Towards Under-ice Sensing using a Portable ROV

Lin Zhao
Department of Ocean Engineering
University of Rhode Island
Narragansett, RI, USA
linzhao@uri.edu

Mingxi Zhou
Graduate School of Oceanography
University of Rhode Island
Narragansett, RI, USA
mzhou@uri.edu

Brice Loose
Graduate School of Oceanography
University of Rhode Island
Narragansett, RI, USA
bloose@uri.edu

***Abstract*—Unmanned Underwater Vehicles (UUVs) have a promising future to explore the polar regions. In this paper, we present our progress on developing a self-contain inertial odometry for under-ice navigation. Firstly, a microcontroller-based hardware time synchronization for multiple devices is demonstrated. Moreover, we present a new IMU, Doppler Velocity Log (DVL) and Pressure dead-reckoning (DR) for state estimation and a robust initialization approach for underwater vehciels. Field trials have been conducted in Utqiagvik, Alaska in March 2022 to gather multi-sensor data under the sea ice. In this paper, we highlight the performance of our method by comparing to the *robot_localization* algorithm, a widely used open-source localization algorithm.**

***Index Terms*—Remotely Operated Vehicle (ROV), Under-ice Navigation, Marine Robotics, Time Synchronization**

I. Introduction

Arctic environment is under drastic changes [1], and the sea ice affects the global biogeochemical cycle [2]. Our understanding of biogeochemical processes such as air bubbles [3], basal ice melting [4], and the drivers for sea-ice algal bloom [5] is still limited due to the lack of robust technology for under-ice sensing. In recent years, UUVs have been proposed as a potential candidate to collect under-ice measurements [6]. Thanks to the modern sensors and sophisticated system design, the UUVs have been used for exploring the limited-access areas beneath the ice shelf, sea-ice and ice floes. One significance advantage of using UUVs is that they could produce a higher spatial coverage compared to ice-anchored instruments and ice coring [7]–[9].

However, operation logistics (deployment and recovery) is complicated when working with sea-ice for large UUVs (such as the Icefin [9] and ARTEMIS [10]) since a relatively large ice-tent and lifting mechanism are required. Nowadays, small low-cost UUVs have been deployed in various underwater applications like coral reef exploration [11] and terrain reconstruction [12]. Since most of these UUVs are originally designed for open-water operation, significant modifications are needed for under-ice operation [13].

Underwater state estimation is particularly challenging due to the lack of GPS [14]. Traditionally, UUV localization relies on expensive inertial navigation system (INS) and deployed acoustic transponders. Recent advancements in electronics has significantly increased the accessibility of compact and affordable navigation sensors, such as Micro-Electro-Mechanical System (MEMS) IMU, sonar and optical camera. To enable robust perception and navigation for these challenging environments, the methods of using multi-sensors fusion [15] [16] has shown promising outcomes. One critical task for sensor fusion is time synchronization since the time difference between sensor modalities will cause unreliable navigation results [17].

In this paper, we will report our progress in using a portable ROV for under-ice sensing, and demonstrate the feasibility of using small a ROV (0.7m long and 0.5m wide) for sampling the under-ice environment. Our contributions are summarized as follows:

- System modification and low-cost hardware time synchronization for the portable ROV. To our knowledge, this is the first work introduced hardware time synchronization for various devices in the underwater domain.
- Tightly-coupled IMU-DVL-Pressure odometry is proposed with derived on-manifold update equations.
- Validated the method by conducting under-ice missions in polar regions.

The remaining paper is organized as follows. In Section II, we review relevant works in the under-ice operation, time synchronization and sensor-fusion with a focus on underwater environments. ROV system upgrade, including sensor integration and multi-sensor time synchronization, will be presented in Section III. The proposed new localization method is introduced in Section IV. Under-ice field deployment has been conducted with results shown in Section V, and we will discuss conclusion and lessons learned in Section VI.

II. Related Works

Since the 21st Century, UUV-based under-ice explorations have drawn increased attention because UUVs have increased accessibility, can lower personnel risks, and can cover a wider area compared to fixed instruments. Herein, several seminal works in under-ice UUVs are reviewed. In 2015, a twin-hull AUV, called SEABED [18], was deployed to obtain the 3D topography the underside of floating ice floes [19]. Autosub is another large-size AUV which is extensively used for under-ice exploration. For example, in 2018, the Autosub Long Range 6000 AUV was deployed 500 m deep to measure the ice shelf and seabed morphology [20]. More recently, seabed mapping

This work is supported by the National Science Foundation (NSF) under the award #1945924, and the Graduate School of Oceanography, University of Rhode Island. We also thank UIC Science LLC for the logistic support and Dr. Chris Roman for borrowing us the USBL transceiver.

and water column assessment missions were carried out under the Thwaite Glacier ice shelf using a Kongsberg HUGIN AUV [21]. Although larger size AUVs have better endurance and sensing capability, portable and affordable UUVs such as Gavia AUV [22], REMUS AUV [23], gliders [24], low-cost ROV [25] are also welcomed because they are easy to operate (without ice camps or icebreakers). However, most of those successfully deployed torpedo-shape AUVs were flying further away from the ice surface at a constant depth. In contrast, this paper presents our work on deploying a modified BlueROV-2 to survey within 2 meters from the ice, aiming to collect biogeochemical measurements at the ice-water interface.

Simultaneous Localization and Mapping (SLAM) is a popular field in robotics, which leverages multiple sensors to provide a better localization result compared to a dead-reckoning algorithm based on inertial measurements. In addition, multi-sensors SLAM has been increasingly applied to the underwater domain by fusing motion sensors (e.g. IMU, DVL, Pressure Sensor) and perception sensors (e.g. Camera, Bathymetric Sonar, Imaging Sonar). The authors in [26] integrated DVL, gyroscope and depth sensors into an EKF odometry that was fed as motion prior inside the ORB-SLAM2 [27] for robust sensing in the visual degraded environment. In [28], EKF-based bathymetric SLAM using a multibeam sonar, Attitude and Heading Reference System (AHRS), DVL, pressure sensor was proposed. The method performs DR using a kinematic model with measurement updates from motion sensors, then it was corrected using the multibeam sonar submap registration. Similar to the previous DR method, a Rao-Blackwellized particle filter (RBPF) SLAM using Forward Looking Sonar (FLS), DVL and IMU was presented in [29]. *SVIn2* [15] is a tightly-coupled nonlinear optimization-based method integrated with IMU, depth sensor, profiling sonar and stereo camera with the capacities of robust initialization, image enhancement and relocalization.

