2022-07-22

Precision Maritime Localization and Landing with Real-time Kinematic GNSS

Alexander Jordan  
*Brigham Young University*, a.jordan6955@gmail.com

Matthew Kent Rydalch  
*Brigham Young University*

Tim McLain  
*Brigham Young University Mechanical Engineering Department*, mclain@byu.edu

Michael Williamson Tabango  
*Brigham Young University*

Follow this and additional works at: https://scholarsarchive.byu.edu/studentpub

Part of the Electrical and Computer Engineering Commons

BYU ScholarsArchive Citation  
Jordan, Alexander; Rydalch, Matthew Kent; McLain, Tim; and Williamson Tabango, Michael, "Precision Maritime Localization and Landing with Real-time Kinematic GNSS" (2022). *Student Works*. 345.  
https://scholarsarchive.byu.edu/studentpub/345

This Conference Paper is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Student Works by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.
This paper presents a highly effective method for UAV precision shipboard localization and landing using Real-time Kinematic Global Navigation Satellite System (RTK GNSS). To assess the feasibility of RTK GNSS for this use case we explicitly exclude vision-based localization techniques which are most often presented in the literature. Instead, the methods presented in this paper use only RTK GNSS with an inertial measurement unit aboard the landing pad to estimate the state of the boat and the relative position of the UAV with respect to the boat. We use a continuous-discrete extended Kalman filter combined with a complementary filter for state estimation. A custom state machine is then implemented that ensures vehicle descent inside of a conical corridor above the landing pad before finally touching down at a precise location on the landing target. We evaluate our proposed methods in a real-world hardware application that includes a landing target with boat-like motion. Consistent landing accuracy below 10 cm is achieved with the proposed methods.

I. Introduction

Precision landing of unmanned aerial vehicles (UAVs) is becoming an increasingly important topic in the field of robotics. Precise landings have several potential applications such as autonomous docking and charging, package delivery, surveillance and more.

Of particular interest is the autonomous landing of UAVs on platforms in maritime environments. Such an environment presents unique challenges with dynamic platform movement in sea conditions and the negative consequences of a failed landing in a body of water. In this work we develop a multirotor landing system for maritime environments that utilizes Real-time Kinematic Global Navigation Satellite System (RTK GNSS) as its primary navigation sensor. This research does not utilize vision-based localization methods that have been the standard in previous literature for localizing in such a landing application. This allows us to show the reliability of RTK GNSS as the primary sensor for precise landings on a boat-like landing platform. In this paper we also discuss our landing system in hardware tests that achieve an average accuracy of below 10 cm. Our contributions include the following flight tested implementations:

- A continuous-discrete extended Kalman filter for boat state estimation with RTK GNSS measurement models
- A novel landing method that defines a conical descent corridor and enforces an accurate landing
- Demonstration of our landing method in hardware on a moving platform with boat-like dynamics

The remainder of this paper proceeds as follows. In Section II we review related work and outline some of our specific contributions that were not previously addressed in the literature. In Section III we give an overview of our technical approach. Section IV provides a description of boat state estimation with a continuous-discrete Kalman filter and complementary filter. Section V describes the state machine which manages our landing method and in Section VI we provide an evaluation of our proposed methods. Finally, in Section VII we draw conclusions based on our results.

II. Related Work

There is an abundance of literature focused on precise UAV landings. While many researchers have investigated a variety of subjects related to boat-landing UAVs, results and methods of evaluation are varied. Additionally, many landing solutions are for land-based, airborne landings, or stationary landings that lack the boat dynamics expected at sea. While still valuable, these solutions only have demonstrations in simulation [1], indoors [2], or in non-sea-like conditions [3][4].
A majority of maritime landing research has focused on vision-based methods [5-7]. These solutions are promising due to their accurate relative measurements, fast update rates, and independence from any communication with the landing pad [8]. Cameras are also preferable because they are often cheaper than traditionally expensive sensors such as an RTK module [8]. In [9] researchers were able to attain successful outdoor landings with computer vision, but they assumed a known constant velocity of the landing vehicle. Researchers in [5] were able to find a real-world solution to land on a boat-like target using image-based visual servoing. In [6], researchers demonstrated a landing technique that mimics that of an actual pilot by utilizing vision to track a gyro-stabilized horizon bar. However, in their work they explicitly do not estimate the boat attitude just as human pilots are taught not to track such motions. In this paper we estimate the boat attitude and utilize it to make decisions about when it is safe to land and to transform spatial offsets of RTK receivers to the center of the landing pad.

