Robust Single Antenna INS Initialization for Low Dynamic Applications

Pierre Bénet, Mourad Saidani, Alexis Guinamard
SBG Systems, Carrière sur Seine, France

BIOGRAPHY

Pierre Bénet is Research Engineer at SBG Systems since 2019. His research interests include alternative navigation solutions for difficult and denied GNSS environment.

Mourad Saidani is Inertial Navigation Algorithms Manager. He joined SBG in 2017. His research interests include GNSS and inertial coupling.

Alexis Guinamard is Chief Technology Officer at SBG Systems. He co-founded SBG Systems in 2007. His research interests include GNSS, Inertial sensors and inertial filtering and more lately vision.

ABSTRACT

When coupling inertial sensors with GNSS, a key to enable robust navigation is to use reliable initialization methods. While it is straightforward to retrieve position and velocity from a standalone GNSS receiver, the attitude can be more challenging to initialize. Several methods for initializing this attitude are available in the literature. However, they all have their limitations. The process of finding the initial attitude is also referred to as the alignment.

- Course Alignment requires the carrier to go on a straight line with a significant velocity and with no lateral velocity. When the carrier is a car, the condition is easily met. However, for a plane or a boat, the wind or the current can break the assumption of no lateral velocity. Hence this initialization method is not recommended for boats or planes. It is also not recommended for pedestrian applications which can move sidewards or backward, and which exhibit very low velocity. Other applications like UAVs or Helicopters can move in all directions, making this procedure unsafe.
- Magnetic alignment which uses a magnetic compass can provide the initial attitude. this system requires a dedicated calibration procedure and is easily disturbed by surrounding metal objects.
- Gyro-compassing is the process of detecting the earth rotation, thus deducing the attitude of the system. However, detecting the rotation of the earth requires an expensive high performance gyrometer.
- Dual-antenna alignment requires two antennas on the carrier which must be separated enough. The constraint on the separation of the two antenna yields a more complex setup and is cumbersome on small carriers such as pedestrians.

More generally, many attempts have been performed to overcome these limitations and provide a free kinematic alignment process. But they usually require significant dynamics and are not suitable for pedestrian applications. We developed an optimization-based solution that overcomes these problems. In particular, it solves initialization in the very difficult scenario of pedestrian single antenna with single point GNSS with MEMS gyrometer. Typical performance on a 10°/h grade gyrometer on a pedestrian yields 1° heading uncertainty within 15s of initialization time. SBG systems has developed this initialization algorithm (patent pending) to make pedestrian mono-antenna inertial-GNSS coupling and tight-coupling possible. Whereas it was not possible before with SBG post-processing software, it has been very recently integrated into our post-processing software Qinertia 4.0. It is also planned to make it available as a real-time initialization procedure running directly on the INS. Our algorithm has been tested in many scenarios with different carriers such as pedestrian or automotive. We will present the details of this algorithm and the detailed analysis of its performance in this paper.

I. RELATED WORK

Many methods have been developed to find the heading of an INS (Gade, 2016). To circumvent difficulties of pedestrian initialization, some dedicated algorithms were developed such as step detection (Angrisano et al., 2019), Some methods require the antenna to be tilted on a pole while receiving RTK positions (Chen et al., 2020). Some gradient descents have been tried in the literature for INS initialization, but only as fine alignment after a coarse initialization, while still exhibiting slow convergence (Strus et al., 2009). Other propositions transform an optimization problem into a continuous filtering problem with self-initialization property, however research on improving its limitation is still ongoing (Zhang et al., 2020). In the other

Proceedings of the 2024 International Technical Meeting,
ION ITM 2024, January 23-25, 2024

https://doi.org/10.33012/2024.19530

hand our algorithm does not exhibit such limitations or issues. Our dedicated parametrization of the problem and intelligent regularization allows high speed convergence, quality and robustness for all kinds of carrier.

The first idea of this work came from the difficulty to initialize the INS of a backpack mapping system. To circumvent this problem, a first LIDAR SLAM processing or visual SLAM processing was necessary, then the alignnment between the SLAM trajectory and the GNSS trajectory allowed to initialize the INS (Benet et al., 2022). Hence a heavy processing was necessary to initialize the INS in this situation. However we noticed that the heading information of the SLAM processing was not necessary to the Extended Kalman Filter (Simon, 2006) to keep on track. We deduced that the problem was purely an initialization problem, and that we could find a standalone initialization procedure. As in our previous work on monocular visual inertial SLAM we had to face difficult initialization problems (Bénet and Guinamard, 2021) (Campos et al., 2021) (Von Stumberg et al., 2018), we decided to apply these algorithmic principles to find an INS alignment algorithm.

II. SOLUTION OVERVIEW

The given solution is to fit a minimal set of position points (ex: GNSS trajectory) with a pure inertial trajectory computed using the set of a high frequency inertial sensor data (3 axes accelerometers and gyrometers), using no prior assumption on the initial state, and minimal constraint on the trajectory. This is achieved by iteratively minimizing a function.