For multiple sensor fusion, precise time synchronization between devices is crucial [17]. To get a robust and accurate sensor fusion result, hardware level time synchronization is needed to eliminate clock offset due to communication bus buffer delay, and operating system (OS) scheduling. For an application that can access Global Positioning System (GPS) signal, pulse per second (PPS) was employed to synchronize onboard computer and sonar [30]. The time synchronization can be done either offline or online. For example, the VIRAL multiple modalities dataset [31] performance offline time offset calibration between camera and IMU using Kalibr [32], which requires specific routines. In contrast, the frameworks like OpenVINS [33] and VINS-Mono [34] are capable of online calibration to make operation easier. For the robots running in GPS-denied areas, e.g., underground, indoor environments and underwater, time synchronization is extremely important. During the DARPA Subterranean Challenge, Team CERBERUS utilized a microcontroller unit (MCU) for the visual–inertial triggering [35]. For the indoor applications, one used Single Board Computer (SBC) for conducting hardware synchronization of different types of sensors [36] including triggerable devices (e.g. IMU and Cameras) and GPS-stamp devices (e.g. Lidars) that only accept National Marine Electronics Association (NMEA) string and PPS. Another example is VersaVIS [17], which not only used MCU for triggering the visual-inertial system but also applied an EKF to estimate the time offset between MCU and host. Because non-triggerable devices like wheel odometers only produce their timestamps based on host clock. Furthermore, sensor time synchronization is also challenging for marine robotics. Having time synchronized could be a great feature for datasets. An underwater caves dataset [37] focusing on navigation perspective containing multi-sensors is appealing, but no time synchronization was provided. The AURORA dataset [38] from the famous and reliable Autosub6000 AUV was time synchronized, however, no detailed information is reported in the paper. An underwater reconstruction dataset [39] applied hardware synchronization but only for stereo cameras.

III. Under-ice ROV Upgrade

A. System Integration

A photo of the upgraded BlueROV-2 is shown in Fig. 1, and the system diagram is displayed in Fig. 2. As shown in Fig. 2, two lithium-ion battery packs (total of 500 Wh) were enclosed in a single pressure housing. The battery packs can be charged via a underwater connector without disassemble. A Battery Management Board (BMB) was used to balance the load on the battery packs, and a Power Monitor Board (PMB) was integrated to track the current and voltage of the entire system. Two DC-DC converters were integrated to provide $5V$ and $24V$ for the instruments.

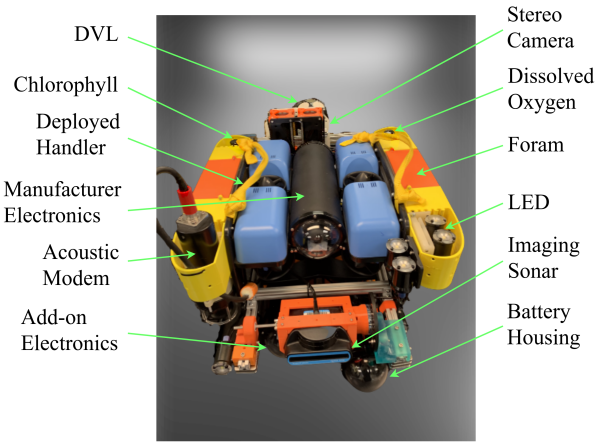

Fig. 1. The under-ice ROV modified from the BlueROV-2 and our prior work [40]

The electronics in the low-level housing are remain unchanged from the manufacturer, but we have customized the electronics in the high-level housing, including a Jetson NX Xavier SBC, a Arduino Zero MCU, an AHRS, an Internet switch and a tether board. The Robotic Operating System (ROS) was running on the Jetson to performance sensor interface and data processing. The Arduino handles the time

synchronization among Microstrain AHRS, FLIR stereo camera, Bluerobotics LED, Nortek DVL, the Jetson and the topside laptop. All the onboard data was streamed to the topside using a Bluerobotics tether board.

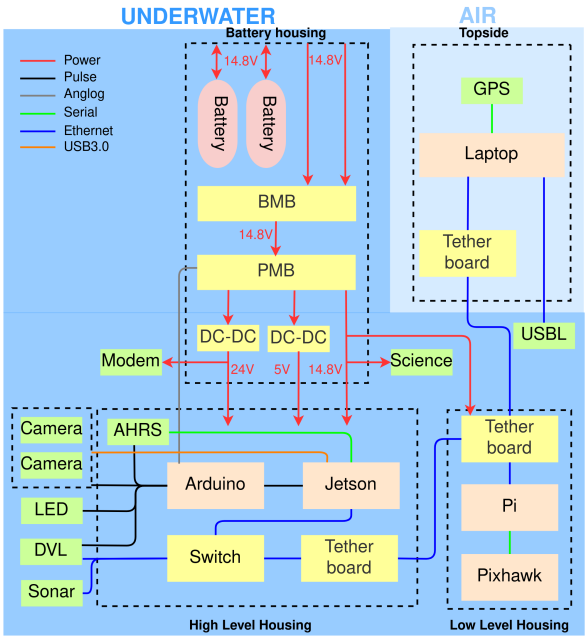

Fig. 2. The ROV system diagram

An Evologics 18/34 Acoustic Modem (shown in Fig. 1) was installed to the vehicle to obtain georeferenced location fix while the ROV is under the water. To characterize the biogeochemical activities in the water column, a dissolved oxygen optode and a chlorophyll sensor are integrated.

On the topside laptop, we have data (including sonar image, camera images, and vehicle odometry) visualized to assist ROV piloting. Meanwhile, the Evologics USBL that was deployed in an ice hole provides online positioning result in the user interface (SiNAPS [1]). To account any clock drift on the ROV and topside lapto, a GPS was attached to the laptop and the data was logged simultaneously for aligning ROV data to the global time.

