tau}(4n + m) - h_0 \left(\sum_{i=-1}^{1} \rho_{0,4n+m-i}^0 \widetilde{w}_0(iT + \Delta\varepsilon T) + \rho_{1,4n+m}^0 \widetilde{w}_1(\Delta\varepsilon T) \right) - h_1 \left(\sum_{i=-1}^{1} \rho_{0,4n+m-i}^1 \widetilde{w}_0(iT) + \rho_{1,4n+m}^1 \widetilde{w}_1(0) \right)$$

(9)

The ML estimate is then

$$\hat{\underline{S}} = \underset{S}{\arg\min} \, \Lambda(\underline{S})$$

(10)

Where $S_n = [b_{4n} \quad b_{4n+1} \quad b_{4n+2} \quad b_{4n+3}]$

Thus the more likely underlying LLR sequence can be estimated with SOVA algorithm using a 16 states trellis (Fig.5 & 6)
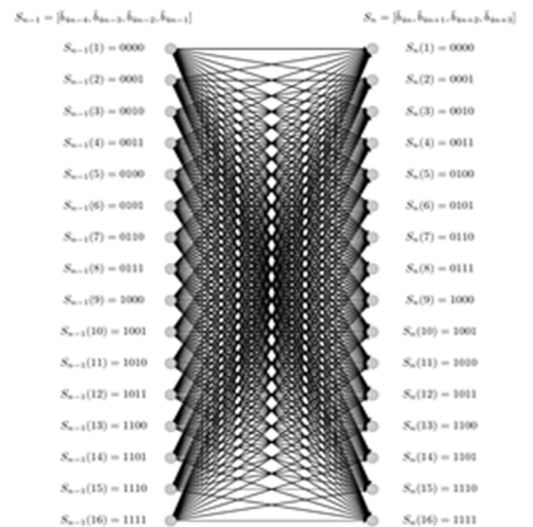

Fig.6 STC-SOQPSK-TG DBD trellis

Instead of trellises derived from XTCQM decomposition [4], it should be noticed that the expressions (8) and (9) remains unchanged whatever the sign of the differential delay between channels, consequently a unique trellis has to be implemented in the decoder instead of two. Moreover considering (8) and (9) and taken into account that each STC symbol is a block of 4 bits, 2048 sub-metric should be computed to decode at bit level, however developing (8) and (9) as a function of the branch bits leads to consider only 320 sub-metric [10] against 480 in [4]. It could also be noticed that on AWGN channel at high SNR regime DBD PAM decoder outperforms XTCQM decoder of more than 1dB at BER 10E-6 for a 0.4 differential delay.
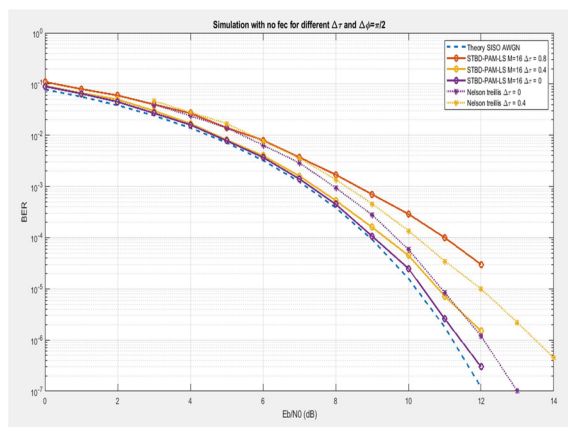
Fig.6 BER proposed decoder vs XTCQM decoder
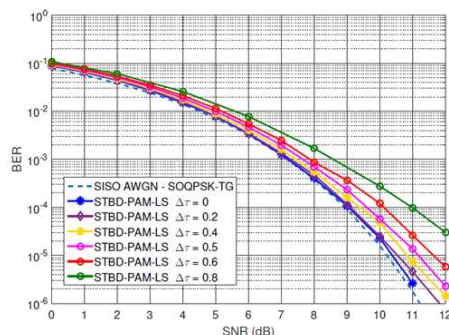
Fig.7 Proposed decoder BER vs differential delay

Instead of improving link budget of 1dB, looking at Fig6. and 7, as proposed decoder has the same BER performance at 10E-6 for a 0.6 differential delay than the XTCQM decoder at 0.4 differential delay, antennae position constraints could be relaxed of 50%, or bit rate could be increased of 50% with the same antennae position constraint .

STC PAM LS soft decision decoder

Soft decisions may be extracted thanks to MAX-log-MAP equivalent SOVA for non binary codes [12]. The classical approach to estimate the underlying bit sequence and its soft information consists then in making a full trellis search up to a certain depth δ and then performing a trace-back loop, which means, for the proposed trellis, to calculate 256 metrics and to store 256 δ reliability values. Instead of extending all states from one symbol time to the next, a suboptimal technique in [11] consists in extending only the M best states, which is called M-algorithm. Othman in [8] proposed to adapt it to its non-binary STC 4 bits, which leads to calculate 16.M branch metrics.
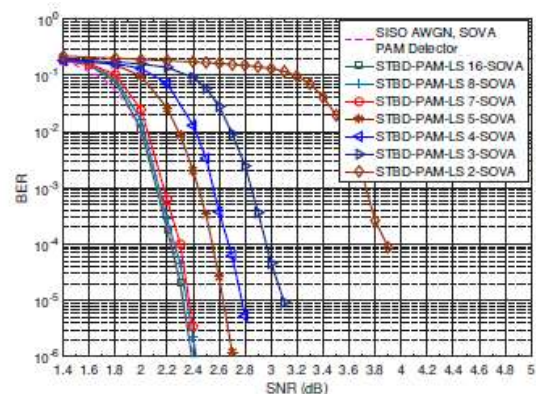
Fig.8 STC M-algorithm AR4JA BER vs M parameter

To estimate optimal M value decoder has then been simulated on AWGN channel in combination with IRIG-106 AR4JA LDPC code K=2/3 L=4096. As shown on Fig.8 for M > 4 BER degradation is below 0.2dB dividing by 3 metrics to be calculated, for M=8 metrics calculation is divided by 2 at no BER cost. Taken into account Fig.9 giving required resources, for M=8 we thus get a significant gain in gates between the XTCQM hard bit decoder and the proposed one with LLR output.

	State of the art STBD-XTCQM-LS	Proposed solution STBD-PAM-LS with M-algorithm	Proposed solution STBD-PAM with M-SOVA	Proposed solution STBD-PAM-ML
Addition	3968	640 + 128M	644 + 176M	576
Multiplication	960	640	640	64
LL²	480	320	320	–
Comparison	240	15M	46M + 40	256
Sorting operation of 16 elements	0	1	1	0
Number of stored values	16	16	17M	64 for hard decoding 128 for soft decoding

Fig.9 Operations required by STC decoder

Both decoders have then been simulated using IRIG-106 AR4JA FEC on Fig. 10 exhibiting a 1.5dB gain. Once again this gain could be used to relax link budget or antenna position constraint or increase bit rate as the proposed decoder enables a higher differential delay margin at constant BER.
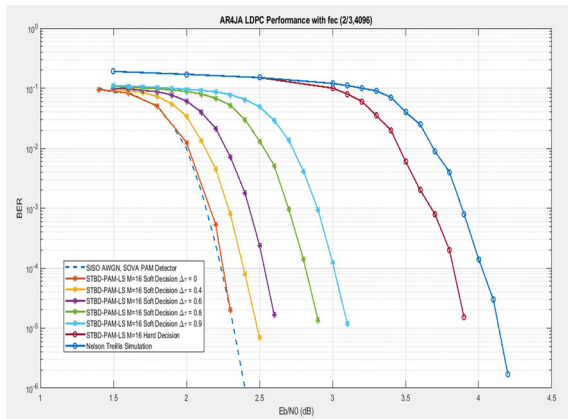
Fig.10 Decoders performance with AR4JA 2/3,4096

Conclusion

Space time coding applied on SOQPSK-TG telemetry link, when constant envelope modulation is mandatory because of on-board power constraint, is a very performant solution to solve the antenna masking problem using a MISO scheme. However the telemetry bit rate is upper-bounded by a maximum bit rate that depends on differential delay (1). XTCQM decoder thus enables, with no significant loss, a differential delay of a quarter of bit, which means for a 20Mbps link a maximum distance of 3.75m between antennae with a perfect calibration of path from modulator outputs to antennae inputs. With proposed PAM-LS decoder it should be possible with the same antennae location to reach 32Mbps at equivalent Eb/N0 and 40Mbps at the cost of an extra 0.2dB. Moreover using IRIG-106 LDPC FEC PAM-LS decoder enables to gain an extra 1.5dB link budget thanks to soft decision output at the price of reduced gate consumption.

References

[1] A. Skrzypczak, A. Thomas, G. Duponchel, "*Paradigms optimization for a C-band COFDM telemetry with high bit efficiency*", International telemetry proceedings, Las Vegas, NV, Oct 2013

[2] L. Falga *"The challenging development and the entry into service of C-Band telemetry at Airbus Test centre",* ETTC proceedings, Nov 2014

[3] A. Cole-Rhodes, H. Umuolo, F.Moazzami *"Real-time CMA equalization for SOQPSK for aeronautical telemetry",* International telemetry proceedings, San Diego, CA, Oct 2014

[4] M. Rice, T. Nelson, J. Palmer, C. Lavin, K. Temple "*Space-Time coding for telemetry : part II – Decoder and system performance*", IEEE Transactions and Electronic Systems, vol. PP, pp. 1-1, Aug 2017

[5] K. Temple "*Telemetry standards that improve link availability",* ETTC proceedings, 2018 DOI 10.5162/ettc2018/2.2

[6] N.T Nelson "*Space time coding with offset modulations",* PhD thesis Brigham Youth University, Provo 2007

[7] Secretariat, Range Commander Council "*Space time coding",* RCC document IRIG106-17 Appendix 2E

[8] R. Othman "*Etude de techniques de reception des modulations de fréquence pour la télémesure aéronautique" ,*PhD thesis tel-02430542 2019

[9] R. Othman, A. Skrzypczak, Y. Louët, "*PAM Decomposition of Ternary CPM with Duobinary Encoding*", IEEE Transactions on Communications, vol. 65, no. 10, pp. 4274-4284, Oct. 2017

[10] Geoghegan, Mark, "*Optimal Linear Detection of SOQPSK,*" in International Telemetering Conference Proceedings, Oct. 2002

[11] K.K.Y Wong "*The soft output M-algorithm and its application", Queen's University, 2006*

[12] J.Tan, GL. Stuber "*A MAP equivalent Sova for non binary Turbo Codes", ICC 2000, Conference Record, vol2 pp602-606, June 2000*

Transitioning to C-band

Paul Cook
Aerospace Instrumentation
Curtiss-Wright
15 Terry Drive, Newtown, PA, 18940, USA
PCook@curtisswright.com

Abstract: With the sell-off of the standard telemetry frequency bands, the transition to C-band in the USA has been slow in adopting the new requirement. Several flight test ranges have moved to C-band but most of the USA has not. As a result, all three frequency bands, L, S, and C, are still being used for flight test despite the government ruling. To address the multiband utilization, several transmitter suppliers have developed a tri-band transmitter to support the frequency allocation on demand approach. This paper reviews the Curtiss Wright's tri-band transmitter development, performance results and the support for space-time processing, forward error correction, and thermal management.

