Inertial navigation framework for multimodal underwater Graph SLAM

Pau Vial[1] pau.vial@udg.edu
Miguel Castillón[1] miguel.castillon@udg.edu
Narcís Palomeras[1] narcis.palomeras@udg.edu
Marc Carreras[1] marc.carreras@udg.edu
Pere Ridao[1] pere.ridao@udg.edu

OCEANS 2023 - Limerick | 979-8-3503-3226-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/OCEANSLimerick52467.2023.10244303

***Abstract*—Robot localization is a fundamental task in achieving true autonomy for Autonomous Underwater Vehicles (AUV). If inertial measurements from an Inertial Measurement Unit (IMU) or a Doppler Velocity Log (DVL) want to be fused with some perception system, such us a multibeam sonar or several acoustic beacons; a full Simultaneous Localization And Mapping (SLAM) problem must be solved. In contrast to filters, in a full SLAM problem the whole robot trajectory is estimated and loop closure events can be detected and closed along it. Common Inertial Navigation Systems (INS), based on filters, only maintain the estimation of the current robot pose. Therefore, these systems cannot be directly used in a full SLAM problem. In this paper we present a graph solution to integrate all inertial measurements in a factor graph that can be extended to different perception modalities and it is solved by applying Smoothing and Mapping (SAM) [1]. The Preintegrated IMU factor, proposed by [2], is combined with priors for other inertial measurements that have been specially designed. This framework is tested on real data from sea experiments, showing how our proposal performance is similar to the estimation provided by high grade commercial INS products based on filters. However, our system has the advantage of allowing for fusion with exteroceptive sensors in SLAM.**

***Index Terms*—Autonomous Underwater Vehicles, Simultaneous Localization And Mapping, Dead Reckoning, Inertial Navigation Systems, Lie Theory, Factor Graphs**

I. Introduction

There is a widespread tradition of basing Inertial Navigation Systems (INS) for robot localization on filters, for instance, the well known Extended Kalman Filter (EKF). This tendency is due to the computational and data efficiency of filters as they only maintain the estimation of the current pose of the robot. However, when exteroceptive sensors, such as optical cameras or range sensors, want to be integrated on the localization system, the whole robot trajectory must be estimated to be able to detect loop closure events along the trajectory. This problem is called full Simultaneous Localization and Mapping (SLAM), it is modeled using a factor graph [1] and it is solved applying Smoothing and Mapping (SAM), which means solving a least squares optimization. Nowadays, there are several algorithms to solve this kind of optimization online, such as `iSAM2` solver [3] (included in the GTSAM library [4]) or `SLAM++` solver [5].

[1] Computer Vision and Robotics Research Institute (VICOROB), University of Girona, 17003 Girona, Spain.

Work on this paper has been supported by the BITER-AUV Project under the Grant agreement PID2020-114732RB-C33 and by the Spanish Government FPU19/03638 PhD grant (to P. Vial).

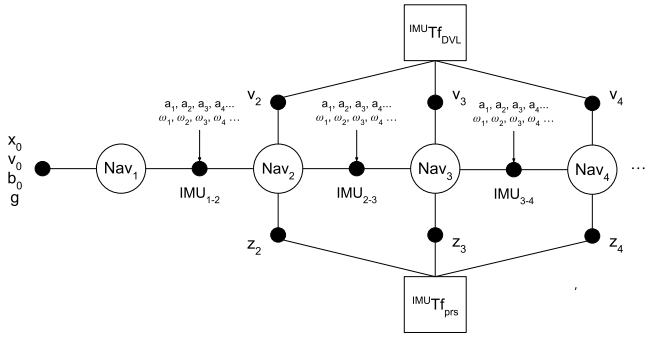


Fig. 1: Factor graph modelling the underwater inertial navigator solved applying SAM.

(a) (b)

Fig. 2: Girona 1000 AUV: (a) general view and (b) robot deployment at sea using Sextant boat from Universitat de Girona.

The filtering approach provides absolute robot pose and it is based on an initial condition from which integration of linear accelerations and angular velocities measured by an Inertial Measurement Unit (IMU) are accumulated. Bias and noise from sensors must be removed before integration and a precise initialization is crucial to avoid bad gravity compensation on the measured acceleration vector. When roboticists want to solve a full SLAM problem, increments of motion between perception key frames are more important than absolute pose. This is because wrong assumptions on bias or gravity compensation can be fixed in future solutions as all the robot trajectory is maintained. As it was suggested in [6], as the gravity vector is constant in the world frame for all key frames; accelerations can be directly integrated at the world frame without any compensation. Is at the time of building

the residuals to solve the graph, when the gravity acceleration should be removed according to the current available solution. This implies that while filter-based INSs usually need a few minutes to initialize, calibrating its bias and the direction of the gravity vector; a graph-based approach can avoid this calibration and start the estimation simultaneously with the mission. After setting a reduced number of key frames, depending on the applied sensor modalities, the problem will converge and an accurate and stable localization will be available. Moreover, the direction of the gravity vector in the world frame is not attached, but it is also estimated every time the graph is optimized. The same happens with sensor bias, that are estimated every time the problem is solved.

When solving a full SLAM problem, equally important as evaluating motion increments is to estimate their related uncertainty. For most commercial INSs the covariance matrix of the filter may not be fully available. Furthermore, this uncertainty may not be parameterized in the Special Euclidean group SE(3) [7], which is the de facto pose parameterization for SAM solvers.