B. Time Synchronization

Our time synchronization solution is created using a low-cost MCU to support various devices as shown in Fig. 3. Each device synchronization, besides the non-triggerable one, can be separated into two parts, time keeping and trigger in the Arduino and timestamp management in Jetson. The stereo camera is configured in the primary-secondary mode that the primary camera will be externally triggered while the secondary camera will be triggered by the exposure signal from the primary camera. In Arduino, a hardware timer is configured in Pulse Width Modulation (PWM) mode to generate jitter-free pulses to trigger the primary camera at $10Hz$. The MCU also sends the trigger information includes trigger timestamp and camera exposure time to Jetson through a serial port. In Jetson, a ROS driver handles the timestamp management in two stages, trigger initialization and timestamp replacement. In the initialization stage, the Jeston intends to find a constant sequence offset between the camera trigger time and the camera data time. In this stage, the MCU is configured to trigger the camera at $1Hz$. Meanwhile, the Jetson receives both trigger information (from the MCU) and image data (from the camera) to check the sequence offset between two messages until a consistent sequence offset is observed. In the replacement stage, trigger will be restore to the regular rate, i.e., 10 Hz. All the timestamps in the image data will then be replaced by the aligned trigger timestamp plus half of the camera exposure time. The primary and secondary cameras will be executed timestamp management using this ROS driver at same time in the same manner.

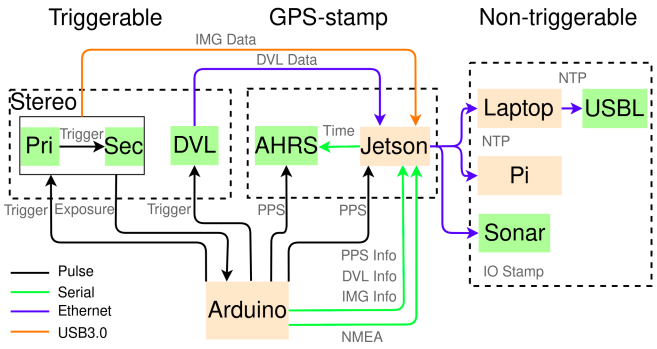

Fig. 3. Time synchornization diagram

The DVL trigger process is similar to the primary camera, but at $8Hz$. Both Bottom Track (BT) and Current Profiling (CP) were enabled, and we configured the DVL such that the CP is available after every 7 BT measurements. The timestamp management for the DVL is done slightly different than the camera, since BT and CP are coming sequentially from the same data stream, but the sequence are updated independently for the BT and CP messages. To successfully align the timestamp message and data message during the initialization, a unique DVL sequence is obtained by summing the latest BT sequence and latest CP sequence. After constant sequence offset between the DVL sequence and trigger timestamp sequence is obtained, same procedure used for the camera time replacement is applied (e.g. trigger restoration, time compensation and replacement of aligned timestamp).

To time synchronize the Jetson and the Arduino, we program the Arduino to generate PPS based on its system clock using a hardware timer. The PPS timestamp is constructed into an NMEA string and sent to the Universal Asynchronous Receiver-Transmitter (UART) port on Jetson. To precisely update time information of Jetson, we use a Network Time Protocol (NTP) implemented software, chrony, to synchronize the Jetson clock to a reference clock which is generated by the gpsd daemon using the MCU NMEA string and PPS.

Our onboard Microstrain AHRS only accepts the GPS-stamp method for time synchronization so the same PPS

[1]https://evologics.de/sinaps

signal from the Arduino is sent to AHRS. However, the PPS timestamp is constructed following a specific format that is required by the AHRS ROS driver for its internal clock synchronization.

The topside laptop and Raspberry Pi (the original Blue ROV computer) are connected with a tether board, they are both time synchronized with Jetson using the NTP. The sonar is not capable of hardware synchronization. Thus, the ros messages are directly stamped with the Input/Output (IO) time when the data has arrived at the host computer. The USBL is connected to laptop with an Ethernet cable and NTP was also used for time synchronization.

IV. Navigation System

The foundation of modern navigation solutions is DR [41], which is typically further fused with other sensors to bound the accumulated position drifts. The widely accepted DR method is to integrate the measurements from several motion sensors using an EKF. Some existing works use the vehicle system model for prediction update, while the Multi-State Constraint Kalman Filter (MSCKF) [42] multi-sensor fusion framework utilizes an IMU model for the state prediction and camera measurements for state update. In [43], a similar approach is presented where the measurements from the USBL, DVL and Pressure are used for update. However, they directly used DVL raw beam measurements instead of 3-axis velocity to avoid the adverse impacts from possible invalid velocities derived from four beams. They also have set the DVL and IMU in the same orientation, resulting in a simplified (but not general) DVL update equation without considering any transformation between this two sensors. Our work follows the same idea from MSCKF [42], but we further integrated the DVL and Pressure measurements to conduct tightly-coupled measurement updates and have included the possible transformation between DVL and Pressure sensor and the IMU. In our work, we only used the bottom track measurements (ROV's body frame linear velocity relative to the ice). Since our ROV was kept close to the ice (less than 2 meters), reliable bottom lock was obtained during the tests.

A. System State

The state vector follows the notation in [33] and is defined as:

$$\mathbf{x} = \begin{bmatrix} {}^{I_k}_{G}\bar{q}^\top & {}^{G}\mathbf{p}_{I_k}^\top & {}^{G}\mathbf{v}_{I_k}^\top & \mathbf{b}_g^\top & \mathbf{b}_a^\top \end{bmatrix}^\top \tag{1}$$

where ${}^{I_k}_{G}\bar{q}$ [44] is the unit quaternion represents the rotation from the global frame $\{G\}$ to the IMU frame $\{I_k\}$ at time k. ${}^{G}\mathbf{p}_{I_k}$ and ${}^{G}\mathbf{v}_{I_k}$ are the IMU position and velocity with respect to $\{G\}$. The vectors $\mathbf{b}_g$ and $\mathbf{b}_a$ describe the biases of the measured angular velocity and linear acceleration from the IMU respectively. We define $\mathbf{x} = \hat{\mathbf{x}} \boxplus \tilde{\mathbf{x}}$, where $\mathbf{x}$ is the true state, $\hat{\mathbf{x}}$ is estimation, $\tilde{\mathbf{x}}$ is the error state, and the $\boxplus$ operation maps the vector to a given manifold [45]. For quaternions, we define the quaternion boxplus operation using the left quaternion error as:

$$\bar{q} \boxplus \delta\boldsymbol{\theta} \triangleq \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta} \\ 1 \end{bmatrix} \otimes \bar{q} \tag{2}$$

B. IMU Propagation

The state is propagated forward based on IMU measurements of linear accelerations ${}^{I}\mathbf{a}_m$ and angular velocities ${}^{I}\boldsymbol{\omega}_m$ following generic nonlinear IMU kinematics model from timestep $k-1$ to k [46].