1. Transitioning to C-band

The government auction of airwaves for use in mobile broadband to support the demand for internet access by smart phones and tablets effects the standard telemetry frequency bands, the transition to C-band in the USA has been slow in adopting the new requirement.

With the Government auction and sell-off of the telemetry frequency bands the everyday operation of the flight test community is drastically affected with reduced operational availability to increased frequency crowding. The Government responded with new standards for bandwidth efficient modulation schemes to "pack" more data bandwidth in the remaining telemetry workspace, allowing the private sector to develop hardware in the new C-band allocations for telemetry.

The vendor telemetry community responded first with the spectrum efficient hardware for both air and ground applications while the requirements for C-band applications evolved from initial experiments to actual orders.

The ground segment in the USA has upgraded to be C-band capable across the country. Of interest, the commercial occupation of their new ownership of the L and S-bands (D and E bands) is regional with local test ranges still using L and S-band due to the current availability. Areas of the country where frequency crowding exists, C-band is widely used with equal activity in L-band as well. With the wide variation of frequency bands availability, tri-band transmitters are the focus for flight test operations.

Figure 1: A tri-band transmitter can support S, L and C-bands

2. Developing a Tri-band Transmitter

Curtiss-Wright developed a tri-band transmitter to provide a multiband solution in a single unit to help engineers avoid having to replace units to change band between test flights or when flying from range to range. The transmitter provides lower and upper L(D) band, full S (E) band and C-band in a form factor designed to replace many existing units in the field.

The unit also includes the latest technology as defined in the IRIG-106-19 standard for advanced range telemetry (ARTM) compliant modulation, forward error correction (FEC) and space time compensation (STC) as well as providing independently control dual RF outputs. The unit went through a full series of in house testing as a design verification as well as many tests by the USA Government to both validate the hardware performance and to become familiar with the new features in a real telemetry environment.

Through these tests, there were several modifications and enhancements that include customization as many of the ranges did not agree on the list useful features based on their test objectives. As a result, four variants of the tri-band now exist to address the desire feature list.

During the test campaigns, several undesirable performance issues were observed and corrected using active compensation. The linearity response of the IQ modulator device (where I is the in-phase component, and Q the 90 degree phase difference),

across the three frequency bands did not match the manufacture's performance data. Unfortunately, the selected part had the highest performance device currently available amongst the leading suppliers.

The corrective action was to incorporate active band linearization compensation using a micro controller to activity monitor temperature and apply compensation factors across with wide bandwidth of the three frequency bands. Similarly, the RF power leveling across the three bands was also a challenge and was enhanced with a second micro controller compensation circuitry to level the RF output power across the three bands. Temperature control, being a challenge and further discussed in the following a paragraph, was implemented to control the overall temperature of the transmitter to prevent over-heating and component damage.

2.1 Interoperability Testing

Much of the funding gained by the sell-off of the telemetry bands was spent on a modern upgrade to the range receiver offering with DSP based receiver designs from several manufactures addressing the multiband, the STC and the IRIG 106 chapter 7 standards.

Where they failed as interoperability testing amongst the vendor's products was to assure compatibility and operability. The community did come together to perform an operability test session at the 2019 International Telemetry Conference in Las Vegas to where the transmitter vendors could test against the many vendor's receiver products allowing, them to record the waveforms for future development. Curtiss-Wright was the only transmitter vendor with an IRIG-106-19 compatible transmitter.

As a testament to the accuracy of the IRIG-106 standard, the tri-band transmitter performed well based on demonstrating the functionality of the many vendor's receiver products to demodulate the tri-band waveforms at various data rates and modulation schemes. This test confirms the stated modulation formulas, the LDPC and STC coding schema in the IRIG-106-19 Standard.

2.2 Performance of Space Time Processing (STP)

There was a need for STP to eliminate interference between an upper and lower antenna interference. Time delays between the upper and lower antennas is typically caused by cable length differences. Typical propagation delay in coaxial cables can vary from 1.2 to 1.6 nanoseconds per foot, depending on the material. Ordinarily with a short cable this has

minimal impact. Due to the nature of telemetry equipment being added after the air vehicle is built, long cable(s) are generally used as they are routed through vehicle causing the difference in the propagation delays.

The author has discovered that most implementations measure and match the cable delays to minimize the effect. The tri-band transmitter provides a feature to delay the STC data pattern from none to one clock period increments of 1,024 steps, independently of the two RF outputs to match the delay of each RF path. Tests performed with a STC enabled receiver (STC-ER) has determined that when a timing mismatch of one clock period or less, the STC-ER performs as expected. Larger mismatches than one clock period results in a significant loss of lock and no data. The vendor of the STC-ER provided a metered representation of the miss-match that provides the user with a visual indication of the balanced time delay with the air system to make the adjustment of the tri-band transmitter delay feature simple and repeatable.

2.3 Forward Error Correction (FEC)

The author has been a telemetry supplier of RF hardware for close to 40 years and has used FEC in several forms (Convolutional & Reed-Solomon primarily) over this long career. Today, with the adoption of LDPC forward error correction, IRIG-106-19 standardizes six varying coding schemes for improved link margin. The improved noise performance is not for free as it increases the bandwidth with the FEC data overhead.

The expansion factor (EF) as described in the IRIG-106-19 standard, states regardless of the block length the EF for ½ rate is 33/16, 2/3 rate is 25/16, and the 4/5 rate to be 21/16 to include the FEC as well as the alternate synchronization method (ASM). The six coding schemes, and their information block length (IBL), are as follows:

- 1 = Code Rate 1/2, IBL 1024
- 2 = Code Rate 1/2, IBL 4096
- 3 = Code Rate 2/3, IBL 1024
- 4 = Code Rate 2/3, IBL 4096
- 5 = Code Rate 4/5, IBL 1024
- 6 = Code Rate 4/5, IBL 4096

As an example of the EF for a 4 Mbps NRZ-L stream with LDPC 4 (2/3 rate; block 4096) would result in a data rate of 6.25 Mbps. The same 4 Mbps NRZ-L stream when encoder with LDPC 5 would result in a rate of 5.25 Mbps, and at the LDPC 1 code rate, the 4 Mbps would result to be 8.25 Mbps. The equation is as follows, where R is rate and F is factor:

$$R_{Transmitter} = R_{NRZ-L} \times F_{code\ rate\ expansion}$$

Experimenting in a lab environment with this FEC determined that the performance of the encoding is well behaved. The algorithm performs as described in the IRIG standard. BER performance without and with FEC can be seen in Figure 2 and Figure 3.
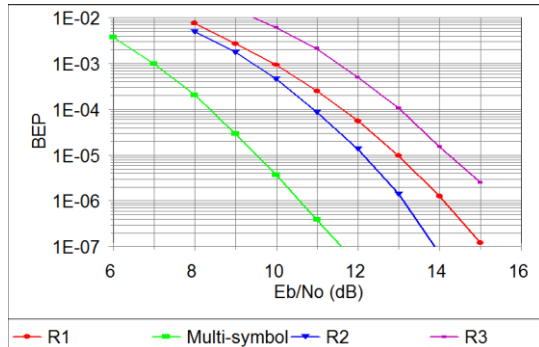

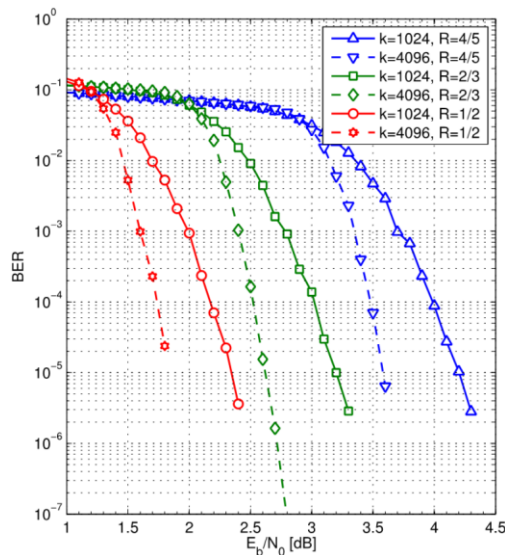Figure 2: BER performance with no FEC

Figure 3: BER performance with FEC

Be advised that the operational region for repeatable performance narrows as the coding gain increased to a point where using the maximum gain with the ½ coding rate, the operation region is somewhat limited down to several dB.

The use of FEC is not useful for all applications and special consideration should be made before deciding to use this tool to obtain your telemetry data. From the tri-band transmitter perspective, switching the LDPC on and off is quite easy by sending a transmitter command of "FE1" to "FE6" to enable your desired code rate or a "FE0" to disable it.

Depending on your ground state receiver of choice the change over from no FEC to FEC being enabled may take several seconds to reconfigure. Planning of when to change over in flight will take scheduling to avoid longer loss of data. Certainly though, FEC works and has been the method of improving link margins for many years.

2.4 Thermal Management

The "drop in replacement" approach for the tri-band transmitter resulted in thermal management issues with the initial prototype of the new transmitter.

The primary heat sink path for transmitters is the bottom-mounting surface. With the thermal dissipation equally distributed, the modules that are farther away from the bottom-mounting surface have a higher temperature rise over the lower modules. Temperature rise of 20°C was measured between the bottom baseplate and the internal temperature of the top module.

2.5 Mitigation of the thermal rise

Decreased the thermal resistance between the module sections with over lapping mechanical joints combined with increased wall thickness on the center (power supply mode) reduced the thermal rise from the 20°C to 15°C. The increased the RF efficiency with optimized RF tuning through the RF chain gained a reduction in current draw and a temperature rise decrease of an addition 3°C.

Thermal protection was implemented with a micro-controller in the transmitter that monitors keys areas in the transmitter to control the dissipation by regulating the output RF power. This control, when activated, would throttle back the RF output power when temperature rises above a preset temperature (typically 75°C) to eliminate the risk of exposing the devices in the top module to excessive temperatures.

The thermal protection is enabled by the end user. To reduce the risk of damage, to the user would set the desired temperature threshold and then enable the control through the transmitter communication port. This feature was found to work well and is an easy safe guard against hardware damage when in the field.

3. Summary:

Every development includes new features, problem resolution, manufacturing maturity, and well as surprising results and the tri-band development was no different.

The complete control over the RF output power with independent control of the RF power going to the upper and lower antenna was evidently a useful tool

for lab testing. No longer does the test engineer need to carry the variety of RF attenuators to match the RF levels driving the test equipment to perform BER tests.

A simple commanded adjustment of the RF power allows a smooth RF level adjustment rather than the manual step attenuator. This eases performance testing. The author of this paper historically did not support the 1 dB step adjustment in RF power but after completing this development, the advantage of this feature is obvious and the credit goes to the telemetry community for this advance.

