Sequential FDIA for Autonomous Integrity Monitoring of Navigation Maps on Board Vehicles

Clément Zinoune, Philippe Bonnifait, Javier Ibañez-Guzmán

Sequential FDIA for Autonomous Integrity Monitoring of Navigation Maps on board Vehicles

Clément Zinoune[1,2], Philippe Bonnifait[1], Javier Ibañez-Guzmán[2]

***Abstract*—This paper addresses the problem of Fault Detection, Isolation, and Adaptation (FDIA) in navigation systems on board passenger vehicles. The aim is to prevent malfunctions in systems such as advanced driving assistance systems and autonomous driving functions that use data provided by the navigation system. The integrity of the estimation of the vehicle position provided by the navigation system is continuously monitored and assessed. The proposed approach uses an additional estimate of vehicle position that is independent of the navigation system and based on data from standard vehicle sensors. First, fault detection consists in comparing the two estimates using a sequential statistical test to detect discrepancies despite the presence of noise. Second, fault isolation and adaptation is introduced to identify faulty estimates and to provide a correction where necessary. The FDIA framework presented here utilizes repeated trips along the same roads as a source of redundancy. Relevant properties of this formalism are given and verified experimentally using an equipped vehicle in rural and urban conditions and with various map faults. Results show that sequential FDIA performed well, even in difficult GNSS conditions.**

I. Introduction

Among the innovations that are transforming today's passenger vehicles, navigation maps are an important component. Maps were first introduced as part of navigation systems used to provide guidance information to the driver. Now they are used to provide context information to informative Advanced Driving Assistance Systems (ADASs) and their use has been extended to actuating ADASs [1]. Maps are also central components in the autonomous vehicles that are currently under development in the automotive industry [2]. Navigation maps are therefore playing an increasingly significant role in vehicle automation and progressively replacing the human driver as regards inferring the current and future vehicle context.

In recent years maps have sometimes been seen by the intelligent vehicle community as a perfect source of information. This assumption originates from robotics-oriented maps that were made manually with high accuracy, but this assumption is no longer valid when using global maps. The imperfections of a global map may not matter very much when the map is interpreted by a human, but they can have serious consequences as the degree of automation of the vehicle increases. Like any other source of information, navigation maps must be treated with caution.

How well the navigation map represents the geometry of the road has a direct impact on the performance of intelligent vehicle navigation systems. Knowledge of the geometry of the road ahead of the vehicle is currently used to improve sensor tracking (e.g., lane markings for lane-keeping functions, or a leading vehicle for adaptive cruise control applications) and enables hazardous situations to be anticipated, by adapting the vehicle speed (e.g., curve warning systems). Geometric information contained in the navigation map is also essential for some elements of highly automated driving, including path planning, decision making and control functions [3].

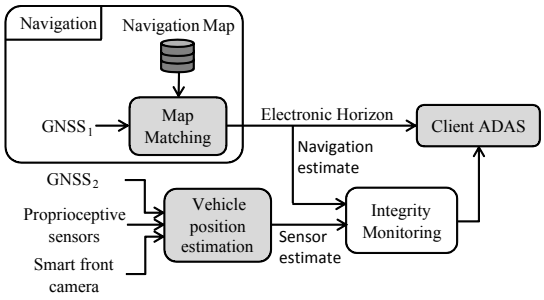


Figure 1. Framework for integrity monitoring in a passenger vehicle.

This paper presents a new method for detecting, isolating, and adapting geometrical errors in maps in order to avoid dysfunctions in client systems. Fig. 1 shows the system architecture used in the proposed approach. In passenger vehicles considered in this work, the navigation system provides context information to client ADASs. Because of industrial constraints, navigation systems cannot be modified for integrity monitoring purposes. A new *Integrity Monitoring* function (Fig. 1) is therefore added to monitor integrity in real time. The estimate of the vehicle position provided by the navigation function is continuously evaluated by comparing it to another position estimate. This sensor estimate is computed independently of the navigation system, using on-board vehicle sensors, and a fault is detected when these two estimates differ. One contribution of this work is to use Page's sequential statistical test to detect discrepancies between these two estimates despite the noise resulting from the use of standard vehicle sensors. When this test detects a discrepancy, ambiguity exists on the estimate affected by the fault (i.e. either the navigation, the independent estimate, or both can be faulty). This paper also develops a complete framework for overcoming this ambiguity by making use of repeated vehicle trips. Using a model of the effects of faults on estimates, fault isolation and adaptation is performed by comparing current and past position estimates. Structural properties of this formalism demonstrate that fault isolation capability improves as the number of trips increases, and that adaptation (i.e. the identification of a fault-free estimate that can serve as a correction) is possible when faults are isolated. Finally, the proposed method is tested using real data provided by a test vehicle in different driving conditions (rural and urban areas).

[1] Sorbonne Universités, Université de Technologie de Compiègne, CNRS Heudiasyc UMR 7253, France. [2] Renault S.A.S, France.

This article is organized as follows. Section II provides the theoretical context and background of this work. Section III states the problem that is addressed. Section IV introduces our fault detection method based on Page's test. Section V presents the proposed framework for fault isolation and adaptation, and details important inherent properties for intelligent vehicle applications. Experimental evaluation of the proposed method is performed with a test vehicle and is presented in Section VI. Conclusions of this work are discussed in Section VII.

II. Background

This section provides definitions of the terminology used, before describing work related to the evaluation of the integrity of each component of the navigation function.

A. Definitions

The terms *fault*, *error*, and *failure* may have different meanings according to the application domain. The following definitions are used in the context of this research and are based on those given in [4]:

- Fault: Error generative process. The presence of a fault may not lead to an error. An incorrect representation of the road network in the digital map of the navigation system is a fault that leads to an error only when the vehicle travels on the road where the fault is present.
- Error: A discrepancy between a computed, observed or measured value and the true, specified, or theoretically correct value.
- Failure: Instant in time when a required function exceeds the acceptable limits or is terminated.

Integrity is an important feature when navigation functions rely on Information technologies. Integrity can be defined as the ability of a system to provide user with accurate timely, complete and unambiguous information and warnings when the system should not be used.

B. Navigation System Integrity Monitoring

In intelligent vehicles, the navigation function provides relevant contextual information to client systems (ADASs or autonomous driving functions) in real time. This might be the distance to the next intersection, the curvature of the road ahead, or the current speed limit. This function can be schematized as having three parts, namely the localization system, the map-matching process and the navigation map. Map-matching consists in finding, within the navigation map, the road on which the vehicle is travelling, according to the position calculated by the localization system.

Localization in passenger vehicles is mainly based on Global Navigation Satellite Systems (GNSS). A GNSS receiver uses the time-of-flight measurements of electromagnetic signals emitted by satellites whose positions can be reconstructed using ephemeris data. The signals will sometimes be perturbed or reflected (i.e. multipath), which induces errors in the computation of position. The integrity risk arises from the use of faulty pseudo-ranges in this process. Classical integrity evaluation involves evaluating the coherency of the satellite measurements (fault detection) and then computing a protection level. This is Receiver Autonomous Integrity Monitoring (RAIM) [5]. It is, however, assumed that there is at most one faulty measurement at any one time, which is an unrealistically optimistic assumption in complex environments such as urban areas. Other approaches extend RAIM principles to a larger number of faulty measurements, using interval-based methods and relaxed intersections of constraints [6], [7] with fast implementations for fault detections [8], or an isotropy-based approach [9]. Terrain elevation models or building heights provided by a three-dimensional navigation map can also be used to determine the Non-Line-Of-Sight (NLOS) satellites, i.e. satellites that must be ignored in the position calculation [10], [11], [12]. The vehicle proprioceptive sensors (e.g., odometer, speedometer, gyroscope) are finally used to estimate the vehicle motion. However, since positional drift increases with time and distance, this technique is combined with GNSS using, for example, an extended Kalman filter [13], [14].

Integrity evaluation of a navigation map is a rather different problem which, unlike RAIM approaches, is not metric. A reference (i.e. ground truth) navigation map can be used to evaluate the vehicle map (subject map). In [15], fuzzy logic is used to compute an outlier index that expresses how a geographical object belongs to its spatio-temporal neighbourhood. This approach aims at detecting faults as well as temporal changes in maps. Studies were done on large-scale databases, in particular by crowdsourcing geographical information, like in the OpenStreetMap initiative [16], [17], [18]. Methods inspired from the SLAM (Simultaneous Localization And Mapping) domain can also be employed. The position information given by the navigation map is treated as an observation analogous to observations from other sensors [19]. To be considered as a ground truth, the reference navigation map must be created by an accurate, complete survey. In the literature some works have used alternative sources of information such as aerial imagery [20], [21], [22] or the mining of a large number of GNSS tracks [23] to create the reference map. These approaches assume that any disparities between the reference and subject maps are due to faults in the subject map, and do not address the possibility of faults in the reference map (due to an offset in aerial imagery or recent changes in the road network) or in both maps.