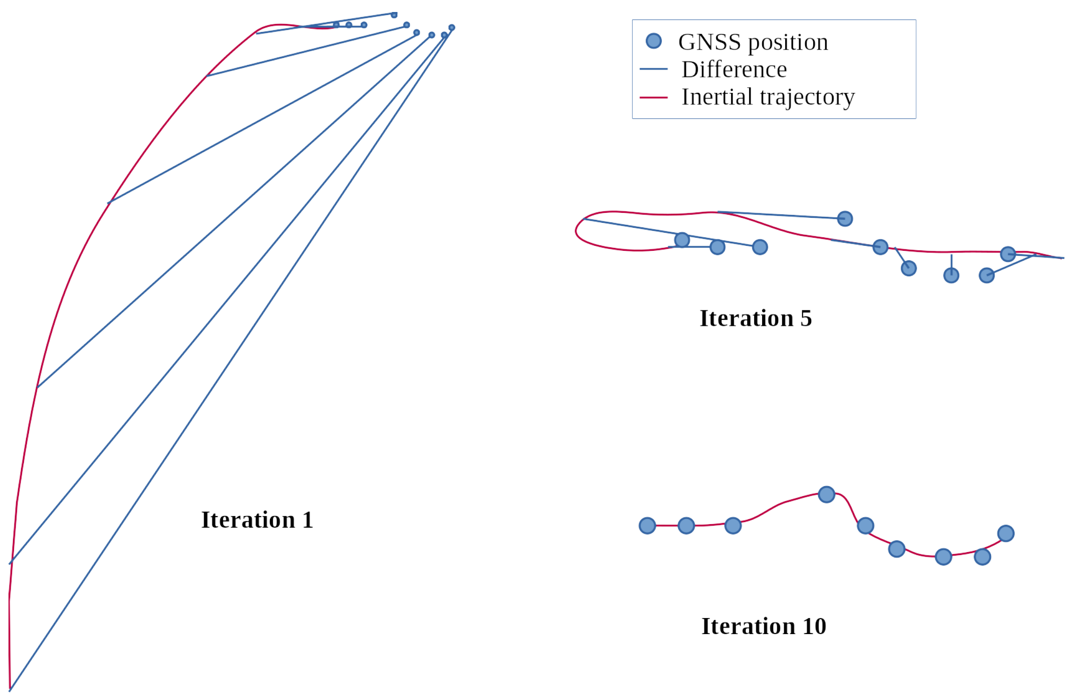


Figure 1: Illustration of the process of initializing an INS on a pedestrian trajectory using our algorithm. **Iteration 1:** At first, the INS is not initialized: roll pitch and yaw are wrong, and, the initial difference between the inertial and GNSS trajectories is important. This is because the inertial trajectory is the integration of the measured acceleration minus the gravity. In practice, the gravity is integrated on the wrong axis. **Iteration 5:** After a few iterations, the algorithm can get roll and pitch to converge: roll and pitch will converge such that the measured acceleration will be in opposition with the real gravity. However the yaw can remain wrong. **Iteration 10:** the yaw converges. As the yaw converges, we get accurate values also for IMU bias and velocity.

III. PRELIMINARY

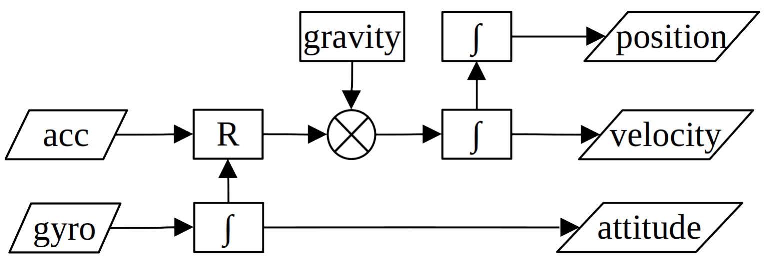
1. Notations

- g The gravity acceleration vector (0,0,-9.807) in (m/s/s).
- a_t The acceleration at time t as measured by the IMU, a 3 element vector in (m/s/s).
- w_t The rotation speed at time t as measured by the IMU, a 3 element vector in (rad/s).
- b_a The accelerometer bias, a 3 element vector in (m/s/s).

- b_g The gyrometer bias, a 3 element vector in (rad/s).
- v_t The velocity in a local NED frame at time t, a 3 element vector in (m/s).
- $GNSS_i$ The ith GNSS point in a local NED frame, a 3 element vector in (m).
- n The number of GNSS points in the window.
- t_i The time of the ith GNSS point.
- R_t The INS attitude at time t, a 3 by 3 rotation matrix.
- θ_0 The angular correction of the attitude of the INS at the starting time, a 3 element vector in (rad).
- $x = (b_a, b_g, v_0, \theta_0)^T$ The argument of the function to minimize, a vector of 12 elements.
- P_{in} The covariance of the prior knowledge of, a 12 by 12 diagonal matrix.
- P_{out} The covariance of the final knowledge of, a 12 by 12 matrix.
- $f(x)$ The observation function to minimize which output a $3 \times (n-1)$ element vector (with n the number of GNSS points in the window).
- $G = \frac{||f(x)||^2}{n}$ The averaged squared residual, a scalar matrix used as the covariance of the observation function f.
- $\frac{\partial f}{\partial x}$ The gradient of the function f, a $3 \times (n-1)$ by 12 matrix, easily approximated by central differences.

2. Inertial Mechanization

The inertial trajectory is obtained via inertial mechanization (Titterton and Weston, 2004). In the inertial filtering domain, the inertial mechanization is the process of computing a trajectory using only inertial data. It is done by integrating several times rotation speed and accelerations to yield attitude, velocity and position, as presented in the following scheme:



- The attitude is the integration of the gyrometer measurements.
- The velocity is the integration of the accelerometers plus the gravity, where the acceleration is brought back in the global frame before integration.
- The position is the integration of the velocity.

3. Repetitive measurement errors

Repetitive measurement error occurs on gyrometers and accelerometers (Titterton and Weston, 2004). In practice a wrong offset appears in the values of the gyrometer and accelerometer measurements. There is a different offset for each axes of each sensor. These offsets in the measurement are called the bias. This bias varies slowly during time. It can be considered constant during a short period of time, typically the time that it takes for an INS to initialize. It can be estimated by some algorithms such as a typical Extended Kalman Filter (Simon, 2006) (EKF) once initialized or our initialization algorithm.

It is typically taken into account in the equations of mechanization. It traduces by the use of $w_t - b_g$ instead of w_t and the use of $a_t - b_a$ instead of a_t. In our initialization algorithm, taking account of this bias and estimating its values allows our inertial trajectory to fit more accurately the GNSS points. This yield a better attitude estimation.

4. Gimbal lock

The inertial mechanization presented earlier implies integrating the gyrometer to yield angles. Any attempt to parametrize the entire set of rotation using three variables will lead to gimbal lock problems (Grassia, 1998). Using Euler angles, the equations become singular when the roll reaches 90°. Using exponential map (i.e. rotation around an axis), the equations become singular when the rotation reaches 360°. Alternatives are available such as rotation matrix or quaternions which have internal constraints. Rotation matrix must be orthonormalized, and quaternion must be normalized. However in the domain of parameter estimation, we don't want to have such additive constraints to work with.

We then need to work with both angular models. A rotation matrix will be used to accumulate small rotations estimated on Euler angles or exponential map. For example, if we estimated a correction of $\Delta\theta_0$. Instead of summing directly on the angle: $\theta_0 = \theta_0 + \Delta\theta_0$ we will apply it on a rotation matrix $R_0 = e^{[\Delta\theta_0]_\times}.R_0$. The exponential map will be allays null $\theta_0 = 0$ and will only serve as a tool for differentiating the equations and estimating an angular correction.