The ease of responding to both the 106-15 programming protocol and the 106-19 command protocol was a surprise. From a transmitter vendor perspective, the transmitter command listing between the various versions of the IRIG standard have evolved to now well over 60 commands. For the user, remember the appropriate protocol for that transmitter causes delays and inappropriate programming. The tri-band acknowledges the previous command structure as well as the latest IRIG standard command structure, which saves time and increased typing accuracy for the test engineer.

Temperature protection to throttle back the RF power to limit the high temperature of the tri-band saves hardware damage. For some applications it is desirable for the transmitter to "die trying" and continue to transmit through high temperature events. This is not true for flight test where there are long preflight and flight times where the temperature limiting function will save the hardware from damage. This is certainly an advance feature and welcome for the flight test community not to have to turn the transmitter off to allow it to cool down.

Thumbwheel and serial port programming is still required for these flight test transmitters and the author found that to be easy to use and very few difficulties in communicating to the tri-band transmitter. The take away here is the simpler the better and it all works. The friend of this interface is the "?" command, where the tri-band prints out the many command structures to guide the test engineer on the proper command structure.

Cable delay matching between the upper and lower antennas providing delay matching between the cables lengths is now quite easy in using the STC balance displays on some of the vendor's receivers. The delay matching to within 1-bit time is require for STC demodulation performance as the interoperability testing results provided.

The tri-band transmitter requirements will continue to evolve as user gain experience in using this modern transmitter tool to assure the data is received successfully in the current and future frequency bands.

4. References

Note 1: Range Commanders Council IRIG-106-19 Telemetry Standard, Chapter 2 and Appendix (performance data provide from the standard for clarity)

Protecting satellite downlink Tracking and Telemetry channels by IMT-2020 5G Base Stations Protection Belts surrounding Satellite TT&C Earth Stations operating in the 40 GHz Band

R. Pinto Prata[12], R. T. de Sousa Jr[2], A. Linhares de Souza Filho[1], E. Barbosa de Souza[1],
Calil Queiroz[3], T. Aurélio Bakaus[1]
[1] National Telecommunication Agency (ANATEL), SAUS, Qd 6, Brasília, DF, Brazil
[2]University of Brasilia (UnB), Asa Norte, Brasília DF, Brazil
[3]Access Partnership, 9th Floor, Southside 105 Victoria Street London SW1E 6QT United Kingdom
rafaelprata@anatel.gov.br

Abstract:

Telemetry, tracking & command (TT&C) earth stations (ES) constitute a sensitive part of the communication critical infrastructure in satellite networks, because harmful interference to TT&C channel can cause the total loss of communication with the space station. The ITU WRC 2019[1] identified frequency bands to be shared by 5G (IMT-2020) and satellite systems, including the 40 GHz band, which requires interference mitigation techniques.

This work presents and discusses simulation results to evaluate the effectiveness of characterizing a zone *(Protection Belts)* where *5G base stations* (BSs) could operate under more restrictive configurations to protect from harmful interferences the satellite downlink *TT&C channels* operating at the 40 GHz band. The "protection belt" concept would reduce the need for a wide "silent zone" surrounding the TT&C earth station (ES). The simulations were performed with an open source tool [1][23] that complies with Recommendation ITU-R M.2101 [2]. For more realistic scenarios, our simulation shows the positive effects of reducing the aggregate I/N (interference-to-noise ratio) generated by 5G BSs as a result of more restrictive parameters for the 5G operation deployed nearby the TT&C earth station, avoiding co-channel harmful interference.

Key words: Radio frequency, Spectrum sharing, IMT-2020, Satellite TT&C, Cybersecurity.

---

[1] Sharm el Sheik, 2019, https://news.itu.int/wrc-19-agrees-to-identify-new-frequency-bands-for-5g/

[2] https://github.com/SIMULATOR-WG/SHARC

[3] https://ieeexplore.ieee.org/document/8121171/

Background

The main driver for the Economy 4.0 will be the 5G mobile networks, also named by the International Telecommunication Union (ITU) as International Mobile Telecommunications 2020 (IMT-2020) radio interface standard. The IMT-2020 technical requirements place the need for enormous amounts of spectrum resources, meaning a high challenge for Administrations, and demanding new approaches on the spectrum management study field.

The ITU World Radiocommunications Conference 2019 (WRC-19) reviewed the Radio Regulations (RR), an international regulatory spectrum treaty, and identified 17.25 GHz of spectrum in millimeter waves for usage by IMT. The newly identified bands are 24.25-27.5 GHz, 37-43.5 GHz, 45.5-47 GHz, 47.2-48.2 and 66-71 GHz [3]. In the scope of the present work it is highlighted a new footnote (5.550B) to the band 37-43 GHz, in the RR international table of frequency allocations, as highlighted bellow (emphasis added).

"5.550B The frequency band 37-43.5 GHz, or portions thereof, is identified for use by administrations wishing to implement the terrestrial component of International Mobile Telecommunications (IMT). This identification does not preclude the use of this frequency band by any application of the services to which it is allocated and does not establish priority in the Radio Regulations. Because of the potential deployment of FSS earth stations within the frequency range 37.5-42.5 GHz and high-density applications in the fixed-satellite service in the bands 39.5-40 GHz in Region 1, 40-40.5 GHz in all Regions and 40.5-42 GHz in Region 2 (see No. 5.516B), administrations should further take into account potential constraints to IMT in these bands, as appropriate. Resolution 243 (WRC-19) applies. …"

The present paper addresses some of the potential constraint solutions to IMT on the shared use of 37-43.5 GHz frequency band, especially considering its possible use for downlink of Tracking and Telemetry data for TT&C functions in satellite master control centers.

TT&C in the satellite world

In the satellite system, the Tracking, Telemetry and Command (TT&C) subsystem is responsible for the monitoring and controlling functions of the space station during its operational lifecycle. In summary, the tracking (or ranging) function collects information about the angles, distance and velocity to estimate satellite position in its orbit. The telemetry function collects operational status data of the satellite allowing the satellite master control center to continuously monitor its condition. The telecommand subsystem receives all the information, including those

received from tracking and telemetry subsystems, and send commands to the space station. The tracking and telemetry are downlink based subsystems and telecommand subsystem is uplink based, as shown in Fig.1 from [4].
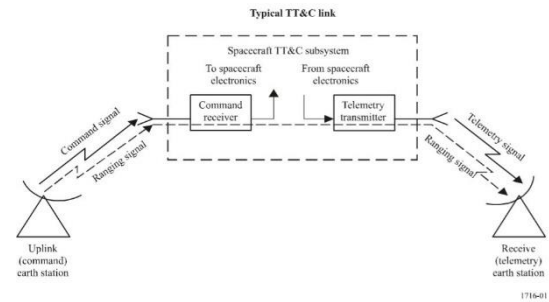
Fig.1. Typical TT&C link (ITU-R S.1716)

In a regulatory level, article 1 of ITU Radio Regulations [5] defines:

"…1.133 space telemetry: The use of telemetry for the transmission from a space station of results of measurements made in a spacecraft, including those relating to the functioning of the spacecraft. …

1.135 space telecommand: The use of radiocommunication for the transmission of signals to a space station to initiate, modify or terminate functions of equipment on an associated space object, including the space station.

1.136 space tracking: Determination of the orbit, velocity or instantaneous position of an object in space by means of radiodetermination, excluding primary radar, for the purpose of following the movement of the object. …"

The TT&C systems are different subsystems jointly working, integrated, so the perfect functioning of satellite command and control processes rely on their total availability. Their earth stations are considered a crucial part of satellite communication critical infrastructures. Harmful interferences to tracking and telemetry downlink channels can cause the total loss of contact with the space station, affecting services rendered to millions of users. In this regard, it is clear the high criticity of such communication systems and the need to ensure their protection from radio interferences.

The increasing relevance of studies about interference prevention on TT&C ES relies on the fact that there are around 200 geostationary satellite network filings under coordination process in ITU which have specific TT&C payloads in Q/V bands. Their orbital positions in the geostationary arc are distributed around all the globe, as shown in Fig.2. These filings reflect satellite projects to be launched and be brought into use in less than 7 years from now.
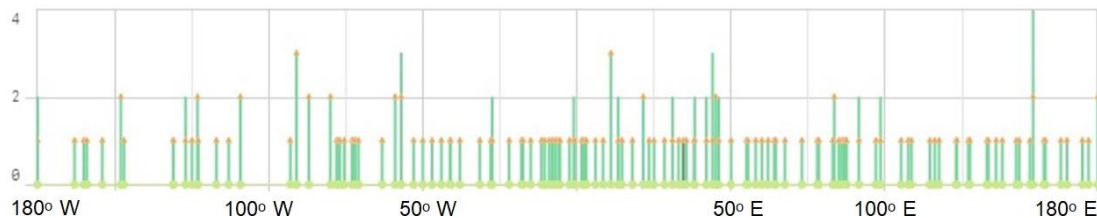
Fig.2. Satellite Network Filings for Q/V TT&C payload per orbital position (Source: the authors)

Previous cases of frequency sharing between TT&C and terrestrial systems

It is not a new debate and references can be found dealing with approaches for spectrum sharing between different TT&C systems and other radio communication system in several frequency bands. TT&C channels are used by highly critical applications, in that sense new approaches should be explored for preserving the security of such sensitive systems while a more efficient usage of spectrum is also promoted. In this context, the Aeronautical Mobile Telemetry (AMT) topic was addressed in the European Test and Telemetry Conference - ETTC 2018 [6], where the spectrum environmental congestion challenges led to new ideas on better tools and methodologies to accommodate the telemetry operations in this new densely occupied spectrum scenario with IMT.

Although it is well understood that the TT&C industry must also evolve by promoting researches into new technologies and concepts to operate in a shared spectral environment, there are also opportunities to study ways to protect the TT&C operations by the adoption of technical variations in the co-shared system, which means the IMT-2020 in this case. In this scenario, taking the perspectives from directional beams domains, advancements in smart phased array antenna field, especially in terms of beamforming techniques and sidelobe cancelling, can help the co-channel interference over TT&C stations to be removed. The concept herein discussed is clearly one of the new tools, as pointed out in [6], to be developed to establish the necessary RF "fortress" for protecting TT&C critical applications in a congested spectrum environment.

In 2018, the national regulatory authority of Hong Kong issued a decision [7] on the change of primary allocation of 3.4 to 3.7 GHz frequency band from Fixed Satellite Service to Mobile Service. Given the fact that existing TT&C earth stations of satellites in Hong Kong´s territory operate in this band and noting that they are important for the operation of licensed satellites currently in orbit, the decision also issued that TT&C will be protected from radio interference of public mobile services. In this regard, an exception was created, allowing TT&C earth stations as having a sort of "primary status" in two restriction zones (in Tai Po and Stanley in Fig. 3) where these TT&C Stations are located. In those regions,

geographical polygons are delineated to constrain the deployment of mobile base stations (BSs) of public mobile services operating in the 3.4–3.6 GHz band.