Integrity evaluation of the map matching process is highly dependent on the method used for the choice of road candidates. Monte Carlo-based approaches such as particle filters can be used when available computational resources allow. A set of particles (each representing a possible vehicle position) are spread over the whole road network. The population changes over time according to available measurements (e.g., GNSS measurement, DR estimation) and finally yields a solution [24], [25]. In [26], [27], the road candidates were represented by a set of hypotheses. A Bayesian framework was used to choose the most likely road. Evidence theory can also be used, since it is a convenient way to handle conflict in data fusion [28]. Fuzzy logic may also be considered, to address the complexity of the map-matching problem and the large number of criteria involved in choosing the road candidate. In [29], [30], the authors used a Sugeno fuzzy inference system to choose the

road in the navigation map based on position uncertainty, the distance between the road and the vehicle position estimate, and the angular difference between the road direction and the vehicle heading. The vehicle navigation system usually provides a confidence index associated with the map-matched vehicle position. This corresponds to the final score of the optimization process employed in the map-matching, according to a given GNSS estimate and a given navigation map. However, this index should not be taken as a measure of the quality of the navigation map. In case of a sparse road network, the map-matching function is likely to provide a high-confidence index despite an offset of the road in the navigation map due to the low number of road candidates.

The concept of user-level integrity was introduced in [31] to emphasize the necessity of taking into account every step of the positioning process (GNSS, navigation map and map-matching) in the vehicle position integrity monitoring problem. The authors presented a strategy based on successive evaluation of GNSS integrity, map complexity and map-matching solution integrity. However, this requires having access to the internal data of every sub-function in the navigation system. In the approach presented here, functions are treated as black boxes due to industrial constraints. It is not possible for us to have access to low-level data such as the time-of-flight measurements, the complete navigation map data or internal variables of the map-matching algorithm. Only high-level data is available, such as the calculated vehicle position before and after map-matching, and the contextual information related to the current vehicle position. Consequently, system monitoring approaches can be appropriate.

Observer-based system monitoring consists in comparing outputs with estimations of the outputs based on the inputs. The residuals are signals that result from the difference between estimates and actual outputs [32]. These are null when the system is not affected by any faults. If a fault is activated, the residuals are non-null. When faults are detected, the consequences they have on the system are observed. A look-up table linking the different faults to their corresponding effects on the system would enable them to be identified unambiguously and therefore to be isolated and excluded and/or corrected from the system to keep it operating correctly or at a different level of performance. This kind of process is known as Fault Detection, Isolation, and Adaptation (FDIA). Based on the system model and the available measurements, a logical link between faults and residual values can be established and summarized in a signature matrix. A complete framework to detect multiple faults in a system was presented in [33]. The sensitivity of a set of residuals is determined using a system model, and diagnoses to be applied are established, based on the observed residuals. In this paper we develop a similar approach for an FDIA navigation system. Some kind of processing of the values of the residuals is essential when real signals are used. Because of the noise affecting them, different change detection strategies must be applied. An extensive description of the mathematical tools available for signal change detection can be found in [34].

The approach presented in [35] uses an architecture similar to the navigation systems studied in the present work. It showed that detecting unexpected large discrepancies between estimated and measured positions is not sufficient, since the noise associated with poor quality sensors creates an excessive sensitivity to outliers. A Cumulative Sum (CUSUM) test is therefore implemented to reduce the number of false alarms.

III. Problem Statement

A. Monitoring System

A systemic diagram of the proposed integrity monitoring system in a vehicle is shown in Fig. 2. Relevant information about the vehicle's current and future road environment is sent to the client systems. This information represents a set of context events encountered by the vehicle as it travels, and is consequently known as an Electronic Horizon (EH) [36].

The black box assumption that is made regarding the navigation system means that the only available observation of the road geometry is the map-matched position estimate denoted as N. The purpose of the method presented in this paper is to provide an indication of the integrity of the navigation system (in particular where road geometry faults are present in the map) to the systems that use this information. If a loss of integrity is detected, a correction can be provided to the client systems. To do so, an estimate of the vehicle position independent of the navigation system is required. This estimated position is denoted as G in the figure and computed using an additional GNSS receiver $GNSS_2$ based on a different technology than $GNSS_1$. Vehicle proprioceptive sensors (e.g. odometer and a yaw rate gyroscope) can be employed to improve its accuracy and availability. This estimation might also be affected by a fault. If the two estimates differ, there is an ambiguity in the faulty estimate. This ambiguity cannot be resolved, owing to the low level of redundancy (the degree of freedom being only one). The main idea behind this framework is to make use of repeated vehicle trips to resolve this ambiguity. The output *Knowledge of fault* (Fig. 2) has three possible values:

- *Use*. The estimate provided by the navigation function to client systems is not affected by any fault.
- *Unknown*. A fault has been detected but has not been isolated. The position estimate from the navigation system is possibly affected by a fault.
- *Don't use*. A fault affects the current estimate from the navigation system and the method provides a fault-free estimate to client systems through the output *Correction*.

Let us recall that the fault detection step is merely declaring that at least one of the estimates is affected by a fault. The isolation step is determining which estimate(s) is (are) affected by a fault.

B. Spatial Sampling

In our proposed approach, the integrity of the vehicle position estimate from the navigation system is spatially evaluated. Each location on the road network is considered as an operating point of the system to be monitored (i.e. the navigation system). For a given location of the vehicle, the presence of a fault is investigated using all the estimates recorded at this location during the course of vehicle trips.

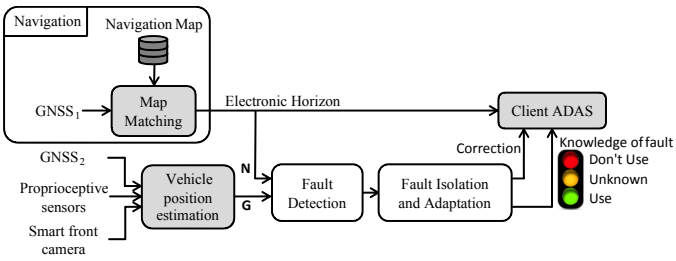


Figure 2. Structure of fault detection isolation and adaptation in a standard passenger vehicle.

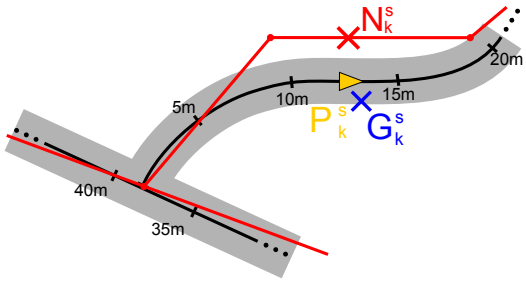


Figure 3. Illustration of the notational convention. The true (i.e. real) road is in grey and its centreline in black. Here it is spatially sampled with a 5-metre interval. The yellow arrow represents the true vehicle pose. The map is in red. The vehicle position as estimated by the navigation (resp. by the vehicle sensors) is the red (resp. blue) cross.

The method is spatially sampled with respect to the curvilinear abscissa of the road. The vehicle curvilinear abscissa on a given road is the distance along the carriageway with respect to its origin and is written $s \in \mathbb{R}^+$, as shown in Fig. 3.

Let $K \in \mathbb{N}$ denote the total number of trips made by the vehicle on a given road. The true vehicle position at abscissa s of a given road and at the k^{th} trip is written P_k^s. This can be encoded as a vector that contains the vehicle's geographic coordinates, that is to say longitude, latitude and ellipsoidal height.

Using the same notational convention, G_k^s and N_k^s are estimates of the vehicle position P_k^s provided by the sensors and the navigation respectively. Whenever the vehicle is at abscissa s of a given road for the k^{th} time, these two estimates are recorded.

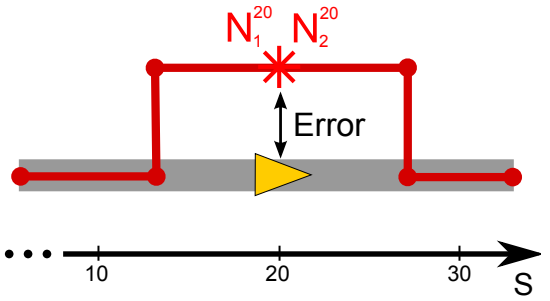
Faults may affect the navigation as well as the position estimate from sensors, and cause their value to be significantly different from the ground truth (if a multipath affects a GNSS receiver for example). In this case, the estimates are said to be faulty.

Let us define the faults $f_{N_k^s}$ and $f_{G_k^s}$ with:

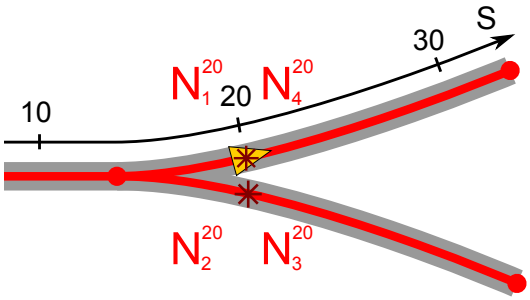
$$f_{N_k^s} \stackrel{\text{def}}{=} \begin{cases} 1 & if\ N_k^s \neq P_k^s \\ 0 & otherwise \end{cases} \quad (1)$$

$$f_{G_k^s} \stackrel{\text{def}}{=} \begin{cases} 1 & if\ G_k^s \neq P_k^s \\ 0 & otherwise \end{cases} \quad (2)$$

Fig. 3 illustrates the notational convention. Since physical quantities cannot be strictly equal, a threshold on the distance between the estimates is employed for implementation. Since the true vehicle position is not measurable directly, the fault detection and isolation are based on a pairwise comparison of estimates.