5. Mechanization equations

The mechanization is done by integrating a set of differential equations representing the mechanization equations (Titterton and Weston, 2004):

$$\begin{array}{lcr} \dot{R}_t = R_t[w_t - b_g]_\times & \text{with} & R_{t_0} = e^{[\theta_0]_\times}.R_0 \\ \dot{v}_t = R_t.(a_t - b_a) + g & \text{with} & v_{t_0} = v_0 \\ \dot{p}_t = v_t & \text{with} & p_{t_0} = GNSS_0 \end{array} \tag{1}$$

The exponential map is used only at the initialization of the integration, since we will estimate the starting attitude only. The velocity integration start at v_0, a variable that will be estimated. The position starts at the first GNSS point. This GNSS point is given and will not be estimated. Here, for simplification in the equations we ommited the lever arm and the earth rotation that should not be neglected in order to obtain highest accuracy and robustness.

IV. MINIMIZATION ALGORITHM

1. Objective function

The goal is to minimize a function f. It is defined as the stack of the difference of several GNSS points and the associated inertial position:

$$f\begin{pmatrix} b_a \\ b_g \\ v_0 \\ \theta_0 \end{pmatrix} = f(x) = \begin{pmatrix} p_{t_1}(x) - GNSS_1 \\ p_{t_2}(x) - GNSS_2 \\ \dots \\ p_{t_9}(x) - GNSS_9 \end{pmatrix} \tag{2}$$

- f has 12 inputs, which are the variables to estimate (accelerometer bias, gyrometer bias, initial attitude, initial velocity).
- f has $3 \times (n-1)$ outputs (the number of GNSS points minus one times three), the difference between the first GNSS point does not belong to the function because it is used as the starting point for inertial mechanization.
- many other variables are required to compute the observation function that are not to be estimated (GNSS points, accelerometer and gyrometer data, lever arm...), we can call these additive data the context of f.
- The starting point of the inertial mechanization is the last GNSS point, and the integration is done backward, so that the estimated values are up to date and ready for an Extended Kalman Filter (EKF). If the starting point is the last GNSS point, then the algorithm will estimate the starting variables at this old time. To use the solution of our initialization procedure, the EKF would have to filter the data of the whole optimization window before getting back up to date. Hence the computation is heavier. Moreover, the EKF would process data that have already been used for initialization. Which is not recommended in the context of optimal estimation (a data should be used only one time, not twice).

2. Generalized Non Linear Least Square With Prior Knowledge

The non linear least square minimizes the squared function:

$$||f(x)||^2 \tag{3}$$

The generalized non linear least square minimizes the squared Mahalanobis distance of the function:

$$||f(x)||^2_{G^{-1}} = f(x).G^{-1}.f(x)^T \tag{4}$$

To take account of the prior knowledge on x, we jointly minimize the Mahalanobis distance between x and the prior (Theil, 1963). The final quantity to minimize is:

$$||f(x)||^2_{G^{-1}} + ||x - x_0||^2_{P_{in}^{-1}} \tag{5}$$

The initial state can be set to zero or to a better approximation of the state variables if available. It's important to setup a relevant initial covariance P_{in} to enable optimal convergence:

- $P_{in[b_a,b_a]}$ can be set to the squared turn on to turn on standard deviation of the accelerometer bias as specified by the datasheet of the accelerometer.
- $P_{in[b_g,b_g]}$ can be set to the squared turn on to turn on standard deviation of the gyrometer bias as specified by the datasheet of the gyrometer.
- $P_{in[v,v]}$ can be set to the squared maximum GNSS speed in the window.
- $P_{in[\theta,\theta]}$ can be set to π^2 as all attitude can be reached within 180° degrees.

3. Gauss Newton Algorithm

To minimize the function f, we apply a Gauss Newton algorithm (Fletcher, 2000). It is an iterative algorithm that will converge to the solution. The Gauss Newton algorithm finds the values of the variables that minimizes a squared function $r(x)^2$ often called residual. A Gauss Newton iteration:

$$x = x - \left(J_r^T J_r\right)^{-1} J_r^T r(x) \tag{6}$$

with $J_r = \frac{\partial r(x)}{\partial x}$ the jacobian of the residual. In our case we define the residual as:

$$r(x) = \begin{pmatrix} G^{-1/2} f(x) & P_{in}^{-1/2} x \end{pmatrix}^T \tag{7}$$

So that the squared residual equals equation 5:

$$r(x)^2 = ||f(x)||^2_{G^{-1}} + ||x - x_0||^2_{P_{in}^{-1}} \tag{8}$$

Developing the jacobian of the residual 7 in the Gauss Newton equation 6 yields our final Gauss Newton iterative scheme:

$$x = x - \left(\left(\frac{\partial f(x)}{\partial x}\right)^T .G^{-1}.\frac{\partial f(x)}{\partial x} + P_{in}^{-1}\right)^{-1} \left(\left(\frac{\partial f(x)}{\partial x}\right)^T .G^{-1}.f(x) + P_{in}^{-1} x\right) \tag{9}$$

The quality of the estimated variables is given by the covariance :

$$P_{out} = \left(\left(\frac{\partial f(x)}{\partial x}\right)^T .G^{-1}.\frac{\partial f(x)}{\partial x} + P_{in}^{-1}\right)^{-1} \tag{10}$$

4. Solution Exploitation

The result of the optimization yields exactly the parameters that an Extended Kalman Filter (Simon, 2006) (EKF) needs. The EKF is the state of the art of INS filtering process. The parameters of a typical EKF are a state x and the covariance P of the state. This state is typically composed of the velocity, the attitude, the position, the accelerometer bias and the gyrometer bias. The initialization of most of the variables in this state is difficult. Therefore we proposed an initialization algorithm that computes these variables and their associated covariance. In the end, the algorithm produces all necessary informations that are needed to start an EKF. The only missing information is the position, which is provided by the GNSS receiver.