As stated in the decision about the "restriction zones", the authority stated that the necessary spatial separations between TT&C Stations and mobile BSs in different directions are devised to prevent desensitization of satellite receivers caused by in-band signals of the public mobile services in the 3.4–3.6 GHz band, taking into account the actual terrain, clutters, buildings in the surrounding areas and deployment of BSs over the years, among others.

It can be understood that this specific condition of confining the TT&C in only two sites was necessary due to C band propagation characteristics, and because it will be mostly used for 5G macro cells, making restriction zones extremely extensive as seen in Fig. 3 [7].
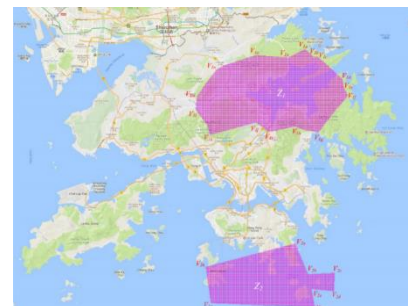
Fig.3. Restriction zones for protection of C Band TT&C earth stations in Hong Kong [7]

Considerations for sharing Q/V bands between IMT and satellite TT&C ES

The use of Q/V band (40/50 GHz) for TT&C will, by nature, face additional challenges due to propagations characteristics, including high levels of attenuation caused by dry air and water vapor (see Fig. 4) from [8]. On the other hand, satellite TT&C links in Q/V will benefit from high building entry losses (see Fig. 5) getting additional protection from terrestrial emissions.

In this regard, the deployment of concrete made structures surrounding the TT&C earth station may be a way to impose at least 20dB or more protection due to attenuation from terrestrial interference signals received by the antenna sidelobes in 40 GHz band, as it may be seen in Fig. 5 [9].
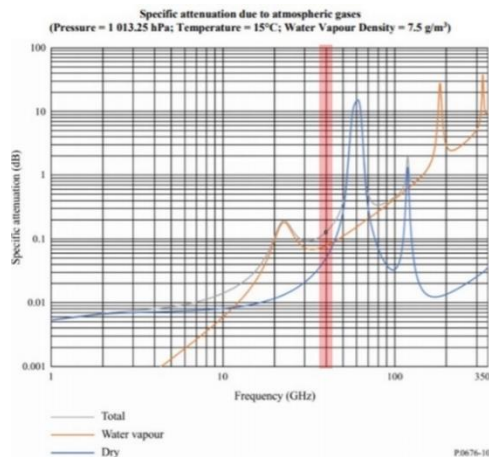
Fig.4. Atmospheric attenuation from 1GHz to 350 GHz in dB/km (Rec. ITU-R P.676-12)

This behavior is particularly important in millimeter waves, and definitely a positive factor for better protection from IMT-2020, mainly in urban, or suburban environments.
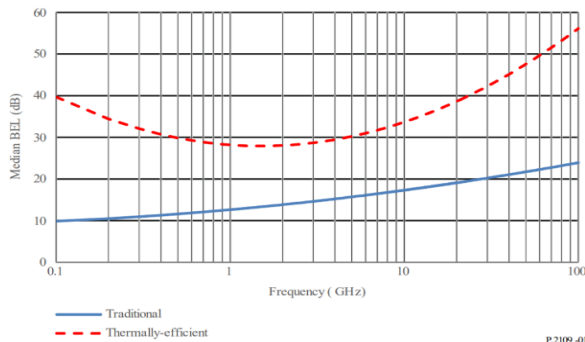

Fig.5. Building Entry Loss (Rec. ITU-R P.2109)

TT&C Protection Belt concept

In a traditional coordination between IMT BS and TT&C in co-channel operation, the solution would be to establish a very wide separation distance surrounding the TT&C ES, acting as a sort of "silent zone" where no IMT operation is allowed in the same frequency band. The proposed approach is to study ways to protect a TT&C ES in a more spectrally efficient way, by adopting restrictions to IMT BS operating in these areas surrounding the TT&C ES, allowing reduced "silent zones".

Fig.6 illustrates a general idea of "protection belt" concept, which benefits from the possibilities of antenna beamforming techniques and flexible adjustments in some RF parameters.

Although this study will focus on the application of only one "protection belt" zone near the TT&C ES, this idea could evolve in the future towards a set of several concentric protection belt layers centered in the TT&C ES position, with different radius and widths, each one providing a different

---

level of protection, depending on the proximity to the TT&C earth station and the QoS required for IMT.
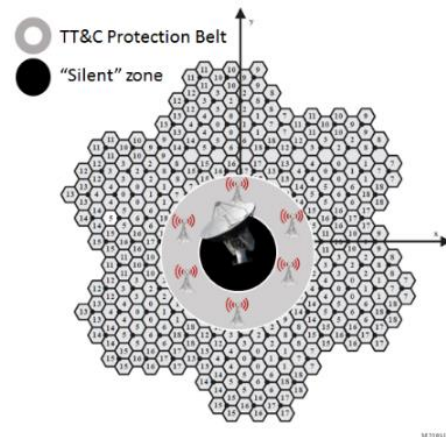
Fig.6. TT&C Protection Belt concept

Spectrum sharing methodology and simulation tool (SHARC)

In order to evaluate the frequency sharing scenario between TT&C ES and IMT, this study made use of an open source collaborative simulation tool developed under leadership of Spectrum, Orbit and Broadcasting Division of the Brazilian National Telecommunication Agency (ANATEL), called SHARC (simulator for SHARing and Compatibility studies) [1]. The tool follows the harmonized methodology defined by ITU [2], which was applied by several sharing and compatibility studies performed in preparation to ITU WRC-19 discussions and regulatory decisions for the identification of new frequency bands for IMT-2020 (5G) such as [10][*].

The ITU-R M.2101 recommendation [2] specifies, among other, how simulations should address the possible deployment scenarios of BSs in the IMT-2020 network (i.e. Indoor, Micro and Micro Suburban and Urban), power control, advanced antenna, antenna height, density distribution, propagation models, network topology, modelling of IMT network for interference calculation, simulation algorithm defining calculation steps, antenna beamforming implementation, determination of aggregate interference and demonstration of results.

Summary of simulation and calculations

The focus on the present study is on co-channel interference emissions from downlink of IMT-2020 BSs into the reception of downlink TT&C ES channel, since it will be responsible for the major impact in the unavailability of satellite TT&C channel caused by interference as illustrated by Fig 7.

---

[*]Reference documents may be provided upon motivated request.
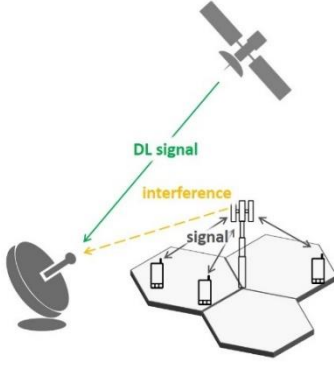
Fig.7. TT&C ES Downlink Interference scenario

The interference scenario to be discussed is the one where the interferer and interfered-with systems operate on the same frequency (co-channel), which is the dominant type of interference. A summary of simulation steps and relevant calculations performed for this specific scenario:

1. Generation of IMT BS locations in hotspot topology

2. Generation of IMT user equipment

3. Apply load BS activity factor probability

4. Connect UEs to BSs and calculate path coupling loss

5. Apply "Power control" / "scheduler"

6. Calculate interference to interfered-with system

7. Calculate interfered-with system performance degradation (I/N) and CDF

Following the same approach of previous studies for this frequency band [11], the simulation considers Recommendation ITU-R P.452-16 [12] model for propagation between IMT and the TT&C earth station and Clutter loss according to Recommendation ITU-R P.2108-0 [13]. As for the channel model within the IMT system, it is considered the (UMi) Urban Micro from 3GPP.

The aggregate external-system interference from IMT-2020 BS towards the TT&C earth station, as indicated in step 6, is performed following [2], according to (1).

$$I_{External} = \sum_j \sum_k I_{External}(TX_{BS_j}^{UE_k}, RX^{victim})$$

(1) Where:

$I_{External}(TX_{BS_j}^{UE_k}, RX^{victim})$ (2) Inter-system interference from base station BS_j (when its k-th UE is served) to the interfered receiver (RX^{victim}).

$I_{External}$: Aggregate external system interference towards the interfered-with system.

One of the most important components for the computation of inter-system interference is the gain from BSs antenna towards the TT&C ES, which is directly affected by IMT BS antenna pattern resulting from the IMT antenna beamforming expressions from [2].

After computing $I_{External}$, and in conjunction with the value of the receiving TT&C ES noise power, finally the main output performance indicator, Interference-to-Noise Ratio (I/N), is obtained by means of a CDF (Cumulative Distribution Function) of I/N over all snapshots samples.

Simulation of interference from IMT base stations into TT&C ES in 40GHz

The sharing study to evaluate the interference from IMT-2020 BSs into a TT&C ES in 40 GHz band is performed by simulating the aggregate interference generated by all the IMT BSs into the victim system (TT&C ES). The aggregate interference is translated into a I/N value, which is compared to protection criteria. When the I/N exceeds the protection criteria, the protection is assumed as not achieved, and harmful interference may occur, otherwise the victim system is not likely to be interfered.

The TT&C earth station and IMT-2020 BS reference parameters used as input data in the simulation tool [1] are presented in Tabs. 1 and 2.

Tab. 1: TT&C Earth Station (ES) reference setup for downlink channel of a geostationary satellite

Parameters – TT&C Earth Station (ES)	
Centre Frequency	40.750 GHz
Channel Bandwidth	1 MHz
Antenna height	6 m
Antenna diameter	6.1 m
Antenna Gain (peak)	66.1 dBi
Antenna pattern	ITU-R S.580 [14]
Antenna elevation	50 degrees
Noise temperature	250 K
Max I/N protection criteria	-12.2 dB

The input parameters of satellite TT&C earth station, of "victim system" are collected from ITU from previous studies and information provided by ITU concerned groups [10] [11] [15].

Regarding the protection criteria, there are different approaches on requirements of Fixed Satellite Services, including long and short-term values. The most relevant [17] is based on percentages of time, probability or location for which the I/N value could be exceeded, and establishes

+8dB (0.02%), -6dB (1%) and -10.5dB (20% or I/N average). Since we are dealing with a critical application (TT&C), and in order to promote a higher protection, a more conservative criteria of -12.2dB I/N was considered as a basis for the assessments of this study. However, it is recognized that this subject deserves further discussions.

For the IMT-2000 system, acting as "interferer" in the simulations, a "reference setup" represents an assumed regular IMT operation, with no additional restrictions to protect TT&C ES from co-channel interference. The set of IMT system parameters and propagation models adopted are based on those recommended for sharing and compatibility studies [11], and further related studies. In order to allow comparability, some values adopted as input parameter are similar to previous studies [10] performed with the same simulation tool [1]. Tab. 2 presents a summary of main parameters used for the reference setup.

