A Smart multi-sensor system for marine animals with embedded geolocation

Pierre Gogendeau

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Système Automatiques et Microélectroniques (SYAM)

École doctorale : Information, Structures, Systèmes

Unité de recherche LIRMM / IFREMER

A smart multi-sensor system for marine animals with embedded geolocation

Présentée par Pierre GOGENDEAU

Le 29-11-2022

Sous la direction de Serge BERNARD
et co-encadrement de Sylvain BONHOMMEAU et Hassen FOURATI

Devant le jury composé de

Christophe GUINET, Directeur de recherche, CEBC-CNRS	Président du jury
Sylvain BONHOMMEAU, Chargé de recherche, IFREMER	Membre du jury
Hassen FOURATI, Maître de conférences, GIPSA-lab	Membre du jury
Serge BERNARD, Directeur de recherche, LIRMM-CNRS	Directeur de Thèse
Gabriel Hugh ELKAIM, Professeur, UC Santa-Cruz	Rapporteur
Mark Peter JOHNSON, Maître de conférences, Aarhus University	Rapporteur

UNIVERSITÉ DE MONTPELLIER

University of Montpellier

Doctoral Thesis

A Smart Multi-Sensor System for Marine Animals with Embedded Geolocation

Author:
Pierre GOGENDEAU

Supervisors:
Dr. Serge Bernard, Dr. Sylvain Bonhommeau and Dr. Hassen Fourati

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

SYAM - I2S
LIRMM - IFREMER

February 13, 2023

"*A smooth sea never made a skilled sailor*"

Acknowledgements

Three years later, this thesis is finished. I wanted to thank all the people who helped me. The task is not easy, so I will try to keep it simple.

I would like to make a special mention to Serge Bernard and Sylvain Bonhommeau, my two thesis supervisors. Without them, nothing would have been possible. Since my arrival in Reunion 5 years ago, you have tried to get the best out of me (not without difficulty). I don't know how to thank you enough for all these opportunities, for always believing in me, even in the craziest ideas. For all the time you devoted to me, all the debates, discussions and especially all the support you gave me. There is still some work to do, but I think I've evolved in the right direction thanks to you. I hope one day to have the chance to work with you again. It was an enriching experience and in a few months, I will have only good memories.

A mention also to Hassen Fourati for his guidance in my researches. Thank you for your precious advices, your corrections, and your support.

Thanks to Anne Renault for believing in me and allowing me to do this thesis.

I would like to express my gratitude to all the colleagues of IFREMER with whom I shared wonderful moments. The list would be too long to detail here, so we will take the time to give personal thanks when I see you! Your good mood, your jokes and your professionalism made the Reunion station an ideal working environment. Thank you for having supported me during these years, especially during this last year of thesis when I was less available.

Thanks to the LIRMM team where I was always welcomed very warmly. It has always been a pleasure to work and spend time with you.

A big thank you to the teams of Kelonia, ID Ocean and TAAF that I have been able to work with during my time in Reunion Island.

For those who know me, you know how much friends mean to me. I can't thank you enough for your love and support. Thank you so much for being patient and taking care of me. We will soon be able to make up for lost time.

I especially thank Guillaume, Damien, and Tristan for waiting for me and for all the work you did in my absence. I know I will be in debt for a long time, but I think it's worth the price.

Finally, thank you to my family for your unconditional support and love. Without you, I wouldn't have gone very far.

5 years after arriving at Sylvain's couch, the step in Reunion Island ends, and not without emotions. Thank you all for these unforgettable moments!

Contents

List of Figures

List of Tables

To my friends,
for their patience and their faith

Résumé

La géolocalisation des espèces sauvages est une source d'information essentielle pour la recherche en écologie et pour la gestion et la conservation des espèces. Dans l'environnement marin, l'acquisition des trajectoires géolocalisées sous-marines est une tâche complexe. Une des premières raisons est qu'il n'existe pas de système global de géolocalisation sous-marine à l'instar des constellations GNSS du domaine terrestre. Une autre raison est la difficulté à équiper les animaux de systèmes permettant leur suivi qui ne doivent pas perturber le comportement naturel des animaux marqués et donc doivent être les plus petites possible. Ainsi, se pose alors de véritables défis pour obtenir des autonomies suffisantes, intégrer les capteurs et les capacités de calcul et pouvoir transmettre les données acquises.

Avec ce type de balises, la solution couramment utilisée pour estimer les trajectoires à fine échelle spatiale et temporelle est la navigation à l'estime (Dead Reckoning : DR). Cette méthode utilise la vitesse et le cap à chaque pas de temps pour estimer la position suivante. Par son côté itératif et sa sensibilité aux éléments externes, la méthode peut présenter des dérives importantes dues à l'accumulation des erreurs de mesure et de calcul. Dans ce contexte, beaucoup de travaux scientifiques tentent d'améliorer les algorithmes utilisés ou cherchent à fusionner les données de nouveaux capteurs pour compenser les dérives.

De nombreuses applications de cette méthode ont été développées pour les animaux marins à l'aide de bio-logger, des équipements permettant d'enregistrer les données mais qui nécessitent la recapture de l'individu. Les biologgers sont à distinguer des bio-télémètres qui comportent un système de transmission des données (satellite, GSM, WiFi, BLE, ...) permettant d'envoyer des messages lorsqu'ils sont à la surface de l'eau. Dans l'état actuel des évolutions technologiques, aucun système ne permet d'envoyer les trajectoires géolocalisées à fine échelle temporelle. En effet, les systèmes sont soit limités par leur portée ou par leurs débits.

Nos travaux ont été initiés dans le cadre du projet IOT (India Ocean sea Turtle). Ce projet vise à mieux **comprendre les habitats fonctionnels** (ex : alimentation, repos) des tortues vertes (*Chelonia mydas* et imbriquées *Eretmochelys imbricata*) juvéniles dans le sud-ouest de l'océan Indien. Les principaux objectifs techniques du projet sur lesquels j'ai pu travailler deux ans en tant qu'ingénieurs sont :

-Une balise basse consommation intégrant un accéléromètre 3 axes, un magnétomètre 3 axes, deux capteurs de pression, un GPS, un microcontrôleur basse consommation programmable en Arduino et un module de transmission LoRa.

-La création d'un réseau de stations de réception pour la transmission LoRa et le stockage des données sur un serveur.

Les premiers résultats nous ont permis de transmettre sur des périodes de plusieurs mois des informations sur les profils de profondeur et des informations sur les plongées. Suite à ces premiers résultats, il nous a semblé que notre balise était

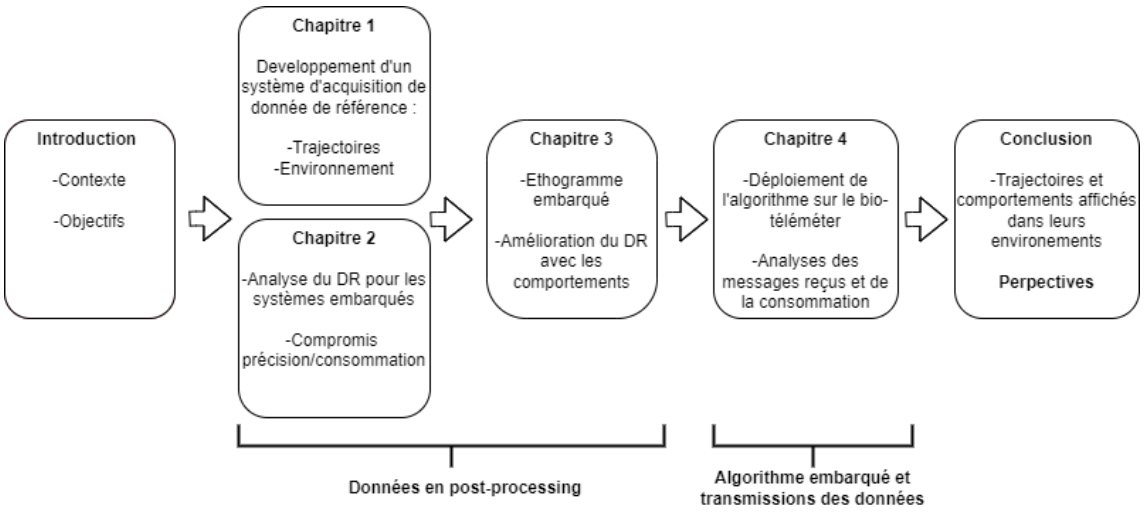


FIGURE 1: Schéma de synthèse des objectives de la thèse

sous-exploitée et nous nous sommes posé la question : Est-ce qu'il serait possible d'estimer des trajectoires sur de longues durées à l'aide du Dead-Reckoning et de les transmettre sur de longues durées ? Quelle serait le niveau de précision et l'autonomie des balises ?

Tout ce travail préparatoire de développement, de déploiement, de test et de validation a permis de lever les premières barrières techniques en termes de taille, de prix, de capacité de transmission et de stockage. Lors de cette thèse qui a suivi ce travail d'ingénieur, le défi est de développer et appliquer une méthode pour surmonter les différentes contraintes qui pèsent sur notre capacité à estimer et à transmettre les trajectoires à fine échelle spatiale et temporelle sur plusieurs mois. Le but est de proposer des solutions ouvertes et abordables en termes de coût et complexité pour être adaptées et utilisées par d'autres chercheurs.

En parallèle, nous avons développé un ASV (Autonomous Surface Vehicle) utilisé comme plateforme d'acquisition des environnements d'études. L'ASV est équipé d'un échosondeur mono-faisceau pour mesurer la bathymétrie et d'une caméra pour la photogrammétrie des petits fonds. Il permet l'acquisition de données environnementales et la résolution par exemple de problématiques liées à l'occupation de zones d'habitats ou de nourrissages en les associant aux données de trajectoire. L'ASV peut être utilisé dans les zones isolées sans station de correction du GPS en installant sa propre station. Il est entièrement open-source et a été développé pour être reproductible facilement et être utilisé comme une plateforme multimodale par des scientifiques.

La thèse est constituée de quatre chapitres (Voir Figure 1) qui détaillent les différentes étapes qui ont permis d'aboutir à une solution embarquée pour transmettre les trajectoires des tortues marines sur de longues périodes.

Afin d'étudier les capacités d'estimation de trajectoire de la balise IOT en termes de précision et de consommation, nous avons besoin de **trajectoires de référence** pour les comparer à celles estimées. Mais dans l'environnement marin l'acquisition de ce type de trajectoires de référence demeure problématique avec les contraintes liées à la transmission des ondes radios.

Le positionnement acoustique sous-marin à l'aide de Short Base Line (SBL)

et d'Ultra Short Base Line (USBL) est adapté en termes de précision (jusqu'au centimètre), mais les technologies sont limitées par leurs portées d'une centaine de mètres. Pour être capable de suivre un animal marin se déplaçant sur de longues distances, un système SBL est monté sur un véhicule de surface autonome (ASV). Nous avons développé et adapté le pilote automatique de l'ASV afin d'être en mesure de suivre un individu équipé d'une balise acoustique et d'acquérir des trajectoires de référence pendant au minimum 5 h avec une précision spatiale de 1 m.

Comme nous l'avons introduit, l'estimation de la trajectoire est possible grâce au DR et les informations de différents capteurs pour estimer l'orientation et la vitesse. Dans le biologging, cette méthode est appliquée essentiellement en post-traitement et n'est pas forcément adaptée au calcul embarqué. Dans cette partie, il a été **défini le contexte logiciel et matériel pour l'utilisation du DR** en considérant les ressources limitées disponibles des bio-télémètres.

Le but est de tester plusieurs fréquences d'acquisitions et l'activation de différents capteurs pour étudier l'influence sur la précision des trajectoires et la consommation électrique. Pour déterminer l'attitude (orientation) de l'animal, nous proposons deux algorithmes. Le premier composé d'un magnétomètre et d'un accéléromètre et le second avec un gyroscope supplémentaire. Pour la vitesse, la mesure est proposée avec un capteur composé d'une turbine et d'un capteur à effet hall ou par son estimation à l'aide des données de l'accéléromètre. Dans chaque cas, nous testons des fréquences d'acquisition pour 100 Hz et 10 Hz.

Nos tests pour la précision des trajectoires s'effectuent dans un environnement contrôlé avec un bio-logger multicapteur pour les données brutes à analyser (accéléromètre, magnétomètre, gyroscope, pression, et capteur de vitesse). Un nageur simule la nage d'une tortue en surface, ce qui nous permet d'utiliser un système de référence avec double système GPS Real Time Kinematic (RTK). Ce système permet d'acquérir des trajectoires de référence avec une précision de 2 cm sur la position et de 1,6° sur le cap. Les tests de consommation énergétique sont effectués en laboratoire sur la carte "Turtle tracker", développée dans le projet IOT, qui est le bio-télémètre cible de l'étude. Nous mesurons et comparons la consommation moyenne des capteurs pour chaque configuration.

Durant l'analyse, nos recherches montrent que dans ce cas d'application, l'estimation de la vitesse est plus impactante que celle du cap sur la précision globale. Pour ce dernier, l'utilisation d'un algorithme demandant de faibles ressources de calcul avec seulement les données d'un accéléromètre et d'un magnétomètre est suffisante sans l'ajout du gyroscope. Par ailleurs, l'utilisation de fréquences de sampling élevées (supérieures à 10 Hz) n'est pas nécessaire. Les résultats montrent que l'utilisation d'un capteur de vitesse adapté et la correction du courant marin, même approximative, nous procurent le meilleur gain de précision. La composante de la vitesse estimée ou mesurée avec un capteur dédié reste très bruitée par les perturbations externes. Nous concluons qu'il est important d'avoir une estimation de la vitesse adaptative pour améliorer sa précision.

Nous avons alors émis l'hypothèse que **l'étude comportementale pourrait aider à améliorer les trajectoires ainsi qu'aider à réduire les calculs et la consommation**. La méthode proposée utilise un éthogramme avec une estimation adaptative de la trajectoire en fonction des phases comportementales de l'animal. Nous comparons la précision, la consommation et la mémoire de la méthode proposée

à des méthodes non adaptatives. Pour acquérir nos données d'études, nous avons utilisé l'ASV développé pour le suivi des trajectoires d'une tortue juvénile. Ce test permet en parallèle de valider la fonction de suivi sur un animal sauvage.

La première étape a été de développer un éthogramme simple et adapté pour être embarqué sur les bio-télémètres. La méthode utilise la correspondance de certaines actions définies et l'information de l'état précédent pour les transitions de comportement. Elle a l'avantage d'être flexible temporellement et ne demande presque aucun calcul. Elle nécessite cependant une connaissance ou des données préalables sur l'animal étudié. Le but de l'éthogramme est de déterminer les comportements avec des dynamiques différentes. Certains comportements sont définis en fonction de la particularité de la nage et des techniques d'estimations. Par exemple, nous recherchons les phases en contact avec le sol (lorsque la tortue se nourrit ou se repose) pour fixer la vitesse à 0.

Différentes fonctions d'estimation de vitesse sont proposées pour les méthodes adaptatives. L'algorithme alterne entre une fonction d'estimation lorsque les nages sont régulières, lors des phases avec un tangage (ou pitch) élevé et avec des vitesses fixes.

Pour chaque algorithme testé, nous estimons les mesures de consommation globale induite par la reconstruction de trajectoire.

Les algorithmes proposés utilisant l'éthogramme montrent une meilleure précision et une plus faible consommation d'énergie que les techniques non adaptatives. Par rapport à la meilleure méthode non adaptative, l'estimation de la vitesse montre une amélioration entre 15,4% à 10 Hz et 7% à 1 Hz et la précision de la trajectoire entre 72,7% et 30,3%. Elles présentent une consommation moyenne optimisée de 9,7% et 13,6%.

Cette méthode propose une solution simple, peu consommatrice d'énergie et adaptable pour calculer des trajectoires embarquées tout en fournissant des informations sur le comportement. Les méthodes proposées sont développées pour être facilement remplacées par des éthogrammes ou des capteurs de vitesse déjà développés par les scientifiques.

L'objectif du chapitre 5 est de d'intégrer les développements algorithmiques basé sur un ethogramme dans un bio-télémètre et intégrant la compression des données, la limitation des calculs et la réduction de la consommation en énergie afin de réaliser des transmissions de données sur de longues durées.

La structure globale de l'algorithme est décrite ainsi que les méthodes de calcul utilisées. La structure s'articule autour des comportements de l'animal et utilise les surfaçages pour découper les trajectoires. Suivant les comportements, l'algorithme peut avoir des modes de fonctionnements différents. Dans notre cas, lorsque nous détectons une phase au sol, seule l'accélération est utilisée ce qui permet de limiter la consommation. Lorsque l'animal arrive à la surface, les messages sont compressés et envoyés. Suivant les besoins des utilisateurs et ces moyens d'analyse en post-traitement, il est proposé deux modes de transmission des trajectoires avec les positions en format géographique ou dans le repère North-East-Down (NED) avec une position géographique ancrée. Les différentes trames de messages sont décrites pour comprendre les nombres de positions et les informations qu'il est possible de transmettre. Pour chaque message, la balise peut envoyer 16 positions en mode géodésique ou 20 en mode NED.

Pour valider la solution, différents axes sont étudiés. Dans la partie de compression de trajectoires, nous sélectionnons deux méthodes adaptées au contexte d'acquisition que nous comparons en simulation avec les données de trajectoire étudiées à la partie précédente. La première méthode est une découpe simple en fonction du nombre de positions à transmettre et de la durée de la plongée. La seconde utilise un algorithme de compression plus complexe qui cherche des points d'inflexions sur la trajectoire pour minimiser la perte de précision. On remarque que pour les plongées d'une durée inférieure à 900 secondes, les précisions des méthodes sont proches et ne nécessitent pas l'utilisation de la méthode complexe. La méthode de découpe fixée est particulièrement mal adaptée pour les plongées avec de longues phases de repos.

Avec l'algorithme embarqué sur le bio-télémètre, les mesures de consommation sont effectuées directement et donnent les valeurs réelles pour son fonctionnement. Avec une consommation de 0,39 mA, la balise peut collecter et envoyer des données pendant 138 jours avec une batterie de 1300 mAh.

Enfin, des tests sont effectués en milieux contrôlés avec un bio-télémètre et un nageur simulant la nage d'une tortue. Différents modes d'affichage sont présentés allant de l'hébergement de serveurs personnalisés aux services web en ligne et gratuits. Ils permettent aux utilisateurs avec différents niveaux de connaissance en programmation de recevoir et d'afficher les données.

Les développements proposés fournissent une solution à faible coût et sur du long terme pour suivre le mouvement à petite échelle et le comportement associé des animaux marins. La quantité de données et leur résolution collectées et transférées ouvrent de nouvelles questions de recherche en écologie marine.

Plusieurs perspectives intéressantes découlent de ces travaux. Pour aider à caractériser les habitats fonctionnels, les trajectoires peuvent être contextualisées avec leurs environnements. La superposition de plusieurs couches acquises par différentes sources (ASV, anciennes missions ou données publiques) permet une analyse affinée des habitats fonctionnels et de mieux comprendre l'utilisation de l'espace et de ses interactions avec les espèces suivies.

Lors de l'analyse en post-traitement des trajectoires avec les données de bathymétrie, plusieurs problèmes ont été identifiés. À certains endroits, des collisions sont identifiées entre le fond marin et l'animal. Dans d'autres cas, il y a des phases au sol où la profondeur de l'animal ne correspond pas àc celle du fond marin. Ces deux cas, visibles grâce à une bathymétrie précise, sont des situations physiquement impossibles. il pourrait être utilisé des algorithmes de correspondance de carte semblables à ceux utilisés en navigation terrestre afin de corriger les trajectoires en post-traitement.

Introduction: Context and Objectives

On August 6, 2012, the Curiosity rover landed in the Gale crater of Mars to transmit data on possible life forms a few billion years ago. 10 years later, I present you this thesis work which aims to develop a solution to know where are the sea turtles living a few tens of meters away from the inhabited coasts of the Reunion Island. How to explain this contrast in observation capacity, especially for the monitoring of biodiversity and marine fauna?

A first element of answer comes from the impossibility of deploying a global underwater geolocation system like the GNSS constellations used on Earth. A second element is related to the need of deploying small electronic systems over long periods of time which are limited in terms of sensors, consumption, computing and transmission capacity. The inertial units used to position submarines, for example, are far too large and expensive to be considered for these applications. If we want to determine the trajectories of marine animals on a fine spatial and temporal scale over long periods, we can also dismiss the use of acoustic solutions that would require a very dense network of receivers at a large scale.

One of the current solutions uses iterative positioning calculations based on information from small sensors measuring orientation and speed of the animal. These algorithms take various forms and are grouped under the term of Dead-Reckoning (DR) (1). Because of its iterative nature, the method is subject to the precision of its components (orientation and speed) and can present significant errors (e.g. $>$ 180 m for a 50 min dive, (2)). Researchers are therefore working to improve those computation techniques (3), developing new sensors (4) and new estimation methods (5; 6). The tag being fixed on the animal, external elements such as the sea current (7), violating the assumption of the displacement in the animal axis or the buoyancy effect (8) bring errors impossible to be estimated by the DR alone.

Geolocated positions are used to correct the estimated trajectories, mostly acquired with the GNSS or Argos systems (9; 10; 11). As noted at the beginning of this chapter, they are however a complex task to get for the marine environment. In recent examples of DR applications, researchers are combining this fine estimation of movement with behaviors to relate distances or swimming patterns to interactions with prey (12; 13).

Numerous DR applications have been developed for marine animals using bio-loggers, i.e. equipment used to record data, which are recovered when the animal is recaptured. These are to be distinguished from bio-telemeter which includes a data transmission system (satellite, GSM, Wi-Fi, Bluetooth, ...) allowing to send messages when they are at the sea surface. To date, no radio bio-telemeters enables to transmit this complex movement information of marine animals. Indeed, the constraints of the related technologies and of the animal behavior make it a difficult

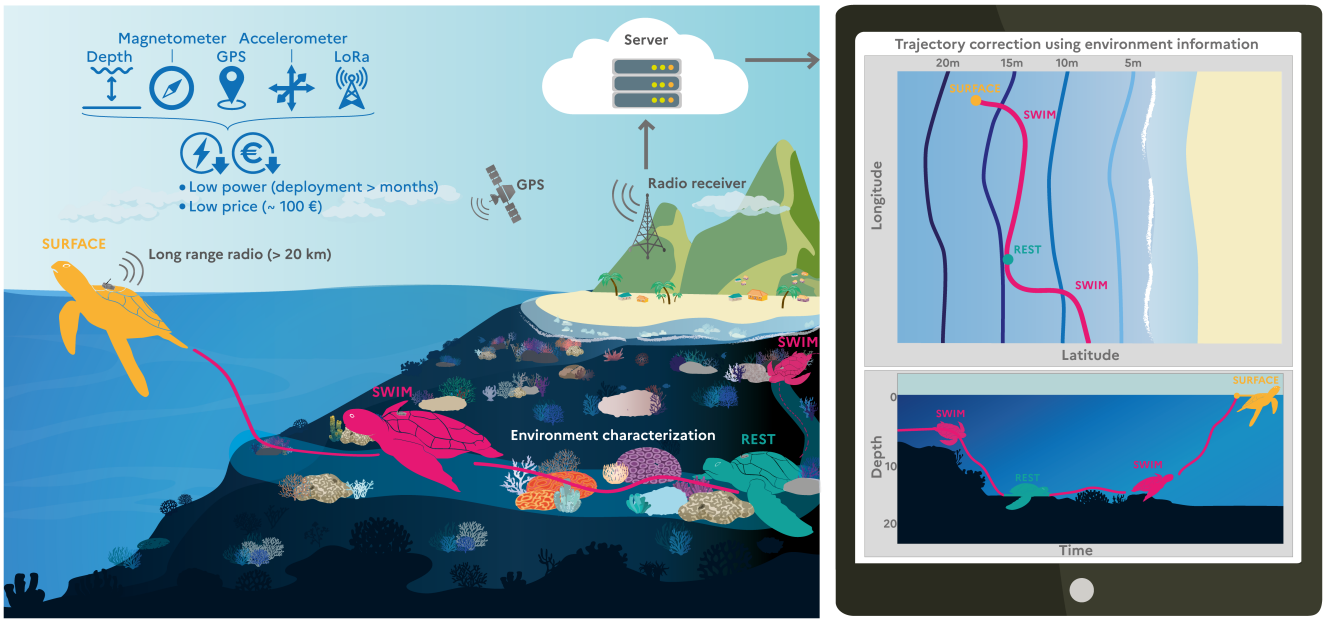


FIGURE 2: Graphical overview of the thesis objectives

task. For example, the volume of data available for Argos transmissions allows the sending of 256 bytes messages every 90 seconds (14). In comparison, uncompressed geodetic positioning data at 1 Hz requires 42 Kbytes for 1 h. (15).

This thesis aims to develop and apply a method to overcome the various constraints on our ability to determine and transmit the trajectories on a fine spatial and temporal scale (Figure 2). The main challenges to overcome are the size of the equipment, its power consumption, its price, its on-board computing, data storage capacity, and the transmission system.

Some of these constraints have been overcome and tested during my previous position as an engineer at Ifremer where I have developed bio-telemeter tags for sea turtles and the system for receiving the transmitted data. This work was carried out within the context of the IOT project [0] which aims to better understand the functional habitats (e.g. feeding, resting) of juvenile green and hawksbill turtles (*Chelonia mydas* and *Eretmochelys imbricata*) in the South West Indian Ocean. The main features of these developments include:

- The development of an electronic board with accelerometer, magnetometer, pressure sensors with a low power microcontroller programmable in *Arduino*. A LoRa data transmission module allowing communications with receiving stations at several tens of kilometers has been integrated into this board. This module has a very low power consumption and with a limited cost for the electronic part and the data transmission (those are relatively high for the satellite connections for example). The mechanical part has been realized in order to produce a tag resistant to pressure and to marine environment.

[0] https://ocean-indien.ifremer.fr/Projets/Innovations-technologiques/pIOT-2018-2020-IOT-2018-2021/IOT-2018-2021

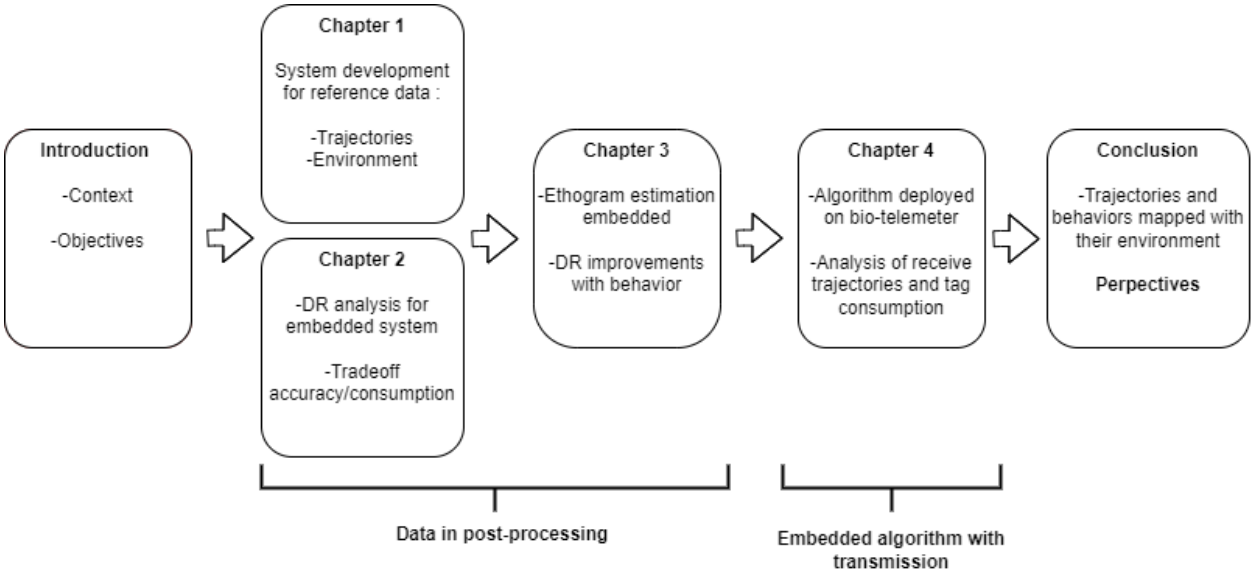


FIGURE 3: Transition graph of the thesis chapters

- The development of a reception network for LoRa transmission. It is designed to be used in isolated areas (without internet connection and power source) by connecting only one gateway (GW) to a satellite internet modem. The user can also use existing private networks. In both cases, the received messages are stored on a server and are instantly available for display or post-processing.

All this preparatory work of development, deployment, test, and validation has permitted to lift the first technical barriers mentioned above in terms of size, price, transmission capacity and storage capacity. The challenge of this thesis work is to integrate an algorithm to obtain the trajectory of the individual underwater. I will expose in 4 chapters the different steps that allowed to reach an embedded solution to transmit the trajectories of marine turtles for each of their 30 min dives during more than four months. In each chapter, the literature is reviewed to provide the state of the art about the related topics.

Chapter 1 focuses on the development of solutions for the acquisition of reference underwater trajectory data. These ground-truth data are used in the calibration and qualification of the algorithms for underwater trajectory estimation. An Autonomus Surface Vehicle (ASV) capable of sampling accurate trajectories at a fine spatial scale ($<$3 m) and time (1 Hz) is presented. It allows in parallel the characterization of the environment by bathymetry and photogrammetry of our study areas.

Chapter 2 proposes an approach to study DR in an embedded system context. With reference data acquired in controlled environments using a swimmer and a high-precision GPS, we test configurations for different sensors and sampling frequencies. In parallel, we perform energy consumption measurements to study the trade-off between the accuracy of the different configurations for trajectory estimation and the power consumption.

Chapter 3 aims at developing a new method to improve the accuracy and the consumption of the embedded DR using the animal behavior. We use data acquired on a wild sea turtle tracked using the ASV. The chapter proposes a simple and flexible on-board method to determine the animal behavior and adapt the speed estimation

method according to it. Finally, we investigate the improvement in accuracy and energy consumption compared to conventional non-adaptive methods.

Chapter 4 describes the integration of the developed algorithm within a bio-telemeter with the study of the associated constraints and its validation by *in situ* tests. We study precisely the compression and the transmission of the trajectories as well as the global real consumption induced by the code. The chapter shows the versatility and open-source features of these developments.

The conclusion summarizes the results of the thesis and proposes perspectives of work in this domain. One of the major perspectives proposed is the post-processing of the data sent to integrate environmental variables and improve the quality of the trajectory estimation.

Bibliography

[1] R. P. Wilson, N. Liebsch, I. M. Davies, F. Quintana, H. Weimerskirch, S. Storch, K. Lucke, U. Siebert, S. Zankl, G. Müller, I. Zimmer, A. Scolaro, C. Campagna, J. Plötz, H. Bornemann, J. Teilmann, and C. R. McMahon, "All at sea with animal tracks; methodological and analytical solutions for the resolution of movement," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 54, no. 3, pp. 193–210, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967064507000045

[2] C. Laplanche, T. A. Marques, and L. Thomas, "Tracking marine mammals in 3D using electronic tag data," *Methods in Ecology and Evolution*, vol. 6, no. 9, pp. 987–996, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12373

[3] P. J. Wensveen, L. Thomas, and P. J. O. Miller, "A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales," *Movement Ecology*, vol. 3, no. 1, p. 31, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0061-6

[4] J. Gabaldon, E. L. Turner, M. Johnson-Roberson, K. Barton, M. Johnson, E. J. Anderson, and K. A. Shorter, "Integration, Calibration, and Experimental Verification of a Speed Sensor for Swimming Animals," *IEEE Sensors Journal*, vol. 19, no. 10, pp. 3616–3625, 2019, conference Name: IEEE Sensors Journal.

[5] D. E. Cade, K. R. Barr, J. Calambokidis, A. S. Friedlaender, and J. A. Goldbogen, "Determining forward speed from accelerometer jiggle in aquatic environments," *Journal of Experimental Biology*, vol. 221, no. 2, p. jeb170449, Jan. 2018. [Online]. Available: https://doi.org/10.1242/jeb.170449

[6] I. A. Bouyoucos, D. W. Montgomery, J. W. Brownscombe, S. J. Cooke, C. D. Suski, J. W. Mandelman, and E. J. Brooks, "Swimming speeds and metabolic rates of semi-captive juvenile lemon sharks (Negaprion brevirostris, Poey) estimated with acceleration biologgers," *Journal of Experimental Marine Biology and Ecology*, vol. 486, pp. 245–254, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022098116302015

[7] K. Shiomi, K. Sato, H. Mitamura, N. Arai, Y. Naito, and P. Ponganis, "Effect of ocean current on the dead-reckoning estimation of 3-D dive paths of Emperor Penguins," *Aquatic Biology - AQUAT BIOL*, vol. 3, pp. 265–270, 2008.

[8] R. M. Gunner, M. D. Holton, M. D. Scantlebury, O. L. van Schalkwyk, H. M. English, H. J. Williams, P. Hopkins, F. Quintana, A. Gómez-Laich, L. Börger, J. Redcliffe, K. Yoda, T. Yamamoto, S. Ferreira, D. Govender, P. Viljoen, A. Bruns, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, C. M. Duarte, M. C. van Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "Dead-reckoning animal movements in R: a reappraisal using Gundog.Tracks," *Animal Biotelemetry*, vol. 9, no. 1, p. 23, 2021. [Online]. Available: https://doi.org/10.1186/s40317-021-00245-z

[9] R. M. Gunner, M. D. Holton, D. M. Scantlebury, P. Hopkins, E. L. C. Shepard, A. J. Fell, B. Garde, F. Quintana, A. Gómez-Laich, K. Yoda, T. Yamamoto, H. English, S. Ferreira, D. Govender, P. Viljoen, A. Bruns, O. L. van Schalkwyk, N. C. Cole, V. Tatayah, L. Börger, J. Redcliffe, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, H. J. Williams, C. M. Duarte, M. C. van Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "How often should dead-reckoned animal movement paths be corrected for drift?" *Animal Biotelemetry*, vol. 9, no. 1, p. 43, 2021. [Online]. Available: https://doi.org/10.1186/s40317-021-00265-9

[10] C. Kuhn, D. Johnson, R. Ream, and T. Gelatt, "Advances in the tracking of marine species: Using GPS locations to evaluate satellite track data and a continuous-time movement model," *Marine Ecology-progress Series - MAR ECOL-PROGR SER*, vol. 393, pp. 97–109, 2009.

[11] Y. L. Bras, J. Jouma'a, and C. Guinet, "Three-dimensional space use during the bottom phase of southern elephant seal dives," *Movement Ecology*, vol. 5, no. 1, p. 18, 2017. [Online]. Available: https://doi.org/10.1186/s40462-017-0108-y

[12] T. Fukuoka, T. Narazaki, C. Kinoshita, and K. Sato, "Diverse foraging habits of juvenile green turtles (Chelonia mydas) in a summer-restricted foraging habitat in the northwest Pacific Ocean," *Marine Biology*, vol. 166, no. 3, p. 25, 2019. [Online]. Available: https://doi.org/10.1007/s00227-019-3481-9

[13] S. Andrzejaczek, A. C. Gleiss, K. O. Lear, C. B. Pattiaratchi, T. K. Chapple, and M. G. Meekan, "Biologging Tags Reveal Links Between Fine-Scale Horizontal and Vertical Movement Behaviors in Tiger Sharks (Galeocerdo cuvier)," *Frontiers in Marine Science*, vol. 6, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2019.00229

[14] S. L. Cox, F. Orgeret, M. Gesta, C. Rodde, I. Heizer, H. Weimerskirch, and C. Guinet, "Processing of acceleration and dive data on-board satellite relay tags to investigate diving and foraging behaviour in free-ranging marine predators," *Methods in Ecology and Evolution*, vol. 9, no. 1, pp. 64–77, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12845

[15] T. Wild, M. Wikelski, S. Tyndel, G. Alarcon-Nieto, B. Klump, L. Aplin, M. Meboldt, and H. Williams, "Internet on Animals: WiFi-enabled devices provide a solution for big data transmission in bio-logging," *Methods in Ecology and Evolution*, 2021.

Chapter 1

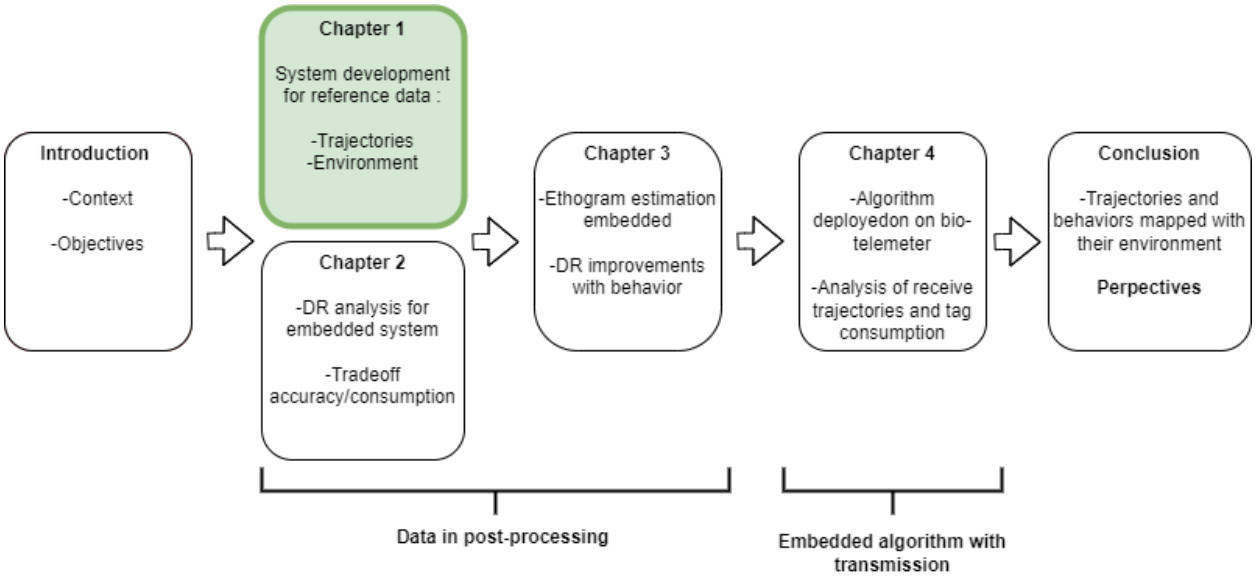
Plancha ASV : Affordable open-source vehicle allowing autonomous acoustic tracking, bathymetric, and photogrammetric surveys

Authors : Pierre Gogendeau, Sylvain Bonhommeau, Hassen Fourati, Mohan Julien, Matteo Contini, Thomas Chevrier, Anne-Elise Nieblas and Serge Bernard.

Article status : In preparation for *IEEE Journal of Oceanic Engineering*

First prototype of the ASV

Keywords: ASV, acoustic tracking, bathymetry, photogrammetry, open-source.



Reminder of the chapter transition graph. The current chapter is highlighted in green

Abstract

Knowledge of marine species and marine ecosystems is a prerequisite for their sound management and conservation. Scientists need regular accurate environmental monitoring, habitat use and animal movement information. To provide this scientific information, we develop an Autonomous Surface Vehicle (ASV). The principal functions developed are the autonomous acoustic tracking of a target, single beam bathymetric survey for depth < 50 m and photogrammetry survey for depth < 10 m. The current specification enables users to cover 100 x 100 m areas in 2h (with 1 m strips within the area) or to track an animal equipped with an acoustic tag for 5 h at a spatial accuracy of 1 m. Developed using a large paddleboard, it provides scientists with a multi-modal, affordable, open source, and reproducible tool to collect information on bathymetry, habitats, and animal ecology in shallow waters even in remote areas. Versions of the ASV come from \$2434 to \$11072 depending on the functions needed. Each function of the ASV is validated and illustrated by field mission examples.

1.1 Introduction

Conservation of endangered marine species and marine ecosystems requires a wide range of scientific knowledge from physics, chemistry, biology, and ecology. Recent technological innovations give scientists several keys to answer these questions (1). For marine species, this can be achieved by linking information about behaviors, trajectories, or functional habitats (2; 3). Marine environments are stressed by the impacts of climate change and human activities. To help implement a sound management plan, conservation measures, and track the impacts of these actions over time, it requires accurate monitoring of key indicators from local to global scales (4; 5; 6)

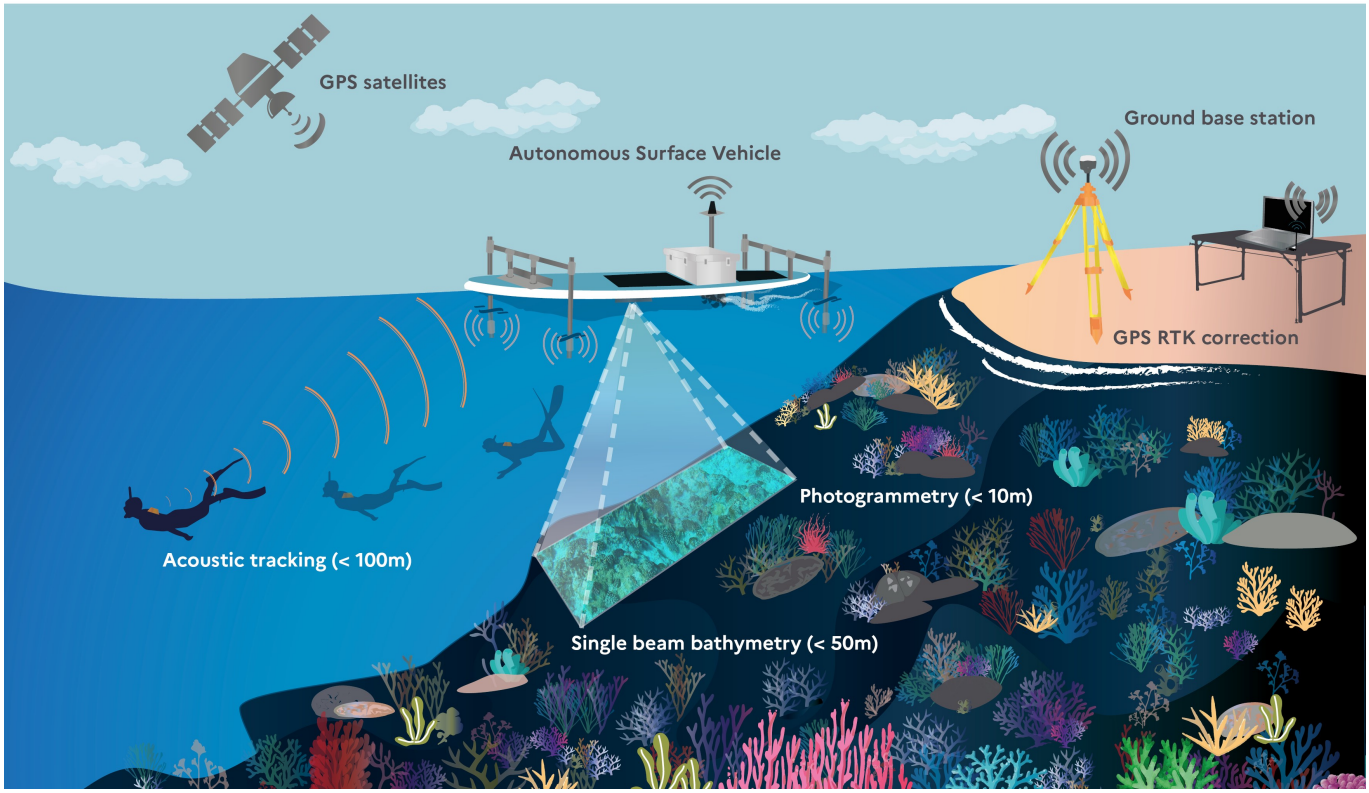


FIGURE 1.1: Different ASV missions: Autonomous acoustic tracking; Single beam bathymetric survey; Photogrammetric survey

The IOT project linked with this study aims to describe the functional habitats of juvenile marine turtles in the western part of the Indian Ocean. While we were estimating sea turtle trajectories, we faced some difficulties in collecting reference positioning data on a fine temporal scale. In this context, among the various possible systems (see Section 1.1.2), we chose to develop an autonomous surface vehicle (ASV) to ensure the acquisition of high precision trajectories. Our global objective is to estimate the link between the trajectory and the environment of the animal.

In this paper, an open-source ASV solution is proposed, allowing the acoustic tracking of slow marine animals using a short baseline (SBL) acoustic system in a range of 100 m as well as the collection of environmental data, such as bathymetry and information from photogrammetry. Moreover, this ASV is developed to be simple to reproduce, low-cost, and multi-modal to allow scientists to add other sensors.

1.1.1 Acquisition of accurate underwater positions

Functional habitats description, behavior and spatial environment occupancy can be addressed with advances in biologging technology that enable fine-scale geolocated trajectories. However, it is a complex task to geolocate the tracks of wild animals in the marine environment (7). A common tool used in biologging studies are radio-transmitters, which can transmit geolocated trajectories of the study subject via *Argos* satellites. However, it is impossible to use radio bio-transmitters for marine species, as it is impossible to send trajectory information underwater. Furthermore,

their bandwidth to send messages through the *Argos* satellites is too low (8) to enable the quantity of data required for fine-scale trajectory estimation. In their recent study, Cox et al. (8) summarized trajectory information by sending five inflection points of a seal diving profile to describe its underwater trajectory. They were able to send these fine-scale trajectory data by embedding the algorithm for trajectory estimation in the bio-transmitters and compressing the data to be sent.

Underwater fine-scale geolocated tracking is possible with acoustic positioning. Some systems with anchored or buoy receivers need dense acoustic receiver arrays (9) to use their geolocation algorithms. These systems are not adapted to our application because the area of movement of the turtles is several kilometers and the area to cover with this acoustic network is too wide. Some other acoustic systems are more compact like ultra-short baseline (USBL) and short baseline (SBL) acoustic systems. USBL and SBL calculate the range of an acoustic transponder based on the signal time of arrival (TOA) or time of flight (TOF). In addition, USBL uses a phase-differencing algorithm with the receiver baseline to get the bearing angle (10). With the calculated relative position, both systems infer the geolocated position of the transponder adding the global navigation satellite system (GNSS) position of the receiver system. USBL receiver systems are more compact and offer a better range and accuracy. For the Blue Print USBL [1], the range is 1 km with 0.1 m accuracy compared to the 100 m with 1 m accuracy of the Waterlinked UGPS G2 SBL [2]. For these reasons, USBL can be installed on robotic system such autonomous underwater vehicle (AUV) (11) or autonomous surface vehicle (ASV) (12). Dodge et al. (11) were able to follow a turtle with their AUV for several hours with a USBL. The drawbacks of the USBL systems are their prices and the loss of accuracy in the shallow environment.

Another common technique for trajectory estimation of marine animals is dead-reckoning, using a fusion of inertial data, sensor speed, and GPS positions (13). Scientists have determined trajectory estimates from dead-reckoning on a wide variety of marine animals such as pinnipeds (14; 15), whales (16; 17), turtles (18), and fishes (19). The accuracy of the trajectories strongly depends on the availability of geolocation information, such as GPS positions (20), even if they are only sporadically available. The GPS positioning itself has minimal uncertainty but is dependent on the quality of the signal (21). For surface-breathing marine animals, the quality of the signal directly depends on the animal's surfacing time and frequency. Other issues with these techniques are to retrieve the logger tag. Retrieval of the tag is possible for short-term deployment (16) or for animals coming back to their initial locations (14), but for long deployments, retrieval is difficult or impossible

Our project goal is to propose affordable hardware and focus on the accurate fine scale trajectory of juvenile green turtles evolving in the shallow reefs of Reunion Island. We chose the SBL Waterlinked UGPS G2 system for these reasons. To overcome the range constraint of 100 m, we have adapted the navigation system of the ASV to follow the acoustic transponder within this range.

[1] https://www.blueprintsubsea.com/seatrac/seatrac-lightweight

[2] https://store.waterlinked.com/product/underwater-gps-g2/

1.1.2 Development of a low-cost and reproducible ASV

ASV and other robotics hardware in marine environments are becoming more affordable. Most of the USV/ASV were known to be very expensive and reserved for the military (22; 23), the industry (24), or some scientific institutes (25; 26). In the past few years several projects emerged, proposing small and low-cost ASV/USV under $5000 without specific sensors (27; 28; 29). This accessibility improvement is made thanks to some companies proposing cheap, reliable and open-source electronics and marine robotic parts. For instance, the T200 thruster made by *Blue Robotics* is used by many hobbyists (30), scientists (31), and industrial (32; 24) projects. We found the same positive evolution in software programs. Professionals and hobbyists developed good quality, easy-to-use, well documented, and open-source autopilots systems. For example, *Ardupilot* is now embedded in various vehicles such as drones, rovers, remotely operated vehicle (ROV) and boats (33; 34; 35). It can also be used for data logging, analysis, and simulation. The open community linked to these projects makes them in constant and dynamic evolution.

1.1.3 Environmental data from bathymetry

Almost only professionals perform ASV bathymetry surveys, as they require expensive sensors such as an echosounder and a differential GPS for sub-centimetric positioning. The echosounder pings a signal to the bottom of the seafloor and measures the depth with the signal travel time and the signal velocity in water (36). For instance, it can be used to map harbors or channels. For ecological purposes, the bathymetric map can be compared with animal depth to understand water column use during specific behaviors (11).

In the same way as other electronic systems, bathymetric sensors tend to be cheaper. For example, we first started with an ECT-400 echosounder [3] at $3700 and we are now testing a S-500 by Cerulean at $595. Several projects emerged in the past few years and offered ASV for bathymetry (25; 28; 27; 37). These projects are not easily reproducible. For instance, the Woods Hole Oceanographic Institution *Jetyak* is not open source (25) . In Carlson et al. (38), the bathymetry accuracy is not specified but the depth map shows pixels around 10 x 10 m. In our application, we want to be able to discriminate small seabed components with at least 5 m radius. Price also limits the use of multibeam echosounders which still cost dozens of thousands of dollars.

1.1.4 Environmental data from photogrammetry

In underwater photogrammetry, almost each research team uses its methodology in image acquisition, pre-processing, and validation. It is often developed in function of the sampling techniques or the accuracy needed. In our work, we proposed an easy method for planning and validation of photogrammetric surveys made with an ASV.

[3] https://www.echologger.com/products/single-frequency-echosounder-deep
[3] https://ceruleansonar.com/products/sounder-s500

Photogrammetry enables the 3D reconstruction and mapping of a scene with overlapping images from different perspectives (39). For underwater purposes, archaeologists introduced it in 1968 (40; 41). Recently, many research teams have applied underwater photogrammetry for scientific goals (42; 43; 44). Primarily focusing on small coral colonies with surveys made by divers, these studies give accurate coral surface estimation ranging between 2 to 19% (45). In Marre et al. (43), they achieve an average model resolution of 3.4 mm.

Lately, some studies have used ASV for photogrammetry surveys (46). These studies necessitate high computing resources and give less accurate resolution (47). Covering a larger area with an ASV is however made possible when images are coupled with accurate GPS position and orientation data. This additional piece of information helps to run the model more quickly and accurately. Orthophotos can then be mapped on the bathymetry from the echosounder. From an environmental point of view, this gives valuable information on the topology and the type of bottom.

Software improvement simplifies the computing process without the need of long and complex pre-processing with automatic camera-ordering or camera calibration. Several software, open-source or not, are now available. Their comparison is hard because it depends on the survey condition and image quality (38).

The drawbacks of using the ASV are the limited depth at which the bottom can be mapped, dependent on the light, image quality, turbidity, and condition at the sea surface. Photogrammetry softwares are not specific for underwater and do not allow for the correction of seawater light attenuation. Photogrammetry softwares such as *Matisse* (48) has been design for underwater applications and seawater light specificity. It is open-source and provides 3D and 2D models.

