Naval Information Warfare Center PACIFIC

TECHNICAL REPORT 3294
NOVEMBER 2022

Drift Improvement with Reinforcement Training – Inertial Sensors - Year 2

Eric Bozeman
Minhdao Nguyen
Jeffrey Onners
Mohammad Alam
NIWC Pacific

Naval Information Warfare Center (NIWC) Pacific
San Diego, CA 92152-5001

This page is intentionally blank.

TECHNICAL REPORT 3294
NOVEMBER 2022

Drift Improvement with Reinforcement Training – Inertial Sensors - Year 2

Eric Bozeman
Minhdao Nguyen
Jeffrey Onners
Mohammad Alam
NIWC Pacific

Administrative Notes:
This report was approved through the Release of Scientific and Technical Information (RSTI) process in October 2022 and formally published in the Defense Technical Information Center (DTIC) in November 2022.

NIWC Pacific
San Diego, CA 92152-5001

NIWC Pacific
San Diego, California 92152-5001

A. D. Gainer, CAPT, USN
Commanding Officer

W. R. Bonwit
Executive Director

ADMINISTRATIVE INFORMATION

The work described in this report was performed by the Non-Linear Dynamics & Materials Branch (Code 71780) of the Basic & Applied Research Division (Code 71700), Naval Information Warfare Center (NIWC) Pacific, San Diego, CA. Funding provided by the Naval Innovative Science and Engineering (NISE) program.

Released by
John deGrassie, Division Head
Basic & Applied Research Division

Under authority of
Carly Jackson, Department Head
Cyber/S&T Department

ACKNOWLEDGMENTS

Editor: MRM

EXECUTIVE SUMMARY

OBJECTIVE

This report is a follow-on to TR-3267, the technical report titled “Drift Improvement with Reinforcement Training - Inertial - Year 1” that described the work and results from the first year of the Drift Improvement with Reinforcement Training – Inertial (DIRT-I) project. This report covers the work and results completed under the second year of this project, which focused on the use of several different inertial sensors with a wide range of performances. The overall goal was to show that the DIRT-I system could be used with any inertial sensor, and to determine the effectiveness of the DIRT-I system with lower quality inertial sensors.

METHODS

The same test environment from the first year was reused. This consisted of a custom Reinforcement Learning (RL) environment that was trained on the raw inertial measurements from an inertial system in such a way that the position more closely matched the position solution after being corrected by a Global Navigation Satellite System (GNSS). When GNSS aiding was removed, the RL system would continue to correct the inertial measurements as it was trained to do before GNSS aiding was removed. Multiple inertial sensor systems were used, with a wide range of performances. The position solutions from the DIRT-I system while using each inertial sensor was compared to the actual GNSS positions, as well as to the position solutions using the other inertial sensors.

CONCLUSIONS AND RECOMMENDATIONS

This report shows that the DIRT-I system can be used with a wide range of inertial sensor systems with minimal effort besides simply correctly formatting the inertial data from the sensor. This report also shows that the quality and performance of the inertial sensor does impact how well the DIRT-I system is able to improve the performance of the system’s position solution. Finally, this report illustrates that the RL system is being trained as expected, but the overall performance is not great – especially when more training data is available. This may indicate that the RL algorithm being used was not dynamic enough for this task, or perhaps the observation space was too narrow. There is potential to use the DIRT-I system to improve the positional error of inertial sensors without access to corrections from external sensors such as GNSS. However, several changes to the system and much more research into the effectiveness of the RL algorithm(s) used, would be required.

This page is intentionally blank.

ACRONYMS

NISE	Naval Innovative Science and Engineering
COTS	Commercial Off The Shelf
DIRT-I	Drift Improvement with Reinforcement Training – Inertial
DVL	Doppler Velocity Log
FOG	Fiber Optic Gyro
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
km	Kilometers
MEMS	Microelectromechanical Systems
min	Minutes
ML	Machine Learning
NavFil	Navigation Filter
NIWC	Naval Information Warfare Center
RL	Reinforcement Learning
RLG	Ring Laser Gyro
TF	Testing File
TRPO	Trust Region Policy Optimization
ZVU	Zero Velocity Updates

This page is intentionally blank.

CONTENTS

FIGURES

TABLES

1. INTRODUCTION

The main objective of the Drift Improvement through Reinforcement Training – Inertial sensors (DIRT-I) project is to extend the holdover time of inertial sensors in the absence of a Global Navigation Satellite System (GNSS), through the use of Reinforcement Learning (RL) or training. For the purposes of this document, the acronyms GNSS and GPS (Global Positioning System) are used interchangeably. This report is a continuation of the year one effort that was reported on in [1]. The year two effort (and this report) focus on the use of different inertial sensors with a wide range of performance specifications. The goal was to determine if the RL system offered similar performance regardless of the inertial sensor being used, or if the inertial sensor's performance limited the amount of improvement the RL system could offer. To answer this question, the same setup that was used in [1] was utilized for this work. The main difference is that data was logged from multiple inertial sensors (instead of one) and the same RL algorithm was used (rather than comparing multiple algorithms).

This page is intentionally blank.

2. BACKGROUND

2.1 INERTIAL SENSORS

Inertial sensors are used to measure the acceleration (accelerometer) and angular velocity (gyroscope) of the platform the sensor is mounted on. For navigation purposes, it is common to have both types of inertial sensors on each of the X, Y and Z axes. The performance of inertial sensors varies widely. It is directly linked to the cost of the sensor, and to a lesser extent, the technology used to build the sensor. A supporting computer and associated software will also affect the system's performance and cost. Generally speaking, inertial sensors based on Microelectromechanical Systems (MEMS) tend to have the lowest cost and worst performance. At the other end of the spectrum are Ring Laser Gyro (RLG) based inertial sensors and Fiber Optic Gyro (FOG) based sensors. Traditionally, RLG sensors offered better performance than FOG sensors, but this is not always the case anymore. Obviously, this discussion is limited to standard Commercial Off The Shelf (COTS) products.

2.2 TEST SETUP

Just like the previous effort [1], the data collections were divided into two types 1) GNSS-Enabled (Training Mode), and 2) GNSS-Denied (Testing Mode). In Training Mode, the GPS receiver provided an input to the inertial sensors being tested and their raw inertial measurements were recorded. For the purpose of this report, inertial measurements refer to acceleration and angular velocity (typically on all three axes). During the Testing Mode, the inertial sensors did not receive any input from the GPS receiver. During post-processing, the RL system was trained using the data recorded while in Training Mode, then tested using the Testing Mode data to emulate a GNSS-Denied situation. The position solutions from the Kalman filter for the GNSS-Denied and RL-Aided inertial measurements were then compared to the true GPS positions for each data point. This process was repeated for each inertial sensor, and the positional errors were compared to determine if the improvement due to the RL system was proportional to the inertial sensors' performance, or if the RL system could offer greater improvement for lower-preforming sensors. Unlike in [1], only the Trust Region Policy Optimization (TRPO) algorithm was used for this effort. TRPO was found to be the best overall preforming algorithm during the first year of this effort. It focuses on the local optimization of the policy, with an approach that attempts to increase performance by using a trusted region rather than the gradient approach used in other algorithms. TRPO is often used with robot localization and video games.

The Kalman filter used in this project is called NavFil. NavFil is a software suite developed at NIWC Pacific that implements an Extended Kalman Filter to produce a navigation solution. NavFil requires acceleration and angular velocity from an inertial measurement unit (IMU) to produce a navigation solution, but it can also utilize data from other sensors like GPS, magnetometers, air data sensors, and speed sensors to produce a better solution. NavFil has a flexible design to easily utilize different IMUs of various grades, and different GPS, magnetometers, air data sensors, and speed sensors.

HARDWARETable 1 shows a performance comparison of the inertial sensors used in this effort. There is a strong correlation between price and gyroscope bias instability (which is a common indicator for the overall performance of the system). The bias instability of the gyroscope is the standard metric for comparing the performance of several inertial sensors. The main focus of this report is the effectiveness of the DIRT-I system on different inertial sensors with varying performance specifications.

Table 1. Drift rate comparison of inertial navigation systems.

Inertial Navigation System	Approximate Price†	Drift Rate (Gyro Bias Instability)††	Type
Kearfott KI-4901S	>$100,000	0.003°/hr.	RLG INS
KVH P-1725 IMU	$9,950	≤ 0.05°/hr.	FOG IMU
SBG Systems ELLIPSE-E	$3,500	8°/hr.	MEMS INS

† *Approximate prices obtained from quotes dated June 2021*

†† *Drift rates based on Gyro in-run bias instability from individual sensor datasheets available through the manufacturers' public websites*

2.3 KEARFOTT KI-4901S INS [2]

The Kearfott KI-4901S is a navigation-grade INS. It is able to receive aiding inputs from a Doppler Velocity Log (DVL), speed sensor, depth sensor and position inputs from a GNSS receiver. The Kearfott utilizes a Monolithic Ring Laser Gyroscope, and has a bias stability of 0.003°/hr. In addition to providing a position solution, the Kearfott also outputs raw inertial sensor data from the gyroscope (angular velocity) and accelerometer (acceleration), as well as attitude, heading and velocity information. This Kearfott is configured to require aiding information from a speed sensor, which provides external aiding in the form of Zero Velocity Updates (ZVU). This does improve the position solution from the Kearfott, but the speed sensor information was not directly used by the DIRT-I RL algorithm. For this reason, this report will differentiate between the standard Kearfott output or position solution (includes ZVU aiding) and the NavFil output or position solution with Kearfott data (raw, unaided inertial measurements).

2.4 KVH P-1725 IMU [3]

The P-1725 IMU from KVH Industries is a Fiber Optic Gyro (FOG) based sensor with a gyroscope bias stability of 0.05°/hr. This puts the P-1725 IMU in the Navigation Grade category, but with far worse performance than the Kearfott. A position solution was calculated by NavFil based on GNSS data from the logging computer (provided by the differential GNSS receiver only while in Training Mode), raw inertial data from the P-1725 and speed data from the OBD interface. Being an IMU, the P-1725 provides raw inertial measurements, but does not contain its own Kalman filter (like the Kearfott). The reduced performance and limited output capabilities allow the P-1725 to be significantly cheaper than the Kerfott.

