



## RESEARCH ARTICLE

# Communication-constrained cooperative bathymetric simultaneous localisation and mapping with efficient bathymetric data transmission method

Teng Ma,<sup>1,2</sup>  Wenjun Zhang,<sup>1\*</sup> Ye Li,<sup>1</sup> Yuxin Zhao,<sup>3\*</sup> Qiang Zhang,<sup>1</sup> Xiaojun Mei,<sup>4,5</sup> and Jiajia Fan<sup>1</sup>

<sup>1</sup> Science and Technology on Underwater Vehicle Laboratory, Harbin Engineering University, Harbin, China.

<sup>2</sup> Laboratory of Science and Technology on Marine Navigation and Control, China State Shipbuilding Corporation, Tianjin, China

<sup>3</sup> College of Automation, Harbin Engineering University, Harbin, China.

<sup>4</sup> Laboratory of Robotics and Systems in Engineering and Science (LARSyS), ISR/IST, University of Lisbon, Lisbon, Portugal.

<sup>5</sup> College of Information Engineering, Shanghai Maritime University, China.

\*Corresponding author. E-mail: [zhangwenjunheu103@163.com](mailto:zhangwenjunheu103@163.com); [zhaoyuxin@hrbeu.edu.cn](mailto:zhaoyuxin@hrbeu.edu.cn)

**Received:** 26 May 2021; **Accepted:** 16 November 2021; **First published online:** 4 April 2022

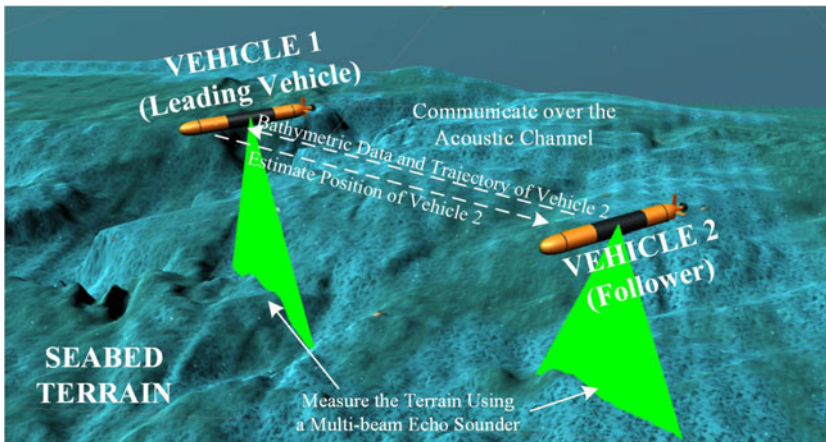
**Keywords:** communication-constrained; cooperative navigation; bathymetric SLAM; AUV; graph method

## Abstract

Bathymetric simultaneous localisation and mapping (SLAM) methods yield accurate navigation results for autonomous underwater vehicles (AUVs) and can construct consistent seabed terrain maps. Multiple independently working vehicles can complete tasks like surveying and mapping efficiently, which means cooperative bathymetric SLAM using multiple AUVs is suitable for large-scale seabed mapping. However, the transmission of bathymetric measurements collected using a multi-beam echo sounder over a low bandwidth, noisy, and unreliable acoustic channel is difficult, making cooperative bathymetric SLAM very challenging. This paper develops a graph-based cooperative bathymetric SLAM system that can compress many bathymetric measurements into small-scale acoustic packets and yield accurate navigation results with a 10% loss of acoustic packets caused by unreliable acoustic communication. According to the simulation conducted using the field data, the new algorithm is shown to be robust and capable of providing accurate location and mapping results over a low bandwidth, noisy, and unreliable acoustic channel.

## 1. Introduction

In recent years, autonomous underwater vehicles (AUVs) have become important tools for ocean exploration and seabed topography mapping, but the consistency of the built seabed map is dependent on the errors associated with AUV navigation systems. Navigation methods such as dead reckoning (DR), long baseline (LBL) acoustic positioning, and ultrashort baseline (USBL) acoustic positioning all provide accurate estimates for the vehicle's positions (Paull et al., 2013). However, the resulting navigational error of a DR system will grow over time even with the supplement of a Doppler velocity log, thus rendering it unsuitable for achieving prolonged autonomy at sea. Locating sea-floor mounted baseline transponders of LBL positioning systems on deep seas can be a particularly complex process and may involve costly undertakings (Casalino et al., 2014). Baseline transponders in USBL positioning systems are mounted on the ship instead of the seabed, but the positioning accuracy achieved by USBL positioning systems is dependent on the working depth of the vehicle and an error of approximately 0.5% of the range between the baseline transponders and the vehicle makes them unsuitable for deep-sea navigation (Mandić et al., 2016).



**Figure 1.** Cooperative bathymetric SLAM system with two vehicles.

Bathymetric simultaneous localisation and mapping (SLAM) methods have shown the potential to yield accurate navigation results for vehicles and construct consistent seabed maps in unknown areas (Barkby et al., 2012). In most bathymetric SLAM methods, a vehicle measures the seabed terrain using a multi-beam echo sounder (MBES) while estimating its location with a DR system simultaneously. Loop closures are then detected according to the similarity between current and historical bathymetric measurements. With these loop closures, the trajectory that includes all states of the vehicle will be estimated. In Ma et al. (2020b), the bathymetric SLAM method was proved to be an accurate and robust navigation method in long endurance underwater navigation. However, due to the limited endurance of a single vehicle, mapping a large-scale seabed terrain using a single vehicle is quite time-consuming.

Multiple working vehicles can complete complex tasks, like surveying targets over a large-scale area, more quickly (Ouyang et al., 2020). Many breakthroughs have been made in cooperative SLAM. However, most of the research on cooperative SLAM was conducted using underwater camera (Bonin-Font and Burguera, 2020) or side-scan sonar (Brown et al., 2008; Paull et al., 2015). Underwater visual SLAM can only work when the AUV operates in clear water and near the seabed (Kim and Eustice, 2013), and the accuracy of cooperative side-scan sonar SLAM is highly dependent on the richness of underwater features, such as manganese nodules. Compared with the techniques listed above, cooperative bathymetric SLAM is more suitable for underwater navigation and mapping.

Figure 1 illustrates a cooperative bathymetric SLAM for a two-vehicle system that includes a leading vehicle and a follower. Both vehicles in the cooperative bathymetric SLAM system measure the seabed terrain using MBES. The follower regularly broadcasts acoustic packets which consist of its historical state estimates and bathymetric measurements through the acoustic channel, while the leading vehicle will estimate the trajectories of both vehicles and broadcast the estimated trajectory of the follower.

The performance of the cooperative SLAM system is contingent on the quality of the communications channel. Communication using the acoustic channel is low bandwidth ( $\approx 1$  Kb/s), high latency (signals travel at speed of sound  $\approx 1,500$  m/s), and even unreliable (about 10% packet loss), making transmitting a substantial amount of bathymetric data over an acoustic channel very challenging. Meanwhile, unlike both visual and side-scan sonar SLAM methods, bathymetric SLAM is featureless, which leads to difficulties in compressing the bathymetric measurements by extracting landmarks from them. Octree algorithms are usually applied to compress the bathymetric data (Palomer et al., 2014), but the octree is a complex structure composed of many highly structured nodes, which means that the entire structure may drop out catastrophically when transmitting over the noisy and unreliable channel.

The contribution of this paper is to propose a communication-constrained cooperative bathymetric SLAM framework. To broadcast a large number of bathymetric points using minimal communication, the proposed cooperative SLAM method applies a simple and efficient bathymetric data transmission

method, that relies on a three-stage architecture, namely, the compression, transmission, and reconstruction stages. Using the proposed bathymetric data transmission method, vehicles can use acoustic packets containing several pseudo points to share a large sub-map in real time, and the size of acoustic packets generated by this method is completely an artificial constant to meet the bandwidth constraints of different acoustic communication devices. Thus, the proposed communication-constrained cooperative bathymetric SLAM algorithm can locate vehicles accurately and construct consistent bathymetric maps over the low bandwidth, high latency, and unreliable acoustic communication.