This chapter describes and gives all the necessary tools to build and use an ASV with acoustic tracking ability as well as bathymetry and photogrammetry data collection. This ASV allows getting environmental data from these bathymetric and photogrammetry features and to link them to fine scale movement data from the acoustic beacon. In the first Section, we described the different specifications for the configuration of the ASV. Then, we described all parts of the vehicle. We first presented the mechanical part, followed by the electrical and software parts. The "validation and characterization" section presents each functionality description and validation of the ASV features with field missions. We provided the complementary information, mounting instruction, hardware, and software files as well as training datasets in the Git repository [4].

1.2 ASV requirements and operation scenarios

In this paper, we present the ASV named Plancha ASV (see Figure 1.2). It was developed for 3 main missions:

- Autonomous acoustic tracking of an underwater target
- Single beam bathymetric survey
- Photogrammetric survey

[4] https://gitlab.ifremer.fr/sb07899/Plancha-ASV.git

TABLE 1.1: ASV requirements for the different operations

Global	
Handling	2 people recommended
Transport	< 2.5 m (for aircraft regulation)
Deployment	From a small boat or the shore
Environment	Tested in tropical weather: Temp : 10 – 35°C
Stance	Stable for wave : 0.5m / wind : 20 kt
Guidance	Autopilot and manual control
Buoyancy	Can support >10 kg
Communication	Telemetry range > 1 km
Power limitation	Motor under 2.5 kW
Surveys	
Lifetime per survey	>2 h
Speed	Between 0.5 and 1.2 m/s
Navigation	Autopilot allow following 1m transect
Bathymetry sensor	Single beam echo-sounder
Photogrammetry sensor	Camera (e.g. GoPro)
Communication mode	Cellular & telemetry
Tracking	
Lifetime per survey	>5 h
Speed	about 0.8 m/s
Mechanic	2 m between each hydrophone
Sensor 1	Acoustic geolocation system (SBL)
Sensor 2	Camera for behavior analysis
Communication mode	Celular & telemetry

The hull is made from a paddleboard which allows adding other sensors according to scientific needs. The ASV requirements are summarized in Table 1.1. It is made to be easily deployed, transportable and rugged. Depending on the deployment location, the needs and the different available resources, the ASV can be used with or without 3G/4G network. All functions are possible for the two modes, even if acoustic tracking is more complex without an internet connection and does not allow checking the tracking live. Bathymetric and photogrammetric surveys need more accurate navigation in comparison to the tracking mode. The global network system architecture of the *Plancha* ASV is detailed in Figure 1.4.

To be affordable and reproducible, all the electronic parts (except the echosounder) are commonly used components of robotics hobbyists (ROV, drone, etc...) and are easy to buy from general robotics sellers. These parts can however be changed according to their availability, the local regulations (e.g. radio-frequency), and other sensors/instruments can be added.

1.3 ASV Description

We divided this section into mechanical and electronic parts. In Table 1.2a, we presented the main components with the total price of each ASV configuration. A complete BOM [5] is provided. The mounting tutorials, wiring, CAO files, and installation configuration of the different software components are available on the Git repository[6].

1.3.1 Mechanical part

TABLE 1.2: Description of the main ASV parts for the different configurations and operations

(a) Different parts classified by mode and operation

Global Mode	Component	Name	Number	Unit Price ($)
Electrical	Fligth controller	Pixhawk cube 2.1 black	1	$315
	GNSS RTK	Emlid reach M2	1	$499
	Telemetry	RFD900	1	$277
	Radio command	RadioLink AT9S	1	$129.99
	Thruster	Blue robotic T200	2	$179
	ESC	Blue robotic Basic ESC	2	$27
	Battery	Tattu 14.8V 25C 4S 10000mAh	2	$149
Remote	GNSS RTK Base	Elmid reach RS2	1	$2199
	GNSS radio communication	Reach LoRa radio	1	$118
Internet	4G dongle	Huawei E3372	1	$50
	Companion board	Raspberry pi 3B+	1	$38
Mechanical	Hull	Paddle board 8", 80L	1	$250
	Waterproof case	HRDR waterproof case	1	$225.20
	Thruster support	Custom aluminum support	1	$150
	Cobalt Series Connector	Blue trail engineering Connector	2	$67
Surveys Mode				
Electrical	Echosounder	ETC400	1	$3850
	Camera	GoPro Hero 7	1	$349
Mechanical	Echosounder holder	Printed custom part	1	$20
Tracking Mode				
Electrical	SBL acoustic receiver system	Waterlinked Underwater GPS	1	$2200
	Acoustic beacon	Waterlinked locator U1	1	$1500
	Additional battery	Tattu 14.8V 25C 4S 10000mAh	2	$149
Mechanical	Aluminum holding arm	Aluminum tubs	2	$200

(b) Price estimation of the ASV for the different modes. Only indicative, it does not include cheap components and spare parts

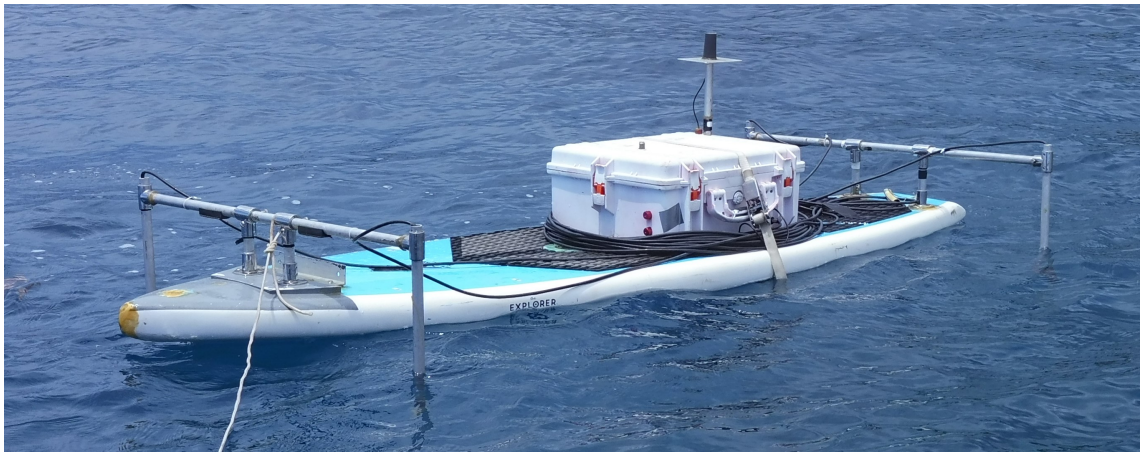
	Global (G)	G + Surveys	G + Tracking	G + Surveys + Tracking	Remote
Total	~ $2434	~ $6634	~ $6802	~ $8672	add $2400

Most custom parts are made using a 3D printer. The main mechanical parts are a paddleboard, a waterproof case for the electronics, and a thruster support underneath the board. For the acoustic mode, arms are added to hold and immerse the 4 hydrophones needed.

[5] BOM link : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Documents/4_BOM.xlsx

[6] Git link : https://gitlab.ifremer.fr/sb07899/Plancha-ASV.git

(a) ASV preparation for a survey in remote mode with the mobile GPS RTK base station (on the yellow tripod)



(b) ASV in acoustic mode

FIGURE 1.2: ASV photos for the different modes: (a) Survey mode for bathymetric and photogrammtric data collection and (b) acoustic mode for animal tracking with the four arms equiped with hydrophones.

Hull, cases and thruster

The hull is made with a simple paddle board of 8" and 80 l. Two thrusters are used and mounted on the protection support when the board is on the ground or in very shallow waters. This support is made in 5 mm marine aluminum to be robust and is screwed to the board. We potted a support base screw and bolted it on both sides to be waterproof and robust. Cables are passed through the board thanks to two printed and coated cable entries. The echosounder support is also printed and potted in a hole drilled in the board.

Electronic parts and sensors are in a waterproof case of 543 × 419 × 218 mm external dimensions (482 × 358 × 198 mm internal dimensions). The connection to external sensors is established with epoxy potted waterproof penetrators from *Blue Robotics*. The GPS antenna mast is made of aluminum and acts as a ground plane for the antenna. It can be printed in plastic but a metallic component of the same size needs to be fixed below the antenna as the ground plane. The echosounder is wired with the Binder 770 Bulkhead Connector and the plug from *Blue Robotics*. For the wiring of the thrusters with high electrical current, we chose cobalt series bulkhead connector and the plug from *Blue Trail Engineering*.

Acoustic integration

In our case, 4 hydrophones are needed for the acoustic system. We mounted them with 2 aluminum holding arms separated by 2 m following the constructor recommendation (see Figure1.2.b). The first arm in the back of the board is composed of 5 aluminum tubes: 2 small tubes of 10 cm, 2 of 60 cm and 1 of 2 m. Connection between the 60 cm and 2 m tubes are made with stainless-steel elbow from marine hardware stores. Fixation of the arm and the board are made with stainless steel bases (on the board) and stainless steel Ts for the long tube. Bases are screwed and inserted into the board. The same kind of arm is on the front with 2 small ones of 10 cm, 2 of 60 cm and 1 of 1.5 m. As the space between the bases is smaller on the front, we reinforce the fixation by fixing the 2 bases on printed support which is potted on the board. To connect the 4 acoustic receivers, we used binder 770 bulkhead connectors and plugs from *Blue Robotics*. They are already mounted on the acoustic electrical.

1.3.2 Main electrical parts

For the electrical and software sections, we first described the power part and then the main components and sensors. In Figure 1.3.b, the power is represented by a blue background and the command and sensors by a green one. In Figure 1.3.c, the numbers given for the main components are the same used in the following description part. The core of this part is standard for an ASV or a rover. It is composed of an autopilot (component 1), a GPS module (component 4) and communication systems (component 7). The sensors and other communication systems used depend on what the user plans to do with the ASV. The entire electrical part, external sensors (Camera, echosounder) and the ESC' thrusters are in a waterproof

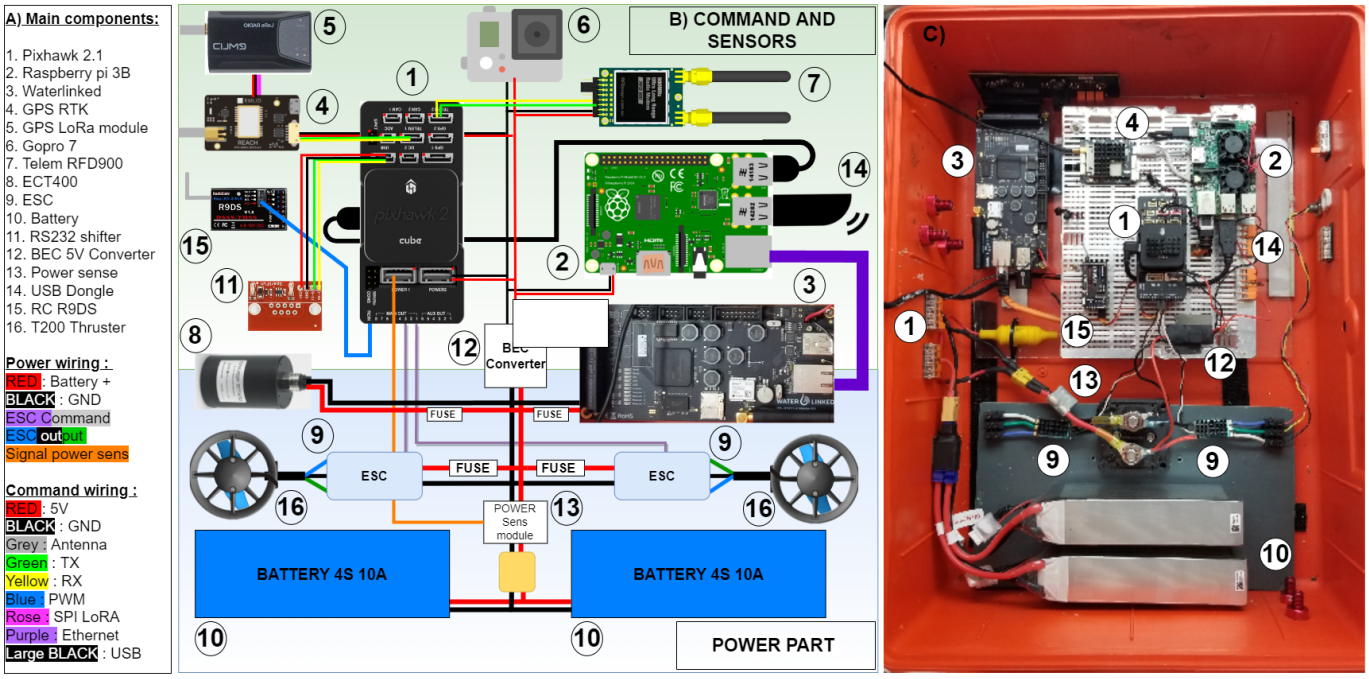


FIGURE 1.3: ASV high level electrical diagram and electrical circuit. On the left (a), the corresponding numbers and names of the main parts. The colored names correspond to different wires on the electrical diagram. In the middle (b), the high level electrical diagram with main components and wiring. On the right (c), the electrical circuit with the corresponding numbers. Some components are fixed on the top of the case or outside and thus are not visible on this photo.

case (Figure 1.3.c). Figure 1.3.a represents the high level electrical diagram and a picture of the ASV electrical circuit with the annotated corresponding components.

Power part

The power part is composed of a minimum of two 4S, 10 Ah batteries (component 10 - Figure 1.3), 2 electronic speed controllers (ESC) (component 12), 2 thrusters (component 16), 1 voltage monitor (component 13), 1 voltage regulator (component 12) and some fuses. Set aside batteries, all the components are from *Blue Robotics*. Thrusters are the *T*200. We chose 4S / 10 Ah to be able to travel by plane and follow international transportation regulations for lithium batteries. Batteries are connected to a power sense module which is connected to the autopilot, *Pixhawk*. Following the power sense, there is a power connection terminal connected to the sensor 1 voltage converter and the 2 ESC thrusters.

The following section describes the software part and how the different components communicate with each other. A graphical summary is available in Figure 1.4.

Autopilot

Autopilot or flight controller is the *Pixhawk* 2.1 cube black (component 1 - Figure 1.3). Except the camera and SBL, all the components and sensors are connected

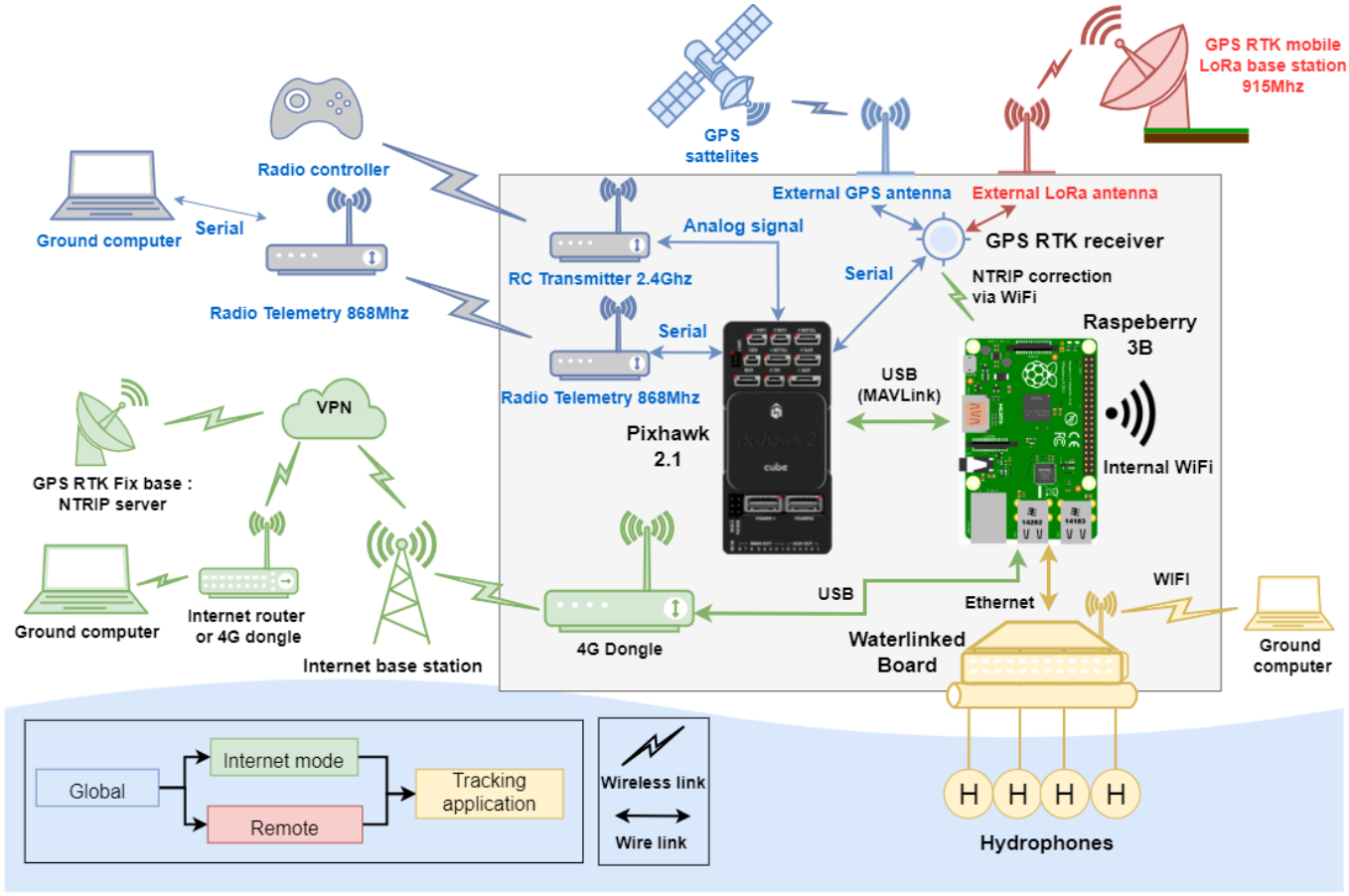


FIGURE 1.4: Network diagram of the ASV showing how the autopilot get and interact the difference sources of information to perform the navigation of the ASV

to the flight controller. The flight controller is powered through the 5V output of the voltage regulator. The power sense module provides information on battery voltage and electrical consumption. It is also connected to the *Pixhawk*. The flight controller is configured with the open-source autopilot *Ardupilot* rover V3.5 in "boat" mode. It handles the navigation rules and the configuration of hardware and sensors. The parameters of our configuration are given in the parameter file available in the Git repository [7]. These settings depends on the board and the hardware used and a calibration should be done. The autopilot uses the mavlink protocol to communicate via USB to the companion computer and with radio telemetry to the ground-based computer. Ground control station software (GCS) is required to communicate with and control the autopilot. Different GCS are available and we used *Mission planner*. GCS displays real-time variables and positions of the ASV. Mission Planner allows mission planning for the surveys and setting all the parameters of the vehicle (Figure 1.5). More information on how to install and use it are available on the *Ardupilot* website [8]. A general tutorial about *Ardupilot* rover is available on this link [9].

[7] Parameter file path: *https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Software/Parameters/param_110122.param*

[8] https://ardupilot.org/copter/docs/common-choosing-a-ground-station.html

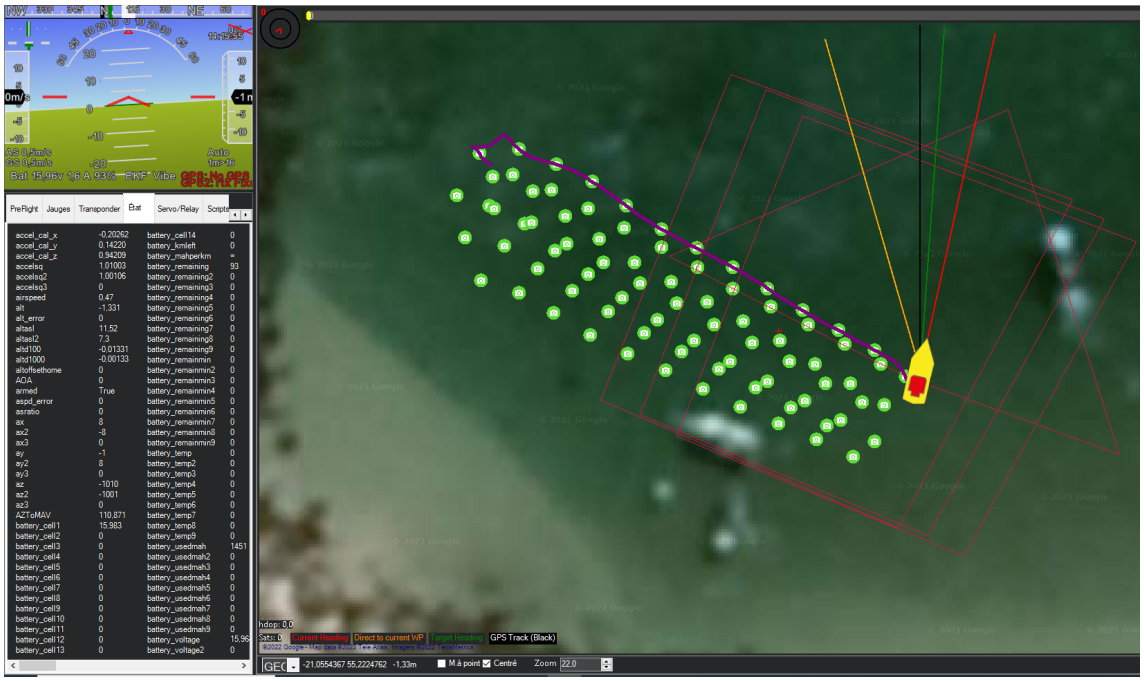


FIGURE 1.5: Screenshot of Mission Planner during a navigation test in Saint-Gilles les Bains (Reunion island). The yellow boat shape corresponds to the ASV position. Purple line is its actual track and the green dots are positions where an external signal is sent to control a camera.

Companion computer

The companion computer is a *Raspberry Pi 3B* (component 2 - Figure 1.3). It is powered by a 5 V regulator. The companion computer and the flight controller are connected with a USB cable for serial communication. The USB cable cannot be used to power the *Raspberry Pi*. The *Raspberry Pi* has multiple uses: it communicates with the acoustic module and the flight controller and allows to run custom scripts used for sensors and ASV components. During tracking mode, we run the *Python* acoustic tracking script on the *Raspberry Pi* which uses information from the flight controller and the acoustic modem. In Internet mode, the connection is made using a USB 4G dongle. The companion computer then acts as a Wi-Fi access point to share its connection to the *Raspberry Pi* and the ground computer is possible through its Wi-Fi access point or via internet. We set up and used our VPN with OpenVPN to be able to access the *Raspberry Pi* with SSH via the internet. More information on the *Raspberry Pi* used as companion computer are available online[10]. Detailed information and procedure to install the *Raspberry Pi* image are

[9] https://ardupilot.org/rover/docs/rover-first-drive.html
[10] https://ardupilot.org/dev/docs/raspberry-pi-via-mavlink.html

available on the Git repository [11].

RTK GNSS

We used *Emlid Reach* M2[12] as a differential GNSS (component 4) with the possibility of Real Time Kinematics (RTK) (Figure 1.3). Connection is made through serial communication with a telemetry port of the flight controller. We powered the *ReachM2* with the micro USB connector connected to a 5V voltage regulator. Do not power it with the telemetry port of the *Pixhawk* 2.1. This module has a plastic enclosure that we removed to avoid overheating. RTK corrections are received via internet from a NTRIP server or over LoRa in remote mode. For this mode, we added the *Emlid* LoRa radio receiver (component 5). A WebGui or a smartphone app is available to configure the *ReachM2*. In internet mode, the GNSS is connected to the Wi-Fi access point of the companion computer and corrections are fetched through our online NTRIP server using a docker available here[13]. For remote mode, corrections are fetched using a LoRa link. In that case, a second GNSS receiver is set as a reference base and sends RTK corrections to the embedded GNSS. For that purpose, we used an *Emlid* RS2 at a known position. LoRa's input correction frequency is set at 868 MHz. This frequency must be adapted with the radio communication local regulations.

The global setup of the GNSS module is available on emlid documentation[14]

Communication

Different methods of communication are possible. For telemetry, we used a radio or internet connection. Even in internet mode, we used radio telemetry as a backup because this system is trustworthy. The Radio telemetry (component 7 - Figure 1.3) allows for communication with the autopilot through ground station software via mavlink protocol. We chose the RFD900x module at 868 MHz which has a range of 20 km. It ensures a reliable link with the ASV and it is used in both modes. The internet connection is made with a 4G dongle with a SIM card (Huawei E3372; component 14 - Figure 1.3) plugged into the *Raspberry Pi 3B*.
To control the board in manual mode and do some simple tasks such arming/disarming the thrusters, we used an RC command using radio communication (RC model R9DS with radiofrequency at 2.4 GHz). The RC receiver is connected to the RCIN port of the flight controller. The RC radio command (component 15 - Figure 1.3) is used to arm, disarm, and change mode. It is also used as a backup in case the other transmission systems fail.

[11] Software instructions link : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Documents/2_software_insctructions.docx
[12] (https://store.emlid.com/product/reachm2/
[13] https://github.com/goblimey/ntripcaster
[14] https://docs.emlid.com/reach/reachview-3/connecting-to-reach

1.3.3 Additional Sensors

Echosounder

The echosounder is the ECT400 by *Echologger*[15] (component (8)). It is a single beam frequency echosounder allowing bathymetry survey up to 50 m with a 5° beam. Its ground and power wires are connected to the output of the battery since its allowed power voltage spans from 8 to 75 VDC and thus does not requires any voltage regulation. The echosounder communicates by serial link with the flight controller. A RS232 level shifter is used to convert the output of the echosounder to a 5 V serial signal. Depth is stored in the ardupilot log as "DPTH" variable. It needs to be configured as described in the *Ardupilot* turorial[16].

SBL acoustic positioning

The SBL system is the underwater GPS G2 from *Waterlinked* R100 (component (3)). It is composed of 4 acoustic receivers, a master board, and an acoustic beacon. The electrical board is connected to the *Raspberry Pi* using an Ethernet cable. The input voltage range is between 10 and 30 V. As for the echosounder, we connected the board directly to the battery voltage by adding a 3 A fuse. The acoustic transmitter is the locator U1[17]. It works after manual activation and has 10 h lifetime. The SBL system has a 100 m of range in the standard version. The accuracy of the position given by the constructor is 1% of the range, i.e., 1 m for this application. A WebGui is available to configure the underwater GPS. The acoustic receiver array needs a specific baseline configuration.

Waterlinked recommends a distance of 2 m between each receiver. Distances between the acoustic receivers are measured on the paddleboard and set in the baseline configuration tab in the WebGui. For our application, orientation and position are fetched from the flight controller and sent by the companion computer. The settings "tab/top-side", GPS, and compass have to be switched to *External*. To record the tracking, we used a custom *Python* script run from the companion computer. The software and system integration information are explained in the documentation. For our specific application, the procedure details are available in documentation folder[18].

The position of the acoustic transmitter to the ASV is calculated with a signal Time of Arrival (TOA) algorithm between each different receiver. Then, the system needs the GPS position and heading of the ASV to calculate the geolocated position. To keep the acoustic transponder within the 100-m range, its position is defined as a new way point to be reached by the ASV.

[15] (https://www.echologger.com/products/single-frequency-echosounder-deep)

[16] https://ardupilot.org/copter/docs/common-echologger-ect400.html:Configuringthesensor

[17] https://store.waterlinked.com/product/locator-u1/)

[17] https://waterlinked.github.io/underwater-gps/quickstart/

[18] Documentation folder : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Documents/2_software_insctructions.docx

Camera

We used the *GoPro* 7 black edition (component (6)). The camera is powered by 5V from the voltage regulator. Both photogrammetry and tracking modes rely on *GoPro* 7 images. We used a specific waterproof system allowing us to power the camera with a USB cable. For the photogrammetry the *GoPro* 7 faces down, whereas in tracking mode, it has a 30° angle from the vertical position. During the photogrammetric survey, the field of view of the *GoPro* 7 needs to be as linear as possible. We set the ISO parameter to the lowest value (ISO 100) and the shutter speed at a high value (1/1920) to get clear images and the *GoPro* 7 is set in video mode. A minimum of 70% of coverage is required between two pictures for photogrammetry. To set the space between transects, we used an excel file[19] calculating this space as a function of the depth of the survey area and the coverage needed. The distance between transects will also highly depend on the navigation accuracy capabilities.

More information on the photogrammetric mission planning and pre-processing are available in the "prototype and survey results" Section.

1.4 Prototype validation and survey results

To illustrate the potential applications of the *Plancha* ASV, we present some survey results. The validation of the functioning of the ASV (e.g. accuracy of the trajectory) and the power consumption estimates are provided as Supplementary Materials. All the data and software presented in the section are fully available here[20].

1.4.1 Autonomous acoustic tracking

The acoustic tracking feature allows us to get a fine-scale live trajectory and an active tracking of the underwater acoustic beacon (U1 Locator). For our application, we aim to follow a marine turtle for several hours to analyze its fine-scale trajectory. However, the acoustic tracking feature can be used for other applications such as tracking AUV, divers or any other animals with a limited swimming speed. The next subsections present the tracking procedure, the data processing, and the results of the survey example.

Protocol

The *WaterLinked* system does not save the trajectory and only displays it on their WebGui. In their github[21], *WaterLinked* gives example scripts in *Python* to use or save the data that can be run directly from a laptop. For tracking and logging, we developed our own logging scripts [22].

[19] https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Sotfware/Photogrammetry/Spacing_between_transect_calculator.xlsx

[20] Illustration examples link : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/tree/main/Features_example

[21] https://github.com/waterlinked/examples

[22] https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/tree/main/Sotfware/Tracking

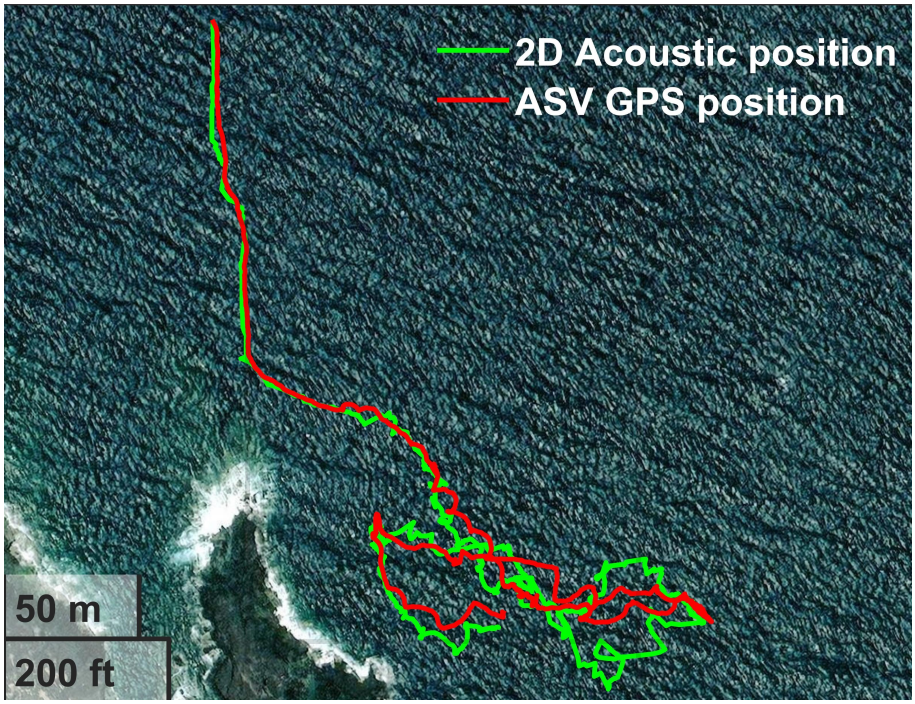


FIGURE 1.6: ASV tracking of a freediver. Green track is the underwater acoustic position. Red is the ASV position

The tracking algorithm[23] enables the calculation of waypoints and their transfer to the autopilot. To start the tracking mode, the user needs to run the command on the *Raspberry Pi* (see software instruction). For the calculation of the next waypoint, the algorithm works as follows: Position and heading are read from the Flight controller of the ASV. It is then sent to the SBL module to calculate the position of the acoustic beacon. The Raspberry then sends a request for the position of the acoustic tag, compares the positions and decides if the ASV needs to move. If the acoustic beacon and the ASV are too close, the autopilot switches to hold mode and stands in its position. If the beacon moves away from the board and the threshold distance is exceeded (here 5 m), then a new position is sent to the autopilot which tries to reach it. Tracking parameters are stored in the Raspberry Pi[24].

The test was carried out at Cap Lahoussaye (-21.017348°N, 55.238212°E). The locator U1 was fixed on the diver's chest with a 50 cm offset from his body towards the seabed so the locator is still underwater when the diver is at the surface and to avoid any loss of the acoustic signal. It is noteworthy that even with this 50 cm offset, we denote more spikes or signal losses when the diver is at the surface. We set the navigation rules to update the distance between the ASV and the diver every second and lower than 5 m.

[23]Tracking script in the raspberry. File name: *main_tracking.py*
[24]Parameter file path in the *Raspberry* : /idocean/parameter.json file

Data processing

Tracking data of the 3D position of ASV and acoustic beacon are logged in the Raspberry. A *MATLAB* script was developed to analyze, filter, and plot the data. We filter out the position data for which the standard deviation of the position estimates are larger than 3 m. A linear interpolation is performed to filter the positions of the acoustic track.

Results

Figure 1.6 shows a 25-minute sequence of a free-diver tracked by the ASV. The ASV successfully tracked the diver over the period. The 3D positions are recorded accurately. This example demonstrates the ability of this system (ASV + acoustic beacon) to collect precise underwater positions that can be used as reference data for animal tracking applications.

For video analyses, the image quality highly depends on the underwater visibility and the distance to the target. Figure 1.7 shows that videos can only be used when the visibility is good so it enables behavioral and trajectory analyses. Moreover, when the ASV is close to the target, it stays in holding mode and drifts and it can lose the target of the camera's field of view.

FIGURE 1.7: Screenshot of the GoPro 7 footage during the tracking test when the diver is going up to the surface. As the seawater is turbid, it limits the ability to use the video for further trajectory and behavioral analyses.

[24]Processing script in Git: https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Features_example/test_tracking_26_10_21/code/main_acoustic_tracking_20_10_21.m

1.4.2 Bathymetry survey

Information extracted from bathymetric data depends on sensor specifications but is also strongly related to the area topology and spacing between collected points. Primary parameters such as the maximum measurement range, the sampling frequency or sensor errors have been fixed during the design phase when we selected the echo-sounder. For each survey, an *a priori* knowledge of the sea ground topology is required to define the aimed data spatial distribution over the survey area. Knowing the average depth and type of ground (e.g. large rocks, sand rift, corals, ...) will help to adjust the spacing between points. The spacing between strips has also to be adapted to the targeted map resolution.

Several standards define and classify the *quality* of bathymetric surveys. For instance, in (49) (section 7.3, Table I), the *International Hydrographic Organization* proposes five categories based on the overall accuracy, the area coverage, and the types of features that can be detected to help classify the *quality* and *goals* of a survey. We use these categories to define our specifications.

The next sections present the protocol, the processing stages, and the final results of a bathymetric survey with an illustration from a survey carried in 2020 on the north shore lagoon of Europa island in the Mozambique Channel.

Protocol

We set up the survey to meet the requirements cited in (49) and described in Supplementary Materials. This enables us to reach the *order 1a* category, i.e. data in harbors, harbor approach channels, coastal areas or inland navigation channels, with a limitation to areas with less than 100 m water depth.

The area of interest was a lagoon in Europa Island. Bathymetry in this area has been estimated using hyperspectral and LiDAR data collected by the Litto3D Ocean Indien project in 2019[25] (see section 1.4.2). From these data, the depth in the area of interest is ranging from about 1 m to 10 m.

From these specifications and to reach the *order 1a* bathymetry standard, the aimed survey area is a rectangle of 49 m × 115 m, with a center coordinates at -22.340984°N, 40.337634°E. The parameters to configure the ASV's autopilot have been set as follow:

- 24 transects in the direction of the largest dimension (width), with a 2-m spacing.
- a target cruise speed of 1 m/s.
- a depth sampling rate of 2 Hz.

These results in a grid of 24 × 228 points over the survey zone, in which the *bathymetric pixels* have a diameter ranging from 9 cm to 90 cm for depth ranging from 1 m to 10 m. Pixels have a widthwise spacing of 0.5 m and a lengthwise spacing of 2 m.

[25] Data accessible here: https://oceans-indien-austral.milieumarinfrance.fr/Acces-aux-Donnees/Catalogue#/metadata/6b796349-d56e-44c3-b572-d5488250637e

Data processing

The data are retrieved from the autopilot log file which includes all information, status and measurements done by the ASV during the survey. A first step is to discard any unnecessary data to keep only the echo-sounder, GPS, and IMU data over the survey area. To achieve an accurate depiction of the seabed, a pre-processing stage is required to correct and filter the measured depths. The raw data processing includes the following steps:

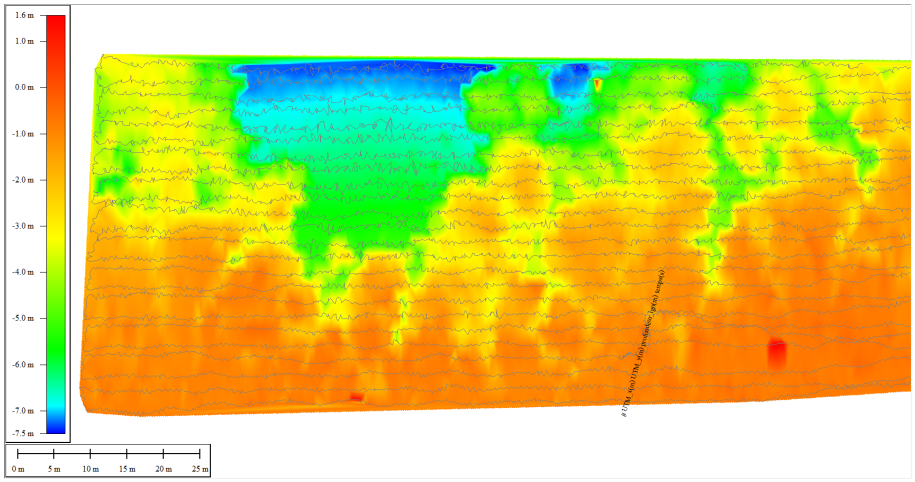
- From the ASV attitude (roll, pitch, yaw) given by the IMU sensor, all points for which the pitch and roll angles are greater than a defined 10° threshold are removed.
- Using a sliding median-filter, depth values that are outside a certain range around the median depth value computed along the sliding window are removed.
- GPS data with position offsets between the GPS antenna and the location of the echo-sounder on the ASV are corrected for the 3 axis.
- Retrieve the true location of the measured depth on the sea floor by correcting the surface GPS positions with ASV attitude.
- Correct the recorded depth values with the ASV attitude, the local datum and the geoid of the survey zone, to eventually get a compensated and georeferenced depth map.

A minimal working example in *Python* is available on the Git repository[26] associated with this article

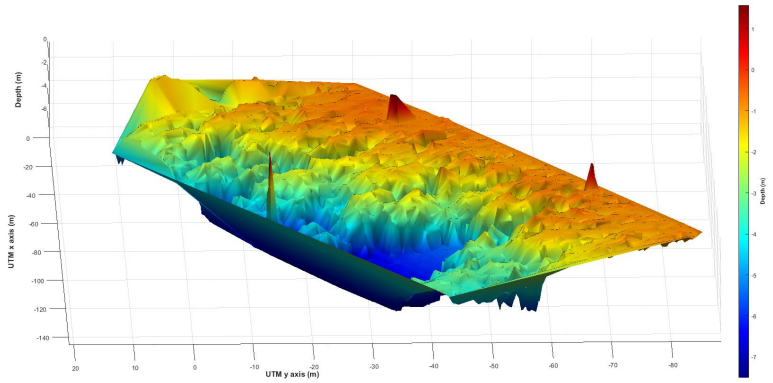
Results and comparison with prior data

For the survey mentioned above, Figure 1.8 shows different depth estimates of the same pre-processed data set. In Figure 1.8.a, the depth map has been automatically computed using the *Global Mapper* software. Overlaid gray lines represent the ASV path extracted from raw GPS data. Figure 1.8.b is a 3D-projection of the same bathymetric data set build with *MATLAB*[27].

[26]Example bathymetric data processing script in Git : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Sotfware/Bathymetry/Compute_depth.py



(a) Computed sea depth map with overlaid ASV paths (grey lines). Map generated with *Global Mapper*



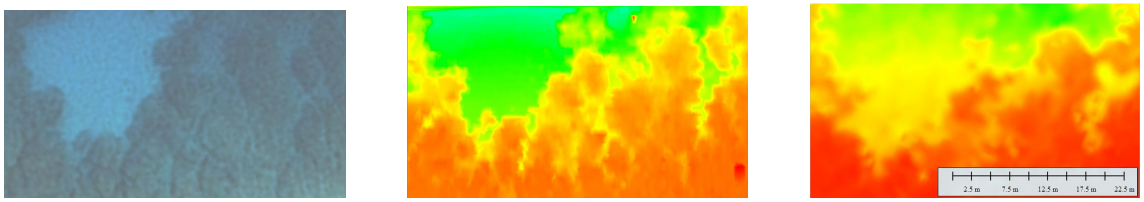
(b) Same bathmetric data with 3D-projection and Delaunay triangulation. Plot generated with *MATLAB*

FIGURE 1.8: Bathymetry results of a survey carried out in 2020 in Europa Island with the ASV

To illustrate the benefits of using a single-beam echo-sounder on such ASVs, we compare three different techniques that have been used to analyze the sea floor of the Europa lagoon (Figure 1.9). Maps are drawn for a portion of the survey area discussed before. Figure 1.9.a shows the satellite imagery of the surveyed area. Figure 1.9.b is a zoom on the map shown in Figure 1.8.a representing the ASV bathymetry data. Figure 1.9.c is the bathymetric data estimated from hyperspectral and LiDAR data collected in 2019 on the same area (Litto3D Océan Indien project).

A strict comparison of feature resolution and depth accuracy obtained with the three methods above is out-of-the-scope of this paper. Such analysis would require special attention to the different geodesic reference frames used, the level of depth correction applied, whether it includes or not environmental/experimental parameters (i.e. temperature, salinity, the effect of tides, ...), and eventually to the interpolations errors introduced by the different spatial distribution of each data set.

[27]Example script in git : https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Features_example/test_bathy_europa_09_10_20/code/main_plot_bathy_09_10_20.m



(a) Satellite imagery Pleiades © CNES 2019, distribution Airbus DS / Spotimage

(b) ASV's bathymetry with the ECT400 echo-sounder (2020 campaign)

(c) LIDAR / Hyper-spectral bathymetry (2010 campaign)

FIGURE 1.9: Three different representations of the sea floor in the survey area located inside the Europa lagoon to compare the results from the bathymetry estimated from the ASV data to the one estimated from hyperspectral and LiDAR data.

However, a qualitative analysis is enough to confirm that the ASV bathymetry gives an accurate depiction of the seabed topology in this area as compared to the satellite imagery. We observe a similar bathymetry between the ASV data and the hyperspectral/LiDAR data but with a higher level of details for the ASV bathymetry. Although aerial hyper-spectral techniques have the advantage of covering larger zones in a much shorter time, for smaller areas, deploying single-beam echo-sounders on such ASVs can be cheaper and a more practical solution with better resolution. Finally, mounting this type of echo-sounder on an ASV instead of a boat has the advantage of much regular and dense sampling patterns, as well as the opportunity to investigate areas where it is too shallow for navigation.

1.4.3 Photogrammetry survey

Camera images collected over the survey area can be used to obtain photogrammetry data. Here we described the protocol, the data processing, and the results for this type of surveys.

Protocol

-*Camera calibration:* To obtain the best possible results for the photogrammetry reconstruction, it is required to calibrate the camera. Indeed, to prevent lens distortion, the parameters of the lens and image sensor of the GoPro camera have to be estimated. For this calibration, multiple images of a 9 by 7 square chessboard pattern are taken in different positions and with varying angles. Camera parameters are set to full resolution. The photogrammetry software, *Matisse*, has a built-in calibration process which proposes to compute and save the camera model. We can choose between different *Distortion models* in the camera calibration settings to correspond to the fisheye distortion of the GoPro.

-*Mission preparation:* To obtain a three-dimensional reconstruction of the survey area, it is necessary that:

- Each image must have an overlap greater than 70% with other images

- photos are clear without surface effects on the seabed or ASV shadow

Using the survey and camera information (i.e., field of view of the camera, sea depth), it is possible to define the distance between transects that approximately satisfies the first condition. We propose a tool[28] to estimate this distance. It does not take into account the sampling frequency of the camera and the speed of the ASV. In the example given in this paper, the speed of the vehicle is set to 0.8 m/s, the distance between transects is set to 2 m, and the depth in the studied area varies between 2 and 4 m.

Data processing

Underwater images suffer from various color degradation (correlated with the depth at which the image was taken, light fluctuations due to sunlight refraction etc). Matisse 3D carries out color and illumination corrections in a pre-processing mode. In our case, since the illumination was pretty uniform, we have checked only the *Correct colors for underwater attenuation* option while limiting the size of the images to 4 megapixels.

Once this preprocessing terminated, the reconstruction with Matisse 3D can be run with the post-processing mode. In order to obtain the higher reconstruction resolution, we choose the *3D Dense* algorithm.

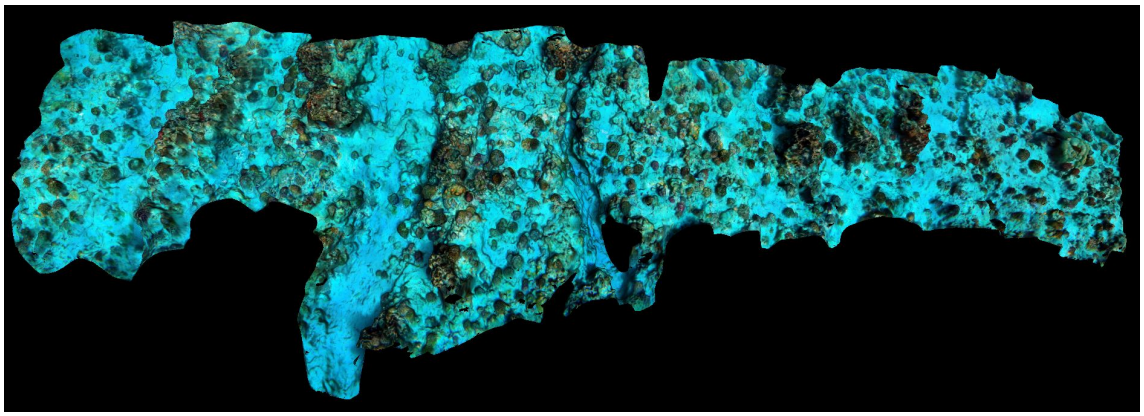
Matisse offers the possibility to use the GPS positions and orientations of the photo metadata in order to improve the result of the photogrammetry process. This piece of information is available through the ASV log but we do not use this functionality yet which need one more pre-processing step to set the metadata of the images.

Results

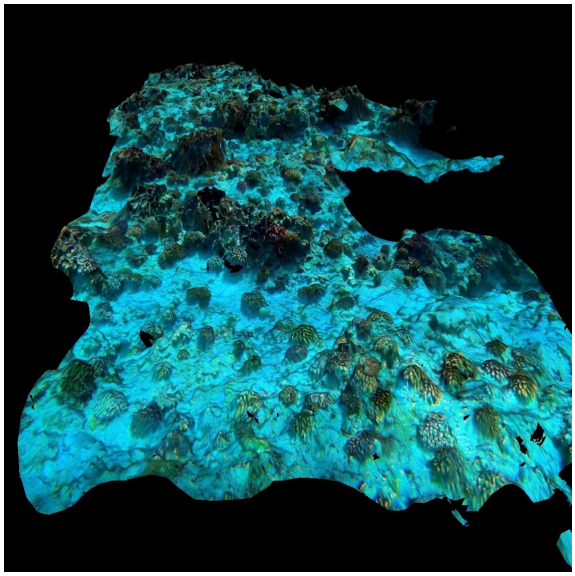
A result of a photogrammetry process on 70 images taken in Europa island is shown in Figure 1.10. Although the photos are all taken from the sea surface and the angle between the GoPro and the seabed remains unchanged (except for small variations due to the waves), the three-dimensional reconstruction can be performed and numerous elements characterizing the morphology of the study area can be identified (Figure 1.10). The geological faults are reconstructed, as well as numerous coral specimens of *Acropore Massive*, *digitised* and other elements such as a specimen of *Clam*.

It must be emphasized that the 3D reconstruction and the level of details are strongly correlated to the amount, position, and angle at which the photos were taken, i.e., some portions are simply not reconstructed (black areas on the image) due to a lack of images or are degraded (black spots on the side of corals).

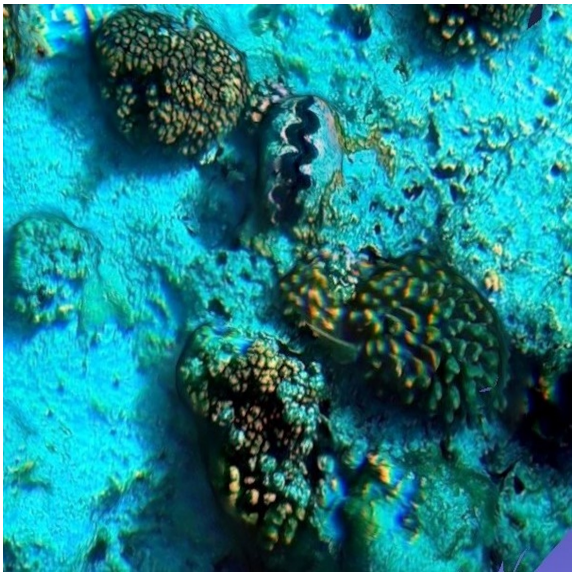
[28] https://gitlab.ifremer.fr/sb07899/Plancha-ASV/-/blob/main/Sotfware/Photogrammetry/Spacing_between_transect_calculator.xlsx



(a) Top view reconstruction



(b) Side view reconstruction



(c) Zoom on a giant clam and on a *Pocillopora* coral

FIGURE 1.10: Different views and zooms of the photogrammetry calculated from 70 images collected by the ASV during a field survey in Europa island in 2020.

1.5 Conclusion

This paper fully describes the hardware, software, and data processing tools for an autonomous surface vehicle. The ASV is able to perform:

- an autonomous navigation with an autopilot
- an autonomous acoustic tracking with an acoustic SBL system.
- a bathymetric survey with a single beam echosounder for depth < 50 m.
- a photogrammetric survey with a low-cost camera

All the components and mechanical parts are chosen to be low-cost, easy to find, and easy to build. Regarding softwares, the firmware, flight controller, and in-house development are open-source. There are limitations to the ASV. For example, it is not designed to be used in rough sea and weather conditions. The ASV has been deployed flipped over in windy conditions (>20 kt) and with small waves breaking ($\approx$0.3 m).

In parallel to the description and the validation sections, we provide a Git repository containing all the documents, instructions, and files to reproduce this ASV. We illustrate the different features exposed for our applicaton with dedicated field surveys. The ASV can be deployed in different environmental conditions, with or without internet. The radio telemetry system allows to control and operate the ASV with a few kilometers range. For inhabited coastal regions such as Reunion island, the ASV never loses its internet connection within the survey area (<1 km from the coastline). The consumption of the ASV allows more than 4 h of survey time with two 4S batteries (10 Ah each). These batteries are compliant with air transportation regulations and makes the board easy to travel with a surf bag.

To summarize, Plancha ASV is reliable, easy to use, reproducible, and adaptable. The system is small and light, and can operated by two operators which is advised to be able to recover the board in case of an issue. With telemetry and ground control software, the ASV can be followed in real-time during the survey with a laptop. This software also offers to create survey missions, change the parameters, and calibrate the ASV. The *Ardupilot* flight controller logs the flight data and makes analyses easy with the appropriate tools.

With its high buoyancy and the space available on the board, other sensors, batteries, and other functionalities can be added. New software integration is straightforward thanks to the *Raspberry pi* as a companion computer and *Pixhawk* 2.1 with *Ardupilot*.