(a) Systematic errors due to a fault in the navigation map (the map is displayed in red). The faulty estimates of the vehicle position provided by the navigation function at abscissa $s = 20$ are the same.



(b) Non-systematic faults in the navigation function. The map-matching chooses either the right or the wrong road from one trip to another. The navigation estimate may or may not be faulty. Errors on the faulty estimates are nevertheless equal (here N_2^{20} and N_3^{20}).

Figure 4. Navigation system fault and error characterization

C. Assumptions

A fault is an error generative process according to the definition stated in Section II-A. An error is therefore the discrepancy caused by the fault, and is measurable by an appropriate external observer. A faulty navigation system (from the client ADAS point of view) can result from a navigation map fault. Map faults cause systematic errors. Every time the vehicle traverses the area shown in Fig. 4a, the navigation system will provide the same faulty position estimate.

A faulty state of the navigation system may not only be a consequence of map faults. Map-matching may choose a wrong road candidate because of a difficult road configuration such as a junction, as shown in Fig. 4b. In this situation the map-matching may or may not choose the right road segment from one trip to another; faults are not systematic. Nevertheless, it will be remarked that where there is a fault, the resulting error is always the same, since the output of the navigation estimate is constrained by the map. Errors by the navigation system, when they occur, are therefore systematic.

In this work we do not address the problem of determining the reason for an estimation error made by the navigation, because this would require access to the internal variables of the system.

The estimate of the vehicle position from the on-board vehicle sensors G depends mainly on the GNSS estimation. Given a location on the road network, faults on a G estimate will have two principal causes: first, multipath (i.e. satellite signal reflection on buildings, for instance) and, second, a

poor satellite configuration. The magnitude of the day-to-day multipath correlation of a static receiver is typically around 85% [37]. We have no knowledge of correlation values for a moving receiver, since the receiver motion mitigates the multipath effects. Moreover, a GNSS multipath error can repeat itself only at the same place with the same satellite configuration (the ground track repeat is 23hr 56min for GPS and 10 days for Galileo). Therefore, we believe that a repetition of the same multipath errors and faults is very unlikely from one trip to another. Errors and faults due to poor satellite geometry (multipath aside) result from the propagation of pseudo-range random measurement errors (dilution of precision), and therefore also have weak correlation between two vehicle trips. Errors on G are assumed to be different from one trip to another.

The assumptions underlying the FDIA framework can be summarized as follows:

- When travelling several times on a road, the vehicle follows the same path with small deviations (which can be compensated if necessary by lane marking measurements from a front camera).
- Any fault affecting position estimates from sensors can cause errors of any values. Errors on faulty vehicle position estimates from sensors are different from one trip to another at a given abscissa.
- The navigation map does not change from one trip to another. Errors on the vehicle position estimates from the navigation (when they occur due to a fault) are therefore always the same at a given abscissa.
- Errors on the vehicle position estimates from sensors and from the navigation are independent of each other at a given abscissa.

Given these assumptions, we have:

$$P_i^s = P_{i+1}^s,\ \forall i \in \{1, \ldots, K-1\} \tag{3}$$

Where P is the true vehicle position, s is the curvilinear abscissa, i is the trip index and K is the total number of trips made on the road. In the sections below, the formalism is developed from the system monitoring point of view, putting temporarily aside the application to intelligent vehicles. The estimates from sensors and from the navigation (G and N respectively) are seen as estimates of the same physical quantity P, which is in accordance with the assumptions above. The curvilinear abscissa is understood as an operating point of the system to be monitored and the vehicle trips are iterations of this system.

IV. Sequential Fault Detection

The first step in FDIA consists in detecting faults by comparing the two position estimates N and G. According to the assumptions stated previously, a significant discrepancy between the estimates indicates that a fault affects at least one of them. However, noise on estimates may cause non-faulty estimates to be different from each other and induce false alarms in the detection process. For this reason, this section details the mathematical formulation of a probabilistic sequential test (called Page's test) and its application to the detection of discrepancies between position estimates.

Statistical tests are an appropriate tool for evaluating the parameters of a probability law based on set of outcomes. In our application, we seek to detect a change in the mean of the probability density function (PDF) of a set of observed data, while the standard deviation of this PDF is of the same order of magnitude as the expected mean gap. Page's test works sequentially and is especially efficient for stream data. The problem is therefore formulated as the detection of a change in the mean of a random variable that represents the distance between the estimates from sensors and from navigation.

Page's test (also known as Page's trend test) consists in statistically detecting a change in the mean of a random variable based on a likelihood ratio of hypotheses [34]. It also identifies the sample at which the change in the mean occurred. Formulation of this test in the context of map fault detection is detailed in [38].

The random variable tested here is the distance between estimates G and N. Let us see how the distance signal is generated and described in terms of mean and standard deviation. Let us consider the estimate N from the navigation as the result of a random process based on the true vehicle position P in a frame $\mathcal{R}_1$ aligned with the road:

$$N = P + \alpha \tag{4}$$

$$\Sigma_\alpha = \begin{bmatrix} \sigma_a^2 & 0 \\ 0 & 0 \end{bmatrix}_{\mathcal{R}_1} \tag{5}$$

where α is a noise assumed to be zero-mean, with a diagonal covariance matrix Σ_α. Since roads are represented in the navigation map by zero-width poly-lines, the variance of the navigation map-matched error normal to the road segment is by definition null. However, a map-matched position error along the road segment exists, and σ_a denotes the longitudinal standard deviation of the navigation estimate.

The estimate of the vehicle position from sensors G can be encoded as a two-dimensional point $G = (x, y)^T$ in the East-North plane $\mathcal{R}_0$ locally tangent to Earth with the covariance matrix Σ_β of the estimation error β provided by the localization system:

$$G = P + \beta \tag{6}$$

$$\Sigma_\beta = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix}_{\mathcal{R}_0} \tag{7}$$

In order to make the distance signal independent of the road direction, an isotropic approach is used, and this consists in using the outer circle of the ellipsoid. Its radius is $\eta = \max(\eta_i)$, η_i being the eigenvalues of Σ_β. So, the covariance matrix expressed in $\mathcal{R}_1$ is $\eta \cdot I$ (with I being the identity matrix).

The vector L is defined to be the difference between the map-matched and estimated positions as stated by the following equation. L has two independent components, namely lateral error d and longitudinal error e.

$$L = \begin{bmatrix} d \\ e \end{bmatrix} = N - G = \alpha - \beta \tag{8}$$

Under the hypothesis of independent errors, the signals d and e have the following variances:

$$\begin{aligned} \sigma_d^2 &= \eta \\ \sigma_e^2 &= \eta + \sigma_a^2 \end{aligned} \tag{9}$$

The most relevant information is the lateral position of the roads in the navigation map. The fault detection is therefore done by detecting mean changes in the signal d using η.

V. Fault Isolation and Adaptation Method

Once a fault has been detected, the problem is now to isolate the faulty estimates and perform adaptation according to the assumptions made previously, using the repetition of vehicle trips as a source of redundancy. The adaptation process consists in providing a non-faulty estimate to a client system so that it can continue to operate normally, even if the current estimate is affected by a fault. Non-faulty estimates therefore need to be identified unambiguously. The concepts of Sets of Faults and Residuals are defined first. The mathematical relationship between these two concepts is then demonstrated. Finally, we show how a set of faults can be isolated, based on the observation of residuals.

A. Definitions

1) Sets of Faults: Let the set of faults e_K^s be composed of all $f_{G_k^s}$ and $f_{N_k^s}$ for the considered iterations K at a given abscissa s:

$$e_K^s \stackrel{\text{def}}{=} \left\{ f_{G_i^s} \, , \, f_{N_j^s} \right\} \, , \, \forall i,j \in \{1,\ldots,K\} \tag{10}$$

The cardinality of e_K^s is $2K$. Each element of e_K^s is a boolean value so there are 2^{2K} possible sets written $e_{K,n}^s$:

$$\begin{cases} e_K^s \in \left\{ e_{K,n}^s \right\} \\ e_{K,n}^s \in \mathbb{B}^{2K} \, , \, \forall n \in \left\{ 1,\ldots,2^{2K} \right\} \end{cases} \tag{11}$$

Let us take an example with $K = 2$. There are $2^{2 \cdot 2} = 16$ different sets. The cardinality of each one is $2 \cdot 2 = 4$. For instance, $e_{2,5}^s = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}$ means $\left\{ f_{G_2^s} = 0 \right.$ and $f_{N_2^s} = 0$ and $f_{G_1^s} = 1$ and $\left. f_{N_1^s} = 0 \right\}$.

2) Residual Processing: At a given abscissa s and at system iteration K, every available estimate at the current iteration is compared to all the others and the result is stored in a residual vector R_K^s. The elements of R_K^s are defined as:

$$r_{G_i^s G_j^s} \stackrel{\text{def}}{=} \begin{cases} 1 & if \; G_i^s \neq G_j^s \\ 0 & otherwise \end{cases} \; \forall i,j \in \{1,\ldots,K\}, i > j \tag{12}$$

$$r_{G_i^s N_j^s} \stackrel{\text{def}}{=} \begin{cases} 1 & if \; G_i^s \neq N_j^s \\ 0 & otherwise \end{cases} \; \forall i,j \in \{1,\ldots,K\} \tag{13}$$

$$r_{N_i^s N_j^s} \stackrel{\text{def}}{=} \begin{cases} 1 & if \; N_i^s \neq N_j^s \\ 0 & otherwise \end{cases} \; \forall i,j \in \{1,\ldots,K\}, i > j \tag{14}$$

Equations (12) and (14) are restricted to $i > j$ to avoid useless redundant residuals.