In mapping applications of the sea bed or the water column, as well as in long term underwater applications, Autonomous Underwater Vehicles (AUV) are always equipped with an INS. In the underwater environment, where Global Navigation Satellite Systems (GNSS) are denied and visual odometry is not always guaranteed due to water turbidity, the IMU must be complemented with a Doppler Velocity Log (DVL) that measures AUV linear velocity. Also, a pressure sensor, providing robot depth, or a Ultra-Short Baseline (USBL) system mounted on a surface vehicle can be easily integrated to the INS filter. However, if several acoustic beacons or exteroceptive measures from optical cameras, imaging sonars, multibeam profiling sonars or laser scanners want to be integrated in the navigation system; a multimodal full SLAM problem must be solved. Therefore, filter-based INS cannot be used and SAM is the alternative. Moreover, SAM provides a graph representation suitable for sensor fusion when different perception modalities are considered.

In this paper we present an inertial framework for AUV navigation based on factor graphs. **The main contribution of this work is to provide a flexible graph-based approach able to seamlessly fuse a wide variety of perception sensor modalities with inertial sensors in a mathematically correct manner. Experimental underwater results on real inertial data show how our proposal matches with high grade commercial INS results while allowing for future integration with exteroceptive sensors.**

II. Methodology

To design an incremental inertial navigator using factor graphs suitable to be integrated in an underwater full SLAM problem, we propose to combine the Preintegrated IMU factor [2] with priors coming from other inertial sensors mounted on an AUV. Using the Preintegrated IMU factor we can synchronize the perception system, that typically runs at low frequencies from 0.1 to 10 Hz, with an IMU that runs at high frequencies from 100 to 1000 Hz. Following this approach, all IMU measurements are accumulated and only a node is set to the factor graph when the perception system sets a key frame, avoiding setting a node for each inertial measurement. This way, the graph growth is slowed down and it is possible to solve it on real time. Moreover, the Preintegrated IMU factor allows to update sensor bias and gravity vector direction at each solver call, avoiding to integrate all IMU measurements from scratch every time one of these parameters is updated.

To build the inertial navigator, a factor graph with the structure at Fig. 1 is build using the GTSAM library [4]. The state at each node in the graph is $X = ({}^{W}t_B, {}^{W}R_B, {}^{W}v_B, b_{acc}, b_{gyr})$ where ${}^{W}t_B$ is the AUV position in the world frame $\{W\}$, ${}^{W}R_B$ is the robot orientation in the world frame, ${}^{W}v_B$ is the linear velocity of the vehicle measured at the world frame and b_{acc} and b_{gyr} are respectively the bias of the accelerometers and gyroscopes. The robot velocity is included in the state as acceleration measurements imply a two step integration to get position, being velocity the intermediate step. Moreover, sensor bias are considered to evolve smoothly and are modeled as constant between two key frames. In the proposed approach, the body reference frame $\{B\}$ of the AUV is set at the IMU frame. As no dynamics of the vehicle are considered, we did not found any reason to move the AUV base link to the center of gravity of the vehicle. Therefore, accelerometers and gyroscopes measurements can be directly integrated without requiring any change of coordinate frame. A final consequence on this, is that the pose of the rest of sensors is directly calibrated against the IMU frame.

Common inertial sensors installed on AUVs, apart from an IMU, are a DVL providing AUV linear velocities and a Sound Velocity Sensor (SVS) providing water pressure and temperature. Combining pressure and temperature, the sound velocity in water can be found. However, the measured water pressure can also be directly related to AUV depth. To incorporate the information provided by the DVL and the pressure sensor to the inertial graph, as AUV velocity and depth are part of the state vector, a linear velocity prior and a pressure prior can be set at each key frame. Using these absolute inertial measures we can restrict nodes and bound navigation drift, allowing the estimation of IMU bias and the gravity direction. These prior factors have been specially designed for this application. Their expectation model and corresponding jacobians are presented at the following subsections.

A. Pressure prior

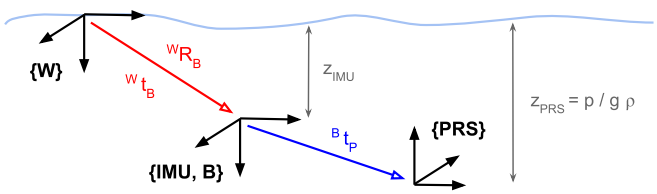


Fig. 3: Pressure prior diagram.

Mounting an underwater pressure sensor on an AUV, if the gravity vector is aligned with the $(0, 0, 1)$ vector of the world frame $\{W\}$ assuming a North East Down (NED) system; the depth of the AUV in the world frame can be measured. Given a pressure measurement p, the depth can be computed as

$$^{W}z_P = \frac{p}{\rho g},$$

where ρ is the water density and g is the gravity acceleration magnitude. As shown in Fig. 3, the body reference frame $\{B\}$ and the pressure sensor frame $\{PRS\}$ could not be aligned with the world frame and translations are present between these frames. Taking into account all these movements, the expectation model for the depth extracted from the pressure measurement is:

$$^{W}z_P = {}^{W}z_B + (^{W}R_B\,{}^{B}t_P)_z - b_{prs}$$

where $^{W}z_B$ and $^{W}R_B$ are respectively the depth and the rotation of the body frame in the world frame, $^{B}t_P$ is the translation between the body frame and the sensor frame, $(a)_z$ is an operator returning the third component of the vector a and b_{prs} is a geometrical constant bias between the pressure transducer and the sensor frame.