## 2. Related work

Most of the research on bathymetric SLAM was conducted using the filtering theory or the graph method. Barkby et al. (2012), Massot-Campos et al. (2019), Norgren and Skjetne (2018), and Kim and Jung (2011) all provide filtering bathymetric SLAM methods, in which the first three methods are based on particle filters and the last one is achieved using the extended Kalman filter (EKF). To deal with the incorrect measurement update caused by the incorrect panel assignment in Kim and Jung (2011), Jang and Kim (2019) proposed a dynamic grid adaptation method that selected panels by calculating the confidence interval of the vehicle's position. Palomer et al. (2016) proposed a probabilistic iterative closest point (ICP) method to estimate uncertainties of measured points in loop closure detection and tested its performance in an EKF implementation. López et al. (2017) applied an EKF module to fuse state estimates yielded by both visual SLAM and scan matching methods. Yuan et al. (2017) proposed an augmented EKF (AEKF) SLAM method in which state parameters are estimated using a recursive and iterative estimation-update process. Marchel et al. (2020a) has proved that the range-bearing SLAM method can yield accurate navigation results, but the proposed algorithm relied on the aid of beacons. The nearly flat seabed terrain may lead to invalid loop closures in bathymetric SLAM, making identifying all invalid loop closures crucial. Filtering bathymetric SLAM methods cannot update or remove historical loop closures, which means a loop closure will have a lifetime impact on the map constructed once it has been detected, whether the loop closure is valid or not.

The graph method proposed by Lu and Milios (1997) is a flexible way for the robust bathymetric SLAM method to consider invalid loop closure identification. Torroba et al. (2019) applied the graph-based bathymetric SLAM method to maximise the geometric consistency of overlapping regions in industrial-scale bathymetric surveys. Ma et al. (2020a) designed a graph-based bathymetric SLAM framework that can identify invalid loop closures according to the consistency of the built seabed map. On the basis of traditional graph-based bathymetric SLAM, Torroba et al. (2020) modelled the uncertainty of the generalised ICP (GICP) registration result of a pair of sub-maps in bathymetric SLAM. They trained PointNetKL, an artificial neural network, using the KL divergence between the learned uncertainty distribution and one computed by the Monte Carlo method as the loss, and performance of their method has been tested using field data collected in two long-term AUV missions. Similar to the graph method, Deschaut (2018) proposed an implicit moving least-squares (IMLS) SLAM framework in which IMLS scan-to-model framework replaced the classical ICP to detect loop closures.

Some breakthroughs have also been made in cooperative SLAM methods. Demim et al. (2018) proposed the ASVSF-DCSLAM method that solved the decentralised cooperative SLAM (DCSLAM) problems using an adaptive smooth variable structure filter (ASVSF) algorithm. To improve the consistency of the cooperative SLAM system, they implemented the adaptive covariance intersection method for the evaluation of the estimates on shared features' positions. Trujillo et al. (2018) designed an EKF-based cooperative SLAM system for the multi-UAV system with monocular cameras. The main contribution of this research is that they analysed the observability of the aerial multi-robot system and proved that the proposed system considerably reduces the unobservable modes.

As for communication-constrained cooperative SLAM, Choudhary et al. (2017) devise a distributed algorithm, named 'two-stage pose graph optimisation', in which globally optimal trajectories of multiple vehicles can be estimated using minimal communication. They also extracted several objects from the

raw RGB (red, green, and blue) data and constructed an object-based model to decrease the sizes of communication packets. Mangelson et al. (2018) aligned trajectories of vehicles using low-dimensional feature vectors instead of image feature descriptors, which decreases the amount of communication data between vehicles. Paull et al. (2015) designed a cooperative SLAM system using side-scan sonar and proposed a marginalisation method of intermediate poses to reduce the size of acoustic packets. Most of the research listed above compressed the raw data by extracting objects or visual features. As a featureless method, it is difficult to extract any objects, lines, or surfaces from the seabed terrain in the bathymetric SLAM methods, making real-time broadcast of bathymetric measurements between vehicles quite challenging.

### 3. System definition

The cooperative bathymetric SLAM system is defined by considering the scenario shown in Figure 2. The cooperative bathymetric SLAM system includes two vehicles: the leading vehicle (vehicle 1) and a follower (vehicle 2). Both navigate at the same depth which can achieve optimal mapping accuracy and coverage for the vehicles.  $X_i^1 (i = 1, 2, \dots, N_1)$  and  $X_i^2 (i = 1, 2, \dots, N_2)$  are states of vehicles 1 and 2 at successive instants of time  $i$ , respectively, in which  $N_1$  and  $N_2$  represent the numbers of states of both vehicles during a complete mission. DR systems are used to yield navigation data (including control vectors  $u_i^1 (i = 1, 2, \dots, N_1 - 1)$  and  $u_i^2 (i = 1, 2, \dots, N_2 - 1)$  for vehicles 1 and 2, respectively) for both vehicles. Bathymetric data collected by MBES includes distances  $\beta_j$  and angles  $\alpha_j$  ( $j = 1, 2, \dots, N_b$ , where  $N_b$  denotes the number of beams transmitted by MBES) from the MBES head to the terrain, and the locations where the beams hit the terrain are named as measured points. With the bathymetric measurements obtained with the MBES and corresponding navigation data, the horizontal positions  $X_i^{MP}[j] (j = 1, 2, \dots, N_b)$  and terrain depths  $h_i^{MP}[j] (j = 1, 2, \dots, N_b)$  of measured points at successive instants of time  $i$  can be computed. Details of computation methods of both  $X_i^{MP}[j]$  and  $h_i^{MP}[j]$  have been reported in Marchel et al. (2020a).

Bathymetric data and vehicle states are stored as various sub-maps in the bathymetric SLAM method. A sub-map in the bathymetric SLAM system usually consists of a local trajectory of the vehicle and a set of swaths, while a swath includes a large number of measured points. Here the  $k$ -th sub-maps constructed by the leading vehicle or vehicle 2 are denoted as  $m_k^1 (k = 1, 2, \dots, M_1)$  or  $m_k^2 (k = 1, 2, \dots, M_2)$ , respectively, where  $M_1$  or  $M_2$  are numbers of sub-maps built by these two vehicles during a complete mission. When vehicle 2 constructs a new sub-map, this sub-map will be broadcast using some acoustic packets, and the leading vehicle will detect loop closures between this sub-map and all historical sub-maps after capturing these packets. Meanwhile, the leading vehicle constructs various sub-maps by measuring the terrain continuously, thus loop closures are also detected between the sub-map constructed by vehicle 1 and all historical sub-maps. Here,  $z_{k,l}^{1,2}$  is used to denote the loop closure associated with states  $X_k^1$  of the leading vehicle and  $X_l^2$  of vehicle 2, while  $z_{k,l}^{1,1}$  is the loop closure associated with both states  $X_k^1$  and  $X_l^1$  of the leading vehicle.

The relative range between these two vehicles can be estimated efficiently according to the time difference between packet sending and receiving, using the synchronous-clock, one-way-travel-time acoustic method. Range measurement  $r_{i,k}$  represents the distance between the state  $X_i^1$  of the leading vehicle and the state  $X_k^2$  of vehicle 2. With vehicles' states, control vectors, loop closures, and range measurements, the estimated trajectories  $\mathbf{X}^{*1}$  and  $\mathbf{X}^{*2}$  of both vehicles 1 and 2 can be generated, and a detailed explanation is given in Section 5.

### 4. Bathymetric data transmission

#### 4.1. Acoustic packet generation

Sharing sub-maps among multiple vehicles is essential in the cooperative bathymetric SLAM system. However, transmitting all measured points of sub-maps through the low bandwidth acoustic channel is

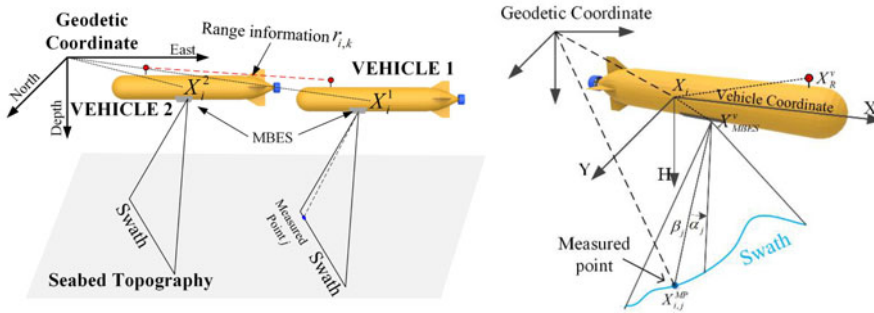


Figure 2. System definition of cooperative bathymetric SLAM system.