Tab. 2: IMT-2020 link reference setup for downlink channel

Parameters – IMT-2020	
Base station (BS) transmitter	
Centre Frequency	40.750 GHz
Duplex method	TDD
Channel bandwidth	200 MHz
Signal bandwidth	>90% of channel
Antenna pattern	ITU-R M.2101 [2]
Antenna array	8×16 elements
Element gain	5 dBi
BS front to back ratio of single element	30 dB
BS horizontal and vertical 3dB beamwidth of single element	65 degrees
Conducted power per antenna element	8 dBm/200 MHz
Deployment (outdoor urban hotspot)	
Hotspots density	30 BSs/km²
Antenna height	6m
Network loading factor	50%
UEs/cell	3
User Equipment characteristics	
Antenna Pattern	ITU-R M.2101 [2]
Antenna array	4×4 elements

Parameters – IMT-2020	
Element gain	5 dBi
Noise figure	12 dB

In order to measure the positive effects in terms of I/N, some of the IMT RF parameters of the reference scenario are isolated, varied and simulations are repeated in order to assess the interference reduction from IMT into TT&C ES from the new scenario. Each simulation performs 5.000 snapshots, and the % CDF I/N output curves quantifies the probability that, for one snapshot, the defined protection criteria is exceeded, and in those cases it is considered that the TT&C ES may be harmfully interfered. The IMT hotspot topology in Fig.7 from [10] is the most likely for the IMT-2020 BSs in this frequency band. The position of IMT BSs and user equipment in the simulation follows random distribution at each simulation snapshot, as recommended by [2].

In modelling of previous studies from [10], the interference generated into a victim system (e.g. Fixed Satellite ES) was assessed by calculating the aggregate interference generated by hotspots deployed inside an IMT cluster with 57 cells concentrated in an area of 1.9km². The hotspots are uniformly distributed, one per cell, and randomly located. Macro cells are not considered because they operate in lower frequencies. The Users Equipment are distributed within the hotspot coverage area, with a Rayleigh distribution with scale parameter $\sigma_d = 32$ m for the distance between UE and BS hotspot, and a normal distribution for the azimuth between them, truncated at the ±60° range, with mean $\mu_a = 0°$ and standard deviation $\sigma_a = 30°$.
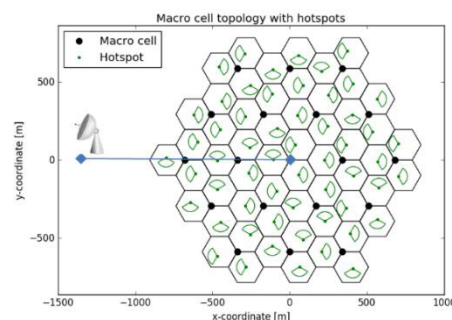
Fig.8. Hotspots in the Macro cell topology and TT&C Earth Station separation distances

In the present study, in order to estimate the aggregate interference generated by hotspots deployed inside a specific area surrounding the TT&C ES, in the format of a "belt", the model was adapted to meet a geometrical modelling of a circular crown.
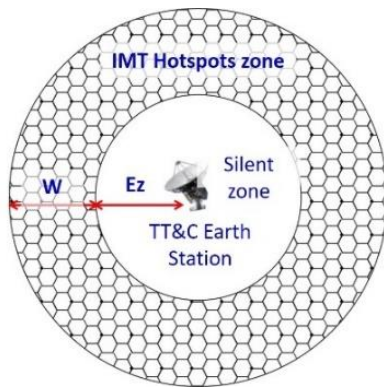
Fig.9. IMT Hotspots zone in the "protection belt"

A segment factor concept from [18][*] is applied in order to adapt the cluster topology and allows estimating the aggregate interference from IMT hotspots in the protection belt (hotspot zone). The separation distance "Ez" also named as "silent zone radius", of 150m between the TT&C ES and the inner border of the protection belt represents the radius of a "silent zone", where no IMT hotspot is deployed in the same frequency of the TT&C ES. A hotspot zone width (w) of 1.7km in the protection belt is defined, as shown in Fig. 9.

Variation of IMT base station parameters

The study was focused only in the IMT BS downlink parameters because its output power is higher than the user terminal, being more harmful for co-channel interference. Also, it is much more complex to customize and promote modifications in billions of user terminals, which involves many factors including increased costs for manufacture and consumer, and a need for industry scale to be effective.

Besides variating the distance between the TT&C ES to the inner border of the protection belt (Ez), the following IMT BS parameters were focused:

- Number of elements of IMT BS antenna array (line and column)

- Conducted power per antenna array element

- Downtilt angle of IMT BS antenna

Separation distance (Ez) between the TT&C ES and the inner border of TT&C ES Protection Belt

The initial simulations, made for reference purposes, adopted input parameters from Tab. 1 and 2, and considered 0m,150m, 300m and 500m separation distances "Ez" between the TT&C ES and the inner border of the protection belt.
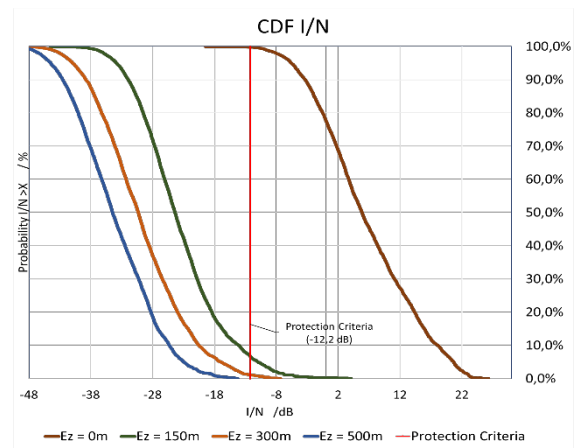
Fig.10. CDF of I/N (0m, 150m, 300m and 500m).

The results, as shown in Fig.10, indicated exceedance of the protection criteria (I/N) on 99.76% of snapshots for 0m, and 6.7% for 150m, considering the aggregate interference from the same total number of IMT BSs. Simulations with 300m and 500m indicated 1% and 0% exceedance of protection criteria, respectively, which confirms that when higher separation distances are applied, the I/N protection criterion is easily achieved. However, this leads to a more inefficient use of spectrum because IMT could not operate in a larger "silent zone'.

Based on that, the present work focused on a smaller separation distance of 150 meters, in which 7% of the snapshots still exceed the I/N protection criteria. This can be considered insufficient for protection of such a critical service as TT&C. In this context, it is assumed that most of the problem is concentrated around this % of cases with higher I/N, and where opportunities for a layer of protection belt can be investigated.

IMT BS antenna array size (number of elements in rows and columns)

The simulations resulted in different curves of % of cases where I/N values are exceeded for each antenna array sizes simulated (8x8, 8x16,16x32, 32x64 and 64x128) as presented in Fig. 11. For better visualization, the plot was focused on the behavior over the 20% of samples with most degraded I/N values. Tab.3 indicates the % of cases where I/N exceeds the protection criteria per BS antenna array size.

[*]Reference documents may be provided upon motivated request.
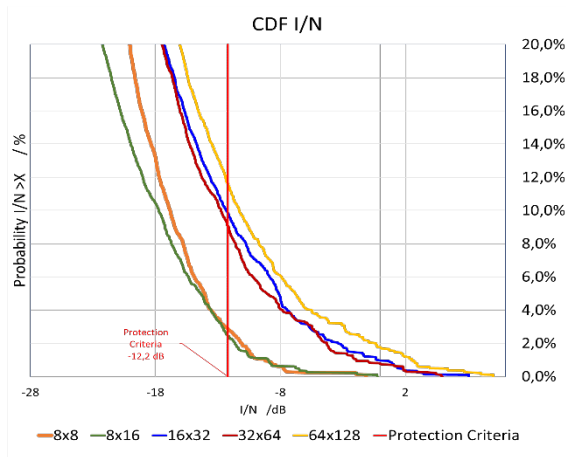
Fig. 11. I/N CDF curves for different antenna array sizes

Tab. 3   % of cases where protection criteria is exceeded for different antenna array sizes

Array Size	% of cases I/N exceeds protection criteria
8x8	2,86%
8x16	6,72%
16x32	9,84%
32x64	9,04%
64x128	11,62%

It is important to have in mind that a higher number of elements in the IMT BS antenna array will increase the array gain, and will also increase the RF power delivered to the array, leading to a higher EIRP (effective isotropic radiated power), which helps increasing the % of I/N exceedance.

In situations of "in line" events, when the main lobe of the IMT BS antenna beamforming points towards the direction of TT&C ES, the interference will be higher than what would happen with a smaller array. However, the different shapes of antenna patterns will also influence on the interference behavior and may compensate (in some measure) the negative effects of EIRP increase. For better clarification on the effects of different array sizes, Figures 12 and 13 present the CDF curves of antenna gain towards the TT&C ES and antenna array patterns for different array sizes, respectively. It is concluded that, once different array configurations are adopted, careful adjustments in other parameters may also be necessary depending on the aimed objective.

Conducted Power per element of IMT BS antenna array

A reduction of 3dB at the conducted power per antenna element resulted in 3,46% reduction in cases where the protection criterion is exceeded

in relation to the reference scenario (8x16 array), as shown in Fig. 14.
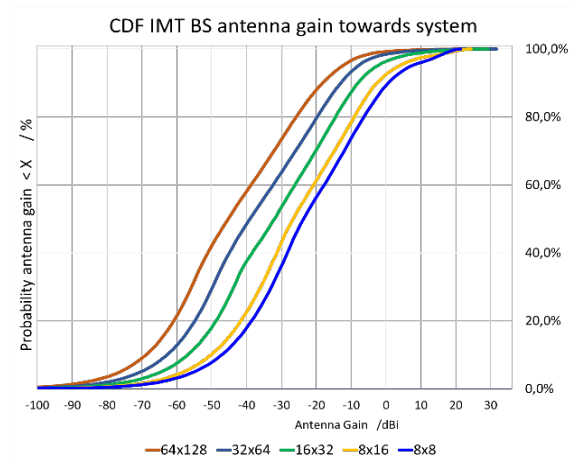
Fig.12.CDF of antenna gain towards TT&C ES
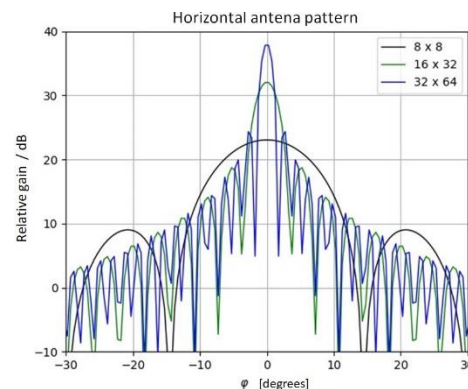
Fig.13. BS antenna patterns for different arrays
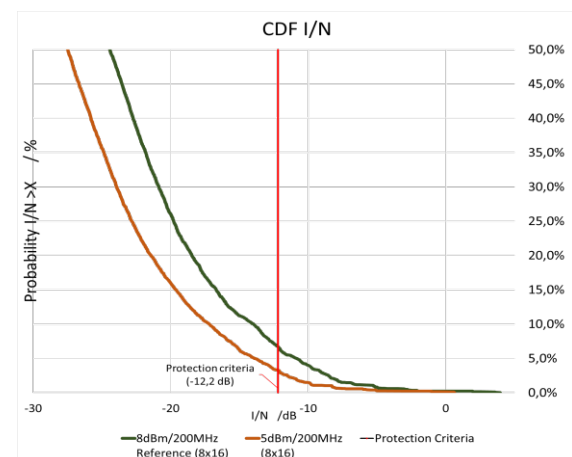
Fig. 14. I/N CDF curve for – 3dB power/element

Tab 4 shows that the different antenna array sizes will have different responses due to different composite antenna pattern, gain characteristics and output EIRP.