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, {}^{I}\mathbf{a}_m, {}^{I}\boldsymbol{\omega}_m, \mathbf{n}_I) \tag{3}$$

where $\mathbf{n}_I = \begin{bmatrix} \mathbf{n}_g^\top & \mathbf{n}_a^\top & \mathbf{n}_{\omega g}^\top & \mathbf{n}_{\omega a}^\top \end{bmatrix}^\top$, $\mathbf{n}_g$ and $\mathbf{n}_a$ are the zero-mean Gaussian noise of the gyroscope and accelerometer measurement, while $\mathbf{n}_{\omega g}$ and $\mathbf{n}_{\omega a}$ represent the random walk bias noise for gyroscope and accelerometer respectively. The estimated value and propagated covariance are:

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, {}^{I}\mathbf{a}_m, {}^{I}\boldsymbol{\omega}_m, \mathbf{0}) \tag{4}$$

$$\mathbf{P}_{k|k-1} = \boldsymbol{\Phi}_{k|k-1}\mathbf{P}_{k-1|k-1}\boldsymbol{\Phi}_{k|k-1}^\top + \mathbf{G}_{k-1}\mathbf{Q}\mathbf{G}_{k-1}^\top \tag{5}$$

where $\hat{\mathbf{x}}_{k|k-1}$ denotes the estimated value at time k given the measurements up to time $k-1$. $\boldsymbol{\Phi}_{k|k-1}$ and $\mathbf{G}_{k-1}$ are system Jacobian and noise Jacobian of the linearized system [42]. $\mathbf{Q}$ is a discrete-time covariance matrix of IMU noise $\mathbf{n}_I$.

C. DVL Update

The DVL velocity measurement [47] is defined as:

$$\begin{aligned} \mathbf{z}_{D,k} &= h_D(\mathbf{x}_k) + \mathbf{n}_D \\ &= {}^{I}_{D}\mathbf{R}^\top({}^{I_k}_{G}\mathbf{R}\,{}^{G}\mathbf{v}_{I_k} + \lfloor {}^{I_k}\boldsymbol{\omega} \rfloor_\times {}^{I}\mathbf{p}_D) + \mathbf{n}_D \end{aligned} \tag{6}$$

where ${}^{I}_{D}\mathbf{R}$ and ${}^{I}\mathbf{p}_D$ are the known DVL rotation and position with respect to IMU frame. ${}^{I_k}_{G}\mathbf{R}$ is the rotation from the global frame to the IMU frame at time k. We have the measurement noise $\mathbf{n}_D \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_D)$. $\lfloor . \rfloor_\times$ is a skew-symmetric operation that maps IMU angular velocity ${}^{I_k}\boldsymbol{\omega} = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^\top \in \mathbb{R}^3$ into:

$$\lfloor {}^{I_k}\boldsymbol{\omega} \rfloor_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{7}$$

We linearize Eq. (6) at the current state with respect to the current zero-mean error state for the EKF update.

$$\tilde{\mathbf{z}}_{D,k} = \mathbf{H}_{D,k}\tilde{\mathbf{x}}_{k|k-1} + \mathbf{n}_D \tag{8}$$

where $\mathbf{H}_{D,k} = \begin{bmatrix} H_{D,\mathbf{R}}^\top & \mathbf{0} & H_{D,\mathbf{v}}^\top & \mathbf{0} & \mathbf{0} \end{bmatrix}^\top$ is the measurement Jacobian, shown in Appendices. B, computed as follows:

$$H_{D,\mathbf{R}} = \frac{\partial h_D(.)}{\partial {}^{I_k}_{G}\tilde{\mathbf{R}}} = {}^{I}_{D}\mathbf{R}^\top \lfloor {}^{I_k}_{G}\mathbf{R}\,{}^{G}\mathbf{v}_{I_k} \rfloor_\times \tag{9a}$$

$$H_{D,\mathbf{v}} = \frac{\partial h_D(.)}{\partial {}^{G}\tilde{\mathbf{v}}_{I_k}} = {}^{I}_{D}\mathbf{R}^\top {}^{I_k}_{G}\mathbf{R} \tag{9b}$$

Then, we use this measurement model for EKF update with the following equations.

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} \boxplus \mathbf{K}_k(\mathbf{z}_{D,k} - h(\hat{\mathbf{x}}_{k|k-1})) \tag{10a}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_{D,k}\mathbf{P}_{k|k-1} \tag{10b}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_{D,k}^\top(\mathbf{H}_{D,k}\mathbf{P}_{k|k-1}\mathbf{H}_{D,k}^\top + \mathbf{R}_D)^{-1} \tag{10c}$$

D. Pressure Update

We model the p_z measurement of state ${}^G p_I$ as:

$$\begin{aligned} z_{p_z,k} &= h_{p_z}(\mathbf{x}_k) + n_P \\ &= \mathbf{s}_G^{I_k}\mathbf{R}^\top {}_D^I\mathbf{R}\, {}_P^D\mathbf{R}({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) + n_{p_z} \end{aligned} \tag{11}$$

where $\mathbf{s} = [\ 0 \quad 0 \quad 1\ ]$ used for selecting third dimension value, ${}^P\mathbf{P}_{in} = [\ 0 \quad 0 \quad {}^P p_{in}\ ]^\top$ and ${}^P p_{in}$ is the Pressure measurement at the initial position, ${}^P\mathbf{P}_k = [\ 0 \quad 0 \quad {}^P p_k\ ]^\top$ and ${}^P p_k$ is the Pressure measurement at timestamp k, ${}_P^D\mathbf{R}$ is the known rotation from Pressure frame to DVL frame and n_{p_z} is zero-mean white Gaussian noises. To perform EKF update, the linearized function is shown below:

$$\tilde{z}_{p_z,k} = \mathbf{H}_{p_z,k}\tilde{\mathbf{x}}_{k|k-1} + n_{p_z} \tag{12}$$

where $\mathbf{H}_{p_z,k} = \begin{bmatrix} H_{p_z,\mathbf{R}}^\top & H_{p_z,\mathbf{p}}^\top & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}^\top$ is the measurement Jacobian (check Appendices. C) list as:

$$H_{p_z,\mathbf{R}} = \frac{\partial h_{p_z}(.)}{\partial {}_G^{I_k}\tilde{\mathbf{R}}} = -\mathbf{s}_G^{I_k}\mathbf{R}^\top \lfloor {}_D^I\mathbf{R}\, {}_P^D\mathbf{R}({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k)\rfloor_\times \tag{13a}$$

$$H_{p_z,\mathbf{p}} = \frac{\partial h_{p_z}(.)}{\partial {}^G\tilde{\mathbf{p}}_{I_k}} = [\ 0 \quad 0 \quad 1\ ] \tag{13b}$$

E. System Initialization

The robust initialization method is critical for a good navigation result. Some platforms can start the mission in a stationary state, which is not always the case for marine robots starts the mission in water and impacted by the waves and currents. Therefore, in this section, we discuss our initialization procedures to extract the biases in gyro, acceleration, the gravity vector, and the initial velocity and pressure.