One can check the consistency of the solution looking at the final residual by comparing the trajectory difference f(x) with the uncertainty of the position given by GNSS. Once the inertial trajectory has been fitted to the GNSS, the remaining error is mostly caused by GNSS uncertainty. Hence if the position difference of the fitted trajectory f(x) exceed the uncertainty given by GNSS, the solution is not reliable. This happens very rarely, but this check allows to robustify the system. In case of failure to initialize, one can wait a few seconds to retry the initialization algorithm on a new window.

V. RESULTS : PERFORMANCE & INTEGRITY ASSESSMENT

Although the alignment algorithm is often overlooked (research focusing on the regular EKF operation), this entry point to the navigation filter is a key to ensure reliability of the navigation solution. An inconsistent result on the alignment algorithm is likely to cause disastrous effects on the navigation filter. That is why we took care to qualify our algorithm's integrity, in addition to the pure performance and availability evaluations. Integrity is a well known subject in the GNSS research field and has been covered in many papers in various conditions (Saidani et al., 2021). In the present publication, the integrity analysis is focused on verifying the consistency of estimated parameters with their associated estimated standard deviation. We used the following methodology:

- For each new GNSS sample, we run the initialization algorithm on the 20 second window before the new GNSS sample.
- We compare the initialization results with a reference trajectory generated by a tightly coupled RTK forward/backward/merge solution.
- We computed the availability as the rate of the sample having a predicted heading standard deviation under 5°.
- We computed the quantity 1s, 2s, 3s as the error rate greater than 1 sigma, 2 sigma, and 3 sigma.

What interests us mostly is that the standard deviations of the heading reflects its true error. In particular we checked that the yaw error follows a normal distribution with respect to its estimated standard deviation. To do so we computed the error rate greater than 1 sigma, 2 sigma, and 3 sigma. We took a particular care in tuning the weights of the observation so that the error rate of 3 sigma follows mostly the theoretical value of 0.3%. In turns, the rate of the 1 sigma and 2 sigma are a bit lower than expected (32% and 5% respectively). This is a typical behavior of applying protection factors to filter outliers.

The analysis has been made using stanford diagrams, comparing actual error with reported standard deviation. The 3 sigmas is used as diagonal, so the scale is adapted accordingly in X and Y axes.

1. Pedestrian scenario

The first pedestrian scenario is a typical pedestrian environment, with buildings and trees around the trajectory, but an overall good standalone GNSS performance is available. Typical pedestrian motion and no specific dynamic maneuvers or runs were necessary. Maximum velocity in this scenario is 1.2 m/s. In these conditions, the alignment algorithm is capable of computing the heading solution 62% of the time over the complete trajectory. Only the least dynamic epochs do not allow heading computation.The integrity assessment shows that the normal distribution is confirmed and a minimal number of outliers (errors above 3x reported standard deviation) is found.

In the second pedestrian scenario, we have a good GNSS environment, and a similar dynamic range as on first scenario, but the GNSS receiver is a different type and provides slightly lower relative performance. The lower GNSS trajectory accuracy makes it more challenging to find good heading solutions. Nevertheless, the integrity condition and the normal distribution remains verified.

The third pedestrian scenario offers a typical pedestrian trajectory in a medium urban environment. Again, no dynamic maneuvers have been performed and we see long straight lines with low dynamics. The GNSS receiver is operating in RTK mode. The performance obtained with RTK confirms that a better GNSS trajectory can positively impact the alignment algorithm.

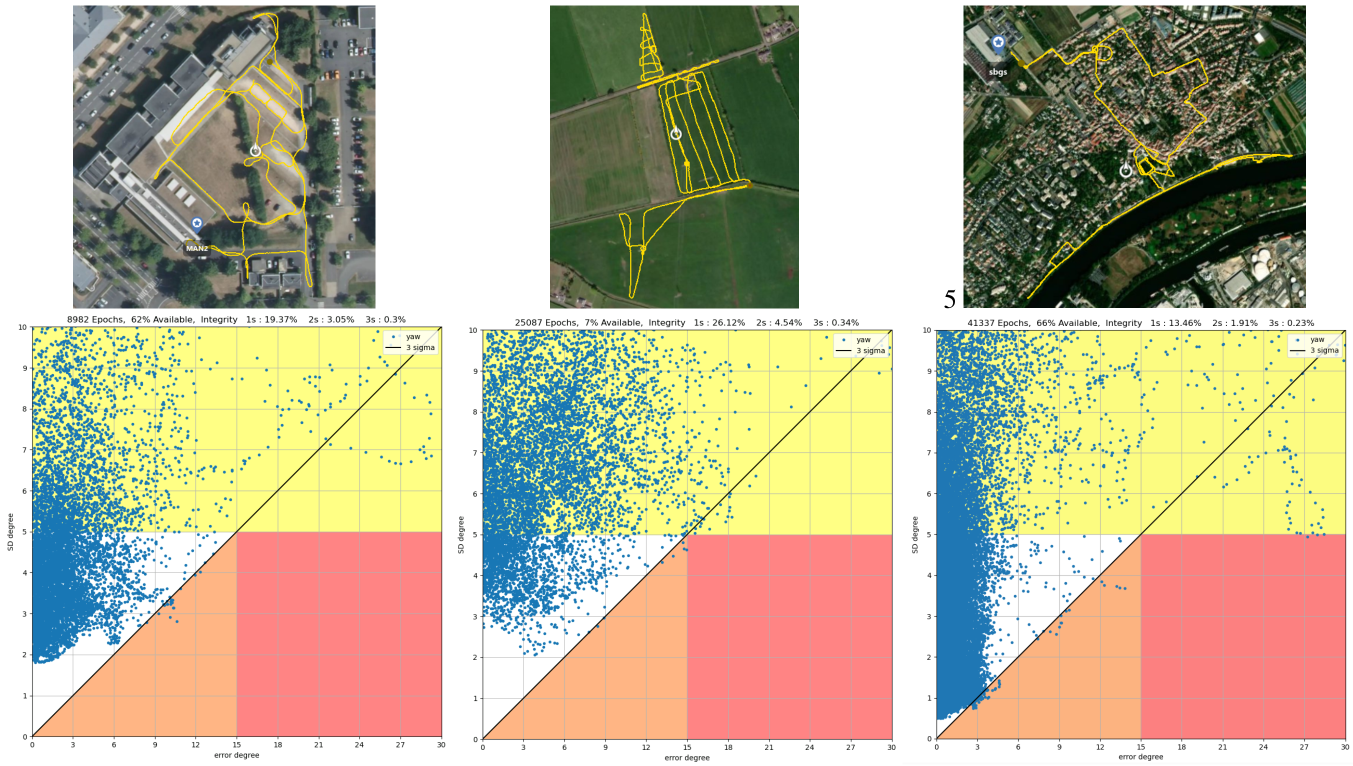


Figure 2: Left : Heading integrity of pedestrian DGPS initialization. GNSS accuracy was about 0.3m STD. Middle : Heading integrity of pedestrian SPP initialization. GNSS accuracy was about 1.0m STD. Right : Heading integrity of pedestrian RTK initialization. GNSS accuracy was about 0.01m STD.