These functions and features prove that low-cost ASV can be used for environmental and ecological purposes and provide accurate monitoring. As far as we know, this is the first time that an ASV is used to track an acoustic beacon using a low-cost SBL system. This ASV can be used to provide accurate fine-scale trajectories of underwater animals even on shallow depth and to simultaneously collect environmental information such as bathymetry and photogrammetry.

Acknowledgement

This work is supported by the "IOT" project funded by FEDER INTERREG V and Prefet de La Réunion: grant #20181306-0018039 and the Contrat de Convergence et de Transformation de la Préfecture de La Réunion. We are grateful to Steven Le Bars and Vincent Macaigne from ID OCEAN for their support and loan of material. Thanks to Denis De Oliveira, Geoffrey Fournier, Pauline Salvatico and Julien Fezandelle for their help in the project. Thanks Pascal Mouquet for the satellite imageries. Finally, we thank *Emlid* and *Blue Robotic* for letting us use some of their images.

Bibliography

[1] C. C. Wilmers, B. Nickel, C. M. Bryce, J. A. Smith, R. E. Wheat, and V. Yovovich, "The golden age of bio-logging: how animal-borne sensors are advancing the frontiers of ecology," *Ecology*, vol. 96, no. 7, pp. 1741–1753, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1890/14-1401.1

[2] M. Oliver, M. Moline, I. Robbins, W. Fraser, D. Patterson, and O. Schofield, "Letting Penguins Lead: Dynamic Modeling of Penguin Locations Guides Autonomous Robotic Sampling," *Oceanography*, vol. 25, no. 3, pp. 120–121, 2012. [Online]. Available: https://tos.org/oceanography/article/letting-penguins-lead-dynamic-modeling-of-penguin-locations-guides-autonomo

[3] F. A. Francisco, P. Nührenberg, and A. Jordan, "High-resolution, non-invasive animal tracking and reconstruction of local environment in aquatic ecosystems," *Movement Ecology*, vol. 8, no. 1, p. 27, Jun. 2020. [Online]. Available: https://doi.org/10.1186/s40462-020-00214-w

[4] A. M. Allen and N. J. Singh, "Linking Movement Ecology with Wildlife Management and Conservation," *Frontiers in Ecology and Evolution*, vol. 3, 2016. [Online]. Available: https://www.frontiersin.org/article/10.3389/fevo.2015.00155

[5] F. Siegwalt, S. Benhamou, M. Girondot, L. Jeantet, J. Martin, M. Bonola, P. Lelong, C. Grand, P. Chambault, A. Benhalilou, C. Murgale, T. Maillet, L. Andreani, G. Campistron, F. Jacaria, G. Hielard, A. Arqué, D. Etienne, J. Gresser, S. Régis, N. Lecerf, C. Frouin, F. Lefebvre, N. Aubert, F. Vedie, C. Barnerias, L. Thieulle, C. Guimera, M. Bouaziz, A. Pinson, F. Flora, F. George, J. Eggenspieler, T. Woignier, J.-P. Allenou, L. Louis-Jean, B. Chanteur, C. Béranger, J. Crillon, A. Brador, C. Habold, Y. Le Maho, J.-P. Robin, and D. Chevallier, "High fidelity of sea turtles to their foraging grounds revealed by satellite tracking and capture-mark-recapture: New insights for the establishment of key marine conservation areas," *Biological Conservation*, vol. 250, p. 108742, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0006320720308004

[6] D. P. Costa, G. A. Breed, and P. W. Robinson, "New Insights into Pelagic Migrations: Implications for Ecology and Conservation," *Annual Review*

of Ecology, Evolution, and Systematics, vol. 43, no. 1, pp. 73–96, 2012, _eprint: https://doi.org/10.1146/annurev-ecolsys-102710-145045. [Online]. Available: https://doi.org/10.1146/annurev-ecolsys-102710-145045

[7] R. M. Gunner, M. D. Holton, M. D. Scantlebury, O. L. van Schalkwyk, H. M. English, H. J. Williams, P. Hopkins, F. Quintana, A. Gómez-Laich, L. Börger, J. Redcliffe, K. Yoda, T. Yamamoto, S. Ferreira, D. Govender, P. Viljoen, A. Bruns, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, C. M. Duarte, M. C. van Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "Dead-reckoning animal movements in R: a reappraisal using Gundog.Tracks," *Animal Biotelemetry*, vol. 9, no. 1, p. 23, 2021. [Online]. Available: https://doi.org/10.1186/s40317-021-00245-z

[8] S. L. Cox, F. Orgeret, M. Gesta, C. Rodde, I. Heizer, H. Weimerskirch, and C. Guinet, "Processing of acceleration and dive data on-board satellite relay tags to investigate diving and foraging behaviour in free-ranging marine predators," *Methods in Ecology and Evolution*, vol. 9, no. 1, pp. 64–77, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12845

[9] M. Espinoza, T. J. Farrugia, D. M. Webber, F. Smith, and C. G. Lowe, "Testing a new acoustic telemetry technique to quantify long-term, fine-scale movements of aquatic animals," *Fisheries Research*, vol. 108, no. 2, pp. 364–371, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165783611000300

[10] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV Navigation and Localization: A Review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, Jan. 2014, conference Name: IEEE Journal of Oceanic Engineering.

[11] K. L. Dodge, A. L. Kukulya, E. Burke, and M. F. Baumgartner, "TurtleCam: A "Smart" Autonomous Underwater Vehicle for Investigating Behaviors and Habitats of Sea Turtles," *Frontiers in Marine Science*, vol. 5, 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2018.00090

[12] B. R. Page, R. Lambert, J. Chavez-Galaviz, and N. Mahmoudian, "Underwater Docking Approach and Homing to Enable Persistent Operation," *Frontiers in Robotics and AI*, vol. 8, 2021. [Online]. Available: https://www.frontiersin.org/article/10.3389/frobt.2021.621755

[13] R. P. Wilson, N. Liebsch, I. M. Davies, F. Quintana, H. Weimerskirch, S. Storch, K. Lucke, U. Siebert, S. Zankl, G. Müller, I. Zimmer, A. Scolaro, C. Campagna, J. Plötz, H. Bornemann, J. Teilmann, and C. R. McMahon, "All at sea with animal tracks; methodological and analytical solutions for the resolution of movement," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 54, no. 3, pp. 193–210, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967064507000045

[14] Y. L. Bras, J. Jouma'a, and C. Guinet, "Three-dimensional space use during the bottom phase of southern elephant seal dives," *Movement Ecology*, vol. 5, no. 1, p. 18, 2017. [Online]. Available: https://doi.org/10.1186/s40462-017-0108-y

[15] G. Elkaim, E. Decker, G. Oliver, and B. Wright, "Marine Mammal Marker (MAMMARK) Dead Reckoning Sensor for In-Situ Environmental Monitoring," in *2006 IEEE/ION Position, Location, And Navigation Symposium*, 2006, pp. 976–987, iSSN: 2153-3598.

[16] P. J. Wensveen, L. Thomas, and P. J. O. Miller, "A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales," *Movement Ecology*, vol. 3, no. 1, p. 31, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0061-6

[17] K. Aoki, M. Amano, K. Mori, A. Kourogi, T. Kubodera, and N. Miyazaki, "Active hunting by deep-diving sperm whales: 3D dive profiles and maneuvers during bursts of speed," *Marine Ecology Progress Series*, vol. 444, pp. 289–301, 2012.

[18] T. Narazaki, K. Sato, K. J. Abernathy, G. J. Marshall, and N. Miyazaki, "Sea turtles compensate deflection of heading at the sea surface during directional travel," *Journal of Experimental Biology*, vol. 212, no. 24, pp. 4019–4026, 2009. [Online]. Available: https://doi.org/10.1242/jeb.034637

[19] S. Andrzejaczek, A. C. Gleiss, K. O. Lear, C. B. Pattiaratchi, T. K. Chapple, and M. G. Meekan, "Biologging Tags Reveal Links Between Fine-Scale Horizontal and Vertical Movement Behaviors in Tiger Sharks (Galeocerdo cuvier)," *Frontiers in Marine Science*, vol. 6, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2019.00229

[20] R. M. Gunner, M. D. Holton, D. M. Scantlebury, P. Hopkins, E. L. C. Shepard, A. J. Fell, B. Garde, F. Quintana, A. Gómez-Laich, K. Yoda, T. Yamamoto, H. English, S. Ferreira, D. Govender, P. Viljoen, A. Bruns, O. L. van Schalkwyk, N. C. Cole, V. Tatayah, L. Börger, J. Redcliffe, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, H. J. Williams, C. M. Duarte, M. C. van Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "How often should dead-reckoned animal movement paths be corrected for drift?" *Animal Biotelemetry*, vol. 9, no. 1, p. 43, 2021. [Online]. Available: https://doi.org/10.1186/s40317-021-00265-9

[21] E. Bryant, *2D Location Accuracy Statistics for Fastloc RCores Running Firmware Versions 2.2 & 2.3, Technical Report TR01*, Aug. 2007.

[22] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, vol. 41, 2016.

[23] R.-j. Yan, S. Pang, H.-b. Sun, and Y.-j. Pang, "Development and missions of unmanned surface vehicle," *Journal of Marine Science and Application*, vol. 9, pp. 451–457, 2010.

[24] "Surfbee Flowseeker with Water NSW," Oct. 2021, section: Government. [Online]. Available: https://firetailrobotics.com/surfbee-flowseeker-with-water-nsw/

[25] P. Kimball, J. Bailey, S. Das, R. Geyer, T. Harrison, C. Kunz, K. Manganini, K. Mankoff, K. Samuelson, T. Sayre-McCord, F. Straneo, P. Traykovski, and H. Singh, "The WHOI Jetyak: An autonomous surface vehicle for oceanographic research in shallow or dangerous waters," in *2014 IEEE/OES Autonomous Underwater Vehicles (AUV)*, 2014, pp. 1–7, iSSN: 2377-6536.

[26] G. Ferri, A. Manzi, F. Fornai, F. Ciuchi, and C. Laschi, "The HydroNet ASV, a Small-Sized Autonomous Catamaran for Real-Time Monitoring of Water Quality: From Design to Missions at Sea," *IEEE Journal of Oceanic Engineering*, vol. 40, no. 3, pp. 710–726, 2015, conference Name: IEEE Journal of Oceanic Engineering.

[27] D. Manda, M.-W. Thein, A. D'Amore, and A. Armstrong, "A Low Cost System for Autonomous Surface Vehicle based Hydrographic Survey," 2015.

[28] G. T. Raber and S. R. Schill, "A low-cost small unmanned surface vehicle (sUSV) for very high-resolution mapping and monitoring of shallow marine habitats," in *Remote Sensing of the Ocean, Sea Ice, Coastal Waters, and Large Water Regions 2019*, vol. 11150. SPIE, 2019, pp. 14–23. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11150/1115004/A-low-cost-small-unmanned-surface-vehicle-sUSV-for-very/10.1117/12.2531361.full

[29] R. Lambert, B. Page, J. Chavez, and N. Mahmoudian, "A Low-Cost Autonomous Surface Vehicle for Multi-Vehicle Operations," in *Global Oceans 2020: Singapore – U.S. Gulf Coast*, 2020, pp. 1–5, iSSN: 0197-7385.

[30] "Boogie Board Boat — Rover documentation." [Online]. Available: https://ardupilot.org/rover/docs/reference-frames-boogieboard-boat.html

[31] A. Martorell, M. Massot-Campos, E. Guerrero, and G. Oliver, "Xiroi ASV: a Modular Autonomous Surface Vehicle to Link Communications," *IFAC-PapersOnLine*, vol. 51, pp. 147–152, 2018.

[32] S. Le Bars, "IDOCEAN 2021 | TOUTES NOS ACTUS." [Online]. Available: https://www.idocean.re/toutes-nos-actus?pgid=k61sajlx-5ed7dde0-8e09-4ed9-85de-486ab46abe54

[33] Y. Zhao, Z. He, G. Li, Y. Wang, and Z. Li, "Design and Application of a Small ROV Control System Based on ArduSub System," in *2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT*, 2020, pp. 585–589.

[34] Z. He, Y. Chen, Z. Shen, E. Huang, S. Li, Z. Shao, and Q. Wang, "Ard-muCopter: A Simple Open Source Quadcopter Platform," in *2015 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, 2015, pp. 158–164.

[35] P. J. Burke, "A 4G-Connected Micro-Rover With Infinite Range," *IEEE Journal on Miniaturization for Air and Space Systems*, vol. 1, no. 3, pp. 154–162, 2020, conference Name: IEEE Journal on Miniaturization for Air and Space Systems.

[36] "Echo sounding," Mar. 2022, page Version ID: 1075538161. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Echo_sounding&oldid=1075538161

[37] D. F. Carlson, A. Fürsterling, L. Vesterled, M. Skovby, S. S. Pedersen, C. Melvad, and S. Rysgaard, "An affordable and portable autonomous surface vehicle with obstacle avoidance for coastal ocean monitoring," *HardwareX*, vol. 5, p. e00059, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2468067219300161

[38] M. Vlachos, L. Berger, R. Mathelier, P. Agrafiotis, and D. Skarlatos, "Software comparison for underwater archaeological photogrammetric applications," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W15, pp. 1195–1201, 2019.

[39] "Photogrammetry," Jun. 2022, page Version ID: 1091964951. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Photogrammetry&oldid=1091964951

[40] P. Drap, *Underwater Photogrammetry for Archaeology*. IntechOpen, 2012, publication Title: Special Applications of Photogrammetry. [Online]. Available: https://www.intechopen.com/chapters/36197

[41] J. Pollio, "Applications of Underwater Photogrammetry," naval oceanographic office nstl station ms, Tech. Rep., 1968, section: Technical Reports. [Online]. Available: https://apps.dtic.mil/sti/citations/AD0851692

[42] R. Ferrari, W. F. Figueira, M. S. Pratchett, T. Boube, A. Adam, T. Kobelkowsky-Vidrio, S. S. Doo, T. B. Atwood, and M. Byrne, "3D photogrammetry quantifies growth and external erosion of individual coral colonies and skeletons," *Scientific Reports*, vol. 7, no. 1, p. 16737, 2017. [Online]. Available: https://www.nature.com/articles/s41598-017-16408-z

[43] G. Marre, F. Holon, S. Luque, P. Boissery, and J. Deter, "Monitoring Marine Habitats With Photogrammetry: A Cost-Effective, Accurate, Precise and High-Resolution Reconstruction Method," *Frontiers in Marine Science*, vol. 6, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2019.00276

[44] W. C. Million, S. O'Donnell, E. Bartels, and C. D. Kenkel, "Colony-Level 3D Photogrammetry Reveals That Total Linear Extension and Initial Growth Do Not Scale With Complex Morphological Growth in the Branching Coral, Acropora cervicornis," *Frontiers in Marine Science*, vol. 8, 2021. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2021.646475

[45] A. Lavy, G. Eyal, B. Neal, R. Keren, Y. Loya, and M. Ilan, "A quick, easy and non-intrusive method for underwater volume and surface area evaluation of benthic organisms by 3D computer modelling," *Methods in Ecology and Evolution*, vol. 6, no. 5, pp. 521–531, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12331

[46] M. Berden-Zrimec, M. Poklar, D. Mozetič, and D. Vranac, *Seabed habitat mapping with underwater photogrammetry*, 2017.

[47] M. Johnson-Roberson, O. Pizarro, S. B. Williams, and I. Mahon, "Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys," *Journal of Field Robotics*, vol. 27, no. 1, pp. 21–51, 2010. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.20324

[48] A. Arnaubec, J. Opderbecke, A.-G. Allais, and L. Brignone, *Optical mapping with the ARIANE HROV at IFREMER: The MATISSE processing tool*, 2015, pages: 6.

[49] I. H. Organization, *IHO Standards for Hydrographic Surveys, Edition 6.0.0*, 2020.

Chapter 2

Dead-reckoning configuration analysis in a controlled environment

Authors : Gogendeau Pierre, Bonhommeau Sylvain, Fourati Hassen, De Oliveira Denis, Taillandier Virgil, Goharzadeh Andrea and Bernard Serge

Article status : Published in *IEEE Sensors Journal (Volume: 22, Issue: 12, 15 June 2022)*

Presentation : Presented at 7th Bio-Logging Symposium, Hawaii (BLS 2022)

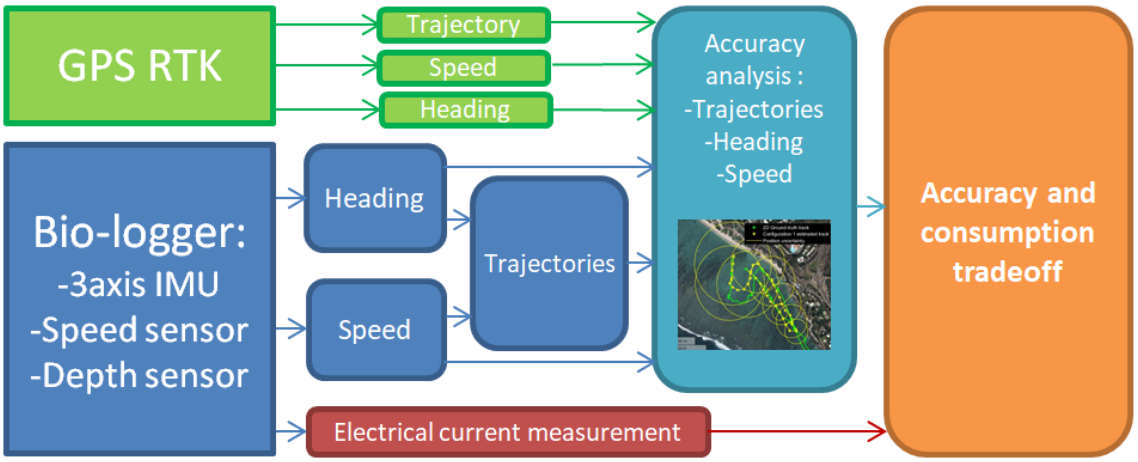
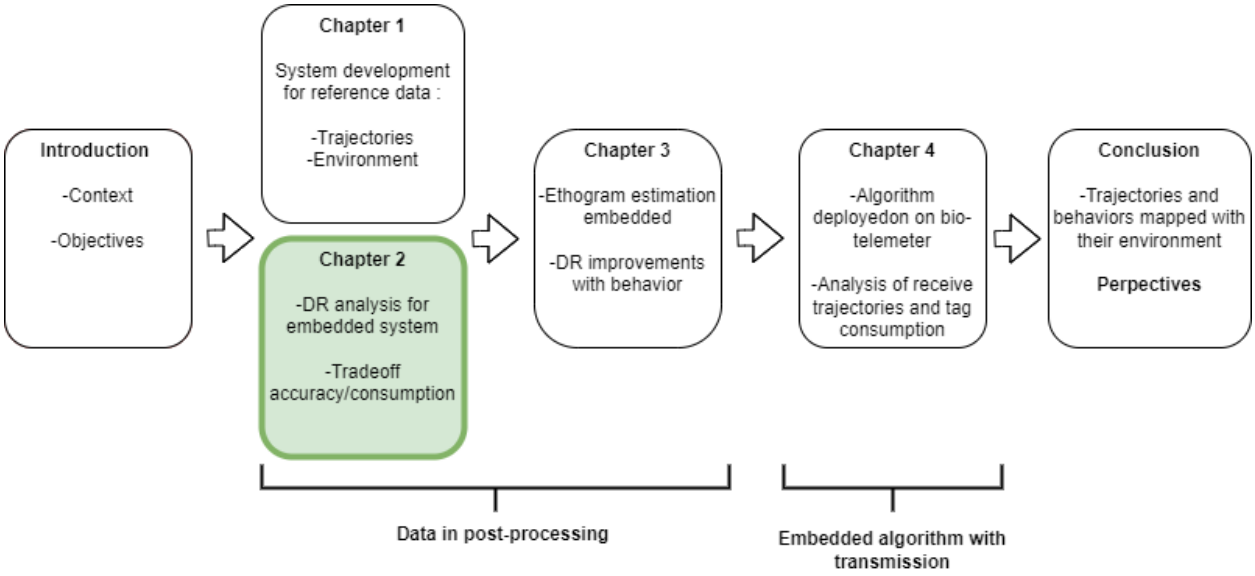


FIGURE 2.1: Graphical abstract

Keywords: Bio-logging, Bio-Transmitter, Dead-Reckoning, Trajectory, Accuracy, Inertial and Magnetic Sensors, GPS RTK



Reminder of the chapter transition graph. The current chapter is highlighted in green

Abstract

In the past few years, dead-reckoning (DR) has been frequently used to estimate the trajectory of marine animals at a fine temporal scale using bio-logger devices. The precision of the swim sequence trajectory estimation depends on various accumulated errors from external forces, sensors and computation. Trajectory accuracy is hard to estimate due to the difficulty of collecting precisely-known underwater positions. In this paper, we aim at estimating this accuracy at a fine temporal scale using a reference system for positioning. This work focuses on how each sensor frequency and algorithm used for the DR affect trajectory accuracy and the global power consumption of the bio-logger. We develop a dual GPS Real Time Kinematic (RTK) system offering us reference trajectories with 2 cm accuracy on position and 1.6° on heading. The DR algorithms use 3-axis Inertial Measurement Unit (IMU), depth and speed sensor data for orientation and speed determination. For the experimental tests, the GPS module and the bio-logger are attached to a swimmer doing breaststroke imitating turtle movement for different swim sequences between 15 and 40 minutes. Power consumption of the electronics is measured during laboratory tests. Results show that using an adapted speed sensor and correcting for marine current, even roughly, provide us with the best gain in accuracy. The use of the gyroscope or high-frequency sampling of sensors does not increase the accuracy of the trajectory reconstruction to a level that would be critical for slow moving marine animal applications.

2.1 Background

The study of marine animal movement at fine temporal scales provides scientists with a large amount of information for foraging behavior, movement ecology, anthropogenic disturbance and swimming kinematics. In particular, movement information associated with physiology and behavioral variables allows researchers to address a large panel of scientific questions. For instance, Fukuoka et al.(1) reconstructs 3D trajectories of marine turtles for a few days to study the distance traveled between prey encounters. This allows them to find changes in foraging behaviors and habitats.

Depending on the scientific questions and studied animals, different levels of accuracy and deployment durations are required or possible. In any case, knowing the uncertainty on the position is important to interpret the results. Few studies provide uncertainty estimates, especially at fine temporal scale.

The present study is part of a project which aims at estimating the trajectory of juvenile marine turtles as well as its uncertainty. Our constrains are the difficulty to recapture the animal and to get long deployment over months. To overcome them, we use bio-transmitter with kilometers range transmission and low-consumption methods to estimate animal trajectory and transfer the data. The goal is to get embedded algorithms and hardware easily adaptable for other marine species.

For our biologging application, only acoustic positioning and inertial positioning through Dead Reckoning (DR) would be possible with small and low cost sensors.

Acoustic systems can give accurate positioning but need a very dense receiver array in a really short baseline (2). For instance, Espinoza et al.(3) geolocated an acoustic tag with 4.09 $\pm$ 2.53 m with an array of 16 receivers separated by 250 m using VEMCO Positioning System (VPS). These acoustic systems are not relevant for long distance range study cases. DR allows a reconstruction of the 3D path at a fine temporal scale for longer range using heading, depth and speed of the animal (4; 5; 6; 7).

Heading is calculated with inertial sensors by the combination of 3-axis accelerometer and 3-axis magnetometer data and can be corrected with an additional 3-axis gyroscope (8; 9; 10; 11; 12; 13). Speed can be determined by speed sensors: propellers (14; 15; 16), flexible paddle (17), derived from dynamic acceleration of the animal (18; 5) or with animal pitch and change of depth (6; 19). DR gives a position estimation for each temporal step without any gap in path reconstruction. This is an iterative method that uses each last position estimation which induces cumulative error over time.

Various sources of errors exist (4; 20). Estimation of these errors is an important but complex task due to the difficulty to get accurate reference data of trajectory in marine environment (21). A DR position estimate without its uncertainty can lead to false conclusions, in particular when positions are analyzed with environmental features. In most studies on air-breathing marine animals using GPS or Argos positioning, the DR error is calculated using these positioning data (22; 6; 5). Error in dive trajectories estimation is calculated with the distance between estimated position from DR and geolocated positions. This distance is called “drift” (23; 4; 21) and often simplified as the error from the marine current. GPS and Argos locations used as references are also subject to error and their uncertainty can be high (24).

Embedded systems are highly dependent on their electrical consumption. In bio-logging the consumption comes mostly from the sensors used, their sampling frequencies and the communication system. For instance, several algorithms are used to determine the orientation of the animal. Computing can be made with the information from accelerometers and magnetometers whereas others use additional gyroscopes increasing the power consumption (11; 12).

Most of the studies analyzing 3D tracks at fine temporal scale (except acoustic) uses DR from bio-logger and performs the computation in post processing from raw data, e.g. for pinniped (23; 10), whales (6; 25), turtles (26), and fishes (7). In general, sensors used are 3-axis accelerometer, 3-axis magnetometer, pressure sensor and have second or infra second sampling frequency and deployment duration is from several hours to weeks. Raw dataset samples represent a large amount of data which are impossible to transmit through satellite (27).

Bio-transmitter refers to tags allowing communications. Different communication techniques are available: satellite, radio, and acoustic. As explained above, acoustic communication is not suitable for our case due to the large habitat of the marine species we focus on. Recently, to overcome the low data rate of satellite, new technology and algorithms for Argos bio-transmitter have been emerging and allow sending more information (28; 27). In (27), the device sends 5 inflection points of a dive profile linked with the effort of the animal. Radio communication as WiFi or Bluetooth (29; 30) are possible but they don't match with our kilometer range constraints.

For a project focusing on juvenile marine turtles, we developed a new generation of bio-transmitter using LoRa radio communication (31). This "IoT tag" can send 200-byte messages every 3 s at the surface with about 500 ms of air-time. This new technology allows us to rethink the way of using bio-transmitters and information that we are able to send. For a 20-min dive, a 3D track and its uncertainty can be sent using a 4-byte encoded position information with a 30-second temporal step.

Facing the difficulty to get ground-truth data at fine temporal scale to estimate DR error, we develop testing methods and hardware to get this information. In this work, we focus on the comparison analysis between the accuracy of the trajectory estimation and its power consumption. We describe a method to calculate the error given by fine temporal scale DR for different sensors and sampling frequency. For each configuration, errors from the heading and speed estimation are analyzed. We propose a simple method to estimate the marine current and examine its influence on trajectory accuracy. We calculate the position uncertainty and map it along the path, simulating the bio-transmitter payload capability.

Then, for the different DR tracks, estimated accuracy is associated with power consumption. These results allow us to analyze the trade-off between accuracy and power consumption.

The developed hardware and algorithms are designed to be embedded in bio-transmitters for marine turtles but are generic and adaptable for other bio-logger studies and a wide range of marine species. All the scripts, model and example data are available in Git-Hub[1].

[1] https://github.com/pierregoge/dead_reckoning_analysis

2.2 Materials for Experiments

2.2.1 Operational context

An extensive inertial dataset associated with speed data for different swim sequences is required for the present study. We use a commercial bio-logger with a speed sensor: Openlogger tag from loggerhead instruments (https://github.com/loggerhead-instruments/OpenTag3). The logger is composed of an inertial measurement units (IMU) with 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer (MPU9250 with a sampling at 100Hz), pressure/temperature sensors (MS5837 with a sampling at 1Hz), speed sensor (14) (AN48846B Hall effect sensor with a sampling at 1Hz), an SD card of 32 GB to record data. The size of the logger is $9.6x4.5x2.4$ cm, and its weight is 80 g in water.

Swim sequences occurred in the La Saline Les Bains lagoon in Reunion island (-21.089827°N, 55.229509°E). This area allows swim sequences horizontal or near horizontal due to the shallow depth. This area is also an habitat for juvenile green (*Chelonia mydas*) and hawksbill (*Eretmochelys imbricata*) turtles.

Swim sequence	Duration (min)	Distance (m)	Location
S1	40	1335	Lagoon
S2	15	420	Lagoon
S3	30	889	Lagoon
S4	25	725	Swimming pool

TABLE 2.1: Summary of the characteristics for the 4 swim sequences including the duration, distance and location

2.2.2 Mimicking turtle movement in a controlled environment

For these experiments, a controlled environment without disturbing and impacting marine species with capture and tag deployment is preferable. For scientific ethics, disturbing wild animals must be done as a final solution. A swimmer with a movement close to a turtle is sufficient for the purpose of this analysis.

Accurate ground truth data is essential to analyze trajectories and better understand the various sources of errors. The kinematics of turtle swimming is described in (32). They use the front flipper to propel frontward and back flippers to navigate.

To be as close as possible to a turtle's movement, we imitate movement, frequency and speed of juvenile green turtle data (n=4, 120 hr.) obtained in the Sanriku Coast, Japan (Fukuoka et al. 2019). The data includes inertial and speed measurements of 4 juvenile green turtles for 120 hours. Turtles are less impacted by current than swimmers due to the hydrodynamic shape of the shell and body. To overcome this difference, the swimmer does breaststroke on a paddle board with a single large fin. The logger is attached under the board.

2.2.3 Dual GPS RTK ground-truth data

We use a dual GPS RTK system (https://docs.emlid.com/reachm2/) mounted on top of the board as ground-truth system. This GPS system gives precise position and heading references with 1 cm of horizontal accuracy. The position is derived to get speed reference. Separated by 70 cm, the two GPS antennas provided us with a 1.6° accuracy for the heading calculation. This system provides us with accurate reference data for speed and heading determination at 10Hz. The reference system average power consumption is 400 mA at 5V (given by the datasheet). The dual GPS allows heading estimation in the turtle frame. However, speed estimation is in the earth frame and includes the current influences (Figure 2.2).

2.3 Proposed Method for Trajectory reconstruction and analysis

The objective of this paper is to analyze the differences in terms of accuracy and power consumption of several algorithms used for trajectory reconstruction, executed at different sample rates. The proposed method is based on DR and needs two variables to be calculated: orientation (also called attitude) and speed.

2.3.1 2D Path reconstruction

Path reconstruction is made with DR on a horizontal plane (2D). It is a typical simplification as the actual depth is measured directly with the pressure sensor of the bio-logger. The datasheet for this pressure sensor gives its error, drift, and uncertainty. They are not accumulated over time. The 2D path vector is defined as $[x(t);\ y(t)]$ at the time t of the swim sequence. The Earth projection used is North-East-Down (NED). The X-axis is northward, Y-axis is eastward, and the Z-axis is downward. The IMU sampling frequency gives the temporal step Δt. The DR formula is equal to:

$$x(t) = x(t - \Delta t) + S_{turtle}(t) \times \Delta t \times cos(\theta_{turtle}(t)) + decli) \tag{2.1}$$

$$y(t) = y(t - \Delta t) + S_{turtle}(t) \times \Delta t \times sin(\theta_{turtle}(t)) + decli) \tag{2.2}$$

where $S_{turtle}(t)$ is the speed in turtle frame at time t, $\theta_{turtle}(t))$ corresponds to the heading in turtle frame at time t and $decli$ is the magnetic declination for the experimental field.

2.3.2 Data processing

Data is processed using MATLAB (2020b) and customized scripts. Algorithms are designed to be embedded on the bio-transmitter. Accelerometer data is analyzed with frequency spectrum and scalogram in order to find movement frequency before filtering. Ground-truth data from dual GPS module and logger data are pre-processed and standardized to be easily analyzed.

2.3.3 Orientation estimation

In bio-logging application for turtle tracking, we need light and easy-to-implement orientation estimation algorithms. Two different algorithms are studied, one using accelerometers and magnetometers data and another adding gyroscope.

For the DR we need the turtle heading projected in the horizontal plan. This projection uses pitch and roll orientation components. The easiest way to calculate the latter is using the gravitational acceleration from the accelerometer, when it is at rest or moving slowly. However, during dynamic phases, the measured acceleration from the accelerometer is also subject to linear acceleration, which even partially filtered can false the pitch and roll calculation. To increase the accuracy, it is possible to fuse accelerometer and gyroscope data (angular velocity). Nevertheless, gyroscopes induce a high electrical consumption compared to the accelerometer and magnetometer only. Algorithm complexity is critical in a embedded application with low computation resources. Complex algorithms with many operations cannot run in parallel with the main program of the tag. In addition, to be used by other scientists with different IMU, it should have a few easily configurable parameters.

We choose the following algorithms: Madgwick (9) and SAAM (Super-fast attitude from accelerometer and magnetometer) (8). The Madgwick algorithm is widely used and has the advantage of not requiring sensor noise information as input, and has only one parameter to adjust. This algorithm has also been chosen because it is computationally inexpensive. Madgwick needs 277 scalar arithmetic operations for each update step and SAAM 37 operations. Number of floating operation is prevailing for embedded application. Both implementations are already made and optimized for MATLAB, C, and C++. We present here briefly these two algorithms:

(1) SAAM algorithm: To estimate the attitude, it uses a simplified version of Davenport's solution for solving Wahba's problem with the magnetic and gravitational reference vector. The solution reduces the number of operation for almost the same accuracy (8). The accelerometer samples the gravitational and dynamic acceleration. In order to compute the orientation, the gravitational acceleration is extracted. A low-pass filter is used at 0.3 Hz for the three axes of the sensor. This value is found by analyzing the swimmer data. The filter is not adaptive and applied during all the analysis.

(2) Madgwick algorithm: This filter needs gyroscope data in addition to the accelerometer and magnetometer data. Accelerometer and magnetometer data is used with a gradient descent algorithm to compute the direction of the gyroscope measurement error. This filter has one parameter β to tune. It corresponds to the weight on gyroscope measurement error. In the present analysis, the reference heading parameter β is set to 0.02. We set β after trial and error tests made with the estimated and references headings.

2.3.4 Speed calculation

Speed is estimated using a speed sensor or inertial data with overall dynamic body acceleration (ODBA) (33). We calibrated speed functions in a controlled environment without external force perturbation.

The first method uses Openlogger's speed sensor. It is composed of a magnetic propeller and a Hall effect sensor. This sensor works as the spin counter of the

propeller and gives the rotation per second (rps).The sampling frequency of the sensor is 1 Hz. We use interpolation and a moving average filter to get 10 Hz and 100 Hz value. To estimate the speed from this sensor information, we perform a linear regression using GPS RTK speed as reference.

The second method uses only the Overall Dynamic Body Acceleration (ODBA). The ODBA is the sum of absolute values of the dynamic accelerations sensed from the three axes. It was first developed for energy expenditure proxies of animals (33), and can be used to calculate speed (5). We perform a linear regression for the speed function with the ground-truth data from GPS RTK.

As speed ground-truth includes marine current (Figure 2.2), we need to calibrate speed in a place without external perturbation. We made calibration of our models at the public swimming pool of Le Port (Reunion Island) during swim sequence S4 (Table 2.1)

2.3.5 Dead-reckoning configurations

Several sampling rates are tested for the sensors. IMU are sampled at 100 Hz and downsampled to 10 Hz. Speed methods using inertial data is computed at the same sampling rate than IMU. Speed and depth sensors are sampled at 1 Hz and then interpolated and filtered to get 100 Hz and 10 Hz value. Table 2.2 gives the experiment number and presents the 8 tested combinations for the 2 sampling frequencies, the 2 speed estimation method, and the 2 algorithms used.

Orientation	Madgwick		SAAM	
	100Hz	10Hz	100Hz	10Hz
Speed : Sensor	C1	C2	C3	C4
Speed : ODBA	C5	C6	C7	C8

TABLE 2.2: Number of the different DR configurations tested

2.3.6 Error calculation and accuracy metrics

Heading and speed estimation errors are calculated with the ground-truth data. We use horizontal position error as an accuracy metric to compare the different DR configurations.

To measure the estimation error of the speed and heading, we use the root mean square of the error (RMSE). For all the DR configurations, ground truth is compared with the estimated data. Heading θ and speed S errors are defined as:

$$RMSE_{\theta/S} = \sqrt{\frac{\sum_{i=1}^{n}(\Delta x(i)_{\theta/S})^2}{n}} \tag{2.3}$$

Where is n number of samples of the swim sequence, $\Delta x(i)_{\theta/S}$ is the difference between the i heading or speed iteration calculated and the ground-truth value of the heading or speed i iteration.

$RMSE_\theta$ includes the error of sensor measurement, sensor drift, computation, and inertial data filtering.

$RMSE_S$ includes error of marine current, sensor measurement, sensor drift, computational error, and accelerometer filtering.

To compare the accuracy of the different configurations, we use the 2DRMS (34). The estimated horizontal position and the ground-truth GPS data is compared in NED frame. The 2DRMS horizontal is defined as:

$$2DRMS = \sqrt{\frac{\sum_{i=1}^{n}(\Delta E(i)^2 + \Delta N(i)^2)}{n}} \tag{2.4}$$

Where $\Delta E(i)$ and $\Delta N(i)$ are the errors in the East and North components of the i-th position estimate sample.

2DRMS includes the marine current-induced error, algorithms estimation errors, sensor measurement error, sensor drift, and filtering errors.

For position uncertainty, we made a simplification using the 2DRMS value of the swim sequence divided by number of samples. This value is used to map uncertainty circles growing over the number of samples on the estimated trajectory.

2.3.7 Marine current measurement and analysis

Marine current highly impacts trajectories estimation. A simple method is proposed to estimate and analyze its influence on accuracy.

For each step of the trajectory estimation, displacement in the Earth frame is composed of the swimmer displacement estimation and the marine current drift (Figure 2.2). The DR gives the swimmer trajectory. Depending on the current strength, it can provide a significant error in the estimation. We propose a method to measure it and its influence.

Marine current measurement is a complex task that necessitates expensive oceanographic equipment. We simplify its measurement by calculating the drift during resting phases. For each resting stop during the different swim sequences, the direction and strength of the drift are calculated. This average drift is used as the marine current vector with North and East components.

$$VecC_N = S_{current} \times cos(\theta_{current}) \tag{2.5}$$

$$VecC_E = S_{current} \times sin(\theta_{current}) \tag{2.6}$$

Where $VecC_N$, $VecC_E$ are the current vector in northward and eastward directions, $S_{current}$ is the strength of the current and $\theta_{current}$ is the direction of the current

Contrary to $RMSE_\theta$, $RMSE_S$ includes the marine current error (Figure 2.2). To analyze the speed-accuracy with the marine current knowledge, we add the current estimation to the speed and compare it with the speed ground-truth.

$$Scorr_N(t) = S_{turtle}(t) \times cos(\theta_{turtle}(t)) + VecC_N \tag{2.7}$$

$$Scorr_E(t) = S_{turtle}(t) \times sin(\theta_{turtle}(t)) + VecC_E \tag{2.8}$$

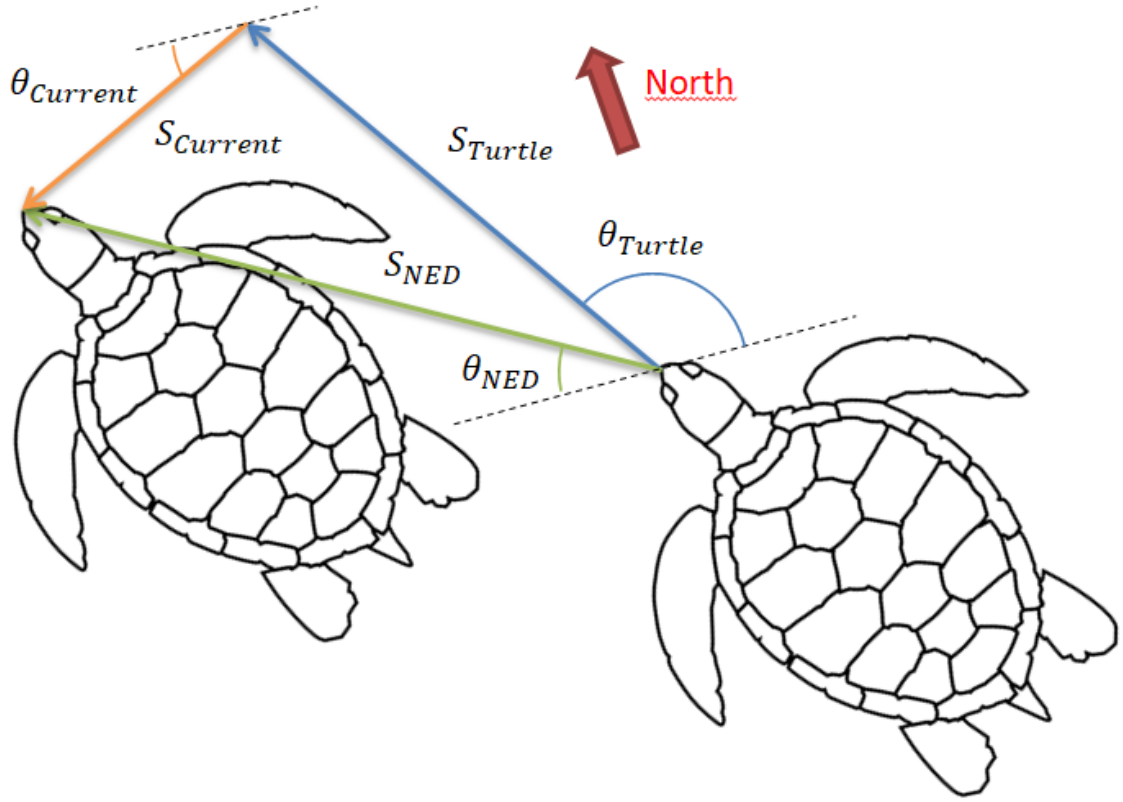


FIGURE 2.2: Turtle displacement on NED and turtle frame for one temporal step. Blue arrow represents estimated displacement in turtle frame. Dual GPS RTK give the θ_{turtle}. Green arrow is the real displacement of the turtle measured by the GPS RTK. Orange arrow is the displacement induced by the marine current

Where $Scorr_E(t)$ and $Scorr_N(t)$ are the corrected speed of the turtle in the NED frame. Then, the corrected current speed is given by :

$$Scorr_{turtle}(t) = norm(Scorr_N(t)), Scorr_E(t)) \tag{2.9}$$

Using the corrected speed, we calculate the new DR trajectory.

2.3.8 Power consumption

The power measurement's goal is not to precisely know the consumption for specific hardware but to underline the relative consumption between sensors and configurations of the biologging tag using consumer-grade hardware. All the power consumption are not available on the datasheet for the different electronic components. We thus made a laboratory test using a power analyzer: N6705B. Due to the impossibility of doing the power test on the Openlogger tag which has an electronic board embedded in resin, we used our "IoT tag" board which as the same grade of sensors (i.e. IMU: LSM303AGR, µc: STM32L082, pressure sensor: MS5837-30BA). We use the power consumption from the component datasheet for the gyroscope and the speed sensor consumption. We develop an example code for each configuration and

sample rate of the sensors. The electrical current drawn is measured and averaged. The current consumption is given in mA.

2.4 Results

2.4.1 Swim comparison with turtles

To ensure that our results are applicable to marine species such as marine turtles, we focus on active displacement phases of the human swimmer with average speeds and frequencies similar to turtles behaviors (Tables 2.3). From the data collected by (1), marine turtle swim speed and frequencies are close to the human swimmer variable with 0.13 Hz of difference for the frequency and 0.22 m/s for the speed. The speed is a bit higher for the human swimmer to ensure a proper propeller rotation of the speed sensor of the OpenLogger tag.

Swimmer type	Frequency (Hz)	Avg speed (m/s)
Turtle	0.45	0.30
Human	0.32	0.52

TABLE 2.3: Comparison in swimming characteristics between turtle and human swimmer during active phases

2.4.2 Error analysis

Every lagoon sequence (S1, S2 and S3) was analysed separately. We reconstruct the 2D track for each swim sequence using the 8 configurations described Table 2.2. As a pressure sensor is generally included in marine species tags, the reconstruction of depth is not necessary. Section 2.3.5 contains details and numbers of the different configurations. Only swim sequence S1 is displayed as an illustration. The other swim sequences results are available on Table 2.4 and all the figures in the supplementary materials[2].

For clarity, only 2 configurations are displayed, showing the different orientation and speed algorithms. Estimates for configuration 1 and 7 are given in Figure 2.3 for the swim sequence S1. On inset a), the ground-truth track and 2 analyzed 2D tracks are displayed and show the differences between predicted trajectories and the ground-truth data from the dual GPS RTK system. Inset b) is the speed estimation error compared to the ground truth.

If we focus on inset b), the ODBA speed error is higher around resting phases (1-7). These large errors are because the swimmer actively moves on the board at the beginning and end of these phases. This movement is made to simulate the high activity when the turtle is landing and launching off the sea bed. The static part of the speed error during resting phases is due to the marine current for both methods. Speed RMSE ($RSME_S$) are given in Table 2.4. Generally, the ODBA method is subject to all external disturbances giving accelerations like shocks or contact with the environment. For instance, on swim sequence 3, RMSE with ODBA is

[2] https://github.com/pierregoge/dead_reckoning_analysis/tree/main/Document

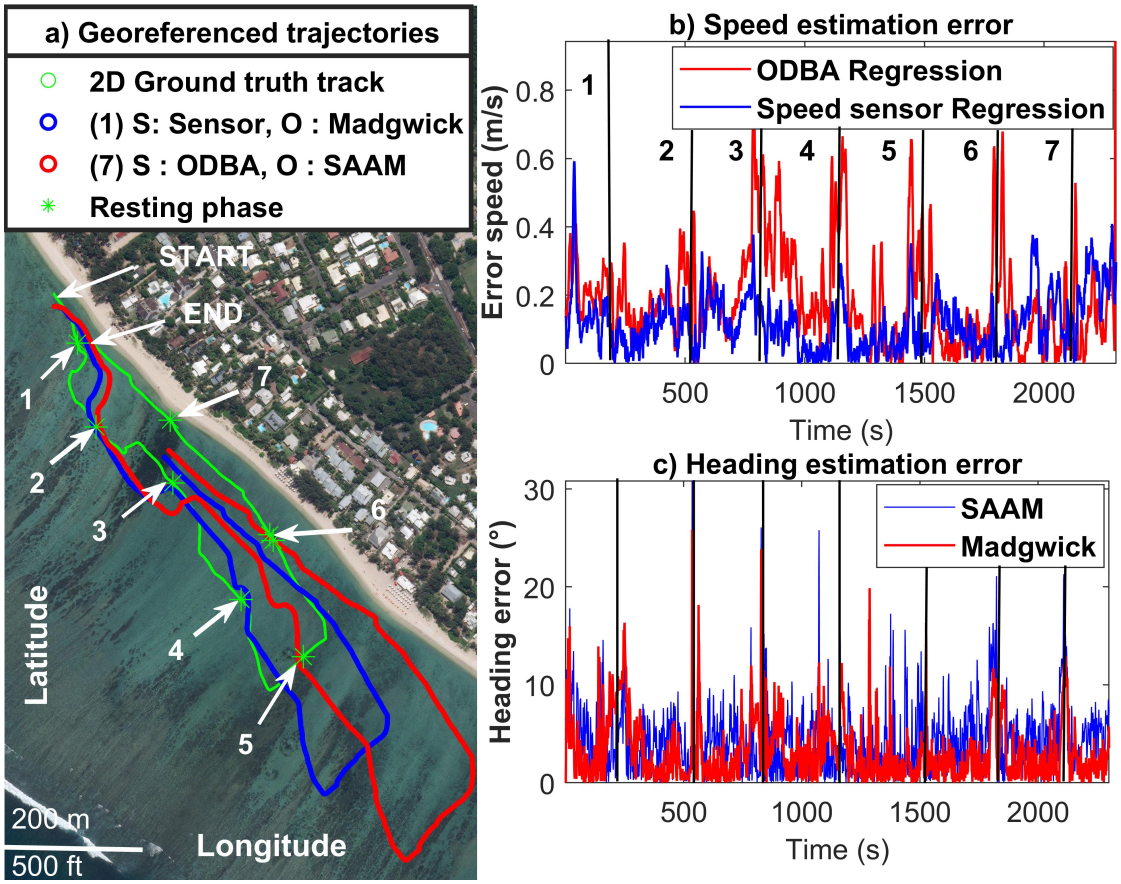


FIGURE 2.3: a) Observed trajectory from the RTK-GPS data (green line) and sequence S1 DR track estimation at 100 Hz for configurations 1 (blue line) and 7 (red line) b) Speed error compared to the RTK-GPS reference speed using using an ODBA regression (red line) and the speed sensor regression (blue line) c) Heading error over the sequence period for the two different algorithm used (SAAM - blue line, and Madgwick - red line)

0.48 m/s, and RMSE with speed sensor is 0.20 m/s at 100 Hz. A strong wind was blowing for sequence S3, creating chop at the sea surface, inducing noise and dynamic acceleration to the data used for the ODBA speed estimation. Estimation of the speed using the speed sensor gives better accuracy. For swim sequence 1 at 100 Hz, it is equal to 0.21 m/s for the speed sensor and 0.30 m/s for ODBA.

Figure 2.3 c) shows the heading error from SAAM (red curve) is more significant with more spikes. Beta parameter of Madgwick filter is set to 0.02. This low value means that the filter gives more trust on gyroscope data, which is less impacted by dynamic movement noise than the accelerometer.

Uncertainties of heading (σ_ψ) are given in Table 2.4. For swim sequence S1, the RMSE given by the heading estimation is 4.38° for the Madgwick algorithm sampled at 100Hz. In comparison, SAAM algorithms have 5.71° of RMSE at 100 Hz. The differences for the same technique are minor between 100Hz and 10Hz. At 10 Hz, for the SAAM algorithm, uncertainty is equal to 5.78°, and for the Madgwick algorithm, the RMSE is 4.40. Movement and change in the heading are slow (Tables 2.3) and do not necessitate a high sampling rate for both algorithms.

RMSE Heading (°)	Madgwick	Madgwick	SAAM	SAAM
Sequence	100 Hz	10 Hz	100 Hz	10 Hz
S1	4.38	4.40	5.71	5.78
S2	4.88	4.84	5.43	5.52
S3	6.07	6.10	6.59	6.60
RMSE Speed (m/s)	Sensor	Sensor	ODBA	ODBA
Sequence	100 Hz	10 Hz	100 Hz	10 Hz
S1	0.21	0.22	0.30	0.30
S2	0.19	0.21	0.34	0.34
S3	0.20	0.23	0.47	0.48

TABLE 2.4: Speed and heading RMSE of the different DR configurations tested for each swim sequence

The 2DRMS gives an easy tool to analyze the estimated trajectory accuracy. According to the accuracy that scientists are looking for and the knowledge of swim sequence duration without geolocated positions, the accuracy of swim sequence for the different configurations can be calculated and compared. 2DRMS reduces accuracy to 1 variable to be able to compare it. 2DMRS for all the configurations of the swim sequence S1 is given in Table 2.5.

The speed estimation techniques give more differences in trajectory error. We have a 288.39 m 2DRMS for speed sensor and 408.89 m for ODBA at 10 Hz with SAAM algorithm for orientation. On the other hand, differences in the heading estimation method show that the impact is relatively small on a trajectory with 2DRMS. We have an 2DRMS of 275.79 m for Madgwick and 288.39 m for SAAM at 10 Hz with speed sensor estimation.

The 2DRMS results follow the same trend for the other sequences with slight differences due to the different swimming patterns.

Orientation	Madgwick		SAAM	
	100 Hz	10 Hz	100 Hz	10 Hz
Speed : Sensor	275.44	275.79	288.71	288.39
Speed : ODBA	394.37	396.67	406.63	408.89

TABLE 2.5: 2DRMS (m) of the different DR configurations tested for the sequence S1

2.4.3 Influence of marine current

Marine current correction gives better improvements in trajectory estimates than changes in sampling frequency or the algorithm used for estimation. Doing this estimation even roughly is highly recommended and needs further investigation.