To initialize the system in a generic case without time synchronization, DVL and IMU timestamp are aligned by detecting measurement spikes caused by a sudden vehicle movement from its relatively static condition (e.g. drift with currents). The vehicle motion in a small duration before the alignment was assumed stationary so gyro bias was computed by averaging the gyro measurements. Similarly, the average of pressure measurements during this time is assigned as the initial pressure ${}^P p_{in}$.

We need to remove vehicle acceleration (e.g. caused by water flow and thrusters) and gravity effect from acceleration measurements to get bias. We first select a short duration (e.g., 2 seconds) after the alignment. Then, we estimate the vehicle acceleration from the velocity measured by the DVL. Since such DVL measurements are obtained when the vehicle is moving, the DVL has a relative low noise-to-signal ratio. By now, we can use the following equation [47] to estimate the vehicle acceleration.

$${}^I\hat{\mathbf{a}} = \lfloor {}^I\boldsymbol{\omega}\rfloor_\times {}_D^I\mathbf{R}\, {}^D\mathbf{v} + {}_D^I\mathbf{R}\, {}^D\mathbf{a} - (\lfloor {}^I\boldsymbol{\omega}\rfloor_\times^2 + \lfloor {}^I\dot{\boldsymbol{\omega}}\rfloor_\times) {}^I\mathbf{p}_D \tag{14}$$

where DVL velocity and acceleration are annotated as ${}^D\mathbf{v}$ and ${}^D\mathbf{a}$ respectively. Furthermore, estimated accelerations linearly interpolated at IMU timestamps can be used for compensation. Up to now, the gravity vector can be found easily since the vehicle acted as standing still. For each measurement, a rotation ${}_G^I\mathbf{R}$ was constructed to remove gravity acceleration. Finally, the acceleration bias was set to the average of the left measurements since vehicle acceleration and gravity were well compensated. The initial velocity ${}^G\mathbf{v}_I$ was estimated using Eq. (6) by rearranging the velocity to the left side.

V. Results and Analysis

A. Field Operation

Field tests were conducted in March 2022 in Utqiagvik, Alaska on a flat landfast ice located about several hundred meters off the coast. The ice thickness was about 1.5 meters. As shown in Fig. 4, ROV was lowered through a rectangle ice hole using straps with hooks at the end. The recovery was done by manually driving the ROV through the hole, then, the straps will hook onto the ROV for lifting.

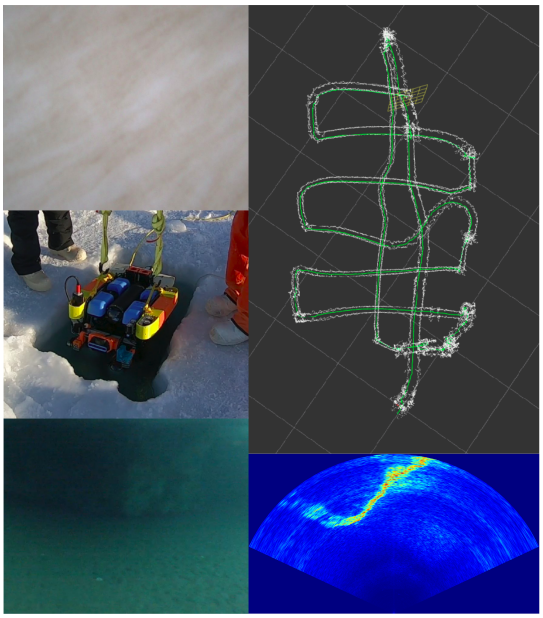

Fig. 4. Field operation: Top-left panel shows the under-ice camera image; Middle-left photo shows the ROV deployment; Bottom-left image presents the ice keel image captured from a forward-looking camera; Top-right panel depicts the ROV trajectory from the online odometry; Bottom-right panel displays the acoustic returns from the ice keel using a forward-looking imaging sonar.

The interval of the USBL position fix was configured at 5 seconds, while the AHRS was sensing at 200 HZ and the DVL was triggered at 8 Hz. We have conducted a total of 4 dives on four different days (each dive lasts about 30-40

minutes) to survey the water under the sea-ice in a lawnmower pattern, as shown in Fig. 4, where the the green path shows the vehicle path and the white point cloud are the sea-ice surface measured by the DVL. During the tests, we maintained the ROV at about 1-2 meters away from the bottom of the ice for imaging purpose. From the upward-looking camera (see Fig. 4), we observed directional striping features at the ice surface which is different from the homogeneous air bubble features observed in the freshwater ice [40]. Also, the ice keel was observed both from a forward-looking camera and sonar as displayed in Fig. 4.

B. Data Analysis

To quantify the performance of the proposed navigation method, we compared the resulting position with the USBL position fixes (ground truth) at the same global time. The USBL's raw data is filtered with internal parameters such as accuracy, received signal strength indicator (RSSI), and integraity to remove the noise data. Since our method do not use magnetometer because its bad performance in polar area, we have to align the estimated odometry with the USBL fixes. To this end, an initial section of the data points was selected and we applied the Umeyama Alignment [48] from the evaluation toolbox evo [49] to align the path and the USBL fixes.