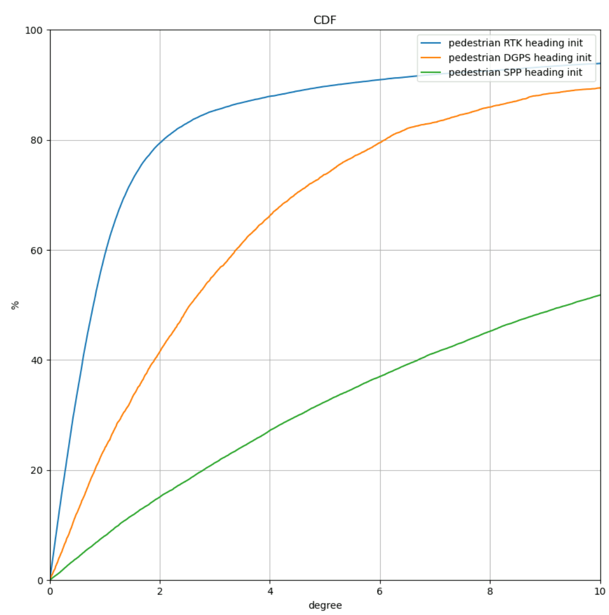


Figure 3: Cumulative distribution function of pedestrian heading alignment error with different GNSS quality scenario.

2. Full output analysis

We also provide the time serie of all other variable estimated during the initialization batch of the DGPS pedestrian log. Each axis of each estimated variables are plotted with a different color and the estimated STD $\times$ 3 are plotted grey.

On Figure 4 roll and pitch error are around 0.1°. This result is interesting because this is much lower that we can expect from a vertical gyrometer algorithm. The heading error is around 5° as presented above.

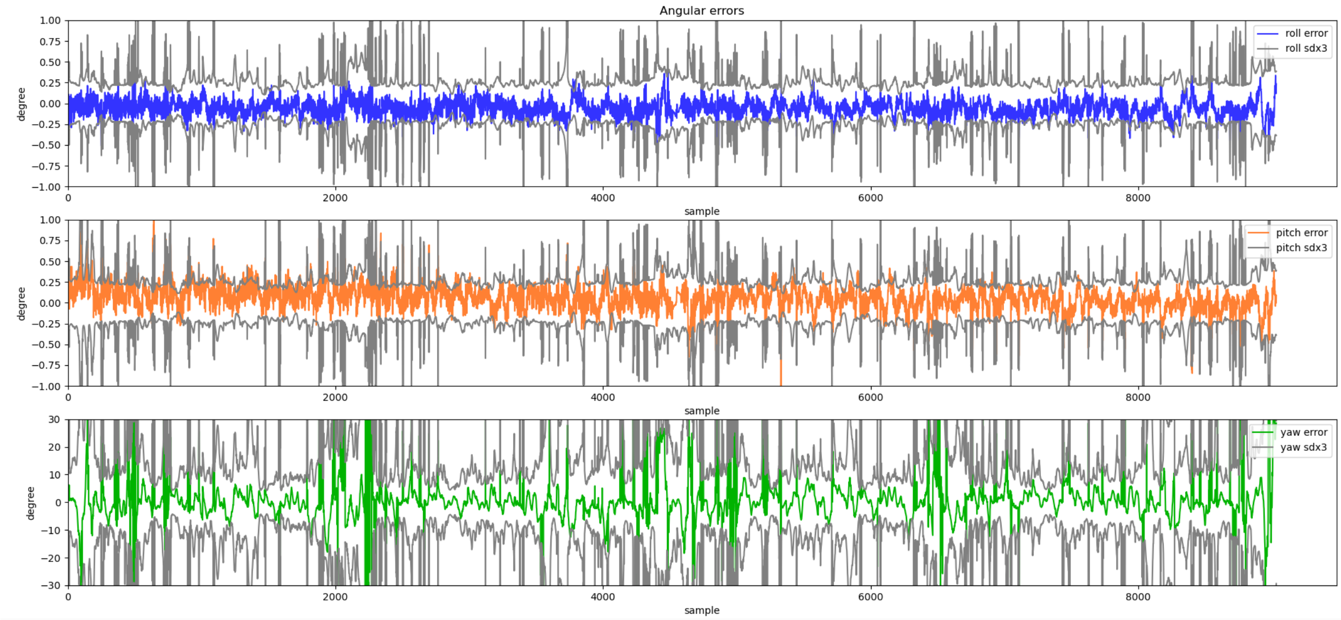


Figure 4: Attitude time series for the pedestrian DGPS log.

The velocity error is around 0.1 m/s. The standard deviation is well estimated for x and y. However the z velocity estimation is too optimistic. This can be explained by unmodelled random walk in the optimization system. To counter this effect, the output standard deviation can be artificially increased. Nevertheless, a performance of 0.1 m/s should be retained in this log.