2.5 SBG ELLIPSE-E INS [4]

The Ellipse-E INS from SBG Systems is a high-performance MEMS-based sensor that provides orientation and navigation data when using an external GNSS reference. It has the ability to receive inputs from speed sensors such as an odometer or Doppler Velocity Log (DVL). Its gyroscope has an in-run bias stability of 8°/hr., which puts it in the Industrial Grade category. For the purposes of the tests conducted for this report, the raw inertial data from the Ellipse-E was feed to NavFil with GNSS data from the logging computer, which received the data from the differential GNSS receiver, to produce a navigation solution. This was only true during while in Training Mode. Also, speed data from the OBD interface was feed to NavFil. Although the Ellipse-E represents the "low-end" of

performance specifications of the sensors tested for this effort, it is still considered a very high-end sensor when compared to other MEMS-based inertial sensors.

2.6 SPARTON AHRS-M2 COMPASS [5]

The Sparton compass was intended to be used as an external aiding sensor that could be provided to NavFil or to the RL environment. Unfortunately, this compass never worked properly. There was always an offset in the heading provided by this compass and the heading from the differential GNSS receiver. The offset was not consistent between data collection events, and it was larger any offset due to a misalignment between the two sensors. Several attempts were made to contact the manufacturer to troubleshoot the compass, but no support was ever provided. Eventually it was decided to remove this sensor from future data collection events and ignore the data that was previously collected from it.

2.7 OTHER HARDWARE

The GPS receiver (V102 GNSS Vector Compass), speed sensor (OBDII module), and data logging computer (Intel NUC) are all the same components used in the first year of this effort, and are described in [1].

This page is intentionally blank.

3. SETUP AND DATA COLLECTION

The data collection and test procedures are similar to those used during the first year of this effort, and are described in [1]. All hardware was mounted to the same platform inside the test vehicle (which was the same vehicle and driver used in [1]). Figure 1 shows all the hardware mounted to this platform during one of the data collection events. Distances were measured in the X, Y and Z directions, between the center of the vehicle's rear axle and each sensor under test. These measurements were used to calculate a lever arm in the calibration of the NavFil Kalman filter. The data logging setup was reconfigured to collect data from all three inertial sensors at the same time. This means that for the Kearfott, KVG and SBG sensors, the raw inertial measurements as well as the outputs from the GPS-enabled, GPS-denied, and RL-aided versions of the NavFil Kalman filter were all logged during data collection events. The post-processing analysis was the same as [1]. Data from each sensor was analyzed separately, then compared to each other and GPS truth data.

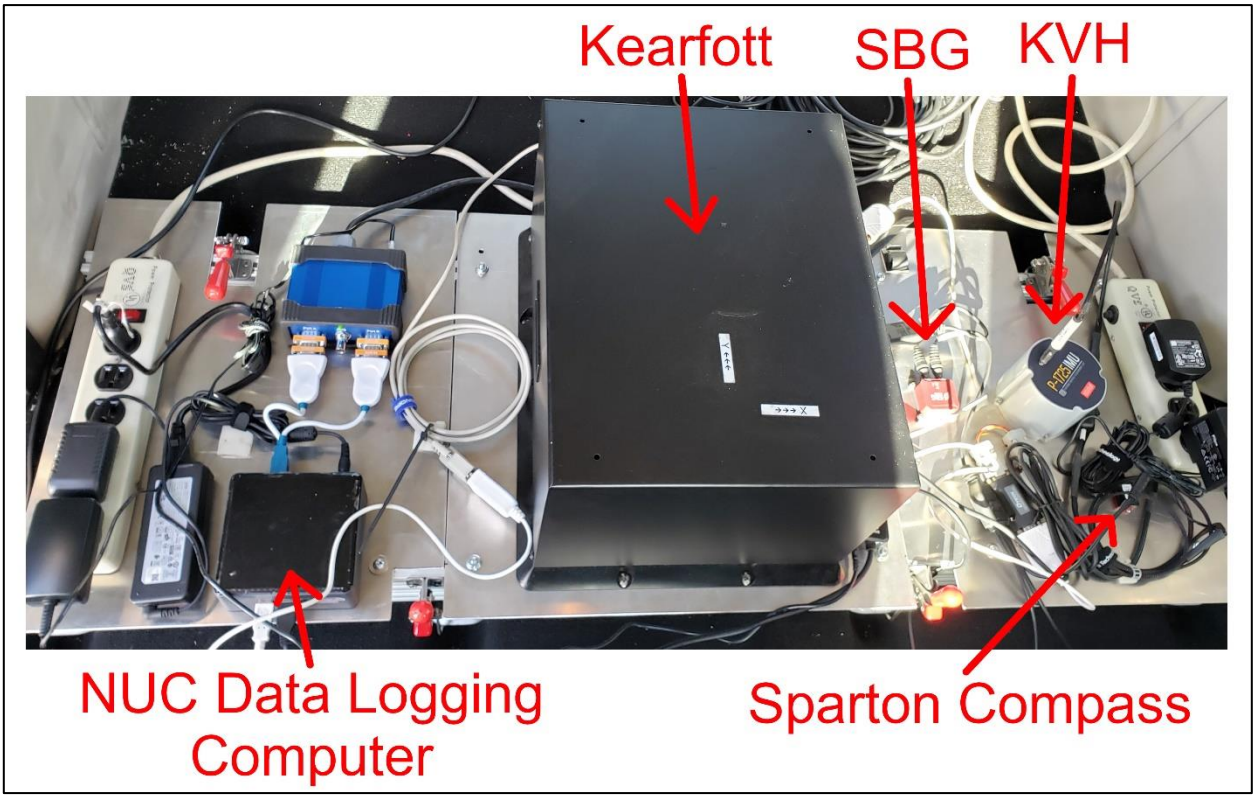

Figure 1. Data collection hardware setup.

3.1 REINFORCEMENT LEARNING (RL) AND SOFTWARE SETUP

Unlike the first year of this effort, only one RL algorithm was used in the DIRT-I system. During the RL algorithm evaluation in year one, it was determined that the Trust Region Policy Optimization (TRPO) algorithm produced the best results. Therefore, TRPO was used for all analysis covered in this report. It focuses on the local optimization of the policy, with an approach that attempts to increase performance by using a trusted region rather than the gradient approach used in other algorithms. TRPO is often used with robot localization and video games.

A number of software changes were added between FY21 and FY22, most of which were optional and do not replace the foundation of what was described in the FY21 technical report [1]. Examples include improved organization to the parameter file that allows the user to quickly test and change the observation space (note that the same observation space was used for all data in the results section), automated scripts to train and/or test a model on multiple files to easily compare performance of different sensors at a different number of trainings, and additional post-processing and analysis tools including RL actions versus reward colormap plots (see Section 4.3).

Major changes include the chosen observation space and Kalman filter initialization process. For instance, after analyzing results of the FY21 observation space, it was found that using the direct degree heading value of the RL and GNSS-Denied NavFils in the observation space could cause some issues. For one the IMU observation states (acceleration and angular velocity) are relative to sensor's mounted position, meaning the NavFils starting heading is somewhat irrelevant. If two random states had identical IMU values but different headings (I.e. +60° and +100°), the ideal RL action would be the same, but the agent would have to explore both independently, ultimately making training longer and more difficult. Moreover, having two separate heading values adds unnecessary dimensions. The difference between the two values is relatively small, making it difficult for the system to make distinctions between observation states, or understand how its actions affect the observation space. For all of these reasons, it was decided to consolidate the headings into a single heading difference value that is calculated with the following equation: ((RL heading – Denied Heading) * Scalar). This allows the observation heading value start at zero at the beginning of an episode. Once an RL action is applied, there will be a heading difference between RL and Denied, causing the observation value to change. This will produce either a positive or negative reward based on if that heading is closer or further away from the ground truth heading. As the magnitude of this difference grows, so does the potential for very positive or very negative rewards. This change reduces the number of dimensions to learn (and thus reduces training time) and solves the described issues by allowing the RL system to more easily recognize how a single action directly correlate to a negative or positive observation value.

Additionally, during training, the observation space values are reset after the duration of the episode, in our case 60 seconds. This causes the maximum difference between the RL and Denied headings to be relatively small. During test mode, the headings do not get reset, since the GPS is unavailable and disallows the system from resetting to the baseline. As non-zero RL actions are chosen, the difference between the RL and Denied headings can become incredibly large, meaning the policy will try to interpret states that have not yet been observed and learned, and therefore would be unlikely to choose actions competently. Therefore, an offset was introduced to be added to both headings every 60 seconds, mimicking an episode reset and allowing the policy to choose actions based on what it has learned.

Finally, an issue was found where the NavFil was not getting initialized properly, causing larger position estimate errors over time. Fixing the NavFil initialization process improved the Denied Baseline positions. While this can improve the RL systems ability to see a more accurate representation of heading vs IMU, it also decreases the performance improvements seen in FY21 since the baseline positions are closer to the ground truth by default.

3.2 DIRT-I OPERATIONAL MODES

The DIRT-I system has three operational modes. These modes would be used in normal real-time operation, and were used for collecting all of the data discussed in this report. All three modes of the DIRT-I system are described below:

1) Training Mode:

The RL system makes corrections to the raw inertial data from the inertial system (Kearfott), and generates a position estimate from both the GNSS-Aided and Denied NavFils. These estimates are compared to the position solution from the inertial sensor which is being aided (corrected) by the GNSS receiver.

This mode can only be enabled when a valid GNSS position solution is available to both the inertial system and the DIRT-I system.

2) Testing Mode:

The DIRT-I system receives the raw inertial data from the inertial sensor and makes corrections to it based on the model that the RL system defined while the system was in Training Mode. The position data from the inertial system and the "aided" NavFil are ignored in Testing Mode. The GNSS-Denied NavFil receives the altered inertial data from the RL system and generates a position estimate, which is logged as the position solution from the DIRT-I system.

This mode can only be enabled when there is no valid GNSS position solution available.

3) Skip Mode:

This is a catch-all, fail-safe mode where the system neither updates the RL model or provides a position solution. It preserves all system states as they are, and addresses issues that arise from essentially pausing the system in one location, and unpausing it in a different location.

This mode will be entered if there is an issue with any of the aiding sensors. This could be a problem with the speed sensor data not being available, or an unreliable GNSS position solution that is not the result of a real GNSS-denied scenario. For instance, when passing tall buildings, the number of visible GNSS satellites may drop to the point that the position is still being provided, but at an increased error. In this case, training could be detrimental, but a DIRT-I position solution may be unnecessary (if situation only persists a short time) or unreliable (if not enough training has occurred yet).