Figure 2.4 shows swim sequence S1 for configuration 1 with and without marine current correction (Speed: sensor / Orientation: Madgwick). Table 2.6 displays for configuration 1 the marine current parameters, and the corrected speed RMSE and 2DRMS. On average, the gain in speed RMSE is around 0.02 m/s, but by analyzing

the 2DRMS, we see that the improvement in positional accuracy is 121.59 m. In Figure 2.4, we see that the corrected trajectory is around 100 m closer than the real one. We made the same conclusion for all configurations on all the swim sequences.

	Current		**Accuracy**	
	Strength (m/s)	Direction (°)	Speed RMSE (m/s)	2DRMS (m)
C1	0.21	0.22	0.19	153.85

TABLE 2.6: Current strength and direction of sequences S1 with speed RMSE and 2DRMS for the DR configuration C1 (Orientation : Madgwick 100Hz, Speed : Speed sensor)

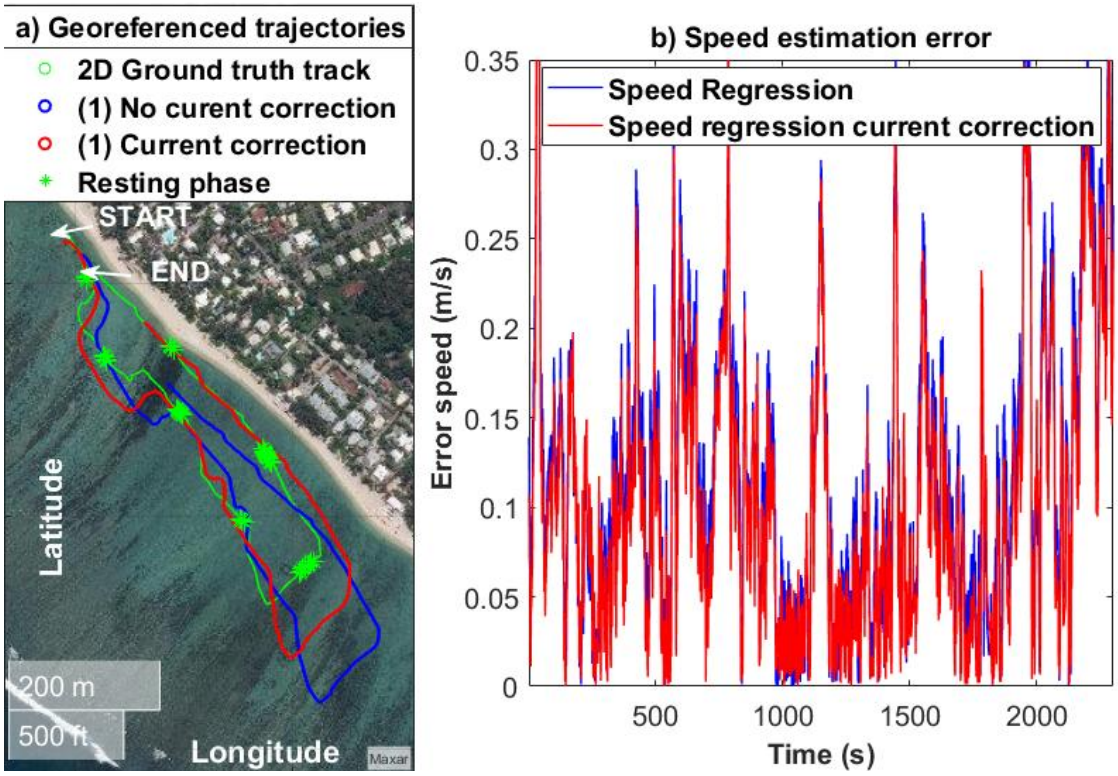


FIGURE 2.4: a) Sequence S1 DR track estimation at 100 Hz for configuration 1 with current correction b) Graph of speed error with current correction

2.4.4 Power consumption

The average power consumption results are summarized in Table 2.7. It shows that configurations with gyroscope have a much greater power consumption. For the same IMU sampling frequency (100Hz) and the same speed acquisition method (Speed sensor 1 Hz), the consumption is 3.62 mA, whereas it is 0.52 mA without it. At 10 Hz, the method, including the gyroscope and speed data, have 2.05 mA of power consumption. For the e-compass, the major consumption is given by the magnetometer. At 10 Hz, its consumption is 43.67 uA and 3.78 uA for the accelerometer. Speed sensor sampled at 1 Hz gives a 55 uA consumption. Ranges in

power consumption given by the sampling frequency are from 0.52 to 0.15 mA with speed sensor from 0.4675 to 0.09 mA with speed from acceleration for configuration using orientation with SAAM method. They are less important than consumption differences induced by the gyroscope.

Orientation	Madgwick		SAAM	
	100 Hz	10 Hz	100 Hz	10 Hz
Speed : Sensor	3.62	2.05	0.52	0.15
Speed : ODBA	3.56	1.99	0.46	0.09

TABLE 2.7: Power consumption (mA) of the different DR configuration tested for sequence S1

2.4.5 Comparison 2DRMS and electrical consumption

In Figure 2.5, we represent the different 2DRMS and power consumption of the sequence S1. The figure clearly shows that the gain of accuracy over consumption given by the gyroscope is negligible compared to the gain of the speed sensor.

For the heading algorithms at 100 Hz, ratio is 110.77 for the 2DRMS/consumption of configuration C5 (Orientation : Madgwick/ Speed : ODBA), whereas it is 883.97 for configuration C7 (Orientation : SAAM / Speed : ODBA). If there is no energy constraint, algorithms using gyroscope can make small improvements on trajectory accuracy; otherwise, it is energy-intensive.

The difference in consumption induced by the speed sensor is slight compared to the accuracy difference. At 10 Hz, the ratio accuracy over consumption is 134.53 and 199.33 for the configurations C2 and C6. The speed sensor is an excellent way to gain precision with small consumption.

As we conclude for the accuracy analysis, the differences between 100 Hz and 10 Hz are relatively small, for an subsequent consumption not negligible. For the configurations C1 and C2, the ratio accuracy over consumption is 76.09 and 134.53. We made the same analysis trend with the other swim sequences studied.

2.4.6 Application on bio-transmitter

The objectives with the bio-transmitter is to send trajectory with corresponding position uncertainty. We calculate the simplified position uncertainty with the 2DRMS divided by the number of samples and displayed it and the track.

In Figure 2.6, we display in yellow the configuration C1 trajectory corrected with current. The green trajectory is the reference one; yellow stars are the position at a time interval of what we can receive for our bio-transmitter. For clarity, only 1/5th uncertainty circles are displayed in yellow. Considering 200 bytes messages that could be sent each time the animal is breathing, it means 50 positions are available over the dive duration for a 4 bytes encoded position (X,Y,Z,σ_{pos}).

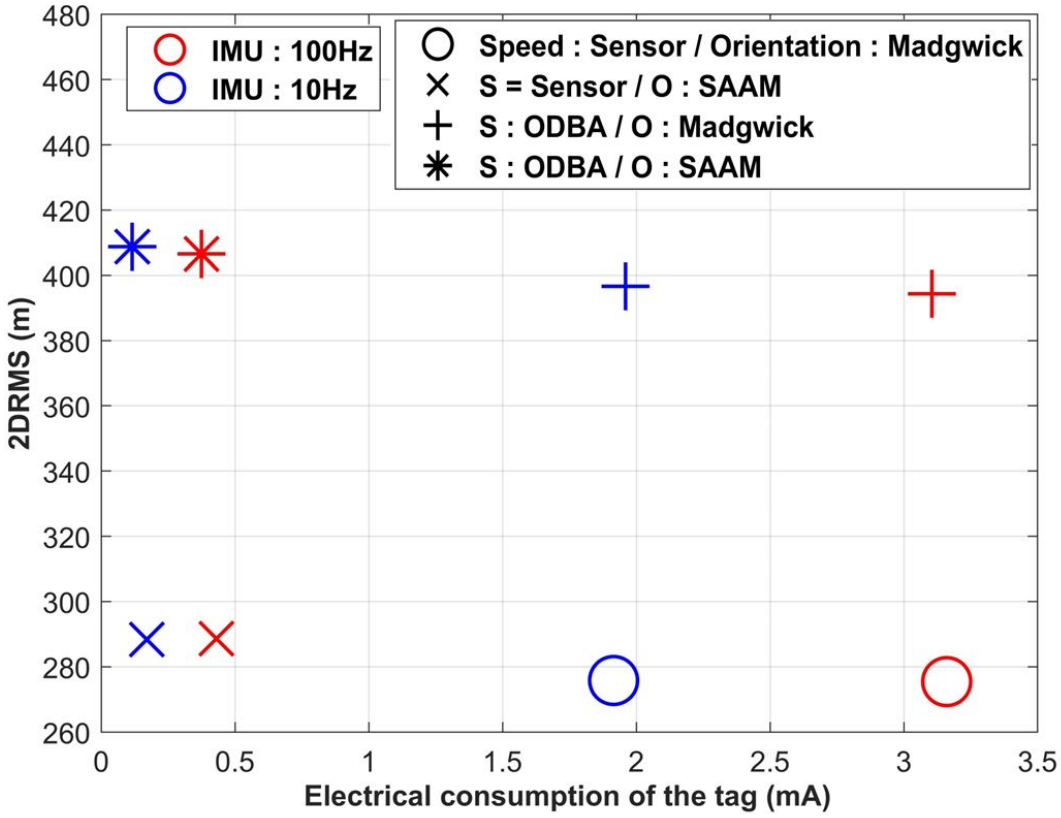


FIGURE 2.5: 2DRMS in function of power consumption for swim sequence S1

2.5 Discussion and Conclusion

This analysis provides scientists with different keys to identify the trade-offs between accuracy in trajectory estimates and power consumption as well as information on algorithm and sensor sampling frequencies to be used for trajectory estimation for bio-logging. The significant improvements needed in trajectory estimation are speed and marine current measures.

In our study, the speed estimation error is relatively high compared to the ground-truth speed value. The speed sensor propeller needs water flow close to its rotational axis. If the incidence angle is too high, the sensor does not work as expected in our regression function. Improvement can be made on its design. More directional propellers with a lower stall speed will be more appropriate for turtles. The stall speed of the sensor is 0.25 m/s (14). It needs to be coherent with the slow-moving phases of the animals, around 0.1 m/s in our case.

In addition, different estimation techniques can be tested. As for pedestrians with step length, a breaststroke length method could be considered. For diving sequences, depth can be added to correct speed as in (10) with a Kalman filter.

Different regression functions can be developed for each behavior and swimming phase. This could be done in particular for the ODBA speed estimation, where accelerometer noise can induce significant errors.

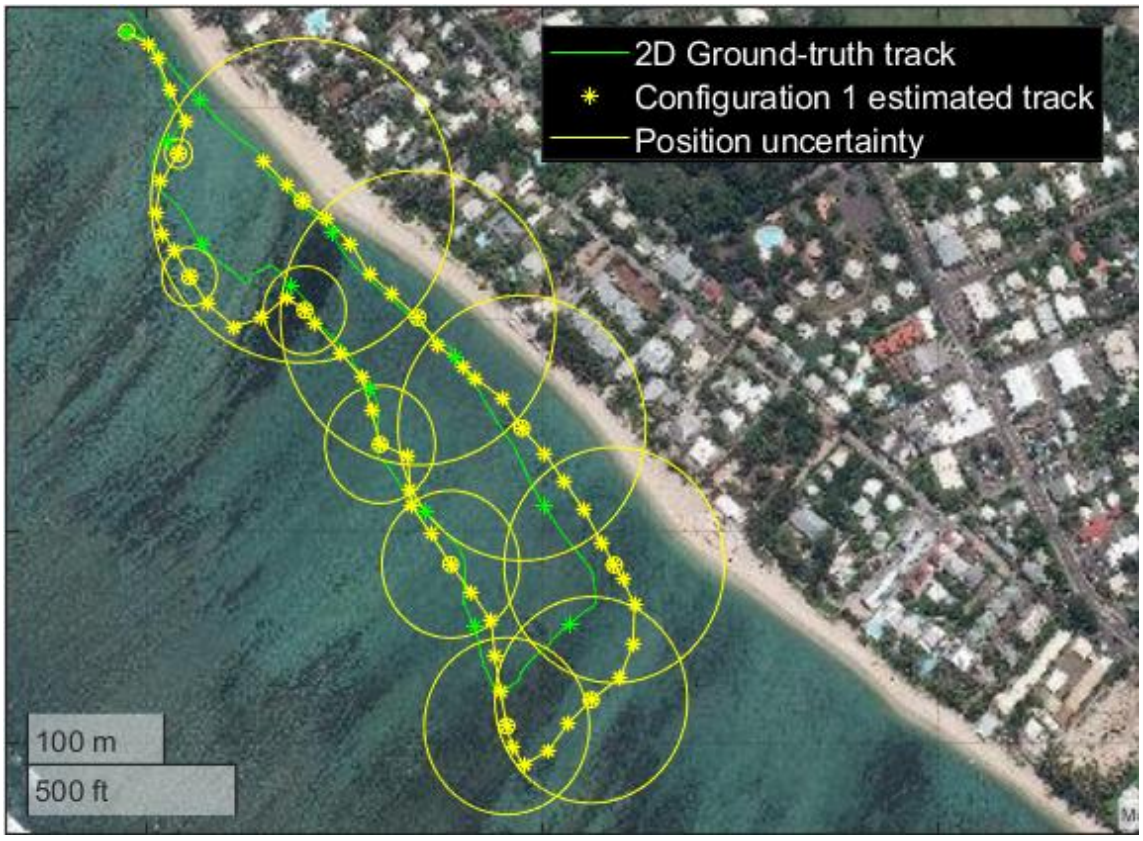


FIGURE 2.6: Swim sequence S1 track reference and configuration 1 and uncertainty circle representation.

The gyroscope method does not significantly improve the accuracy for the heading estimation and induces a high current consumption. It is not recommended for embedded applications with long deployments. We validated this affirmation for slow moving animals. It would be interesting to do the same study with faster animals and with more tortuous trajectories to see the influence of gyroscope in this case. If data is available and before considering whether or not to use the gyroscope, it will be relevant to do a frequency analysis of the animal under study. This will also help to estimate the sampling rate needed for the IMU.

The trajectory is highly impacted by marine current. Its estimation is essential to improve accuracy. The standard method to estimate is with the drift between GPS position and estimated trajectory. To roughly estimate marine current strength and direction without geolocated data, one could use the difference between the speed sensor and the ODBA method.

Behavior could be taken into account to adapt developed algorithms. The uncertainties do not increase at the same rate for each swimming phase. Adjusting a function for each behavior could give more accurate estimates for the uncertainty circles.

Bibliography

[1] T. Fukuoka, T. Narazaki, C. Kinoshita, and K. Sato, "Diverse foraging habits of juvenile green turtles (Chelonia mydas) in a summer-restricted foraging habitat in the northwest Pacific Ocean," *Marine Biology*, vol. 166, no. 3, p. 25, 2019. [Online]. Available: https://doi.org/10.1007/s00227-019-3481-9

[2] J. Leander, J. Klaminder, M. Jonsson, T. Brodin, K. Leonardsson, and G. Hellström, "The old and the new: evaluating performance of acoustic telemetry systems in tracking migrating Atlantic salmon (Salmo salar) smolt and European eel (Anguilla anguilla) around hydropower facilities," *Canadian Journal of Fisheries and Aquatic Sciences*, 2019. [Online]. Available: https://cdnsciencepub.com/doi/abs/10.1139/cjfas-2019-0058

[3] M. Espinoza, T. J. Farrugia, D. M. Webber, F. Smith, and C. G. Lowe, "Testing a new acoustic telemetry technique to quantify long-term, fine-scale movements of aquatic animals," *Fisheries Research*, vol. 108, no. 2, pp. 364–371, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165783611000300

[4] R. P. Wilson, N. Liebsch, I. M. Davies, F. Quintana, H. Weimerskirch, S. Storch, K. Lucke, U. Siebert, S. Zankl, G. Müller, I. Zimmer, A. Scolaro, C. Campagna, J. Plötz, H. Bornemann, J. Teilmann, and C. R. McMahon, "All at sea with animal tracks; methodological and analytical solutions for the resolution of movement," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 54, no. 3, pp. 193–210, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967064507000045

[5] O. R. Bidder, J. S. Walker, M. W. Jones, M. D. Holton, P. Urge, D. M. Scantlebury, N. J. Marks, E. A. Magowan, I. E. Maguire, and R. P. Wilson, "Step by step: reconstruction of terrestrial animal movement paths by dead-reckoning," *Movement Ecology*, vol. 3, no. 1, p. 23, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0055-4

[6] P. Wensveen, L. Thomas, and P. Miller, "A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales," *Movement Ecology*, vol. 3, p. 31, Sep. 2015.

[7] S. Andrzejaczek, A. C. Gleiss, K. O. Lear, C. B. Pattiaratchi, T. K. Chapple, and M. G. Meekan, "Biologging Tags Reveal Links Between Fine-Scale Horizontal and Vertical Movement Behaviors in Tiger Sharks (Galeocerdo cuvier)," *Frontiers in Marine Science*, vol. 6, 2019, publisher: Frontiers. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00229/full

[8] J. Wu, Z. Zhou, H. Fourati, and Y. Cheng, "A Super Fast Attitude Determination Algorithm for Consumer-Level Accelerometer and Magnetometer," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 3, p. 375, 2018. [Online]. Available: https://hal.inria.fr/hal-01922922

[9] S. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," 2021.

[10] G. Elkaim, E. Decker, G. Oliver, and B. Wright, "Marine Mammal Marker (MAMMARK) Dead Reckoning Sensor for In-Situ Environmental Monitoring," in *2006 IEEE/ION Position, Location, And Navigation Symposium*, Apr. 2006, pp. 976–987, iSSN: 2153-3598.

[11] T. Noda, Y. Kawabata, N. Arai, H. Mitamura, and S. Watanabe, "Animal-mounted gyroscope/accelerometer/magnetometer: In situ measurement of the movement performance of fast-start behaviour in fish," *Journal of Experimental Marine Biology and Ecology*, vol. 451, pp. 55–68, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022098113003729

[12] T. Noda, J. Okuyama, T. Koizumi, N. Arai, and M. Kobayashi, "Monitoring attitude and dynamic acceleration of free-moving aquatic animals using a gyroscope," *Aquatic Biology*, vol. 16, pp. 265–276, 2012.

[13] T. Michel, H. Fourati, P. Genevès, and N. Layaïda, "A Comparative Analysis of Attitude Estimation for Pedestrian Navigation with Smartphones," 2015.

[14] J. Gabaldon, E. L. Turner, M. Johnson-Roberson, K. Barton, M. Johnson, E. J. Anderson, and K. A. Shorter, "Integration, Calibration, and Experimental Verification of a Speed Sensor for Swimming Animals," *IEEE Sensors Journal*, vol. 19, no. 10, pp. 3616–3625, 2019, conference Name: IEEE Sensors Journal.

[15] R. Wilson, E. Shepard, and N. Liebsch, "Prying into the intimate details of animal lives: Use of a daily diary on animals," *Endangered Species Research*, vol. 4, pp. 123–137, 2008.

[16] M. Whitford and A. Klimley, "An overview of behavioral, physiological, and environmental sensors used in animal biotelemetry and biologging studies," *Animal Biotelemetry*, vol. 7, 2019.

[17] E. L. Shepard, R. Wilson, N. Liebsch, F. Quintana, A. G. Laich, and K. Lucke, "Flexible paddle sheds new light on speed: a novel method for the remote measurement of swim speed in aquatic animals," 2008.

[18] D. E. Cade, K. R. Barr, J. Calambokidis, A. S. Friedlaender, and J. A. Goldbogen, "Determining forward speed from accelerometer jiggle in aquatic environments," *Journal of Experimental Biology*, vol. 221, no. 2, Jan. 2018. [Online]. Available: https://jeb.biologists.org/content/221/2/jeb170449

[19] P. J. O. Miller, M. P. Johnson, P. L. Tyack, and E. A. Terray, "Swimming gaits, passive drag and buoyancy of diving sperm whales Physeter macrocephalus," *Journal of Experimental Biology*, vol. 207, no. 11, 2004. [Online]. Available: https://jeb.biologists.org/content/207/11/1953

[20] R. M. Gunner, M. D. Holton, M. D. Scantlebury, O. L. van Schalkwyk, H. M. English, H. J. Williams, P. Hopkins, F. Quintana, A. Gómez-Laich, L. Börger, J. Redcliffe, K. Yoda, T. Yamamoto, S. Ferreira, D. Govender,

P. Viljoen, A. Bruns, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, C. M. Duarte, M. C. van Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "Dead-reckoning animal movements in R: a reappraisal using Gundog.Tracks," *Animal Biotelemetry*, vol. 9, no. 1, p. 23, 2021. [Online]. Available: https://doi.org/10.1186/s40317-021-00245-z

[21] R. M. gunner, M. D. Holton, M. D. Scantlebury, P. Hopkins, E. L. Shepard, A. J. Fell, B. Garde, F. Quintana, A. Gómez-Laich, K. Yoda, T. Yamamoto, H. English, S. Ferreira, D. Govender, P. Viljoen, A. Bruns, O. L. v. Schalkwyk, N. C. Cole, V. Tatayah, L. Börger, J. Redcliffe, S. H. Bell, N. J. Marks, N. C. Bennett, M. H. Tonini, H. J. Williams, C. M. Duarte, M. C. v. Rooyen, M. F. Bertelsen, C. J. Tambling, and R. P. Wilson, "How Often Should Dead-Reckoned Animal Movement Paths be Corrected for Drift?" Tech. Rep., 2021. [Online]. Available: https://www.researchsquare.com/article/rs-587959/v1

[22] C. Laplanche, T. A. Marques, and L. Thomas, "Tracking marine mammals in 3D using electronic tag data," *Methods in Ecology and Evolution*, vol. 6, no. 9, pp. 987–996, Sep. 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12373

[23] Y. L. Bras, J. Jouma'a, and C. Guinet, "Three-dimensional space use during the bottom phase of southern elephant seal dives," *Movement Ecology*, vol. 5, no. 1, p. 18, 2017. [Online]. Available: https://doi.org/10.1186/s40462-017-0108-y

[24] E. Bryant, *2D Location Accuracy Statistics for Fastloc RCores Running Firmware Versions 2.2 & 2.3, Technical Report TR01*, Aug. 2007.

[25] K. Aoki, M. Amano, K. Mori, A. Kourogi, T. Kubodera, and N. Miyazaki, "Active hunting by deep-diving sperm whales: 3D dive profiles and maneuvers during bursts of speed," *Marine Ecology Progress Series*, vol. 444, pp. 289–301, 2012.

[26] T. Narazaki, K. Sato, K. J. Abernathy, G. J. Marshall, and N. Miyazaki, "Sea turtles compensate deflection of heading at the sea surface during directional travel," *Journal of Experimental Biology*, vol. 212, no. 24, pp. 4019–4026, Dec. 2009. [Online]. Available: https://doi.org/10.1242/jeb.034637

[27] S. L. Cox, F. Orgeret, M. Gesta, C. Rodde, I. Heizer, H. Weimerskirch, and C. Guinet, "Processing of acceleration and dive data on-board satellite relay tags to investigate diving and foraging behaviour in free-ranging marine predators," *Methods in Ecology and Evolution*, vol. 9, no. 1, pp. 64–77, 2018. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12845

[28] T. Photopoulou, P. Lovell, M. A. Fedak, L. Thomas, and J. Matthiopoulos, "Efficient abstracting of dive profiles using a broken-stick model," *Methods in Ecology and Evolution*, vol. 6, no. 3, pp. 278–288, 2015. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12328

[29] D. C. Lidgard, W. D. Bowen, I. D. Jonsen, B. J. McConnell, P. Lovell, D. M. Webber, T. Stone, and S. J. Iverson, "Transmitting species-interaction data from animal-borne transceivers through Service Argos using Bluetooth communication," *Methods in Ecology and Evolution*, vol. 5, no. 9, pp. 864–871, 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12235

[30] T. A. Wild, M. Wikelski, S. Tyndel, G. Alarcón-Nieto, B. C. Klump, L. M. Aplin, M. Meboldt, and H. J. Williams, "Internet on animals: Wi-Fi-enabled devices provide a solution for big data transmission in biologging," *Methods in Ecology and Evolution*, vol. n/a, no. n/a, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13798

[31] "pIOT (2018-2020) - IOT (2018-2021)." [Online]. Available: https://wwz.ifremer.fr/lareunion/Projets/Innovations-technologiques/pIOT-2018-2020-IOT-2018-2021

[32] J. Davenport, S. A. Munks, P. J. Oxford, and G. E. Fogg, "A comparison of the swimming of marine and freshwater turtles," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 220, no. 1221, pp. 447–475, 1984, publisher: Royal Society. [Online]. Available: https://royalsocietypublishing.org/doi/10.1098/rspb.1984.0013

[33] R. P. Wilson, C. R. White, F. Quintana, L. G. Halsey, N. Liebsch, G. R. Martin, and P. J. Butler, "Moving towards acceleration for estimates of activity-specific metabolic rate in free-living animals: the case of the cormorant," *The Journal of Animal Ecology*, vol. 75, no. 5, pp. 1081–1090, 2006.

[34] C. Fernández-Prades, "1.- Accuracy." [Online]. Available: https://gnss-sdr.org/design-forces/accuracy/

Chapter 3

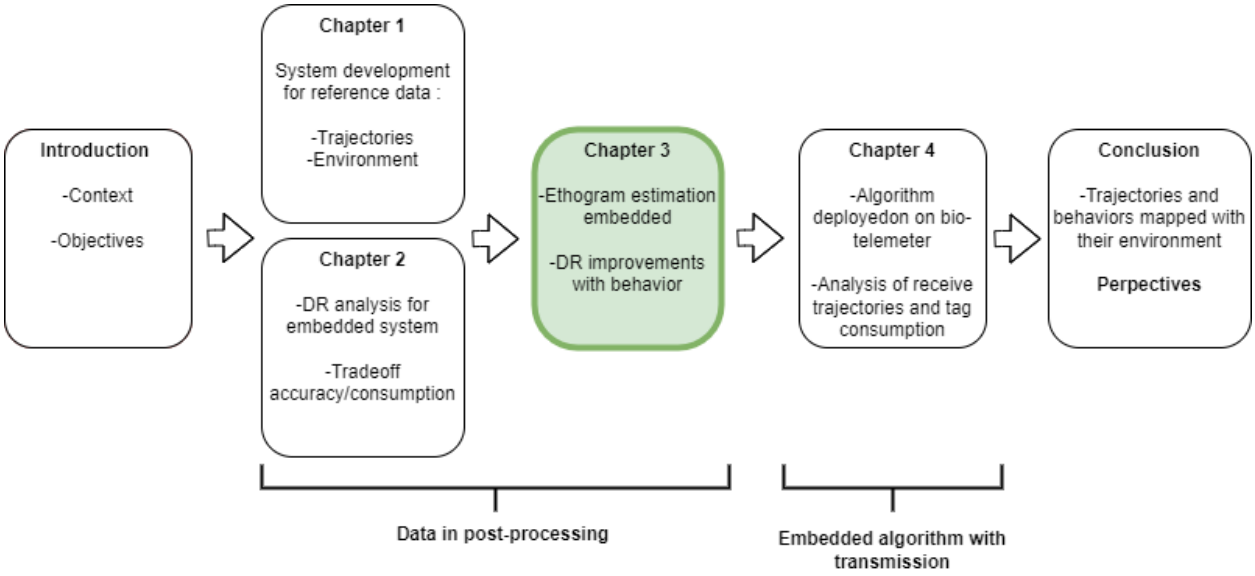
Behavioral based Dead-Reckoning for embedded application on marine animals

Authors : Gogendeau Pierre, Bonhommeau Sylvain, Fourati Hassen and Bernard Serge

Article status : In preparation for *Methods in Ecology and Evolution*

FIGURE 3.1: Turtle capture and ASV deployment

Key words: Ethogram, Bio-logging, Dead-reckoning, Marine animals, Speed estimation



Reminder of the chapter transition graph. The current chapter is highlighted in green

Abstract

- For marine animals, analyzing trajectories at a fine temporal scale over long deployments offers the opportunity to strengthen our knowledge about species' spatial ecology and its relationship with the environment. However, retrieving the data to estimate these trajectories is complex. In this paper, we develop a solution to efficiently estimate the trajectories and send them with a bio-telemeter during long deployments. This type of solution has many constraints: accuracy, power consumption, computing, memory, and available sensors.

- To address these issues, a method is proposed using an ethogram with an adaptive trajectory estimation according to the animal behavioral phases. Several speed estimation functions depending on the behavioral phase are developed to optimize the trajectory accuracy and algorithm computing. We compare the accuracy, consumption, and memory of the proposed method to non-adaptive methods, based on experimental tests on a juvenile green turtle (*Chelonia mydas*).

- The proposed algorithm using the ethogram shows better accuracy and lower power consumption than the non-adaptive techniques. Compared to non-adaptive methods, speed estimation shows an improvement between 15.4% at 10 Hz and 7% at 1 Hz and trajectory accuracy between 72.7% and 30.3%. This adaptive method is embedded in a bio-telemeter and presents an average consumption optimized of 9.7% and 13.6%.

- This method offers a simple, low-power consumption, and adaptable solution to compute embedded trajectories while providing information on the behavior. Similarly, the proposed ethogram has been developed with the aim of

being flexible and versatile for other behaviors or for other species. The proposed methods are developed to be easily replaced with ethograms or speed sensors already developed by scientists

3.1 Introduction

Tracks geolocated (geo-tracks) or not (pseudo-tracks) are critical data in ecology to understand foraging behaviors, movement ecology, swimming kinematics, environmental occupancy or anthropogenic disturbances. Recently, more studies on behavior using ethogram have been available, especially with machine learning techniques. This behavioral information provides additional information for the analysis of movements (1). The behavior information can also be used to improve and adapt the use of some energy consuming sensors or pre-process the data to optimize memory and transmissions (2; 3).

For marine animal tracking, fine temporal scale trajectories over long distances ($>$ several km) are limited to the use of bio-loggers using dead-reckoning (DR) (4; 5). This requires the recovery of the bio-logger, which can be complex depending on the context (6) and to post-process the raw dataset (7). Bio-telemeters, in the current state of technology, do not allow the transmission of full trajectories because the transmission range and bandwidth are limited (8; 9). Bio-telemeters can only send summarized dive profiles and some compressed information, such as selected behaviors. Only acoustic systems allow to transmit dive profiles or reconstruct underwater trajectories, but require dense receiver networks and the range of the systems is reduced to a few hundred meters (10). A key challenge in bio-telemetry technology is hence to embed the trajectory estimation within the bio-telemeter and transfer the data while keeping a low-power consumption and a minimal amount of data to transfer.

Dead reckoning methods require the estimation of speed and heading and are commonly used in bio-logging for multiple animals, pinniped (11; 12), whales (6; 13), turtles (14; 15), and fishes (1). Chapter 2 showed that the main source of errors is the estimation of the speed. In marine environment, speed can be estimated with speed sensors such as propellers (16; 17; 18), flexible paddle (19), or using the information on the depth and pitch of the animal (20). Some studies used proxies for dynamic acceleration to infer speed such as vibration (21) or the vector of dynamic body acceleration (VeDBA) (22; 23; 24). Chapter 2 showed that the estimation of speed with the VeDBA might introduce important errors during the contact phases with the external environment, e.g. when a turtle bangs a rock at the sea bottom. Speed estimation techniques are therefore not always suitable or necessary for all phases of animal behavior. They may actually introduce errors during specific behaviors.

Recently, several studies propose lighter and optimized solutions to embed the calculation of an ethogram on a bio-logger (25; 2; 3; 26). Wilson et al. (27) proposed a technique based on the Lowest Common Denominator, i.e. decision trees with predefined sequences of behaviors. These methods allows to optimize computing (27), limit data preprocessing (28; 26; 3) or sensor usage (2; 29; 30). In (2), these methods are used to trigger a camera and film the predation actions of seagulls.

In this paper, we propose a method to improve the trajectory estimation techniques in terms of accuracy and power consumption using the information from the

behaviors of the animal. The developed ethogram is simple, flexible, and adaptable. For each behavior phase, the speed estimation function is adapted and used to inform the DR algorithm. The estimated trajectories are compared to a reference trajectory at a fine scale resolution ($<$ 1 m) obtained from an acoustic system deployed on a marine turtle.

3.2 Materials

The equipment used to calibrate the trajectory algorithm consists of an acoustic system to collect position data at very fine scale to be used as a reference trajectory and a bio-logger which collects input data for a dead-reckoning algorithm, i.e. 3-axis accelerometers and magnetometers and a pressure sensor. This equipment is deployed on a sea turtle in the Reunion Island over several hours. The timing and consumption tests are estimated on the target bio-telemeter of the study.

3.2.1 Reference trajectory and data acquisition

To study and compare sea turtle trajectories on a fine temporal scale, accurate geolocated positional data and raw bio-logger data are needed. To collect geolocated reference data at a fine temporal scale, we have developed an Autonomous Surface Vehicle (ASV) with an acoustic tracking system described in Chapter 1, shown in Figure 3.2. The acoustic module used is a Short Base-Line (SBL) system Underwater GPS G2 from Waterlinked[1]. It requires 4 acoustic receivers spaced at 2 m each. The position of the acoustic beacon is calculated with its distance measured from the 4 receivers. To geolocate the acoustic transmitter, we used the position from a GPS Real Time Kinematic (RTK) and the heading of the ASV. The acoustic transmitter deployed on the animal is the Waterlinked U1[2]. Its size is 3.2 x 12.1 cm for 75 g in water. The system's accuracy is given at 1 m minimum, but decreases with the distance between the transmitter and receivers. The acoustic sampling frequency is 1 Hz. To estimate the accuracy, we used the Standard Deviation (STD) calculated by the system for each position. A filtering procedure, detailed in appendix B, is applied to the raw data. The maximum range of the acoustic receiving system is 100 m from the acoustic beacon deployed on the sea turtle. To keep the ASV within this range, we developed a tracking function with the SBL and the Autopilot. To ensure the autonomous guiding of the ASV, we used the Pixhawk 2.1[3] with Ardupilot[4]. All details about the developed ASV and its functionalities are presented in Chapter 1.

3.2.2 Targeted embedded system

The study aims to embed a solution to estimate trajectories on the bio-telemeter that we developed for the IOT project (IOT tag), shown in 3.3. The tag includes a "TurtleTracker" board composed of the STM32L082 microcontroller and the LoRa transceiver SX1276 (CMWX1ZZABZ module), 3-axis accelerometers and

[1] https://store.waterlinked.com/product/underwater-gps-g2/
[2] https://store.waterlinked.com/product/locator-u1/
[3] https://docs.px4.io/v1.9.0/en/flight_controller/pixhawk-2.html
[4] https://ardupilot.org/rover/

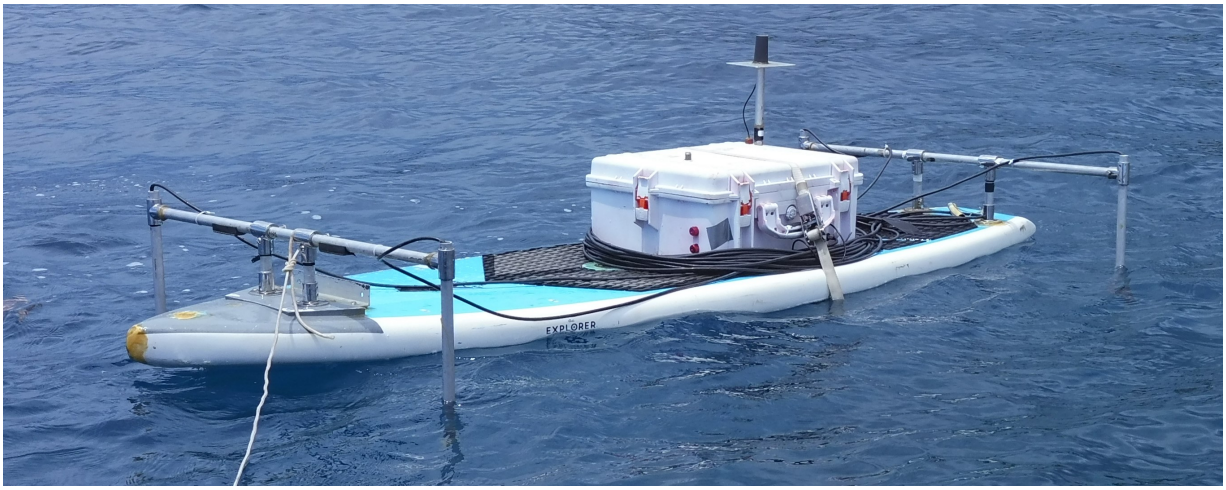


FIGURE 3.2: The ASV "Plancha" with acoustic tracking system mounted. The acoustic receivers are attached to the submerged arms.

magnetometers (LSM303AGR), and two pressure sensors MS5837-30BA (for depth measurements > 2 m) and MS5803-01BA (for shallower depths < 2 m). The device have also an 8 Mb SPI NOR flash memory. For the final application, we aim to deploy the algorithm on this bio-telemeter with embedded calculation. We used the bio-telemeter to verify if the proposed solution could be embedded considering the limited computing resources. Moreover, the bio-telemeter measure the actual power consumption. The timings of the code execution, the associated power consumption, and the memory used were measured in the laboratory to evaluate the feasibility of this approach in real deployments.

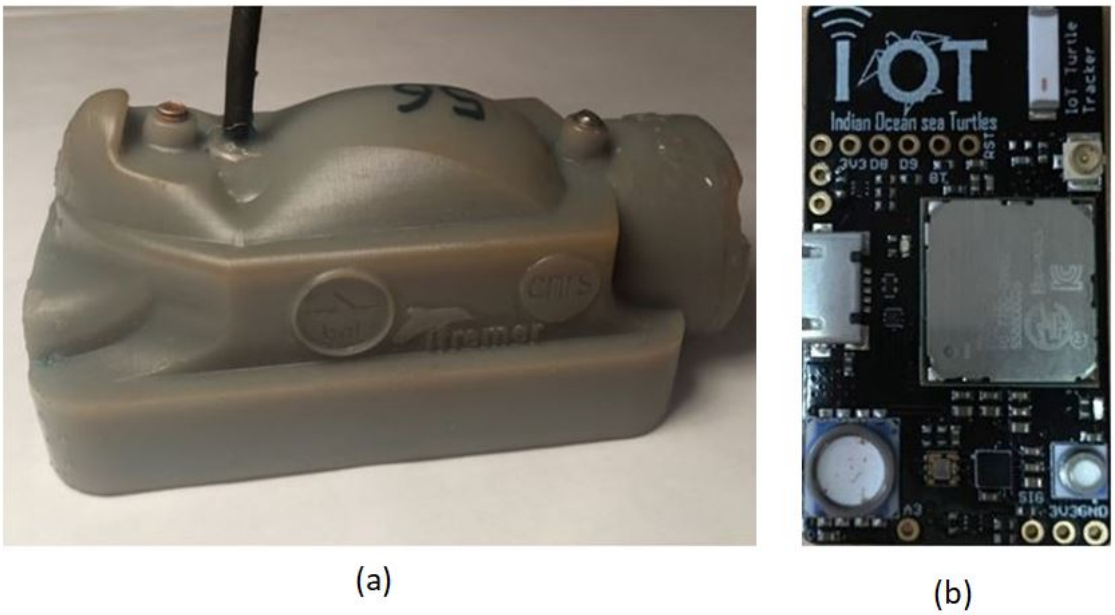


FIGURE 3.3: IOT tag (a) and its "TurtleTracker" board (b)

3.3 Methods

In the context of embedded computing on microcontrollers for long deployments, our goal is to improve the trajectory estimation, while optimizing the power consumption, i.e. computation time and sensors usage. In this section, we develop a simple method to determine a temporally flexible ethogram that can be adapted to different study cases, as well as speed estimation methods associated to these behaviors.

3.3.1 Algorithm variables calculation

The animal heading is a key information for DR algorithms. In (31), Gogendeau et al. concluded that for slow-moving marine animals, the additional information from a gyroscope is not required and that the SAAM algorithm (32) is well suited in terms of accuracy and it only requires 37 arithmetic operations. In addition, the following method for trajectory estimation proposes to use other variables from acceleration proxies, i.e. the Pitch (given by the SAAM algorithm), Jerk (differential of acceleration) (33), and VeDBA (24). This information is used in a decision algorithm to calculate the animal's speed depending on its behavior. This speed estimation can then be used as an input for the DR algorithm.

3.3.2 Trajectory estimation

Path estimation with the DR algorithm can be simplified on a horizontal plane (2D) as the actual depth is measured directly with the pressure sensor of the bio-logger. The 2D path vector is defined as $[x(t); y(t)]$ at the time t of the movement sequence. The Earth projection used is North-East-Down (NED). The X-axis is northward, Y-axis is eastward, and the Z-axis is downward. The sampling frequency Δt is 1 Hz. The DR formula can be written as follows:

$$x(t) = x(t - \Delta t) + S_{turtle}(t) \times \Delta t \times cos(\theta_{turtle}(t)) + decli) \tag{3.1}$$

$$y(t) = y(t - \Delta t) + S_{turtle}(t) \times \Delta t \times sin(\theta_{turtle}(t)) + decli) \tag{3.2}$$

where $S_{turtle}(t)$ is the speed in turtle frame (body frame) at time t, $\theta_{turtle}(t))$ corresponds to the heading in turtle frame at time t and $decli$ is the magnetic declination for the experimental field.

3.3.3 Ethogram

Principle

As for the main software architecture, the ethogram is designed to be run on a microprocessor with low computation capabilities. Most of the others ethograms developed require complex methods or are not flexible in time or adaptable. Behaviors can be added and modified without changing the structure of the algorithm.

Rather than making a classical ethogram associating each time window with a behavior, our method is based on single variable thresholds, variable thresholds with occurrence counters, timings and previous behavior following the concept developed

by (27). These allow us to trigger starting and ending flags for transitions between behaviors. We divided our ethogram into two layers. The first layer defines the main groups of behavior (Table 3.4). The transition diagrams are in Figure 3.4. The second layer of the ethogram defines the type of swimming, i.e. swimming with a high pitch (upward or downward), a regular swimming, or other types of swimming which does not fall in the first two cases. For each combination of the two ethogram layers, we set a speed function which is defined in the section "speed algorithms". The proposed ethogram is calibrated on sea turtle data but is easily adaptable to other species.

First Layer: Main behavior

In the proposed ethogram, most of the behaviors are not ambiguous and easily identifiable. In Table 3.1 we described the main behaviors. Figure 3.4 shows the details of the transition diagram with the first layer. The ethogram is run for each turtle dive and starts with the *SURFACE* behavior. The ethogram of a dive ends when the turtle returns back to the surface. The ethogram function has 3 steps. First, depending on the current behavior, the necessary variables for the transition are calculated. These variables update the occurrence counters, or timings useful for the behavior transition. Figure 3.4 represents the variables that follow the *IF*, *WHILE* or *FOR* conditions. Then the algorithm checks if the transition conditions are satisfied. Finally, if the conditions of the rhombuses are satisfied, the ethogram enters the behavior that the condition points to, and otherwise, it remains in the same state.

For some behaviors, we added another conditional step, e.g. the depth difference is calculated to confirm the transition to the *GROUND* behavior. Some variables are set up as authority variables. These variables are often represented by a limit value that defines a behavior that does not need time flexibility. The end-of-dive variable with a single threshold on the depth preempts the other conditions.

Second layer : Active behaviors

The ethogram second layers aim to describe the sub-phases during the active behaviors (*SWIM*, *UP* and *DOWN*) of the animal. Three sub-behaviors are defined:

***High pitch* (*H-PITCH*):** For the detection of *high pitch* phases, we used the same principle with an occurrence counter and a threshold on the value. If the pitch satisfies the condition defined by the threshold, we incremented its occurrence counter. As soon as the pitch value does not meet the threshold condition, the counter is reset to zero. When the counter is greater than the defined timing, we validated the entry into the *high pitch* (*H-PITCH*) phase with the first occurrence of the counter. The *H-PITCH* phase ends as soon as the pitch no longer meets the threshold condition. In parallel, the pitch average is calculated and stored in the ethogram.

***Regular swim (R-SWIM)* :** For the *regular swim* phases, the dynamic acceleration on the longitudinal axis (Turtle x-axis) is used. First, a function is applied to detect acceleration zero crossing, minimum and maximum. With this data we deduced the time between two zero crossings called stroke time (Figure 3.5.a). If

Behavior name	Description
SWIM	This general behavior is set when the turtle is moving. This behavior contains several *SWIM* sub-behaviors. It includes when the turtle swims regularly, with *H-PITCH* or does some unknown behavior without being on the seabed.
REST	The REST behavior corresponds to the moment when the turtle is not moving and resting on the seafloor. It is set when the VeDBA is below 0.006 g for more than 10 seconds. During the resting phase, all sensors are switched off except the accelerometer.
DOWN	This behavior corresponds to the turtle going down for longer than 5 seconds and with a slope of depth curve higher than 0.008 m/s.
UP	This behavior corresponds to the moment when the turtle is going up for more than 15 seconds and the slope of the depth curve should be inferior to -0.004 m/s. Timing and threshold are different of *DOWN* because the turtle pitch angle and swim speed are not the same.
GROUND	This state variable corresponds to when the turtle is on the ground but with high dynamic acceleration. This behavior is hard to interpret without video, but it can be when the turtle is eating on the seabed or is moving before/after resting phases. This state set the speed to 0. We used Jerk instead of VeDBA because this acceleration proxy also includes the change of orientation which occurs when the turtle is eating facing down. Jerk and difference of depth are the thresholds to trigger this behavior.
SURFACE	This state is the initial behavior of any dive. We used a depth threshold of 0.2 m to end this behavior without timing occurrence. After changing the starting *SURFACE* behavior, when the turtle depth is lower than a depth threshold of 0.2 m, the dive ends.

Table 3.1: Description of main behaviors given by the ethogram first layer

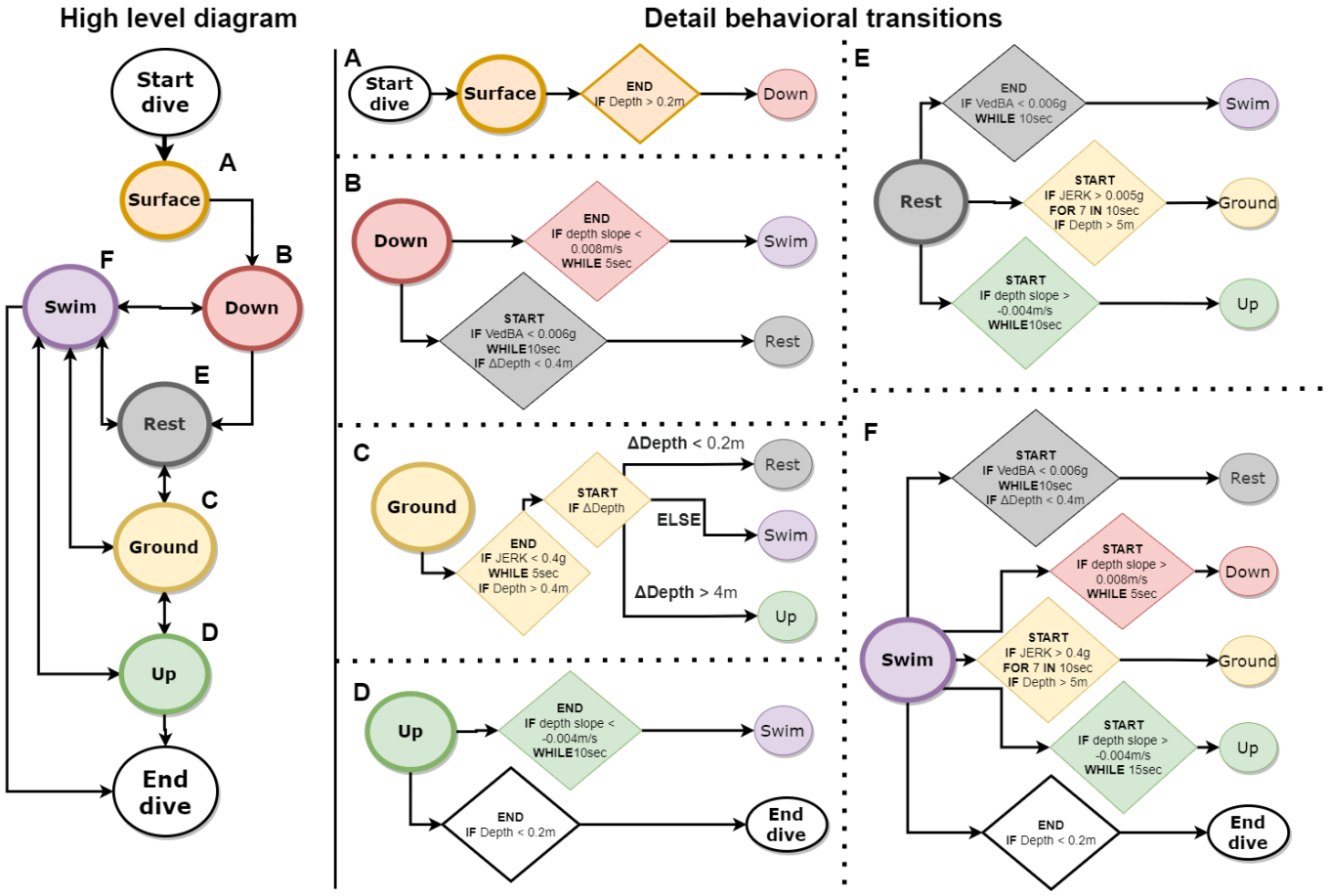


FIGURE 3.4: Transition diagram of the ethogram with the first layer at 10 Hz. On the left is the high level transition diagram. Each arrow represents in which direction the transition is allowed. On the right, there are detailed transitions with the condition required. For each behavior, transition to the next can occur if the conditions in the diamonds are met. The conditions in white with bold diamonds have the priority over the others. For the 1 Hz ethogram, the principle is the same, but the thresholds are slightly different. For the behavior *DOWN*, the slope of the depth curve must be higher than 0.08 m/s. For *UP*, the slope of the depth curve should be lower than -0.04 m/s. The behavior *REST*, is defined by a VeDBA value lower than 0.02 g. To start the *GROUND* behavior, Jerk must be larger than 0.22 g during 4 seconds. The *GROUND* behavior is ended when the Jerk is lower than 0.22 g during 15 s.

the minimum, the maximum and timing values are between certain intervals for a defined number of occurrences, then we considered regular swimming.

Other swim (O-SWIM) **:** If none of the above swimming patterns are detected, the phase is classified as *other swim*.

3.3.4 Speed algorithms

The main work in trajectory reconstruction is focused on improving speed estimation. In Chapter 2 we showed that general speed estimation techniques based only on acceleration proxies could induce many errors. Indeed, the dynamic accelerations of the animal like regression with VeDBA, are not only created by its movement but also by the parasitic movements and interactions with the external environment. It is therefore important to study the animal behaviors in more detail to associate them with a specific function for speed calculation. These speeds can be fixed or derived from variables. The different functions proposed are thought to have a good ratio of computing and precision to be embedded. Different techniques that are not based on behavior for estimating the speed are compared to the proposed one. The first one is based on a VeDBA regression at 10 Hz. The second is based on a fixed swimming speed coupled with estimating the speed by orientation and vertical velocity in the high-pitch phases at 1 Hz.

Speed associated with behaviors

Table *3.2* describes the speed functions for each behavior of the ethogram. We proposed two algorithms using accelerometer data at 1 Hz and 10 Hz and the ethogram:

Speed function using accelerometer data at 1 Hz ($F1_E$): For *H-PITCH* phases we used the Orientation Corrected Depth Rate (OCDR) function with the pitch and the difference of depth described below. Every other behavior has a fixed speed. At 1 Hz, the sampling frequency is too low to accurately estimate the derivative of the variables from acceleration such as VeDBA or Jerk in speed function.

Speed function using accelerometer data at 10 Hz ($F10_E$): For the *R-SWIM* behavior, speed is estimated with the stroke function described below. OCDR function is applied for *H-PITCH* phases. The speed of other behaviors is fixed.