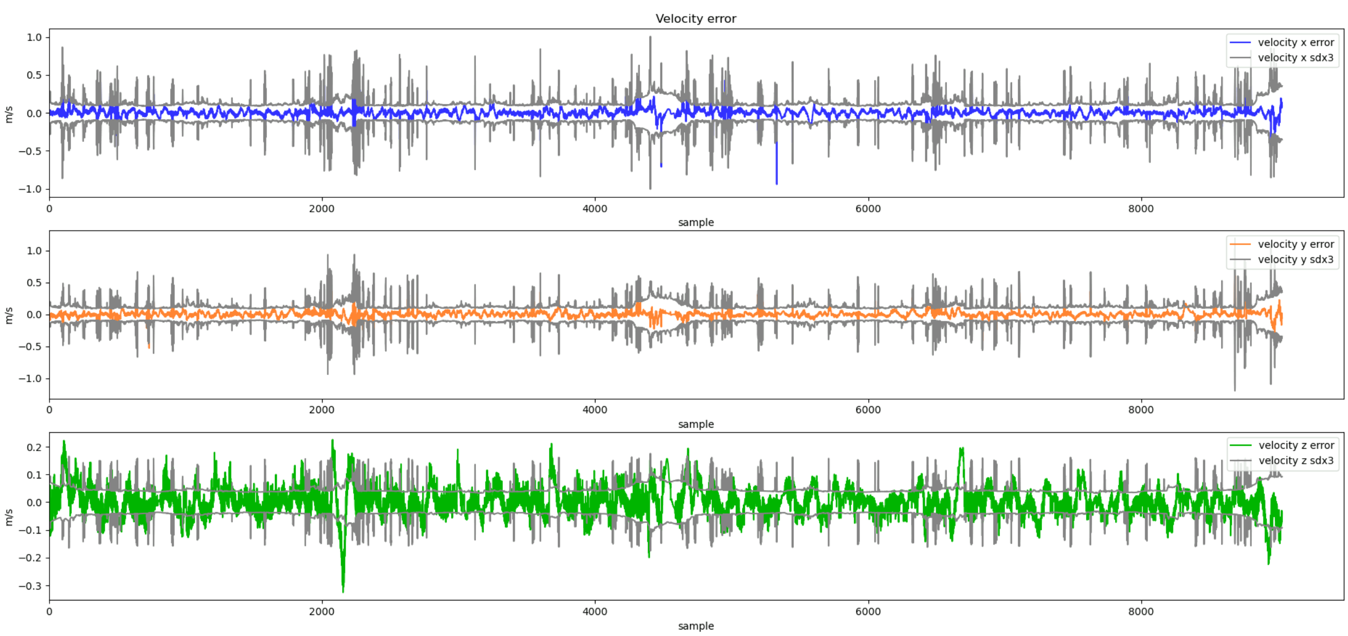


Figure 5: Velocity time series for the pedestrian DGPS log.

The gyrometer bias estimation is very anisotropic. The x and y axis are partially observed but the z axis is not observed and remain to its prior standard deviation of 150°/h, which comes from the pulse 40 specifications. During RTK log, this z bias can be observed. Depending of how is mounted the IMU, the z bias can be observed instead of others.

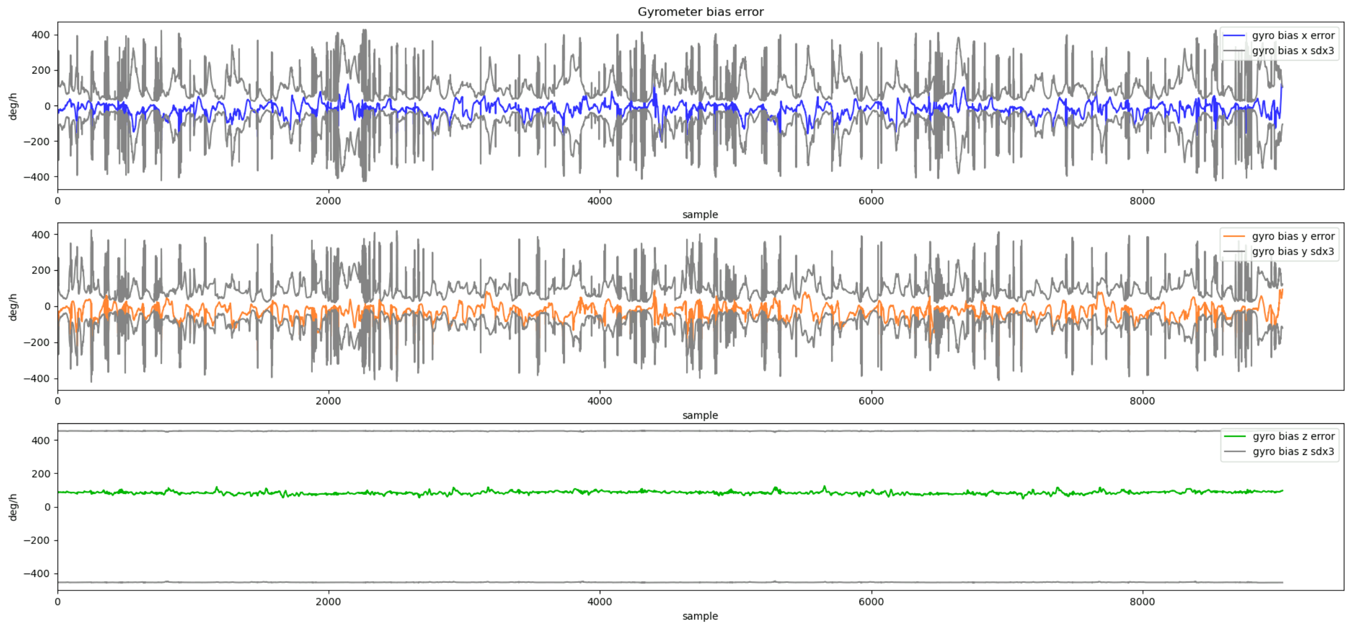


Figure 6: Gyrometer bias time series for the pedestrian DGPS log.

The accelerometer bias is not well observed. Only the z axis aligned with gravity is observed but the uncertainty is a bit optimistic. Again, this can be explained by unmodelled random walk in the optimization system and can be counter by artificially increasing the standard deviation of this accelerometer axis. Looking at these graph we can ask if the accelerometer bias estimation should be used in the initialization. The estimation does not seem to improve the initial uncertainty of 0.01 m/s/s given by the pulse 40 specifications. However these state should be kept in the system, because they are responsible for the roll and pitch uncertainty. Hence, the accelerometer bias remain important for the integrity of the system.