While vision has many advantages, it also has its drawbacks. In order to use vision it is assumed that the vehicle can be maneuvered close enough to the landing target to identify features used for localization. This is often achieved by augmenting the system with standard GNSS [5]. However, if the vision sensor fails, standard GNSS is not sufficient to support a precise landing. Potential failure modes of vision sensors in maritime environments include target occlusion when using fiducial markers, image distortion due to ocean spray on the lens and loss of target line-of-sight in foggy or low-light conditions.

A broad range of landing accuracy has been reported in the literature with several reaching accuracy below 1 m in hardware applications [5,6,10,11]. Others reported sub-cm accuracy, but only showed results in simulation [12]. In this paper we sought to achieve a high level of accuracy consistent with the accuracy of RTK GNSS which has been shown to be within 1-3 cm when the baseline is relatively short (below 20 km) [13]. With this in mind we have developed a landing system that can consistently land within a 10 cm radius in real-world hardware applications.

### III. Overview of Approach

The goal of this paper is to develop a reliable method for precision landing in dynamic maritime environments. Prerequisite to this achievement is an accurate estimate of the boat state and the UAV’s position relative to the boat. In Section IV we describe our method of estimation based on an extended Kalman filter and complementary filter that combine measurements from RTK GNSS and an inertial measurement unit (IMU) attached to the boat.

These state estimates are then used inside the state machine defined in Section V to determine what step of the landing process to execute and to produce velocity commands for the vehicle which are subsequently communicated to a commercial autopilot. Pixhawk, a well-tested and reliable autopilot, along with the PX4 autopilot software were chosen to handle low-level control of the vehicle. MAVSDK-Python was used to exchange information with PX4. A block diagram summarizing our landing system is given in Figure 1.

![Fig. 1 Landing System block diagram](image)

### IV. Estimation

Estimation of several states of the boat and UAV is essential for an accurate landing. The relative position of the boat with respect to the UAV along with the inertial velocity of both vehicles provide the necessary information to generate velocity commands for the UAV. Boat roll and pitch help determine whether the platform is sufficiently level for a landing attempt and the yaw of the boat allows for landing of the UAV with a specific heading relative to the boat. The boat states are represented below as
\[
\begin{align*}
 x_b &= [p^T_{b/m}, v^T_{b/m}, \Theta^T_b]^T \\
 \Theta_b &= [\phi_b, \theta_b, \psi_b]^T
\end{align*}
\]  

where \(x_b\) are the boat states, \(p^T_{b/m}\) is the position of the boat with respect to the multirotor in the inertial frame, \(v^T_{b/m}\) is the velocity of the boat in the inertial frame, and \(\Theta_b\) represents the Euler angles of the boat. Estimation of these states is primarily carried out by an extended Kalman filter with the exception of boat roll and pitch which are estimated with a complementary filter.

### A. Extended Kalman Filter

An extended Kalman filter (EKF) is used to estimate the UAV-to-boat relative position, boat and multirotor velocities and boat heading. Specifically a continuous-discrete extended Kalman filter as shown in [14] was implemented. This method allowed us to update the dynamics as a continuous system with several updates in-between discrete measurement updates. The EKF state vector is given by

\[
 x = [p^T_{b/m}, v^T_{i/m}, \psi_b, v^T_{b/b}]^T,
\]

where \(v^T_{i/m}\) is the multirotor velocity in the inertial frame, \(\psi_b\) is the heading of the boat in the inertial frame and \(v^T_{b/b}\) is the boat velocity in its own body frame.

#### 1. Model Propagation

In between sensor readings the system state is updated frequently with the following system input

\[
 u = [y^T_{accel}, y^T_{gyro}, \hat{\phi}_b, \hat{\theta}_b]^T
\]

where \(y_{accel}\) and \(y_{gyro}\) are accelerometer and gyro measurements from the IMU and \(\hat{\phi}_b\) and \(\hat{\theta}_b\) are roll and pitch estimates received from the complementary filter (Section IV.B).