Taking into account that $^{W}z_B$ and $^{W}R_B$ are part of the state vector, the expectation model of the measured pressure expressed from the state vector is:

$$h(X) = \rho g \left[X.t()_z + (X.R()^{B}t_P)_z - b_{prs}\right].$$

Analysing this equation we see that it corresponds to the action on the SE(3) group between the robot pose and the vector $^{B}t_p$. Considering this mathematical fact and applying Appendix D from [7], the jacobian of the expectation model through the state vector is:

$$\frac{\partial h}{\partial t_B} = \rho g J_t^{Mq} \overset{SE(3)}{=} \rho g\, {}^{W}R_B,$$

$$\frac{\partial h}{\partial R} = \rho g J_R^{Mq} \overset{SE(3)}{=} -\rho g\, {}^{W}R_B[{}^{B}t_P]_\times,$$

where $[]_\times$ is the hat operator and the rest of jacobians are zero. As we are only considering the z component of the AUV position, only the last row of each jacobian is needed.

Finally, the position of the pressure sensor can be estimated by setting $^{P}t_B$ at the state vector. This translation vector must be considered constant for all key frames, as the sensor is mechanically attached to the AUV. Thus, the jacobian of the expectation model through this parameter is:

$$\frac{\partial h}{\partial t_P} = \rho g J_q^{Mq} \overset{SE(3)}{=} \rho g\, {}^{W}R_B.$$

B. Linear velocity prior

Using a DVL, we can measure the linear velocity of an AUV referenced at the sensor reference frame $\{DVL\}$. However, this frame can be rotated and translated from the body frame $\{B\}$, as it is shown in Fig. 4. As these two frames are static

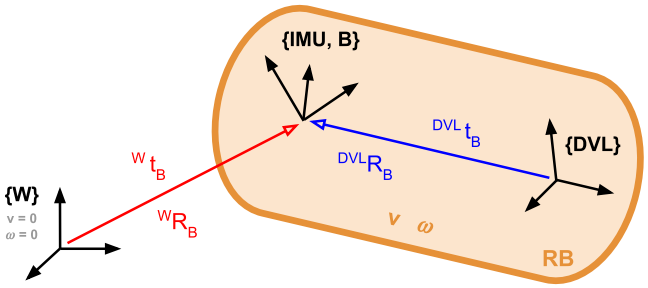


Fig. 4: Linear velocity prior diagram.

on a Rigid Body (RB) that is in a rototranslation motion, a linear velocity measured in the body frame ^{B}v can be mapped to the DVL frame applying:

$$^{DVL}v = {}^{DVL}R_B\,{}^{B}v + \left[{}^{DVL}t_B\right]_\times {}^{DVL}R_B\,{}^{B}\omega, \qquad (1)$$

where $^{DVL}R_B$ and $^{DVL}t_B$ are respectively the orientation and the position of the body frame in the DVL frame and $^{B}\omega$ is the angular velocity of the body measured at the body frame. If the body frame is placed at the IMU frame, as it is the case, the angular velocities $^{B}\omega$ are directly related to those measured by the gyroscopes.

However, we also need to map linear velocities expressed in the world frame ^{W}v to linear velocities expressed in the body frame ^{B}v, as in the problem state vector the AUV velocity is expressed in the world frame. As these two frames are not attached to the same RB in motion and we can consider the world frame as an inertial coordinate system, this map is $^{B}v = {}^{B}R_W\,{}^{W}v$ and equation 1 becomes

$$^{DVL}v = {}^{DVL}R_B(^{W}R_B)^{TW}v + \left[{}^{DVL}t_B\right]_\times {}^{DVL}R_B\,{}^{B}\omega,$$

where $^{W}R_B$ is the AUV orientation in the world frame.

Taking into account that $^{W}R_B$ and ^{W}v are part of the state vector and that gyroscopes measures $^{B}\overline{\omega}$ are affected by a bias b_{gyr} that is also part of the state vector, the expectation model of the measured linear velocities at DVL frame expressed from the state vector is:

$$\text{h(X)} = {}^{DVL}R_B X.R()^T X.v() + \left[{}^{DVL}t_B\right]_\times {}^{DVL}R_B(^{B}\overline{\omega} - X.b_{gyr}()).$$

Analysing this equation we see that the first term depends on the robot orientation and velocity and the second term only depends on the gyroscopes bias. Considering that velocities are manipulated in the Special Orthogonal group SO(3), applying Appendix B from [7] the jacobian of the expectation model through the state vector is:

$$\frac{\partial h}{\partial R} = {}^{DVL}R_B J_Q^{Qq} J_R^{R^{-1}} \overset{SO(3)}{=} {}^{DVL}R_B R^T[v]_\times R,$$

$$\frac{\partial h}{\partial v} = {}^{DVL}R_B J_q^{Qq} \overset{SO(3)}{=} {}^{DVL}R_B R^T,$$

$$\frac{\partial h}{\partial b_{gyr}} = -J_b^{a\times b} J_q^{Rq} \overset{SO(3)}{=} -\left[{}^{DVL}t_B\right]_\times {}^{DVL}R_B,$$

where the rest of jacobians are zero.