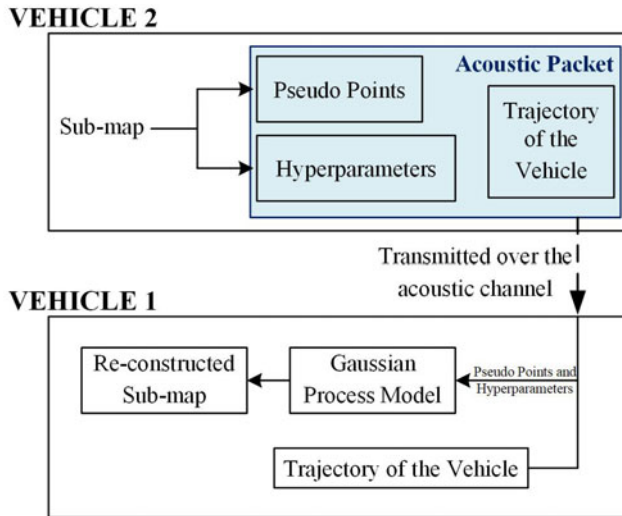


Figure 3. Transmission of the acoustic packets from vehicle 2 to vehicle 1.

infeasible. Vehicles must compress the information included in sub-maps before broadcasting them to reduce the data throughput. Meanwhile, unlike visual SLAM algorithms, loop closures in bathymetric SLAM methods are detected according to the similarity between terrain depths of all measured points in a pair of sub-maps instead of some features extracted. Thus, the terrain depths of nearly all measured points in a sub-map are still need to be estimated after the sub-map is highly compressed.

The sparse pseudo-input Gaussian processes (SPGPs) registration method is applied for registration and prediction by constructing a SPGPs model using historical data. In the authors' previous work, the SPGP registration has been proved to be both efficient and accurate in terrain depth estimates. Unlike traditional Gaussian processes registration (GPR) methods, the SPGPs model is trained using only a small number of pseudo points which maintain most of the information of the sub-map, instead of all historical bathymetric measurements. Thus, as shown in Figure 3, vehicle 2 compresses the sub-map into some pseudo points and calculates the hyperparameters of the SPGPs model associated with the sub-map, then the compressed sub-map will be reconstructed in vehicle 1, which makes it possible to transmit sub-maps using limited data throughput.

For sub-map  $i$ , with generalised terrain depth  $\mathbf{H}_i = \{h_i[n], n = 1, 2, \dots, N_b\}$  from MBES data consisting of  $N_b$  measured points corresponding to depths  $h_i[n]$  and horizontal positions  $x_i[n]$ , the horizontal positions  $\mathbf{X}'_i = \{x'_i[m], m = 1, 2, \dots, M\}$  of  $M$  pseudo points can be computed by minimising

$$p(\mathbf{H}_i | \mathbf{X}_i, \mathbf{X}'_i, \Theta) = N(0, \mathbf{K}_{NM} \mathbf{K}_M^{-1} \mathbf{K}_{MN} + \Lambda + \sigma^2 \mathbf{I}), \tag{1}$$

where  $\mathbf{X}_i = \{x_i[n], n = 1, 2, \dots, N_b\}$  denotes horizontal positions of measured points in the sub-map  $i$ ,  $\Theta$  is the set of hyperparameters, and  $\sigma^2\mathbf{I}$  represents the covariance of terrain depths in this sub-map. In the above,  $\Lambda = \text{diag}(\lambda)$  ( $\lambda = \{\lambda_n\}, n = 1, 2, \dots, N_b$ ), and

$$\lambda_n = K(x_i[n], x_i[n]) - \mathbf{k}_n^T \mathbf{K}_M^{-1} \mathbf{k}_n. \tag{2}$$

In Equations (1) and (2),  $[k_n]_m = K(x_i[n], x'_i[m])$ ,  $[\mathbf{K}_M]_{mm'} = K(x_i[m], x_i[m'])$ , both for  $m = 1, 2, \dots, M$ ;  $[\mathbf{K}_{NM}]_{nm} = K(x_i[n], x_i[m])$ , for  $n = 1, 2, \dots, N_b$  and  $m = 1, 2, \dots, M$ .  $K(x_i[n], x_i[n'])$  is the kernel function of the SPGPs model between measured point  $n$  and  $n'$ , given by

$$K(n, n') = c \exp\left(-\frac{1}{2} \|b(x_i[n] - x_i[n'])\|_2\right), \tag{3}$$

where  $c$  and  $b$  are hyperparameters of the SPGPs model, and  $\theta = \{b, c\}$ . In (1),  $\Theta = \{\theta, \sigma^2\}$ . Equation (3) is chosen as the kernel function because it has been proved to be effective in terrain depth estimation according to reference (Barkby et al., 2012).

With the positions  $\mathbf{X}'_i = \{x'_i[m], m = 1, 2, \dots, M\}$  of pseudo points and hyperparameters  $\Theta = \{b, c, \sigma^2\}$  of the SPGPs model, the predicted depth  $h'_i[m]$  of the pseudo point  $m$  is given by

$$p(h'_i[m]|x'_i[m], \mathbf{X}_i, \mathbf{H}_i, \Theta) = N(h'_i[m]|\mu_m, \Omega_m), \tag{4}$$

where

$$\mu_m = \mathbf{k}_{\mathbf{X}'_i}^T (\mathbf{K}_N + \sigma^2\mathbf{I})^{-1} \mathbf{H}_i, \tag{5}$$

$$\Omega_m = K(x'_i[m], x'_i[m]) - \mathbf{k}_{x'_i[m]}^T (\mathbf{K}_N + \sigma^2\mathbf{I})^{-1} \mathbf{k}_{x'_i[m]} + \sigma^2. \tag{6}$$

In Equations (5) and (6),  $[\mathbf{k}_{x'_i[m]}]_n = K(x_i[n], x'_i[m])$ ,  $[\mathbf{K}_N]_{nn'} = K(x_i[n], x_i[n'])$ , for  $n = 1, 2, \dots, N_b$  and  $n' = 1, 2, \dots, N_b$ .

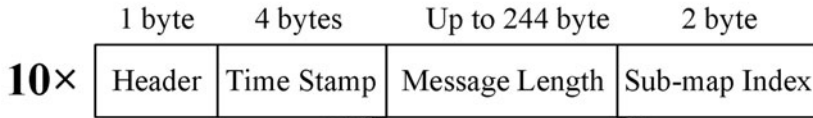
Unlike the octree method in which the size of acoustic packets is dependent on the terrain complexity, the size of acoustic packets generated using the proposed method is completely artificial constant. The number of pseudo points are chosen according to the average size of the sub-maps. In this paper, the number  $M$  of pseudo points is set as 150 (it has been proved that 150 pseudo points are enough for bathymetric data transmission, see Section 6), thus the acoustic packet of a sub-map will consist of 150 pseudo points (including both  $\mathbf{X}'_i$  and  $\mathbf{H}'_i$ ), hyperparameters  $\Theta$ , and the local trajectory included in sub-map  $i$ . The length of the trajectory included in a sub-map is usually no more than 40 s.

As shown in Figure 4, pseudo points and the local trajectory included in a sub-map are broadcast using 10 acoustic packets by vehicle 2 with a frequency of nearly 0.25 Hz. When one or two packets out of 10 packets travelling across an acoustic channel fail to reach their destination, the leading vehicle still captures 120~135 pseudo points and all hyperparameters of the SPGPs model. Simulation results in Section 6 proved that valid loop closures can be detected when the leading vehicle received no more than 120 pseudo points. The structures of packets broadcast by the leading vehicle are also given in Figure 4. When new loop closures are detected in the leading vehicle, the leading vehicle will beam the estimated position of the last state of vehicle 2 contained in the acoustic packets captured.