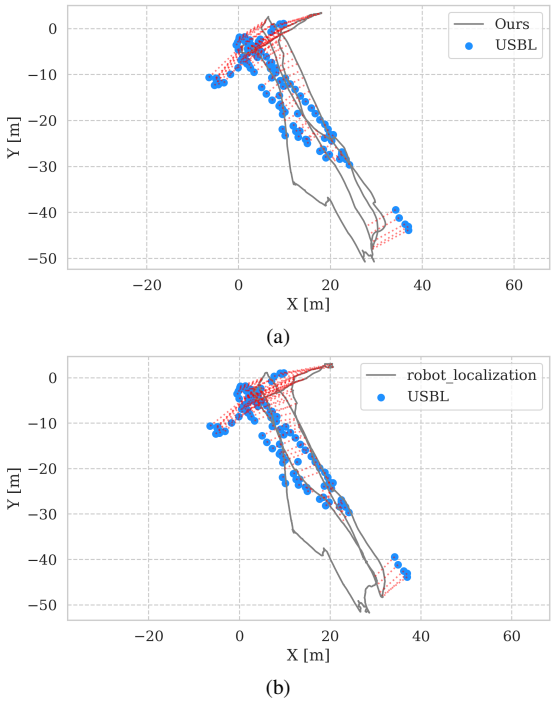

Fig. 5. A top view of the position fixes (blue dots) obtained by the USBL and the tracks (black tracks) estimated using the dead-reckoning algorithms. The associated points between the dead-reckoning and the USBL position is highlighted using the red dashlines.(a) USBL and our method; (b) USBL and *robot_localization*.

As shown in Fig. 5, our odometry implemented from Section IV and *robot_localization* odometry both fused DVL and IMU data compared with the position fixes obtained from the USBL. The correspondences (red dash-lines) show the Euclidean distance between the USBL position fixes and the estimated path at the same timestamp. In Fig. 6, we present the distance in x and y direction between two connected points, one from the USBL and one from the estimated odometry. In Fig. 6, we indicated the data points used for path alignment in red. As a result, the position differences are relatively small in this region. To compare the performance of our method and the *robot_localization*, we computed the Root Mean Square Error (RMSE) in the X-axis and Y-axis. As a result, our tightly-coupled DVL-IMU odometry produced a better performance than the *robot_localization*, lower RMSE and smaller maximum error. Noted that, our implementation of odometry did not use Eq. 9a and 13a for the orientation update because the performance became worse. The main reason may be the inaccurate extrinsic calibration between DVL and IMU.

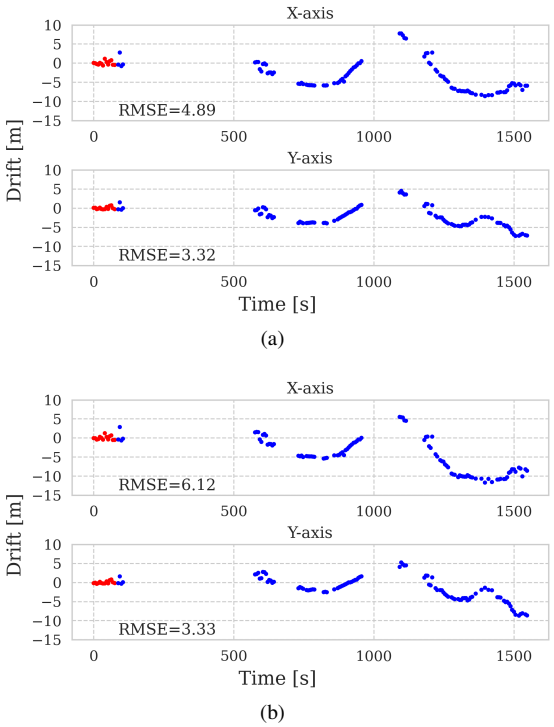

Fig. 6. Position error between the USBL position fixes and the estimated tracks in X-axis and Y-axis. (a) Ours; (b) *robot_localization*

VI. Conclusion and Future Works

In this paper, we have presented an upgraded portable ROV [40] for under-ice environment sampling. Hardware time synchronization was integrated for multiple devices including sensors and computer systems to provide reliable sensor-fusion navigation results. The collected data from the field work revealed arctic ice has more significant features than freshwater ice, not only the pattern on the ice surface but also the morphology of the ice keel. A novel DVL-IMU-Pressure tightly-coupled odometry has been presented with detailed math derivation for robot state estimation. The algorithm is

evaluated by comparing to the widely used *robot_localization* method, and improvement was found.

Through the field work, we have also gained experience in working in extreme conditions (e.g., -30 degC). In the field, we found that vacuum test is extremely hard in the cold weather using a hand pump so an electronic pump is highly recommended. Also, the polyurethane underwater cables become stiff and bending the cables will have a chance to pull out the end cap from the housing or break the cable jacket. We also recommend to keep the vehicle under water to prevent from flash frozen which may jam the thrusters and control surfaces. Therefore, hovering capability will be a great feature. Currently, we are integrating the visual measurement into the presented method to reduce the drift. Leveraging the state-of-the-art MSCKF [33] framework, online extrinsic calibration between multi-sensor is also in progress. Furthermore, sonar images could also provide additional measurements fused into the established localization framework.

REFERENCES

[1] L. Landrum and M. M. Holland, "Extremes become routine in an emerging new arctic," *Nature Climate Change*, vol. 10, p. 1108–1115, 2020.

[2] M. Vancoppenolle *et al.*, "Role of sea ice in global biogeochemical cycles: emerging views and challenges," *Quaternary Science Reviews*, vol. 79, pp. 207–230, 2013.

[3] M. Nakawo, "Measurements on air porosity of sea ice," *Annals of Glaciology*, vol. 4, pp. 204–208, 1983.

[4] B. Loose, A. C. Naveira Garabato, P. Schlosser, W. J. Jenkins, D. Vaughan, and K. J. Heywood, "Evidence of an active volcanic heat source beneath the pine island glacier," *Nature Communications*, vol. 9, no. 2431, 2018.

[5] A. L. Forrest *et al.*, "Exploring spatial heterogeneity of antarctic sea ice algae using an autonomous underwater vehicle mounted irradiance sensor," *Frontiers in Earth Science*, vol. 7, 2019.

[6] L. D. L. Barker *et al.*, "Scientific challenges and present capabilities in underwater robotic vehicle design and navigation for oceanographic exploration under-ice," *Remote Sensing*, vol. 12, no. 16, 2020.

[7] L. C. Lund-Hansen *et al.*, "A low-cost remotely operated vehicle (rov) with an optical positioning system for under-ice measurements and sampling," *Cold Regions Science and Technology*, vol. 151, pp. 148–155, 2018.