These inputs are used to propagate the dynamic model of our system as described by

\[
 \left[ \begin{array}{c}
 \dot{p}^i_{b/m} \\
 \dot{v}^i_{i/m} \\
 \dot{\psi}_b \\
 \dot{v}^b_b
\end{array} \right] =
 \left[ \begin{array}{c}
 R^i_b(\hat{\phi}_b, \hat{\theta}_b, \psi_b)v^i_b - v^i_m \\
 0 \\
 T_{gyro}^\phi(y_{gyro}) \\
 y_{accel} + R^i_b(\hat{\phi}_b, \hat{\theta}_b, \psi_b)g^i - y_{gyro} \times v^b_b
\end{array} \right]
\]

where \(R^i_b\) is the rotation matrix between the boat body frame and inertial frame, \(g^i\) is the gravity vector in the inertial frame and \(T_{gyro}^\phi\) transforms body angle rates as given by the rate gyros to Euler angle rates.

As shown in [14] the transformation \(T_{gyro}^\phi\) is given by

\[
 T_{gyro}^\phi = \sin \frac{\hat{\phi}_b}{\cos \hat{\theta}_b} y_{gyro,q} + \cos \frac{\hat{\phi}_b}{\cos \hat{\theta}_b} y_{gyro,r}
\]

This model is propagated each time an IMU measurement is received according to the system and uncertainty dynamics specified by

the propogate and update step don’t actually match the uav book exactly and what we do in the code is a bit different

\[
 \dot{x} = f(x, u)
\]

\[
 \dot{P} = AP + PA^T + Q
\]

where \(A\) is the Jacobian \(\frac{df}{dx}\), \(P\) is a covariance matrix describing the state uncertainty and \(Q\) is the process noise.

Note that although the input is indeed discrete, the sample rate of the IMU (500 Hz), is significantly faster than RTK GNSS sample rates (5 Hz) and therefore allows for application of the continuous-discrete EKF.
2. Measurement Updates

Our RTK GNSS configuration which will be shown in Section VI provides several useful measurement updates with measurements that represent the multirotor-boat relative position, boat heading and inertial velocities of both vehicles. The measurement models have been derived as

\[ y_{rtk, relpos} = -p_{b/m} - R_i^b \delta^b_{\text{antenna}} \]  
\[ y_{gps, mvel} = v_i^m \]  
\[ y_{gps, bvel} = R_i^b v_b^b - \omega_b, LPF \times \delta^b_{\text{antenna}} \]  
\[ y_{rtk, compass} = \psi_b \]  

where \( \delta^b_{\text{antenna}} \) is the spatial offset between the RTK base receiver on the boat and the center of the landing pad as expressed in the boat frame.

The continuous-discrete EKF allows each measurement update to be handled separately. This means that each update is conducted as its corresponding measurement is received. When a measurement from the \( i^{th} \) sensor is received, the Kalman update is

\[ L_i = P^{-} C_i^T (R_i + C_i P^{-} C_i^T)^{-1} \]  
\[ x^+ = x^- + L_i (y_i(t_n) - C_i x^-) \]  
\[ P^+ = P^- - L_i C_i P^- \]

where \( L \) is the Kalman Gain, \( C_i \) is the measurement Jacobian of the \( i^{th} \) measurement \( \frac{dy_i}{dx} \), \( R \) is the covariance of the sensor noise and superscripts indicate states and covariances before (-) and after (+) the measurement update.

B. Complementary Filter

The complementary filter fuses measurements from the IMU to estimate the attitude of the boat. The IMU measurements are a combination of rate gyro and accelerometer measurements and can be modeled as followed.

1. Rate Gyros

The rate gyros measure the body angular rates of the boat along with a bias and noise terms as

\[ y_{gyro,x} = p + \beta_p p + \eta_p \]  
\[ y_{gyro,y} = q + \beta_q q + \eta_q \]

where \( p \) and \( q \) are the body angular rates about the boat x and y axes respectively, \( \beta_p \) and \( \beta_q \) are slowly varying bias terms and \( \eta_p \) and \( \eta_q \) are zero mean Gaussian noise terms.