R_K^s is therefore composed of $C\left(2K,2\right)$ boolean elements, where $C\left(2K,2\right)$ stands for the number of 2-combinations from a given set of $2K$ elements. We know that $C\left(2K,2\right) = K\left(2K-1\right)$ so the residual vector therefore contains $K\left(2K-1\right)$ elements.

For example, at the second iteration, the size of R_2^s is 6:

$$R_2^s = \begin{bmatrix} r_{N_2^s G_2^s} & r_{G_2^s G_1^s} & r_{N_1^s G_2^s} & r_{N_2^s G_1^s} & r_{N_1^s N_2^s} & r_{N_1^s G_1^s} \end{bmatrix} \tag{15}$$

If, for example, the estimates are such that $G_1^s \neq N_1^s = G_2^s = N_2^s$ then the residual vector is $R_2^s = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$.

3) Relationships Between Faults and Residuals: Let $\vee$ and $\oplus$ denote boolean *or* and *exclusive or* operators respectively.

Proposition 1. *The elements of the residual vector are the result of boolean operations between the faulty states of the estimates, according to the following equations:*

$$r_{G_i^s G_j^s} = f_{G_i^s} \vee f_{G_j^s} \, , \, \forall i,j \in \{1,\ldots,K\} \, , \, i > j \tag{16}$$

$$r_{G_i^s N_j^s} = f_{G_i^s} \vee f_{N_j^s} \, , \, \forall i,j \in \{1,\ldots,K\} \tag{17}$$

$$r_{N_i^s N_j^s} = f_{N_i^s} \oplus f_{N_j^s} \, , \, \forall i,j \in \{1,\ldots,K\} \, , \, i > j \tag{18}$$

The demonstration of this proposition is developed in [39].

Eq. (16), (17) and (18) of Proposition 1 establish a link between the available estimates (i.e. G and N) and the faults which affected them (i.e. f_G and f_M). The first two equations tell us that if there is at least one fault on the considered estimates, then the residual is affected. In (18), the residual is equal to one if there is a single fault among the two estimates. It is now possible to deduce the presence of faults based on observation and comparison of the estimates.

B. Fault Isolation Principles

The fault detection and isolation strategy involves listing all the possible sets of faults for a given iteration K, and calculating the corresponding theoretical residual vectors with (16), (17) and (18). This forms the truth table for K. In parallel, available estimates are used to compute the observed residual vector based on (12), (13) and (14). This vector, present in the truth table, allows the corresponding set of faults to be determined. Faults affecting each estimate can finally be deduced from this set. It will be remarked that the truth tables are valid for every operating point, so the superscript s is omitted in the tables.

Let us take the example given in Section V-A2. At the first system iteration at operating point s, two estimates are available: G_1^s and N_1^s. The truth table for one iteration is shown in Table I. It is assumed in this example that $G_1^s \neq N_1^s$ is observed, therefore $r_{G_1^s N_1^s} = 1$, according to (12). Table I shows that this residual can be due to three sets of faults: $e_{1,2}^s$, $e_{1,3}^s$ and $e_{1,4}^s$. After one system iteration, it can be concluded that there is at least one faulty estimate among G_1^s and N_1^s, but it is not possible to determine which one. The fault is detected, but not isolated.

Table I
TRUTH TABLE FOR ONE ITERATION ($K = 1$). THE FIRST RESIDUAL $r_{G_1N_1} = 0$ APPEARS ONLY ONCE IN THE TABLE, AND SINCE THIS MAKES ISOLATION POSSIBLE, IT IS SHOWN IN GREEN. CONVERSELY, $r_{G_1N_1} = 1$ IS DUE TO MORE THAN ONE SET OF FAULTS AND IS SHOWN IN RED. THE RESIDUAL USED AS EXPLANATION EXAMPLE IS IN BOLD.

Sets of faults $e_{K,n}$			Residuals
	f_{G_1}	f_{N_1}	$r_{G_1N_1} = f_{G_1} \vee f_{N_1}$
$e_{1,1}$	0	0	0
$e_{1,2}$	1	0	**1**
$e_{1,3}$	0	1	**1**
$e_{1,4}$	1	1	**1**

At the second system iteration at the operating point s, a new pair of estimates is available: G_2^s and N_2^s. The truth table for two system iterations is calculated with (16), (17) and (18) and is shown in Table II. In this example and similarly to Section V-A2, it is assumed that $G_1^s \neq N_1^s = G_2^s = N_2^s$ is observed. This leads to the residual $R_2^s = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$. Table II shows that this residual (in bold) is exclusively due to the set of faults $e_{2,5}^s$. After the second system iteration, fault isolation is performed by concluding that $\{f_{G_2^s} = 0$ and $f_{N_2^s} = 0$ and $f_{G_1^s} = 1$ and $f_{N_1^s} = 0\}$.

C. Conditions of Isolability

By definition, the truth table is exhaustive; the observed residual vector is necessarily included within it. However, some sets of faults induce the same residual vector, as shown by the red colour in Tables I and II. In this case, isolation is not possible. These are called *Adverse* sets. At least one more system iteration is required to perform isolation.

Being adverse depends on the number of faults affecting the estimates, as stated in Proposition 2.:

Proposition 2. *A set of faulty states is adverse if and only if it corresponds to one of the following conditions:*

$f_{N_i} = 1$, $\forall i \in \{1, \ldots, K\}$ *and* $\exists! j \in \{1, \ldots, K\}$ *such that* $f_{G_j} = 0$

$f_{G_i} = 1$, $\forall i \in \{1, \ldots, K\}$

In other words, it is not possible to isolate faults if:

1) Every estimate N is faulty and there is a single fault-free G.
2) Every G is faulty.

The proof of this Proposition can be found in [40].

It will be remarked in the example developed previously that after the first system iteration (i.e. $K = 1$), the situation corresponded to the second condition of this proposition because $f_{N_1^s} = 0$ and $f_{G_1^s} = 1$. This is why fault isolation was impossible. However, after the second iteration, the set chosen for the example $\{f_{G_2^s} = 0$ and $f_{N_2^s} = 0$ and $f_{G_1^s} = 1$ and $f_{N_1^s} = 0\}$ no longer corresponded to either of these conditions. Fault isolation had therefore become possible.

Proposition 2 is fundamental for demonstrating the internal formalism properties. These are detailed and demonstrated as follows.

D. Formalism Properties

Once the bases of the formalism are established, we have the properties shown below in the listed propositions. Demonstrations of these properties can be found in [39].

Proposition 3. *Guaranteed fault detection: The formalism always detects the presence of faulty estimates. In other words, each time there is a faulty estimate, the formalism detects it (but may not be able to isolate the faulty estimate).*

Proposition 4. *Isolation convergence: The ratio of the number of adverse sets of faulty states to the total number of sets tends to zero as the number of iterations increases. In other words, increasing K improves fault isolation capabilities.*

Proposition 5. *Conservation of isolability: Once fault isolation is performed, fault isolation will be performed at any new iteration.*

Proposition 6. *Adaptation: If fault detection and isolation are performed, then adaptation is possible.*

It should be recalled that adaptation consists in identifying a fault-free estimate once detection and isolation have been performed.

Proposition 7. *Conservation of adaptation: If fault isolation is achieved at the K^{th} iteration, adaptation is possible at iteration $K+1$ whatever the faults affecting the new estimates.*

These propositions have important consequences for the application of the method in intelligent vehicles. First, Proposition 3 shows that the presence of a fault among the available estimates is always detected by the method. This means that where there is no fault, the method is able to declare this fact with certainty even at the first system iteration, allowing client systems to function. Integrity monitoring is therefore possible with this method. Second, Proposition 4 shows that a new iteration will always contribute information for fault isolation, which justifies multiple system iterations. Third, according to Propositions 5 to 7, once a fault has been isolated, a fault-free estimated can be provided to client systems at any future iteration, allowing client systems to anticipate being able to operate properly at any future iteration.

E. Illustrative Example

We now take the FDIA formalism proposed above and apply it to monitoring the integrity of the navigation vehicle position estimate as introduced in Fig. 2. Using a simple example, each step is described in detail. The map contains an error and we show how the method performs fault detection, isolation and adaptation. In addition to detailing each step of our proposed method, we illustrate the properties introduced in Section V-D.

In this example (depicted in Fig. 5), the real road is straight, while the map's representation of the road includes a bend. Let us detail the proposed formalism at abscissa 25 m in the first trip shown in Fig. 5a.