3.3 DATA COLLECTION

The data collection events were separated into two categories that aligned with either the Training Mode (GPS available) or the Testing Mode (GPS not available). In Training Mode, the GPS receiver provided an input to the inertial sensors being tested and their raw inertial measurements were recorded. During the Testing Mode, the inertial sensors did not receive any input from the GPS receiver. During post-processing, the RL system was trained using the data recorded while in Training Mode, then tested using the Testing Mode data to emulate a GNSS-Denied situation. The position solutions from both the GNSS-Denied and RL-Aided Kalman filters were then compared to the true GPS positions for each data point. This process was repeated for each inertial sensor, and the positional errors were compared to determine if the improvement due to the RL system was proportional to the inertial sensors' performance, or if the RL system could offer greater improvement for lower-preforming sensors. Table 2 shows a list of each data collection done during the second year of this effort. The date of collection, duration, and distance of each data set is shown, as well as whether or not GPS was made available. Data sets with a "No" in the "GPS Aided" column are testing data sets, and those with a "Yes" are training data sets. The highlighted rows are data sets that are represented in this report. The remaining data sets were used for analysis, but not specifically referenced in this report.

Table 2. Data Collection File Information.

Date	Run #	Length (mins)	Distance (km)	GPS Aided
03/22/22	1	25	9.786	Yes
03/22/22	2	42	48.195	Yes
03/22/22	3	35	42.452	Yes
03/22/22	4	40	16.403	Yes
04/04/22 (TF2)	1	128	94.713	No
04/12/22 (TF3)	1	91	132.372	No
04/12/22	2	96	137.135	No
04/25/22	1	25	9.116	Yes
04/25/22	2	27	15.233	Yes
04/25/22	3	27	9.303	Yes
04/25/22	4	21	5.893	Yes
04/25/22	5	36	18.19	Yes
05/31/22	1	26	11.135	Yes
05/31/22	2	32	23.981	Yes
05/31/22	3	50	27.103	Yes
05/31/22	4	26	8.871	Yes
06/28/22	1	147	157.067	No
07/05/22	1	35	14.266	Yes
07/05/22	2	38	14.272	Yes
07/05/22	3	36	14.279	Yes
07/05/22 **(TF1)**	4	35	14.269	No
07/12/22	1	235	291.473	No

† *GPS Aided files used for training, GPS Denied files used for testing.*

†† *Testing Files used in results section are labeled as TF1, TF2, and TF3 in 'Date' column*

4. RESULTS

Overall, the results are very mixed across testing files, and on average are worse than the GNSS-Denied baseline. It appears that the accuracy of a sensor affects the amount of improvement the RL actions can make (where improvement is a percent error against the Denied baseline and can be positive or negative). This section will include plots to demonstrate that although performance is poor, the RL system is learning appropriately and still shows potential to improve position estimates. Probable reasons for performance issues and possible solutions are explored in here and in the Conclusion section.

4.1 POSITION ERROR PLOTS

After creating and training an RL model on the full dataset for each inertial sensor, its performance was tested on three testing files (TF1,2, and 3). These files were selected to hit different observation space conditions. TF1 followed a similar route to the first few training files. TF2 was a longer run with a large variety of observation states (episodes with many and large turns, episodes going in a single direction, variable speeds, etc.). TF3 was a longer run with a relatively straight driving path, meaning less expected variety in observation states. Each color represents the NavFil position estimates using different sensors IMU values as input, where red represents the Kearfott, blue represents the KVH, and green represents the Ellipse. Solid lines represent the GNSS-Denied baseline, whereas dotted lines show NavFil position estimates after receiving RL-modified IMU inputs. A solid yellow line is used to represent the ground truth NavFil with GPS Aiding.

4.1.1 Test File 1

Figure 2 shows a comparison of the position solutions for each inertial sensor when tested using TF1 (test file 1). The Kearfott data is shown on the left, the KVH data is shown in the middle, and the Ellipse data is shown on the right. Each plot compares the position solutions from NavFil, using the raw inertial data from each sensor, when NavFil was aided by GPS, aided by the RL system, or had no aiding.

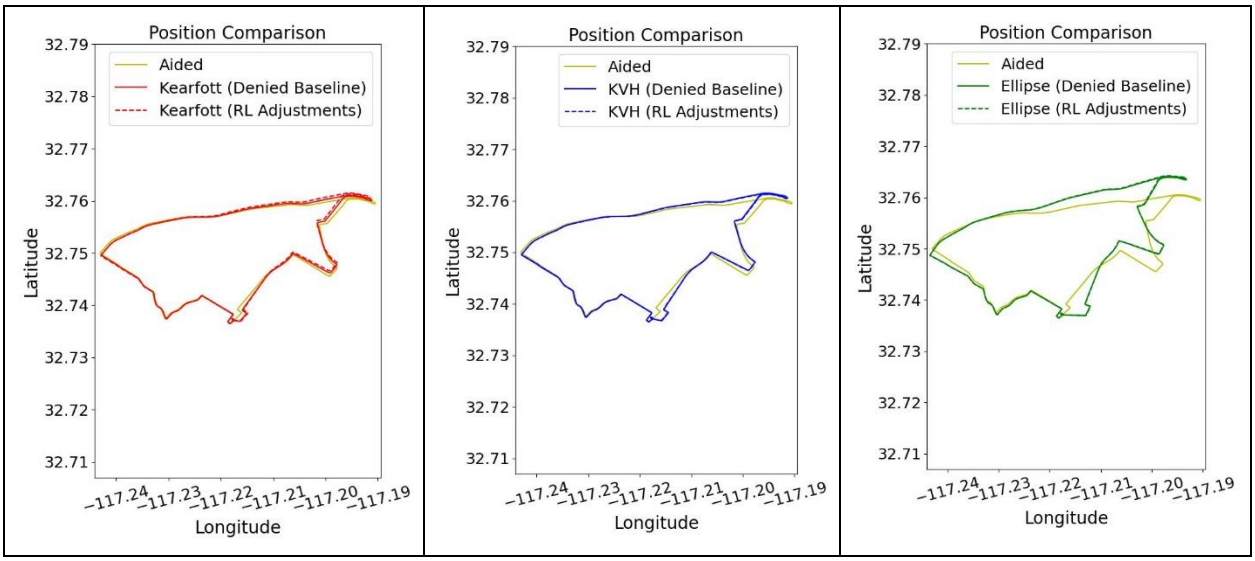

Figure 2. Comparison of position solutions for Kearfott, KVH and Ellipse using TF1 data set.

Figure 3 shows the average cumulative position error for each sensor. The plot compares the position errors (compared to GPS truth positions) when NavFil receives data from each sensor and is either aided by the RL system or not aided at all. The position error from the Kearfott's own Kalman Filter is also plotted (labeled "Kearfott INS"). This is used as the lower bound, or most accurate because the Kearfott's Kalman Filter is aided by a speed sensor by default. Only the NavFil Kalman Filter can be aided by the RL system. The dashed lines represent the cumulative error for RL aided systems, and the solid lines (except for Kearfott INS) are the unaided systems.

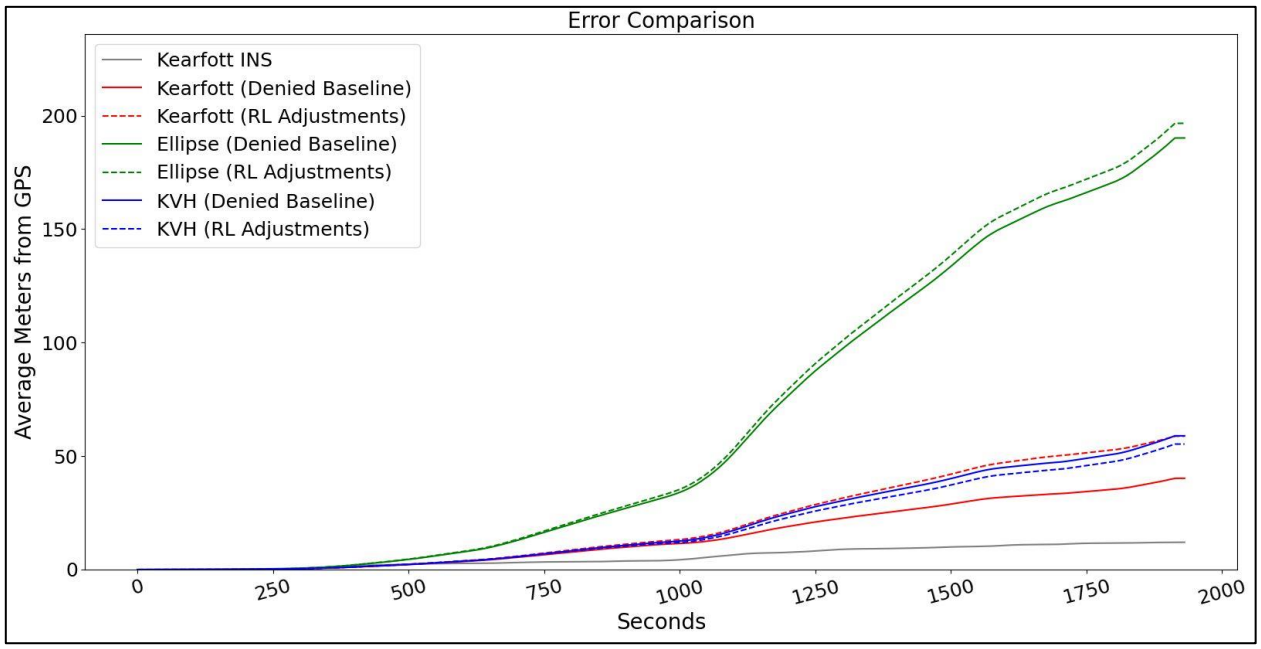

Figure 3. Cumulative error comparison of each sensor using TF1 data set.

4.1.2 Test File 2

Figure 4 shows a comparison of the position solutions for each inertial sensor when tested using TF2 (test file 2). The Kearfott data is shown on the left, the KVH data is shown in the middle, and the Ellipse data is shown on the right. Each plot compares the position solutions from NavFil, using the raw inertial data from each sensor, when NavFil was aided by GPS, aided by the RL system, or had no aiding.