#### 4.2. Sub-map reconstruction

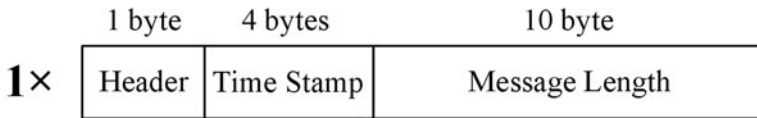
In the sub-map reconstruction stage, a gridded sub-map is constructed using the GPR method. With the acoustic packet described above, the terrain depth in each node  $\mathbf{X}_i^G = \{x_i^G[k], k = 1, 2, \dots, K\}$  of the gridded sub-map and corresponding variance  $\Omega_i = \{\Omega_i[k], k = 1, 2, \dots, K\}$  can be estimated using Equations (5) and (6) by replacing  $\mathbf{X}_i$ ,  $\mathbf{X}'_i$ , and  $\mathbf{H}_i$  with  $\mathbf{X}_i^G$ ,  $\mathbf{X}_i^G$ , and  $\mathbf{H}'_i$ , respectively. The kernel function used in the sub-map reconstruction stage is shown in Equation (3) and all hyperparameters required have been included in the captured acoustic packets.

**(1) Acoustic Packets From Vehicle 2 to Vehicle 1**



ITEM	DESCRIPTION	TYPE	LENGTH
Trajectory	A sub-trajectory with length of 3~4 seconds	Float	45~48 bytes
Pseudo Points	Positions and depths of 15 pseudo points		180 bytes
Hyperparameters	All hyperparameters of Gaussian process registration		16 bytes

**(2) Acoustic Packets From Vehicle 1 to Vehicle 2**



ITEM	DESCRIPTION	TYPE	LENGTH
Sub-map Index	Index of latest sub-map captured by vehicle 1 (sub-map $j$ , for example)	Int	2 bytes
Estimated Position	Estimated position of vehicle 2 at the last time of the trajectory included in sub-map $j$ (horizontal position)	Float	8 bytes

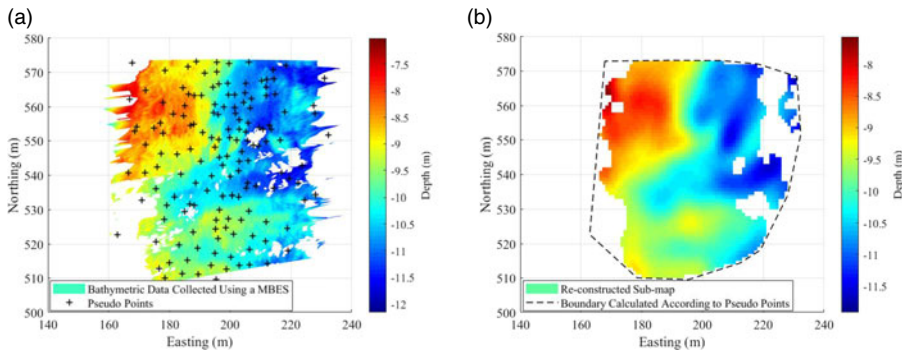
*Figure 4. Acoustic packet frame structures.*

The coverage of sub-map  $i$  is estimated according to the positions of pseudo points received. The horizontal boundary of the reconstructed sub-map is calculated according to the Delaunay triangulation constructed using all pseudo points. More specifically, unstructured grids are generated based on all pseudo points, and the outermost edges of all grids are represented as the boundaries of the sub-map. In the reconstructed map, terrain depths with a covariance of more than 0.1 will be dropped to improve the accuracy of residual error calculation in the loop closure detection.

As shown in Figure 5(a), a sub-map is constructed using bathymetric data collected in vehicle 2 and then a set of pseudo points is generated. Using the acoustic packet transmitted over the acoustic channel, the gridded sub-map (shown in Figure 5(b)) is reconstructed using Equations (5) and (6) in the leading vehicle. The reconstructed sub-map is smoother than that measured by vehicle 2, but both sub-maps show similar terrain changes. The loop closure detection results associated with reconstructed sub-maps will be shown in Section 6.

**4.3. Loop closure detection between sub-maps**

Loop closures between a pair of sub-maps are detected by terrain matching methods. The terrain matching method will find the estimated difference between positions of sub-maps 1 and 2 which can minimise the residual error in the overlap between these two sub-maps. For the leading vehicle, the residual error  $E$  in the overlap between a pair of sub-maps that are both measured by the leading vehicle



**Figure 5.** An example of sub-map compression and reconstruction: (a) sub-map built and compressed in vehicle 2, (b) gridded sub-map reconstructed in the leading vehicle.

is given by

$$E = \begin{cases} \frac{1}{N_o} \sum_{n=1}^{N_o} \|h_i^-[n] - h_j[n]\|_{\sigma^2} & N_o > N_T \\ \text{inf} & N_o \leq N_T \end{cases}, \tag{7}$$

where  $h_j[n]$  and  $\sigma^2$  have been defined in Equation (1),  $N_o$  is the number of measured points in the overlap area between two sub-maps, and  $h_i^-[n]$  describes the depth at the same place as  $h_j[n]$  in sub-map  $j$ , which can be obtained using bilinear interpolation. If the number of measured points in the overlap is no more than a given threshold  $N_T$ , the residual error will be set as the infinite value. The value of  $N_T$  is usually calculated according to the rule of thumb. Increasing the value of  $N_T$  will improve the reliabilities of loop closures, but also reduce the number of loop closures. Considering that the measurement variances of measured points are the same in all sub-maps, Equation (7) can be simplified as

$$E = \begin{cases} \frac{1}{N_o} \sum_{n=1}^{N_o} \|h_i^-[n] - h_j[n]\|_2 & N_o > N_T \\ \text{inf} & N_o \leq N_T \end{cases}. \tag{8}$$

The residual error  $E'$  between a sub-map measured by the leading vehicle and a reconstructed sub-map is

$$E = \begin{cases} \frac{1}{N_o} \sum_{k=1}^{N_o} \|h'_i[k] - h_j^-[k]\|_{\Omega_k} & N_o > N_T \\ \text{inf} & N_o \leq N_T \end{cases}. \tag{9}$$

Different from Equation (8),  $h_j^-[k]$  denotes the terrain depth at the same place as  $h'_i[k]$  in sub-map  $i$  because the variance  $\Omega_k$  of  $h'_i[k]$  can be easily computed using Equation (6).

To adjust residual errors calculated using both Equations (8) and (9) to a notionally common scale, Equation (9) can be simplified to Equation (10), thus

$$E = \begin{cases} \frac{1}{N_o} \sum_{k=1}^{N_o} \exp(-\Omega_k) \|h'_i[k] - h_j^-[k]\|_2 & N_o > N_T \\ \text{inf} & N_o \leq N_T \end{cases}. \tag{10}$$

A loop closure between a pair of sub-maps will be detected if the minimal residual error calculated using Equation (8) or (10) in the overlap between a pair of sub-maps is less than a threshold. All the point cloud registration methods, including the template matching method and various ICP methods, can provide an accurate terrain matching result. Marchel et al. (2020b) proposed a modified ICP algorithm



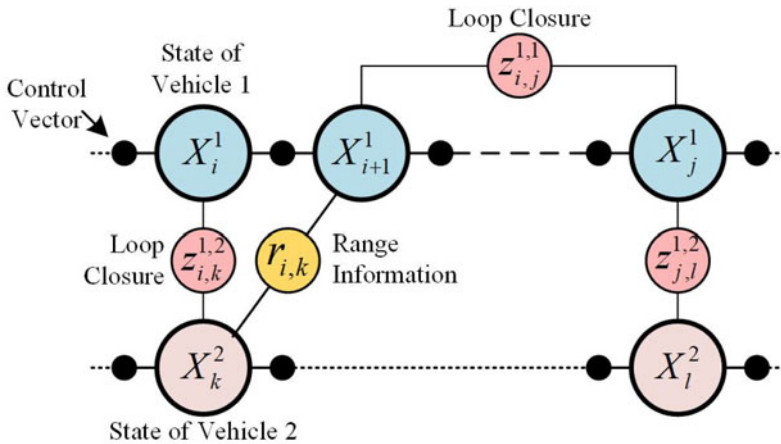


Figure 6. Pose graph of the cooperative bathymetric SLAM system.

that calculated the similarity between a pair of measurements using multimodal weighting factors based on the error modelling. In this paper, the template matching method is applied in the terrain matching process. Detailed explanation of the usage of template matching method in the terrain matching process is also drawn in Ma et al. (2020a).