The first time the vehicle is at abscissa $s = 25$, position estimates are provided by the vehicle state (G_1^{25}) and by the navigation (N_1^{25}) functions. The observed residual can be computed using (13):

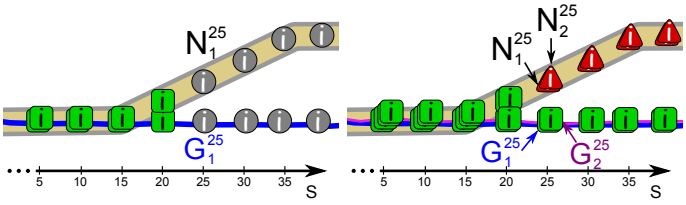
$$G_1^{25} \neq N_1^{25} \Rightarrow r_{G_1^{25}N_1^{25}} = 1$$

This residual is found three times in the truth table for one FDIA trip (Table I): the sets of faults $e_{1,2}^{25}$, $e_{1,3}^{25}$ and $e_{1,4}^{25}$ give

Table II

TRUTH TABLE FOR TWO ITERATIONS ($K = 2$). RESIDUALS OCCURRING ONLY ONCE ARE IN GREEN, SINCE THEY MAKE ISOLATION POSSIBLE. CONVERSELY, RESIDUALS THAT ARE DUE TO MORE THAN ONE SET ON FAULTY STATES ARE IN RED. THE RESIDUAL USED AS EXPLANATION EXAMPLES IS IN BOLD.

	Sets of faults $e_{K,n}$				Residuals					
	f_{G_2}	f_{N_2}	f_{G_1}	f_{N_1}	$r_{N_2G_2}$	$r_{G_2G_1}$	$r_{N_1G_2}$	$r_{N_2G_1}$	$r_{N_1N_2}$	$r_{N_1G_1}$
$e_{2,1}$	0	0	0	0	0	0	0	0	0	0
$e_{2,2}$	1	0	0	0	1	1	1	0	0	0
$e_{2,3}$	0	1	0	0	1	0	0	1	1	0
$e_{2,4}$	1	1	0	0	1	1	1	1	1	0
$e_{2,5}$	0	0	1	0	***0***	***1***	***0***	***1***	***0***	***1***
$e_{2,6}$	1	0	1	0	1	1	1	1	0	1
$e_{2,7}$	0	1	1	0	1	1	0	1	1	1
$e_{2,8}$	1	1	1	0	1	1	1	1	1	1
$e_{2,9}$	0	0	0	1	0	0	1	0	1	1
$e_{2,10}$	1	0	0	1	1	1	1	0	1	1
$e_{2,11}$	0	1	0	1	1	0	1	1	0	1
$e_{2,12}$	1	1	0	1	1	1	1	1	0	1
$e_{2,13}$	0	0	1	1	0	1	1	1	1	1
$e_{2,14}$	1	0	1	1	1	1	1	1	1	1
$e_{2,15}$	0	1	1	1	1	1	1	1	0	1
$e_{2,16}$	1	1	1	1	1	1	1	1	0	1



(a) First trip (blue line) (b) Second trip (purple line)

Figure 5. A faulty map area. Circular grey marks are for estimates where the method has detected but not isolated a fault. Green squares are for true estimates and red triangles are the faulty estimates.

$r_{G_1N_1} = 1$. The proposed method then detects a faulty estimate among G_1^{25} and N_1^{25} but is not able to isolate it. The integrity monitoring system cannot specify the faultiness of N_1^{25}, but simply sends *Knowledge of fault: Unknown* to client systems, as shown by circular grey marks in Fig. 5a.

The second time the vehicle traverses abscissa $s = 25$ of the same road (Fig. 5b), a new pair of position estimates becomes available: G_2^{25} and N_2^{25}. The dimension of the residual vector increases to 6. The elements are calculated using (12), (13) and (14):

$$N_2^{25} \neq G_2^{25} \Rightarrow r_{N_2^{25}G_2^{25}} = 1$$
$$G_2^{25} = G_1^{25} \Rightarrow r_{G_2^{25}G_1^{25}} = 0$$
$$N_1^{25} \neq G_2^{25} \Rightarrow r_{N_1^{25}G_2^{25}} = 1$$
$$N_2^{25} \neq G_1^{25} \Rightarrow r_{N_2^{25}G_1^{25}} = 1$$
$$N_1^{25} = N_2^{25} \Rightarrow r_{N_1^{25}N_2^{25}} = 0$$
$$G_1^{25} \neq N_1^{25} \Rightarrow r_{G_1^{25}N_1^{25}} = 1$$

Then $R_2^{25} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$.

Table II is the truth table for two trips. Following the first trip observation it is known that $f_{G_1^{25}}$ and $f_{N_1^{25}}$ are not both null, and the first four rows of Table II may therefore be ignored. The observed residual is found only once in this table (caused by the set of faults $e_{2,11}^{25}$). Consequently, it can be concluded that $f_{G_2^{25}} = 0$, $f_{N_2^{25}} = 1$, $f_{G_1^{25}} = 0$ and $f_{N_1^{25}} = 1$.

The integrity monitoring system returns the instruction *Knowledge of fault: don't use the navigation position estimate* (N_2^{25}) and provides a fault-free estimate instead in the output *Correction* (either G_2^{25} or G_1^{25}). On Fig. 5b, faulty (resp. true) estimates are represented by red triangles (resp. green squares). From Proposition 7 we know that the integrity monitoring system will be able to perform adaptation, i.e. provide an error-free position estimate for all future trips along this road, whatever the faults affecting the future estimates.

F. Complete Fault Detection, Isolation and Adaptation Method

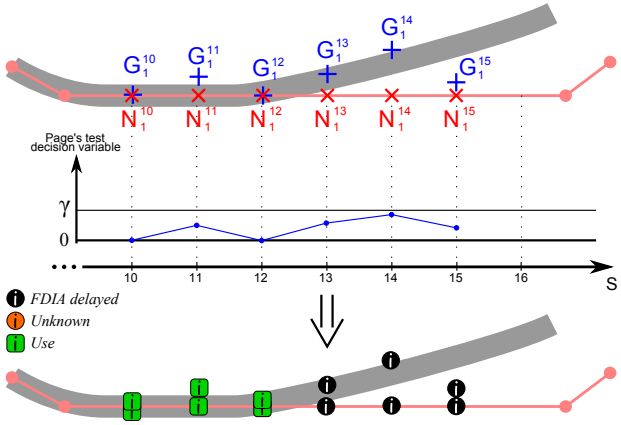
The FDIA framework introduced previously is based on the calculation of a residual vector R_K^s (s is the vehicle curvilinear abscissa on the road and K is the number of trips on this road). In practice, the elements of R_K^s are defined on the basis of comparisons of the distance between each pair of available estimates N and G with a threshold λ_d, and denoted by $r_{G_i^sG_j^s}$, $r_{G_i^sN_j^s}$ and $r_{N_i^sN_j^s}$, $\forall i, j \in \{1, \ldots, K\}$ which we recall below:

$$r_{G_i^sG_j^s} = \begin{cases} 1 & if\ dist\left(G_i^s, G_j^s\right) > \lambda_d \\ 0 & otherwise \end{cases} \quad \forall i, j \in \{1, \ldots, K\}, i > j \tag{19}$$

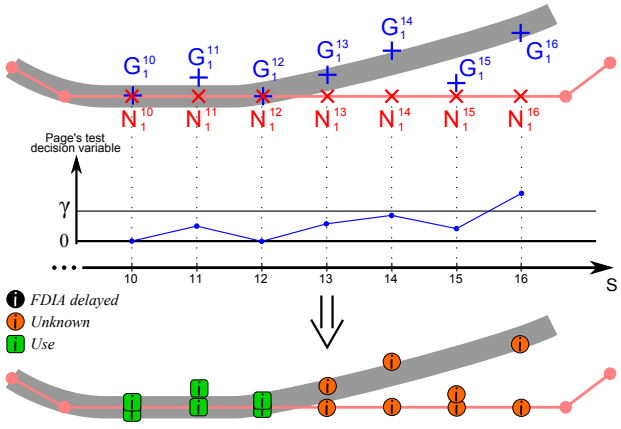
$$r_{G_i^sN_j^s} = \begin{cases} 1 & if\ dist\left(G_i^s, N_j^s\right) > \lambda_d \\ 0 & otherwise \end{cases} \quad \forall i, j \in \{1, \ldots, K\} \tag{20}$$

$$r_{N_i^sN_j^s} = \begin{cases} 1 & if\ dist\left(N_i^s, N_j^s\right) > \lambda_d \\ 0 & otherwise \end{cases} \quad \forall i, j \in \{1, \ldots, K\}, i > j \tag{21}$$

Page's test is used here instead of the distance measure for comparing G and N. According to this new formulation, the residual vector element $r_{G_K^sN_K^s}$ is zero if Page's test gives the mean of the signal d as zero. Reciprocally, $r_{G_K^sN_K^s}$ is one if the test detects a mean change in d. The manner in which the other residual elements ($r_{G_i^sG_j^s}$ and $r_{N_i^sN_j^s}$) are calculated remains unchanged.



(a) The vehicle is at abscissa $s = 15$. Page's decision variable is greater than 0 and lower than the threshold. The FDIA is then delayed since $s = 13$.



(b) Page's decision variable exceeds the threshold at $s = 16$. FDIA is run for abscissas 13 to 16 with $r_{G_1 N_1} = 1$. This results in *Knowledge of fault: Unknown*.