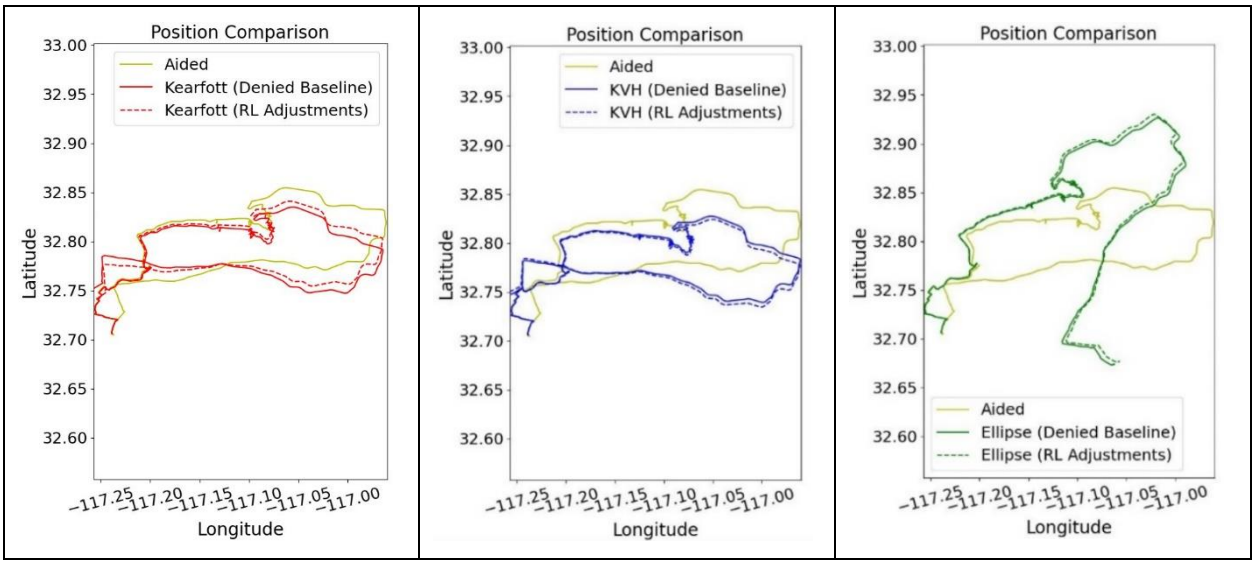

Figure 4. Comparison of position errors for Kearfott, KVH and Ellipse using TF2 data set.

Figure 5 shows the average cumulative position error for each sensor. The plot compares the position errors (compared to GPS truth positions) when NavFil receives data from each sensor and is either aided by the RL system or not aided at all. The position error from the Kearfott's own Kalman Filter is also plotted (labeled "Kearfott INS"). This is used as the lower bound, or most accurate because the Kearfott's Kalman Filter is aided by a speed sensor by default. Only the NavFil Kalman Filter can be aided by the RL system. The dashed lines represent the cumulative error for RL aided systems, and the solid lines (except for Kearfott INS) are the unaided systems.

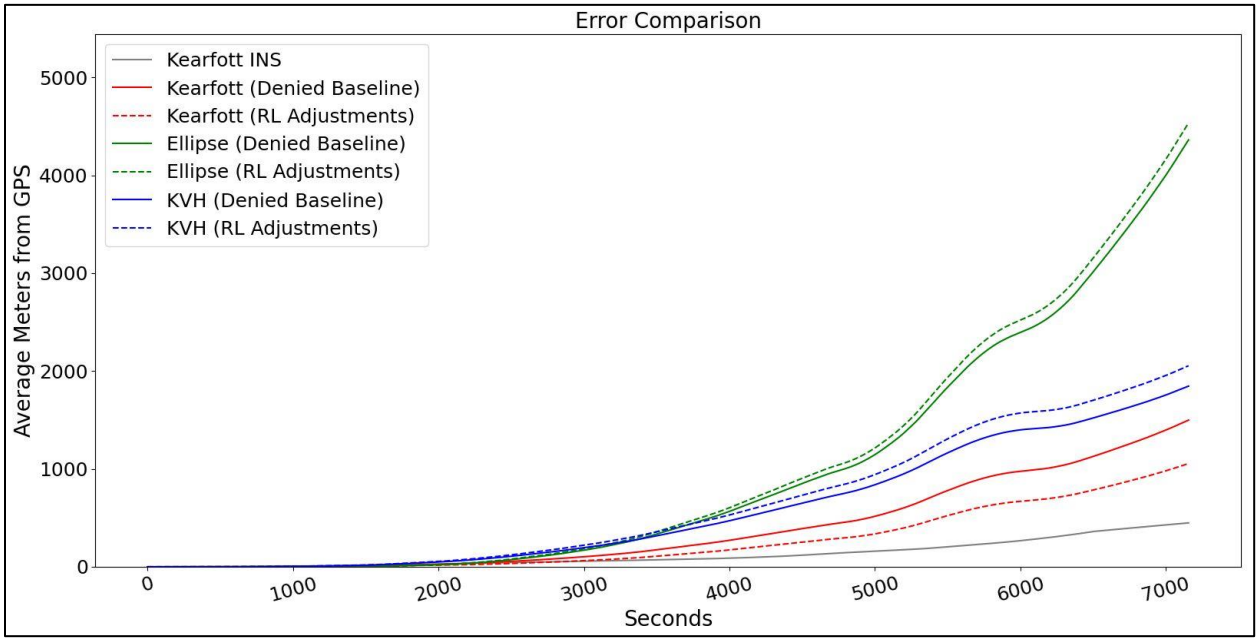

Figure 5. Cumulative error comparison of each sensor using TF2 data set.

4.1.3 Test File 3

Figure 6 shows a comparison of the position solutions for each inertial sensor when tested using TF3 (test file 3). The Kearfott data is shown on the left, the KVH data is shown in the middle, and the Ellipse data is shown on the right. Each plot compares the position solutions from NavFil, using the raw inertial data from each sensor, when NavFil was aided by GPS, aided by the RL system, or had no aiding.

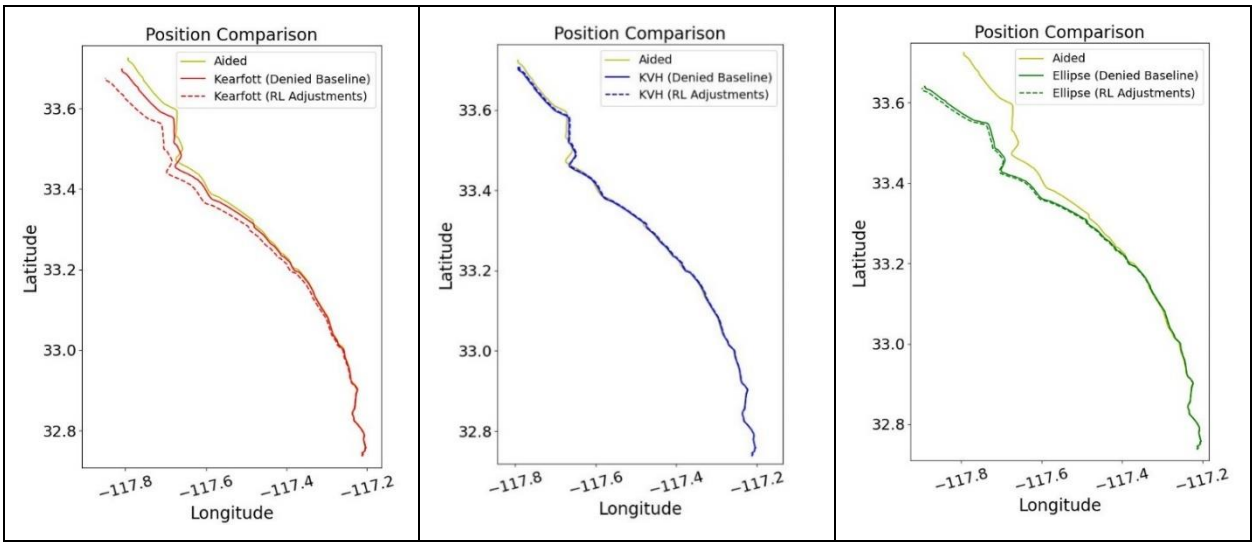

Figure 6. Comparison of position errors for Kearfott, KVH and Ellipse using TF3 data set.

Figure 7 shows the average cumulative position error for each sensor. The plot compares the position errors (compared to GPS truth positions) when NavFil receives data from each sensor and is either aided by the RL system or not aided at all. The position error from the Kearfott's own Kalman Filter is also plotted (labeled "Kearfott INS"). This is used as the lower bound, or most accurate because the Kearfott's Kalman Filter is aided by a speed sensor by default. Only the NavFil Kalman Filter can be aided by the RL system. The dashed lines represent the cumulative error for RL aided systems, and the solid lines (except for Kearfott INS) are the unaided systems.

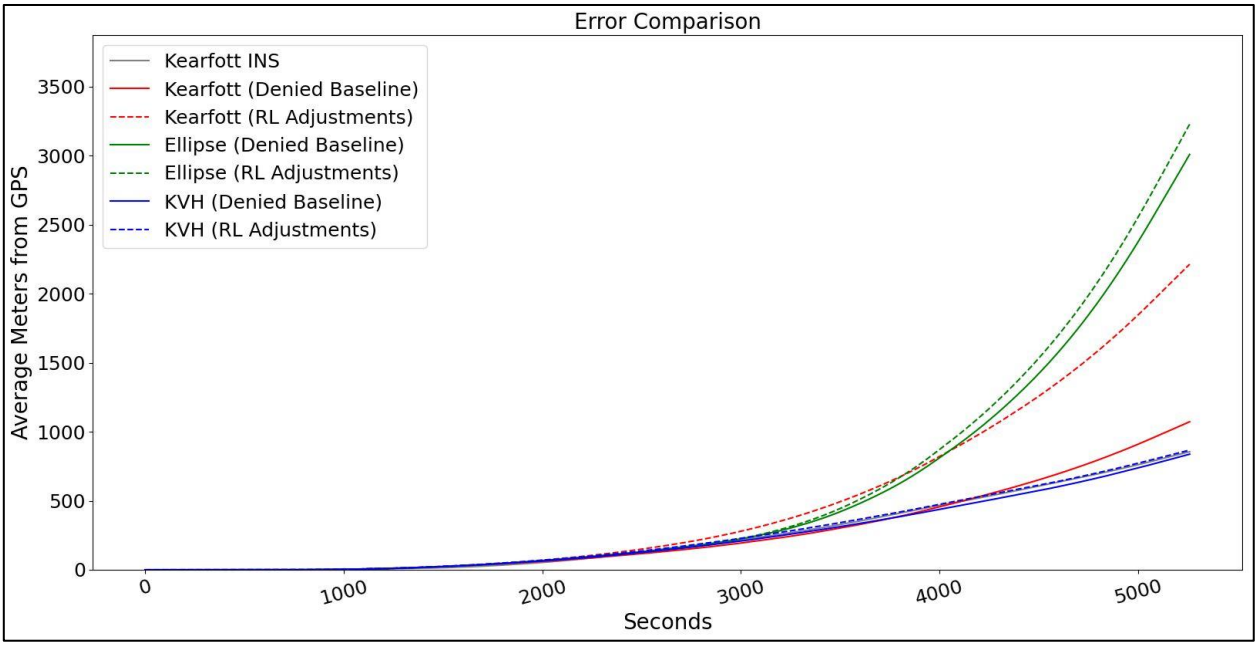

Figure 7. Cumulative error comparison of each sensor using TF3 data set.