Finally, the pose of the DVL sensor in the body frame can be estimated by setting ${}^{DVL}R_B$ and ${}^{DVL}t_B$ at the state vector. This pose must be considered constant for all key frames, as the sensor is mechanically attached to the AUV. Thus, the jacobians of the expectation model through these parameters are:

$$\begin{aligned}\frac{\partial h}{\partial t_{DVL}} &= J_a^{a\times b} = -\left[{}^{DVL}R_B({}^B\overline{\omega} - X.b_{gyr}())\right]_\times,\\ \frac{\partial h}{\partial R_{DVL}} &= J_R^{Rq} + J_b^{a\times b}J_R^{Rq}\\ &\overset{SO(3)}{=} -{}^{DVL}R_B\left[X.R()^T X.v()\right]_\times - \left[{}^{DVL}t_B\right]_\times {}^{DVL}R_B\left[{}^B\overline{\omega} - X.b_{gyr}()\right]_\times.\end{aligned}$$

C. Inertial SAM algorithm

Combining the described priors with the Preintegrated IMU factor [2], a factor graph with the structure at Fig. 1 is build. Following Algorithm 1, a list of actions are repeated every time a new key frame is received. A key frame can be set at a temporal rate or, if some perception system exists, every time a relevant measurement is received. In an underwater environment this can happen every time an acoustic beacon signal is received, or every time a complete sonar scan is available. In the underwater domain, due to acoustics, this happens every several seconds.

Algorithm 1 Inertial SAM algorithm

```
pim = PreintegratedCombinedMeasurement( IMU )
pim.set_gravity( (0, 0, 1) )
svs = SoundVelocitySensor()
dvl = DopplerVelocityLog()
graph = NonlinearFactorGraph()
graph.set_node( 0 )
graph.set_prior( 0, x⃗_0 )
graph.set_seed( 0, seed )
i=1
while true do
    if key frame received then
        y_pim = pim.get_state_and_reset()
        graph.set_node( i )
        graph.set_combined_imu_factor( i-1, i, y_pim )
        graph.set_pressure_prior( i, svs.get_pressure() )
        graph.set_linear_velocity_prior( i, dvl.get_vel() )
        seed = graph.get_estimate(i-1) ⊕ y_pim
        graph.set_seed( i, seed )
        graph.solve()
        i++
    else
        pim.integrate_imu_measurement()
    end if
end while
```

Following Algorithm 1, between the reception of key frames the preintegration of IMU measurements is running. All accelerometers and gyroscopes measurements are accumulated - or preintegrated - in a single measurement assuming a constant bias on the sensors. When a key frame is received, the preintegration is stoped, the result is stored and a new preintegration is started. Using the preintegrated measurement, a Preintegrated IMU factor [2] is set between the current key frame and the previous one. Moreover, at the current key frame two priors are set. A pressure prior using pressure sensor measurements is applied and a linear velocity prior using DVL measurements is set. Therefore, a chained factor graph is build, where all trajectory nodes depend on the static pose of each inertial sensor on the AUV. If the perception system is able to detect loop closure events along the AUV trajectory, this chained graph can evolve to a looped graph. Finally, combining the last estimation for the previous key frame with the preintegrated measurement, a seed for the current key frame is provided to the solver. Using an incremental solver, such as `iSAM2` [3], the graph is solved at each key frame, reaching an online AUV navigation system estimating the whole robot trajectory.

III. Experimental setup

Sea experiments have been carried using the Girona 1000 AUV [8] (Fig 2). This AUV is a reconfigurable platform suitable for intervention and surveying that is equipped with several inertial sensors. Linear accelerations and angular velocities are provided by Phins Compact C3 [9] - from iXblue, Saint-Germain en Laye (France) -, an INS based on Fibre-Optic Gyroscopes (FOG) and Microelectromechanical Systems (MEMS) accelerometers that is used as an IMU by simply reading the raw sensor measurements. Linear velocity is measured by DVL1000-4000m [10] - from Nortek, Rud (Norway) - mounted at the lower AUV cylinder pointing the seabed. Water pressure is provided by miniSVS1000 [11] - from Valeport Ltd, Totnes (United Kingdom) - a SVS measuring water temperature, pressure and sound velocity. Finally, AUV position at surface is provided by L86 [12] - from Quectel, Shanghai (China) -, a compact GNSS mounted at the AUV antenna. All these inertial sensors are connected to the INS to run the commercial iXblue navigation filter [9]. Fig. 5 shows the spacial distribution of these sensors on the AUV and Table I provides the necessary dimensions.

${}^Bt_p$	$(-825.70, -27.25, 129.45)mm$
${}^{DVL}t_B$	$(-898.65, -377.25, -783.55)mm$
${}^{DVL}R_B$ (RPY)	$(0.0, 0.0, 180.0)°$

TABLE I: Girona 1000 AUV geometric parameters taken from CAD drawings.

Experiments have been carried in front of Sant Feliu de Guixols harbor (Girona, Catalonia) at shallow waters of 15-20 m depth with a flat seabed. In these experiments the AUV was deployed at surface, acting as a surface vehicle. By this set up, the DVL could measure the AUV linear velocity against the seafloor and the AUV antenna was not submerged, providing GNSS measurements during the whole mission. The GNSS was disabled on the INS filter in order to simulate an underwater performance. Therefore, these measurements do not

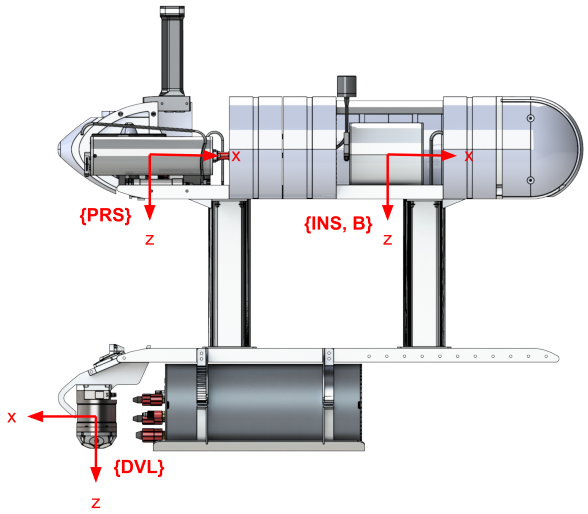


Fig. 5: Inertial sensors mounted on Girona 1000 AUV.

affect the INS filter and can be used as AUV position ground truth to benchmark against the proposed inertial navigator and the INS filter.