When pitch and roll are small \( \phi \) and \( \theta \) can be approximated as the body rates \( p \) and \( q \)

\[ \dot{\phi} \approx p \]  
\[ \dot{\theta} \approx q \]

With this small angle approximation, integration of the rate-gyros yields

\[ \dot{\phi}_{\text{gyro}} = \phi + \beta_\phi(t) \]  
\[ \dot{\theta}_{\text{gyro}} = \theta + \beta_\theta(t) \]

2. Accelerometers

The accelerometers can be modeled to measure the acceleration of the UAV with additional bias and noise. In addition, the accelerometers measure the force of gravity in all three axes as follows
\( y_{\text{accel,}x} = \dot{u} + qw - rv + g \sin \theta + \beta_x + \eta_x \)  
(22)

\( y_{\text{accel,}y} = \dot{v} + ru - pw + g \cos \theta \sin \phi + \beta_y + \eta_y \)  
(23)

\( y_{\text{accel,}z} = \dot{w} + pv - qu + g \cos \theta \cos \phi + \beta_z + \eta_z \)  
(24)

where \( u, v \) and \( w \) are velocities and \( p, q \) and \( r \) are angular velocities both about the boat \( x, y \) and \( z \) axes respectively.

Assuming that the boat is stationary (i.e. \( \dot{u} = \dot{v} = \dot{w} = p = q = r = 0 \)), the accelerometers will measure only the gravity vector plus noise and bias. With this assumption, we can use the direction of the gravity vector to estimate the pitch and roll of the boat with the following

\[
\hat{\phi}_{\text{accel}} = \tan^{-1} \left( \frac{y_{\text{accel,}y} - \beta_y}{y_{\text{accel,}z} - \beta_z} \right) = \phi + v_\phi + \eta_\phi
\]

\[
\hat{\theta}_{\text{accel}} = \sin^{-1} \left( \frac{y_{\text{accel,}x} - \beta_x}{g} \right) = \theta + v_\theta + \eta_\theta
\]

(25)

(26)

The terms \( \dot{u}, \dot{v}, \dot{w}, p, q, r \) which we assumed to be zero are all high frequency signals, thus our approximation with the accelerometers will be most accurate when roll and pitch are changing at low frequencies. Rate-gyro measurements, on the other hand, provide high-frequency content with a low-frequency bias term. To extract the desired information from both sensors, a low-pass filter is applied to the accelerometers and a high-pass filter is applied to the rate-gyros.

**Algorithm 1** Complementary Filter

1: \( \text{if IMU Measurement Received then} \)
2: \( \hat{\Theta} = \begin{bmatrix} \hat{\phi} \\ \hat{\theta} \end{bmatrix}^T \)  \hspace{1cm} \( \text{Current roll and pitch estimate} \)
3: \( \hat{\phi}_{\text{accel}} = \tan^{-1} \left( \frac{y_{\text{accel,}y} - \beta_y}{y_{\text{accel,}z} - \beta_z} \right) \)
4: \( \hat{\theta}_{\text{accel}} = \sin^{-1} \left( \frac{y_{\text{accel,}x} - \beta_x}{g} \right) \)
5: \( \hat{\Theta} = \begin{bmatrix} \hat{\phi}_{\text{accel}} - \hat{\phi} & \hat{\theta}_{\text{accel}} - \hat{\theta} \end{bmatrix}^T \)
6: \( \hat{\beta} \leftarrow \hat{\beta} - \Delta t \ast k_i \ast \hat{\Theta} \)  \hspace{1cm} \( \text{Gyro bias estimate} \)
7: \( \hat{\Theta} = (T_{\text{gyro}} \ast y_{\text{gyro}} - \hat{\beta}) + k_p \ast \hat{\Theta} \)
8: \( \hat{\Theta} \leftarrow \hat{\Theta} + \Delta t \ast \hat{\Theta} \)
9: \( \text{end if} \)

As shown in [15], the complementary filter can be applied in a feedback configuration that includes an integrator to reject the rate-gyro biases. The implementation of this feedback complementary filter is given in Algorithm 1. The performance of this filter is given in Figure 2. To measure the accuracy of the complementary filter, the boat was

![Fig. 2 Complementary filter performance compared with motion capture ground truth](image-url)
manually maneuvered by the lever-arm shown in Figure 6 while being tracked live by an OptiTrack motion capture system. The filter output was compared with motion capture estimated attitude and a maximum error of less than 1 degree was observed.

V. State Machine

To ensure a successful landing a state machine was implemented to manage each step of the landing process. The possible states in the state machine are rendezvous, descend, go-around and land. Throughout the entire landing process a cone shaped volume is defined above the landing pad which governs the execution of each state as shown in Figure 4. This conical threshold is referred to in this paper as the descent cone. This method allows for the inherent variations in relative position that come when landing on dynamic maritime vehicles while also enforcing increasingly stringent lateral tracking as the vehicle descends toward its target. This method also constrains the direction of landing approach to be from above the pad and implements safety measures to avoid unintended collisions with the platform when desired waypoints are missed. A summary of the state-machine architecture is given in Figure 3.