4.2 POSITION ERROR PLOTS SUMMARY

The results seem fairly inconsistent across all sensors. In some cases, the RL actions cause its NavFil's position estimates to be closer to the ground truth than compared to the GNSS-Denied NavFil (Kearfott in Figure 5 or KVH in Figure 3). However, on average, most sensors perform worse than the GNSS-Denied NavFil. This indicates that the DIRT-I system did not (or possibly cannot) learn how to modify a sensors IMU data to improve position estimates. However, the following result sections should illustrate that the system is actually learning, just not what was intended to learn, and that the system has the capability to be feasible.

4.3 ACTION DISTRIBUTION VS REWARD PLOTS

The plots in this section show how the amount of training on the model affects the RL system's action distribution and resulting reward values.

4.3.1 Colormap Explanation

Figure 8 shows an example of an Action Distribution vs Episode Reward plot that is used to explain how to interpret the image and illustrate the kind of information these plots can provide.

The X-axis represents each 60-second episode. The left Y-axis represents the RL action chosen, where negative values correlate to actions that rotate the NavFil's heading clockwise, positive values correlate to actions that rotate the NavFil's heading counter-clockwise, and zero is no action (or more accurately, a chosen action that does not affect the NavFil's heading). The color bar shows the number of each action that was chosen for each episode, where darker (purple) represents a low number of actions, and brighter (yellow) represents a low number of actions. The right Y-axis display red dots showing the reward obtained for each episode. Positive rewards mean that the RL actions caused the NavFil to rotate its heading closer to the ground truth Navfil heading with respect to the GNSS-Denied NavFil heading. Negative values mean the heading moved further away from the ground truth heading. Grey vertical lines represent the start of new training file (TF), followed by the number of the file.

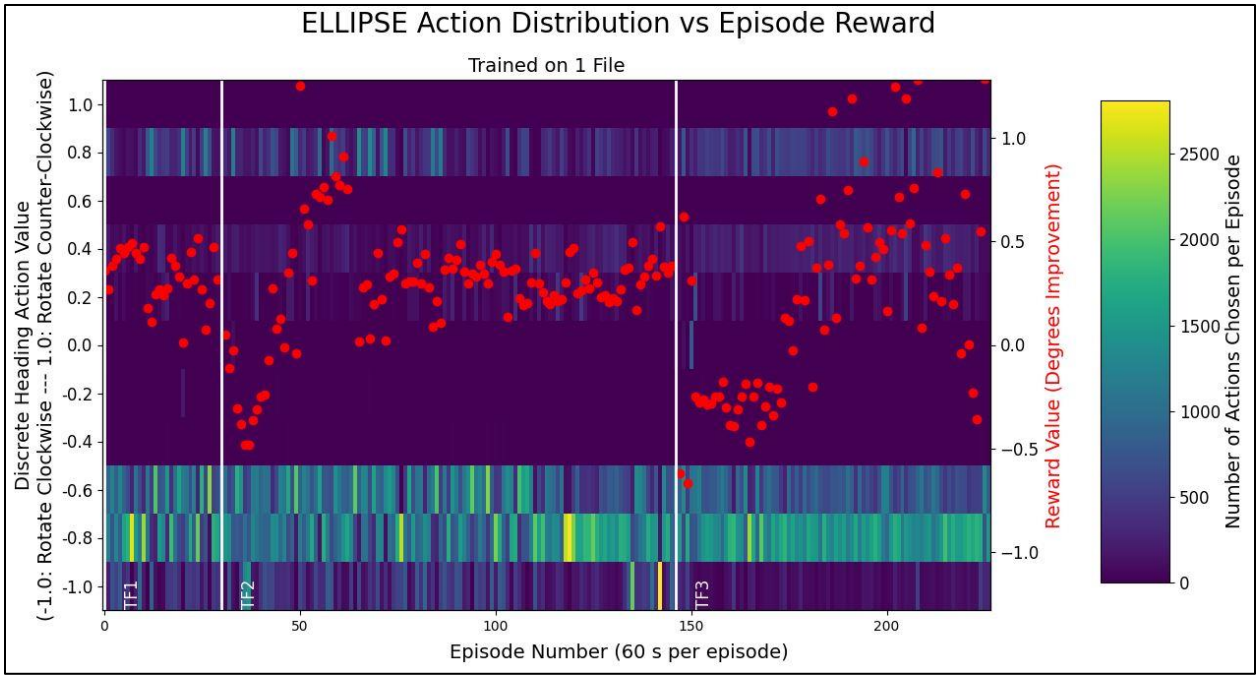

Figure 8. Ellipse Action Distribution vs Reward Colormap.

Figure 8 specifically shows IMU data from the Ellipse that was trained on a single training file. The model was tested on the 3 test files, and this plot shows which actions the RL system believes will improve the NavFil's heading, and how the actions actually affected the heading. For instance, on TF1 there is a heavy dose of actions between –1.0 and –0.6, meaning it took actions to move the NavFil heading clockwise. The bulk of the red dots here appear to be positive, meaning the RL actions moved the NavFil's heading closer to the ground truth heading per episode, and in turn caused its position estimates to improve. This pattern generally continues for training files 2 and 3. While it is good to see many positive rewards, it is also suspicious that the actions do not change a bit more between files, which would be expected since each file has different observations states, and thus should produce different actions.

A variation of the action distribution plots is shown in Figure 9, where the reward values are not reset at the start of a new episode. The red dots now accumulate and appear as a line, which will result in a path similar to the error plots in Section 4.1. Figure 8 and Figure 9 show training on one file, whereas the plots in the following sub-sections show the results when the system is trained on multiple files.

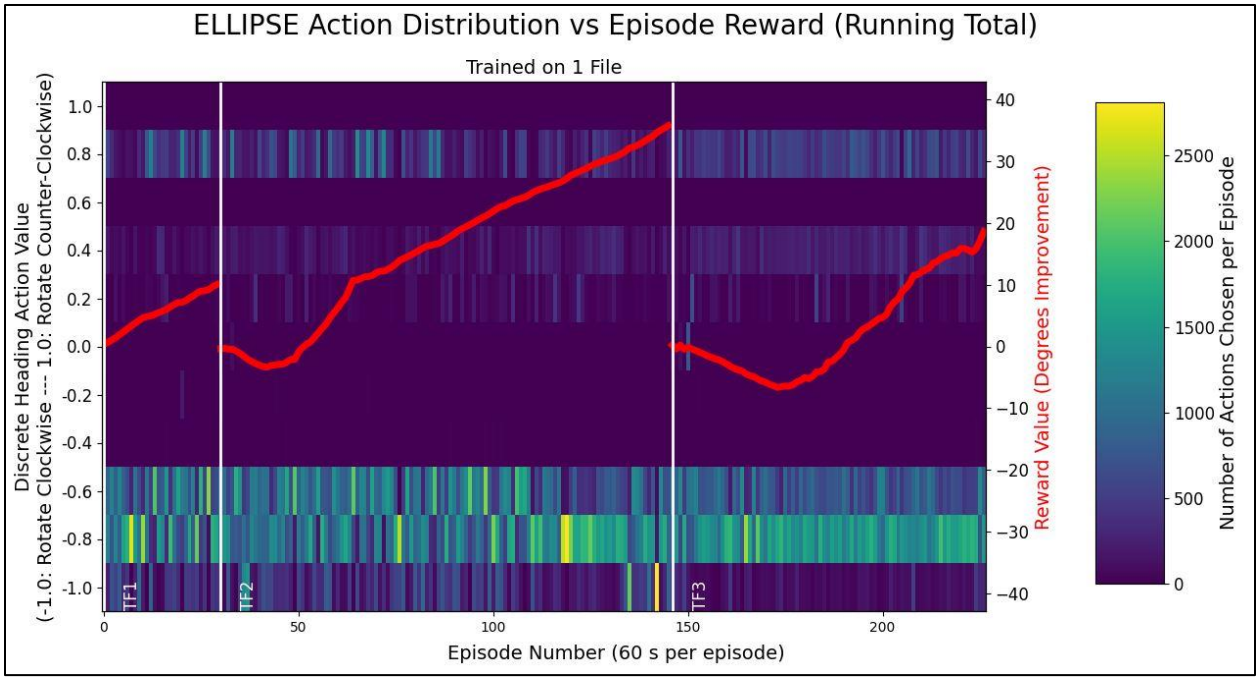

Figure 9. Ellipse Action Distribution vs Cumulative Reward Colormap.

4.3.2 Kearfott

Figure 10 shows the Action Distribution vs Episode Reward plots when the NavFil Kalman Filter uses inertial data from the Kearfott. The plots show the results of the system being trained on one, two and three files, as well as being trained on all available data sets. As with Figure 8 and Figure 9, each plot shows the results when tested on test files TF1, TF2, and TF3.