Figure 6. Example of the use of Page's test with the FDIA framework. The road as recorded in the navigation map is the red poly-line. The estimates from navigation N_1^s are the red crosses and the estimates from sensors G_1^s are the blue crosses. The vehicle is travelling from left to right, so its curvilinear abscissa is denoted by the axis S. The decision variable used in Page's test is plotted on the middle graph. The result of FDIA with Page's test is shown in the lower parts of the figures.

As shown in Section IV, Page's test may require a few samples before it is able to give definitive results. This is highlighted by the distance-to-alert and distance-to-recovery metrics. In such a situation, the estimates are buffered until Page's test provides a definitive output. The FDIA framework is then run on each estimate, taking the result of the test as an input.

As an example, Fig. 6 shows the progression of Page's test as the vehicle advances, and illustrates the strategy employed here for the FDIA. We are looking at the first trip along this road. The method is detailed step by step, as shown in this figure. When the vehicle is at abscissa $s = 10$, the estimates from sensors G_1^{10} and N_1^{10} are very close to each other. The Page's test decision variable equals zero, and therefore the estimates are considered to be equal. The FDIA framework is run with $r_{G_1^{10} N_1^{10}} = 0$: this residual is found only once

Figure 7. Test vehicle

in the truth table so it is not necessary to use previous trips, and the FDIA concludes that the estimates are fault-free. At abscissa $s = 11$, the Page's test decision variable is not null, but has not yet reached the threshold γ. A discrepancy between the estimates G and N is likely, but not yet detected. The estimates that correspond to abscissa $s = 11$ (i.e. G_1^{11} and N_1^{11}) are stored in the memory buffer until the Page's decision variable either exceeds the threshold or returns to zero. When the vehicle reaches abscissa $s = 12$, the estimates G_1^{12} and N_1^{12} make the decision variable return to zero, which indicates that Page's test gives no discrepancy for the two last abscissas. The FDIA is run at every abscissa in the memory buffer: at $s = 11$ with $r_{G_K^{11} N_K^{11}} = 0$, the FDIA concludes that there is no fault affecting G_1^{11} and N_1^{11}; at $s = 12$ with $r_{G_K^{12} N_K^{12}} = 0$, the FDIA concludes that there is no fault affecting G_1^{12} and N_1^{12}.

The memory buffer is cleared. At $s = 13$ to $s = 15$, the estimates are such that the decision variable is not null so these are buffered as shown by black marks on Fig. 6a. At abscissa $s = 16$ the decision variable finally exceeds γ, and so Page's test now declares that a discrepancy, starting at $s = 13$, is detected. The FDIA is then successively run at $s = 13$, $s = 14$, $s = 15$ and $s = 16$ with $r_{G_1^{13} N_1^{13}} = 1$, $r_{G_1^{14} N_1^{14}} = 1$, $r_{G_1^{15} N_1^{15}} = 1$ and $r_{G_1^{16} N_1^{16}} = 1$ respectively. Since these residuals are adverse and there are no previous trips available, the FDIA outputs *Unknown* for the estimates, as shown by orange marks on Fig. 6b. The memory buffer is emptied and the decision variable is set to zero for the following evaluation points.

VI. Experimental Evaluation

A. Test Vehicle

Experiments were done in real conditions using the Renault Espace passenger vehicle shown in Fig. 7. The navigation system used in the vehicle is fed by a standard single frequency Ublox 6T GPS receiver (corresponding to $GNSS_1$ in Fig. 2). The GNSS receiver denoted by $GNSS_2$ in Fig. 2 is a Ublox 4T GPS receiver. The vehicle odometer, speed, rear wheel speed difference and yaw rate are production-standard sensors and are available on the vehicle CAN-bus. An extended Kalman filter is used to compute the position estimate from sensor G, based on the vehicle sensors and the Ublox 4T GPS receiver [38].

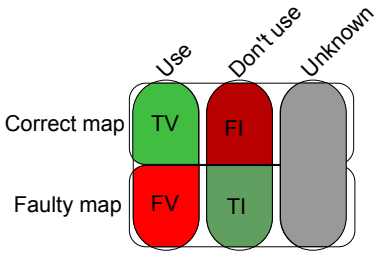


Figure 8. Metrics employed for method evaluation. The columns are the output of the method and the lines are the actual state of the navigation map.

An Ixsea LandIns Inertial Navigation System (INS) tightly coupled to a Novatel GPS receiver provides position estimates with an error of less than 1 m and is considered as position ground truth for the experiments.

B. Metrics

We saw above that our proposed method has three possible output states that refer to the current navigation estimate integrity, namely *Use, Unknown, Don't use.* The navigation map can be *correct* or *faulty*. A set of metrics is introduced as follows and illustrated in Fig. 8. These are evaluated with respect to the number of vehicle trips so that the performance of our method can be evaluated precisely.

The overall efficiency corresponds to the number of relevant diagnoses made by the method, equal to the sum of True Validations (TV) and True Isolations (TI). A TV occurs when a correct point of the navigation map has been declared with no fault. A False Validation (FV) occurs when the method trusts a faulty navigation estimate. A False Isolation (FI) occurs when a correct navigation estimate is classified as faulty by the method. The Overall Efficiency Rate (OER) is:

$$OER \stackrel{\text{def}}{=} \frac{TV + TI}{\Omega - \Omega_{unknown}} \tag{22}$$

where Ω is the number of navigation points evaluated by the method and $\Omega_{unknown}$ is the number of navigation estimates for which the method outputs *Unknown*. An OER close to one would indicate that whenever the method provides an output different from *Unknown*, this diagnosis is reliable.

The output *Unknown* does not provide information on the integrity of the navigation estimate from the point of view of client systems. From the applicative point of view, this output should occur as little as possible. The performance of the method in terms of information availability is measured by the Information Availability Rate (IAR):

$$IAR \stackrel{\text{def}}{=} \frac{\Omega - \Omega_{unknown}}{\Omega} \tag{23}$$

This is expected to converge to one as the number of trips increases.

C. Urban Test Track

In this experiment, the vehicle was driven close to large buildings. The GPS receiver was perturbed by multipath effects caused by signals reflecting off buildings. These measurements are expected to be isolated by the method. As shown in Fig. 9, over parts of the circuit conditions are good, and the deviation of the GPS measurements is less than the width of the road. It will be remarked that for testing the method, these experimental conditions are challenging. The length of each trip is 1100 m and the spatial sampling has been done along the map with a 10 m period, and the tolerance on the vehicle curvilinear abscissa is $\lambda_s = 2$ m. Hence, $\Omega = 110$ points on the navigation map need to be evaluated at each trip. This value varies by a few points from one trip to another because data recordings were not started and stopped rigorously at the same positions. The threshold on the distance between the estimates must be chosen according to two criteria. First, it must be as small as possible to comply with assumptions made as bases for the method. Second, it must be greater than the tolerance on the vehicle abscissa λ_s, so that two estimates from navigation that correspond to the same abscissa are considered as equal by the method. Page's test is therefore set to detect a discrepancy of $\delta_m = \lambda_s = 2$ m between the estimates with the detection threshold $\gamma = 4.\sigma/\delta_m$.

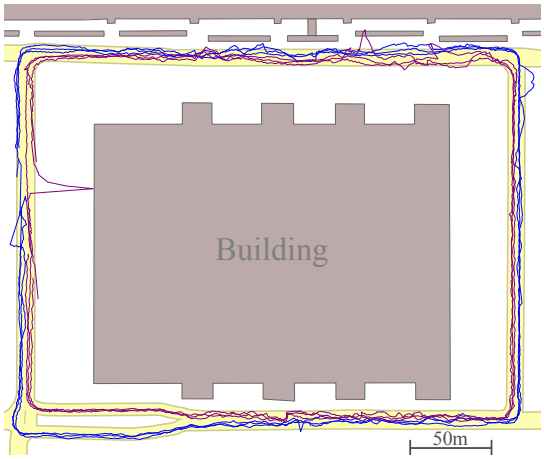


Figure 9. GNSS tracks on the correct navigation map. Clockwise (blue) and anticlockwise (purple)

Faults were generated randomly in five different maps using dedicated software. The performance of the complete FDIA method is evaluated using the metrics introduced previously and detailed in Fig. 10 and 11.

Fig. 10 shows the ratio of correctly identified points to the number of isolated or validated points. At the first vehicle trip the method cannot perform isolation. The OE is then only composed of TV. The OER at the first trip is therefore favoured by the absence of false validation; the OER of five of the ten tests therefore equal one. It will be noted that the OER of map 1 anticlockwise is especially low at the first trip (50%), but this is not significant since it is calculated using only four points. The OER tends to remain constant from the second to the third trip with medians equal to 84% and 83% respectively.

Fig. 11 summarizes the ratio of the number of validated or isolated points to the number of points considered *Unknown*. The IAR increases with the number of trips for all the tests and exceeds 90% at the third trip. The FDIA method is therefore seen to converge as stated by Proposition 4.