Stroke speed function: Stroke detections are made with zero crossing of the acceleration on the longitudinal axis (Turtle x-axis, Figure 3.5.a). Times between two zero crossings give the stroke frequencies and the min and max values give the peak to peak values. Then, linear regression is applied to these values to estimate the speed.

Orientation Corrected Depth Rate (OCDR): Following the assumption that the turtle displacement is in its forward orientation, we estimated speed with the difference of depth and the average pitch of the behavior with the following formula:

$$h\_speed_t = \frac{\Delta(d1, d2)}{tan(\bar{\theta}(t1, t2))} \times Fs \tag{3.3}$$

<table>
<tr><th>Behavior</th><th>Speed function at 1Hz ($F10_E$)</th><th>Speed function at 10Hz ($F1_E$)</th></tr>
<tr><td>SWIM</td><td colspan="2">-R-SWIM : Fixed to 0.37 m/s | -R-SWIM : Stroke function
-H-PITCH : OCDR function
-O-SWIM : Fixed to 0.35 m/s</td></tr>
<tr><td>REST</td><td colspan="2">Fixed to 0 m/s</td></tr>
<tr><td>UP</td><td colspan="2">-R-SWIM : Fixed to 0.49 m/s | -R-SWIM : Stroke function
-H-PITCH : OCDR function
-O-SWIM : Fixed to 0.47 m/s</td></tr>
<tr><td>DOWN</td><td colspan="2">-R-SWIM : Fixed to 0.40 m/s | -R-SWIM : Stroke function
-H-PITCH : OCDR function
-O-SWIM : Fixed to 0.3 9 m/s</td></tr>
<tr><td>GROUND</td><td colspan="2">Fixed to 0 m/s</td></tr>
<tr><td>SURFACE</td><td colspan="2">Fixed to 0 m/s or marine current speed if available</td></tr>
</table>

TABLE 3.2: Speed for the different behaviors. All active phases are divided in sub-behaviors with the second layer of the ethogram

with h_speed_t is the horizontal speed in turtle frame, $\Delta(d1, d2)$ is the depth difference between t1 and t2, $\overline{\theta}(t1, t2)$ is the average pitch between t1 and t2 and Fs is the sampling frequency of the algorithm.

This function can not be applied for small pitches. For $F10_E$ and $F1_E$, the OCDR function is triggered with the *H-PITCH* behavior defined in the ethogram section. This function was only applied for phases where the pitch is larger than 20°. This allows us to have a larger depth difference and reduce the computation cost.

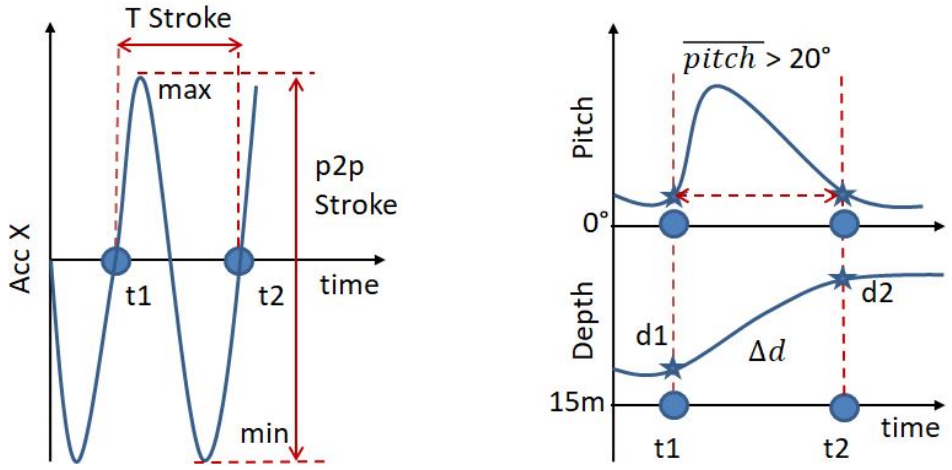
Fixed speed: *REST* and *GROUND* speed are considered to be 0 m/s. Surface speed is equal to marine current estimation if available or 0 m/s if not. *SWIM*, *UP* and *DOWN* speeds are calculated by averaging the speed from the fine-scale referenced data during each phase.

Speed without behavior

To compare with the functions presented above, we proposed 2 functions for the speed that do not use the ethogram :

Speed function at 1 Hz ($F1_S$) : $F1_S$ is inspired by the classical speed estimation functions in bio-logging proposed in the Tag Tools project toolkit[5]. The speed of horizontal displacement of the turtle in Earth frame is considered fixed at 0.40 m/s. This speed is found by averaging the active phases of the turtle. When the turtle

[5] http://www.animaltags.org/doku.php?id=tagwiki:tutorials (Practical 3)



(a) (b)

FIGURE 3.5: Stroke function diagram (a) and OCDR function diagram (b) applied to a phase for $F10_E$ and $F1_E$

has a pitch greater than 20° the vertical speed is calculated with OCDR function. This OCDR technique is similar in principle to the first technique proposed at 1 Hz but is applied to each time step and not to the whole phase of a defined behavior.

Speed function at 10 Hz ($F10_S$) : The $F1_S$ function described in Chapter 2 is defined from a linear regression between the turtle VeDBA and the speed calculated from the reference data obtained by the acoustic system. This speed function needs 10 Hz VeDBA data input.

For these two techniques without behavior, we also set the speed to zero with a threshold on the VeDBA at 0.006 g. Indeed, without correction phases, where the turtle is at a resting state, an important error is introduced.

3.3.5 Comparison of the algorithms

The assessment of the different algorithms is performed on (i) the accuracy of speed and trajectory estimation and (ii) the power consumption.

Comparison with reference data

Speed and horizontal position errors are calculated from the difference between the algorithm predictions and the "ground-truth" positions from the acoustic system. The error is evaluated using the root-mean-square error (RMSE).

Speed (S) errors are defined as follows:

$$RMSE_S = \sqrt{\frac{\sum_{i=1}^{n}(\Delta S(i)_t)^2}{n}} \tag{3.4}$$

where n is the number of samples of the swim sequence, $\Delta S(i)_t$ is the difference between observed and predicted speeds in the turtle frame (body frame) at i-th sample between ground-truth and estimated speed. $RMSE_S$ includes observation

and process errors, i.e. sensor measurement, sensor drift, computational and modeling errors.

To compare the accuracy of the trajectory estimates, we used the two-dimension (X and Y horizontal components) root-mean-square error (2DRMS (34)). The estimated horizontal position and the ground-truth acoustic position data are compared in the NED frame. The 2DRMS is defined as:

$$2DRMS = \sqrt{\frac{\sum_{i=1}^{n}(\Delta E(i)^2 + \Delta N(i)^2)}{n}} \tag{3.5}$$

where $\Delta E(i)$ and $\Delta N(i)$ are the residuals between the observation and the prediction of the position in the East and North components of the i-th estimation of the position sample.

To compare the influence of our different algorithms on the trajectory, we estimated the ground-truth trajectory with the reference speed and the estimated orientation. All the estimated trajectories used the same orientation. The 2DRMS includes estimation errors and sampling frequency influence.

Measurements on embedded system

Hardware measurements are performed on the "IOT Turtle tracker" electronic board, described in the Material section. Such measurements highlight the advantages of the proposed algorithms using the ethogram in terms of execution time, and thus on power consumption. The tag programs are coded and compiled via the Arduino IDE in C++, and codes are available on GitHub[6].

The aim of the proposed algorithms is to manage the functions used and the sensor acquisition according to the animal behavior to reduce the tag consumption, e.g. only calculating the VeDBA during the *REST* phases. This allows us not to use the magnetometer, the pressure sensor and not to compute the orientation.

We calculated the total duration of each function for each behavior. The minimum frequency of the magnetometer is 10 Hz. The consumption difference for accelerometer 1 Hz and 10 Hz is negligible. We have therefore set the sampling frequency of the accelerometer and magnetometer at 10 Hz for all algorithms.

Consumption estimates on the electrical board are made in laboratory with the power analyzer N6705B. The measured power consumptions are applied to the calculated duration to determine the average consumption. The details of our calculation method are provided in Appendix C.

3.4 Results

After our filtering procedure, the reference data include around 138 minutes of accurate positioning over the 228 minutes of data collection. The data are divided into 6 sequences. The last one is used as a validation sequence and represents around 25% of the reference data (Figure 3.6.a).

[6] https://github.com/pierregoge/Behavioral-based-Dead-Reckoning/tree/main/Arduino/IoT_Turtle_conso

	SWIM	*REST*	*UP*	*DOWN*	*GROUND*	*SURFACE*
10Hz	47.25%	25.79%	12.96%	11.04%	1.16%	1.34%
1Hz	53.30%	19.51%	13.83%	10.99%	2.32%	0.03%

(a) First layer : Main behavior percentage for 10 Hz and 1 Hz algorithms. Resting percentage is used to calculate the processing timing for $F10_E$ et $F1_E$ functions

	R-SWIM		***H-PITCH***		**O-SWIM**	
Main Behavior	10 Hz	1 Hz	10 Hz	1 Hz	10 Hz	1 Hz
SWIM	40.85%	37.07%	2.37%	2.61%	4.34%	13.62%
UP	7.92%	6.32%	3.34%	3.70%	1.64%	3.79%
DOWN	3.85%	2.35%	6.21%	6.17%	0.99%	2.47%
TOTAL	52.62%	45.74%	11.88%	12.48%	6.97%	19.88%

(b) Second layer : Active swimming behavior percentage for 10 and 1 Hz algorithm. Percentages are made in total of the dataset analysed.

TABLE 3.3: Behaviors percentages

3.4.1 Ethogram

The ethogram calculated for the reference trajectory of sequence 6 shows a succession of all the different behaviors but *SURFACE* (Figure 3.6.a). The dive profile allows us to validate the prediction of the ethogram (Figure 3.6 b). When the dive profile has a strong downward curve, the behavior predicted is *DOWN*. For other sequences including a surfacing, the *SURFACE* behavior is well detected (data not shown). At the seabed, *REST* or *GROUND* are also predicted. The latter is complex to identify because it can be confused with the behavior of *SWIM* without changing depth.

The second layer of the ethogram (*H-PITCH*, *R-SWIM* or *O-SWIM* behaviors) is used to determine the main speed function to be used. The speed during *H-PITCH* is set using the OCDR function which requires an average pitch for the different active phases: *SWIM*, *UP* and *DOWN*. For *UP* and *DOWN*, the pitch is often low at the end of the ascent and descent phases. This is often because the turtle settles a little more horizontally and floats when the positive or negative buoyancy takes over the swim. Similarly, the average pitch is less strong and the times are longer for the ascent phases. The turtle ascent is with a shallower angle and is helped more by its buoyancy. With this information, we adapted the timing values and the thresholds that trigger the *H-PITCH* second layer behaviors. We used a trial and error method by comparing improvements in speed estimation. The chosen coefficients are 17° during 5 s for *H-PITCH* in *UP* behavior and 20° during 5 s for *DOWN* behavior.

Table 3.3 provides the percentage of the time in each behavior estimated from the algorithms at 1 Hz and 10 Hz for the whole reference data. The two algorithms provide very similar results with the highest differences for the *SWIM* and the *REST* behaviors.

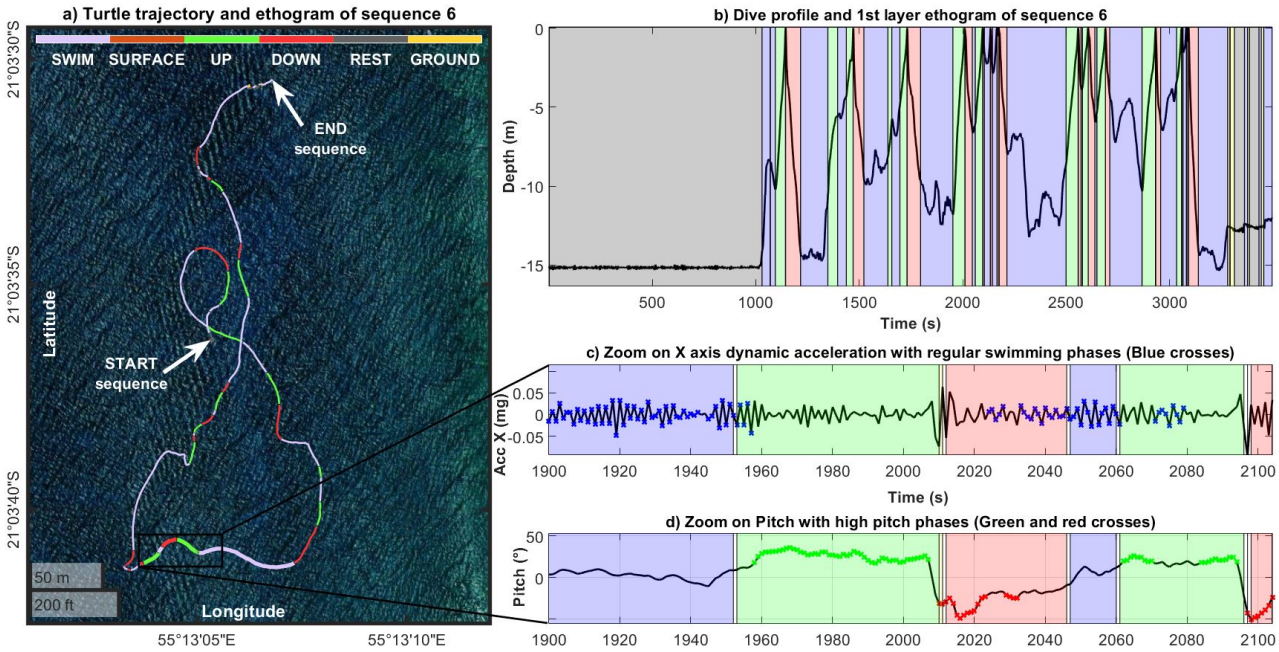


FIGURE 3.6: a) The horizontal trajectory of the turtle for sequence 6 with the ethogram first layer at 10 Hz. Each colored portion corresponds to a different behavior. b) The dive profile associated with this trajectory with the same color code. (c) and (d) Zoom on a part of the trajectory to highlight the behaviors of the second layer of the ethogram. In c) we have the dynamic acceleration on the x-axis of the turtle with blue crosses that represent the moments when our algorithm detects a *R-SWIM*. On d) we displayed the pitch and the green and red crosses represent the positive and negative *H-PITCH* phases respectively.

	Validation sequence		All sequences	
Speed function	**RMSE speed** (m/s)	**2DRMS** (m)	**RMSE speed** (m/s)	**2DRMS** (m)
$F10_E$ (Behavior)	0.1074	28.71	0.093	19.11
$F1_E$ (Behavior)	0.1157	34.82	0.1003	25.34
$F1_S$	0.1208	38.20	0.1073	33.01
$F10_S$	0.1567	35.45	0.1477	31.97

TABLE 3.4: Speeds RMSE and trajectories 2DRMS for the different speed functions. The validation sequence spans over 33.5 minutes and represents 24.24% of the dataset. 2DRMS comparison cannot be compared between datasets (e.g. validation data and full data) as the error is growing with time.

3.4.2 Accuracy comparison for speed and trajectory

The speed function $F10_E$ using the ethogram has an RMSE of 0.093 m/s for the speed estimates which represents an improvement of 15.37% compared to the $F1_S$ function (best without ethogram). For the trajectory estimates, the speed function $F10_E$ has a 2DMRS of 19.11 m which represents an improvement of 72.74% compared to the $F1_S$ function. The $F1_E$ function with ethogram has an RMSE of 0.1003 m/s for the speed and a 2DRMS of 25.34 m for the trajectory. The function which does not use the ethogram and is simply based on a regression between VeDBA and speed gives a lower accuracy on the speed and trajectory. The results for the validation data as well as for the complete data sets are presented in Table 3.4. To compare the whole reference data with the validation data, we only analyzed the RMSE on the speed. As the 2DRMS depends on the accumulated trajectory error over time, we cannot compare data sets of different sizes. For $F10_E$, the RMSE of speed is 0.1074 m/s and for $F1_S$, RMSE is 0.1157 m/s.

3.4.3 Comparison of power consumption

In this section, the influence of the proposed algorithms on tag consumption is studied. We used the power consumption presented in Chapter 2 for the "Turtle tracker" board. These data are available in Appendix C. In Table C.4 are displayed the consumption measurement. Consumption induced by GPS and sending LoRa signals are not added to the total but are given to understand the order of magnitude. A complete measurement report is available in Appendix C. The consumption estimation associated with the computing of the methods $F1_E$ and $F1_S$ at 1 Hz are equal to 0,196 mA and 0,227 mA. Consumption of $F10_E$ with a part of its algorithm at 10 Hz gives a power consumption 5.3% higher than $F1_E$ with 0.207 mA. The method using $F10_S$ also running with some functions at 10 Hz gives a consumption of 0.227 mA.

Compared to $F1_S$, the most accurate method without using behavior, we have an improvement of 13.6% for $F1_E$. Despite the acquisition and computing of the

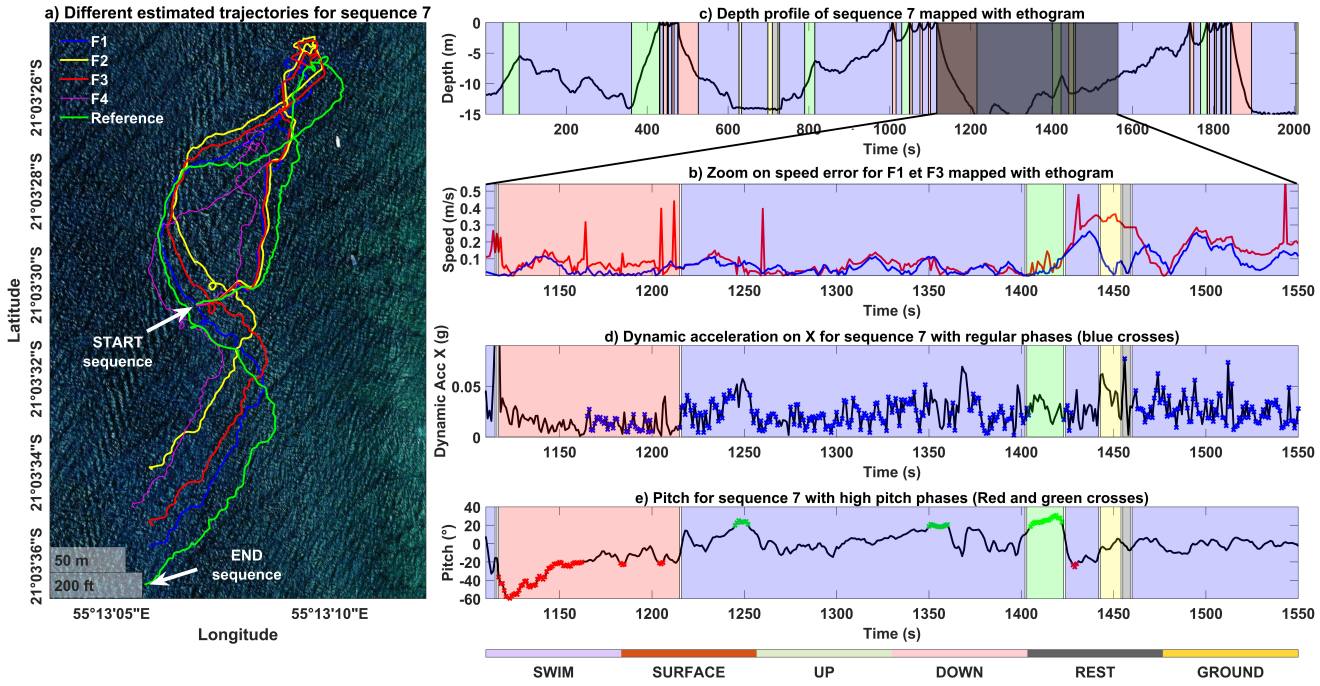


FIGURE 3.7: a) Trajectory estimation for the validation dataset with the different speed functions. For Figures b), c), d), e) the color code of the ethogram is defined at the bottom of the figure. The associated dive profile is displayed in b). The shaded part represents the zoom studied in the following inset. c) the speed error for the functions $F10_E$ and $F1_S$. We have chosen to display only the best ones with and without ethogram for readability reasons. The figure with all the curves is available in the supplementary materials. d) VeDBA and blue crosses are zones where the swimming is regular. e) pitch with the *H-PITCH* phases in red and green crosses.

e-compass variables at 10 Hz, the $F10_E$ method also shows a 9.7% improvement in consumption.

CPU/sensor	Timing (ms)	Consumption (mA)
CPU rest	Average	0.020 mA
CPU run	Average	5.5 mA
Acc + Mag 10 Hz	Average	0.089 mA
Pressure sensor	20	5.3 mA
LoRa message (222 byte)	650	38.5 mA
GPS	Average	30 mA

(a) Electrical consumption measurement on "IOT turtle tracker" board

Average electrical consumption (mA)			
$F10_E$	$F1_E$	$F1_S$	$F10_S$
0.207	**0.196**	**0.227**	**0.236**

(b) Total of the consumption for the different algorithms computed with the proposed speed functions

TABLE 3.5: Consumption of the board sensors/functions (A) and the average consumption for each algorithm (B). Total consumption on (B) includes accelerometer, magnetometer, pressure sensor and LoRa message consumption. GPS schedule is highly dependent on the application and induces a high consumption, so its consumption is given for information purposes.

3.5 Discussion

Here we show that the use of an ethogram for trajectory reconstruction for an embedded system can improve the accuracy as well as the power consumption. However, some biases are present in the data. The duration of the data collection is short and the analyses are done for a single animal.

To our knowledge, no study in marine environment uses an embedded ethogram to optimize the speed estimation and more globally the trajectory. Our method ensures the detection of animal behaviors with different swimming dynamics and then adapts the speed estimation function. Our solution, inspired by the one proposed in Wilson (27), ensures great adaptability, temporal flexibility and simplicity of use. To embed the algorithm on a bio-telemeter, the method is developed to optimize the computing time, and thus the consumption. This method also has the advantage of providing scientists with the ethogram along the animal's trajectory.

3.5.1 Ethogram improvement

Most embedded ethograms use fixed time windows on the acceleration data to detect behavior (25; 26). These methods do not allow temporal flexibility for detection. The advantage of the proposed method is that it is adaptable to multiple conditions for changing states. It can use single thresholds, timers, or occurrence counters of

any size as well as information from previous behaviors. All these conditions can be used independently or in parallel. To describe the swimming phases, two layers of the ethogram are proposed. The first one details the main phases and the second one details the active phases (regular swimming, swimming with strong pitch, other swim). Some behaviors can be added or removed in layer 1 or 2 according to the researchers' needs. For example, we identified quite simply the swimming phases when the turtle glides. In Figure 3.7.c after 2080 s, we can identify that at the end of the *UP* phase there is no more dynamic acceleration while the turtle changes depth. In our model, it did not give improvement and we chose not to keep it. The *GROUND* behavior and more generally the feeding phases of the turtle are not easy to determine with acceleration and depth. In Figure 3.6.a between 600 s and 700 s in the zone where the depth is stable, we detected some *GROUND* phases. By analyzing the reference trajectory we suspected that the turtle is feeding and has a speed close to zero at other times but the identification of these phases remains complex with the available data. The parallel use of video would help to fine-tune the parameters to detect these phases further to improve the trajectory and biological data of the animal. False detection of behavior between dynamic and static leads to the largest error as the speed is set to 0 in static behavior. Our goal is not to validate the ethogram as a behavioral answer tool, but as a means to improve trajectories. If scientists want to use it to answer environmental questions, then it must be validated with reference data such as video analysis.

3.5.2 Analysis of the speed estimation

The $F10_S$ function shows us the limitation of using the acceleration derivatives (VedBA here) when we have strong accelerations due to the external environment. In our case, there are strong dynamic accelerations when the turtle is arriving at the surface (Figure 3.7.c (around 1210 s) and when it is touching the ground (Figure 3.7.c around 1250 s). In $F10_S$ we used all the data (except the *REST* phase) to calculate the regression coefficients between VeDBA and speed. With these extreme values, the weights used in the regression by the derivatives of the acceleration become minimal and its result is close to the average speed. Accelerations are interpreted in the opposite way to their real effect. They will give a high velocity when the real velocity approaches zero. The ethogram associated with the regular swimming function allows us to solve this problem simply by isolating these special cases and by applying the speed estimation functions only for the defined phases. The difference in velocity for $F10_E$ between the validation and training data is small which shows the efficiency of the algorithm. On the contrary, we had a bigger difference for $F1_E$ and $F1_S$. This difference is due to the fact that during the validation sequence, the turtle presents regular slow and fast swimming phases compared to the fixed average speeds. The regression with the acceleration on x-axis used in $F10_E$ allows us to estimate the speed of these phases better. We notice that $F1_E$ and $F1_S$ give quite close results. Indeed, at 1 Hz, except for the OCDR function, the speeds used are fixed. The gains between the two are due to the improvements brought by the use of phases with the OCDR and to the better detection of the *REST* but these phases represent both only 6% of the time and hence a small global difference. On Figure 3.7.b and d in the *UP* phase after 1400 s, the speed estimation with $F1_E$ gives

a better estimation thanks to the OCDR applied on a phase. Still, on Figure 3.7.b at 1200 s at the end of *DOWN* phase, we noticed that $F1_S$ confuses the phases where the turtle glides with *REST* phases and gives an important error on the speed estimation. Our method allows us to correct these errors. Further improvements are possible with the use of OCDR. Some animals show a high difference between their incidence and displacement angles, called the pitch anomaly (7). It can lead to errors in the use of the OCDR. With the reference data and the dive variables, it is possible with regression to correct these errors. In our case the improvement is minimal, so we have kept the basic technique which gives less computation.

The speed estimation model still gives some estimation errors. Several improvements can be added such as accounting for the previous speed. During the transition between a *REST* phase and an active phase, an acceleration period could be modeled. For these improvements, we are confronted with the accuracy of our reference speed. To improve it, several solutions are possible. The first would be the implementation of more complex filtering using the acoustic positions, the IMU data and the depth. Hardware improvements are also possible with a better layout of the acoustic receivers, but this requires profound changes in the ASV.

3.5.3 Sampling frequency on acceleration derivated data

Speed estimation function at 1 Hz gives reduced improvement from acceleration proxies. The method defined to detect the *REST* phase with VeDBA is less efficient at 1Hz, reducing the *REST* behavior percentage. For $F1_E$, we defined regression coefficients with the variables from the accelerations on x-axis given with the stroke function at 1 Hz. At this frequency, the loss of precision on the frequency and the peak-to-peak value does not improve the speed estimation. In the same way, with the sampling at 1 Hz, it is difficult to detect the *GROUND* phases using the jerk.

3.5.4 Influence on the electrical consumption

The advantage of 1 Hz algorithms is to decrease the number of executions of some functions resulting in lower power consumption. The difference is not proportional because in both algorithms some functions such as orientation are sampled at 1 Hz. Our two functions using ethograms offer a lower consumption than the ones without. Even the $F10_E$ at 10 Hz gives a lower power consumption than the algorithm using the $F1_S$ function at 1 Hz despite its simplicity. This is due to the reduction of computing by using the behavioral phases. For example, for the *H-PITCH* function, the $F1_S$ function will process the $tan()$ function at each iteration rather than only once over the time of the behavior for $F10_E$ and $F1_E$. In addition, for the algorithm with the $F10_E$ and $F1_E$ functions, the ethogram allows switching to a low energy mode with only the calculation of the VeDBa while switching off the pressure sensor during 25% of the *REST* behavior.

3.5.5 Method flexibility

The behaviors proposed for the ethogram are adaptable to almost all marine mammals. For the main variables of layer 1, it is possible to determine the coefficients and timings by simply analyzing the accelerometer and depth data. For the active

phase, variable thresholds using dynamic acceleration proxies could be adapted to the animal's size. To prove this, it would be necessary to perform the test on turtles of different species and sizes. In the study, we proposed to estimate the speed using accelerometer data or fixed speeds. Depending on the tag sensors, other methods, such as a speed sensor, can replace/complete them and be coupled with the ethogram behaviors. Many research teams have already developed their own ethograms. They have most of the time common behaviors with the one proposed. Depending on the capacity of the tags, our method allows using another ethogram instead or running in parallel. Our second layer used to estimate the speed in the active phases can be detached from the first layer in order to be associated with another method.

3.6 Conclusion

With the new transmission technologies, new perspectives open up regarding the type of data that can be transmitted. However, bio-telemeters generally have very limited computing power, memory, and batteries. The method proposed using the ethogram allows improving the estimation of the trajectory and reducing the power consumption. It then offers the scientists the possibility to post-process the received trajectories for deeper analysis of the movements of the animal and the use of its environment, while allowing longer deployments. The innovative method developed for acquiring reference data offers the scientists precise reference data at a fine temporal scale in an environment where their acquisition is very complex. These data allow the development of more accurate speed estimation functions and/or optimized for onboard computing. This work encourages using behavior to effectively improve many parts of data processing and analysis in bio-logging.

Acknowledgements

This work is supported by the "IOT" project funded by FEDER INTERREG V and Prefet de La Réunion: grant #20181306-0018039 and the "Contrat de Convergence et de Transformation de la Préfecture de La Réunion". We thank the marine reserve of Reunion Island and DEAL with the data collection is possible under the permit 2020-45. We are grateful to IDOCEAN for their support and loan of material and the TMSOI for the services of their pilot, boat, and diver. Finally, I would like to thank the colleagues from IFREMER who helped me with the data collection: Magali Duval, Blandine Brisset, Denis De Oliveira, and Mohan Julien.

Data accessibility

Data is processed using MATLAB (2020b) and customized scripts. Tag data are analyzed with "Signal analyzer" tool box in temporal and frequency domain. Regressions are developed with the "Regression learner" toolbox. Embedded programs are coded and compiled via the Arduino IDE in C++. All the data, scripts, codes, and supplementary materials are available on GitHub[7].

[7] https://github.com/pierregoge/Behavioral-based-Dead-Reckoning

Bibliography

[1] S. Andrzejaczek, A. C. Gleiss, K. O. Lear, C. B. Pattiaratchi, T. K. Chapple, and M. G. Meekan, "Biologging Tags Reveal Links Between Fine-Scale Horizontal and Vertical Movement Behaviors in Tiger Sharks (Galeocerdo cuvier)," *Frontiers in Marine Science*, vol. 6, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2019.00229

[2] J. Korpela, H. Suzuki, S. Matsumoto, Y. Mizutani, M. Samejima, T. Maekawa, J. Nakai, and K. Yoda, "Machine learning enables improved runtime and precision for bio-loggers on seabirds," *Communications Biology*, vol. 3, no. 1, pp. 1–9, 2020. [Online]. Available: https://www.nature.com/articles/s42003-020-01356-8

[3] R. J. M. Nuijten, T. Gerrits, J. Shamoun-Baranes, and B. A. Nolet, "Less is more: On-board lossy compression of accelerometer data increases biologging capacity," *Journal of Animal Ecology*, vol. 89, no. 1, pp. 237–247, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/1365-2656.13164

[4] R. P. Wilson, N. Liebsch, I. M. Davies, F. Quintana, H. Weimerskirch, S. Storch, K. Lucke, U. Siebert, S. Zankl, G. Müller, I. Zimmer, A. Scolaro, C. Campagna, J. Plötz, H. Bornemann, J. Teilmann, and C. R. McMahon, "All at sea with animal tracks; methodological and analytical solutions for the resolution of movement," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 54, no. 3, pp. 193–210, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967064507000045

[5] O. R. Bidder, J. S. Walker, M. W. Jones, M. D. Holton, P. Urge, D. M. Scantlebury, N. J. Marks, E. A. Magowan, I. E. Maguire, and R. P. Wilson, "Step by step: reconstruction of terrestrial animal movement paths by dead-reckoning," *Movement Ecology*, vol. 3, no. 1, p. 23, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0055-4

[6] P. J. Wensveen, L. Thomas, and P. J. O. Miller, "A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales," *Movement Ecology*, vol. 3, no. 1, p. 31, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0061-6

[7] C. Laplanche, T. A. Marques, and L. Thomas, "Tracking marine mammals in 3D using electronic tag data," *Methods in Ecology and Evolution*, vol. 6, no. 9, pp. 987–996, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12373

[8] T. Photopoulou, P. Lovell, M. A. Fedak, L. Thomas, and J. Matthiopoulos, "Efficient abstracting of dive profiles using a broken-stick model," *Methods in Ecology and Evolution*, vol. 6, no. 3, pp. 278–288, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12328

[9] S. L. Cox, F. Orgeret, M. Gesta, C. Rodde, I. Heizer, H. Weimerskirch, and C. Guinet, "Processing of acceleration and dive data on-board satellite

relay tags to investigate diving and foraging behaviour in free-ranging marine predators," *Methods in Ecology and Evolution*, vol. 9, no. 1, pp. 64–77, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12845

[10] J. Leander, J. Klaminder, M. Jonsson, T. Brodin, K. Leonardsson, and G. Hellström, "The old and the new: evaluating performance of acoustic telemetry systems in tracking migrating Atlantic salmon (Salmo salar) smolt and European eel (Anguilla anguilla) around hydropower facilities," *Canadian Journal of Fisheries and Aquatic Sciences*, vol. 77, no. 1, pp. 177–187, 2020. [Online]. Available: https://cdnsciencepub.com/doi/abs/10.1139/cjfas-2019-0058

[11] G. H. Elkaim, E. B. Decker, and B. Wright, "Marine Mammal Marker (MAMMARK) Dead Reckoning Sensor for In-Situ Environmental Monitoring."

[12] Y. L. Bras, J. Jouma'a, and C. Guinet, "Three-dimensional space use during the bottom phase of southern elephant seal dives," *Movement Ecology*, vol. 5, no. 1, p. 18, 2017. [Online]. Available: https://doi.org/10.1186/s40462-017-0108-y

[13] K. Aoki, M. Amano, K. Mori, A. Kourogi, T. Kubodera, and N. Miyazaki, "Active hunting by deep-diving sperm whales: 3D dive profiles and maneuvers during bursts of speed," *Marine Ecology Progress Series*, vol. 444, pp. 289–301, 2012.

[14] T. Fukuoka, T. Narazaki, C. Kinoshita, and K. Sato, "Diverse foraging habits of juvenile green turtles (Chelonia mydas) in a summer-restricted foraging habitat in the northwest Pacific Ocean," *Marine biology*, 2019. [Online]. Available: https://dx.doi.org/10.1007/s00227-019-3481-9

[15] T. Narazaki, K. Sato, K. J. Abernathy, G. J. Marshall, and N. Miyazaki, "Sea turtles compensate deflection of heading at the sea surface during directional travel," *Journal of Experimental Biology*, vol. 212, no. 24, pp. 4019–4026, 2009. [Online]. Available: https://doi.org/10.1242/jeb.034637

[16] J. Gabaldon, E. L. Turner, M. Johnson-Roberson, K. Barton, M. Johnson, E. J. Anderson, and K. A. Shorter, "Integration, Calibration, and Experimental Verification of a Speed Sensor for Swimming Animals," *IEEE Sensors Journal*, vol. 19, no. 10, pp. 3616–3625, May 2019.

[17] M. Whitford and A. P. Klimley, "An overview of behavioral, physiological, and environmental sensors used in animal biotelemetry and biologging studies," *Animal Biotelemetry*, vol. 7, no. 1, p. 26, 2019. [Online]. Available: https://doi.org/10.1186/s40317-019-0189-z

[18] R. Wilson, E. Shepard, and N. Liebsch, "Prying into the intimate details of animal lives: Use of a daily diary on animals," *Endangered Species Research*, vol. 4, pp. 123–137, 2008.

[19] E. L. Shepard, R. Wilson, N. Liebsch, F. Quintana, A. G. Laich, and K. Lucke, "Flexible paddle sheds new light on speed: a novel method for the remote measurement of swim speed in aquatic animals," 2008.

[20] P. J. O. Miller, M. P. Johnson, P. L. Tyack, and E. A. Terray, "Swimming gaits, passive drag and buoyancy of diving sperm whales Physeter macrocephalus," *Journal of Experimental Biology*, vol. 207, no. 11, pp. 1953–1967, 2004. [Online]. Available: https://doi.org/10.1242/jeb.00993

[21] D. E. Cade, K. R. Barr, J. Calambokidis, A. S. Friedlaender, and J. A. Goldbogen, "Determining forward speed from accelerometer jiggle in aquatic environments," *Journal of Experimental Biology*, vol. 221, no. 2, p. jeb170449, 2018. [Online]. Available: https://doi.org/10.1242/jeb.170449

[22] O. R. Bidder, L. A. Qasem, and R. P. Wilson, "On Higher Ground: How Well Can Dynamic Body Acceleration Determine Speed in Variable Terrain?" *PLoS ONE*, vol. 7, no. 11, p. e50556, 2012. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3511514/

[23] O. R. Bidder, M. Soresina, E. L. C. Shepard, L. G. Halsey, F. Quintana, A. Gómez-Laich, and R. P. Wilson, "The need for speed: testing acceleration for estimating animal travel rates in terrestrial dead-reckoning systems," *Zoology*, vol. 115, no. 1, pp. 58–64, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0944200611000985

[24] L. Qasem, A. Cardew, A. Wilson, I. Griffiths, L. G. Halsey, E. L. C. Shepard, A. C. Gleiss, and R. Wilson, "Tri-axial dynamic acceleration as a proxy for animal energy expenditure; should we be summing values or calculating the vector?" *PloS One*, vol. 7, no. 2, p. e31187, 2012.

[25] H. Yu, J. Deng, R. Nathan, M. Kröschel, S. Pekarsky, G. Li, and M. Klaassen, "An evaluation of machine learning classifiers for next-generation, continuous-ethogram smart trackers," *Movement Ecology*, vol. 9, no. 1, p. 15, 2021. [Online]. Available: https://doi.org/10.1186/s40462-021-00245-x

[26] S. P. le Roux, J. Marias, R. Wolhuter, and T. Niesler, "Animal-borne behaviour classification for sheep (Dohne Merino) and Rhinoceros (Ceratotherium simum and Diceros bicornis)," *Animal Biotelemetry*, vol. 5, no. 1, p. 25, 2017. [Online]. Available: https://doi.org/10.1186/s40317-017-0140-0

[27] R. P. Wilson, M. D. Holton, A. di Virgilio, H. Williams, E. L. C. Shepard, S. Lambertucci, F. Quintana, J. E. Sala, B. Balaji, E. S. Lee, M. Srivastava, D. M. Scantlebury, and C. M. Duarte, "Give the machine a hand: A Boolean time-based decision-tree template for rapidly finding animal behaviours in multisensor data," *Methods in Ecology and Evolution*, vol. 9, no. 11, pp. 2206–2215, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13069

[28] H. Yu, J. Deng, T. Leen, G. Li, and M. Klaassen, "Continuous on-board behaviour classification using accelerometry – a case study with

a new GPS-3G-Bluetooth system in Pacific Black Ducks," *Methods in Ecology and Evolution*, vol. n/a, no. n/a, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13878

[29] M. Lichtenstein and G. Elkaim, "Efficient GPS Scheduling in Wildlife Tags using an Extended Kalman Filter-based Uncertainty Suppression Strategy," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, pp. 1472–1475, iSSN: 2153-3598.

[30] A. Makni, H. Fourati, and A. Y. Kibangou, "Energy-Aware Adaptive Attitude Estimation Under External Acceleration for Pedestrian Navigation," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1366–1375, Jun. 2016, conference Name: IEEE/ASME Transactions on Mechatronics.

[31] P. Gogendeau, S. Bonhommeau, H. Fourati, D. De Oliveira, V. Taillandier, A. Goharzadeh, and S. Bernard, "Dead-Reckoning Configurations Analysis for Marine Turtle Context in a Controlled Environment," *IEEE Sensors Journal*, vol. 22, no. 12, pp. 12 298–12 306, Jun. 2022, conference Name: IEEE Sensors Journal.

[32] J. Wu, Z. Zhou, H. Fourati, and Y. Cheng, "A Super Fast Attitude Determination Algorithm for Consumer-Level Accelerometer and Magnetometer," *IEEE Transactions on Consumer Electronics*, vol. 64, p. 1, 2018.

[33] K. S. Ydesen, D. M. Wisniewska, J. D. Hansen, K. Beedholm, M. Johnson, and P. T. Madsen, "What a jerk: prey engulfment revealed by high-rate, super-cranial accelerometry on a harbour seal (Phoca vitulina)," *Journal of Experimental Biology*, vol. 217, no. 13, pp. 2239–2243, 2014. [Online]. Available: https://doi.org/10.1242/jeb.100016

[34] "Positioning Error - Navipedia." [Online]. Available: https://gssc.esa.int/navipedia/index.php/Positioning_Error

Chapter 4

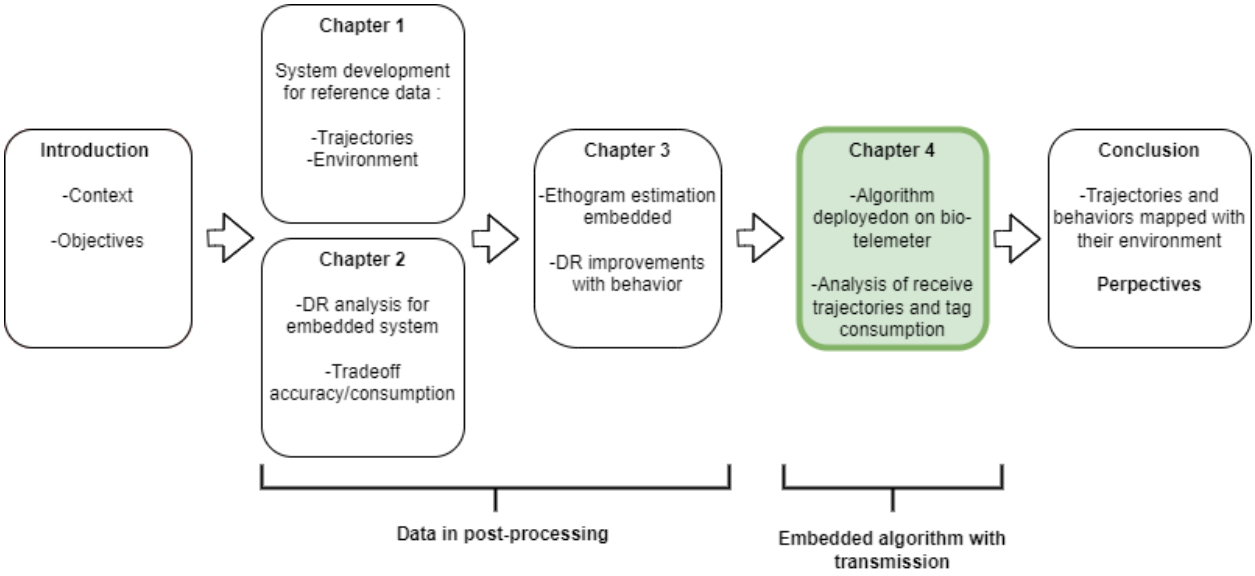
Transmission of trajectories and ethogram by LoRaWAN for long-term deployment with embedded calculation

Authors : Gogendeau Pierre, Bonhommeau Sylvain, Fourati Hassen, Mohan Julien and Bernard Serge

Article status : In preparation for *Methods in Ecology and Evolution*

FIGURE 4.1: Fields testing in Reunion lagoon

Key words: Bio-telemeter, Embedded calculation, Dead-reckoning, Ethogram



Reminder of the chapter transition graph. The current chapter is highlighted in green

Abstract

- Tracking marine animals in the wild is a key challenge to improve our knowledge on spatial ecology, e.g. migration patterns, habitat use, spawning and feeding locations. Electronic tags including a geolocation system (GNSS, Argos or light-based) and radio-transmission have been widely used to collect information on the 2D or 3D trajectories of animals. Due to the constraints of the marine environment and the small size of these devices, there is currently no tag able to transmit the fine-scale underwater trajectories of marine animals over long periods ($>$ several months). The objective of this work is to transmit fine-scale 3D trajectories and associated ethogram of marine animals using a low-cost and open-source electronic tag.

- A DR algorithm using the animal behavior information is embedded into an electronic tag while limiting computation and reducing power consumption to achieve long-term data collection and transmission. The LoRa transmission system ensures low power consumption and a long data transmission range. The information on 3D trajectories and behaviors is compressed with a minimum loss of accuracy and sent on the free LoRa network. This solution is tested for applications on surface-breathing animals living close to the shore.

- The bio-telemeter successfully transferred 3D trajectories and behaviors along the trajectories through the LoRa network. With a consumption of 0.39 mA, the tag can collect data for 138 days with a small battery. Different solutions for data storage and visualization are proposed from custom servers to online and free web services.

- The proposed developments offer a low-cost and long-term solution for tracking fine-scale movement and associated behavior of marine animals. While the

current tags transfer data regularly when the animal comes at the surface to breathe, the messages could be stored over a deployment period and sent after the tag popped up and float at the surface for applications on non-surface-breathing species such as fish. The amount of data and their resolution collected and transferred open new research perspectives in marine ecology.

4.1 Introduction

In the marine environment, the analysis of behaviors and underwater trajectories of animals at fine temporal and spatial scales requires a large dataset which is complex to transmit by bio-telemeter. These data generally include the values from a pressure sensor, a 3-axis accelerometer, a 3-axis magnetometer with second or infra-second sampling frequency for deployment duration from several hours to weeks. DR algorithms use these inputs to estimate 3D tracks for pinnipeds (1; 2), whales (3; 4; 5), turtles (6), and fish (7). The same data can be used for behavioral studies based on the dynamic acceleration of animals (8; 9; 10). The required memory is however substantial (11). For these reasons, most of the studies on animal behaviors or underwater trajectories post-process data from a bio-logger. This technique however requires to recover the tag which can be a complex task, i.e. the use of a dropoff system (7; 4; 12) or the recapture of the animal (1). The case study of this chapter remains on the juvenile green turtles (*Chelonia mydas*) traveling up to several kilometers but remaining close to the coast.

In addition, the power consumption associated with the acquisition of data from these sensors does not allow several-month studies in continuous acquisition. For example, the widely deployed daily diary (DD) tag (13) developed by Wilson and al. with a 1035 mAh battery allows the acquisition of raw data from 3-axis accelerometer (10 Hz), 3-axis magnetometer (5 Hz) and speed, pressure, temperature, light, humidity sensors (0.2 Hz) for about 7 days.

Bio-telemeter refers to tags transmitting the data by satellite, radio, or acoustic systems. Acoustic transmissions are limited by their range of a few hundred meters and therefore require dense receiver networks (14; 15). Satellite communication has been used for many years in bio-logging but have a limited data rate and a low transmission success rate. Argos system is the most used in bio-logging with several thousand active tags (16). Argos 3 technology can send 256-byte messages every 90 s. (17; 18; 19). However, the transmission is limited to satellite visibility during short surfacing behaviors of marine animals or when the tag floats at the surface. Typical data transmitted are dive information, dive timing, low-resolution depth profile, GPS position or Argos position (20). Recently, new technology and algorithms for satellite bio-transmitter have emerged and allow to send compressed data such as inflection points of a dive profile (21; 18).

The Swansea University has developed a bio-transmitter using GSM communication (22). This tag, called phone-tags , tested on seals can send 160-byte messages up to 35 km (23; 22). Its two main drawbacks are the delay of network registration and the power consumption. The median delay is 9.6 s. Depending on the studied animals, it can be impossible to send messages during short breathing events. During the registration delay the average power consumption is 40 mA increasing the global consumption of the tag. Using GSM and Argos also implies monthly fees of

around \$25/months and \$126/tag. For terrestrial bio-logging, some systems allow transmitting large volumes of data via Wireless Local Area Network (WLAN) using Bluetooth (BLE) (24) or Wi-Fi (11) technologies. It requires close reception stations because their range is a few hundred meters through the air. These methods are difficult to apply to marine animals.

With the emergence of the Internet of Thing (IoT), new means of radio communication are possible with the Low-power Wide-Area Networks (LWPAN). The LoRa (Long Range) (25; 26) and Sigfox (27) communications are the most deployed (27). These two techniques require receiving stations called gateway (GW) which constitute private and public networks that are accessible broadly. Using a private network may require a subscription ($< 2€$ per object per month). Users can deploy their own network, open or not, and transmissions are free. These communications offer a data rate up to 50 kBps for LoRa and 100 Bps for Sigfox. The latter is limited to 144 messages of 32 bytes per day compared to LoRa messages with 243 bytes without limiting the number of messages sent (except the duty cycle). The size of the message is however too limited to transmit raw sensor data sampled at several Hertz. It is therefore necessary to process the embedded data and compress them before sending. The method developed in Chapter 3 to process the data for trajectory and ethogram estimation can be applied in this context to reduce the volume of data to be sent. Here, we further develop this method by adding compression steps for the trajectory and the ethogram.

The trajectory compression can rely on different available algorithms for reducing polylines (28; 29). Each of them depend on the application but the most common and effective ones are the Douglas-Peucker algorithm (DP) (30) , Visvalingam–Whyatt (VVW) (31), N^{th} Point (N-Points) (28), Reumann-Witkan (32) or Opheim (33). These algorithms have been used to compress dive profiles with the aim of optimizing the data before sending them (34; 21; 18). The algorithm used in these studies is called "broken stick model" similar to the Douglas-Peucker (DP) algorithm. To our knowledge, none of the bio-transmitters for marine animals allows sending 3D track information, even reduced and compressed.

For our application, "online" algorithms (29) cannot be used as the data can only be sent when the animal surfaces. "Batch" or "offline" offer better solutions as they can be applied once the trajectory is completed. Our selection criteria for the compression algorithms combines efficiency and complexity to reduce the computing time on the microcontroller while keeping a sufficient resolution. The algorithm applied on the complete trajectory needs to stop when it reaches a predefined number of points corresponding to the maximum message size. We selected the DP, VVW and N-Points algorithms for comparison purposes.

Here we present the results for data compression, reception, and decryption of messages from a new generation of bio-transmitter using LoRaWAN radio communication (35). For this tag, we have built a specific electronic card with an embedded algorithm to compute the trajectory, ethogram and dive data for marine animals. To illustrate this, several case studies in the field are analyzed. Finally, we estimate the associated energy consumption of the sampling and calculations.

4.2 Materials and methods

4.2.1 Hardware description

"Turtle tracker" board

In the context of the IOT (Indian Ocean turtle) project, we have developed a multi-sensor electronic board (Figure 4.2). The board is 36 x 20 x 6 mm for 4.64 g. All the files needed to build the board are open-source and available on the following GitHub[1]. The main components of the board are presented in Table 4.1.

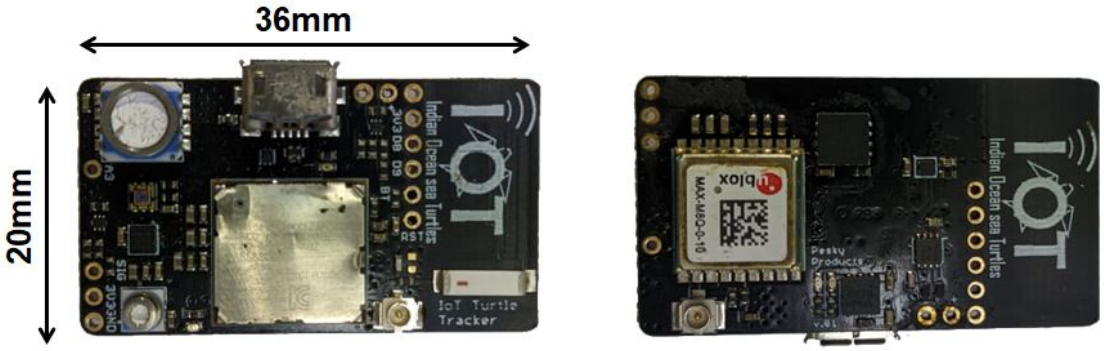


FIGURE 4.2: The "Turtle tracker" board

The STM32L082 microcontroller has 20 kBytes of RAM memory and of 19 kBytes EEPROM. The board is programmable through a micro USB port. It includes a Li-on battery charger STBC08 also using the USB port. The circuit allows measuring the battery level. The board includes two pressure sensors. The first one with a maximal resolution of 2.5 mm up to 2 m measures precisely the shallow depths. It is mainly used for surface detection. The second one with a maximal resolution of 20 mm up to 300 m is used to acquire animal depth. For surface detection, there is a dedicated analog pin to measure the conductivity. For LoRa communication, the board has a LoRa module SX1276 with an u.FL connector for an external antenna and an internal LoRa ceramic antenna in option (Johanson 0900AT43A0070).