Tab. 4:   % reduction in the cases of protection criteria exceedance (-3dB power per element)

Size	Reduction % I/N exceedance
4x8	0,14%

8x16	3,12%
16x32	3,54%
32x64	3,92%

Downtilt angle of IMT BS antenna

The reference scenario was simulated with a downtilt angle of 10 degrees for the hotspot antennas. Further simulation applying 20 degrees to the same reference setup indicated additional reduction of around 1% in cases where I/N is exceeded, as shown in Fig. 15.
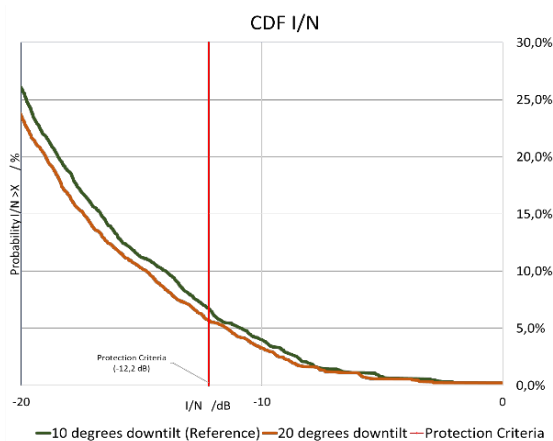
Fig. 15 I/N CDF curves for 10 and 20 degrees IMT BS antenna downtilt angles

From Tab. 5 it can be concluded that the improvement on the I/N curve for 10 degrees higher downtilt angle has also an intimate relation with the IMT BS antenna array sizes, as a result from additional simulation scenarios. In this case, 8x16 array had a better result.

Tab. 5: % reduction in the cases of protection criteria exceedance (+10 degrees in IMT BS downtilt angle)

Size	Reduction % I/N exceedance
8x8	0,56%
8x16	1,16%
16x32	0,78%
32x64	0,36%

A last simulation scenario was tested in which variations of the different parameters were combined. In this scenario the following parameters of the IMT BS antenna array were adjusted. An array size 8x8, 5 dBm/200 MHz in the conducted power per element, and 20 degrees downtilt were adopted. This resulted in only 0,22% of cases where the I/N exceeded the protection criteria.

Conclusions for TT&C ES Protection Belts in Q/V bands

This study concludes that there are possible alternatives for the effectiveness of implementing a geographical layer ("Protection Belts") where the deployment of IMT BSs could operate with a more restrictive configuration in order to prevent co-channel interference to TT&C ES in 40 GHz band.

Simulations performed with higher separation distances between the TT&C ES and inner border of the IMT hotspots zone (500m and 300m) confirmed that the most effective way to prevent interference into a TT&C ES from IMT in co-channel operation is by increasing this distancing. However, it leads to a wider "silent zone". In order to promote a more efficient use of spectrum in 40 GHz, it is possible to reduce this distance by promoting a "fine tuning" on the parameters of IMT BS deployed in an area surrounding the TT&C ("protection belt"), with good results up to a reduced separation distance of 150m.

This can be achieved by combining the adjustments of IMT BSs antenna parameters such as alternate array sizes, reduction on the conducted power per array element and increasing downtilt angle. When adopted, it could additionally prevent up to 10% of cases where I/N exceeds protection criteria. If the adjustments were not adopted, a wider "silent zone" would be needed.
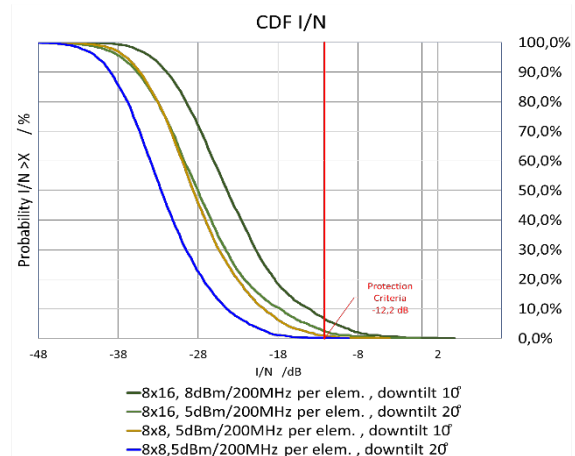
Fig.16. CDF of I/N with % cases exceeding protection criteria

Fig. 16 presents simulation results for more restrictive configurations of IMT that could be applied in a protection belt concept in order to reduce % I/N exceeding the protection criteria.

Further studies

It is important to highlight new opportunities for innovative spectrum coordination approaches in Q/V band. Due to its physical characteristics, and depending on the services involved, lower protection efforts will be necessary to promote its

shared use, than what happens in lower bands. Interesting solutions are expected and, in a more refined spectrum coordination scenario, some of the parameters for IMT operating inner a "Protection Belt" zone could be subject to coordination agreements jointly developed between the 5G and satellite operators. Although it has not been subject to simulations performed in this study, it can be understood from [9] that, due to particularly high Building Entry Losses in the 40GHz band, new opportunities are in place to combine building infrastructure concrete wall as a shielding implementation in the satellite operator side, in order to improve even more the protection of TT&C ES.

In addition, further work should be done for evaluating the results of simulating the impact of wider, or several layers of IMT protection belts, representing larger urban and suburban areas. Studies should also be done on further mitigation techniques for reducing the aggregate impact on TT&C earth stations connecting to NGSO, with other elevation angles, the impact of unwanted emissions from IMT in adjacent bands and new techniques for avoidance angles to reduce in line events between IMT BS and TT&C ES and antenna sidelobe cancelling techniques.

This is a study with academic purposes, based on simulated scenarios. For real deployments, other factors may fluctuate the results, and field test approaches are also recommended.

Acknowledgements

References

[1] An Open Source Implementation of Recommendation ITU-R M.2101 – https://www.itu.int/oth/R0A0600007A/en

[2] ITU-R Recommendation M.2101: Modelling and simulation of IMT networks and systems for use in sharing and compatibility studies – https://www.itu.int/rec/R-REC-M.2101/en

[3] ITU World Radiocommunication Conference 2019 (WRC-19), Sharm el Sheik. Final Acts - https://www.itu.int/dms_pub/itu-r/opb/act/R-ACT-WRC.14-2019-PDF-E.pdf

[4] ITU-R Recommendation S.1716 Performance and availability objectives for fixed-satellite service telemetry, tracking and command systems - https://www.itu.int/dms_pubrec/itu-r/rec/s/R-REC-S.1716-0-200502-I!!PDF-E.pdf

[5] ITU Radio Regulations 2016, Volume 1 - https://www.itu.int/pub/R-REG-RR-2016

[6] Ninja Telemetry AMT Survival in a Congested Spectral Environment – https://www.ama-science.org/proceedings/details/2811

[7] Statement of the Communications Authority, Change in the Allocation of the 3.4 – 3.7 GHz Band from Fixed Satellite Service to Mobile Service, 28 march 2018. https://www.coms-auth.hk/filemanager/statement/en/upload/441/ca_statements20180328_en.pdf

[8] ITU-R Recommendation P.676-12 Attenuation by atmospheric gases and related effects. Aug 2019 - https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.676-12-201908-I!!PDF-E.pdf

[9] ITU-R Recommendation P.2109-0 Prediction of building entry loss. Jun2017 -https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.2109-0-201706-I!!PDF-E.pdf

[10] ITU-R – Brazil - TG5/1 Doc. 5-1/436-E Update on sharing and compatibility study between IMT systems and fixed-satellite service (Space-to-Earth) in the 37-42.5 GHz frequency band –https://www.itu.int/md/R15-TG5.1-C-0436/en

[11] ITU-R – TG 5/1 Doc. 5-1/478-E Annex 1 to Task Group 5/1 Chairman's Report System parameters and propagation models to be used in sharing and compatibility studies - https://www.itu.int/md/R15-TG5.1-C-0478/en

[12] ITU-R Recommendation P.452-16 Prediction procedure for the evaluation of interference between stations on the surface of the Earth at frequencies above about 0.1 GHz -https://www.itu.int/rec/R-REC-P.452/en

[13] ITU-R Recommendation P.2108-0 – Prediction of Clutter Loss - https://www.itu.int/rec/R-REC-P.2108/en

[14] ITU-R Recommendation S.580-6 – Radiation diagrams for use as design objectives for antennas of earth stations operating with geostationary satellites – https://www.itu.int/dms_pubrec/itu-r/rec/s/R-REC-S.580-6-200401-I!!PDF-E.pdf

[15] ITU-R Working Party 4A – TG 5/1 Doc. 5-1/183 - Liaison Statement to Task Group 5/1 FSS/BSS Technical parameters for sharing studies under WRC-19 Agenda item 1.13 -https://www.itu.int/md/R15-TG5.1-C-0183/en

[16] ITU-R Working Party 5D – TG 5/1 [36] attachment 2 - Liaison statement to Task Group 5/1 – Spectrum needs and characteristics for the terrestrial component of IMT in the frequency range between 24.25 GHz and 86 GHz -https://www.itu.int/md/R15-TG5.1-C-0036/en

[17] ITU-R - Working Party 4A – TG 5/1 Doc. 5-1/411 - Reply to Liaison Statement to task group 5/1 -E https://www.itu.int/md/R15-TG5.1-C-0411/en

[18] ITU-R - Brazil - TG5-1 Doc. 5-1/435-EUpdate on the Sharing and compatibility study between RAS in the 42.5-43.5 GHz frequency band and IMT systems operating in the 37-43.5 GHz frequency band https://www.itu.int/md/meetingdoc.asp?lang=en&parent=R15-TG5.1-C-0435

Cellular 4G LTE Aeronautical Mobile Telemetry Flight Test Results

Eric Beck[1], Shobha Erramilli[1], Sarry Habiby[1], William Johnson[1], Achilles Kogiantis[1], Nan Maung[1], Kiran Rege[1], Anthony Triolo[1], Jeffrey Young[1]
[1] Perspecta Labs, 331 Newman Springs Road, Red Bank, New Jersey 07701

Abstract:

Flight test results on a novel approach to Aeronautical Mobile Telemetry (AMT) employing 4G Long Term Evolution (LTE) Cellular Technology are presented. In this system, Doppler shifts are handled by an appliqué providing a front-end to a Commercial-Off-The-Shelf (COTS) LTE modem. The appliqué performs Doppler compensation via real time analysis of LTE signals and allows seamless operation of LTE signaling, including connection setup and handover events. The flight test results clearly demonstrate the capability of the system to sustain an uninterrupted LTE connection at varying aircraft speeds that produce Doppler shifts that are significantly higher than what typical commercial LTE systems can handle. In these tests, the radio link delivered substantial bi-directional data rates up to a maximum range of 60km, while the combination of Doppler compensation and handovers at the TA provided an uninterrupted radio link in both directions. Overall, the flight test confirmed that an airborne COTS LTE transceiver in combination with a Doppler compensation appliqué can successfully operate at high aircraft speeds. This proof-of-concept flight test also forms a basis from which to further refine AMT solutions based on 4G, and the emerging 5G technologies.