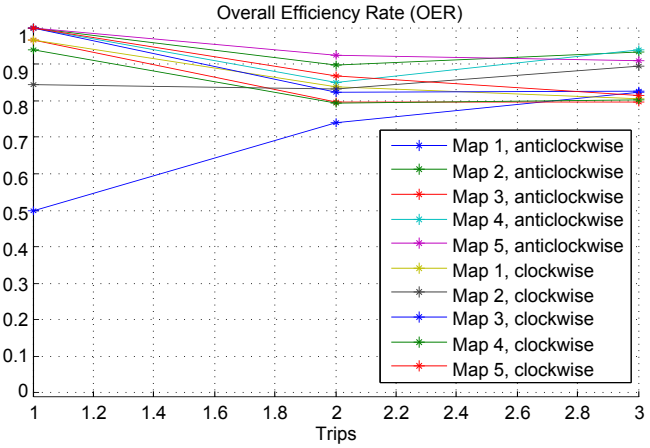


Figure 10. Overall Efficiency Rate.

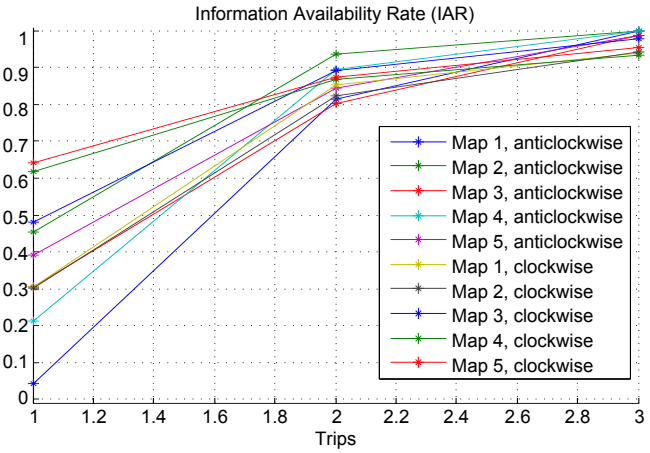


Figure 11. Information Availability Rate.

D. Rural Test Track

Here we look at how the method performed in an area where real map errors were present. The road had been modified when a new motorway was built. A 2008 *Navteq* navigation map was used to run the FDIA method. Fig. 12 shows that this map contains three major faults, described below from left to right.

The first fault is where the road now deviates as it passes over the motorway. The second is where a completely new stretch of road has been created, deviating significantly from the old one. For these two cases the confidence accorded to the estimates both from the sensors and from navigation are high. In a rural environment, many satellites are in the receiver line-of-sight, which increases the level of confidence and reduces the position standard deviations and Dilution of Precision. Moreover, the road network is quite simple, so the map-matching algorithm provides a high level of confidence even if the GNSS measurement is a few metres away from the road. The challenge is therefore to determine precisely the reason for any disparity between estimates from sensors and from navigation, that is to determine which estimate is affected by a fault. When the real road is too far from the map road, the map-matching confidence index suddenly decreases and

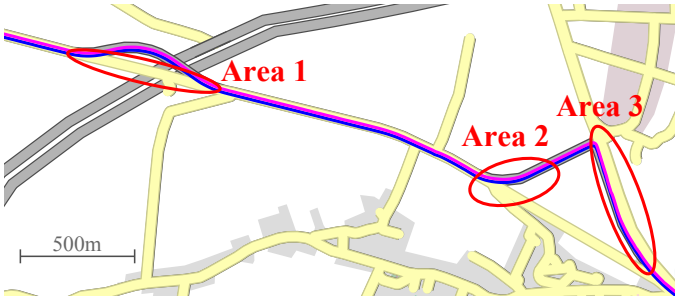


Figure 12. Rural test track. The navigation map used in the experiment is in yellow, the correct map is in grey in background. The vehicle goes from left to right. The estimates from sensors of the first (resp. second) trip is in blue (resp. purple).

the navigation function switches to *off-road* mode and stops providing navigation estimates. The FDIA method consequently stops until a new estimate is provided by the navigation system.

The third fault is where a new road now exists parallel to the old one. Even if the estimate from navigation is relatively close to the true vehicle position in this area, the method should identify the fault. Fig. 12 also shows the estimates from sensors for the two trips used in this experiment.

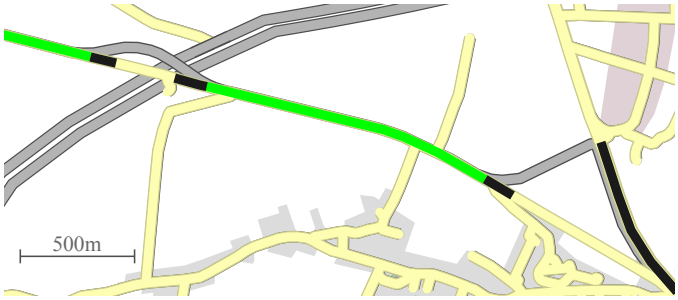
Fig. 13a shows the result of the FDIA applied to this dataset after the first trip. The green stretches are where the method returned *Use* and the black stretches are where the output was *Unknown*. There is no FI, since the method cannot isolate a fault at the first trip, as described above. It will nevertheless be noted that there is no FV of 0% and the OER is 100%. This means that the method correctly identified situations where estimates were not affected by faults and consequently provided the output *Use,* and also that it detected situations where at least one fault affected the estimates and consequently provided the output *Unknown* to client systems. The IAR of this first trip is 77% which corresponds to the proportion of erroneous roads in the navigation map.

The results obtained after the second vehicle trip in this area are shown in Fig. 13b. Here again, $OER = 100$ % which means that every estimate not declared *Unknown* at the second trip was correctly identified. Moreover, every point traversed during the course of two trips was declared either *Use* or *Don't use,* and so the Information Availability Rate equals 100%.

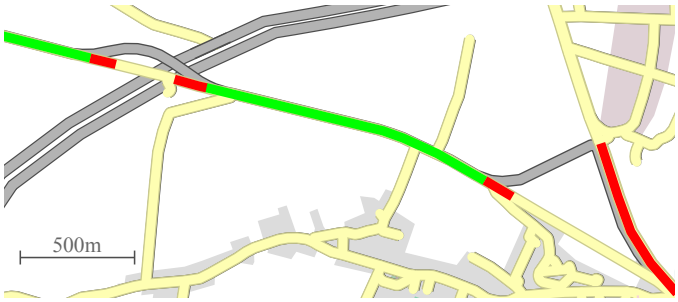
This experiment shows that the method performed well when using real vehicle data and a real navigation map with faults. The absence of False Isolations and particularly False Validations, and the high Information Availability in these conditions indicate that the FDIA framework is a realistic option for navigation integrity monitoring.

E. Discussion

These results, obtained using map faults that were either injected or real, show that the isolation convergence property is verified, since the number of points for which the method cannot perform isolation decreases and can reach zero. The



(a) First trip.



(b) Second trip.

Figure 13. Results of the FDIA for the rural test track. The road sections for which the method outputs *Use* are in green, those for which the method outputs *Don't use* are in red, and those for which the method outputs *Unknown* are in black. The true navigation map is in the background in grey.

information availability rate increases and can reach one as the number of trips increases.

Faults that are not correctly isolated by the method (i.e. False Validations and False Isolations) result mainly from the trade-off between spatial re-sampling tolerance of sensor data and the comparison threshold used for the computation of residuals.

The design of the method is based on several assumptions (random faults in the observer estimates, systematic faults in the map, and independence between them). The results confirm experimentally the validity of these assumptions. Faults in the observer are essentially due to the additional GPS receiver and can arise from multipath. If at a given abscissa, the same multipath induces the same error on the receiver computation fix at two different trips used by the FDIA method, the first assumption is violated. In this case, the method fails to isolate faults. Nevertheless, this requires two conditions to be fulfilled: first, the same satellite geometry at the same abscissa during two different trips, and, second, the same position fix error after filtering. For these reasons, we believe that the violation of the first assumption is very unlikely. This situation was never encountered during the course of our experiments.

VII. Conclusions

This paper introduces a framework for monitoring the integrity of navigation map geometry by detecting and isolating faults on the estimate of the vehicle position from the navigation system. We showed that the context of intelligent vehicles in which this work takes place limits the quality of the sensors and the redundancy of the sources of information. The FDIA framework detailed in this work fills this gap by making use of repeated vehicle trips.

The framework is based on a pairwise comparison of spatially-sampled vehicle position estimates between the current and past vehicle trips that gives rise to residual vectors. We demonstrate that under the assumptions made the proposed FDIA framework is theoretically always able to perform fault detection. However, depending on the number of faults that affect the estimates and on the number of vehicle trips, it may not be possible to perform isolation, that is, to determine without ambiguity which estimate(s) is (are) affected by a fault. By defining such sets of faults mathematically, we demonstrate that the fault isolation and adaptation capabilities of the method improve as the number of vehicle trips increases. The proposed framework was tested using real sensor data and navigation map faults. Performance was excellent in open sky areas and promising in urban conditions. This highlights the interest of using this FDIA approach in intelligent vehicles.

References

[1] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. Keller, E. Kaus, R. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Knoppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb, "Making bertha drive, an autonomous journey on a historic route," *Intelligent Transportation Systems Magazine, IEEE*, vol. 6, no. 2, pp. 8–20, Summer 2014.

[2] A. Eskandarian, *Handbook of Intelligent Vehicles*, ser. Handbook of Intelligent Vehicles. Springer, 2012, no. vol. 2.

[3] S. Durekovic and N. Smith, "Architectures of map-supported ADAS," in *IEEE Intelligent Vehicles Symposium (IV)*, june 2011, pp. 207 –211.