Fig. 3 State machine flowchart

A. Rendezvous

In the rendezvous state the vehicle tracks the boat at a specified constant height above the platform. To pass the rendezvous state the vehicle must maintain its position within a specified area for a given amount of time and therefore is managed by both a spatial and time threshold. The spatial threshold for this state is defined as a slice of the descent cone about the rendezvous height. The conical shape of the descent cone introduces greater leniency in lateral tracking high above the landing target where only moderate accuracy is required before descent begins. In addition, the slice of the cone is chosen to be sufficiently large to allow for longitudinal variations caused by heaving motion of the boat.

B. Descend

In the descend state the vehicle begins to descend within the descent cone. In this state the descent cone defines the space within which the vehicle is allowed to descend. If at any point the multirotor exits the conical threshold it will no longer descend, but will instead return laterally to the descent cone. This behavior ensures that the vehicle does not attempt to land unless it is closely tracking the target landing location. However, this behavior presents a potential danger to the vehicle if it is close to the landing pad. In such a situation the wave-like motions of the landing pad could cause a collision with the vehicle before a landing is desired. To prevent such a situation the go-around state was introduced which is discussed below. If the vehicle exits the descent cone below some predetermined safe height, the go-around state is initiated.
Otherwise, if the vehicle successfully navigates to the bottom of the descent cone and maintains its position within a specified slice at the bottom of the cone for enough time the land state will be initiated.

1. Repulsive Force

Once the UAV has entered the descent cone it is desired to maintain the vehicle’s position within the cone for the entirety of its descent. To encourage this behavior, control inputs were generated so that the inside surface of the cone would act as a repulsive surface to the UAV. In practice this meant augmenting the velocity command with a vector that pointed toward the center of the cone.

Considering the commanded positions when the UAV is inside and outside of the cone leads to a simple solution for this repulsive command. When the UAV is inside of the cone, the vehicle is commanded towards the center of the landing pad plus a small height offset. This location is equivalent to the relative position estimated in Section IV.A plus the desired offset:

\[ p_{c, in} = p_{h/m}^i + \begin{bmatrix} 0 & 0 & -h_{\text{descend}} \end{bmatrix}^T \]  

(27)

where \( h_{\text{descend}} \) is a height offset in the inertial frame that causes the multirotor to hover just above the target before landing.

The command generated outside of the cone is similar, except the offset height is adapted so that the vehicle will return laterally to the cone, rather than continue to descend. This position is given by

\[ p_{c, out} = p_{h/m}^i + \begin{bmatrix} 0 & 0 & -h_{\text{return}} \end{bmatrix}^T \]  

(28)

where \( h_{\text{return}} \) is the last recorded height inside the cone.

Since the velocity command outside the cone directs the vehicle horizontally towards the center of the cone, we can use it to generate a pseudo-repulsive force. A linear transition between the command inside the cone and outside the cone provides the desired behavior. The transition begins at a predefined distance from the cone surface and ends at the surface. The result is an inner and outer conical surface with a small area in between where the commanded velocity is a combination of the inside command (pointed toward the target) and outside command (pointed towards the centerline). Thus, when in-between these two surfaces, a component of the commanded velocity tends to push the vehicle away from the outer surface and towards the cone centerline. In addition, this gradual transition also eliminates a discontinuity in the velocity command at the surface of the cone.

With a predetermined inner and outer radius, the commanded position at any location during the descent state can be described by

\[ p_c(r) = \begin{cases} 
  p_{c, in} & r < r_{in} \\
  p_{c, out} & r > r_{out} \\
  k_{in} p_{c, in} + k_{out} p_{c, out} & r_{in} < r < r_{out} 
\end{cases} \]  

(29)

\[ k_{in} = 1 - \frac{1}{t} (r - r_{in}) \]  

(30)

\[ k_{out} = \frac{1}{t} (r - r_{in}) \]  

(31)

where \( r_{in} \) and \( r_{out} \) are inner and outer radii at a given height of the cone, \( r \) is the vehicle’s lateral position relative to the cone centerline and \( t \) is the thickness of the transition period given by \( r_{out} - r_{in} \).

This position is relative to the desired location of the vehicle, therefore, rather than commanding the position directly, it can be used to generate a velocity command and passed to the autopilot for low-level control.