[8] A. B. Phillips *et al.*, "Autosub 2000 under ice: Design of a new work class auv for under ice exploration," in *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, St. Johns, NL, Canada, Sep. 2020, pp. 1–8.

[9] A. Spears *et al.*, "Under ice in antarctica: The icefin unmanned underwater vehicle development and deployment," *IEEE Robotics Automation Magazine*, vol. 23, no. 4, pp. 30–41, 2016.

[10] W. Stone, K. Richmond, C. Flesher, B. Hogan, and V. Siegel, *Sub-ice Autonomous Underwater Vehicle Architectures for Ocean World Exploration and Life Search*, 2018, pp. 429–541.

[11] Y. Girdhar, P. Giguère, and G. Dudek, "Autonomous adaptive exploration using realtime online spatiotemporal topic modeling," *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 645–657, 2014.

[12] J. Wang, T. Shan, and B. Englot, "Underwater terrain reconstruction from forward-looking sonar imagery," in *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, May 2019, pp. 3471–3477.

[13] P. Norgren, R. Lubbad, and R. Skjetne, "Unmanned underwater vehicles in arctic operations," in *22nd IAHR International Symposium on Ice*, Singapore, 2014, p. 14.

[14] J. J. Leonard and A. Bahr, "Autonomous underwater vehicle navigation," in *Springer Handbook of Ocean Engineering*, M. R. Dhanak and N. I. Xiros, Eds. Cham: Springer International Publishing, 2016, ch. 14, pp. 341–358.

[15] S. Rahman, A. Q. Li, and I. Rekleitis, "SVIn2: An underwater SLAM system using sonar, visual, inertial, and depth sensor," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, Nov. 2019, pp. 1861–1868.

[16] H. Jang, S. Yoon, and A. Kim, "Multi-session underwater pose-graph slam using inter-session opti-acoustic two-view factor," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, Jun. 2021, pp. 11 668–11 674.

[17] F. Tschopp *et al.*, "Versavis-an open versatile multi-camera visual-inertial sensor suite," *Sensors (Basel)*, vol. 20, p. 1439, 2020.

[18] H. Singh, T. Maksym, J. Wilkinson, and G. Williams, "Inexpensive, small auvs for studying ice-covered polar environments," *Science Robotics*, vol. 2, no. 7, p. eaan4809, 2017.

[19] G. Williams *et al.*, "Thick and deformed antarctic sea ice mapped with autonomous underwater vehicles," *Nature Geoscience*, vol. 8, pp. 61–67, 2014.

[20] S. McPhail, R. Templeton, M. Pebody, D. Roper, and R. Morrison, "Autosub long range auv missions under the filchner and ronne ice shelves in the weddell sea, antarctica - an engineering perspective," in *OCEANS*, Marseille, France, Jun. 2019, pp. 1–8.

[21] A. K. Wåhlin *et al.*, "Pathways and modification of warm water flowing beneath thwaites ice shelf, west antarctica," *Science Advances*, vol. 7, no. 15, p. eabd7254, 2021.

[22] A. L. Forrest, B. Laval, M. Doble, R. Yeo, and E. Magnusson, "Auv measurements of under-ice thermal structure," in *OCEANS*, Quebec City, QC, Canada, Jun. 2008, pp. 1–10.

[23] A. Kukulya *et al.*, "Under-ice operations with a remus-100 auv in the arctic," in *2010 IEEE/OES Autonomous Underwater Vehicles*, Monterey, CA, USA, Sep. 2010, pp. 1–8.

[24] Z. Duguid and R. Camilli, "Improving resource management for unattended observation of the marginal ice zone using autonomous underwater gliders," *Frontiers in Robotics and AI*, vol. 7, 2021.

[25] L. C. Lund-Hansen *et al.*, "A low-cost remotely operated vehicle (rov) with an optical positioning system for under-ice measurements and sampling," *Cold Regions Science and Technology*, vol. 151, pp. 148–155, 2018.

[26] E. Vargas *et al.*, "Robust underwater visual SLAM fusing acoustic sensing," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, Jun. 2021, pp. 2140–2146.

[27] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[28] A. Palomer, P. Ridao, and D. Ribas, "Multibeam 3d underwater slam with probabilistic registration," *Sensors*, vol. 16, no. 4, p. 560, 2016.

[29] C. Cheng, C. Wang, D. Yang, W. Liu, and F. Zhang, "Underwater localization and mapping based on multi-beam forward looking sonar," *Frontiers in Neurorobotics*, vol. 15, 2022.

[30] K. Krasnosky, C. Roman, and D. Casagrande, "A bathymetric mapping and slam dataset with high-precision ground truth for marine robotics," *The International Journal of Robotics Research*, vol. 41, pp. 12–19, 2022.

[31] T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "Ntu viral: A visual-inertial-ranging-lidar dataset, from an aerial vehicle viewpoint," *The International Journal of Robotics Research*, vol. 41, pp. 270–280, 2022.

[32] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, Nov. 2013, pp. 1280–1286.

[33] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: A research platform for visual-inertial estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, May 2020, pp. 4666–4672.

[34] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, pp. 1004–1020, 2018.

[35] F. Mascarich, S. Khattak, C. Papachristos, and K. Alexis, "A multi-modal mapping unit for autonomous exploration and mapping of underground tunnels," in *2018 IEEE Aerospace Conference*, Big Sky, MT, USA, Mar. 2018, pp. 1–7.

[36] H. Chen *et al.*, "Advanced mapping robot and high-resolution dataset," *Robotics and Autonomous Systems*, vol. 131, p. 103559, 2020.