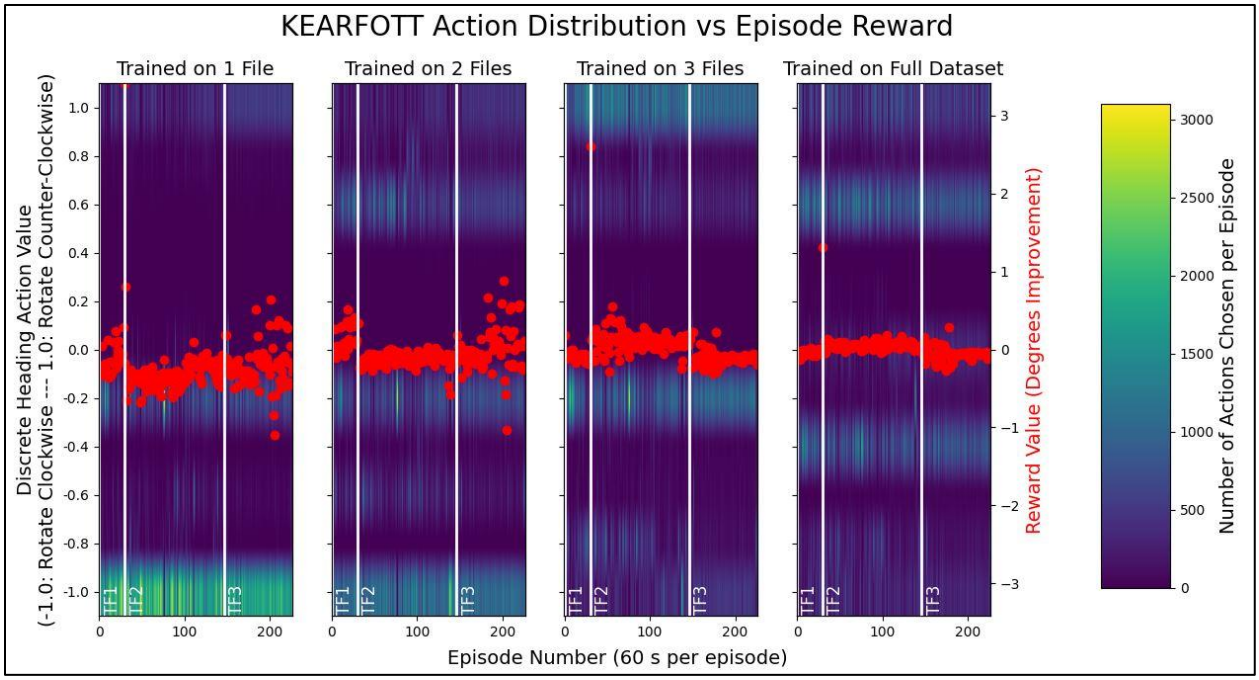

Figure 10. Kearfott Action Distribution vs Reward Colormap.

4.3.3 KVH

Figure 11 shows the Action Distribution vs Episode Reward plots when the NavFil Kalman Filter uses inertial data from the KVH. The plots show the results of the system being trained on one, two and three files, as well as being trained on all available data sets. As with Figure 8 and Figure 9, each plot shows the results when tested on test files TF1, TF2, and TF3.

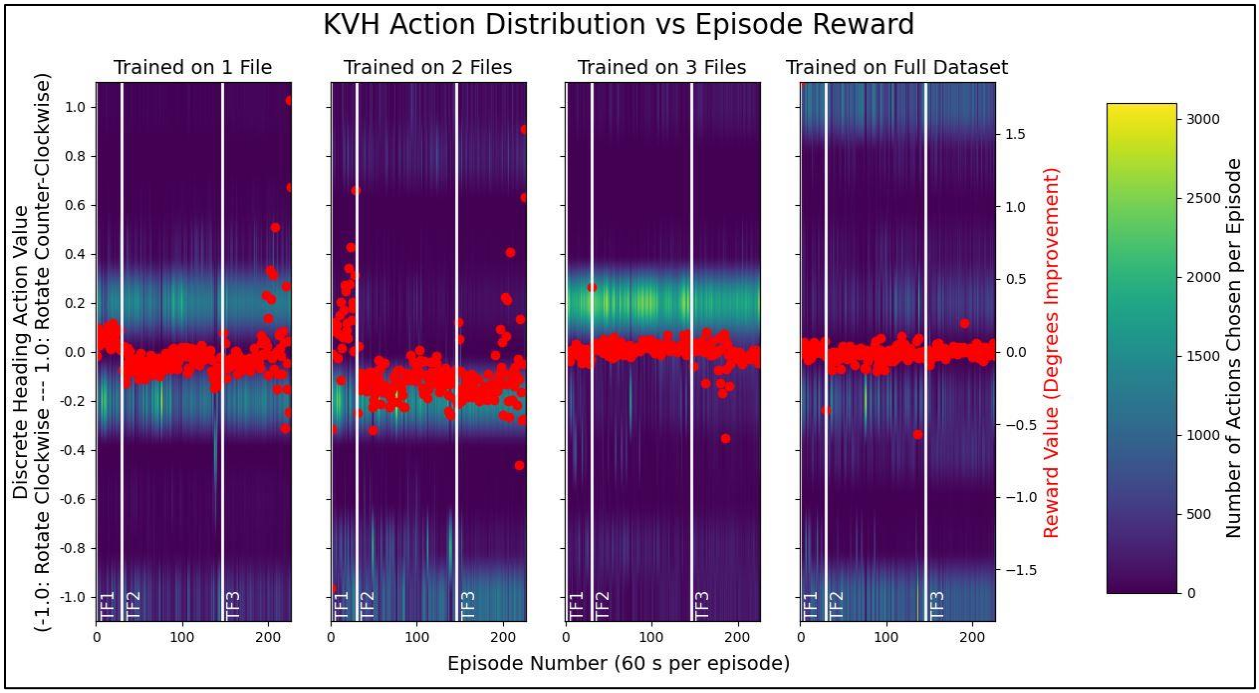

Figure 11. KVH Action Distribution vs Reward Colormap.

4.3.4 Ellipse

Figure 12 shows the Action Distribution vs Episode Reward plots when the NavFil Kalman Filter uses inertial data from the Ellipse. The plots show the results of the system being trained on one, two and three files, as well as being trained on all available data sets. As with Figure 8 and Figure 9, each plot shows the results when tested on test files TF1, TF2, and TF3.

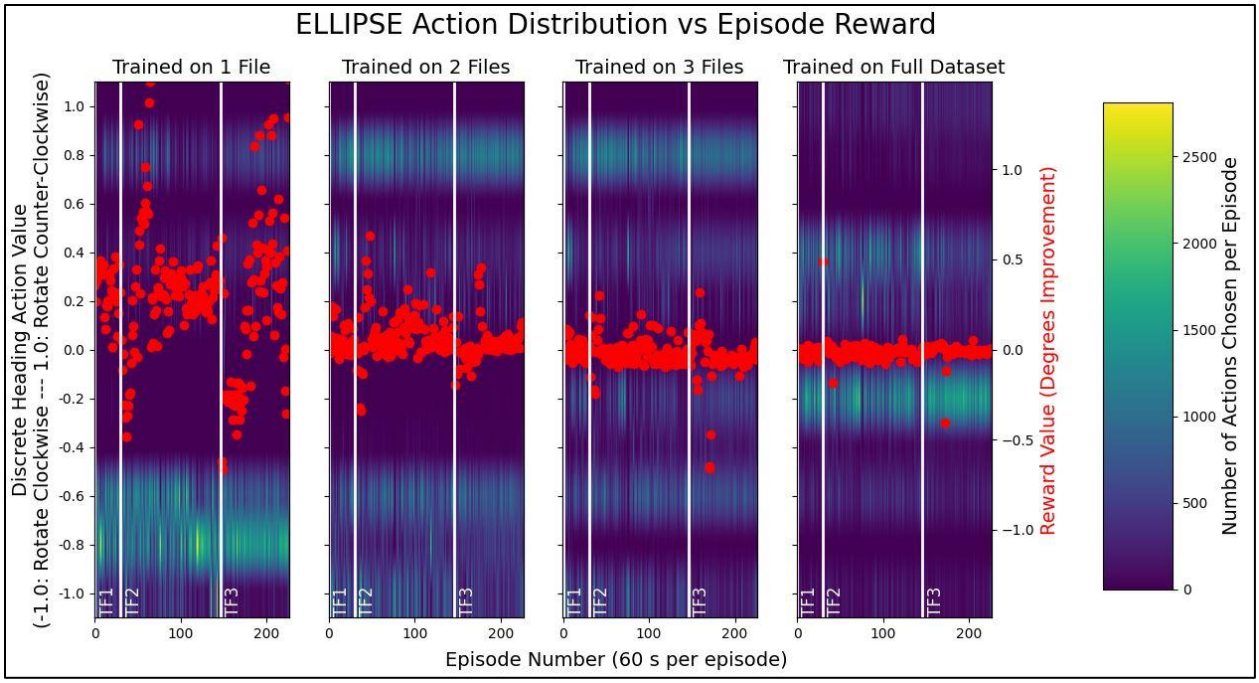

Figure 12. Ellipse Action Distribution vs Reward Colormap.

4.3.5 Summary of Action Distribution vs Reward Plots

Figure 10 through Figure 12 each represent results for a different sensor, and each subplot represents a different amount of training. The first subplot shows results after training on one file, the second on two files, the third on three files, and the fourth on the full dataset which includes a total of 15 files. Across all sensors, the action distribution changes as the model receives more training files, indicating that the RL system is updating as it explores the environment. However, each individual subplot still has a similar distribution between episodes, meaning the RL is not making a large distinction between observation states. This may mean that the system is learning more about an individual sensor's bias, rather than magnitude of noise depending on the accuracy of the sensor; this is not necessarily bad as knowing sensor bias and accounting for it in the NavFil can improve position estimates. Another interesting outcome is that across each sensor and as each model trains more, the rewards begin to stabilize around zero.

One might expect that as the model gets more data, the rewards should converge across training files (showing generalizability) to some positive value. Instead, this converging value is closer to zero than positive. This illustrates that the system is learning and updating properly; the issue is that it looks like the system learns that in order to minimize the very negative rewards seen in the first couple of subplots, it also decreases the magnitude of potential positive rewards. The final outcome of this results is very slightly negative rewards per episode, but a cumulate position error slightly worse than the Denied NavFil. Although it appears the system is learning, it is not able to learn how to consistently improve heading across testing files.

4.4 ACTION SCALING RESULTS

During training, the agent heavily explores all actions in order to find which ones produce high rewards, making the action distribution relatively uniform. During testing, the 'best' action is chosen, resulting in actions generally skewed towards a few discrete values. Additionally, high actions boundaries (A_{bounds}) were used during training in order to allow the network to understand how its actions affect rewards more easily. This in turn makes consecutive testing actions have an even heavier impact, increasing the likelihood of poor performance. Therefore, during test mode, a scalar A_{test_scale} is introduced to be multiplied to the learned RL actions to decrease its directional magnitude, and immediately showed improved results. Inclusion of A_{test_scale} should not affect good results since actions are based on heading direction. For instance, a learned action of $+1.0*A_{bounds}$ will change to $+1.0*A_{bounds}/A_{test\_scale}$. Both actions will rotate the heading in the same direction, just by different magnitudes. The action bounds were mapped to -0.0002 to +0.0002, independent of bounds chosen during training.