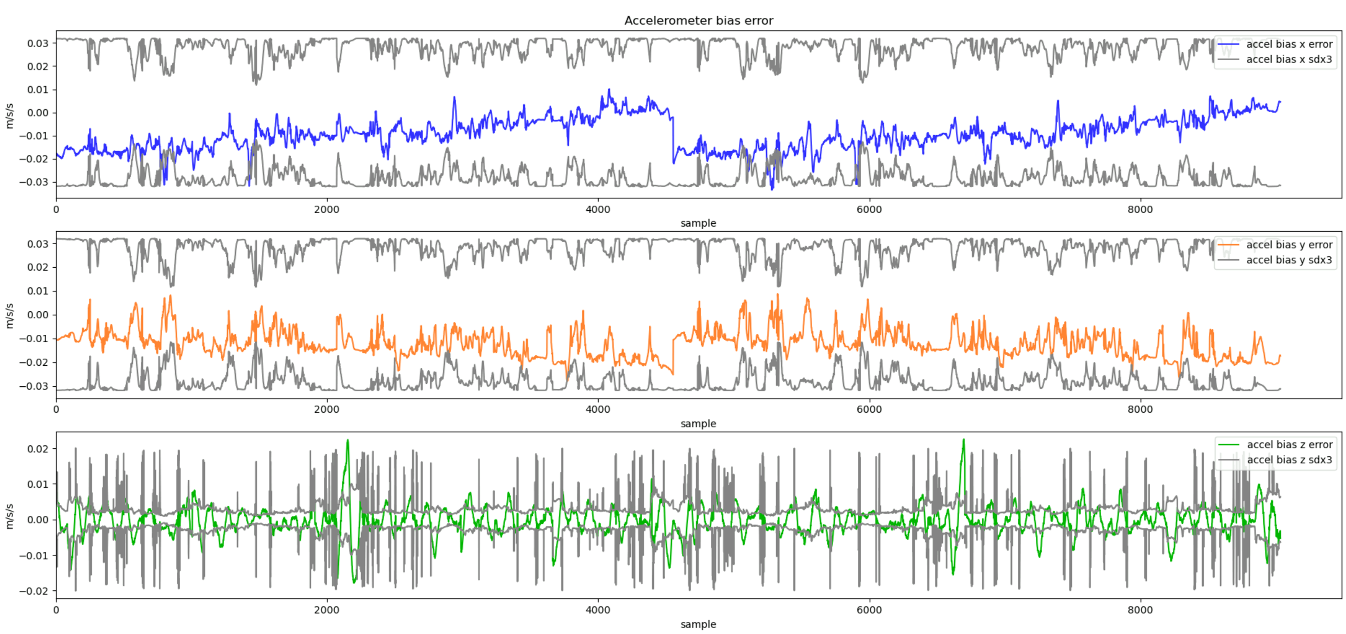


Figure 7: Accelerometer bias time series for the pedestrian DGPS log.

3. performance assessment of other applications

We also evaluated the alignment algorithm on other applications : automotive, marine and airborne to see how the system behave in these conditions.

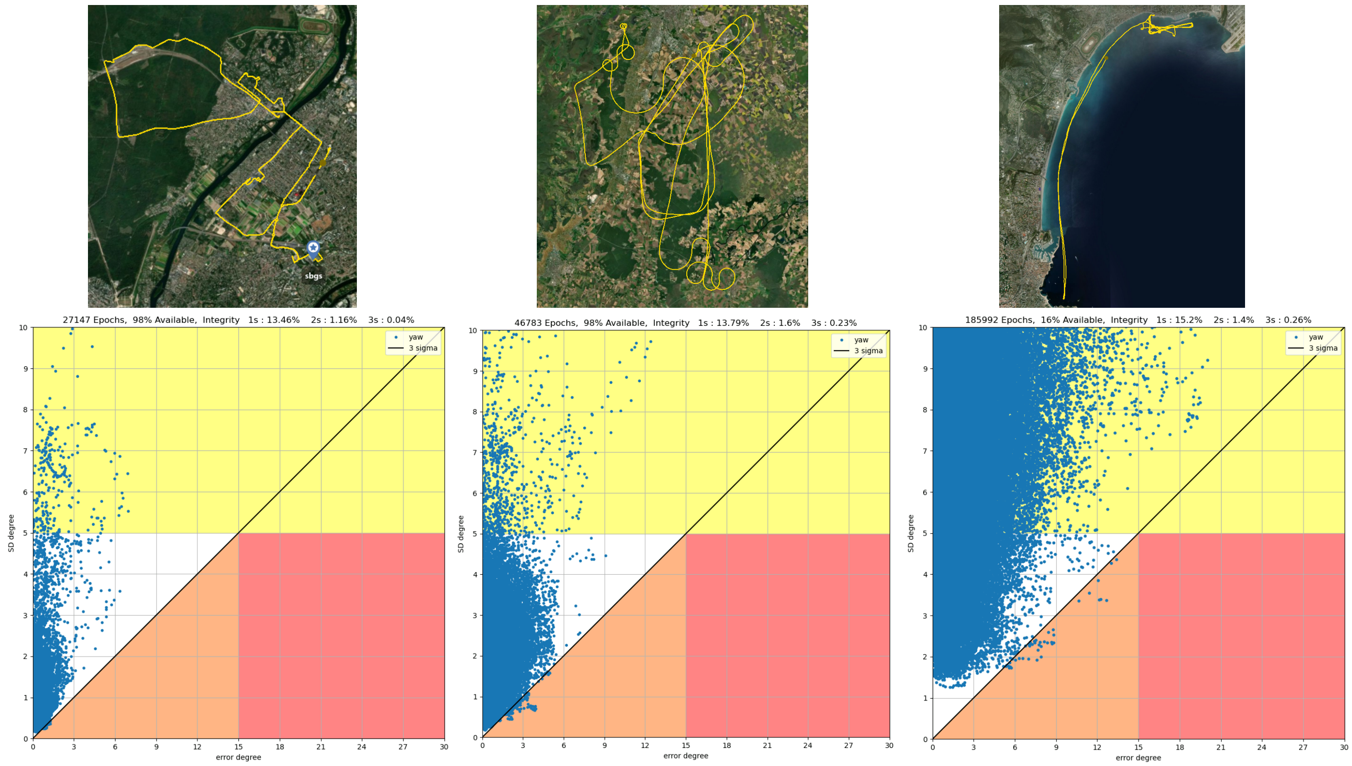


Figure 8: Left : Automotive RTK initialization. Middle : Airplane test with DGPS. Right : Marine test with an DGPS

As expected, the automotive scenario is a good use case for this alignment algorithm. The higher dynamics of a car eases the alignment, while the possibility of running in forward or backward makes it more robust than the traditional course alignment (which needs special attention for reverse detection).