### 5. Cooperative SLAM with graph method

In this paper, trajectories of both vehicles are estimated using a graph method that relies on a two-stage architecture, namely, the front-end and back-end stages. In the front-end stage, a pose graph is constructed to represent all historical bathymetric measurements and trajectories of both vehicles in the cooperative bathymetric SLAM.

As shown in Figure 6, the pose graph of the cooperative bathymetric SLAM system consists of nodes and edges, in which nodes represent states of both vehicles and the edge connected by two nodes encodes loop closures, range information between two vehicles, and control vectors from a DR system. Due to both model and measurement noise, erroneous edges may appear in the pose graph. Using so-called pose-graph optimisation, which maximises the consistency of all edges in the back-end stage, the optimal configuration of all vehicle states (also named as a trajectory of the vehicle) can be generated.

In the back-end stage, the estimated trajectories  $\mathbf{X}^{*1}$  and  $\mathbf{X}^{*2}$  of both vehicles can be generated by solving

$$\begin{aligned}
 \mathbf{X}^{*1}, \mathbf{X}^{*2} = \arg \min_{\mathbf{X}^{*1}, \mathbf{X}^{*2}} & \left\{ \sum_{i=1}^{N_1-1} \|f(X_i^1, u_i^1) - X_{i+1}^1\|_Q + \sum_{i=1}^{N_2-1} \|f(X_i^2, u_i^2) - X_{i+1}^2\|_Q \right. \\
 & + \sum_{(j,k) \in L_1} \|X_j^1 - X_k^2 - z_{j,k}^{1,1}\|_{\Lambda_{j,k}^{1,1}} + \sum_{(j,k) \in L_2} \|X_j^1 - X_k^2 - z_{j,k}^{1,2}\|_{\Lambda_{j,k}^{1,2}} \\
 & \left. + \sum_{(j,k) \in L_3} \|X_j^1 - X_k^2 - r_{j,k}\|_{\Psi} \right\}, \tag{11}
 \end{aligned}$$

in which  $f(X_i^1, u_i^1)$  and  $f(X_i^2, u_i^2)$  are state transition functions of both vehicles associated with control vectors  $u_i^1$  and  $u_i^2$ , with a model noise  $N(0, Q)$ ;  $L_3$  is the set of pairs of times connected by range information, in which  $(j, k) \in L_3$  denotes that the state  $X_j^1$  of the leading vehicle and the state  $X_k^2$  of vehicle 2 are connected by the range information  $r_{j,k}$  with variance  $\Psi$  of the measurement noise.

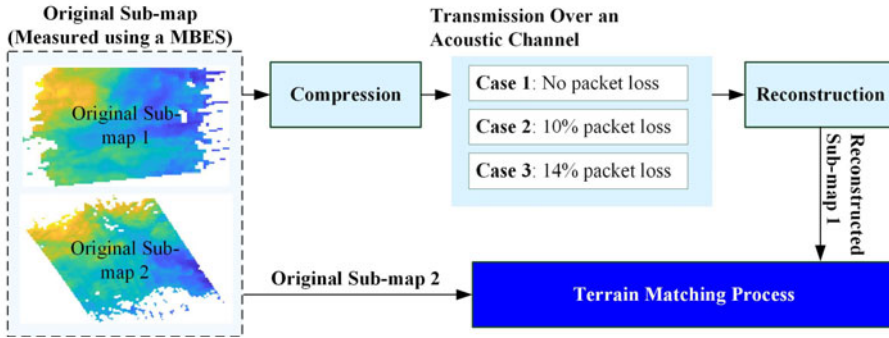


Figure 7. Simulation setups.

In the above,  $L_1$  and  $L_2$  both represent the sets of pairs of times connected by loop closures, where  $(j, k) \in L_1$  represents the states  $X_j^1$  and  $X_k^1$  of the leading vehicle which are connected by the loop closure  $z_{j,k}^{1,1}$  with variance  $\Lambda_{j,k}^{1,1}$ , and  $(j, k) \in L_2$  denotes the state  $X_j^1$  of the leading vehicle and the state  $X_k^2$  of vehicle 2 connected by the loop closure  $z_{j,k}^{1,2}$  with variance  $\Lambda_{j,k}^{1,2}$ . Although variances of position estimates and range information change during the mission, both  $Q$  and  $\Psi$  are considered as constants for simplification in this paper.

This paper focuses on cooperative navigation for two vehicles. The designed system can be easily extended to more vehicles, however, by adding nodes and edges associated with new vehicles in the pose graph which can also be solved using Equation (11).

## 6. Simulation results

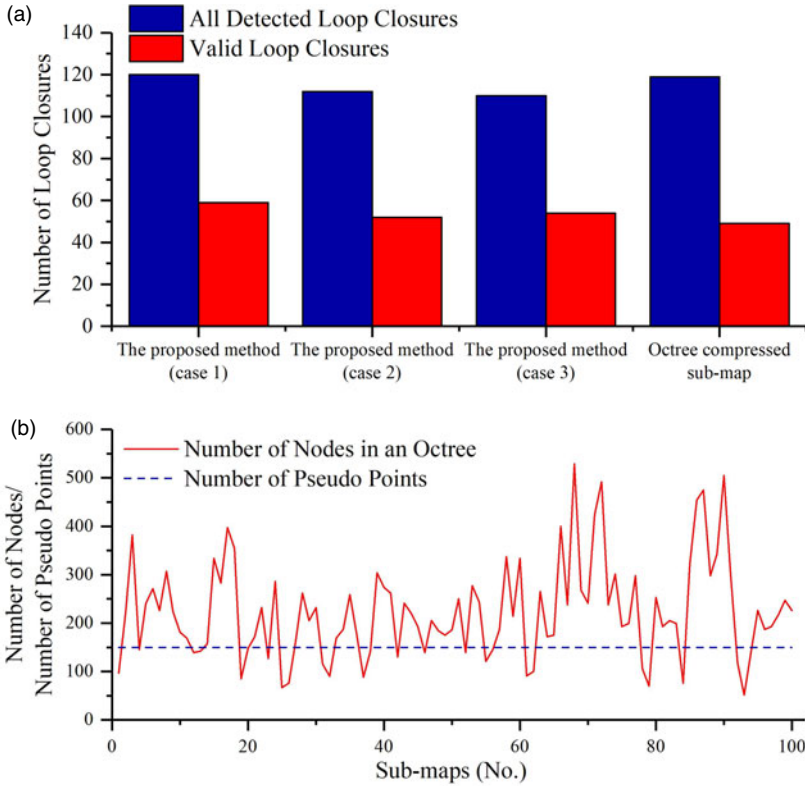
### 6.1. Validity of the bathymetric data transmission method

As shown in Figure 7, in the simulation, a loop closure is detected between a pair of sub-maps, in which the first sub-map (sub-map 1) is a reconstructed sub-map and the latter (sub-map 2) is an original one measured using MBES. All sub-maps used in the simulation were collected using shipborne MBES in the at-sea experiments. More specifically, the seabed terrain around Zhongsha Reef, China, was measured using the CMBS200 MBES, which emits 192 beams uniformly across a 120-degree swath with a frequency of 4 Hz. Three cases (no packet loss, 10% packet loss, and 14% packet loss) were applied to simulate the packet loss of acoustic packets caused by errors in data transmission over an unreliable acoustic channel.

In the simulation, the number of sub-maps used is 200. The threshold of residual error  $E$  is 0.02 and  $N_T$  is set as 120. For comparison, the sub-map 1 in Figure 7 was also compressed using the octree method. The octree splits a block if the maximum depth of the block elements minus the minimum depth of the block elements is greater than 1.25 m, making the number of nodes in the octree built using sub-maps from set A around 200. The maximum and minimum sizes of a block in the octree are 2 m and 32 m.

As shown in Figure 8(a), in case 1 (no packet loss), 120 loop closures (where 49.17% of 120 loop closures are valid) are detected between reconstructed sub-maps using this method and original sub-maps. The process of detection of loop closures between sub-maps compressed using the octree method and original sub-maps generates 119 loop closures, in which 49 out of 120 loop closures are valid. As shown in Figure 8(b), 63% of octrees built in the simulation have more than 200 nodes, leading to a bigger storage consumption than that of the proposed method. Thus, sub-maps compressed and reconstructed using the proposed method can provide more valid loop closures while reducing memory consumption.