Casing of the bio-telemeter "IOT Turtle tag"

The tag is to be attached to the hard-shell marine turtles. The tag is 9 x 4.5 x 4 cm for 124 g in water (Figure 4.3). In this first version, the battery is a Lithium-Ion 1.3 Ah, 3.7V. The gray housing is printed in Tough 1500 *Formlabs* resin. The board is placed inside and filled with dielectric oil. The back of the tag is sealed with a flexible membrane. The membrane and the oil enable the equi-pressure inside the tag to measure the depth without direct contact of the pressure sensor with the seawater. There are two electrodes, one at the front and one at the back, to measure the conductivity with the dedicated analog pin and help to detect the surface. In the front, there is a 4-pin connector for the USB connection allowing to charge and

[1] https://github.com/pierregoge/Turtle-tracker-tag/tree/main/Schematic_and_PCB/Turtle_Tracker_V1b

Component	Reference	Principal specification
Microcontroller	CMWX1ZZABZ	Murata microcontroller with ST32L082 and semtech LoRa module SX1276 (Bus : SPI)
E-compass	LSM303AGR	Composed of an accelerometer and a magnetometer (Bus : I2C)
Pressure and Temperature sensor	MS5837-30BA	Pressure up to 30 bars (300 m). Resolution 20 mm (Bus : I2C)
Pressure and Temperature sensor	MS5803-02BA	Pressure up to 0.2 bars (2 m). Resolution 2.5 mm (Bus : SPI)
GNSS	MAX M8Q-0-10	Concurrent reception of up to 3 GNSS constellations. uFl connector for external antenna (Bus : Serial)
Flash memory	MX25R6435FZAI	8 MByte NOR flash memory (Bus : SPI)
Battery Charger	STBC08	500 mA fixed charge current
Voltage regulator	MCP1810T-33	Quiescent Current of 20 nA

TABLE 4.1: "Turtle tracker" board main electronic components

program the tag. During deployment, it is protected with silicone grease. The horn shape on the front protects the front electrode and the LoRa external antenna from impacts (Figure 4.3 in Black).

4.2.2 Software architecture

In order to develop a solution to program the tag as simply as possible and be adapted to the greatest number of users, the code is implemented using the Arduino language and its IDE. A specific core for the STM32L0 is available to use the Arduino software. This open-source core has been developed by Thomas Roell[2]. For the sensors, we use libraries made by Kris Winner[3] with slight modification. The code and libraries used in the algorithm are available in the GitHub[4].

The algorithm to estimate the 3D trajectories using the animal behavior follows the principle described in Chapter 3. Here we describe the modification added to the ethogram to be implemented easily in Arduino as well as the compression algorithm used and the message encoding. The data was collected with the tag mounted on a swimmer in the Saint Gilles lagoon where the depth does not exceed 1 m.

[2] https://github.com/GrumpyOldPizza/ArduinoCore-stm32l0

[3] https://github.com/kriswiner/CMWX1ZZABZ/tree/master/TurtleTracker/AssetTracker_IoT_Cricket

[4] https://github.com/pierregoge/Turtle-tracker-tag/tree/main/Software/Arduino_code/IoT_Turtle_F2_NED

FIGURE 4.3: Bio-transmitter "IoT tag".

Ethogram

The algorithm structure has been developed around an ethogram calculated onboard. It manages different parts of the system, such as the choice of sensors, the estimation of speed and the message transmission. The version of the ethogram corresponds to the one proposed at 1 Hz with only layer 1 as described in Chapter 3. The transitions between behaviors are done using a variable occurrence counter and the last state. The advantage is that it allows temporal flexibility, simple adaptability and can be embedded on system with small resources.

For this validation test, the ethogram is simplified to a minimal version and adapted to the environment. This version only includes the behaviors *SURFACE*, *SWIM*, *REST* following the diagram in Figure 4.4. The behaviors of swimming *UP* and *DOWN*, *GROUND* are removed as the swimmer remained at the surface/subsurface.

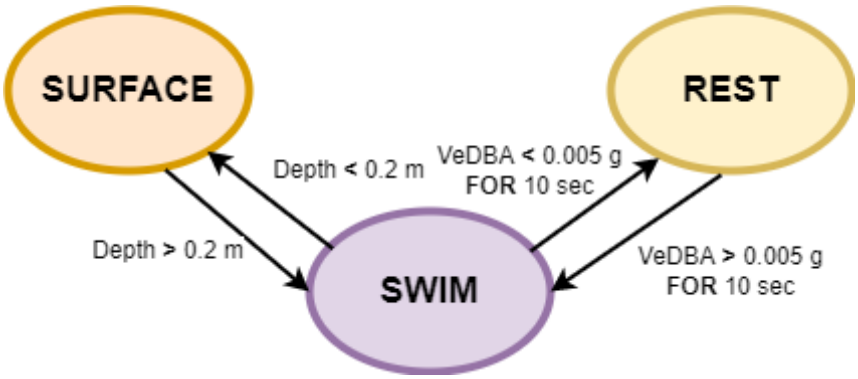


FIGURE 4.4: Transition diagram of the algorithm ethogram. The algorithm starts at the *SURFACE* and go to *SWIM* behavior when the depth is higher than 0.2 m. When the VeDBA is lower than 0.005 g for 10 s, the behavior state switches to *REST*. During swimming behavior, if the depth is lower than 0.2 m, the dive is supposed to be completed, the trajectory algorithm is run on the whole dive with data compressed and sent, and the behavior state switches to the *SURFACE* mode

The *REST* behavior corresponds to the phases when the swimmer is immobile and without drifting. To determine this state, the VeDBA (36) is calculated which translates the swimmer's activity via his dynamic acceleration. During these presumed immobile phases, sampling and computing are adapted by using only the accelerometer to calculate the VeDBA. To switch to *REST*, the current behavior must be SWIM, and the VeDBA is less than 0.005 g during 10 iterations. The *SURFACE* behavior corresponds to swimmer depth less than 0.2 m deep. It considers the swimmer immobile in this phase, but subjected to the marine current. In *SURFACE*, the sending of messages is possible. This behavior can only be followed by a *SWIM* behavior. The *SWIM* behavior corresponds to the phases when the swimmer is moving. The transition is done from *SURFACE* when the depth is higher than 0.2 m and from *REST* when the VeDBA is higher than 0.005 g for 10 iterations.

As described in Chapter 3, the speed estimation is behavior-dependent. For the *SWIM* behavior, a fixed swimming speed is set to 0.52 m/s. The speed is determined by the average speed of the active phases for the data in Chapter 2 because the operational context for the acquisition of the data of this study are the same. For *SURFACE* phases the speed is 0 m/s or equal to the sea surface current if available. For *REST* phases, the speed is 0 m/s. The estimation of the speed can be implemented with more complex models according to the needs and the available data of the users. In this study on a swimmer in a controlled environment, the aim is to show the overall performance of the algorithm rather than to analyze its accuracy in details.

Structure of the algorithm

The structure of our algorithm is described in Figure 4.5. The program is clocked by an interrupt at 1 Hz which wakes up the microcontroller to sample the sensors and process the necessary functions. The program runs in separate sequences. When the swimmer returns to the surface after the message has been processed and sent, a new sequence is started in *SURFACE* mode. The surface is defined by a depth of lower than 0.2 m. The algorithm can be used in two different ways. The first is if a GPS is available and activated/used. It allows reconstructing and sending geo-tracks. This type of trajectory uses the GPS position acquired on the surface or the last geolocated position of the previous dive and adds the new trajectory. Conversely, if the GPS is not present or activated, the algorithm gives "pseudo-tracks", i.e. non-geolocated trajectories.

After the new dive, the turtle is at the surface and the algorithm enters the GPS management part. If the GPS is not activated, the program samples the pressure every second and waits until the swimmer depth exceeds the 0.2 m threshold to enter the next phase and exit the surface behavior. If the GPS is activated, a GPS position is searched for as long as the animal is at the surface or the GPS time-out is not reached. In our case, the algorithm turns on the GPS during a cumulative time of 30 s every 2 h which corresponds to the half the time during which the ephemeris are kept in the GPS memory. The user for this phase can implement his own GPS scheduling algorithm. This scheduling is crucial because the GPS is an important source of power consumption. Its fine-tuning is however out of the scope of the study.

The next part is the sampling of the sensors, the calculation of the state variables, and the construction of the ethogram. The microcontroller exits the SLEEP phase with the interruption of the real time clock (RTC). The current behavior then conditions the algorithm. If it is not *REST*, it performs its normal computing, reading the accelerometer, magnetometer, and pressure sensor. The data from the accelerometer and the magnetometer are then filtered with a moving average low-pass filter. The variable VeDBA and then the orientation are computed. The orientation is estimated with the SAAM algorithm (37) chosen because it uses few computational resources (See Chapter 2). The data are then processed by the ethogram algorithm defined in 4.2.2. At the beginning of this phase, if the animal is in *REST*, then we only perform the reading of the accelerometer, the calculation of VeDBA and the ethogram. This simplification allows us not to use the magnetometer and the pressure sensor to reduce the computing. The next part is specific to our algorithm and is activated when there is a change in behavior. The calculation of speed and trajectory is behavior-dependent, however due to the occurrence counters in the change of behavior, there is a delay and the transition is retroactive. For example, it takes 10 s to validate the transition from *SWIM* to *REST*. When the change to *REST* is validated, we apply the speed and path reconstruction for *SWIM* from its start to the current iteration minus the 10 s. For each behavior transition, we calculate the speed of the swimmer, its trajectory, and the total distance covered. We repeat the phases of sampling of the sensors, calculation of the variables and calculation of the trajectory at a frequency of 1 Hz, until the condition of end of diving is reached, i.e. when the depth is lower than 0.2 m. When the swimmer reaches the surface, we calculate the trajectory and the distance covered with the speed calculated for the last behavior. The complete trajectory is then read and compressed. Two compression algorithms are used and will be presented in the next section "Compression".

In the next steps (in blue on the Figure 4.5), the Earth referential can be set to the geodetic positions (LLA) called "full GPS position" option or in local north-east-down (NED) with a geographic anchor. These two options and their particularities are described in more detail in the following subsection. If the GPS is not available/activated and the algorithm runs on "pseudo track" mode the trajectory is sent in NED frame with [0;0] as geographic anchor. Full GPS position and NED with anchor positions options can be used in "geo-track" mode. The LLA positions allow for some APIs to display the positions without any external action from the user and the NED positions allow sending more positions with better accuracy but require decoding. The compressed trajectory, the ethogram, and the data to be sent are then encoded, stored in the message payload and sent via the LoRaWan transmission. The process starts over after the data transmission.

Full GPS position: When the full position GPS mode is activated, all NED positions ready to be sent are converted into geodetic positions. When it is not activated we use the geodetic position of the beginning of the trajectory as geographic anchor and then send the compressed positions of the NED local coordinates. The NED positions take 8 bytes giving a horizontal accuracy of 0.1 m against 11 bytes for the geodetic positions with an accuracy of 11.1 m.

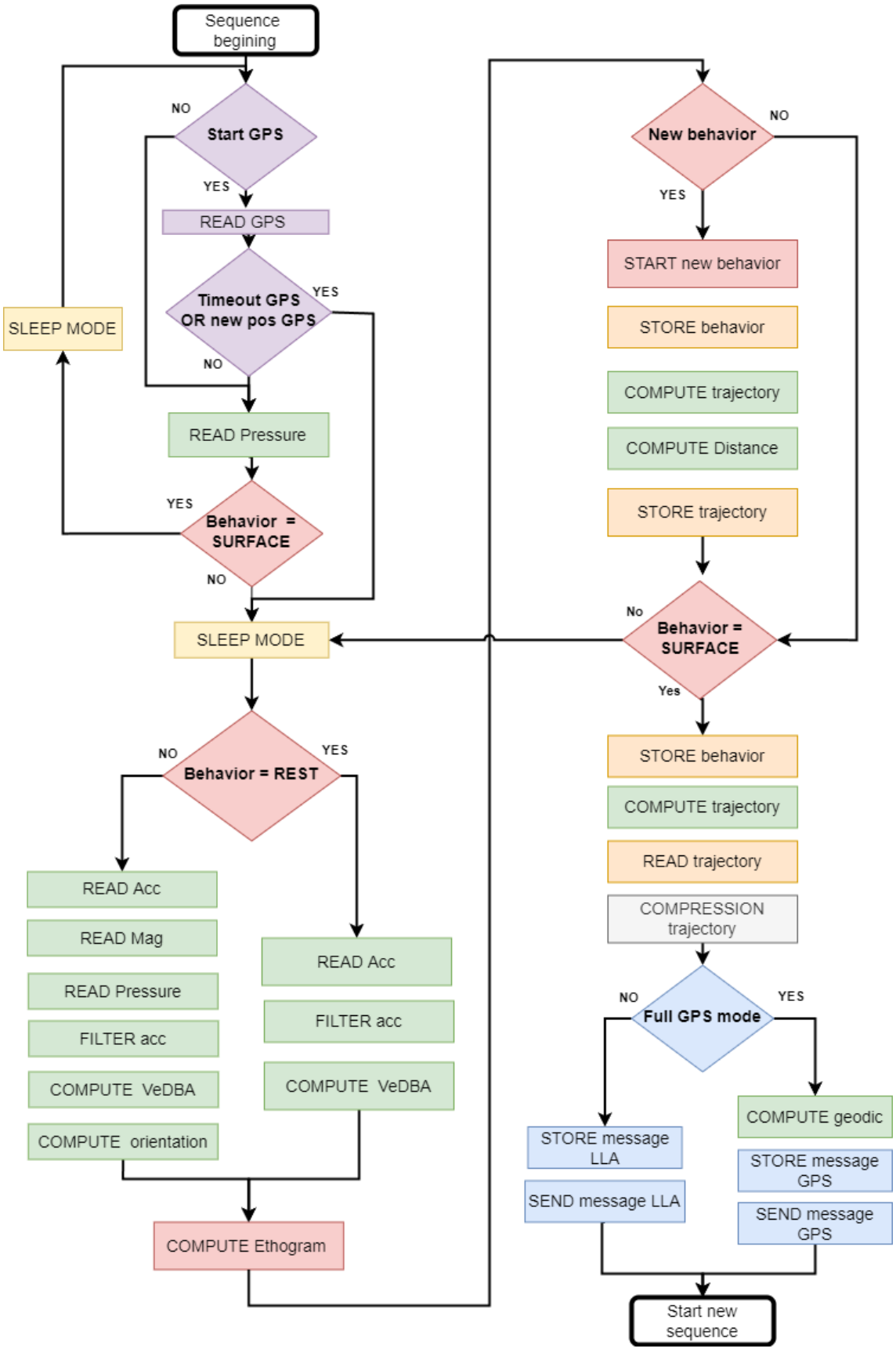


FIGURE 4.5: High-level diagram of the embedded algorithm

Trajectory compression

Among the most efficient and widely used polyline simplification algorithms, we select 2 that meet our criteria: DP (30), and N-Points (28). At this step, VVW is excluded because according to preliminary tests, it is less accurate and requires more computing than DP. In some studies comparing compression algorithms, DP is considered as one of the fastest with good efficiency (28; 29). The N-Points algorithm is chosen because of its simplicity and the fact that it requires almost no computation. Their respective parameters are given in Table 4.2. N is the dive's number of samples.

Algorithm	Time complexity	Memory complexity	Size bounded	Error bounded
Douglas Peucker	$O(Nlog(N))$	$O(N)$	No	Yes
N-Points	$O(1)$	$O(N)$	Yes	No

TABLE 4.2: Tables with the characteristics of Douglas Peucker and N-Points algorithms. Here, DP is not size bounded and in the function of its Error variable ϵ it can return less position than asked. For N-Points, it is only dependent on the dive time and returns the number of position asked.

Douglas-Peucker: To understand the method of DP algorithm we present an example in the Figure 4.6. We use the algorithm in 3D. The algorithm uses a recursive method called "divide and conquer". It is defined for a tolerance input parameter: ϵ. It starts by searching for the farthest point of the line formed by the starting point (P_{start}) and the arrival point of the trajectory (P_{end}). If the distance between these two points is greater than epsilon, the orthogonal projection of the middle of the straight line between P_{start} and P_{end} on the trajectory ($P1$) is kept (Figure 4.6 A). The algorithm is then applied recursively between P_{start} and $P1$ and $P1$ and P_{end}. It is applied until for all the portions of the trajectories the most distant points are lower than ϵ (Figure 4.6 E).

We estimate the maximum number of points possible to send per message and DP algorithm is stopped when we have a sufficient number of points. We define the parameter epsilon by trials and errors and set it for the final algorithm. This greatly speeds up the start of the algorithm.

N-Points: The N-Points method cuts the trajectory with a fixed time step (28). For the number of possible positions to be sent per message and the duration of the dive we determine in ΔT to sample the horizontal trajectory. This method has the advantage of being extremely simple to implement and needs a low computing.

Memory management

As mentioned in the hardware description our microcontroller has only a RAM of 20 kB and an EEPROM is 19 kB. EEPROM stores the program and the libraries and the RAM the variables in the program. A flash of 8 MB is also available. Our goal

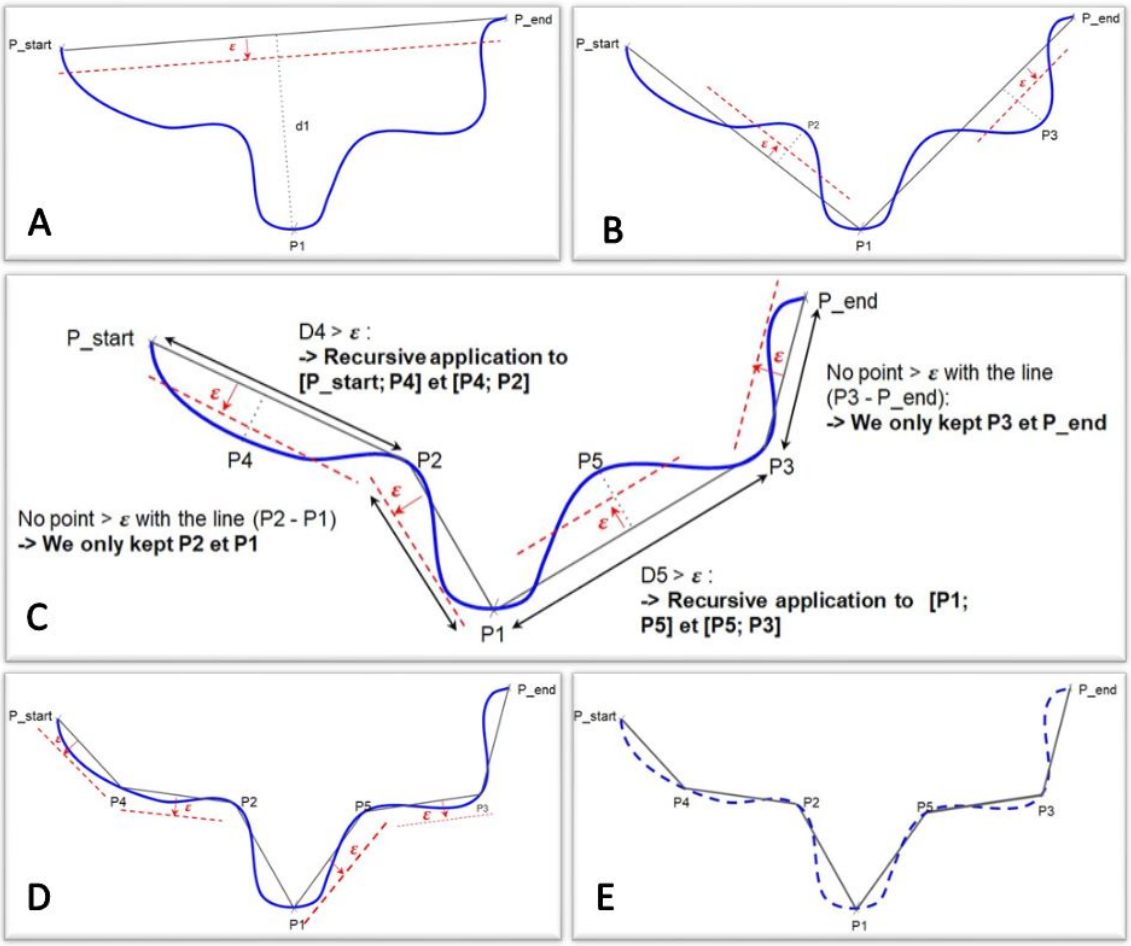


FIGURE 4.6: Step diagram of DP Algorithm. A) Search for the farthest point from the line (P_{start} - P_{end}) and comparison of the maximum distance to threshold ϵ input parameter. B) $D1 > \epsilon$ $P1$ point is kept, recursive application on [P_{start}; $P1$] and [$P1$; P_{end}]. We kept $P2$ and $P3$. C) Recursive application between the points [P_{start}; $P2$], [$P2$; $P1$], [$P1$; $P3$] and [$P3$; P_{end}]. We kept $P4$ and $P5$. D) Recursive application between the remaining intermediate points. All distances are less than ϵ so no kept point. E) Final approximation with P_{start}, $P4$, $P2$, $P1$, $P5$, $P3$ and P_{end}.

is to avoid storing data in the flash memory to reduce computing time and power consumption. Therefore variables without floating point are favored. The float variables generate more complex calculations and thus a computation time largely higher than the integers. Moreover, for a 30 min dive at 1 Hz, the displacement in X, Y, Z, requires 21600 bytes which is more than the space available in the RAM.

A simple and commonly used method is to convert the float to an integer by multiplying it by a constant according to the desired precision. In the case of a trajectory from a 30 min dive at 0.5 m/s, the distance traveled could go from -900 to 900 m. By multiplying by 10, we can have from -9000 dm to 9000 dm (decimeter) which can be stored in int16_t. We will then have a precision of 0.1 m for the positions of the animal. This technique allows reducing the memory usage for the variables which are stored in large arrays. Behavior information is stored in intermediate arrays. These arrays are overwritten and reused for each new dive. We store the start time, the behavior and the depth at the beginning of the behavior. The heading is stored in intermediate arrays as uint16_t by multiplying the heading by 10000. Its value in radian is between 0 and $2pi = 6.2832$ or 62832 after conversion.

For variables that are not in arrays and updated at each step, some of them need floats depending on the precision needed or the operations to be performed with them. The variables updated at each step useful for the description of our algorithm are Pitch (float), Roll (float), VeDBA (float), and depth slope (int16_t). The script includes many other variables for the different parts of the algorithm such as: GPS, RTC, etc...The memory calculation is simple and is done by the compiler during compilation. Dive up to 1800 s can be stored without using the Flash memory. It needs 70% of EEPROM and 89% of RAM. After this duration, it needs to apply specific methods discussed in Section 4.4.1

Data transmission

LoRa communication The communications of the tag is done with LoRa (Long Range) signals (25; 26). It is a method of communication being part of the Low-Power Wide Area Networks (LPWANs). It allows the sending of messages between 5 and 255 bytes of payload over distances of several kilometers with low consumption. This technology is part of the Internet of Things (IoT). To communicate, the system needs a receiving station called Gateway (GW) connected to the internet which relays the messages. (25) describes LoRa as 2 distinct layers with "a physical layer using the Chirp Spread Spectrum (CSS) radio modulation technique" and "a MAC layer protocol (LoRaWAN)". The detailed definition of the protocol is out of the scope of this article and is described in the literature (25; 26).

For our case study with surfacing marine animals which stay close to the shore, LoRa transmission has several advantages:

- Fast message sending in a few ms without the need to be connected to the receiving station
- Long range. This depends on several parameters (GW height, antenna height, sea conditions) but for this configuration, our tests give us over 10 km.
- Low-power consumption, e.g. for our configuration 30 mA during a 650 ms transmission for a 222-byte message

- Free transmissions (depending on the network)
- Large GW network is available. In France, several operators have deployed large LoRa networks that cover almost all the territory. However, it is necessary to pay a subscription that can be expensive (2€ / object). Scientists can also install their own networks at specific locations, but this requires buying and maintaining GW.

Several configurations of the module for LoRa transmissions are possible. This configuration will play on different parameters that define the size of the available payload, the range, the duration of the transmission and the power of the transmission. The configurations and some associated characteristics are available in the official documentation (https://www.thethingsnetwork.org/docs/lorawan/regional-parameters/). The lower the spreading factor, the lower the time on air (TOA) and the higher the maximum payload. However, the range of transmission will be reduced. A longer TOA will result in higher power consumption. In the same way, the transmission power influences the power consumption. The use of the network is also limited by transmission regulations such as European Telecommunications Standards Institute (ETSI) and courtesy rules as 'TTN Fair Access Policy', which ensures good practices so that the network is not saturated. For our study, we choose the SF8 with a transmission power of +14 dBm (maximum in Europe). The effective payload available is 222 bytes for a TOA of 655.9 ms. This configuration does not give the maximum range, nevertheless, the advantage of our application is to be in an open and clear environment favorable to the propagation of radio waves. In preliminary tests in the Gogendeau et al. (38), this configuration allows transmissions >10 km. In our tests, we could not reach the maximum transmission range at sea for this configuration. It is also strongly affected by the height of the receiving stations.

Payload description In LoRaWan message, the payload is the part which the user writes the information to send. In our case, we have 222 bytes available. To simplify the coding and the use of the data by external applications, we use the CayenneLPP formalism with its library developed for Arduino (https://github.com/ElectronicCats/CayenneLPP)

The payload of the proposed program in full GPS position mode (a) and without (b) is described in Figure 4.7. Each data is composed of Data channel (1 byte), Data IPSO type (1 byte), Data payload (variable number of bytes). This formalism is imposed by Cayenne to facilitate data decoding. In (a), the first 15 bytes of the message are composed of the dive number, the distance traveled, the duration of the dive and its tortuosity. For (b), the initial GPS position (11 bytes) is added to the fixed payload. The remaining 185 bytes (or 175 for the case (b)) are decomposed in different ways depending on the size of the ethogram. Each behavior is encoded in 8 bytes in the payload. They are composed in addition to the 2 identification bytes of a number corresponding to the behavior, the start time and its start depth. The number of bytes remaining then determines the number of positions that can be sent. Each GPS position requires 11 bytes and the NED positions require 8 bytes. Without full GPS mode activated, for a 30-min dive with 4 different behaviors ($4 \times 8 = 32$ bytes, the global variables, the ethogram and the first geodetic position

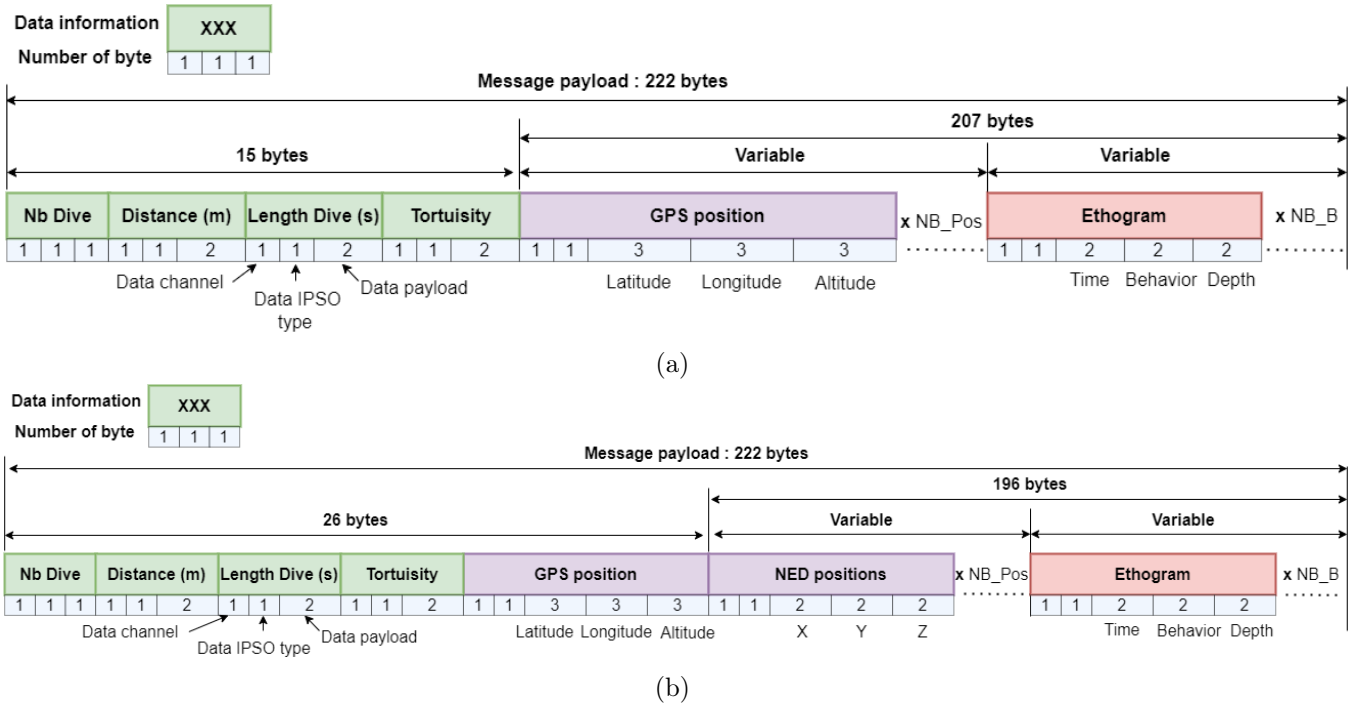


FIGURE 4.7: Payload of the message sent by the bio-transmetter. (a) is the payload with full GPS position option. In green, it's the fixed payload with the dive information (15 bytes in total). (b) payload without full GPS position option, the anchor GPS position is added to the fixed payload (26 bytes in total). For both payload, the remaining bytes are separate between the position and the ethogram. One behavior needs 8 bytes. In (a) the GPS positions are sent (11 bytes each), and in (b) NED positions are sent (8 bytes each)

(geographic anchor) (26 bytes) is encoded in 66 bytes. For a 222 bytes frame, 16 geodetic positions (11 bytes per position) and 20 NED positions (8 bytes per position) can be sent (+1 counting the geographic anchor). This gives about one position every 85 s for the NED positions in addition to the other data.

Geolocated and behavioral corrections of received messages

Dead-reckoning induces estimation errors growing over time. The correction of these estimations is applied in post-processing once messages are received. To correct the estimated trajectories, geolocated positions from the GPS are required. In the presented algorithm, if the tag gets a GPS position at the surface, the trajectory of the following dive is calculated from this starting point. The difference between the last position of the previous dive and the first position of the new dive is the error induced by the DR also called "drift" (39). In most studies, the drift is considered linear with time and gives a vector of error for each position. We apply this vector to the whole trajectory so that the last point matches the first point of the new trajectory received. To do so, we use the fit_{tracks} function proposed in animals tool kit[5]. If no GPS positions is acquire at the surface, the first position is set to the last

[5] http://www.animaltags.org/doku.php?id=tagwiki:tools:processing

estimated position of the previous dive.

The use of the behavior allows us to perform a correction on the trajectory for each behavior. In our case, the correction is not performed on the *REST* behaviors. A new vector of error per measurement is then calculated without considering the number of measurements where the turtle is in *REST*. The correction is then applied to the whole trajectory, excluding the REST phases. This avoids the addition of an important offset in the correction of the trajectory.

4.2.3 Experimentation with field data

Trajectories compression

To compare the compressed trajectory data and the actual trajectory, we analyze in post-processing the trajectory data of a green turtle (*Chelonia Mydas*) studied in Chapter 3. We reconstruct the trajectory with the $F1_E$ function of the chapter. This function, defined to be embedded, uses the Dead-Reckoning at 1 Hz with an estimated speed according to the ethogram and orientation calculated with the SAAM algorithm. We compare the shape of the 3D reference trajectory with the one simplified by the Douglas-Peucker algorithm and the N-Points point algorithm. In our analysis, we differentiate the trajectories with REST phases to better understand the compression results. In the same way, we classify the dives by duration scale. The dives are classified into two different classes, short ($< 150s$) and long ($>= 150s$). To analyze the trajectories, classical methods using the distance between points such as the 2D horizontal RMS (2DRMS) (40) are not possible. Compression gives a non-regular sampling rate and an inconsistent temporal scale. There are several methods to compare trajectories (29; 41). The methods available in the literature to compare trajectories are Hausdorff, Discrete Fréchet, SSPD. In Makris et al. (29), Discrete Fréchet and SSPD exhibit the same results for boat trajectories. From preliminary tests, Hausdorff and Discrete Fréchet give almost identical results with the same orders of magnitude. For the following, we have arbitrarily chosen to use the Hausdorff distance method. The objective is to embed the compression algorithms on a microcontroller. The resource requirements and the computing time are important parameters that can influence the choice of algorithm. For the different trajectories, we calculate the execution time of the algorithms for comparison purposes using a computer CPU (Processor: Intel(R) Core i7-8700 CPU @ 3.20HGz / RAM 32GO). The raw data and our analysis scripts are available on the GitHub[6].

Power consumption

For the power consumption measurements, we use the N6705B power analyzer[7]. The measurements of timings and consumption are made on the "Turtle tracker" board with the algorithm developed in this study and are available on the GitHub.[8].

[6] https://github.com/pierregoge/Turtle-tracker-tag/tree/main/Analysis/compression

[7] https://www.keysight.com/us/en/assets/7018-01824/product-fact-sheets/5989-8615.pdf

[8] https://github.com/pierregoge/Turtle-tracker-tag/tree/main/Analysis/consumption

The consumption formula are the following:

$$C_m = (\%R \times C_{mr} + \%A \times C_{ma}) + C_{mp} + C_{ml} + C_{mg} \tag{4.1}$$

with C_m the average consumption of the developed algorithm on the tag. C_{mr} and C_{ma} are the average consumption of the tag during sensor sampling on rest and active mode. $\%R$ and $\%A$ are the percentage of time associated to the modes. C_{mp} is the average consumption for the processing (trajectory and compression), C_{ml} the LoRa transmission module, and C_{mg} the GPS module. Consumption are in mA.

We calculate C_{mr} and C_{ma} directly with the average function of the power analyzer as the consumption is periodic with a sampling frequency of 1 Hz. For the other average consumption, we use the following :

$$C_{mp} = \frac{(T_{traj} + T_{comp})}{N \times T_{samp}} \cdot (C_a - C_s) \tag{4.2}$$

with C_{mp} the average consumption for the processing which is not periodic (trajectory and compression). C_a and C_s are instantaneous consumption for running and sleeping CPU in mA. T_{traj} is the computing time to process all the trajectory and to make the compression. N is the number of sample of the trajectory. T_{samp} is the sampling time of the algorithm. Time are in ms.

$$C_{mg} = \frac{T_g}{T_{sg} \times T_{samp}} \cdot (C_g - C_s) \tag{4.3}$$

with C_{mg} is the average consumption of the GPS. C_g and C_s are instantaneous consumption for GPS ON and sleeping CPU in mA. T_g is the time ON of the GPS. T_{sg} is the time between two GPS sampling. T_{samp} is the sampling time of the algorithm. Time are in ms.

$$C_{ml} = \frac{T_l}{T_{sl} \times T_{samp}} \cdot (C_l - C_s) \tag{4.4}$$

with C_{ml} is the average consumption for LoRa transmission. C_l and C_s are instantaneous consumption for transmission and microcontroller in sleep mode in mA. T_l is the computing time to send a message. T_{sl} is the time between message transmissions. T_{samp} is the sampling time of the algorithm. Time is in ms.

Some consumptions are associated with functions that use the ethogram defined in the Chapter 3 for a green turtle.

Data transmission and experimental design

The study areas are l'Ermitage-Les-Bains lagoon (-21.071860°N, 55.220403°E), and Saint Leu. These areas are chosen because juvenile green turtles (*Chelonia mydas*) and hawksbill turtles (*Eretmochelys imbricata* are present there. It is a very shallow area (max 1 m) that turtles choose for food and shelter. These tests do not require being on wild animals. All the features we need to test can be simulated. For ethical reasons, we prefer to equip a swimmer with a tag. The bio-telemeter is fixed under a paddleboard. During surfacing simulations, the swimmer puts the tag at the surface of the water for a defined time. The swimming speed in *SWIM* mode is set to 0.57 m/s. This is the average speed of the swimmer during previous tests with the same

configuration (See Chapter 2. We simulate short dives (<900 s) with a swimming sequence interspersed with *REST* and *SURFACE* behaviors. For this shallow zone, the analysis of the depth has little interest. The surfacing behavior is between 1 s and 60 s to test the fast transmission of messages for the short ones and the quality of the GPS acquisition for the longer surfacing.

First, we test the full GPS mode to analyze the data that can be received on Cayenne myDevice web API. Then we deactivate the full GPS mode and analyze the data received on an online database (*InfluxDB* through an API) and in post-processing with a custom script. This mode allows us to receive a GPS anchor position and then NED positions. When we deactivate the full GPS mode, we are able to send more positions with a better accuracy but it requires a processing step by the user. For the first test, the messages are sent to The Thing Network (TTN) and our *InfluxDB* database via one of our receiving station. For the second test, the messages are sent on the Orange private network (requiring a subscription) and transmit to our database.

All the raw data and analyzed script are provided on the GitHub[9].

4.3 Results

4.3.1 Accuracy of the compression and computing time

When sorting the dives, we separate them into 3 groups. 13 dives are short (< 150 s), 17 are long (>= 150 s), and 4 dives include *REST* phases. The latter are also the longest dives and are between 967 s and 1848 s. For comparison, in the group of long dives, the longest duration is 837 s.

	DP	N-Points	DP	N-Points
Dive (mean time (s))	Mean HD (m)	Mean HD (m)	Mean compute Time (ms)	Mean compute Time (ms)
Short (93)	0.168	0.304	0.251	0.1006
Long (390)	1.133	2.456	0.282	0.125
Rest (1258)	2.317	10.182	0.332	0.123

TABLE 4.3: Result of the Hausdorff Distance (HD) to estimate the accuracy of the compressed trajectory and the computing time for two compression algorithms: Douglas Peucker and N-Point algorithms. Dives are sorted by time classes, with short (<150 s) and long dive (>150 s). The last class includes dives for which the turtle has resting time longer than 5 minutes. We give the average dive duration for each class.

For short trajectories (<150 s), the Hausdorff distance for DP algorithm is 44% lower than using the N-points algorithm. The relative difference is however small (0.136 m difference between the DP and the N-points algorithms) (Table 4.3). The execution time is circa 50% higher for the DP algorithm. The difference in accuracy

[9] https://github.com/pierregoge/Turtle-tracker-tag/tree/main/Analysis/post_processing/trajectories_computing

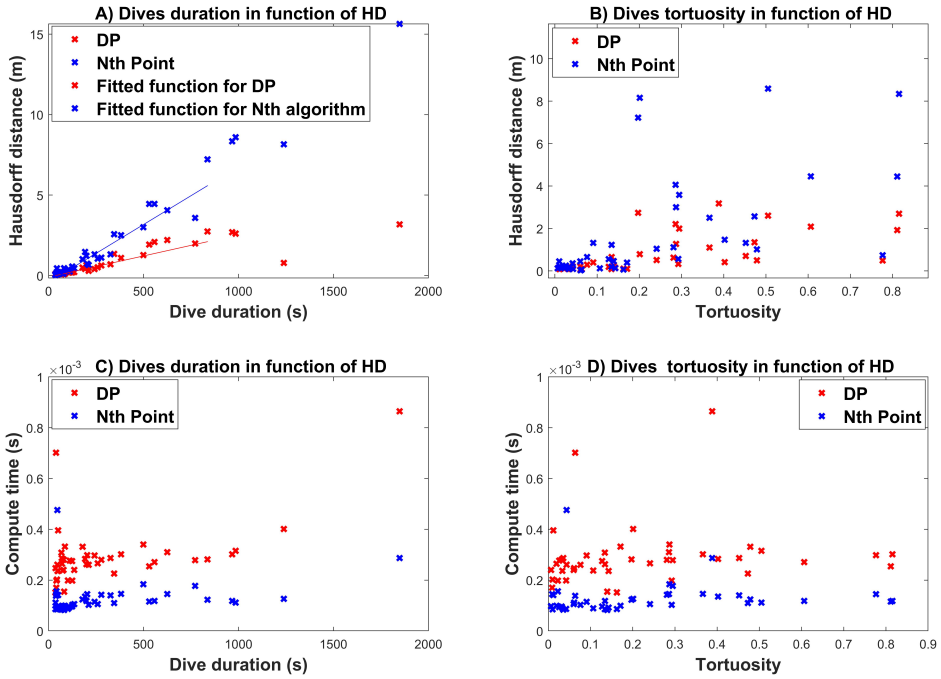


FIGURE 4.8: Hausdorff distance and computing time as a function of dive durations and tortuosity. A-B) Hausdorff distances as a function of the dive duration and tortuosity for the two different compression algorithms (DP and N-points). The fitted curves (linear regression) do not take into account the dives with rest phases (duration $>$ 900 s). C-D) Computing time as a function of the dive duration and tortuosity for the two different compression algorithms (DP and N-points).

is higher for long trajectories. The Hausdorff distance is 1.133 m for DP and 2.45 m for N-Points. The execution time increases by 55% for DP compared to the N-points algorithm. The most significant difference in terms of accuracy is for dives with rest. The DP algorithm has a Hausdorff distance of 2.31 m while it is 10.18 m with the N-points algorithm. The DP algorithm increases computing time by around 63% for the dives which include REST.

Apart from a few outliers, the computing time of N-Points seems relatively stable according (Table 4.3 and Figure 4.8 inset C). We denote a slight increase in the computing time as a function of the dive duration for DP. Moreover, computation time is consistently higher time for the DP algorithm. This computing time is consistent with the complexity of the algorithms. In Figure 4.8 inset B) and D), there is no relationship between the Hausdorff distance or the computing time and the tortuosity for the two different compression algorithms.

Examples of horizontal trajectories and dive profiles for the different dive classes illustrate the differences between the compression results for the DP and N-points algorithms (Figure 4.9 and 4.10). For each class of dives, we present one with low tortuosity (left column) and high tortuosity (right column). The number of points estimated is displayed because the DP method depends on the epsilon coefficient

which influences the number of points in the output. The ϵ is fixed and for some trajectories, the compressed trajectory is defined by fewer positions than the number of positions that can be sent. The message is thus smaller and faster to send.

A long resting phase ($>$ 300 s) can be observed for dive 55 (black circle in Figure 4.9). In contrast to dive 55, the dive 51 has several resting phases punctuated by movements difficult to interpret with the dive profile shown in Figure 4.10.

4.3.2 Power consumption estimation

As an illustration of the previous calculations, we arbitrarily chose dive times of 15 minutes. These are close to the typical behavior of juvenile turtles in the Reunion lagoon. For this duration, we use the N-Points algorithm. The percentages of time for each different behavior correspond to those described in chapter 3, i.e. 25% for REST and 75% SWIM. One message per dive is sent and the GPS is turned on for 30 s every 2 h. The GPS schedule is fixed to avoid losing the GPS ephemeris/calendar and switching to a cold start mode which requires more time to acquire the positions.

Table 4.4 gives the timings calculated on the board for the different parts of the algorithm necessary to calculate the total average consumption.

	Active	Rest	LoRa	Processing	GPS	Sleep
C_m (mA)	0.30	0.073	0.028	0.0004	0.116	0.02

TABLE 4.4: Power consumption of the tag for the different electronic parts and algorithm computing. Columns are the different functions or algorithm parts defined. C_m is the average power consumption

According to the Formula 4.1 - 4.4, the average consumption is equal to around 0.39 mA. The IOT tag is equipped with a 1300 mAh battery which gives us a theoretical lifetime equals to 138.8 days. The period during which we continuously receive the compressed trajectory, the ethogram, the distance traveled, the tortuosity, and the dive times, every time the turtle reaches the surface to breathe. With a message loss rate of 0%, this would give 13333 receive messages over the battery lifetime.

4.3.3 Transmission of the trajectories

The first test lasts 30 min and took place in Ermitage-Les-Bains lagoon. We send 4 messages in Full GPS mode. Figure 4.11 corresponds to the received messages displayed on the Cayenne MyDevice API. Geodetic positions, dive number, tortuosity, distance traveled, and dive duration are displayed. Cayenne does not offer a pre-coded option to directly display the ethogram. Apart from changing the variable names, it requires no knowledge or effort from the user to display the other variables. The gaps in the trajectory correspond to the error between the last position estimated by the algorithm and the GPS position of the next sequence Figure 4.11. In our post-processing algorithm, we can correct this difference, but the Cayenne WebApi only allows to display the raw data.

The second test took place in la Saline Les Bains lagoon for 20 min (Figure 4.12). We switch OFF the full GPS mode and activate sending NED position with 1

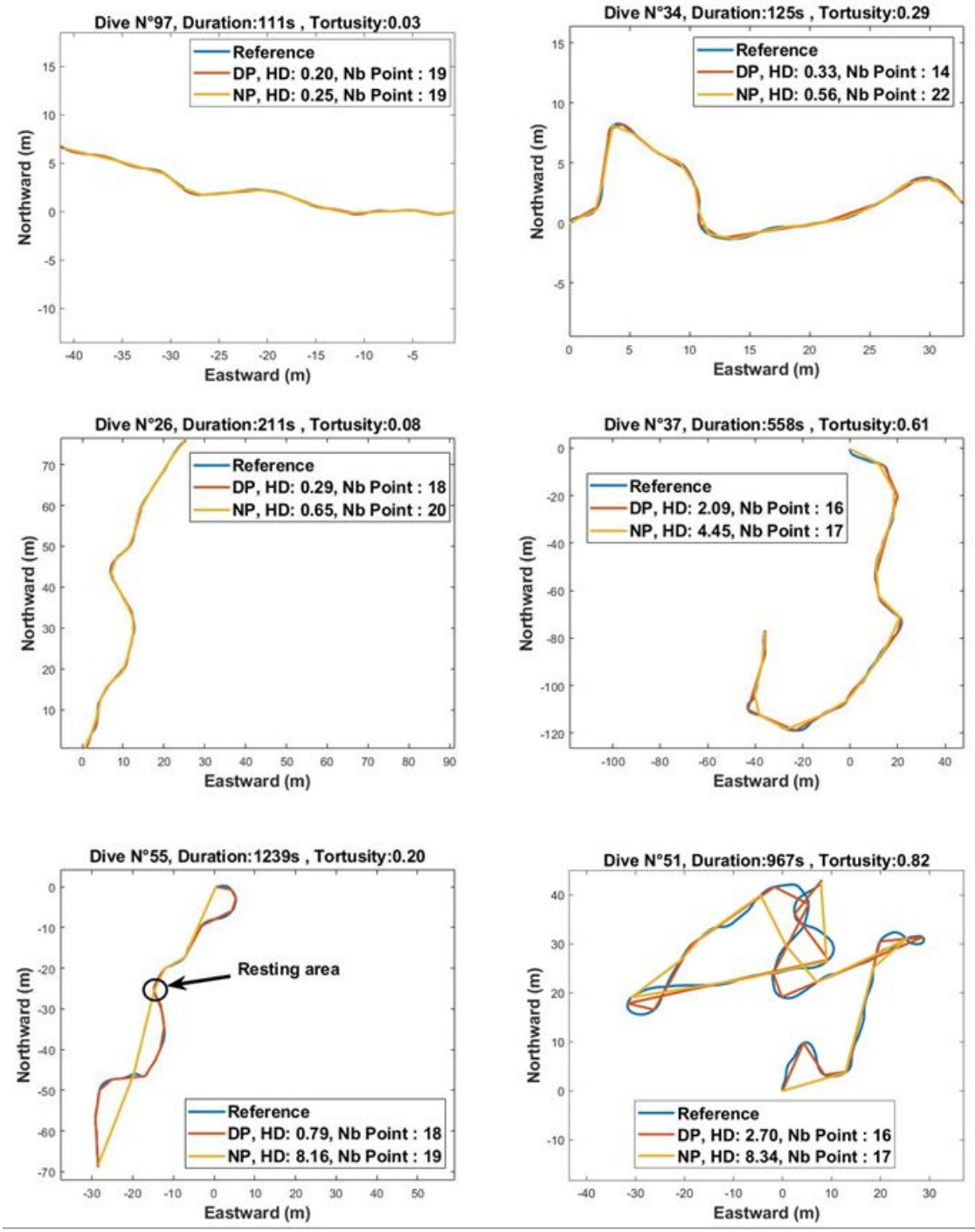


FIGURE 4.9: Horizontal trajectory for six different dives. We display the reference and compressed trajectories with Douglas-Peucker and N-Points. The first row corresponds to the trajectories for short dives, the second row, long dive, and the third row, dives with *RESt* phases. The left column is dives with low tortuosity and the right column is for dives with high tortuosity. Dive 55 is composed of long resting phases, whereas dive 51 has several resting phases punctuated by displacement.

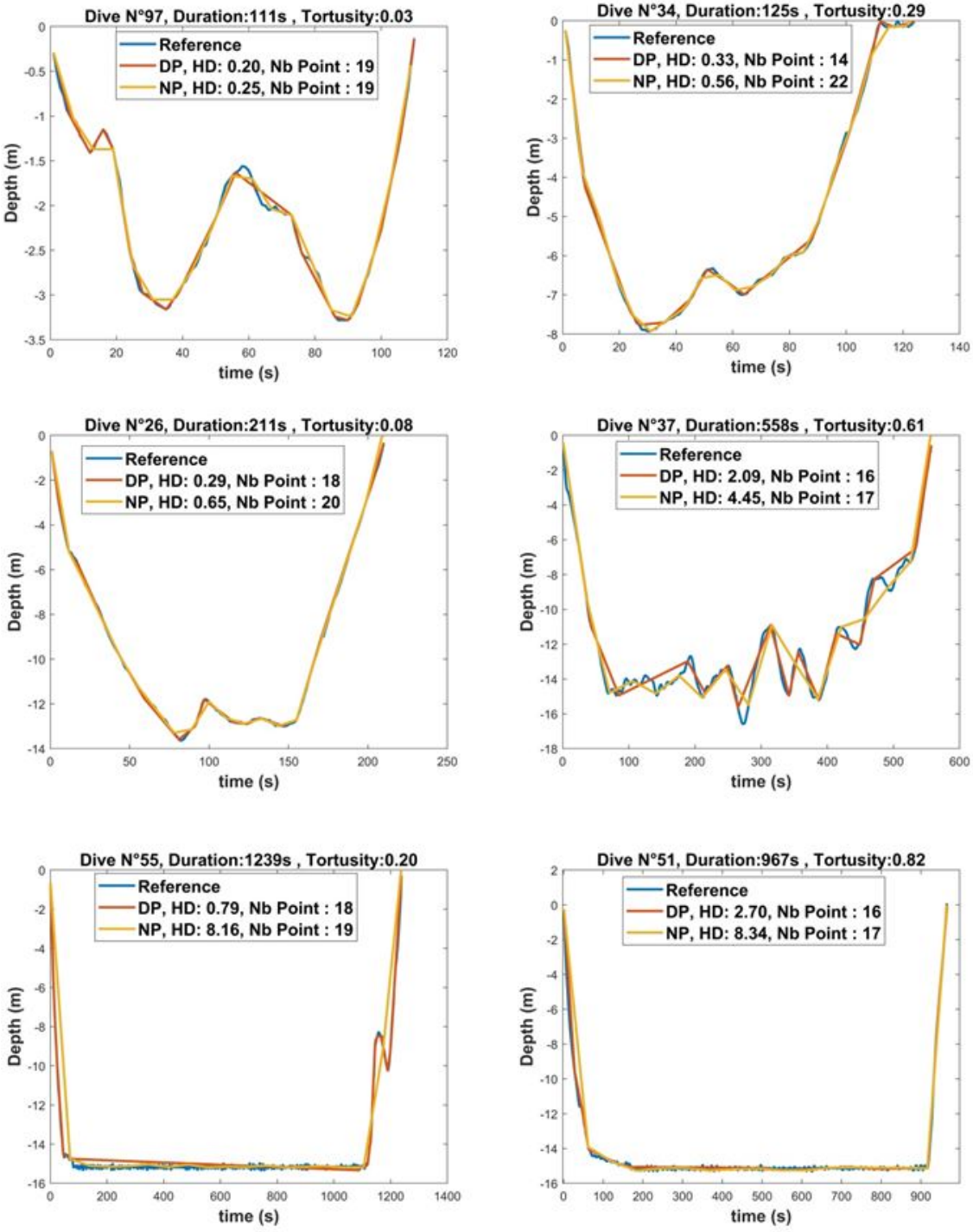


FIGURE 4.10: Dive profile for six different dives. The reference and compressed trajectories from the Douglas-Peucker (red line) and N-Points (yellow line) algorithms are represented on top of the reference dive profile (blue line). The first row corresponds to the trajectories for short dives, the second row, long dive, and the third row, dives with *REST* phases. The left column is dives with low tortuosity and the right column is for dives with high tortuosity. Dive 55 is composed of long resting phases, whereas dive 51 has several resting phases punctuated by displacement.