Same comments can be made on the airplane scenario. The aircraft dynamics makes it easy to align the heading with a very good performance level, with no sensitivity to side slip.

However, on a ship with low dynamic the initialization is avalaible mostly if the ship is turning. its velocity was 2m/s in average, only two times more than the velocity of a pedestrian. The acceleration that undergoes a boat is lower, this explains that the boat initialize only upon turning in single antenna.

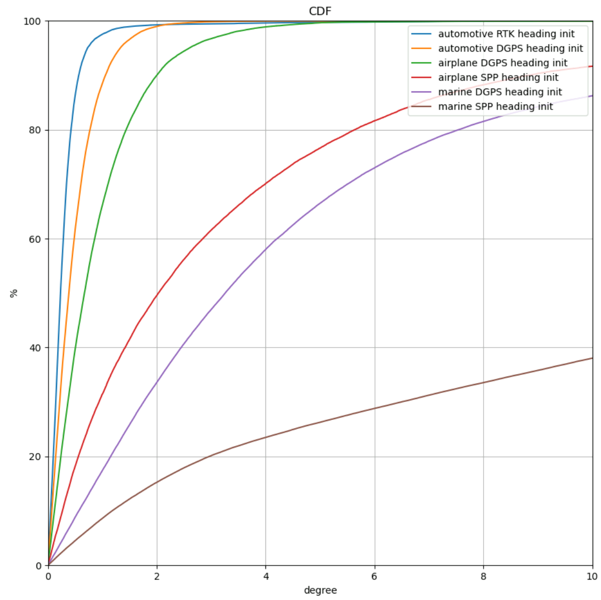


Figure 9: Cumulative distribution function of heading alignment error with different carrier and different GNSS quality.

VI. CONCLUSION

Our patent pending heading initialization algorithm has been demonstrated in both availability and integrity. It proves to be efficient in the most difficult use cases. Despite beeing the slowest, the pedestrian motion profile can be easier to initialize than a boat going very straight. Without surprise automotive, plane or drone have no difficulties to initialize.

Table 1: Performance summary of the initializer depending on the use case.

	Pedestrian			boat		airplane		car	
GNSS solution type	RTK	DGPS	SPP	DGPS	SPP	DGPS	SPP	RTK	DGPS
GNSS accuracy STD	0.01m	0.3m	1.0m	0.1m	0.4m	0.3m	2.0m	0.01m	0.1m
mean velocity	1.0m/s	1.0m/s	1.0m/s	2.0m/s	2.0m/s	70m/s	70m/s	10m/s	10m/s
availability[1]	66%	62%	7%	16%	11%	98%	56%	98%	99%
heading accuracy[2]	2.3°	3.3°	4.2°	3.6°	3.5°	1.9°	2.6°	0.7°	0.7°

[1] Rate of sample under 5° heading STD
[2] Mean estimated heading STD of the samples availables (ie. under 5° heading STD).

Not to forget that our initialization algorithm estimates many other variable as the roll, pitch, velocity, gyrometer bias and accelerometer bias. Typical accuracy of roll and pitch are about 0.5°. This is also an interesting achievement since the usual vertical gyroscopes algorithms used to initialize INS are usually of lower precision. The accuracy of the velocity is about 0.1 m/s. Accuracy of gyrometer bias and accelerometer bias is much more anisotropic. Only the accelerometer that measures gravity is observable, and while the gyrometer bias along the axis of gravity is not observable, the two other axes are observed. One might be tempted to remove some bias axis from the system to optimize, however, this would prevent the system to initialize if it is 45° tilted for instance. And sometimes, with harsh movment, all bias are observed.

The choice of a 5° standard deviation threshold on heading is arbitrary and can be reevaluated for final application needs. One can ask if 5° is too much for initiating an EKF. Many EKF do not support non linearity cause by big attitude uncertainty. But in our case our EKF was carefully design to support such non linearity without affecting operation in precise mode. Our initialization algorithm is now available for pedestrian motion profile in our post-processing software Qinertia. We plan to make it the default initialization procedure for all other carrier. We also plan to make it available for real time initialization.

REFERENCES

Angrisano, A., Vultaggio, M., Gaglione, S., and Crocetto, N. (2019). Pedestrian localization with pdr supplemented by gnss. In *2019 European Navigation Conference (ENC)*, pages 1–6. IEEE.

Bénet, P. and Guinamard, A. (2021). Deterministic visual mapping. In *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pages 3173–3188.

Benet, P., Saidani, M., and Guinamard, A. (2022). Tightly coupled inertial visual gnss solution-application to lidar mapping in harsh and denied gnss conditions. In *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, pages 1784–1794.

Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., and Tardós, J. D. (2021). Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*.

Chen, Q., Lin, H., Guo, R., and Niu, X. (2020). Rapid and accurate initial alignment of the low-cost mems imu chip dedicated for tilted rtk receiver. *GPS Solutions*, 24:1–13.

Fletcher, R. (2000). *Practical methods of optimization*. John Wiley & Sons.

Gade, K. (2016). The seven ways to find heading. *The Journal of Navigation*, 69(5):955–970.

Grassia, F. S. (1998). Practical parameterization of rotations using the exponential map. *Journal of graphics tools*, 3(3):29–48.

Saidani, M., Maya, D. G., and Guinamard, A. (2021). High integrity multi-sensor navigation system for safety critical applications. In *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pages 1767–1781.

Simon, D. (2006). *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons.

Strus, J. M., Kirkpatrick, M. R., and Sinko, J. W. (2009). Gradient-type minimization methods for initializing state variables in gps/ins integration. In *Proceedings of the 22nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2009)*, pages 177–185.

Theil, H. (1963). On the use of incomplete prior information in regression analysis. *Journal of the American Statistical Association*, 58(302):401–414.

Titterton, D. and Weston, J. L. (2004). *Strapdown inertial navigation technology*, volume 17. IET.

Von Stumberg, L., Usenko, V., and Cremers, D. (2018). Direct sparse visual-inertial odometry using dynamic marginalization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2510–2517. IEEE.

Zhang, Q., Li, S., Xu, Z., and Niu, X. (2020). Velocity-based optimization-based alignment (vboba) of low-end mems imu/gnss for low dynamic applications. *IEEE Sensors Journal*, 20(10):5527–5539.