Although the numbers of detected loop closures and valid loop closures decrease with an increase in the number of losing pseudo points, 46.43% and 49.09% of detected loop closures are valid in cases



**Figure 8.** Simulation results of loop closure detection: (a) number of detected loop closures, (b) storage consumption in octree and the proposed method.

2 and 3, respectively. This means that losing some pseudo points during the acoustic communication will not lead to more invalid loop closures, and the method can provide accurate loop closure detection results with an unreliable acoustic channel.

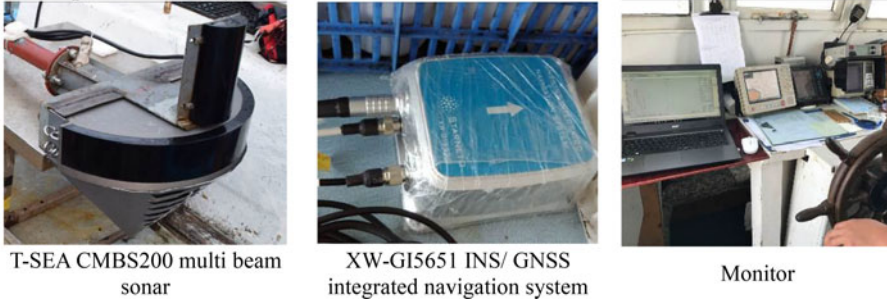
**6.2. Cooperative bathymetric SLAM**

The cooperative bathymetric SLAM simulation was conducted using the field data collected in the at-sea experiment around Zhongsha Reef. The equipment used is shown in Figure 9. The GPS trajectories of vehicles were obtained using two GNSS receivers of the XW-GI5651 INS/ GNSS integrated navigation system, installed in the longitudinal section of a vessel, with a distance of 4 m and delivering  $O(2\text{ m})$  position accuracy using precise point positioning method. The heading, pitch, and roll angles were measured using the XW-GI5651 INS/ GNSS integrated navigation system with angular accuracies of  $O(0.1\text{ deg})$ ,  $O(0.01\text{ deg})$ , and  $O(0.01\text{ deg})$ , respectively, with a frequency of 4 Hz. Bathymetric data were collected using a shipborne T-sea CMBS200 MBSE with a frequency of 4 Hz. The speed of the vehicle was about 4 kn and the DR trajectories of both vehicles were simulated using

$$\begin{aligned}
 X_i^{1DR} &= X_i^{1DR} + u_i^1 + v \\
 X_i^{2DR} &= X_i^{2DR} + u_i^2 + v.
 \end{aligned}
 \tag{12}$$

In Equation (12),  $X_i^{1DR}$  and  $X_i^{2DR}$  are positions of both vehicles estimated using a DR system at successive instants of time  $i$ , while  $X_i^{1GPS}$  and  $X_i^{2GPS}$  are corresponding GPS positions of vehicles. In

**B. Cooperative Bathymetric SLAM**



**Figure 9.** Equipment used in field experiments.

the above,  $u_i^1$  and  $u_i^2$  have been described in Equation (11), and  $v$  denotes the simulated process noise of the discrete-time model used to propagate vehicle state, with a distribution  $N(0.1, 0.04)$ .

The acoustic packets were broadcast by vehicle 1 with a frequency of 0.1 Hz. Packet loss of 14% of acoustic packets was assumed due to the noisy acoustic channel. Due to the nearly flat terrain, invalid loop closures identification is essential in bathymetric SLAM. In this paper, the MCM method discussed in Ma et al. (2020a) will be applied for invalid loop closures identification.

According to Alves (2019), the range information between positions  $X_i^1$  and  $X_k^2$  of two vehicles was simulated as

$$z_{i,k}^{1,2} = \Gamma_{i,k} \|X_i^{1GPS} - X_k^{2GPS}\|_2 + \tau_{i,k}, \tag{13}$$

where  $\tau_{i,k}$  denotes the measurement noise with distribution  $N(0, 1)$  and  $\Gamma_{i,k}$  can be given by

$$\Gamma_{i,k} = \begin{cases} 1, & \text{for a non outlier with probability } \varepsilon \\ \delta_r, & \text{for a outlier with probability } 1 - \varepsilon \end{cases} \tag{14}$$

In this paper, the value of  $\varepsilon$  was set as 0.2 and  $\delta_r$  was 1.1.

**Simulation 1**

The play-back experiment in simulation 1 was conducted using field data obtained in the course of two trajectories with an approximate duration of 40 min (vehicle 1) and 30 min (vehicle 2), respectively. After the trajectory of vehicle 2 is completed, vehicle 2 will turn off the MBSE and move randomly near its endpoint, but still maintain acoustic communication with vehicle 1. Here the cooperative bathymetric SLAM with an extremely high communication bandwidth (>2 Mb/s) and the cooperative bathymetric SLAM with octree method are chosen for comparison, in which the first means vehicles in this cooperative bathymetric system can share all bathymetric measurement in real time. This extremely high communication bandwidth is impossible in practice, but the navigation accuracy of the proposed algorithm can also be shown by comparison with this system.

It should be noted that the position of vehicle 2 will be continuously estimated until vehicle 1 reaches its endpoint though the planned trajectory of vehicle 2 has been completed after nearly the 30th minute of the mission. As shown in Figure 10, for vehicle 1, the mean location error of the proposed cooperative bathymetric SLAM system is 19.76 m, which is 82.02% less than that of the DR system, and the corresponding values for vehicle 2 are 22.27 m and 82.31%. Mean location errors of vehicles 1 and 2 in the cooperative bathymetric SLAM without communication limitation are 21.05 m and 22.40 m, and the corresponding values for cooperative bathymetric SLAM with octree method are 23.69 m and 26.64 m, respectively. Over the low bandwidth, high latency, and unreliable acoustic communication, the proposed system has a similar real-time navigation error as the cooperative bathymetric SLAM without communication limitation. Due to the 14% acoustic packet loss, cooperative bathymetric SLAM with

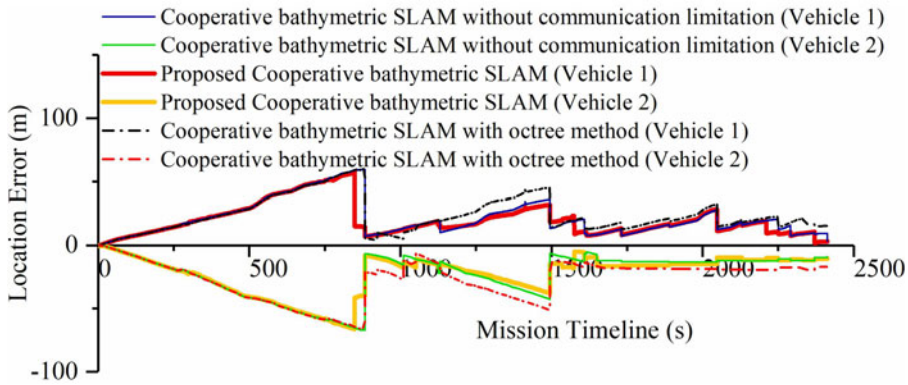


Figure 10. Location errors of both vehicles for increasing mission durations in simulation 1.