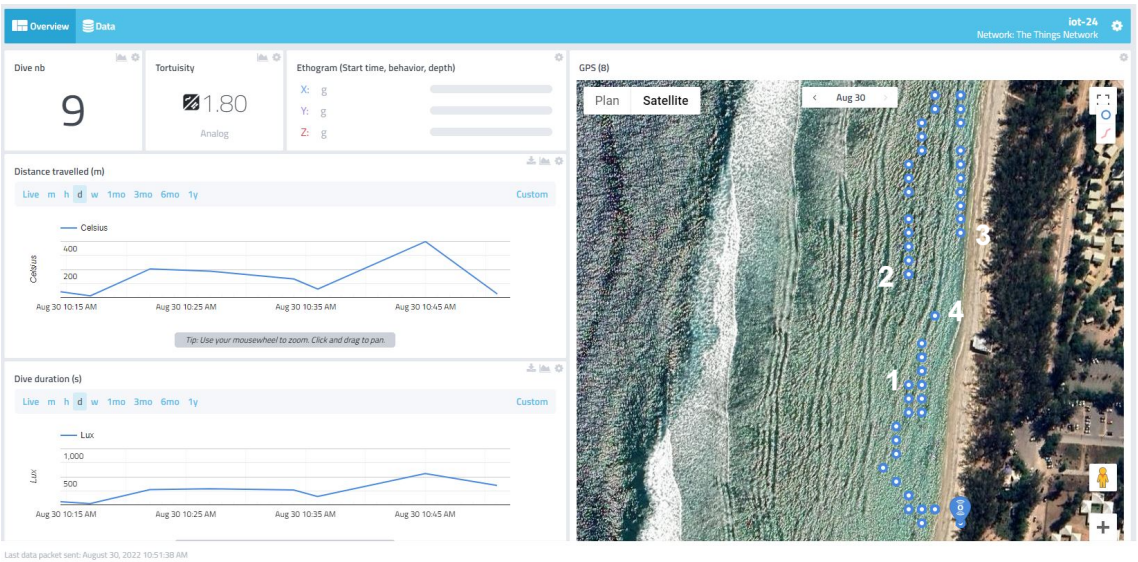


FIGURE 4.11: Example of full GPS data received and displayed on the Mydevice Cayenne API. The gaps in the trajectory correspond to the error between the last position estimated by the algorithm and the GPS position of the next sequence. It happens between 1:2 and 3:4. The test took place in l'Ermitage-Les-Bains lagoon (-21.071860 S, 55.220403 E) with a swimmer simulating a turtle.

anchored geodetic location per message. The custom API from the online database *InfluxDB* gives more flexibility in the display of raw data. The same data are displayed except that it is possible to display the NED positions, and the ethogram. For the NED positions they are displayed without post-processing and are therefore all starting from the point [0,0]. The reading and analysis of the ethogram is not straightforward without post-processing. Only the anchor geodetic positions, that are coded as GPS positions, are on the map.

For the last test, the full GPS mode remains switched OFF. The test took place in Saint-Leu lagoon for 20 min. The software used here is *Matlab*. The ground-truth trajectory acquired with a GPS system in parallel with bio-telemeter is displayed (Figure 4.13 A). The purple crosses are the estimated trajectories from NED position converted into geodetic positions. The purple line is the estimated trajectory corrected with the anchor positions of each sequence. The filled circles are the static phases (i.e. *REST* and *SURFACE*) with their sizes related to the duration and color to the behavior. The black circles represent the estimated measurement uncertainty of the positions. The uncertainty of a GPS position is equal to 10m.

On the insets B) and C), latitude and longitude are displayed as a function of time. This representation allows showing the ethogram in the background. Light purple is for *SWIM*, yellow for *SURFACE* and grey for *REST* behavior. We choose not to display the dive profile, which is of no interest for a surface test

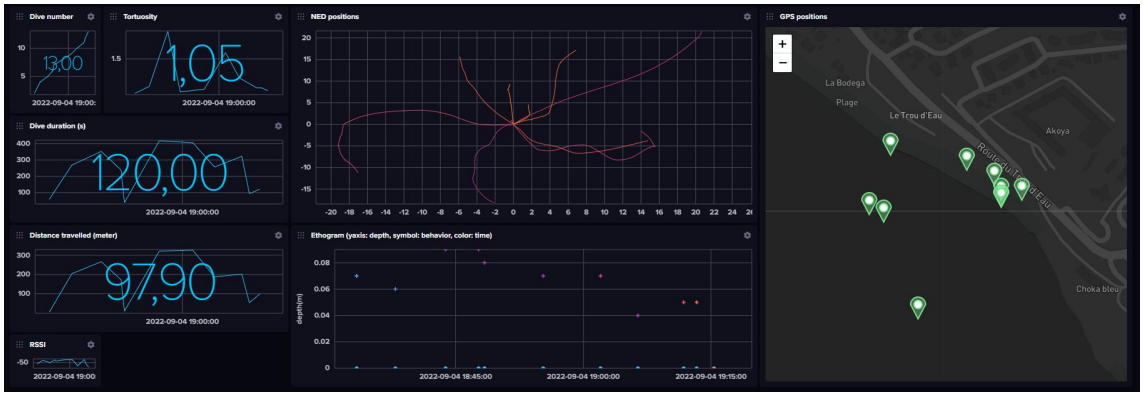


FIGURE 4.12: Example of data receive without full GPS mode in the *InfluxDB* API. The test took place in la Saline Les Bains lagoon (-21.071860°N, 55.220403°E) with a swimmer simulating a turtle

4.4 Discussion

The onboard calculation of trajectories and their transmission for marine animals can bring new knowledge to scientists. For some animals where the recapture is complex or even impossible, this would give access to these new types of data for deployments of several months. The use of the ethogram allows the improvement of several points. It reduces consumption and improves trajectory reconstruction. For studies using bio-logger where the recovery of the tags is possible, the calculation of the trajectory and the ethogram can be used to optimize the consumption and storage of raw data to have longer deployments.

The algorithm proposed on the tag can send for each dive: its id number, the distance traveled, the tortuosity, the ethogram (behavior, start time and depth), the GPS position at the beginning of the trajectory and between 16 and 20 3D positions compressed with DP or N-Point (GPS or NED) giving a faithful representation of the trajectory. The code is available as open-source and thanks to the Arduino programming language, it can be easily modified to suit the scientist needs. This algorithm has a consumption of 0.49 mA with GPS and transmission which correspond to deployment on the animal of 110 days. The ethogram, proposed in Chapter 3 and adapted for these tests, is simple and has the advantage of being adaptable to deduce any type of behavior according to researchers' questions.

In this section, some points on memory usage, trajectory compression, power consumption, and data transmission are discussed.

4.4.1 Memory usage

By saving directly position to tables in RAM or without using flash memory, the method allows the acquisition of dives up to 1800 s. For the acquisition of longer dives, the memory is saturated and thus the way the trajectories are stored has to be changed and adapted for the compression methods.

For both compression algorithms, a simple method is possible. It consists in lowering the frequency of acquisition of the positions according to the available memory and the duration of the dive. Once the trajectory is longer than a threshold value, the size of the trajectory is halved by downsampling the tables where the

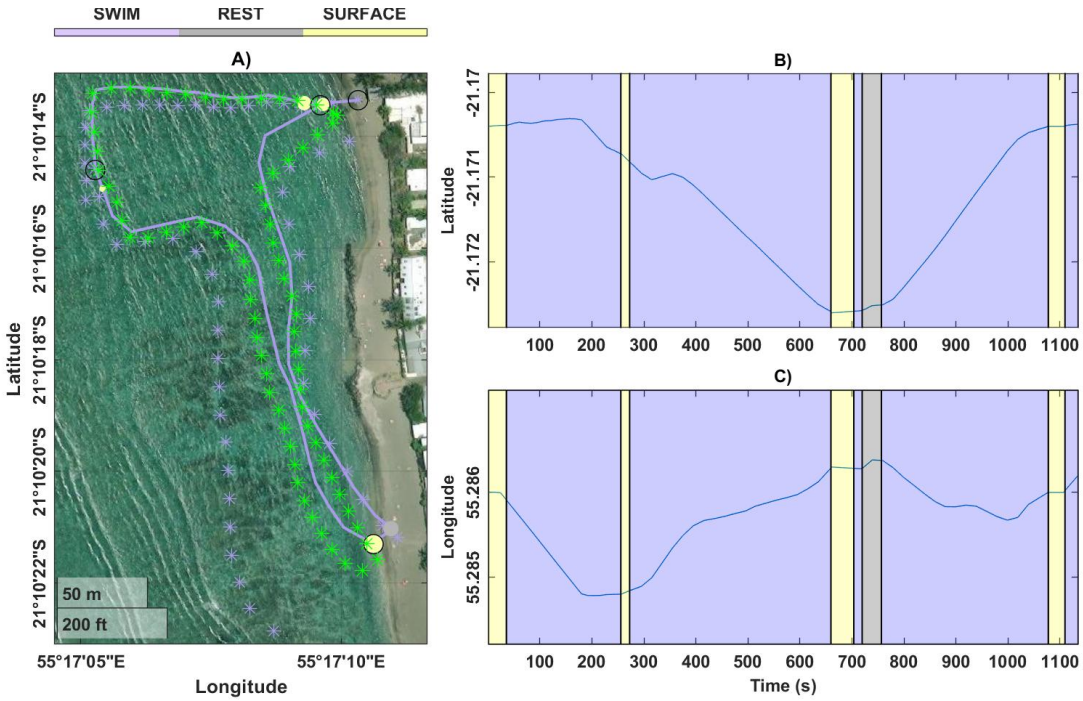


FIGURE 4.13: Example of data receive without full GPS mode after post-processing. The test took place in la Saline Les Bains lagoon (-21.071860°N, 55.220403°E) with a swimmer simulating turtle. The raw trajectory as we receive it is with the purple cross. The corrected trajectory is the purple line. The green crosses are the ground-truth trajectory from an external GPS.The filled circles are the static phases with sizes related to their durations and color for the behavior. The black circles represent the estimated measurement uncertainty of the GPS positions

positions are stored. Sampling continues at half the array with the new data storage frequency.

4.4.2 Compression

On Figure 4.8, we notice a cluster of points for short dives <150 s and without *REST*. In this case, N-Point method presents a Hausdorff distance very close to DP with a lower computing time. For these short dives, the use of the N-Point method is recommended. For longer dives, this trade-off is questionable and will depend on the resources and needs of the scientists. For our reconstructed trajectories using the DR we want a Hausdorff distance <5 m after compression. According to Figure 4.8, we use DP for trajectories longer than 900 s or those with *REST* phases. Indeed in Figure 4.9 for the dive 55, the N-Point method is not adapted to the *REST* phase because it cuts the trajectory independently of the movements. There is no obvious relationship between the tortuosity and the Hausdorff distance. In some special cases, even on long dives, DP brings little gain in precision compared to N-Point. On Figure 4.14 for dive 14 (203 s), a strong tortuosity of 0.78 can be noted while the Hausdorff distance is close between compression algorithms. This trajectory is composed of a straight trajectory followed by a regular circular trajectory. In

this case, N-Points is quite efficient. A specific indicator using the information on the type of shape of the trajectory would allow planning more efficiently the use of N-Points for some long dives and to save processing time.

To further improve compression, other solutions exist. For example, Google proposes a method of encoding GPS positions which allows to gain x3.5 on the size of the encoded positions (42). The library is available for Arduino (43).

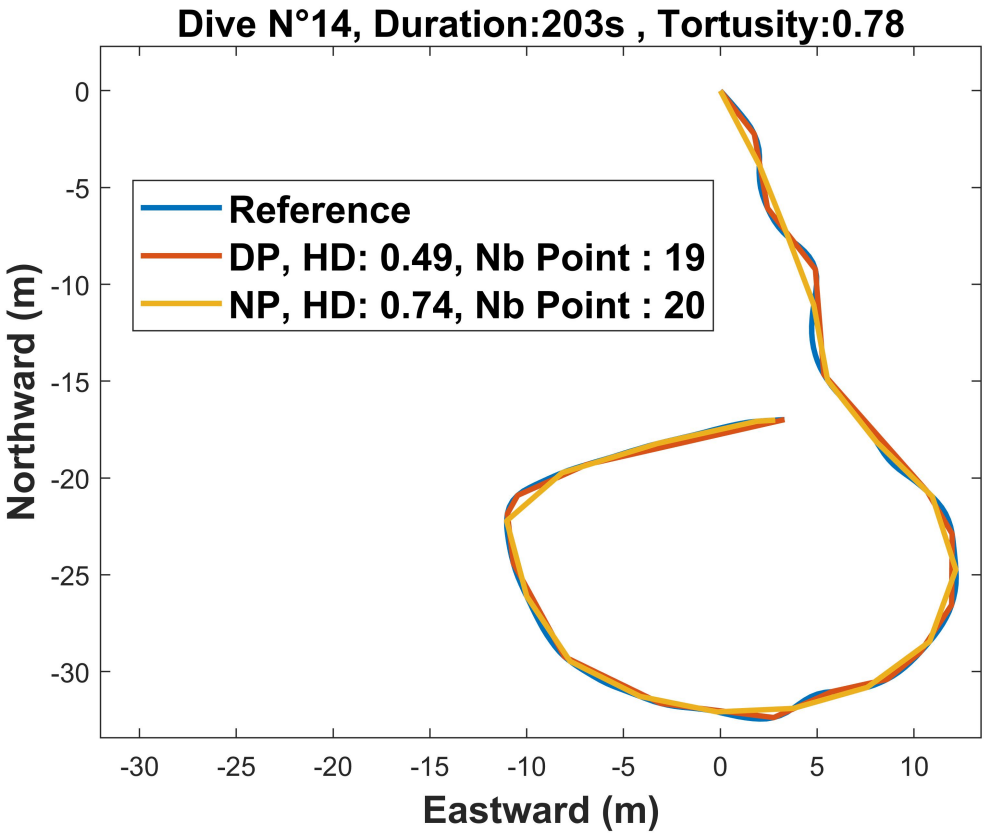


FIGURE 4.14: Dive 14: For this 203*s* trajectory with high tortuosity, the regular patterns (Straight line + circle) show a little difference in Hausdorff distance between the N-Points and DP algorithms.

4.4.3 Consumption

According to the measurements, our solution offers a lifetime of approximately 138 days. This time is consistent with the physical life span of tags used during previous tagging operations.

Actually, the power consumption is lower than the value presented in this chapter. As we have little knowledge of the percentage of time the turtle spends on the *surface*, this case is excluded. During the duration of this behavior, only the pressure sensor is ON and it should give a power consumption of around 0.10 mA rather than 0.30 mA.

In the study of the energy consumption (Table 4.4), we notice that the main power costs are the data acquisition and the GPS. The software improvement of the acquisition has already been optimized thanks to the behavior shown in chapter 4.

For the GPS, following its scheduling it can become an even more important source of consumption. With 60 s ON every 20 min it gives 1,5 mA average consumption. This blind scheduling is not very efficient. In some studies for terrestrial animals, researchers have proposed algorithms to manage the GPS according to the animal activity (44; 45). In our case, the trajectory information can be used rather than the activity. Indeed, there is no interest in starting the GPS if the animal has not moved. For marine turtles, the surfacing can be short and turning on the sensor during this time is useless because it does not allow a fixed position. The optimization of the GPS scheduling will require the study of surfacing events.

In both cases, hardware improvement is also possible, but it requires the development of a new electronic board or an extension. For example, for the accelerometer, ST has just released a new sensor (46). This chip includes a core for machine learning, decision trees that simplify computing with reduced sensor consumption. For GPS, the GPS fastloc (47; 48) has been available for a few years. Its implementation could limit the acquisition time of the GPS.

4.4.4 Data transmission

This method offers new opportunities for the type of data that can be received from bio-transmitters in a marine environment. The ethogram used in the example illustrating the article is very simple, but the method described in chapter 4 is easily adaptable.

The code proposes message frames that can be used and analyzed for different types of users. Using LoRa receiving stations connected to The Thing Network (TTN), messages encoded with CayenneLPP can be directly decoded and displayed in the myDevice web API (Figure 4.11) without external action or coding skills. For users using a private network or with their own networks, the messages can be stored in databases and displayed by developing their own API. In our case, we use a mix of private and our own networks. Data are stored in *influxDB* and its graphical interface (4.12). However, this requires the user to decode the messages and create his own API. The user can also retrieve the messages from a database and study or display them with statistical software. We use Matlab (4.13)) but there are many other solutions that we do not present in this article to retrieve messages and display them. These solutions must be studied and developed according to the user's skills.

CayenneLPP allows simple decoding but is not the most optimized method to encode GPS/NED positions. If the user chooses not to use CayenneLPP to encode his messages, he can save up to several bytes per message by removing the headers of each data. For a dive with 5 behaviors, it is possible to send 37 positions (Figure 4.15). The user has however to develop a payload decoder.

The number of messages received will depend on the deployment area and the placement of the LoRa receiving stations. In some of our deployments in the lagoon of Europa Island (-22.349708°N, 40.378160°E) located in the Indian Ocean, surfacing are more frequent than every 15 min and for some areas, important message losses can be observed (e.g. in dense mangrove). For these areas and during long surfacing, we could plan a repeated sending of some dives to reduce the message loss rate.

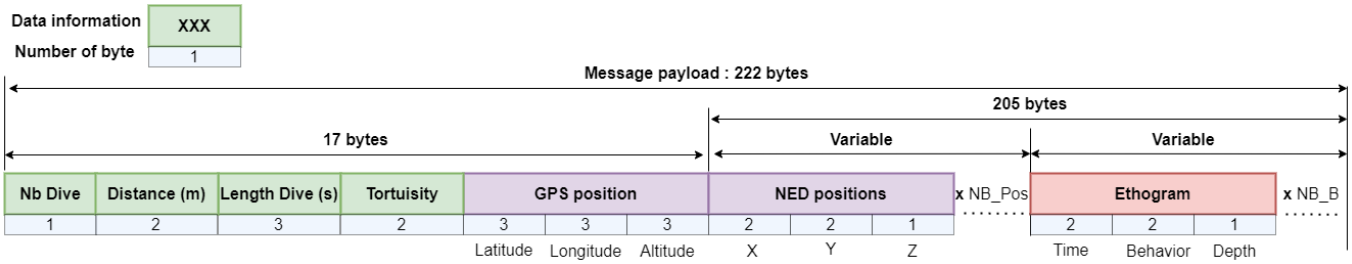


FIGURE 4.15: Payload of the message sent by the bio-telemeter without Cayenne formatting. In green, the fixed payload with the dive information and anchor GPS position (17 bytes in total). The last bytes remaining are separate between the position and the ethogram. One NED position needs 5 bytes and one behavior needs 5 bytes.

Acknowledgements

This work is supported by the "IOT" project funded by FEDER INTERREG V and Prefet de La Réunion: grant #20181306-0018039 and the Contrat de Convergence et de Transformation de la Préfecture de La Réunion. We are grateful to Quentin Ponzo and Virgil Taillandier from LIRMM for their support, building a tag for the field tests and the electrical consumption acquisition.

Data accessibility

Data is processed using MATLAB (2020b) and customized scripts. Embedded programs are coded and compiled via the Arduino IDE in C++. All the data, scripts, codes are available on GitHub[10].

Bibliography

[1] Y. L. Bras, J. Jouma'a, and C. Guinet, "Three-dimensional space use during the bottom phase of southern elephant seal dives," *Movement Ecology*, vol. 5, no. 1, p. 18, 2017. [Online]. Available: https://doi.org/10.1186/s40462-017-0108-y

[2] K. Benoit-Bird, B. Battaile, C. Nordstrom, and A. Trites, "Foraging behavior of northern fur seals closely matches the hierarchical patch scales of prey," *Marine Ecology Progress Series*, vol. 479, pp. 283–302, Apr. 2013.

[3] P. J. Wensveen, L. Thomas, and P. J. O. Miller, "A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales," *Movement Ecology*, vol. 3, no. 1, p. 31, 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0061-6

[4] P. J. Wensveen, S. Isojunno, R. R. Hansen, A. M. von Benda-Beckmann, L. Kleivane, S. van IJsselmuide, F.-P. A. Lam, P. H. Kvadsheim, S. L. DeRuiter, C. Curé, T. Narazaki, P. L. Tyack, and P. J. O. Miller,

[10] https://github.com/pierregoge/Turtle-tracker-tag

"Northern bottlenose whales in a pristine environment respond strongly to close and distant navy sonar signals," *Proceedings of the Royal Society B: Biological Sciences*, vol. 286, no. 1899, p. 20182592, 2019. [Online]. Available: https://royalsocietypublishing.org/doi/10.1098/rspb.2018.2592

[5] K. Aoki, M. Amano, K. Mori, A. Kourogi, T. Kubodera, and N. Miyazaki, "Active hunting by deep-diving sperm whales: 3D dive profiles and maneuvers during bursts of speed," *Marine Ecology Progress Series*, vol. 444, pp. 289–301, 2012.

[6] T. Narazaki, K. Sato, K. J. Abernathy, G. J. Marshall, and N. Miyazaki, "Sea turtles compensate deflection of heading at the sea surface during directional travel," *Journal of Experimental Biology*, vol. 212, no. 24, pp. 4019–4026, 2009. [Online]. Available: https://doi.org/10.1242/jeb.034637

[7] S. Andrzejaczek, A. C. Gleiss, K. O. Lear, C. B. Pattiaratchi, T. K. Chapple, and M. G. Meekan, "Biologging Tags Reveal Links Between Fine-Scale Horizontal and Vertical Movement Behaviors in Tiger Sharks (Galeocerdo cuvier)," *Frontiers in Marine Science*, vol. 6, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fmars.2019.00229

[8] J. Korpela, H. Suzuki, S. Matsumoto, Y. Mizutani, M. Samejima, T. Maekawa, J. Nakai, and K. Yoda, "Machine learning enables improved runtime and precision for bio-loggers on seabirds," *Communications Biology*, vol. 3, no. 1, pp. 1–9, 2020. [Online]. Available: https://www.nature.com/articles/s42003-020-01356-8

[9] L. R. Brewster, J. J. Dale, T. L. Guttridge, S. H. Gruber, A. C. Hansell, M. Elliott, I. G. Cowx, N. M. Whitney, and A. C. Gleiss, "Development and application of a machine learning algorithm for classification of elasmobranch behaviour from accelerometry data," *Marine Biology*, vol. 165, no. 4, p. 62, 2018. [Online]. Available: https://doi.org/10.1007/s00227-018-3318-y

[10] H. Yu, J. Deng, T. Leen, G. Li, and M. Klaassen, "Continuous on-board behaviour classification using accelerometry – a case study with a new GPS-3G-Bluetooth system in Pacific Black Ducks," *Methods in Ecology and Evolution*, vol. n/a, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13878

[11] T. A. Wild, M. Wikelski, S. Tyndel, G. Alarcón-Nieto, B. C. Klump, L. M. Aplin, M. Meboldt, and H. J. Williams, "Internet on animals: Wi-Fi-enabled devices provide a solution for big data transmission in biologging," *Methods in Ecology and Evolution*, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13798

[12] L. Jeantet, V. Planas-Bielsa, S. Benhamou, S. Geiger, J. Martin, F. Siegwalt, P. Lelong, J. Gresser, D. Etienne, G. Hiélard, A. Arque, S. Regis, N. Lecerf, C. Frouin, A. Benhalilou, C. Murgale, T. Maillet, L. Andreani, G. Campistron, H. Delvaux, C. Guyon, S. Richard, F. Lefebvre, N. Aubert, C. Habold, Y. le Maho, and D. Chevallier, "Behavioural

inference from signal processing using animal-borne multi-sensor loggers: a novel solution to extend the knowledge of sea turtle ecology," *Royal Society Open Science*, vol. 7, no. 5, p. 200139, 2020. [Online]. Available: https://royalsocietypublishing.org/doi/10.1098/rsos.200139

[13] R. Wilson, E. Shepard, and N. Liebsch, "Prying into the intimate details of animal lives: Use of a daily diary on animals," *Endangered Species Research*, vol. 4, pp. 123–137, 2008.

[14] M. Espinoza, T. J. Farrugia, D. M. Webber, F. Smith, and C. G. Lowe, "Testing a new acoustic telemetry technique to quantify long-term, fine-scale movements of aquatic animals," *Fisheries Research*, vol. 108, no. 2, pp. 364–371, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165783611000300

[15] J. Leander, J. Klaminder, M. Jonsson, T. Brodin, K. Leonardsson, and G. Hellström, "The old and the new: evaluating performance of acoustic telemetry systems in tracking migrating Atlantic salmon (Salmo salar) smolt and European eel (Anguilla anguilla) around hydropower facilities," *Canadian Journal of Fisheries and Aquatic Sciences*, 2019. [Online]. Available: https://cdnsciencepub.com/doi/abs/10.1139/cjfas-2019-0058

[16] "Wildlife monitoring: birds, terrestrial and marine animals." [Online]. Available: https://www.argos-system.org/applications-argos/wildlife-monitoring/

[17] X. Andre, B. Moreau, and S. Le Reste, "Argos-3 Satellite Communication System: Implementation on the Arvor Oceanographic Profiling Floats," *Journal Of Atmospheric And Oceanic Technology*, vol. 32, no. 10, pp. 1902–1914, 2015. [Online]. Available: https://archimer.ifremer.fr/doc/00284/39519/

[18] S. L. Cox, F. Orgeret, M. Gesta, C. Rodde, I. Heizer, H. Weimerskirch, and C. Guinet, "Processing of acceleration and dive data on-board satellite relay tags to investigate diving and foraging behaviour in free-ranging marine predators," *Methods in Ecology and Evolution*, vol. 9, no. 1, pp. 64–77, 2018. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12845

[19] "ARGOS (ARTIC R2) Satellite Communication Guide - learn.sparkfun.com." [Online]. Available: https://learn.sparkfun.com/tutorials/argos-artic-r2-satellite-communication-guide/all

[20] "Wildlife Computers Inc. - SPLASH10." [Online]. Available: https://wildlifecomputers.com/our-tags/splash-archiving-tags/splash10/

[21] T. Photopoulou, P. Lovell, M. A. Fedak, L. Thomas, and J. Matthiopoulos, "Efficient abstracting of dive profiles using a broken-stick model," *Methods in Ecology and Evolution*, vol. 6, no. 3, pp. 278–288, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12328

[22] B. Mcconnell, R. Beaton, E. Bryant, C. Hunter, P. Lovell, and A. Hall, "Phoning Home-a New Gsm Mobile Phone telemetry System to Collect

Mark-Recapture Data," *Marine Mammal Science*, vol. 20, no. 2, pp. 274–283, 2004. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1748-7692.2004.tb01156.x

[23] M. A. Cronin and B. J. McConnell, "SMS seal: A new technique to measure haul-out behaviour in marine vertebrates," *Journal of Experimental Marine Biology and Ecology*, vol. 1, no. 362, pp. 43–48, 2008. [Online]. Available: https://www.infona.pl//resource/bwmeta1.element.elsevier-96dde939-fde3-352a-a0cf-f7fb6600e1b7

[24] L. Kirkpatrick, I. Olivares, A. Massawe, C. Sabuni, H. Leirs, R. Berkvens, and M. Weyn, *ProxLogs: Miniaturised proximity loggers for monitoring association behaviour in small mammals*, 2021.

[25] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things," *Sensors*, vol. 16, no. 9, p. 1466, 2016. [Online]. Available: https://www.mdpi.com/1424-8220/16/9/1466

[26] B. Paul, "An Overview of LoRaWAN," *Wseas transactions on communications*, vol. 19, pp. 231–239, Jan. 2021.

[27] R. Fujdiak, P. Blazek, K. Mikhaylov, L. Malina, P. Mlynek, J. Misurec, and V. Blazek, "On Track of Sigfox Confidentiality with End-to-End Encryption," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, ser. ARES 2018. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1–6. [Online]. Available: https://doi.org/10.1145/3230833.3232805

[28] W. Shi and C. Cheung, "Performance Evaluation of Line Simplification Algorithms for Vector Generalization," *Cartographic Journal, The*, vol. 43, pp. 27–44, 2006.

[29] A. Makris, I. Kontopoulos, P. Alimisis, and K. Tserpes, "A Comparison of Trajectory Compression Algorithms Over AIS Data," *IEEE Access*, vol. 9, pp. 92 516–92 530, 2021, conference Name: IEEE Access.

[30] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, Dec. 1973. [Online]. Available: https://utpjournals.press/doi/10.3138/FM57-6770-U75U-7727

[31] M. Visvalingam and J. D. Whyatt, "Line generalisation by repeated elimination of points," *The Cartographic Journal*, vol. 30, no. 1, pp. 46–51, 1993. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1179/000870493786962263

[32] R. K and W. a. P. M, "Optimizing curve segmentation in computer graphics." *Int Comput Symp*, pp. 467–472, 1974. [Online]. Available: https://jglobal.jst.go.jp/en/detail?JGLOBAL_ID=201002064588396801

[33] H. Opheim, "Smoothing a digitized curve by data reduction methods," 1981. [Online]. Available: https://diglib.eg.org:443/xmlui/handle/10.2312/eg19811012

[34] M. Fedak, P. Lovell, B. McConnell, and C. Hunter, "Overcoming the Constraints of Long Range Radio Telemetry from Animals: Getting More Useful Data from Smaller Packages1," *Integrative and Comparative Biology*, vol. 42, no. 1, pp. 3–10, 2002. [Online]. Available: https://doi.org/10.1093/icb/42.1.3

[35] "IOT (2018-2021)." [Online]. Available: https://ocean-indien.ifremer.fr/Projets/Innovations-technologiques/pIOT-2018-2020-IOT-2018-2021/IOT-2018-2021

[36] L. Qasem, A. Cardew, A. Wilson, I. Griffiths, L. G. Halsey, E. L. C. Shepard, A. C. Gleiss, and R. Wilson, "Tri-axial dynamic acceleration as a proxy for animal energy expenditure; should we be summing values or calculating the vector?" *PloS One*, vol. 7, no. 2, p. e31187, 2012.

[37] J. Wu, Z. Zhou, H. Fourati, and Y. Cheng, "A Super Fast Attitude Determination Algorithm for Consumer-Level Accelerometer and Magnetometer," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 3, p. 375, 2018. [Online]. Available: https://hal.inria.fr/hal-01922922

[38] P. Gogendeau, N. M. Murad, S. Bernard, V. Kerzerho, L. Deknyff, and S. Bonhommeau, "Oversea Radio Measurements and Channel Characterization with LoRa Technology," in *2018 IEEE Radio and Antenna Days of the Indian Ocean (RADIO)*, 2018, pp. 1–2.

[39] R. P. Wilson, N. Liebsch, I. M. Davies, F. Quintana, H. Weimerskirch, S. Storch, K. Lucke, U. Siebert, S. Zankl, G. Müller, I. Zimmer, A. Scolaro, C. Campagna, J. Plötz, H. Bornemann, J. Teilmann, and C. R. McMahon, "All at sea with animal tracks; methodological and analytical solutions for the resolution of movement," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 54, no. 3, pp. 193–210, Feb. 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967064507000045

[40] J. Sanz Subirana, J. Juan Zornoza, and M. Hernández-Pajares, "Positioning Error - Navipedia," 2011. [Online]. Available: https://gssc.esa.int/navipedia/index.php/Positioning_Error

[41] A. Salarpour and H. Khotanlou, "Direction-based similarity measure to trajectory clustering," *IET Signal Processing*, vol. 13, no. 1, pp. 70–76, 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-spr.2018.5235

[42] E. RobotGrrl, "Polyline on Arduino for Low-Power Geolocation Devices," Jul. 2021. [Online]. Available: https://medium.com/@robotgrrl/polyline-on-arduino-for-low-power-geolocation-devices-13c97c9dd50c

[43] RobotGrrl, "Polyline Library," Jul. 2021, original-date: 2021-07-05T22:20:19Z. [Online]. Available: https://github.com/RobotGrrl/PolylineArduino

[44] D. Brown, S. LaPoint, R. Kays, W. Heidrich, F. Kümmeth, and M. Wikelski, "Accelerometer-Informed GPS Telemetry: Reducing the Trade-Off Between Resolution and Longevity," *Wildlife Society Bulletin*, vol. 36, Mar. 2012.

[45] M. Lichtenstein and G. Elkaim, "Efficient GPS Scheduling in Wildlife Tags using an Extended Kalman Filter-based Uncertainty Suppression Strategy," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, pp. 1472–1475, iSSN: 2153-3598.

[46] "ISM330IS - iNEMO inertial module: always-on 3D accelerometer and 3D gyroscope with embedded ISPU - intelligent sensor processing unit - STMicroelectronics." [Online]. Available: https://www.st.com/en/mems-and-sensors/ism330is.html

[47] E. Bryant, *2D Location Accuracy Statistics for Fastloc RCores Running Firmware Versions 2.2 & 2.3, Technical Report TR01*, 2007.

[48] A. M. Dujon, R. T. Lindstrom, and G. C. Hays, "The accuracy of Fastloc-GPS locations and implications for animal tracking," *Methods in Ecology and Evolution*, vol. 5, no. 11, pp. 1162–1169, 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12286

Conclusion

The main objective of this PhD work was to develop an embedded algorithm within a bio-transmitter to reconstruct underwater trajectories for wild marine animals. A combination of scientific challenges was to be overcome to be able to reach this objective. When starting my PhD work, the challenges such as the transmission and the development of an electronic board including the different sensors within a waterproof and shockproof casing were well advanced through the IOT project I participated in. In my view, the remaining challenges were two-fold: (*i*) develop a way to collect reference underwater trajectories from an animal to be able to compare and assess the quality of the algorithm for trajectory estimation, (*ii*) find the best trade-off between accuracy, computing resources, data and message size, and power consumption for the algorithm. Here are summarized and discussed the main outcomes for these two aspects.

Reference data for algorithm comparisons

Few studies have compared the results of 3D trajectory predictions to actual data for marine applications. The main originality of this work was to develop low-cost and reliable instruments to collect position ground-truth data over relatively long periods. The first two chapters describe these instruments and how their use enabled us to define the type of algorithms that meet the requirements for being embedded within an electronic tag.

These solutions allow acquiring underwater geolocated 3D positions at fine spatial scale ($\approx$ 1 m) every second. In parallel, an ASV (Autonomous Surface Vehicle) has been developed to collect data on the environment surrounding the trajectory. The ASV is equipped with a single-beam echosounder to perform bathymetry and a camera for photogrammetry. This ASV enables us to get a ground-truth trajectory which can be used to assess the accuracy of different algorithms which reconstruct underwater trajectories from bio-logger data.

This work can be used further to collect data on various animals and for bathymetry or photogrammetry surveys. Another project is currently deploying the ASV to map habitats and species distributions from AI identification using collected videos. As an open source/hardware project, I hope the reproduction and the improvement of the ASV will offer more and more scientific opportunities.

An adaptative algorithm using behavior to improve trajectory estimations and scientific knowlkedge

Trajectory reconstruction is possible thanks to Dead-Reckoning (DR) with information from different sensors to estimate orientation and speed. My contribution in this field is to define the software and hardware context for an embedded DR

algorithm with limited computational and storage resources. I was looking for the best trade-off between the accuracy of the estimation, the power consumption and the capacity to transfer the trajectory data. Over Chapters 3 and 4, I developed the different methodological steps to achieve this trade-off and illustrated them through field experiments. The main originality stems from the use of behavior to improve the trajectory estimations and provide an ethogram along the trajectory.

It is noteworthy I rely on one acquisition performed on a single animal. It would be necessary to repeat the experiment over longer periods and diversify the animal sizes and the studied species. However, this experiment using the ASV presented in Chapter 1 is rather complex to carry out and requires constraining means: divers, boat, ASV handling. The complexity of the test also comes from the recapture of the animal. It could be simplified with a release system for the acoustic system and logger to avoid the second capture which is used to recover the material. On the other hand, these data could be complemented with other tests in a controlled environment that are easier to perform. For example, the speed estimation during the swimming phases could be tested in a swimming corridor.

The estimation and transmission solution I propose in Chapter 4 is validated in controlled environments. The next step is to validate it during the next missions on wild animals and test if the received data and the lifetime correspond to the expectations.

In my view, this embedded and adaptive algorithm which can transfer 3D positions and behavior data for each dive over periods longer than 4 months is an interesting contribution to the field. As mentioned in the chapters, data and code sources are open and I hope they can be reused and further developed and improved by other scientists.

Perspectives for trajectory improvements and biological information

From this work, I identified three main improvement areas that could be developed over time with different levels of complexity. It includes (*i*) to display and contextualize the trajectories within the environment and the improvement of geolocation by developing (*ii*) a smart GPS scheduling, and (*iii*) a method to fit the estimated trajectory to environmental variables to constraints the range of possible trajectories.

Environmental context of trajectories

The main goal of all this work and the IOT project is a better understanding of the species to improve their conservation. This better understanding is achieved by strengthening the knowledge of their movements and behaviors.

To add biological information to the analyses, the trajectories can be contextualized with their environments. The most common method for geolocalized trajectories is to display them with satellite maps. In Chapter 4, the solutions presented with the Web-API give us an overview of what it is possible to do with this type of layer and the data we receive. However, scientists may have different environmental

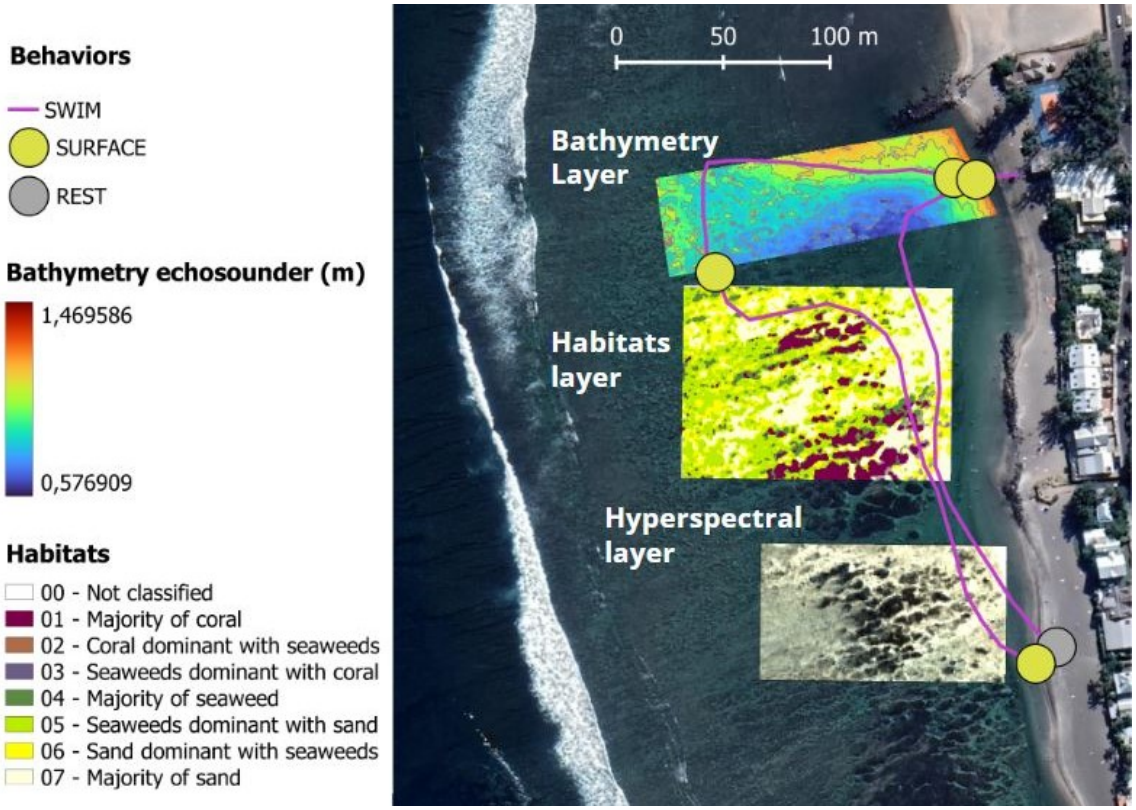


FIGURE 4.16: Example of a map with a trajectory and the behavior sent by the bio-telemeter. Several layers are displayed: single beam bathymetry, habitat mapping and hyperspectral imagery.

data layers to analyze and compare them with the trajectories. For example, the ASV describes in Chapter 1 offers the possibility of single beam bathymetry acquisition surveys. Other data types may be available depending on the geographical area with open access data or previous research projects. In the Reunion island area, in addition to the data acquired with the ASV, we have access to habitat mapping, lidar and hyperspectral imagery, coral vitality map, and various other types of mapping available on Sextant platform[11]

The display of these data layers for analysis is possible with different supports, but it is facilitated with Geographic Information System (GIS) software. They allow the overlaying of the different layers of the environment by various representations in 2D or 3D and associate the transmitted data of trajectory and behavior for a fine and comparative analysis.

To illustrate this, two examples are given. On Figure 4.16, with *QGIS* software, we display the trajectory of Chapter 4 in the lagoon of Saint-Leu received from the bio-telemeter. It is associated with the ethogram received, giving the *SWIM*, *REST* and *SURFACE* behaviors. In addition to the satellite imagery, three layers are proposed: a layer of bathymetry sampled with the ASV, an habitat layer and a hyperspectral imagery layer (available in open data on the Sextant platform).

The second example, in Figure 4.17, displays the reference trajectory acquired for the ASV. In the same way, with this trajectory, we describe the useful behaviors

[11] https://sextant.ifremer.fr/ocean-indien

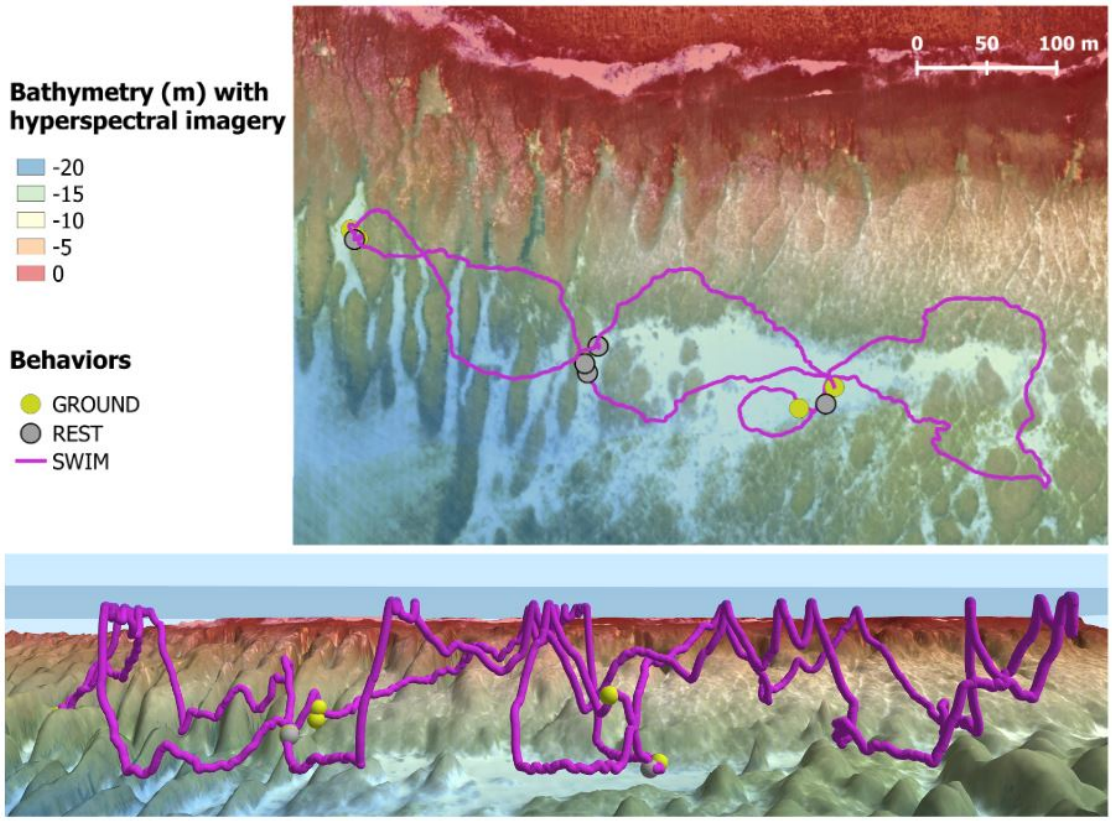


FIGURE 4.17: Example of a map with a trajectory and the behavior sampled with the ASV (A). The same data are shown in 3D (B). The layers displayed are single beam bathymetry associated with hyperspectral imagery of the area.

for the analysis. The *SWIM*, *REST* and *GROUND* behaviors are presented. For this deep area, it is represented in insets A) the bathymetry associated with the hyperspectral imagery. This layer gives a faithful representation of the bottom mapping of the area and facilitates the analysis of the trajectory and behaviors. We propose on the inset B) a 3D view to better represent space occupation for these same data.

Smart GPS planning

During the surface phases, the GNSS module is used to receive accurate geolocated positions (<10 m). In practice, this technique encounters limitations that complicate the acquisition of positions and the power consumption is too high so the GPS cannot be used continuously. The ephemeris is the detailed library of satellite orbit parameters. It is used to predict their position and accelerate the acquisition of positions. If it is up-to-date, the GPS is in "hot start" mode and the acquisition is fast (from a few seconds to about ten seconds depending on the configuration). However, the ephemeris must be updated every 4 hours. In a classical terrestrial operation, the GPS module automatically updates it every 30 min. For our deployment in marine environment, this is impossible because we depend on the dives and

the surface time of the animal. In recent years, some beacons have used fastloc-GPS systems that acquire millisecond snapshots of the signals for processing while the animal is underwater (1). Although this method is less dependent on the animal surfacing time, it is still energy consuming during position processing and optimized scheduling is also important.

If the GPS does not have its ephemeris up to date, the module goes into "cold start" and acquiring a position in addition to the ephemeris can take more than a minute. Moreover, the activation of the GPS during all the periods at the surface is impossible, because it is the most consuming source of the beacon. In chapter 4, the electrical consumption of the GPS is 30 mA on average against 0.3 mA for the algorithm and the sensors in the active phase. All these acquisition constraints together show us that smart scheduling of the GPS is crucial.

Improvements of the GPS schedule have been proposed for terrestrial use where the frequency of acquisitions is set according to an indicator of the animal activity calculated using the accelerometer (2). The more active the animal, the faster they consider it to be moving and the need to increase the acquisition frequency of GPS positions. (3) developed a method to estimate the uncertainty of the trajectory which is used to plan the activation of the GPS. This solution could be adapted for our study by decreasing the GPS activation time and thus the energy consumption.

With the new information available from the tag (trajectories and behaviors), it is possible to imagine other research directions to improve scheduling. We have identified three interconnected fields of study that would be interesting to investigate further: Do we need a new position? Is it useful to turn on the GPS? Do we have the energy resources to turn it on?

With the estimation of the trajectory, the distance traveled is calculated. We assume that even if the GPS acquisition conditions are optimal (long surfacing and available energy) the acquisition of a geolocated position is not necessary if the animal has moved little. For example, in our case study of marine turtles, during the night, the animal almost doesn't move and is most often in a resting phase punctuated by long surfacing. These phases do not require the correction of positions with the GPS and would save energy. It is then necessary to develop a function that conditions the activation of the GPS with the distance traveled, adaptable to the user needs.

The second research question is about the analysis of the surfacing to determine if they are long enough for an acquisition or the ephemeris update. Indeed, during our previous deployments, we notice a possible correlation between the behavior, the duration of the dives, and the time the turtle spends at the surface. In the same way, studying these data could allow the development of a function conditioning the activation of the GPS when the animal arrives at the surface according to its behavior. The goal is to reduce the activation of the GPS for surfacing if the system does not manage to have a position or if the ephemeris are up to date in order to avoid any unnecessary energy consumption.

The last point deals directly with the energy issue. In Chapter 4, we propose formulas to calculate precisely the consumption of the beacon. If the user defines the lifetime, the consumption of the algorithm is fixed, and the energy budget could be determined for the GPS scheduling (translatable in seconds of use). This information is very useful for the development of a scheduling algorithm. For example, it can

help to define a daily budget or to relax and reinforce some other previously defined activation rules.

Access to new types of live data by the trajectory algorithm then opens up possibilities for intelligent GPS position acquisition and consumption reduction.

Trajectory correction with bathymetry in post-processing

During the post-processing analysis of the estimated trajectories from Chapter 3 with the bathymetry data, several issues were identified. In some places, there is collisions between the seabed and the animal. In other cases, we found *REST* phases where the animal is not at the bottom. These two cases, visible using an accurate bathymetry, are physically impossible situations.

To solve these problems, terrestrial navigation uses map matching methods (4; 5). For example, when we are driving and the GPS gives an erroneous position in a building, it keeps the position on the right road because the situation is physically impossible. To our knowledge, these methods are not applied to correct the trajectories of marine animals. In this case, the map matching as it is used on land is difficult to apply directly because we have a movement in 3D and not constrained by the map as with the roads of a city.

In marine environment, the method can be applied using the bathymetry associated with the animal trajectory and its behavior. The erroneous positions are found by comparing the depth of the animal (considered accurate in the range of accuracy of the sensor) and the bathymetry. When the depth for *REST* and *GROUND* behaviors (where the animal is in contact with the bottom) is different from the depth of its position or when the depth of the animal is greater than the depth of the associated bathymetry. It faces physically impossible situations. In practice, this technique is limited by the accuracy of the pressure sensor and bathymetry. The accuracy of both measurements must be considered with safety coefficients to identify non-erroneous points for correction.

Once problematic points have been identified, the difficulty lies in the search patterns of the corrected points on the horizontal plane. Indeed, the ocean floor is not a linear slope and contains faults, reef walls, coral head. To restrict the search area, we can use the uncertainty of the position measurement and use the terrain level line of the sea bottom to search for the nearest possible depth. When the algorithm finds the new corrected points, the same correction method used for the GPS points including the animal behavior described in chapter4 can be applied.

In our first tests with the estimated trajectory data from Chapter 3 and the available bathymetry data, this method seems promising. It corrects part of the estimation error on the Eastward axis (perpendicular to the coastline) by identifying some collision points.

What's next?

The technologies associated with this research work are in constant evolution. In the field of communications, Semtech, the company that develops LoRa, and Lacuna Space will soon propose a satellite transmission solution[12]. We can then imagine systems with hybrid satellite - LoRa radio transmissions depending on where the

animal is located. Other satellite transmission solutions have been developed with, for example, the ICARUS project (6) developed to transmit bio-telemeter data to the ISS spatial station. The advantage of this solution compared to Argos (in addition to its low-power consumption) is the possibility of having a dedicated downlink per tag that would adapt its sampling or the use of sensor according to the information sent.