IMU rate	100.0 Hz
SVS rate	8.0 Hz
DVL rate	4.0 Hz
Navigator rate	0.4 Hz
Gravity acceleration g	9.80665 m/s^2
Water density ρ	1030 kg/m^3
Pressure sensor bias b_{prs}	35 mm

TABLE II: Navigator parameters

Finally, Table II shows the principal parameters used by the navigator. Note that, although the rates for the SVS and the DVL are given, pressure and linear velocity priors are set at navigator rate. Therefore, these values are only provided to show that these sensors are faster than the navigator and only the closest measurement to each prior is used. The rest of measurements are discarded. In Table III it is provided the parameterization for the covariance matrices of the factors applied to the navigator. The sensors standard deviation values (σ) have been characterized according to sensors data sheet.

IV. Results

The navigation framework is validated on a surface survey of one hour long covering an area of 150 $\times$ 70m describing corridors of 10m width. At the end of the survey, the AUV turned around the perimeter of the surveyed area (see Fig. 6).

Noise σ	
Accelerometer	1e-2 m/s^2
Gyroscope	1e-5 rad/s
Acceleration integration	1e-8
Accelerometers bias random walk	(1e-3, 1e-3, 1e-4) m/s^2
Gyroscopes bias random walk	(1e-4, 1e-4, 1e-5) rad/s
Bias	1e-5
Pressure prior	0.5 m
Sensor position prior	0.01 m
DVL prior	0.1 m/s
Sensor position prior	0.01 m
Sensor orientation prior	0.002 rad
AUV position prior	0.5 m
AUV orientation prior	0.01 rad
AUV linear velocity prior	0.1 m/s
Accelerometer bias prior	0.001 m/s^2
Gyroscope bias prior	0.001 rad/s

TABLE III: Dead reckoning noise standard deviation

The AUV moved at an averaged forward velocity of 0.5 m/s, always keeping at surface.

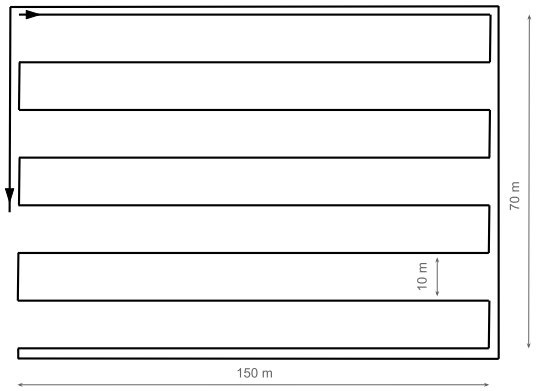


Fig. 6: Survey followed by Girona 1000 AUV.

Fig. 7 shows the top view of the spatial trajectory followed by the AUV. GNSS measurements (in blue) are considered as the robot ground truth. We plot in red the estimation provided by the commercial iXblue INS filter and the output of the proposed navigator is plotted in green and orange. The green trajectory corresponds to the current AUV pose estimation at each time stamp, as if the navigator was a filter, and the orange one corresponds to the hole AUV trajectory smoothed at the last key frame. We can see how the smoothed trajectory is more coherent with the ground truth, as it considers the hole history of measurements. As it can be seen, after one hour of survey both navigators provide trajectories spatially coherent, being the iXblue solution the most coherent one. However, this is because the AUV position control executes the survey based on this estimation. Therefore, this coherence on the iXblue solution does not mean that it provides the best estimation. To answer this, we need to check the positional error evolution, or drift, through time, shown at Fig. 8a. This error is computed against the GNSS measurements. We can see that our system behaves similarly to the commercial INS and bounds the positional error at the same magnitude order, reaching a maximum peak error of $8m$. Also, we see how the online and