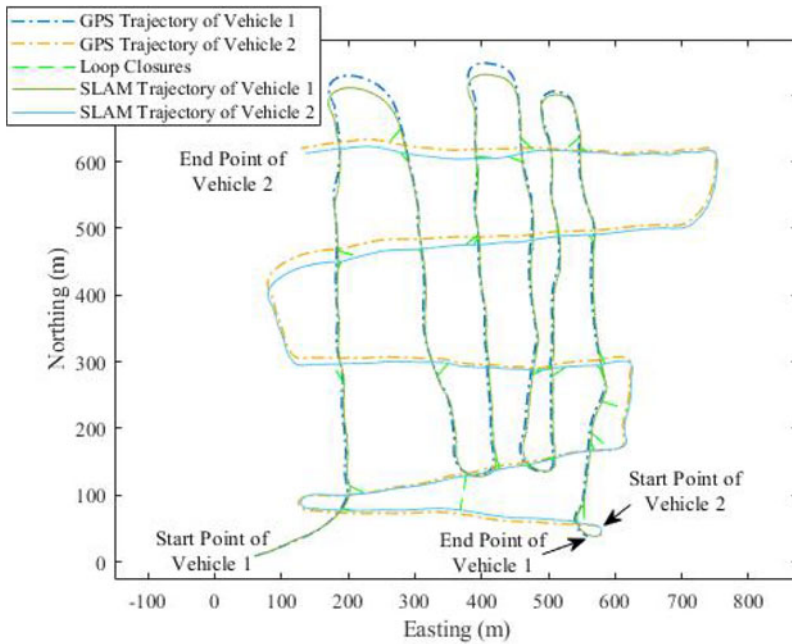
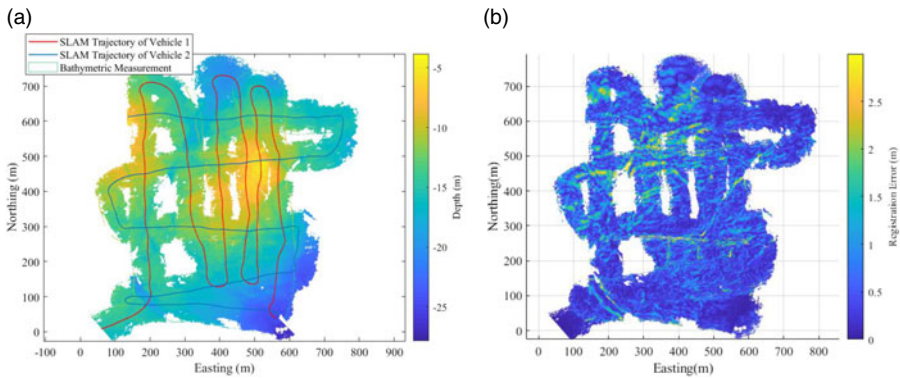


Figure 11. Trajectories of both vehicles during a complete mission in simulation 1.

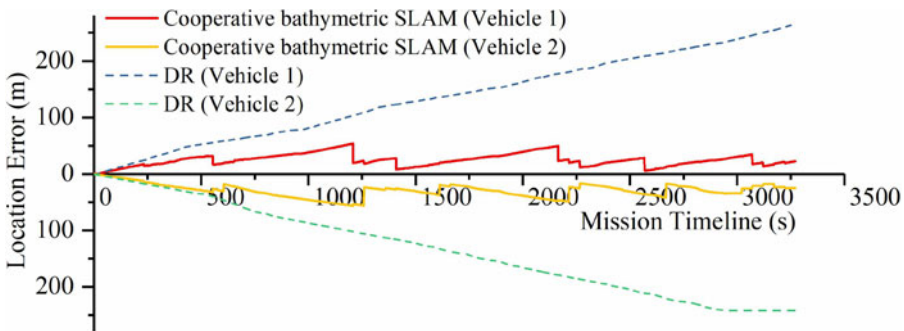
octree method lost some sub-maps over the limited communication channel, resulting in the largest navigation error among the three algorithms.

At the end of a complete mission, the location errors of both vehicles associated with the proposed cooperative bathymetric SLAM are 3.41 m and 10.06 m, respectively, while the final location errors generated by the cooperative bathymetric SLAM without communication limitation are 3.34 m and 9.17 m. The simulation proved that the proposed cooperative bathymetric SLAM achieves a navigation precision similar to the cooperative bathymetric SLAM without communication limitation and yields accurate navigation results with low bandwidth and unreliable acoustic communication.

Figure 11 shows the trajectories of both vehicles associated with the cooperative bathymetric SLAM system and GPS during a complete mission. During a complete mission, 46 valid loop closures are detected. The cooperative bathymetric SLAM system shares similar trajectory estimates of both vehicles with GPS. The mapping result of the cooperative bathymetric SLAM system is shown in Figure 12.



**Figure 12.** Mapping results during a complete mission in simulation 1: (a) bathymetric map, (b) registration error distribution.



**Figure 13.** Location errors of both vehicles for increasing mission durations in simulation 2.

The GPS positions of a vehicle are considered as to the ground truth positions, so the registration errors can be computed according to the difference between the map built using a cooperative bathymetric SLAM system and that associated with GPS. According to the simulation results shown in Figure 12, the mean and median values of registration errors over the whole bathymetric map built by the cooperative bathymetric SLAM system are 0.47 m and 0.33 m, while 88.06% of measured points have a registration error less than 1 m. The proposed algorithm can construct accurate and consistent seabed maps.

**Simulation 2**

Simulation 2 was conducted using field data including two trajectories with an approximate duration of 55 min (vehicle 1) and 50 min (vehicle 2), respectively. Like simulation 1, vehicle 2 will also turn off the MBES but broadcast acoustic packets which only include its trajectory and the timestamp continuously.

The same as the location results of vehicles in simulation 1, cooperative bathymetric SLAM system yields accurate navigation results, and during the mission, the location errors for both vehicles are bounded. As shown in Figure 13, the biggest and average location errors of vehicle 1 for increasing mission durations are 53.80 m and 24.50 m, respectively, which are much smaller than 264.9 m and 135.67 m of the DR system. Corresponding errors associated with the cooperative bathymetric SLAM system in vehicle 2 are 56.55 m and 29.26 m.

Figures 14 and 15 illustrate the trajectories and the mapping result associated with cooperative bathymetric SLAM during a complete mission. The bathymetric map in Figure 15 has a mean registration error of 0.54 m and a median registration error of 0.41 m, meanwhile, nearly 84.95% of terrain depths over the whole map have a registration error less than 1 m.

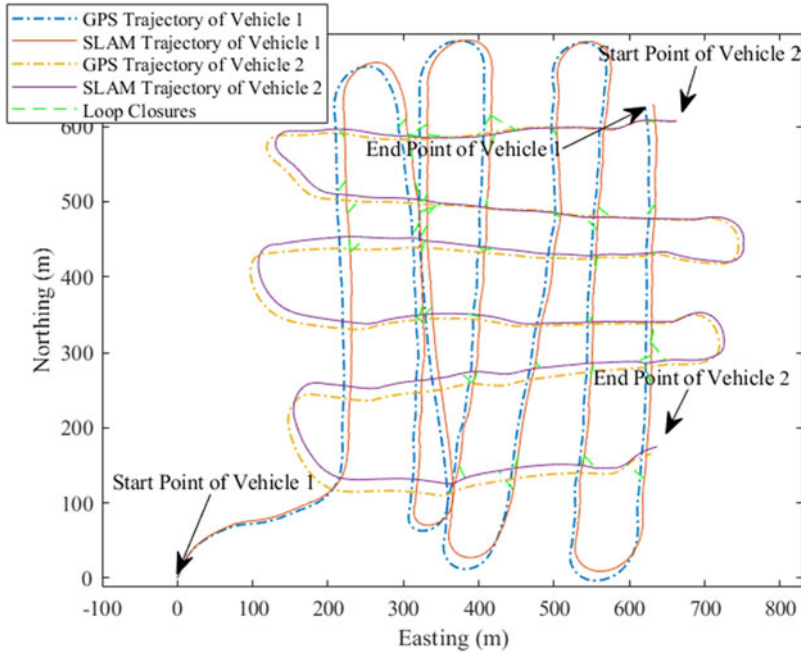


Figure 14. Trajectories of both vehicles during a complete mission in simulation 2.

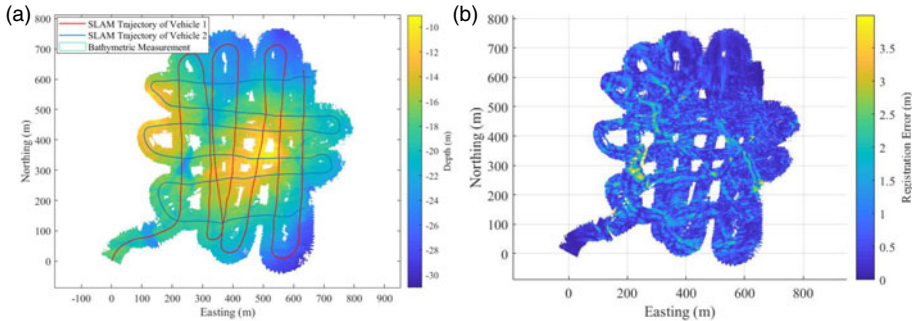


Figure 15. Mapping results during a complete mission in simulation 2: (a) bathymetric map, (b) registration error distribution.