C. Go-around

The go-around state is initiated when the vehicle exits the descent cone close to the pad. This is determined by a pre-defined safe height. Upon exiting the cone, the vehicle is commanded to return to the cone at the safe height. Similar to the rendezvous state, a slice of the cone is defined as a threshold about the safe height and is required to be held for a time before continuing to descend.
D. Land

During the land state the vehicle is commanded to a position at the center of the landing pad. To ensure a downward commanded velocity while on the pad, the actual commanded position is a few centimeters below the top of the landing pad. Upon landing, an operator manually disarms the vehicle.

VI. Evaluation

A. Experimental Setup

To evaluate our landing system, we prepared hardware for outdoor flight tests. The vehicle used for these flight tests was built on a DJI Flamewheel F450 frame equipped with a Pixhawk 4 mini autopilot, Odroid XU4 onboard computer, and a UBLOX ZED-F9P RTK module as shown in Fig 5.

To mimic conditions of a barge at sea, a landing platform was attached to a lever arm that allows an operator to manually generate pitch, roll and heave motions. The platform was placed in the back of a truck to add translational
motion to the platform (Figure 6). The landing platform was equipped with its own Odroid XU4 for an onboard computer, an IMU and two U-Blox ZED-F9P modules: one for the RTK base station and one for RTK compassing.

With the above hardware prepared a set of consistent test landings were conducted outdoors. Operators were directed to maneuver the vehicle at approximately 4 mph (1.8 m/s) with pitch and roll inputs at a period of 3 s and magnitude of ±10°. However, since our boat hardware was human operated, these values were given as qualitative directions for the operators and precise values for each were measured and are reported in Section VI.B.

B. Results

Ten flight tests were conducted during our evaluation and each test resulted in a successful landing with an average landing error of 7.34 cm. Only one of the ten flights fell outside of our goal accuracy of 10 cm. A summary of landing performance is shown in Table 1. To evaluate our results, landing error, time of flight, and starting distance were recorded. Each flight was measured from the moment the operator switches the vehicle to autonomous mode, to the moment of touch-down.

As described above, motions of the landing platform were controlled by human operators and, therefore, were not identical across all test sets, although each test was qualitatively similar. For this reason, boat motions were recorded during testing. The average values for the platform motions are given in Table 2.
Table 1  Combined flight test data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Mean</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy, cm</td>
<td>7.34</td>
<td>2.38</td>
</tr>
<tr>
<td>Time of flight, s</td>
<td>29.15</td>
<td>9.92</td>
</tr>
<tr>
<td>Starting Distance, m</td>
<td>3.87</td>
<td>0.83</td>
</tr>
<tr>
<td>Success rate, %</td>
<td>100</td>
<td>N/A</td>
</tr>
<tr>
<td>Total flights</td>
<td>10</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 2  Base Motion Data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Mean</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base Velocity, m/s</td>
<td>1.42</td>
<td>0.09</td>
</tr>
<tr>
<td>Base Acceleration m/s²</td>
<td>0.76</td>
<td>0.21</td>
</tr>
<tr>
<td>Pitch Rate, deg/s</td>
<td>5.04</td>
<td>2.75</td>
</tr>
<tr>
<td>Pitch Amplitude, deg</td>
<td>7.75</td>
<td>1.38</td>
</tr>
<tr>
<td>Roll Rate, deg/s</td>
<td>1.95</td>
<td>1.15</td>
</tr>
<tr>
<td>Roll Amplitude, deg</td>
<td>6.31</td>
<td>1.40</td>
</tr>
</tbody>
</table>

VII. Conclusion

With RTK GNSS as the primary source of localization we have developed an accurate and reliable landing system for UAV maritime landings. This landing system includes a combination of a continuous-discrete Kalman filter and a complementary filter for state estimation. We have also developed a systematic state machine that ensures a safe landing. This state machine introduces a three-dimensional conical volume that defines the space within which the UAV is allowed to descend toward the landing pad. In hardware landing applications we have demonstrated the reliability of our landing system with consistent landing success and an average accuracy below 10 cm.

The success of this landing system demonstrates the feasibility of RTK GNSS for localization in maritime landings. Potential future work includes increasing robustness by fusing RTK GNSS with other common sensors such as vision.

VIII. Acknowledgments

This project was funded by Lawrence Livermore National Laboratory under subcontract number B643355.

References