Removing this scalar showed interesting results and helps illustrate how that continued training does in fact stabilize the rewards, illustrating capability to learn. Figure 13, Figure 14 and Figure 15 show Action Distribution vs Episode Reward plots without scaling, for the Kearfott, KVH and Ellipse sensors, respectively.

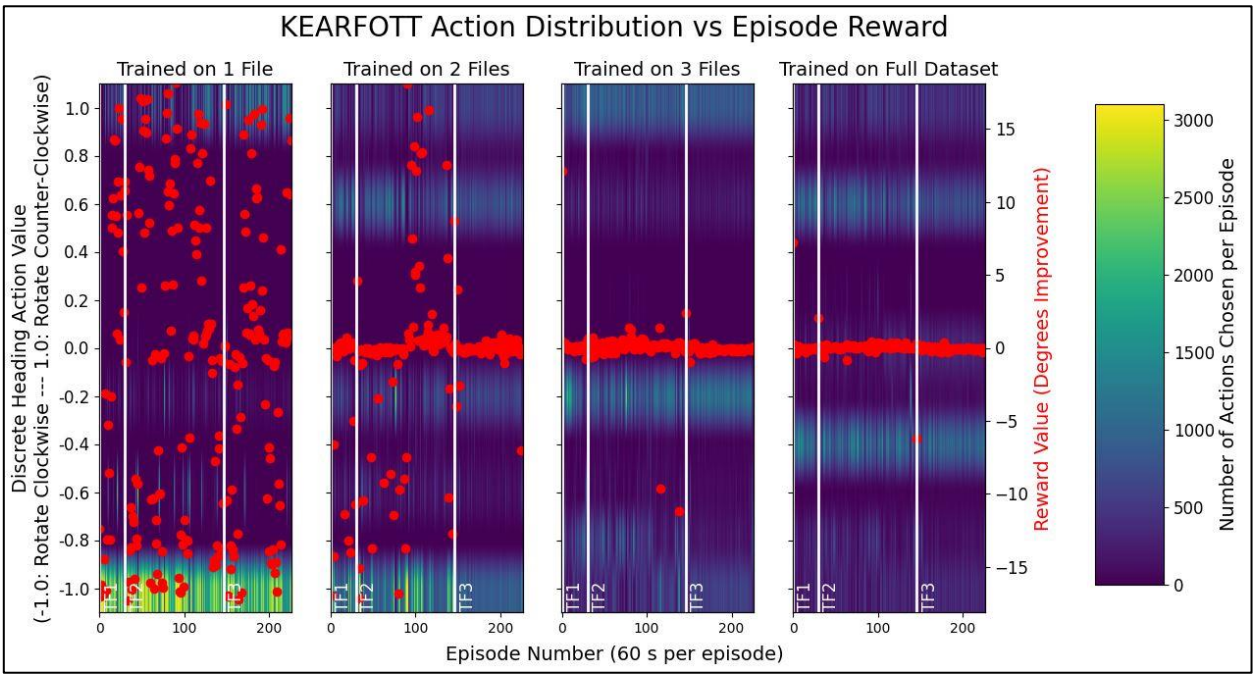

Figure 13. Kearfott Action Distribution vs Reward Colormap with No Action Scaling.

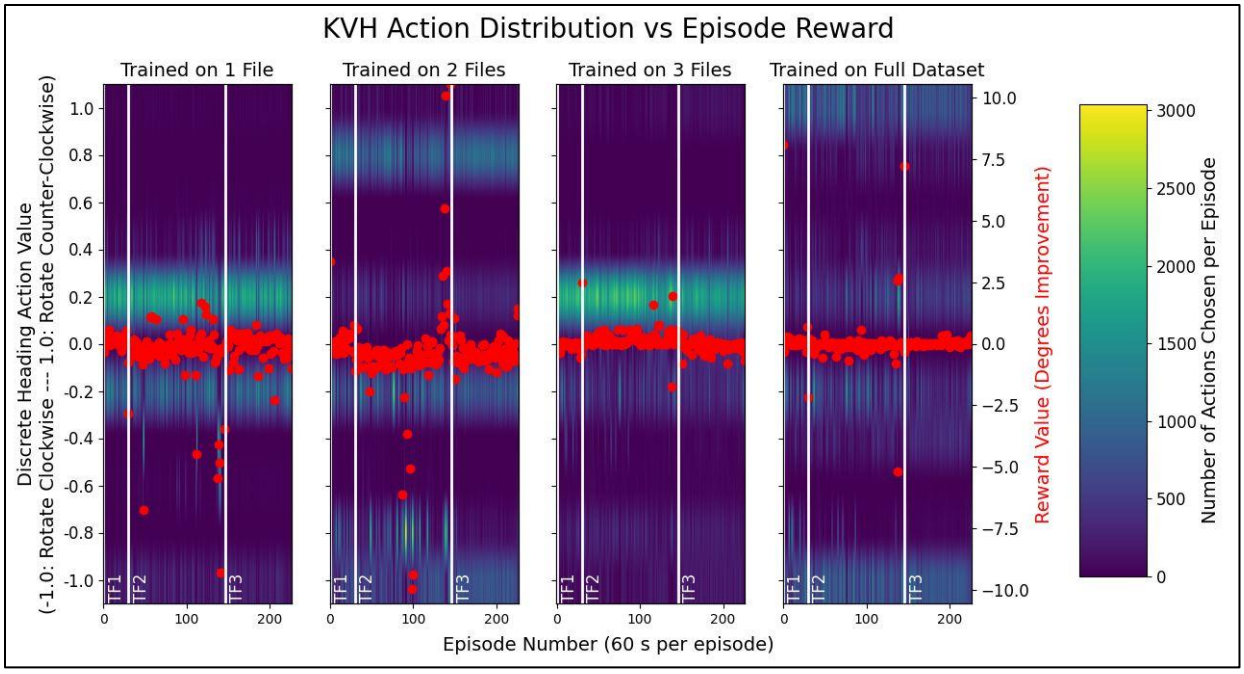

Figure 14. KVH Action Distribution vs Reward Colormap with No Action Scaling.

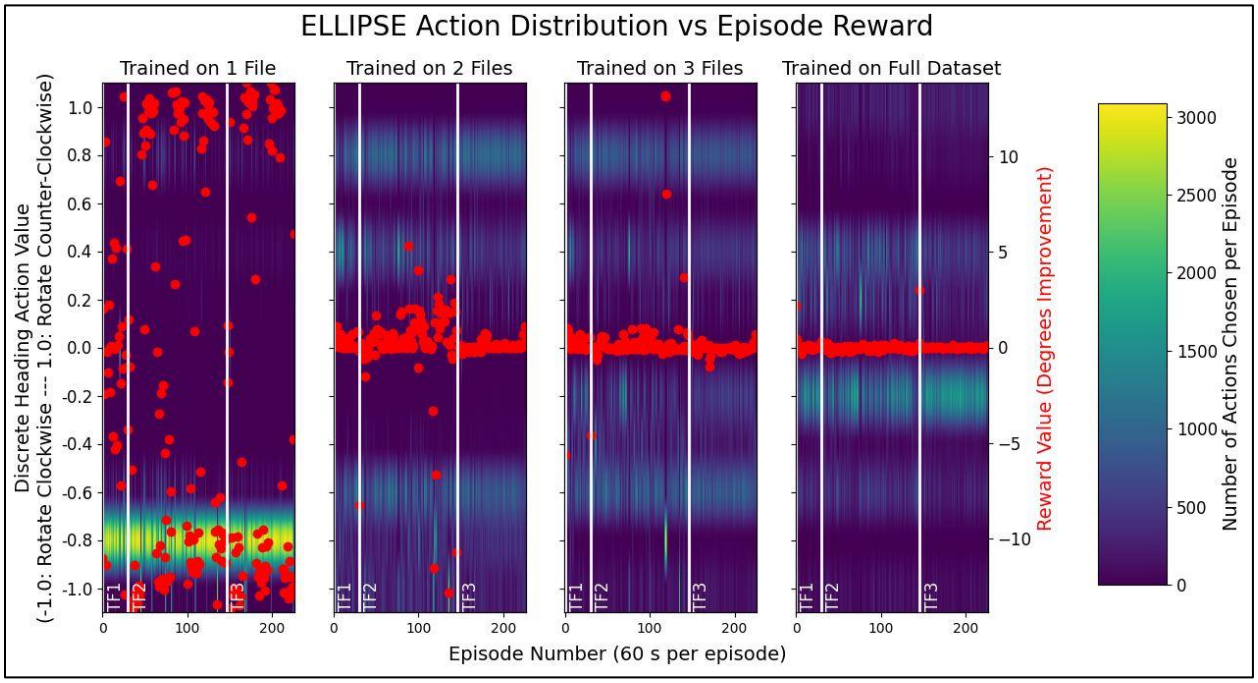

Figure 15. Ellipse Action Distribution vs Reward Colormap with No Action Scaling.

Figure 16 compares the position solutions from the NavFil with Kearfott inertial data. The results with scaled and unscaled actions are shown for both the unaided and RL-aided NavFil, as well as the true positions from GPS (labeled “Aided”). This plot shows incredibly poor results when the actions are not scaled, due to the heavily-chosen clockwise rotation actions showed in the Action Distribution

plot in Figure 13. While the reason these actions were chosen so often is not entirely known, it is likely caused by not enough learning, which in turn causes a uniform probability distribution of actions that causes the system to pick the same index. The positive reward values in the action distribution plot's (Figure 13) first subplot likely occur after a full 360° rotation, causing its heading to coincidently sync up perfectly with the ground truth heading.