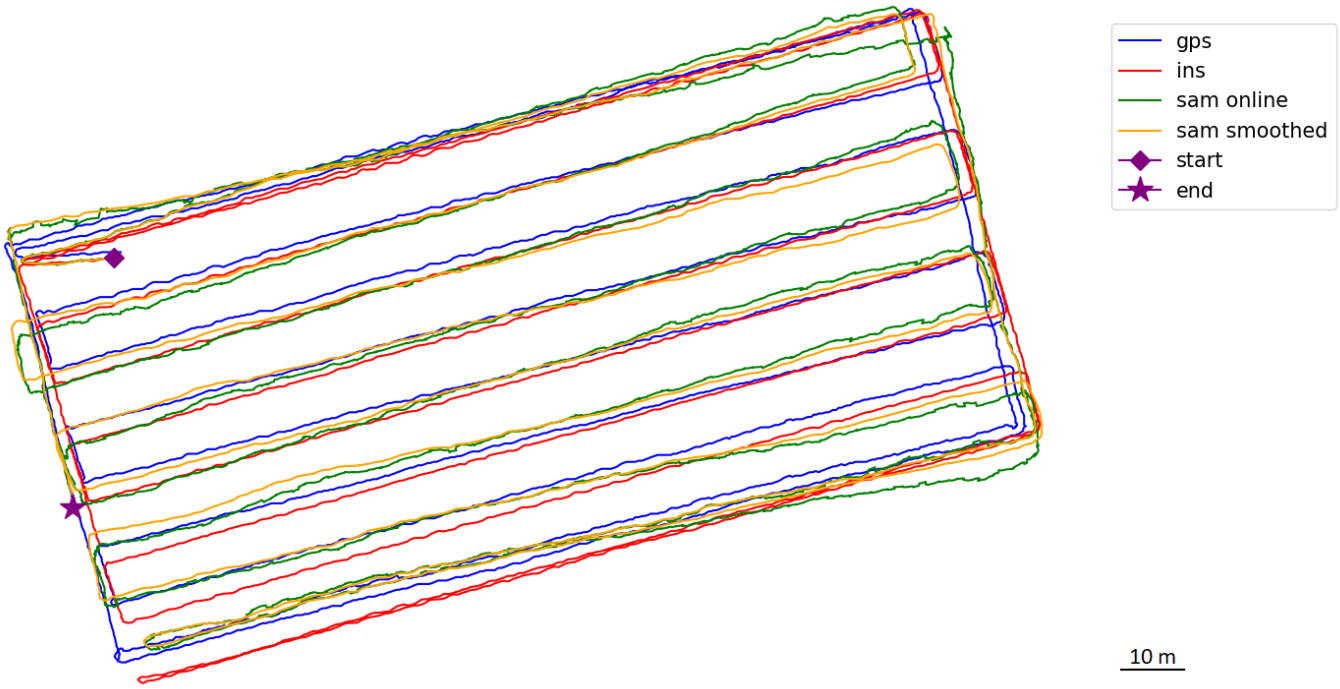
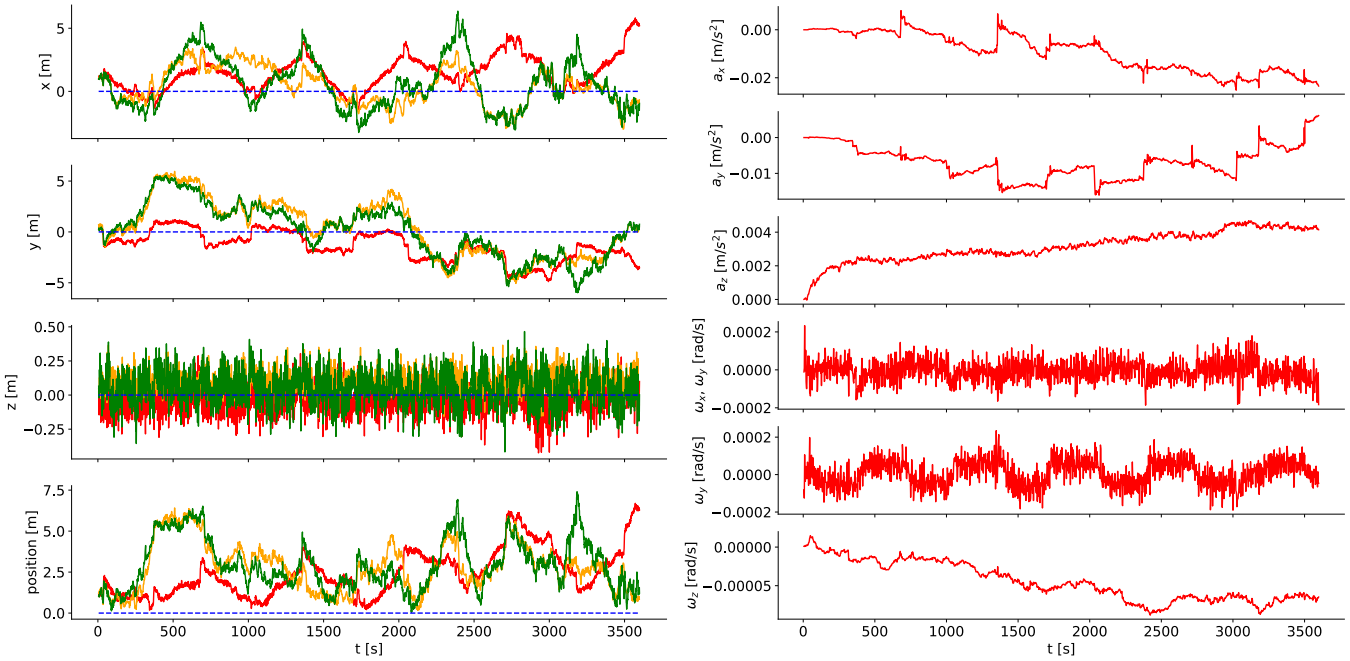