[37] A. Mallios, E. Vidal, R. Campos, and M. Carreras, "Underwater caves sonar data set," *The International Journal of Robotics Research*, vol. 36, no. 12, pp. 1247–1251, 2017.
[38] M. Bernardi *et al.*, "Aurora, a multi-sensor dataset for robotic ocean exploration," *The International Journal of Robotics Research*, vol. 41, no. 5, pp. 461–469, 2022.
[39] S. Rahman, N. Karapetyan, A. Q. Li, and I. Rekleitis, "A modular sensor suite for underwater reconstruction," in *2018 MTS/IEEE OCEANS*, Charleston, SC, USA, Oct. 2018, pp. 1–6.
[40] L. Zhao, M. Zhou, B. Loose, V. Cousens, and R. Turrisi, "Modifying an affordable rov for under-ice sensing," in *OCEANS*, San Diego, CA, USA, Sep. 2021, pp. 1–5.
[41] 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, 2014.
[42] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy, May 2007, pp. 3565–3572.
[43] P. A. Miller, J. A. Farrell, Y. Zhao, and V. Djapic, "Autonomous underwater vehicle navigation," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 3, pp. 663–678, 2010.
[44] N. Trawny and S. I. Roumeliotis, "Indirect kalman filter for 3 d attitude estimation," Department of Computer Science, University of Minnesota, Tech. Rep., 2005.
[45] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder, "Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds," *Information Fusion*, vol. 14, no. 1, pp. 57–77, 2013.
[46] A. B. Chatfield, *Fundamentals Of High Accuracy Inertial Navigation*. AIAA, 1997.
[47] G. Troni and L. L. Whitcomb, "Advances in in situ alignment calibration of doppler and high/low-end attitude sensors for underwater vehicle navigation: Theory and experimental evaluation," *Journal of Field Robotics*, vol. 32, no. 5, pp. 655–674, 2015.
[48] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.
[49] M. Grupp, "evo: Python package for the evaluation of odometry and slam." https://github.com/MichaelGrupp/evo, 2017.

APPENDIX A
USEFUL PROPERTY

We define perturbation on rotation matrix $\mathbf{R} \in \mathbf{SO}(3)$ [44]:

$$ {}^I_G\mathbf{R} \approx (\mathbf{I}_3 - \lfloor {}^I_G\tilde{\boldsymbol{\theta}} \rfloor_\times) {}^I_G\hat{\mathbf{R}} \tag{A.1} $$

Anti-Commutativity of the skew-symmetrix matrix [44], given two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$:

$$ \lfloor \mathbf{a} \rfloor_\times \mathbf{b} = -\lfloor \mathbf{b} \rfloor_\times \mathbf{a} \tag{A.2} $$

APPENDIX B
DVL UPDATE

DVL measurement Jacobian related to rotation:

$$
\begin{aligned}
&H_{D,\mathbf{R}} \\
&= \frac{\partial h_D(\centerdot)}{\partial {}^I_G\tilde{\mathbf{R}}} \\
&\approx \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{{}^I_D\mathbf{R}^\top (({\mathbf{I}_3 - \lfloor \delta {}^I_G\boldsymbol{\theta} \rfloor_\times}) {}^I_G\mathbf{R} {}^G\mathbf{v}_I + \lfloor {}^I\boldsymbol{\omega} \rfloor_\times {}^I\mathbf{p}_D) - h_D(\centerdot)}{\delta {}^I_G\boldsymbol{\theta}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{-{}^I_D\mathbf{R}^\top \lfloor \delta {}^I_G\boldsymbol{\theta} \rfloor_\times {}^I_G\mathbf{R} {}^G\mathbf{v}_I}{\delta {}^I_G\boldsymbol{\theta}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{{}^I_D\mathbf{R}^\top \lfloor {}^I_G\mathbf{R} {}^G\mathbf{v}_I \rfloor_\times \delta {}^I_G\boldsymbol{\theta}}{\delta {}^I_G\boldsymbol{\theta}} \\
&= {}^I_D\mathbf{R}^\top \lfloor {}^I_G\mathbf{R} {}^G\mathbf{v}_I \rfloor_\times
\end{aligned} \tag{B.1}
$$

DVL measurement Jacobian related to velocity:

$$
\begin{aligned}
&H_{D,\mathbf{v}} \\
&= \frac{\partial h_D(\centerdot)}{\partial {}^G\tilde{\mathbf{v}}_I} \\
&\approx \lim_{\delta {}^G\mathbf{v}_I \to 0} \frac{{}^I_D\mathbf{R}^\top ({}^I_G\mathbf{R} ({}^G\mathbf{v}_I + \delta {}^G\mathbf{v}_I) + \lfloor {}^I\boldsymbol{\omega} \rfloor_\times {}^I\mathbf{p}_D) - h_D(\centerdot)}{\delta {}^G\mathbf{v}_I} \\
&= \lim_{\delta {}^G\mathbf{v}_I \to 0} \frac{{}^I_D\mathbf{R}^\top {}^I_G\mathbf{R} \delta {}^G\mathbf{v}_I}{\delta {}^G\mathbf{v}_I} \\
&= {}^I_D\mathbf{R}^\top {}^I_G\mathbf{R}
\end{aligned} \tag{B.2}
$$

APPENDIX C
PRESSURE UPDATE

Measurement Jacobian related to rotation:

$$
\begin{aligned}
&H_{p_z,\mathbf{R}} \\
&= \frac{\partial h_{p_z}(\centerdot)}{\partial {}^I_G\tilde{\mathbf{R}}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{\mathbf{s}(\exp(\lfloor -\delta {}^I_G\boldsymbol{\theta} \rfloor_\times) {}^I_G\mathbf{R})^\top {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) - h_{p_z}(\centerdot)}{\delta {}^I_G\boldsymbol{\theta}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{\mathbf{s} {}^I_G\mathbf{R}^\top \exp(\lfloor -\delta {}^I_G\boldsymbol{\theta} \rfloor_\times)^{-1} {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) - h_{p_z}(\centerdot)}{\delta {}^I_G\boldsymbol{\theta}} \\
&\approx \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{\mathbf{s} {}^I_G\mathbf{R}^\top \lfloor \mathbf{I}_3 + \delta {}^I_G\boldsymbol{\theta} \rfloor_\times {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) - h_{p_z}(\centerdot)}{\delta {}^I_G\boldsymbol{\theta}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{\mathbf{s} {}^I_G\mathbf{R}^\top \lfloor \delta {}^I_G\boldsymbol{\theta} \rfloor_\times {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k)}{\delta {}^I_G\boldsymbol{\theta}} \\
&= \lim_{\delta {}^I_G\boldsymbol{\theta} \to 0} \frac{-\mathbf{s} {}^I_G\mathbf{R}^\top \lfloor {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) \rfloor_\times \delta {}^I_G\boldsymbol{\theta}}{\delta {}^I_G\boldsymbol{\theta}} \\
&= -\mathbf{s} {}^I_G\mathbf{R}^\top \lfloor {}^I_D\mathbf{R} {}^D_P\mathbf{R} ({}^P\mathbf{P}_{in} - {}^P\mathbf{P}_k) \rfloor_\times
\end{aligned} \tag{C.1}
$$