According to results in both simulations, over the low bandwidth, high latency, and unreliable acoustic communication, the proposed cooperative bathymetric SLAM system can still yield accurate navigation results for the vehicle, and the corresponding absolute trajectory error (ATE) value is no more than 20% of a DR system. Meanwhile, in the bathymetric map constructed by the proposed cooperative bathymetric SLAM system, the registration error of nearly 85% of the measured points is less than 1 m, which means the proposed method constructs high-resolution and consistent bathymetric maps.

7. Conclusion

In this paper, a communication-constrained cooperative bathymetric SLAM algorithm for multi-AUVs is proposed. The following conclusions can be drawn:

1. The proposed bathymetric data transmission method can help to share a great number of bathymetric measurements among multiple vehicles over a low bandwidth acoustic channel.

2. Under a 10%~14% packet loss, the sub-maps transmitted can still yield many loop closures in which half of them are valid.
3. The cooperative bathymetric SLAM system can yield accurate navigation results in real time for multiple vehicles and construct consistent seabed maps over low bandwidth, noisy, and unreliable acoustic communication.

Due to the lack of the degree of freedoms on the range information between vehicles, the difference between the positions of two vehicles may be unobservable during the mission. In this paper, the observability of the range information is ignored because of the supplement of a large number of loop closures detected during the complete mission. However, a small number of valid loop closures will be detected when the cooperative bathymetric SLAM system is applied to navigate vehicles above the flat seabed terrain, making it essential to analyse the observability of the cooperative bathymetric SLAM. Future work will focus on the observability analysis and corresponding path planning for the cooperative bathymetric SLAM system.

**Acknowledgements.** This work was supported by the National Natural Science Foundation of China [Nos. 52001093, U1806228, 51909245, 51879057], and Open fund of Laboratory of Science and Technology on Marine Navigation and Control [No. KLMNCT-2021010107]. Hainan Provincial Joint Project of Sanya Yazhou Bay Science and Technology City [No. 520LH006].

## References

- Alves, B. A. da S. (2019). *A navigation system for under-the-ice robotic operations*. Master of Science thesis. University of Lisbon, Portugal.
- Barkby, S., Williams, S. B., Pizarro, O. and Jakuba, M. V. (2012). Bathymetric particle filter SLAM using trajectory maps. *The International Journal of Robotics Research*, **31**(12), 1409–1430.
- Bonin-Font, F. and Burguera, A. (2020). Towards multi-robot visual graph-SLAM for autonomous marine vehicles. *Journal of Marine Science and Engineering*, **8**(6), 437.
- Brown, H., Kim, A. and Eustice, R. M. (2008). Development of a Multi-AUV SLAM Testbed at the University of Michigan. *OCEANS 2008*, Quebec City, QC, Canada. IEEE, 1–6. DOI: 10.1109/OCEANS.2008.5151880
- Casalino, A. T. G., Simetti, E., Sperindè, A. and Torelli, S. (2014). Impact of LBL calibration on the accuracy of underwater localization. *IFAC Proceedings Volumes*, **47**(3), 3376–3381.
- Choudhary, S., Carlone, L., Nieto, C., Rogers, J., Christensen, H. I. and Dellaert, F. (2017). Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models. *The International Journal of Robotics Research*, **36**(12), 1286–1311.
- Demim, F., Nemra, A., Louadj, K., Hamerlain, M. and Bazoula, A. (2018). An adaptive SVSF-SLAM algorithm to improve the success and solving the UGVs cooperation problem. *Journal of Experimental & Theoretical Artificial Intelligence*, **30**(3), 389–414.
- Deschaud, J. E. (2018). IMLS-SLAM: Scan-to-Model Matching Based on 3D Data. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 2480–2485.
- Jang, J and Kim, J. (2019). Dynamic Grid Adaptation for Panel-Based Bathymetric SLAM. *2019 IEEE Underwater Technology (UT)*, IEEE, 1–4.
- Kim, A. and Eustice, R. M. (2013). Real-time visual SLAM for autonomous underwater hull inspection using visual saliency. *IEEE Transactions on Robotics*, **29**(3), 719–733.
- Kim, J. and Jung, H. S. (2011). An Approach Towards Online Bathymetric SLAM. *OCEANS'11 MTS/IEEE KONA*, IEEE, 1–6. DOI: 10.23919/OCEANS.2011.6106951.
- López, E., Garcia, S., Barea, R., Bergasa, L. M., Molinos, E. J., Arroyo, R., Romera, E. and Pardo, S. (2017). A multi-sensorial simultaneous localization and mapping (SLAM) system for low-cost micro aerial vehicles in GPS-denied environments. *Sensors*, **17**(4), 802.
- Lu, F. and Milios, E. (1997). Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, **4**(4), 333–349.
- Ma, T., Li, Y., Zhao Y., Jiang, Y. and Pascoal, A. M. (2020a). Efficient bathymetric SLAM with invalid loop closure identification. *IEEE/ASME Transactions on Mechatronics*, **26**(5), 2570–2580.
- Ma, T., Li, Y., Zhao, Y. X., Zhang, Q. and Zhang, Q. Y. (2020b). Robust bathymetric SLAM algorithm considering invalid loop closures. *Applied Ocean Research*, **102**, 102298.
- Mandić, F., Rendulić, I., Mišković, N. and Nad, Đ. (2016). Underwater object tracking using sonar and USBL measurements. *Journal of Sensors*, 8070286. doi:10.1155/2016/8070286
- Mangelson, J. G., Vasudevan, R. and Eustice, R. M. (2018). Communication Constrained Trajectory Alignment for Multi-Agent Inspection via Linear Programming. *OCEANS 2018 MTS/IEEE Charleston*, SC, USA. IEEE, 1–8. DOI: 10.1109/OCEANS.2018.8604775.



- Marchel, L., Naus, K. and Specht, M.** (2020a). Optimisation of the position of navigational aids for the purposes of SLAM technology for accuracy of vessel positioning. *Journal of Navigation*, **73**(2), 282–295.
- Marchel, L., Specht, C., and Specht, M.** (2020b). Testing the accuracy of the modified ICP algorithm with multimodal weighting factors. *Energies*, **13**, 5939.
- Massot-Campos, M., Oliver-Codina, G. and Thornton, B.** (2019). Laser Stripe Bathymetry Using Particle Filter SLAM. *OCEANS 2019*, Marseille, France. IEEE, 1–7.
- Norgren, P. and Skjetne, R.** (2018). A multibeam-based SLAM algorithm for iceberg mapping using AUVs. *IEEE Access*, **6**, 26318–26337.
- Ouyang, X., Zeng, F., Lv, D., Tianbao, D. and Haibing, W.** (2020). Cooperative navigation of UAVs in GNSS-denied area with colored RSSI measurements. *IEEE Sensors Journal*, **21**(2), 2194–2210.
- Palomer, A., Ridao, P., Ribas, D., Mallios, A. and Vallicrosa, G.** (2014). Octree-Based Subsampling Criteria for Bathymetric SLAM. *XXXV Jornadas de Automática*, September 2014, Valencia, Spain, 3–5.
- Palomer, A., Ridao, P. and Ribas, D.** (2016). Multibeam 3D underwater SLAM with probabilistic registration. *Sensors*, **16**(4), 560.
- Paull, L., Saeedi, S., Seto, M. and Li, H.** (2013). AUV navigation and localization: A review. *IEEE Journal of Oceanic Engineering*, **39**(1), 131–149.
- Paull, L., Huang, G., Seto, M. and Leonard, J. J.** (2015). Communication-Constrained Multi-AUV Cooperative SLAM. *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 509–516.
- Torroba, I., Bore, N. and Folkesson, J.** (2019). Towards Autonomous Industrial-Scale Bathymetric Surveying. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Macau, China, 6377–6382.
- Torroba, I., Sprague, C. I., Bore, N. and Folkesson, J.** (2020). PointNetKL: Deep inference for GICP covariance estimation in bathymetric SLAM. *IEEE Robotics and Automation Letters*, **5**(3), 4078–4085.
- Trujillo, J. C., Munguia, R., Guerra, E. and Grau, A.** (2018). Cooperative monocular-based SLAM for multi-UAV systems in GPS-denied environments. *Sensors*, **18**(5), 1351.
- Yuan, X., Martínezortega, J. F., Jas, F. and Eckert, M.** (2017). AEKF-SLAM: A new algorithm for robotic underwater navigation. *Sensors*, **17**(5), 1174.