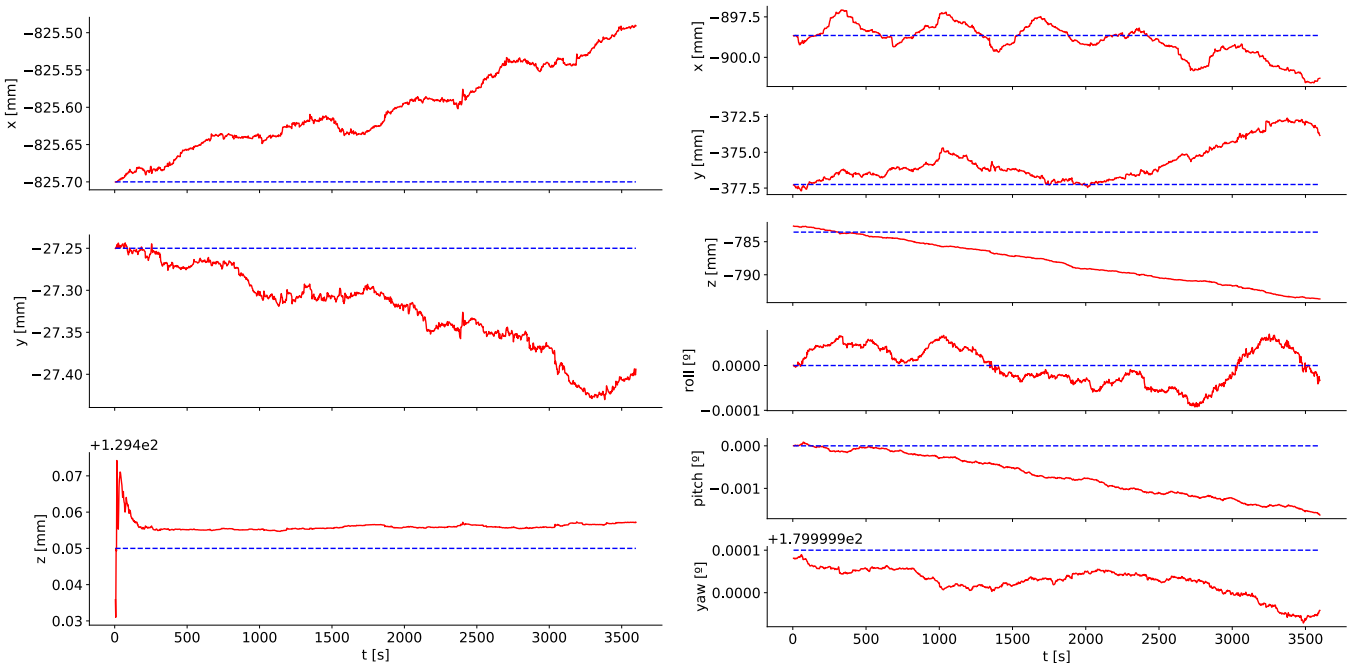
Fig. 7: Top view of the estimated trajectory followed by the AUV, where the starting point is marked by a diamond and the end point by a star. Blue: GNSS measurements considered as ground truth. Red: output of the iXblue INS filter [9]. Green: Online estimation provided by the proposed navigator at each time stamp. Orange: Smoothed trajectory at the last key frame estimated by our navigator.



(a) Position error

(b) IMU bias

Fig. 8: a) Estimated position error evolution through time computed against GNSS measurements following the aforementioned color convention. Error split up by components and, below, error norm. b) Accelerometers and gyroscopes bias estimation evolution through time.



(a) SVS position calibration

(b) DVL pose calibration

Fig. 9: Inertial sensors calibration evolution through time.

the smoothed solutions behave similarly, being the smoothed trajectory a bit better at the end of the mission. Finally, this figure shows that the z component of the positional error is bounded at $0.5m$ thanks to the information provided by the pressure priors. However, as any direct measurement is available in the planar directions, the error is around $10m$ in x and y directions. All estimations show this same behaviour on the drift.

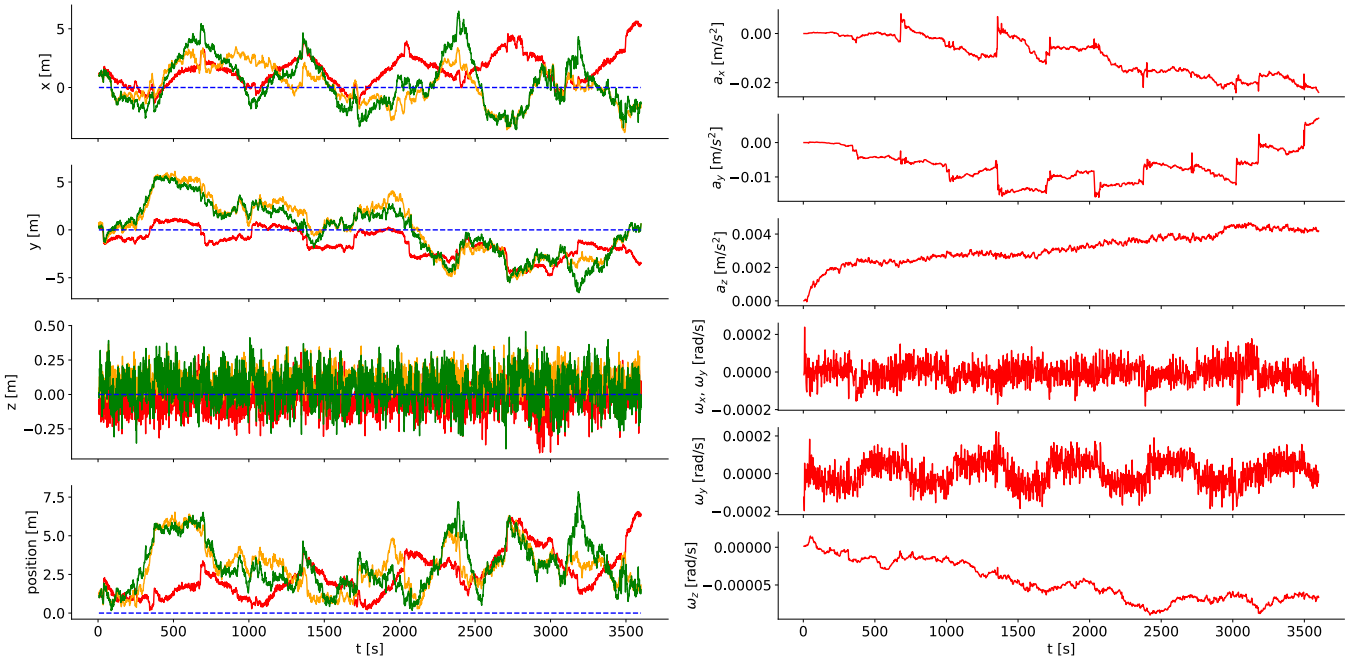
Fig. 8b show the accelerometers and gyroscopes bias estimation evolution through time. We can see that, as expected using FOG, the gyroscopes bias are very small. Analysing the bias evolution it is perceived a change of tendency at each transect of the survey, specially at the x and y directions. Fig. 9a shows the calibration of the SVS position. The continuous line shows the estimation, whereas the dashed line shows the design value extracted from draws. It can be seen how in the z direction the value is stabilized. However, the other directions show a tendency without reaching a steady state. For all directions the change is less than one millimeter, meaning that or the SVS is perfectly assembled or this calibration is not observable with the provided information. Finally, Fig. 9b shows the calibration of the DVL pose during the mission. Again, any stabilization is perceived in any translation or orientation. Moreover, any substantial change in orientation is observed. However, all translations evolve some millimeters but without reaching any steady state.