Key words: "Test Range," Telemetry, Aeronautical, Mobile, Cellular, LTE, 4G, Handover, Speed, "Doppler Shift," "Frequency Offset," Compensation, "Flight Test".

1. Introduction

Aeronautical Mobile Telemetry (AMT) for test ranges is typically implemented with point-to-point radio links using a wide-beam antenna on the Test Article (TA) that is paired with a high-gain parabolic ground station antenna pointing to and tracking the TA. The spectrum used by the radio link is dedicated to that link for the duration of the test. As pointed out in [1] and [2], an alternative implementation of AMT, using a cellular network employing 4th generation 3GPP LTE standards [3], offers several potential benefits over current practice. However, in order to implement such a system, a few technical hurdles need to be overcome. The most formidable of these technical challenges is one posed by the high Doppler shifts in the uplink (UL) signals encountered in an AMT system. In the 4G LTE-based AMT system described in [2], the problem of high Doppler shifts in the UL signal is overcome by deploying a Doppler compensation appliqué between the LTE terminal (located in the TA) and its transmit antennas. (See fig. 1 below.) The appliqué estimates the Doppler shift by processing the UL signals transmitted by the LTE terminal and proactively compensates for the shift expected to be present in the signals as they arrive at the base

station receiver. As a result, UL signals arriving at the base station have their center frequency close to where the base station receiver expects it to be.
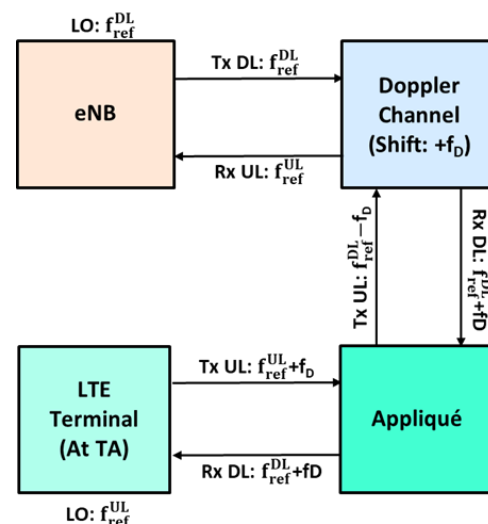
Fig. 1. A Schematic of LTE-Based System with a Doppler Compensation Appliqué

The combination of a COTS LTE terminal and the appliqué has been extensively tested in the lab with a variety of simulated environments. As

the next step in proving the viability of the LTE-based AMT system, a set of flight tests was planned and executed at Edwards Air Force Base (EAFB) in October, 2019. This paper presents the test setup that was used in the flight tests and describes the highlights of the results of these tests. Over all, the flight tests provided positive confirmation of the viability of an LTE-based AMT system. They also brought out the key features of system behavior that point out the areas where further refinement may help improve the overall system performance.

The rest of this paper is organized as follows: In Section 2, we describe the test setup at EAFB including the layout of the terrestrial segment of the LTE network. Section 3 presents a discussion of the results obtained in the flight tests and their analysis. Finally, Section 4 rounds out the paper by summarizing our conclusions.

2. Flight Test Setup

The proposed AMT system will essentially comprise a complete cellular network based on the 3GPP LTE standard [3], which was developed by the 3rd Generation Partnership Project (3GPP). 3GPP is a worldwide organization for mobile wireless standards, which developed the first LTE standard (Release 8) in 2008. Several enhancements of the LTE standards have been published in subsequent releases since then.

As in a typical LTE network, the ground-based network infrastructure of the proposed AMT system will have base stations (also referred to as eNodeBs or eNBs) connected via backhaul links to the Enhanced Packet Core (EPC). The EPC includes the key elements of the LTE core network. An LTE User Equipment (UE), which is the LTE term for mobile devices that connect to the network, will be mounted on the TA to provide the communication link to carry telemetry data to the ground network. The UE will establish an LTE connection to the wireless infrastructure on the ground, which will then forward the telemetry data to the appropriate server. Fig. 2 shows an illustration of the proposed AMT network architecture.

As shown in Fig. 2, each eNB is connected to the Mobility Management Entity (MME) through an S1 interface on the control plane, and to the Serving GateWay (SGW) with an S1-U interface on the user plane. In LTE terms, the eNBs comprise the Radio Access Network (RAN), while the rest of the functional elements other than the Network Management System (NMS) and Integrated Cellular Network Controller (ICNC) comprise the LTE Evolved Packet Core (EPC). The RAN and the EPC together are referred to as the LTE network.
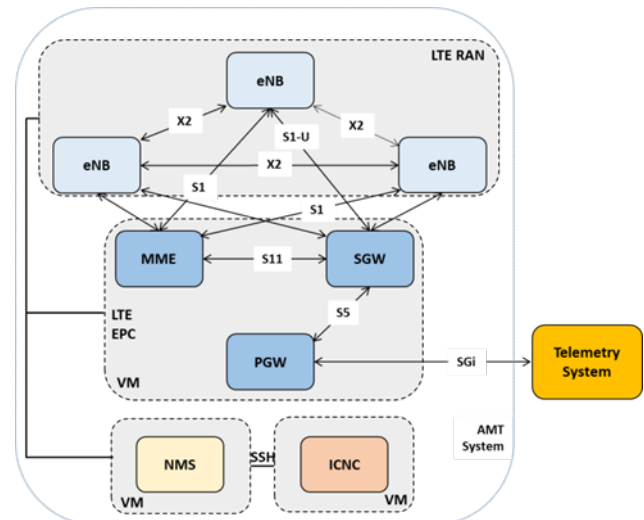
Fig. 2. An Illustration of the Network Architecture of the Proposed AMT System

It should be highlighted that the interfaces referred to in the previous paragraph are Internet Protocol (IP)-based and logical. That is, no direct eNB-eNB physical connection is required. The control plane's physical connection links can be routed through the MME if the MME is placed at a central location. The Evolved Packet Core delivers data to external elements, like the existing range telemetry system, via the SGi interface. The entire LTE network is managed through a Network Management System (NMS). Each vendor's implementation of the NMS is different as it is not a standardized element. However, the functionality remains the same; that is, all NMSs provide facilities to manage parameters and settings on all of the LTE network elements, and obtain key performance indicators (KPIs).

A key element of the proposed AMT system is the Integrated Cellular Network Controller (ICNC). The ICNC interacts with elements of the LTE network to force handovers to the desired eNB; the mechanism to do so is through the NMS. Parameters of the LTE network can be modified in real time through the interface to the NMS. The EPC, the NMS, and the ICNC can be located on the same server, with a virtualized implementation of each element.

Perspecta Labs installed a complete stand-alone LTE network at EAFB. The LTE RAN consisted of 3 LTE base stations (eNB1, eNB2 and eNB3) installed at different locations at EAFB. The objective was to provide coverage to the runway during taxiing and take-off and a broad open area while flying at low and high altitudes. The eNBs operated with power amplifiers and 4-way receive diversity in the Lower C-Band. The antennas had a broad beamwidth in azimuth and elevation and were up-tilted for maximum elevation-angle-range coverage. The

three eNBs' backhaul connections to the LTE core network were wireline, providing IP connectivity. The ICNC was communicating in real time with the eNBs receiving real-time reports from the physical and RRC layers on the status of the connections.

Perspecta Labs' Velocite airborne LTE transceiver consisted of a COTS LTE terminal, an RF power amplifier and a Doppler Compensation appliqué based on software-defined radio. The system was installed on a rack and mounted in the pressurized cabin of a Beechcraft C-12J aircraft. Two antennas, both situated at the aft and bottom of the fuselage were used for the test. The aircraft flew at different altitudes up to 20,000ft and at up to 253KGS (knots Ground Speed) during the test. The Velocite transceiver, would connect to a File Transfer Protocol (FTP) server on the ground through the over-the-air LTE connection. The FTP server was connected to the SGi interface of the LTE PGW node. The airborne Velocite transceiver would connect to the FTP server and generate IP traffic in both directions. A layer was built on top of this iperf application to enable optimized operation in the presence of disruptions in the radio link whose performance. The link was assessed through the iperf-provided statistics.

Actual time and position information was recorded during the flight, and compared with the LTE system measurements after the flight.

3. Flight Tests: Results and Analysis

The flight tests were carried out over two days – October 30-31, 2019. On each day, a specific flight plan was prepared and executed. Each flight plan comprised a take-off segment and various routes including linear segments and circular paths. These routes tested different features of the LTE-based AMT system. Figure 3 shown below illustrates the flight plan executed on the first day. The figure also shows the three base stations (with their respective orientations) that constituted the ground segment of the AMT system. A similarly complex flight plan was executed during the second day of flight tests.
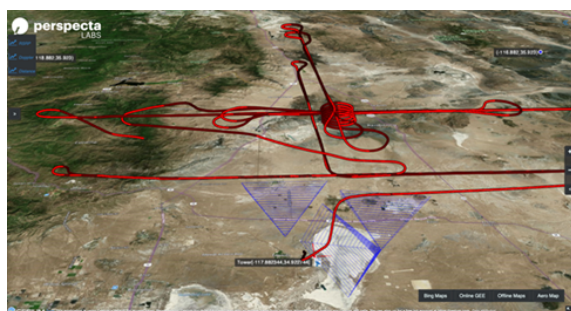
Fig. 3. Flight Plan Executed on the First Day

During the flight tests, a large amount of data was collected. While the main focus of the test was to study the system's ability to maintain the bi-directional link between the airborne LTE terminal and the ground-based base stations and deliver high throughput, data on a large number of parameters was collected to study how they affected system performance. The more important among them were the Doppler shift, distance, speed, received signal strength (RSRP), handover successes and failures.

On-board systems provided Time-Space-Position Information (TSPI) data, which included the GPS-synchronized universal time, instantaneous location (latitude, longitude, altitude), speed, and heading of the TA. This data was collected in real-time. The Doppler shift, distance from each eNB, and the Line-Of-Sight (LOS) angle of arrival of RF signals were then calculated from the TSPI data after the test was complete. The Doppler shift calculated in this manner is referred to as the actual Doppler.