[4] V. Popovic and B. Vasic, "Review of hazard analysis methods and their basic characteristics," *FME Trans.*, vol. 36, no. 4, pp. 181–187, 2008.

[5] A. K. Brown, "Receiver autonomous integrity monitoring using a 24-satellite GPS constellation," in *Institute of Navigation, Technical Meeting*, vol. 1, 1987, pp. 256–262.

[6] L. Jaulin, M. Kieffer, I. Braems, and E. Walter, "Guaranteed nonlinear estimation using constraint propagation on sets," *International Journal of Control*, vol. 74, no. 18, pp. 1772–1782, 2001.

[7] V. Drevelle and P. Bonnifait, "Localization confidence domains via set inversion on short-term trajectory," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1244–1256, 2013.

[8] ——, "Interval-based fast fault detection and identification applied to radio-navigation multipath," *International Journal of Adaptive Control and Signal Processing*, 2015. [Online]. Available: http://dx.doi.org/10.1002/acs.2535

[9] J. Cosmen-Schortmann, M. Azaola-Saenz, M. A. Martinez-Olague, and M. Toledo-Lopez, "Integrity in urban and road environments and its use in liability critical applications," in *IEEE/ION Position, Location and Navigation Symposium*, 2008, pp. 972–983.

[10] C. Pinana-Diaz, R. Toledo-Moreo, D. Betaille, and A. Gomez-Skarmeta, "GPS multipath detection and exclusion with elevation-enhanced maps," in *Intelligent Transportation Systems, IEEE 14th Int. Conf. on*, 2011.

[11] S. Peyraud, D. Bétaille, S. Renault, M. Ortiz, F. Mougel, D. Meizel, and F. Peyret, "About non-line-of-sight satellite detection and exclusion in a 3D map-aided localization algorithm," *Sensors*, vol. 13, pp. 829–847, 2013.

[12] M. Obst, S. Bauer, P. Reisdorf, and G. Wanielik, "Multipath detection with 3D digital maps for robust multi-constellation GNSS/INS vehicle localization in urban areas," in *IEEE Intelligent Veh. Symp. (IV)*, 2012, pp. 184–190.

[13] P. Bonnifait, P. Bouron, P. Crubille, and D. Meizel, "Data fusion of four ABS sensors and GPS for an enhanced localization of car-like vehicles," in *Robotics and Automation, IEEE Int. Conf. on*, 2001.

[14] O. Le Marchand, P. Bonnifait, J. Ibanez-Guzman, and D. Betaille, "Vehicle localization integrity based on trajectory monitoring," in *Intelligent Robots and Systems, IEEE/RSJ Int. Conf. on*, 2009.

[15] G. Grekousis and Y. Fotis, "A fuzzy index for detecting spatiotemporal outliers," *GeoInformatica*, vol. 16, pp. 597–619, 2012.

[16] J.-F. Girres and G. Touya, "Quality assessment of the french openstreetmap dataset," *Transactions in GIS*, vol. 14, no. 4, pp. 435–459, 2010.
[17] V. Noronha and M. F. Goodchild, "Map accuracy and location expression in transportation, reality and prospects," *Transportation Research Part C: Emerging Technologies*, vol. 8, no. 1 - 6, pp. 53 – 69, 2000.
[18] M. Haklay, "How good is volunteered geographical information? a comparative study of openstreetmap and ordnance survey datasets," *Environment and planning. B, Planning & design*, vol. 37, no. 4, p. 682, 2010.
[19] C. Boucher and J.-C. Noyer, "Automatic detection of topological changes for digital road map updating," *Instrumentation and Measurement, IEEE Transactions on*, vol. 61, no. 11, pp. 3094–3102, Nov 2012.
[20] L. Deren, S. Haigang, and X. Ping, "Automatic change detection of geo-spatial data from imagery," *Geo-Spatial Information Science*, vol. 6, pp. 1–7, 2003.
[21] N. de Freitas, R. Dearden, F. Hutter, R. Morales-Menendez, J. Mutch, and D. Poole, "Diagnosis by a waiter and a Mars explorer," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 455 – 468, mar 2004.
[22] O. Pink and C. Stiller, "Automated map generation from aerial images for precise vehicle localization," in *Intelligent Transportation Systems (ITSC), IEEE 13th Int. Conf. on*, sept. 2010, pp. 1517 –1522.
[23] G. Agamennoni, J. Nieto, and E. Nebot, "Robust inference of principal road paths for intelligent transportation systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, pp. 298–308, 2011.
[24] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *Signal Processing, IEEE Transactions on*, vol. 50, no. 2, pp. 425–437, 2002.
[25] I. Szottka and M. Butenuth, "Advanced particle filtering for airborne vehicle tracking in urban areas," *Geoscience and Remote Sensing Letters, IEEE*, vol. 11, pp. 686–690, 2014.
[26] C. Fouque and P. Bonnifait, "Matching raw GPS measurements on a navigable map without computing a global position," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 887–898, 2012.
[27] O. Mazhelis, "Using recursive bayesian estimation for matching GPS measurements to imperfect road network data," in *Intelligent Transportation Systems (ITSC), IEEE 13th Int. Conf. on*, 2010, pp. 1492–1497.
[28] G. Nassreddine, F. Abdallah, and T. Denoeux, "Map matching algorithm using interval analysis and dempster-shafer theory," in *Intelligent Vehicles Symposium, IEEE*, June 2009, pp. 494–499.
[29] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Integrity of map-matching algorithms," *Transportation Research Part C: Emerging Technologies*, vol. 14, no. 4, pp. 283–302, 2006.
[30] W. Y. Ochieng, M. Quddus, and R. B. Noland, "Map-matching in complex urban road networks," *Revista Brasileira de Cartografia*, vol. 2, 2009.
[31] N. R. Velaga, M. A. Quddus, A. L. Bristow, and Y. Zheng, "Map-aided integrity monitoring of a land vehicle navigation system," *Intelligent Transportation Systems, IEEE Trans. on*, vol. 13, pp. 848–858, 2012.
[32] S. X. Ding, *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, 2008.
[33] M. Krysander, F. Heintz, J. Roll, and E. Frisk, "Flexdx: A reconfigurable diagnosis framework," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 8, pp. 1303 – 1313, 2010.
[34] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, E. Cliffs, Ed. Prentice-Hall, Inc, 1993.
[35] P. Sundvall and P. Jensfelt, "Fault detection for mobile robots using redundant positioning systems," in *Robotics and Automation (ICRA). IEEE Int. Conf. on*, may 2006, pp. 3781 –3786.
[36] C. Ress, D. Balzer, A. Bracht, S. Durekovic, and J. Löwenau, "Adasis protocol for advanced in-vehicle applications," in *15th World Congress on Intelligent Transport Systems*, 2008.
[37] R. S. Radovanovic, "High accuracy deformation monitoring via multipath mitigation by day-to-day correlation analysis," in *13th International Technical Meeting of the SAT Division of the ION, September*, 2000, pp. 19–22.
[38] C. Zinoune, P. Bonnifait, and J. Ibanez-Guzman, "A sequential test for autonomous localisation of map errors for driving assistance systems," in *Intelligent Transportation Systems (ITSC), IEEE 15th Int. Conf. on*, 2012, pp. 1377–1382.
[39] ——, "Integrity monitoring of navigation systems using repetitive journeys," in *Intelligent Vehicles Symposium Proceedings, IEEE*, June 2014, pp. 274–280.
[40] A. Monteil and C. Zinoune, "Demonstration of the rules non isolability of sets of faulty states," University of Technology of Compiegne, http://hal.archives-ouvertes.fr/hal-01072412, Tech. Rep., Jan 2014.

Clément Zinoune obtained his Ph.D. in Computer Science and Engineering at the Université de Technologie de Compiègne (UTC) in France in 2014. He graduated as a mechanical engineer in 2010 at UTC. In 2010 he also obtained his master of science in autonomous vehicle dynamics and control at Cranfield University in the UK. In 2014 he joined the team charged with the development of autonomous vehicle at Renault S.A. in France.

Philippe Bonnifait is a professor in the Computer Science and Engineering department of the Université de Technologie de Compiègne (UTC) in France. He obtained his Ph.D. in automatic control and computer science at the École Centrale de Nantes in 1997. Since 1998, he has been with Heudiasyc UMR 7253, a joint research unit between UTC and CNRS. His research interests are Intelligent Vehicles, high integrity positioning and map-matching for mobile robot navigation in structured outdoor environments.

Javier Ibañez-Guzmán (IEEE Member) obtained his Ph.D. at the University of Reading on a SERC-UK fellowship, and his MSEE at the University of Pennsylvania (USA) as a Fulbright scholar. In 2011, he was visiting scholar at the University of California, Berkeley (CITRIS), working on connected vehicle applications. He is currently a member of the technical staff at Renault S.A., carrying out work on autonomous vehicle navigation technologies and driving assistance systems. Formerly he was senior scientist at a national research institute in Singapore, where he spearheaded work on autonomous ground vehicles operating in unstructured environments. Dr. Ibañez-Guzmán has several publications and patents in the robotics and automotive domains. He has successfully supervised a number of Ph.D. students. He is a C.Eng. (UK) and Fellow of the Institute of Engineering Technology (UK).