As calibration results are not concluding we propose to repeat the same experimentation without taking into account the calibration of the sensors pose, maintaining the design values as parameters. As the previous results suggested and as Fig. 10a and Fig. 10b show, no changes on the estimation are appreciated, reaching the same behaviour on the estimation. In conclusion, in this experimentation we showed how the proposed inertial navigator based on factor graphs reaches a performance comparable to a commercial INS based on filters. Inertial sensors pose calibration was proposed and derived but results show how this calibration is not relevant using only proprioceptive information.

V. Conclusions

In this paper we presented the inertial base for an underwater full SLAM system applying some acoustic perception system. We introduced an inertial factor graph combining measurements from accelerometers, gyroscopes, DVL and pressure sensor; that real experimentation showed how this system has a performance comparable to a high grade commercial INS based on filters. However, our system is already prepared to support exteroceptive information from other acoustic or optical devices. Following our approach, an online estimation is available and navigation uncertainty is estimated in the SE(3) group, the base parameterization for SAM solvers. Moreover, using factor graphs, no precise initial condition or IMU bias is need. Therefore, no initial calibration is required when turning on the robot and estimation can start simultaneously with AUV motion, as IMU bias, sensor poses and gravity vector direction are estimated by the graph.

Future work is to develop a preintgrated DVL factor to use all measurements provided by the DVL applying the same principle beyond the preintegrated IMU factor. By applying this methodology, no DVL measurements will be discarded and we expect to improve the system performance. Moreover,



(a) Position error

(b) IMU bias

Fig. 10: Same experiment but without using sensors pose calibration, considering its pose as a parameter. a) Estimated position error evolution through time computed against GNSS measurements following the aforementioned color convention. Error split up by components and, below, error norm. b) Accelerometers and gyroscopes bias estimation evolution through time.

more tests using cheaper and less precise IMUs must be done, applying gyroscopes based on MEMS instead of FOG. Finally, this inertial framework should be applied in a full SLAM problem in combination with several acoustic beacons or the registration of sonar scans gathered by a multibeam sonar mounted on a Pan&Tilt platform.

REFERENCES

[1] F. Dellaert and M. Kaess, *Factor Graphs for Robot Perception*. Now Publishers Inc., 2017.

[2] C. Foster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posterior estimation," *Georgia Institute of Technology*, 2015.

[3] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "isam2: Incremental smoothing and mapping using the bayes tree," *The International Journal of Robotics Research*, vol. 31, pp. 216–235, 2012.

[4] F. Dellaert. Gtsam 4.0: Factor graphs for sensor fusion in robotics. [Online]. Available: https://gtsam.org/

[5] V. Ila, L. Polok, M. Solony, and P. Svobodo, "Slam++: A highly efficient and temporally scalable incremental slam framework," *The International Journal of Robotics Research*, vol. 36, pp. 210–230, 2017.

[6] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in build environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, pp. 61–76, 2011.

[7] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *ArXiv*, vol. abs/1812.01537, 2020.

[8] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, "Girona 500 auv: From survey to intervention," *IEEE/ASME Transactions on mechatronics*, vol. 17, pp. 46–53, 2011.

[9] iXblue. Phins compact c3. [Online]. Available: https://www.ixblue.com/store/phins-compact-c3/

[10] Nortek. Dvl1000-4000m. [Online]. Available: https://www.nortekgroup.com/products/dvl1000-4000m/pdf

[11] Valeport. minisvs1000. [Online]. Available: https://www.valeport.co.uk/content/uploads/2022/06/Valeport-miniSVS-Datasheet-June-2022.pdf

[12] Quectel. L86. [Online]. Available: https://www.quectel.com/wp-content/uploads/pdfupload/Quectel$_L86_G NSS_S pecification_V 1.3.pdf$