During the test, the CeRTN ICNC software was used to capture LTE protocol messages generated by the eNB. These were signaling messages associated with the link between the LTE terminal and the eNB that provided information regarding the reported RSRP, timing advance, connectivity, and pending handovers. They also included information on any neighboring eNBs that were detected by the terminal. The information on neighbor eNBs was limited to RSRP, eNB ID, and pending handovers.

Figures 4a-4c, shown below, are illustrative of some of the data collected during the tests. Fig. 4a shows the RSRP from the serving and neighbor base stations as reported by the LTE terminal. Successful handover events (represented by green vertical lines) have been superimposed on the RSRP measurements. Note that RSRP measurements from neighbor base stations are available only if it is strong enough to be detected by the LTE terminal. Notice the close bunching of handover events near the 09:33 mark. Here, the RSRP levels associated with two or three base stations are close to each other. As a result, relatively small fluctuations in the measured RSPP levels cause corresponding changes in the base station seen by the terminal as the best to connect to, leading to frequent handovers. It is interesting to note that the distances from the base stations involved in the handovers to the LTE terminal are close to each other during this phase. (See fig. 4b.) Fig. 4c shows the actual and estimated Doppler shift for eNB1 and eNB2 during the take-off phase. (The estimated Doppler is referred to as measured Doppler in the figure.) Note that the appliqué estimates Doppler only

for the serving base station. It is clear that the measured (estimated) Doppler closely tracks the actual Doppler for the serving base station regardless of the base station it is connected to.
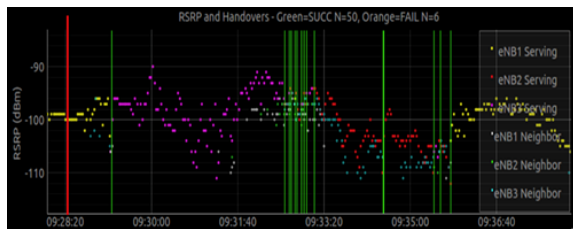

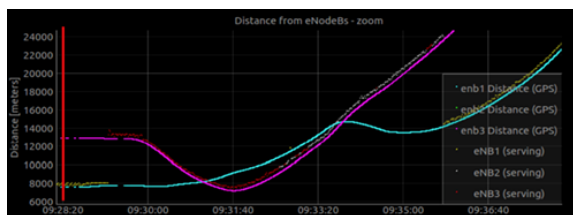*Fig. 4a. RSRP and Handovers during Takeoff Phase*

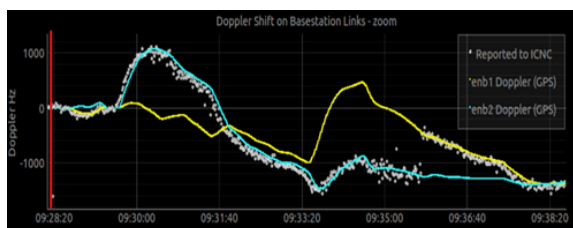*Fig. 4b. Distance to Base Stations during Takeoff Phase*

Fig. 4c. Doppler Estimation during Takeoff Phase

One of the important characteristics that was sought to be tested was the maximum range for the wireless connection between the LTE terminal and its serving base station. In order to determine the maximum range, the fight path included a segment where the TA was flown in a straight line until the connection was dropped.
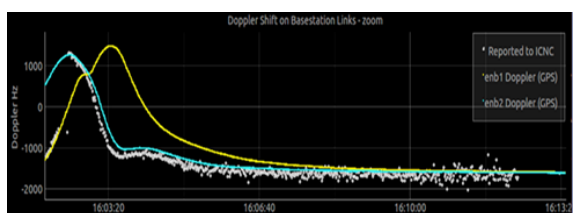
Fig. 5a. Doppler Shifts during Max. Range Phase
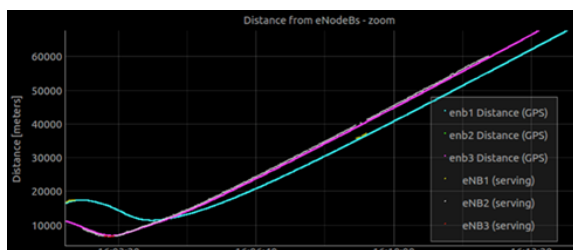
Fig. 5b. Distances to Base Stations during Maximum Range Phase

Figures 5a and 5b respectively show the measured and actual Doppler shifts and the distances to the base stations eNB1 and eNB2 during this segment. As one can see from these figures, the flight test determined the maximum range for the LTE connection to be around 60 km. Also, it is evident that the appliqué was able to accurately estimate and compensate for the Doppler shift over this entire range.

Data throughputs for the downlink and uplink directions during the maximum-range segment are shown in figs. 6a and 6b, respectively. They show clear correlation to distance dependent pathloss with strong peak data rate near the eNBs and a sustained data rate up to 60Km.
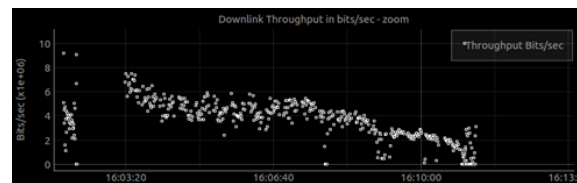
Fig. 6a. Ground-to-Air (DL) Data Throughput
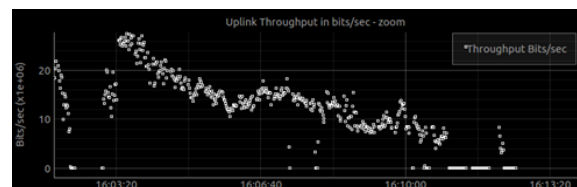
Fig. 6b. Air-to-Ground (UL) Data Throughput

We now present plots of the probability distribution functions (PDFs) of uplink (UL, i.e. air-to-ground) and downlink (DL, i.e. ground-to-air) throughputs and the Doppler shifts that were observed during the course of the tests.
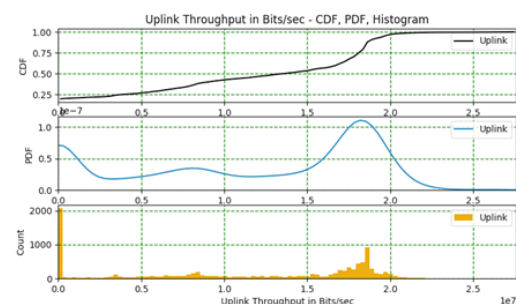
Fig. 7a. PDF of Uplink Throughput
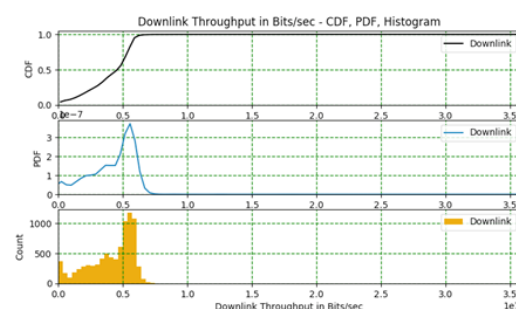
Fig. 7b. PDF of Downlink Throughput

Figures 7a, 7b respectively show the PDFs of UL and DL throughputs. As these figures show, the median UL throughput was 13 Mbps with the 75th percentile at 18 Mbps, while the median DL throughput was 4.5 Mbps. The disparity between the UL and DL throughputs was due to the fact that in the Time-Division-Duplex (TDD) scheme used in the wireless link, the UL connection had access to the wireless medium 80% of the time. The probability mass at zero can be attributed mainly to the following causes: some quirks in the iperf software used to generate traffic, and breaks in the link during certain maneuvers that caused the TA antenna to be obscured by the aircraft body.
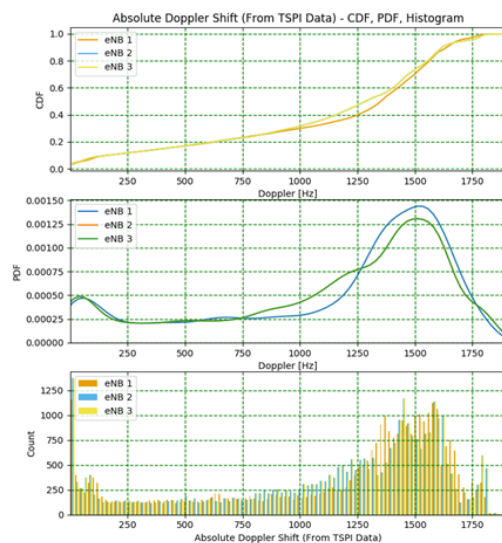
Fig. 8. PDF of Doppler Shifts

Fig. 8 shows the PDF of the Doppler shifts observed during the flight tests. As can be seen in the figure, most of the probability mass in the Doppler shift associated with both base stations is located between 1.20 and 1.75 kHz, which is well beyond the tolerance of COTS base station receivers. However, as demonstrated throughout this test, the combination of the LTE modem with the Doppler compensation appliqué enables the AMT system to maintain a good-quality link to the base stations on the ground, delivering a high throughput.

We conclude this section with some remarks on the handovers observed during the flight tests. Handovers enable seamless connectivity in the cellular-based AMT system, and a high rate of handover successes indicate a well-designed network. During the flight test, a large number of handovers were observed (more than 140) with an overall success rate of 82%. This demonstrated the agility of the prototype system to follow the LTE signaling procedures. Moreover, the data showed us areas where the handover thresholds and hysteresis levels

needed further fine-tuning to adjust to the flight characteristics.

4. Conclusions

The flight tests carried out at EAFB to study the viability of the Velocite AMT system demonstrated a strong agility to adapt to varying high Doppler shifts. The prototype system used in the test showed that it was able to estimate and compensate for the Doppler shift at the serving base station under a variety of challenging conditions, helping to maintain link connectivity and delivering a high throughput. It was also demonstrated that the system could support a maximum range of 60 km and a high probability of handover success. Overall, it was shown that the addition of a Doppler appliqué on a COTS LTE modem at the airborne system in combination with a well-designed ground network enables a full duplex AMT link over large areas that can be covered with relatively few cells equipped with broad-beam antennas. Future prototypes are expected to provide enhancements for higher speeds and more features to further improve the robustness of the radio link.

Acknowledgements

References

[1] A. Kogiantis, K. Rege and A. Triolo, "LTE System Architecture for Coverage and Doppler Reduction in Range Telemetry," presented at *The International Telemetering Conference*, Las Vegas, Nevada, Oct. 23-26, 2017.

[2] E. Fung, W.H. Johnson, A. Kogiantis and K.M. Rege, "Doppler Estimation and Compensation for LTE-Based Aeronautical Mobile Telemetry," presented at *The International Telemetering Conference*, Glendale, Arizona, Nov. 5-8, 2018.

[3] S. Sesia, I. Toufik, and M. Baker (eds.), *LTE – The UMTS Long Term Evolution – From Theory to Practice*, Second Edition, John Wiley and Sons, Chichester, U.K., 2011.