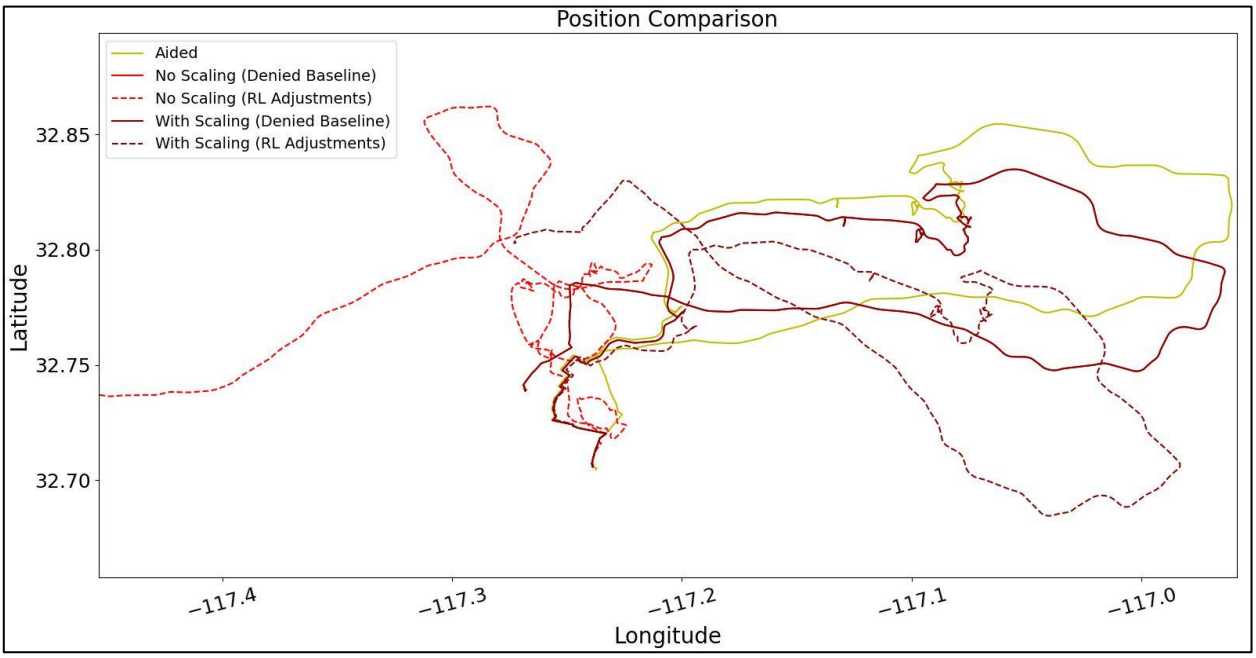

Figure 16. RL vs Denied Position Comparison with No Action Scaling.

4.4.1 Summary of Action Scaling Results

Removing A_{test_scale} causes terrible results when trained on a small amount of data, exemplified in the position comparison of Figure 16. This result can be expected since the model may not have had enough time to learn. It may have applied large actions since applying the largest actions during testing produced the highest potential for large rewards. Consistently across all sensor plots, continued training massively decreased this error, and shows stabilization when training on the entire dataset. It may be learning that it can get consistently 'less negative' errors by spreading out its action distribution and/or picking actions closest to zero. While this may show that the system is learning and updating correctly, the downside is consistently 'slightly negative' rewards result in position errors slightly worse than the GNSS-Denied baseline.

4.5 RESULTS TABLE

Table 3. Denied vs RL Cumulative Distance Error.

Scaling Mode	Sensor	Number of Training Files	Test File 1			Test File 2			Test File 3			Average Improvement Across All Test Files
			Denied	RL	Percent Improvement	Denied	RL	Percent Improvement	Denied	RL	Percent Improvement	
Without Action Scaling	Kearfott	1	40.22	2580.5	-6316%	1499.05	13462.5	-798%	1072.77	65549.3	-6010%	-3024%
		2	40.22	1046.6	-2502%	1499.05	14487.4	-866%	1072.77	5326.55	-397%	-699%
		3	40.22	36.52	9%	1499.05	1290.85	14%	1072.77	2792.35	-160%	-58%
		Full	40.22	65.89	-64%	1499.05	1207.86	19%	1072.77	2268.81	-111%	***-36%***
	KVH	1	59.13	23.91	60%	1845.27	4323.28	-134%	841.07	1012.65	-20%	-95%
		2	59.13	120.31	-103%	1845.27	11175.6	-506%	841.07	12321.4	-1365%	-760%
		3	59.13	82.62	-40%	1845.27	630.54	66%	841.07	2522.6	-200%	-18%
		Full	59.13	34.63	41%	1845.27	2382.49	-29%	841.07	903.58	-7%	***-21%***
	Ellipse	1	190.4	2572.4	-1251%	4362.92	15580.6	-257%	3011.43	56158.0	-1765%	-882%
		2	190.4	75.01	61%	4362.92	2752.89	37%	3011.43	2171.12	28%	34%
		3	190.4	113.47	40%	4362.92	2957.85	32%	3011.43	4172.42	-39%	4%
		Full	190.4	196.55	-3%	4362.92	4696.92	-8%	3011.43	3111.34	-3%	***-6%***
With Action Scaling	Kearfott	1	40.22	133.56	-232%	1499.05	4601.58	-207%	1072.77	7560.85	-605%	-371%
		2	40.22	40.6	-1%	1499.05	2649.16	-77%	1072.77	2474.59	-131%	-98%
		3	40.22	65.69	-63%	1499.05	560.61	63%	1072.77	3356.62	-213%	-52%
		Full	40.22	58.78	-46%	1499.05	1055.32	30%	1072.77	2214.43	-106%	***-27%***
	KVH	1	59.13	31.14	47%	1845.27	2493.23	-35%	841.07	1927.06	-129%	-62%
		2	59.13	50.08	15%	1845.27	3749	-103%	841.07	6463.21	-668%	-274%
		3	59.13	62.59	-6%	1845.27	1620.07	12%	841.07	1005.01	-19%	2%
		Full	59.13	55.5	6%	1845.27	2053.14	-11%	841.07	870.61	-4%	***-9%***
	Ellipse	1	190.4	53.17	72%	4362.92	1693.14	61%	3011.43	4454.02	-48%	18%
		2	190.4	161.26	15%	4362.92	3595.53	18%	3011.43	2490.99	17%	17%
		3	190.4	187.86	1%	4362.92	4453.03	-2%	3011.43	4048.97	-34%	-15%
		Full	190.4	196.85	-3%	4362.92	4535.61	-4%	3011.43	3230.49	-7%	***-5%***

† *Table colors match colors used in path/error figures and are used to separate each sensor*

†† *Percent Improvement take the absolute difference between RL and Denied error divided by Denied error. Average Improvement first takes the sum of RL and Denied error across each testing file, then calculates percent improvement*

Looking at 'No Action Scaling' results, across all sensors, the average improvement across all test files after receiving only one training file is abysmal. This may show that training with a single file is not nearly enough data to learn anything relevant. It appears that KVH trained on two files performed worse than just one, which can illustrate that two files are still not enough, and the better results from one training file are likely coincidental. The 'No Action Scaling' results help demonstrate that the system appears to understanding of how its actions affect rewards, and that while large actions give the highest potential for high rewards, consistent high actions will quickly lead the RL NavFil heading in the wrong direction, causing extremely large errors.

Training on the full dataset, although still negative, reduced the error across all sensors. Note that in some cases (KVH and Ellipse without action scaling, KVH without action scaling), it appears that

improvement declines from 2-3 trainings to the full dataset. This might illustrate the relationship between sensor accuracy and possible overfitting. Since the Kearfott is the most accurate sensor, it is more difficult to determine which actions yield the highest rewards, thus more training may likely be required. The Kearfott steady improvement as it continues to train on more data fits this hypothesis. The KVH is the next best sensor, so less training may be needed. It looks like 3 training files get the best results, meaning it may have learned as much as possible, but giving it more too much data after that caused overfitting. Finally, the Ellipse showed maximum results after only 2 files, and started to decrease as it received more data. This consistent pattern further illustrates the relationship between sensor accuracy, required training time, and peak results, and may be a result of overfitting.

This page is intentionally blank.

5. CONCLUSIONS

There are many potential reasons for these less than ideal results: a better reward function that considers more variables other than just heading; a deeper observation space that can look over more dimensions; using an asynchronous algorithm that learns using multiple training files at once. However, we ultimately did not have the manpower/hours to solve the problem, or continue to solve it in the future. The robust infrastructure exists to continue exploring this topic down the road.

This page is intentionally blank.

REFERENCES

[1] A. N. O. Bozeman, "Drift Improvement with Reinforcement Training – Inertial Sensors – Year 1," DTIC, San Diego, 2022.

[2] Kearfott Corporation, KI-4901S High Performance SEANAV INS/IMU Kit, Kearfott Corporation, 2012.

[3] KVH Industries Inc., "P-1725 IMU Photonic Inertial Measurement Unit," Middletown, Rhode Island, 2021.

[4] SBG Systems SAS, "Ellipse Series Hardware Manual," SBG, Santa Ana, California, 2020.

[5] Sparton, "Product Data Sheet: AHRS-M2," Sparton, DeLeon Springs, Florida, 2017.

This page is intentionally blank.

INITIAL DISTRIBUTION

84310	Technical Library/Archives	(1)
71780	E. Bozeman	(1)
71780	M. Nguyen	(1)
71740	J. Onners	(1)
71740	M. Alam	(1)

Defense Technical Information Center
Fort Belvoir, VA 22060–6218 (1)

This page is intentionally blank.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-01-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
November 2022	Final	

4. TITLE AND SUBTITLE

Drift Improvement with Reinforcement Training – Inertial Sensors - Year 2

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHORS

Eric Bozeman
Minhdao Nguyen
NIWC Pacific

Jeffrey Onners
Mohammad Alam
NIWC Pacific

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

NIWC Pacific
53560 Hull Street
San Diego, CA 92152–5001

8. PERFORMING ORGANIZATION REPORT NUMBER

TR-3294

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Innovative Science and Engineering
53560 Hull Street
San Diego, CA 92152-5001

10. SPONSOR/MONITOR'S ACRONYM(S)

NISE

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report is a follow-on to TR-3267, the technical report titled "Drift Improvement with Reinforcement Training - Inertial - Year 1" that described the work and results from the first year of the Drift Improvement with Reinforcement Training – Inertial (DIRT-I) project. This report covers the work and results completed under the second year of this project, which focused on the use of several different inertial sensors with a wide range of performances. The overall goal was to show that the DIRT-I system could be used with any inertial sensor, and to determine the effectiveness of the DIRT-I system with lower quality inertial sensors. This report shows that there is potential to use the DIRT-I system to improve the positional error of inertial sensors without access to corrections from external sensors such as GNSS. However, several changes to the system and much more research into the effectiveness of the RL algorithm(s) used, would be required.

15. SUBJECT TERMS

DIRT-1; RL system; INS; GNSS; GPS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Eric Bozeman
U	U	U	SAR	46	19B. TELEPHONE NUMBER *(Include area code)* (619) 553-1044

This page is intentionally blank.

This page is intentionally blank.

Naval Information Warfare Center (NIWC) Pacific
San Diego, CA 92152-5001