In the same way, as for transmissions, sensors are constantly evolving. Speed measurement remains an essential point, and the development of new sensors that are less subject to external disturbances could be a major advance in the DR. It is however important to consider the trade-off with the gain that new technologies bring, the constraints brought and the tests necessary for its integration.

Soon, live trajectories of thousands of marine animals, even tiny, even in abysses, will seem as easy to get as the image of the DART vessel impact 11 millions km away (Fig. 4.18).

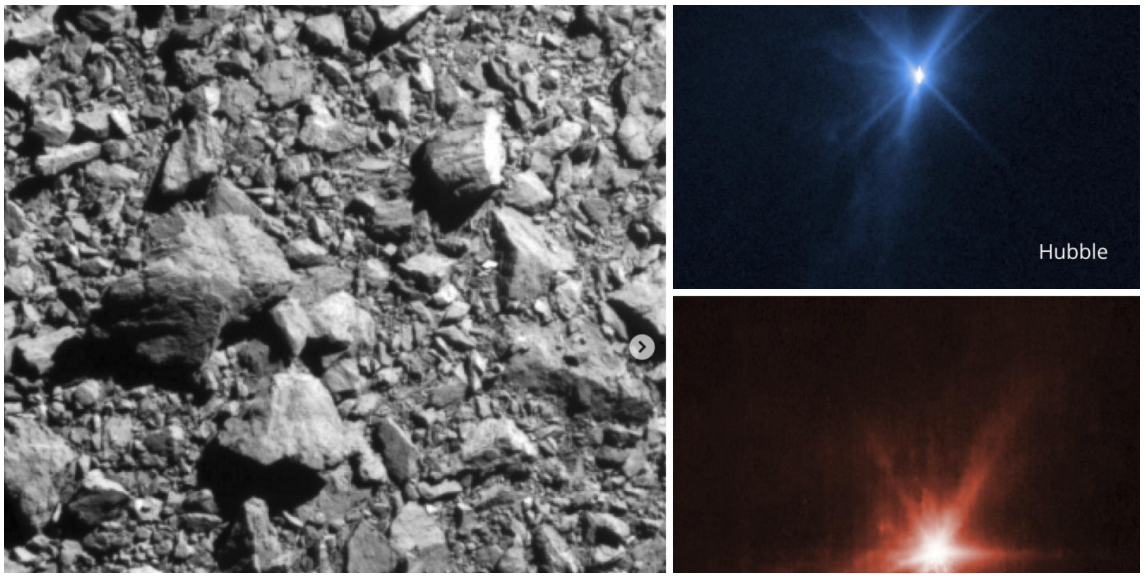


FIGURE 4.18: Crash of the DART vessel at the surface of Dimorphos on Oct. 1 2022. Actual picture on the left-hand side and Hubble and James Webb telescope images on the right-hand. ASI/NASA.

Bibliography

[1] A. M. Dujon, R. T. Lindstrom, and G. C. Hays, "The accuracy of Fastloc-GPS locations and implications for animal tracking," *Methods in Ecology and Evolution*, vol. 5, no. 11, pp. 1162–1169, 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12286

[2] D. D. Brown, S. LaPoint, R. Kays, W. Heidrich, F. Kümmeth, and M. Wikelski, "Accelerometer-informed GPS telemetry: Reducing the trade-off between resolution and longevity," *Wildlife Society Bulletin*, vol. 36, no. 1, pp. 139–146, 2012. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/wsb.111

[12] https://lacuna.space/about/

[3] M. Lichtenstein and G. Elkaim, "Efficient GPS Scheduling in Wildlife Tags using an Extended Kalman Filter-based Uncertainty Suppression Strategy," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr. 2020, pp. 1472–1475, iSSN: 2153-3598.

[4] M. Hashemi and H. A. Karimi, "A critical review of real-time map-matching algorithms: Current issues and future directions," *Computers, Environment and Urban Systems*, vol. 48, pp. 153–165, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0198971514000908

[5] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, *Map-matching for low-sampling-rate GPS trajectories*, Jan. 2009, pages: 361.

[6] M. Krondorf, S. Bittner, D. Plettemeier, A. Knopp, and M. Wikelski, "ICARUS—Very Low Power Satellite-Based IoT," *Sensors*, vol. 22, p. 6329, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/17/6329

List of publications

Journals

Published:

1. **P. Gogendeau**, S. Bonhommeau, H. Fourati, D De Oliveira, V. Taillandier, A. Goharzadeh and S. Bernard. "Dead-Reckoning Configurations Analysis for Marine Turtle Context in a Controlled Environment" in IEEE Sensors Journal, vol. 22, no. 12, pp. 12298-12306, 15 June15, 2022, doi: 10.1109/JSEN.2022.3170414.

In Preparation:

1. **P. Gogendeau**, S. Bonhommeau, H. Fourati, M. Julien, M. Contini, T. Chevrier, A-E. Nieblas and S. Bernard. 2022. "Plancha ASV : Affordable open-source vehicle allowing autonomous acoustic tracking, bathymetric and photogrammetric surveys" in IEEE Journal of Oceanic Engineering

2. **P. Gogendeau**, S. Bonhommeau, H. Fourati and S. Bernard. 2022. "Behavioral based Dead-Reckoning for embedded application on marine animals" in Method in Ecology and Evolution

3. **P. Gogendeau**, S. Bonhommeau, H. Fourati, M. Julien and S. Bernard. 2022. "Transmission of trajectories and ethogram by LoRaWAN for long-term deployment with embedded calculation" in Method in Ecology and Evolution

International conference

1. **P. Gogendeau**, S. Bonhommeau, H. Fourati, M. Renovell, F Soulier, D. Oliveira, S. Bernard, 2021. Dead-reckoning for marine animals: trade-offs between estimation accuracy and power consumption. 7th Bio-Logging Symposium, Hawaii

Appendix A

Chapter 2 : Supplementary information

Here we develop technical aspect about the specification validation and the power consumption estimates.

A.1 Specification validation

The section will be split into two parts. The first shows the navigation capabilities given by the autopilot for the board, the accuracy of the track and its limitations. The second part gives the power consumption for the different configurations.

To illustrate this part we choose a deployment that we made in Reunion Island during calibration tests. During these tests we were able to try different speeds and loads to compare them.

A.2 Autonomous navigation

Precise trajectory is an important factor to complete accurate bathymetric and photogrammetric surveys. Indeed, for these surveys, we need an overlap of the depth sample or photos to be sure that the mission area is well covered. In function of the area, we create the survey mission with a known spacing between transects. To set this spacing right, we must know the board navigation capabilities in terms of transect line following accuracy and drift.

To analyze the mission and validate the autopilot calibration, we use Mission Planner analyzer. At the end of the autopilot calibration[1], we evaluate the trajectory accuracy by comparing the programmed route to the realized one.

Fig. A.1 displays the result of the calibration of the January 11 2022 test in the Saint Gilles marina (-21.055440°N, 55.222527°E). The spacing between transects is 1 m and cruise speed was set to 0.8 m/s. The wind was blowing at 8 kt and there was no significant wave or current. After the U-turn on the first 5 m of every transect line, the board goes beyond half the spacing between the transects and, after that, it remains under a 50 cm distance from the path (Fig. A.1). The deviation on the U-turns can be bypassed with the elongation of the transects beyond the zone of interest to make sure these turns are performed outside the zone. In general, it is

[1] https://ardupilot.org/rover/docs/rover-first-drive.html

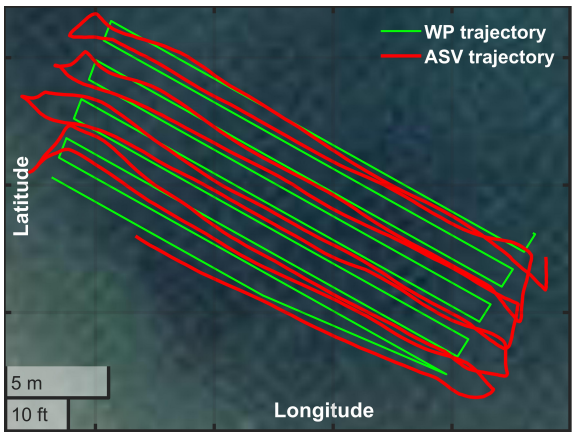


FIGURE A.1: Survey mode: ASV path during calibration as programmed using Mission Planner with the planned trajectory in green and the realized one in red.

better to set long transects during mission planning. Once the U-turn is made the ASV sticks well to the mission path.

Electrical consumption

Electrical consumption is an essential variable during the planning of an ASV mission and survey. It impacts the survey/tracking lifetime or the number of batteries to embed. The electrical consumption is calculated through the power sensor module and saved to the flight controller log.

The consumption mainly depends on the speed of the ASV. Other variables could impact the power consumption such as the behavior of the board, the sea condition, and the buoyancy/drag of the board. In this part, we analyze the results of speed differences and the load onboard the ASV.

For survey mode the ASV displacement pattern is divided into two parts in which the consumption and timing are different. Mission patterns are made with U-turn followed by straight line navigation (Figure 1.5). During the U-turn the ASV is in pivot turn mode and make two 90° sharp turn. Depending on the mission pattern the time to make U-turn is relatively small compared to navigation time in straight line. Electrical consumption and timing are averaged and compared for the two different phases in Table C.4 to find out the ratios of speed over consumption.

With the thrusters disconnected, the electrical consumption of all the electronics in internet mode is 0.75 A. The consumption for the remote mode is 0.66 A

The electrical consumption is measured for a speed of 0.8 m/s, 1 m/s, and 1.2 m/s after the calibration made on January 11 20222 in the Saint-Gilles marina mission (Fig. A.2). The average consumption for the different speeds in U-turn is about 1.6 A. The duration for the calculation of the average current consumption is 25 s for 0.8.m/s and 1m/s. For 1.2 m/s, the U-turn time is not relevant because the ASV struggles to make its turn. It would necessitate an appropriate calibration for this speed.

At 1 m/s in a straight line it is 5.20 A during 23 s. The mission pattern is the same as Figure A.1 with a short straight line. For the same mission at 1.2 m/s

TABLE A.1: ASV consumption for different configuration, i.e. different maximum authorized speed and different weight loads on the board

(a) Electrical consumption vs. ASV speed

Speed	**Average U-turn**		**Average transect**	
(m/s)	Current (A)	Time (s)	Current (A)	Time (s)
0.8	1.62	25	3.33	27
1	1.62	25	5.20	23
1.2	1.64	32	7.20	20

(b) Additional electrical consumption vs. supplementary load at 1.2m/s

Load	**Average U-turn**	**Average transect**
(kg)	Current (A)	Current (A)
5	+0.3	+0.9
10	+0.6	+1.1

the consumption is 7.20 A during 20 s for a straight transect. We summarize the consumption analysis in Table C.4.

Peak consumption >20A is reached during the turns at the end of each transect when thrusters are the most used. We made sure that the pivot turns would not be faster than 30°/s which limits this consumption. During mission planing, to estimate the electrical consumption, the user can calculate the total budget with the number of U-turn, their time and the estimated time of transect with their associated consumption. For example, at 1m/s, 100 m transects are made in around 105 s with the acceleration phase. Average consumption during transect is 5.20 A and 1.62 A during 25 s for the U-turn. The ASV mounted with two 10 Ah batteries could theoretically be able to complete 4.43 hours of surveys.

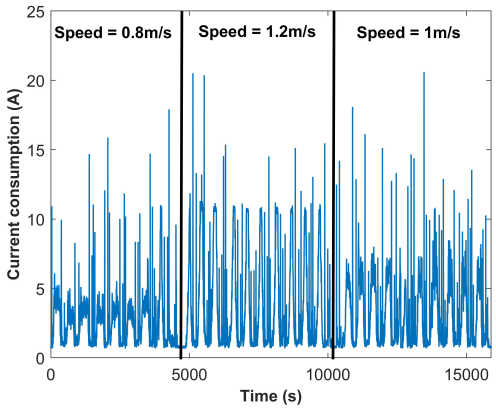


FIGURE A.2: Instantaneous current consumption of the ASV for different speeds in Saint Gilles marina tests during a survey mode (bathymetry and photgrammetry)

For the tracking mode we aim at following turtles and we do not need to reach a speed superior to 0.8 m/s. We find this value by analyzing the active swimming

phases of green turtles (1). The tracking is needed for a minimum of 5 h (see Table 1.1). With the thrusters disarmed, the electrical consumption for the internet mode of all the electronics including the *WaterLinked* board and the GoPro power supply is 0.66 A. During a tracking mission of 83 minutes in Saint-Gilles les Bains (-21.056312°N, 55.220266°E; on December 14 2021), the mean current consumption was 6.96 A with a target speed of 0.8 m/s (Figure A.3). These tests occur in open sea which tends to increase the consumption with the perturbation of the wave during navigation. It includes also some resting phases where the ASV does not move. Some consumption peaks induced by rapid change of direction in navigation can be observed (Figure A.3). These peaks are about 30 A. The maximal peak current is 59 A but occurs only once and may be due to a brutal change of direction. We conclude that for the 5 h wanted as minimal tracking duration, the battery requirement will be >35Ah.

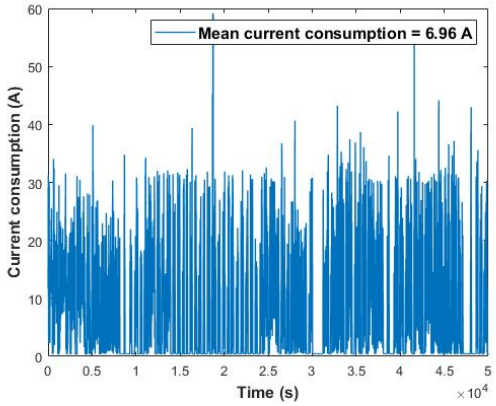


FIGURE A.3: Current consumption of the ASV during a tracking mode (acoustic survey)

A.3 Bathymetric survey standard from the International Hydrographic Organization

Evaluating criteras of the *order 1a* category are defined as follows:

- A maximum Total Horizontal Uncertainty (THU) of 5.5m
- $THU = 5m + 0.05 \times \text{max depth}$
- A maximum Total Vertical Uncertainty (TVU) of 0.52m
- $TVU = \sqrt{0.5^2 + (0.013 \times \text{max depth})^2}$
- A maximum Line Spacing (LS) of 3m
- $LS = 3 \times$ min depth or 25m if greater
- A bathymetric coverage less than 100%, as long as the least depths are obtained, and the bathymetry provides an adequate depiction of sea ground topography

- Detection of *cubic* features greater than 2m and in depths down to 40m, with a confidence level of 95%

The difference between bathymetry *order 1a* and *order 1b* is the complete coverage of the area to detect all features and obstacles. In our example we have a hyper-spectral bathymetry (Figure 1.9 (c)) of the whole area to which we will refer to plan the study so we fall in *order 1a*. Otherwise, our survey is in *order 1b*. A better feature detection of smaller shapes and a more accurate ground depiction necessarily go through a reduction of the overall uncertainties and/or an increase of the area coverage. However, those uncertainties are often significantly impacted by uncontrolled environmental conditions like, sea current, wind and wave strength. Some margins have to be applied when uncertainties estimations come from typical system specifications. Data oversampling and/or outliers filtering may help to reduce the influence of such random phenomenons during the

Bibliography

[1] P. Gogendeau, S. Bonhommeau, H. Fourati, D. De Oliveira, V. Taillandier, A. Goharzadeh, and S. Bernard, "Dead-Reckoning Configurations Analysis for Marine Turtle Context in a Controlled Environment," *IEEE Sensors Journal*, vol. 22, no. 12, pp. 12 298–12 306, Jun. 2022, conference Name: IEEE Sensors Journal.

Appendix B

Chapter 3 : Supplementary Material 1

In the material part 3.2.1, it is detailed how reference trajectories are acquired. The accuracy of the acoustic method with the Short Base Line (SBL) system can be impacted by several factors. Generally, the positions acquired have a error of at least 1 m. To study the speed and trajectory at fine time we need to filter the positions. For example, between two positions at 1 Hz with an accuracy of 1 m, we can have a maximum speed of 2 m/s while the turtle is stationary. Moreover, in some phases, depending on the sampling conditions, the accuracy of the measurement must be low. To clean the reference data we propose a simple and versatile method that does not require the development of complex filters. In the first part, we present the different steps of the proposed method. The second part shows the results of the reference data after the processing

B.1 Processing steps and validation method

-The first step of the reference data cleaning uses position's standard deviation value (STD) which is given by the manufacturer of the acoustic system. In our case, we set a threshold of 2.5 m and remove all positions above. This first cleaning step puts the reference data in several sequences because some complete swim phases give bad STD. In each sequence, we perform a re-sampling with a linear interpolation for the cleaned outliers.

-The second step is filtering. The difficulty is that we have to reduce the positioning error of the system without losing information on the fine-scale trajectory. A too strong filtering on the positions will give us a less tortuous trajectory as well as a loss of precision on the total distance. On the contrary, a too weak filtering will give aberrant speeds between two positions. To estimate if the fine scale tortuosity is related to the positioning error or to the real trajectory of the turtle we compare it with the heading changes calculated with the IMU. On the other hand, if speed measurements are outliers (>1.5 m/s) it is due to positioning error and our filtering is not strong enough. We used a simple moving average filter and tried different coefficient.

-The last step is to identify the movements where the turtle is resting and set the speed to 0. Indeed, the accuracy of 1 m of the system can induce residual speed even when the turtle is stationary. Some phases where the turtle is on the ground but active are complex to identify so we apply this correction only during the rest

phases. To identify these phases we used the VeDBA and the depth difference. If the VeDBA is lower than 0.006 g during 10 s we consider that the turtle is resting and that its speed is set to 0 m/s.

To validate our filtering method, we reconstructed the trajectory of the filtered sequences with the reference horizontal speed and the estimated orientation. By trial and error of different filtering coefficients, we chose the method that has the best 2DRMS horizontal error (See 3.3.5) for the reference positions with an STD of 1 m. We find 30 as best coefficient (For 1 Hz sampling).

The behavior of the animal after release may differ from its natural behavior. The stress induced by the capture generates different swimming behavior. In our case, the turtle do many shallow dives with a fast swimming. After the first long rest phase (>10 min), we estimate that the turtle's swimming is close to normal. The reference data study begins around 22 min after release. The complete dataset is decomposed into training and validation datasets. This verifies that the proposed method is not correlated with the training data. The best practice with machine learning (1) is at least 80% for training and 20% for validation.

B.2 Reference trajectory after processing

After our processing procedure, the reference data gives around 138 min of accurate positioning over the 228 min of acquisition. These are divided into 6 sequences. The first sequences are relatively short, ranging from 6.8 min to 23.3 min. The longer sequences are 58.3 min and 33.5 min.The last one is used for as validation sequence and represent around 25% of the reference data.

These data are presented on Figure B.1. The red trajectory represents the raw positions received from the SBL system. The green path represents the filtered data for which we estimated a good associated reference trajectory.

On Figure B.1, inset b) represents the estimated reference speed. In red is the speed without applying step 4 of the reference data filtering. The green curve is the speed after all filtering steps. This speed is used to develop our trajectory estimation algorithms. We have shaded the areas where we estimate that the turtle is stationary at the bottom. For these estimates, we compare the ODBA with the depth difference. The shaded areas are located with arrows in Figure B.1 a). It is clear that the speed associated with b) is not representative of the distance traveled. Considering the positioning error of the SBL system, this confirms the hypothesis of the static turtle. For these phases, we fix the speed at 0, an assumption close to the truth.

Bibliography

[1] A. Kumar. Machine learning - training, validation & test data set. [Online]. Available: https://vitalflux.com/machine-learning-training-validation-test-data-set/

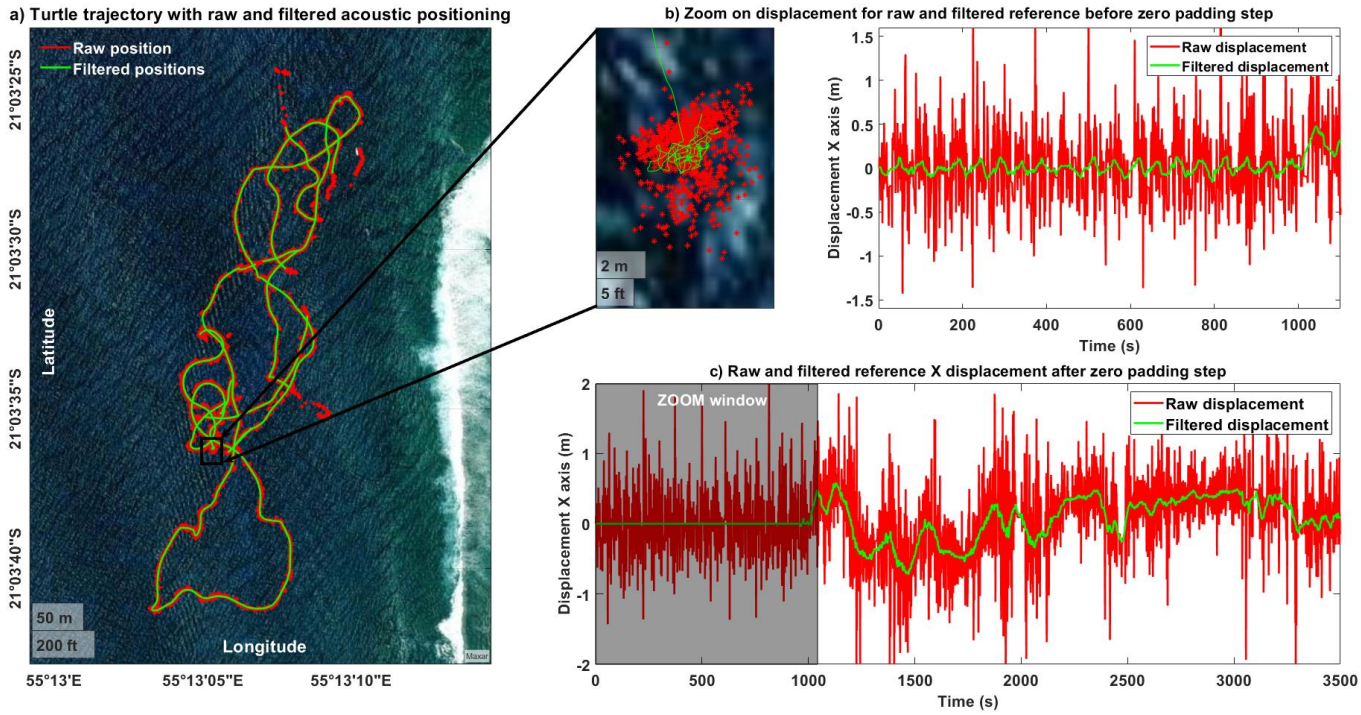


FIGURE B.1: a) the raw reference trajectory (red curve) and the filtered trajectory (green curve) of our complete dataset. b) zoomed trajectory, and the displacement of the turtle on the north axis of the terrestrial reference frame. On this part where the turtle is at rest, the position that is not stable and a displacement along the axis. In c) same displacement for a large sample after our step of correction of the phases of rest. The gray shaded area represents the part displayed on the insert b).

Appendix C

Chapter 3 : Supplementary Material 2

In this document, the method for measuring the timing of the proposed algorithms and calculating their associated power consumption is described. The tag program is coded and compiled via the Arduino IDE in C++, and codes are available on our GitHub[1]. Consumption tests on the electrical board are made in laboratory with the power analyzer N6705B.

To evaluate the average power consumption of our methods, we need to calculate the power consumption of the sensors as well as the computing consumption for each algorithm. For the latter, we need to calculate the CPU usage time and the sleeping time over a sampling acquisition duration. Then, these timings are associated with the average CPU consumption in RUN or SLEEP mode that we have calculated on the "IOT Turtle tracker" board.

C.1 Computing timing measurement

Timing are calculated for each function of the different algorithms for our analysis. We used the micros() function of Arduino. It allows recording the timestamp in microseconds when it is called. The different computational techniques used are not fixed and are dependent on the animal's behavior. To determine the average time of microcontroller use we calculated the timing of each function coupled with the result of the ethogram for the percentage.

Some functions are common to each technique but can be used at a different sampling frequency. At 10 Hz, the common function during 1 s sampling period includes 1x orientation, 10x low-pass filter, 10x VeDBA and 1x trajectory calculation. At 1 Hz common function during 1 s sampling period includes 1x orientation, 1x low-pass filter, 1x VeDBA and 1x trajectory calculation. Orientation calculations are processed at 1 Hz for each algorithm. The corresponding functions are defined below :

$$T_{10Hz} = T_O + T_{TR} + 10 \times T_{VeDBA} + 10 \times T_{LP} \tag{C.1}$$

with T_{10Hz} is the timing for common function for 10 Hz algorithms. T_O, T_{TR}, T_{VeDBA} and T_{LP} are timing to compute orientation, trajectory, VeDBA and low-pass

[1] https://github.com/pierregoge/Behavioral-based-Dead-Reckoning/tree/main/Arduino/IoT_Turtle_conso

filter.

$$T_{1Hz} = T_O + T_{TR} + T_{VeDBA} + T_{LP} \tag{C.2}$$

with T_{1Hz} is the timing for common function for 1 Hz algorithms. T_O, T_{TR}, T_{VeDBA} and T_{LP} are timing to compute orientation, trajectory, VeDBA and low-pass filter.

In Table C.1, we present the measured timing results

Function	T_O	T_{TR}	T_{VeDBA}	T_{LP}
Timing (ms)	1.285	0.666	0.111	0.135

(a) Timing of the different common fonction components

Function	T_{10Hz}	T_{1Hz}
Timing (ms)	**3.301**	**2.197**

(b) Total common timing at 10 Hz et 1 Hz

TABLE C.1: Tables with the common timings

In our calculation method, for each speed and ethogram function, we applied a percentage of occurrence (from the ethogram results) and its sampling frequency. Some conditional functions also give a different timing depending on the input variables. In the same way, depending on the percentage of occurrence of the conditions, we applied this percentage to the timing. The formulas to calculate the computing timing of the 4 different functions are defined below.

Computing time $F10_E$:

$$T_{F10_E} = T_{10Hz} + 10 \times T_j + T_{e1} + T_{e2} + T_{S_{F10_S}} \tag{C.3}$$

with T_{F10_E} is the total computing timing of $F10_E$, T_{10Hz} is the timing for common function for 10Hz algorithms, T_j is the timing of Jerk function, T_{e1}, T_{e2} are the timings for the different ethograms. $T_{S_{F10_S}}$ is the speed function timing. Timings are in ms.

T_{e1} and T_{e2} are dependent on several conditions. In every case, their computing is very fast, as it requires almost no calculation. For their values, we take a conservative fixed enhanced time.

To calculate the speed at 10 Hz we need the number of times the animal goes into the *H-PITCH* mode as a percentage over an acquisition time step and the percentage of time spent in the regular swimming phase.

$$T_{S_{F10_E}} = ((T_{oc} + T_{div}) \times N_{HP/s}) + \%HP \times T_+) + T_{S_R} \times \%S_R + T_{LP} \tag{C.4}$$

with $T_{S_{F10_E}}$ is the speed function computing timing of $F10_E$, T_{oc} is the timing for OCDR function, T_{div} is the timing of divide function, T_+ the timings for addition function and T_{S_R} the timing for regular swim speed function. Timings are in ms. $N_{HP/s}$ is the number of times the algorithm goes in *H-PITCH* mode per second. $\%HP$ and $\%S_R$ are percentage of *H-PITCH* and *S-REGULAR* swim behavior.

For the adaptive algorithms, the computing is different during resting phases as we don't need to calculate the orientation and speed.

$$Tr_{F10_E} = 10 \times T_{VeDBA} + 10 \times T_{LP} + 10 \times T_{e1} \tag{C.5}$$

with Tr_{F1} is the total computing timing of $F10_E$ during rest. T_{VeDBA} and T_{LP} are timing to compute VeDBA and low-pass filter.
Timing are in ms.

Computing time $F1_E$:

$$T_{F1_E} = T_{1Hz} + T_j + T_{e1} + T_{S_{F1_E}} \tag{C.6}$$

with T_{F1_E} is the total computing timing of $F1_E$, T_{1Hz} is the timing for common function for 1 Hz algorithms, T_j is the timing of Jerk function, T_{e1} is the timings for the ethogram layer 1. Timings are in ms.

To calculate the speed at 1 Hz, the number of times the animal goes into the *H-PITCH* mode as a percentage over an acquisition time step is needed.

$$T_{S_{F1_E}} = ((T_{oc} + T_{div}) \times N_{HP/s} + \%HP \times T_+) \tag{C.7}$$

with $T_{S_{F1_E}}$ is the speed function computing timing of $F1_E$, T_{oc} is the timing for OCDR function, T_{div} is the timing of divide function and T_+ the timing for addition function. Timings are in ms. $N_{HP/s}$ is the number of times the algorithm goes in *H-PITCH* mode per second. $\%HP$ is the percentage of *H-PITCH* swim behavior.

For the adaptive algorithms, the computing is different during resting phases as we don't need to calculate the orientation and speed.

$$Tr_{F1_E} = T_{VeDBA} + T_{LP} + T_{e1} \tag{C.8}$$

with Tr_{F1} is the total computing timing of $F1_E$ during the rest. T_{VeDBA} and T_{LP} are timing to compute VeDBA and low-pass filter. Timings are in ms.

Computing time $F10_S$:

$$T_{F10_S} = T_{10Hz} + T_{VeDBA} + T_{S_{F10_S}} \tag{C.9}$$

with T_{F10_S} is the total computing timing of $F10_E$, T_{10Hz} is the timing for common function for 10 Hz algorithms. Timings are in ms.

To calculate the speed for $F10_S$, a regression on the VeDBA variable is applied at 1Hz.

$$T_{S_{F10_S}} = T_{reg} \tag{C.10}$$

with $T_{S_{F10_S}}$ is the speed function computing timing of $F10_S$, T_{reg} is the timing for regression function on VeDBA. Timings are in ms.

Computing time $F1_S$:

$$T_{F1_S} = T_{1Hz} + T_{S_{F1_S}} \tag{C.11}$$

with T_{F1_S} is the total computing timing of $F1_E$, T_{1Hz} is the timing for common function for 10 Hz algorithms. Timings are in ms.

$$T_{S_{F1_S}} = T_{oc} \times \%HP \tag{C.12}$$

with $T_{S_{F1_S}}$ is the speed function computing timing of $F1_S$. T_{oc} is the timing for OCDR function. Timings are in ms. $\%HP$ is the percentage of *H-PITCH* swim behavior.

To calculate the $N_{HP/s}$ we consider two changes on *H-PITCH* mode per 15 min dive.

In Table C.2, we present the measured timing results

Name	T_j	T_{e1}	T_{e2}	T_{oc}	T_{div}	T_+
Timing (ms)	0.109	0.038	0.100	0.105	0.050	0.004

(a) Timing of the different proccesing functions

Name	$N_{HP/s}$	$\%S_R$	$\%HP$
Percentage	0.022	40%	12%

(b) Percentage to calculate speed's processing timing

Function	$T_{S_{F10_E}}$	$T_{S_{F10_S}}$	$T_{S_{F1_E}}$	$T_{S_{F1_S}}$
Timing (ms)	0.022	0.004	0.045	0.0126

(c) Processing timing of speed functions

Function	Tr_{F1_E}	Tr_{F1_E}
Timing (ms)	2.84	0.284

(d) Processing timing during *REST* for adaptative functions

Function	T_{F10_E}	T_{F10_S}	T_{F1_E}	T_{F1_S}
Timing (ms)	**6.072**	**2.348**	**3.457**	**2.210**

(e) Total processing time for each functions

TABLE C.2: Result of the timing for the components and in total

C.2 Consumption calculation

For the calculation of the average power consumption, we combined the active and resting microcontroller consumption and averaged it over one-time step (the bigger: 1000 milliseconds). In active microcontroller time, we added the calculated computing time and the acquisition time of the pressure sensor. The rest time of the microcontroller is the remaining time of the chosen time step. For the accelerometer and the magnetometer, we added the average consumption already calculated. This introduces a small bias in our calculation because the acquisition time of these sensors is not subtracted from the standby consumption time. According to the datasheet, it would be 7.4 ms. The formula used is the following:

$$Ca_{fct} = \frac{(T_{fct} \times C_r) + (T_p \times C_p) + ((1000 - T_{fct} - T_p) \times C_s)}{1000} + C_{a/m} \tag{C.13}$$

with Ca_{fct} the mean consumption for different functions in active phase, T_{fct} is the timing calculated at Equation C.3,C.9,C.12 and C.11 for the different functions, T_p is the timing of pressure sensor acquisition. Timings are in ms. C_r is the consumption of the tag in run mode, C_s is the consumption during sleep, C_p is the consumption of the pressure sensor, and $C_{a/m}$ is the mean consumption of the accelerometer and magnetometer. Consumption is in mA.

For the $F10_E$ and $F1_E$ functions during the *REST* phases, the consumption is slightly different. Indeed, we do not use the pressure sensor and magnetometer to save battery. For the behavior transition with the ethogram, the algorithm only needs the accelerometer data.

$$C_{rest} = \frac{(Tr_{F10_E/F1_E} \times C_r) + ((1000 - Tr_{F10_E/F1_E}) \times C_s)}{1000} + C_a \tag{C.14}$$

with C_{rest} is the mean consumption during resting phase for $F10_E$ or $F1_E$, $Tr_{F1/F2}$ is the timing during rest calculated at Equation C.5 and C.8 for $F10_E$ or $F1_E$. Timing are in ms. C_r is the consumption of the tag in run mode, C_s is the consumption during sleep and C_a is the mean consumption of the accelerometer. Consumption are in mA.

The total consumption for $F10_E$ or $F1_E$ is then equal to function C.15.

$$C_{F10_E/F1_E} = \%Re \times C_{rest} + (1 - \%Re) \times Ca_{F10_E/F1_E} \tag{C.15}$$

with $C_{F10_E/F1_E}$ are the mean consumption for $F10_E$ or $F1_E$ including low-power consumption during resting phase. $Ca_{F10_E/F1_E}$ is the active consumption of $F10_E$ or $F1_E$. $\%Re$ is the percentage in rest mode. Consumptions are in mA.

For the non-adaptative functions, their total consumption equal to their active consumption :

$$C_{F10_S/F1_S} = Ca_{F10_S/F1_S} \tag{C.16}$$

with $C_{F10_S/F1_S}$ is the mean consumption for $F10_S$ or $F1_S$. $Ca_{F10_S/F1_S}$ is the active consumption of $F10_S$ or $F1_S$. Consumptions are in mA.

The minimum frequency of the magnetometer is 10 Hz. The consumption difference for accelerometer 1 Hz and 10 Hz is negligible. We have therefore set the sampling frequency of accelerometer and magnetometer at 10 Hz for all algorithms.

Tests of the electrical consumption on the board are made in laboratory with the power analyzer N6705B. Their results are given in table C.3

CPU/sensor	**Timing (ms)**	**Consumption (mA)**
CPU rest	Average	0.020 mA
CPU run	Average	5.5 mA
Acc + Mag 10Hz	Average	0.089 mA
Acc	Average	0.048 mA
Pressure sensor	20	5.3 mA

TABLE C.3: Electrical consumption measurement on "IOT turtle tracker" board

Thanks to the measured timing, consumption, and the above formula, the average consumptions are calculated and presented in table C.4.

Name	$F10_E$	$F1_E$	$F10_S$	$F1_S$
C_{rest} **(mA)**	0.084	0.070	X	X
Ca_{fct} **(mA)**	0.248	0.227	0.236	0.227

(a) Consumption during *REST* and actives phases

Name	$F10_E$	$F1_E$	$F10_S$	$F1_S$
Consumption total (mA)	**0.207**	**0.196**	**0.236**	**0.227**

(b) Total consummation for all the studied functions

TABLE C.4: Tables with the consummation results

Appendix D

Chapter 4 : Data computation

In this document, the computation of the different algorithm parts, from raw sensor data to trajectory with Dead-reckoning (DR) are presented. The functions are coded and compiled via the Arduino IDE in C++, and codes are available on our GitHub[1].

D.1 Frames definition :

For positioning in the terrestrial frame such as GNSS position, our study refers to geodetic coordinates. These positions are given with the reference frame WGS-84.

In the application, two different local level frames are used and presented in Figure D.1. The first is called NED for North-East-Down. It is a local tangent plane attached to a geographic position (see Figure D.1). X-axis is North, Y-axis is East et Z-axis down. The second frame is the Turtle frame (also body frame). X-axis gives the direction of the animal through its anteroposterior axis, Y-axis is the lateral axis and Z-axis the dorsoventral axis. The axis differences between the frame give the animal orientation: Pitch, Roll and Yaw

D.2 2D Trajectories :

The trajectory is estimated with dead-reckoning (DR) in the horizontal plane. The depth is given by the pressure sensor. We calculate the displacement in the North-East-Down (NED) frame with the vector [x(t), y(t)]. The X axis is the northward axis, Y axis is eastward and Z axis is downward.

$$x(t) = x(t - \Delta t) + S_{turtle}(t) \times \Delta t \times cos(\theta_{turtle}(t))) \tag{D.1}$$

$$y(t) = y(t - \Delta t) + S_{turtle}(t) \times \Delta t \times sin(\theta_{turtle}(t))) \tag{D.2}$$

where $S_{turtle}(t)$ is the speed in turtle frame at time t, $\theta_{turtle}(t)$ corresponds to the heading in turtle frame at a time t

D.3 Sensor calibration:

Magnetometer: This sensor is the most affected by its calibration and if done incorrectly will provide aberrant results. The correction of the magnetometer is made by

[1] https://github.com/pierregoge/Behavioral-based-Dead-Reckoning

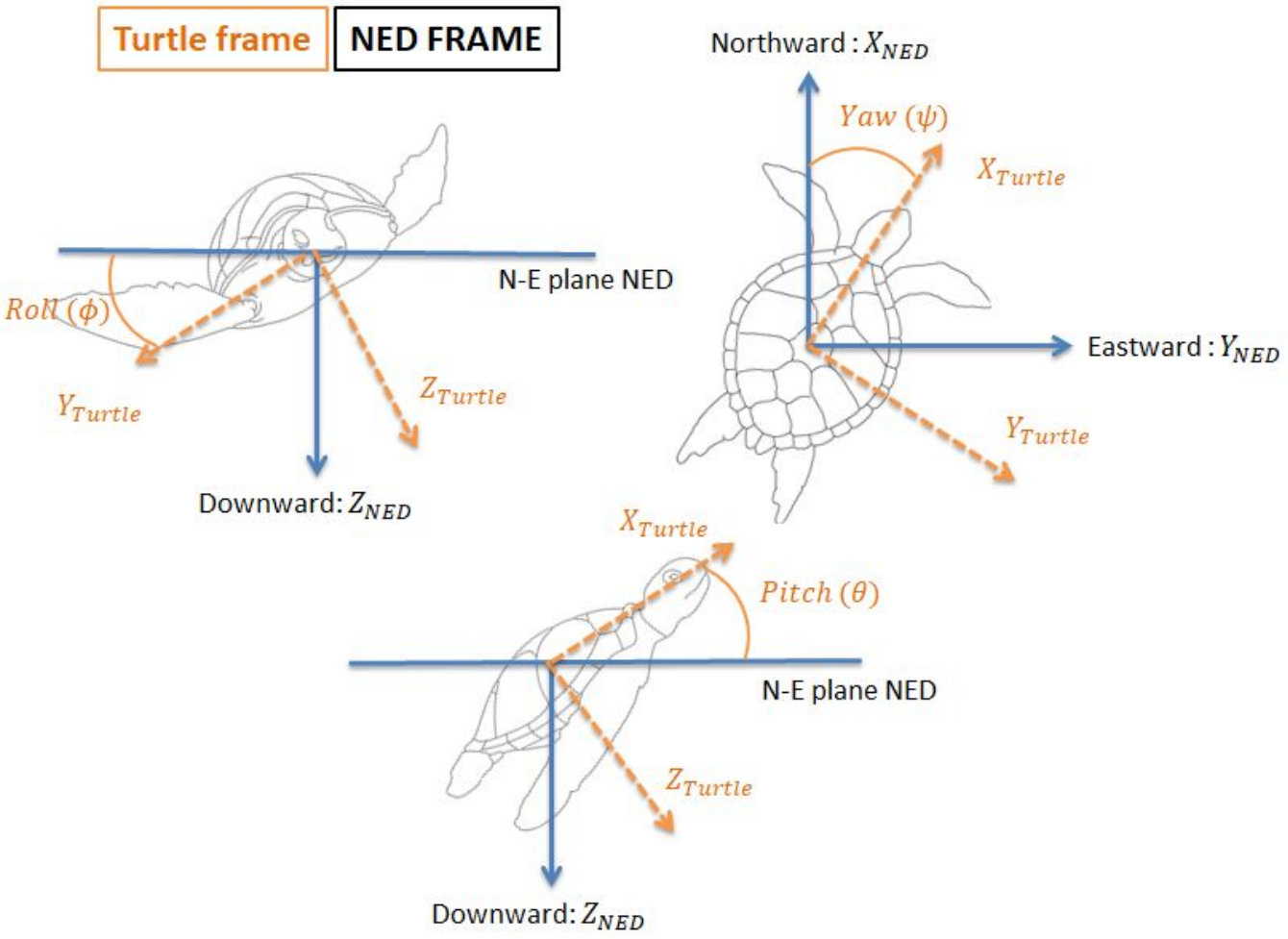


FIGURE D.1: Local frame with the animal attitude directions definition. Tag frame with the sensors is supposed to be aligned with the turtle frame

correcting two components, the hard and soft iron errors. The first one corrects the presence of the magnetic field around the sensor and the second one for the deformation due to the presence of ferromagnetic materials around the sensor. In our case, we use the simple solution proposed and described by Kris Winer (1). This solution is an approximation and other solutions are requiring complex resolutions (2). Once a satisfactory calibration has been performed, it can be saved in the program and used as long as the environment near the tag does not change. In case of a suspicious heading value it is a good idea to redo the calibration

Accelerometer: Calibration involves correcting the offsets when the tag is stationary and flat. The average value of each axis during the calibration time corresponds to the axis offsets. For the axis subject to earth's gravity (Z in our case), its component must not be removed.

Pressure sensors: For the pressure measurement we notice good linearity of the measurements but drift in the time. This drift can be due to several parameters such as sensor drift, atmospheric pressure or temperature change. The latter is very important in our algorithm because it is used to detect the surface. Our surface sensor does not give good results over time and the pressure measurement becomes essential. However, if the drift of the sensor is too strong ($>$ the surface threshold) we may not detect surfacing events anymore. For its calibration, we propose a simple method using the maximum dive time of the animal. Like marine mammals, turtles need to breathe and the minimum pressure during this time will be the surface pressure. This pressure becomes the offset of the sensor. Note that this method only applies to surface-breathing animals.

D.4 Low pass filter:

The low pass filter is a simple moving average filter with a 3-second window. It is applied to the accelerometer and magnetometer data. We set this filter for its simplicity and the low computational resources required. When analyzing data from a tag on a green turtle in Chapter 3 we determine the swimming frequency at 0.42 Hz. The targeted cutoff frequency of the filter is then 0.3 Hz. By trial and error, we test several coefficients of the moving average filter to get as close as possible to the result of a native low pass filter proposed by Matlab. The function developed is very simple. We sum the values of the acceleration and the magnetometer separately for each axis during the duration of a defined time window then we divide the result by the window size. For each axis, we use a circular buffer of the size of the filter window. The result obtained is then the acceleration or the magnetism without the frequencies higher than the cutoff frequency. The value obtained is the iteration in the middle of the filter window. The filter introduces a delay equivalent to half the size of the window. For the following iterations, we update the sum before division by subtracting the oldest value from the sum and summing the new value of the sensor. The latter is also pushed into the circular buffer to update it. This method saves computing time by reducing the number of additions at each iteration.

D.5 Orientation :

The orientation algorithm applied is SAAM (3) giving orientation in quaternion form. We choose this algorithm for its simplicity and low computation resources needed with 37 arithmetic operations (See Chapter 2). We convert the Quaternion to Euler angle via the formula :

$$\begin{bmatrix} \theta \\ \phi \\ \psi \end{bmatrix} = \begin{bmatrix} asin(2(q_0q_2 - q_3q_1) \\ atan2(2(q_0q_1 + q_2q_3), 1 - 2(q_1^2 + q_2^2)) \\ atan2(2(q_0q_3 + q_1q_2), 1 - 2(q_2^2 + q_3^2)) \end{bmatrix} \tag{D.3}$$

With θ the Pitch, ϕ the Roll and ψ the yaw. Angles are in radian. q_0, q_1, q_2 and q_3 are quaternion vector members.

For our algorithm, we need the heading. Its calculation is derived from the yaw. We add the magnetic declination and for all yaw < 0 we add $2 \times pi$ to get the heading between 0 and $2pi$. The magnetic declination of the terrain is determined via the NOAA website[2]. For our test location (-21.101903 S, 55.242766 E) it is 19.32°. The value dependents on the position and the date.

D.6 Speed and distance:

For the swimming behavior, we use a fixed speed, determined by the average speed of the active phases for the data of Chapter 2 because the operational context for the acquisition of the data of this study is the same. This speed is 0.52 m/s. The calculation of the distance traveled ΔD_t in meters is simple and can be summarized by adding the displacement of the animal at each iteration.

D.7 Conversion of NED trajectory to geodesic position:

As defined in the description of the algorithm, it allows the use of two modes: Geotrack or pseudotrack. For the pseudo-track mode only the X and Y positions in the NED frame are used and sent. For the geotrack we need to convert positions in geodetic coordinates (LLA). This phase follows the compression of the trajectory. We use the last position in the geodetic coordinates of the GNSS or the one of the previous trajectory, to calculate the next one with the distance traveled and the radius of the earth.

Function for conversion from displacement to geodetic position:

$$q(t) = \frac{d(\Delta t)}{r_{earth}} \tag{D.4}$$

[2] https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml

With q(t) the distance traveled divided by the earth radius. $d(\Delta t)$ the distance traveled between two temporal steps, r_{earth} the approximate radius of the earth. All distances are in meters.

$$lat(t) = asin(sin(lat(t-1))cos(q(t)+cos(lat(t-1))\times sin(q(t))cos(\theta_{turtle}(\Delta t)) \quad \text{(D.5)}$$

$$lon(t) = lon(t-1) + atan2(\frac{sin(\theta_{turtle}(\Delta t))sin(q(t))cos(lat(t-1))}{cos(q(t))sin(lat(t-1))sin(lat(t))} \quad \text{(D.6)}$$

With $lat(t)$ and $lon(t)$latitude and longitude of the position at the time t in rad. $\theta_{turtle}(\Delta t)$ is the heading of the turtle between two temporal step in radian.

These formulas described in (4) are an approximation by estimating that the earth is spherical This approximation in our case or the test area which is only a few kilometers is acceptable for our level of accuracy. The formulas can be simply adapted to overcome this approximation by using more complex models, but which adds computation time

D.8 Tortuosity:

The function for the calculation of the tortuosity is proposed and defined in Animal tag tools wiki. It compares the distance between the start and end points with the real distance traveled. The formula is:

$$T_o = \frac{\Delta D_t - \Delta D}{\Delta D_t} \quad \text{(D.7)}$$

With T_o the tortuosity index, ΔD_t the total distance traveled of the dive, and ΔD the distance between the starting and ending point of the dive. Distances are in meters.

The value of the index is close to 0 for a rectilinear movement.

Bibliography

[1] K. Winer, "Simple and Effective Magnetometer Calibration · kriswiner/MPU6050 Wiki." [Online]. Available: https://github.com/kriswiner/MPU6050

[2] T. Ozyagcilar, "Calibrating an eCompass in the Presence of Hard- and Soft-Iron Interference," Tech. Rep., 2013. [Online]. Available: https://www.nxp.com/docs/en/application-note/AN4246.pdf

[3] J. Wu, Z. Zhou, H. Fourati, and Y. Cheng, "A Super Fast Attitude Determination Algorithm for Consumer-Level Accelerometer and Magnetometer," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 3, p. 375, Aug. 2018. [Online]. Available: https://hal.inria.fr/hal-01922922

[2] [http://www.animaltags.org/doku.php?id=tagwiki:tools:processing:tortuosity]

[4] O. R. Bidder, J. S. Walker, M. W. Jones, M. D. Holton, P. Urge, D. M. Scantlebury, N. J. Marks, E. A. Magowan, I. E. Maguire, and R. P. Wilson, "Step by step: reconstruction of terrestrial animal movement paths by dead-reckoning," *Movement Ecology*, vol. 3, no. 1, p. 23, Sep. 2015. [Online]. Available: https://doi.org/10.1186/s40462-015-0055-4

Abstract

Geolocation of wild species is a critical source of information for ecological studies and for managers to make decisions related to the habitat and distribution of these species. Several devices (e.g. electronic tags) have been developed to track animals. It is however very difficult to have accurate trajectories at fine temporal and spatial scale for marine species. Indeed, sea water and wild animals complicate the acquisition of accurate positions, especially for long distance.

The subject of the thesis is the development of a solution for underwater trajectory estimation and transmission, which can be embedded within a bio-telemeter. It uses the fusion of data from different sensors. These sources can be particularly heterogeneous (inertial data, depth and velocity estimation, geolocated position).

To estimate trajectories and accuracy of this estimation, reference data are collected using platforms developed over this PhD, i.e. an Autonomous Surface Vehicle (ASV) with underwater acoustic geolocation and a differential GPS. First, we test different algorithms and present the trade-off of these different solutions in terms of accuracy, consumption, and portability. The trajectory estimation techniques developed and embedded are then improved through the inclusion of the animal behavior to speed estimation. This has another advantage, which is to complete the trajectory information with an ethogram calculated in-board. A final step, the developed algorithms are embedded within a bio-telemeter. Several tests are performed in a controlled environment for different contexts of analysis. The proposed solutions allow novice or experienced users in programming to use the tools for different degrees of analysis. Finally, post-processing of the transmitted data is used to show analysis potential and possible improvements that could be performed. The biological model used for this PhD is the sea turtles.

Résumé

La géolocalisation des espèces sauvages est une source d'information essentielle pour les études écologiques et les gestionnaires afin de prendre des décisions en lien avec l'habitat et les distributions de ces espèces. De nombreux dispositifs ont été développés pour suivre les animaux. Il est cependant très difficile d'avoir des trajectoires précises à une fine échelle temporelle et spatiale pour les espèces marines. En effet, l'eau de mer et le comportement des animaux sauvages compliquent l'acquisition de positions précises, surtout sur de longues distances.

Le sujet de cette thèse est le développement d'une solution d'estimation et de transmission de trajectoires sous-marines, qui peut être embarquée dans un bio-télémetre. Elle utilise la fusion de données provenant de différents capteurs. Ces sources peuvent être particulièrement hétérogènes (données inertielles, estimation de la profondeur et de la vitesse, position géolocalisée).

Pour estimer les trajectoires et la précision de l'estimation, des données de référence sont collectées à l'aide de plateformes développées lors cette thèse, à savoir un véhicule de surface autonome (ASV) avec géolocalisation acoustique sous-marine et un GPS différentiel. Dans un premier temps, nous testons différents algorithmes et présentons le compromis de ces différentes solutions en termes de précision, de consommation d'énergie et de portabilité. Les techniques d'estimation de trajectoires développées et embarquées sont ensuite améliorées en prenant en compte le comportement de l'animal pour l'évaluation de sa vitesse. Ceci présente un autre avantage, celui d'apporter des informations supplémentaires à la trajectoire par un éthogramme calculé en embarqué. La dernière étape consiste à intégrer et à tester les algorithmes développés dans un bio-télémètre. Plusieurs tests sont effectués dans un environnement contrôlé, et ce pour différents contextes d'analyses. Les solutions proposées permettent aux utilisateurs novices ou expérimentés en programmation d'utiliser les outils pour différents degrés d'analyses. Enfin, le post-traitement des données transmises est utilisé pour montrer le potentiel d'analyses et les améliorations possibles qui pourraient être réalisées. Le modèle biologique utilisé pour cette thèse est la tortue de mer.

UNIVERSITÉ DE MONTPELLIER