# Guidance and Control Law Design for a Slung Payload in Autonomous Landing A Drone Delivery Case Study

Longhao Qian Silas Graham Hugh H.T. Liu

***Abstract*—This paper presents a novel guidance and control design for a parcel tethered to a drone in its delivery mission. The proposed proportional navigation based guidance, integrated with custom built control, enable to achieve autonomous payload landing. The novelty of the proposed method lies on two aspects: 1) the guidance law allows for soft landing; while 2) the path following control ensures the swing-free payload transportation that sets the solid foundation for landing. The development is verified by extensive simulations and further demonstrated by flight experiments.**

***Index Terms*—guidance, control, autonomous landing, drone delivery**

## I. Introduction

An Unmanned Aerial Vehicle (UAV) is a flying system that does not carry a human pilot for operation. The popularity of these vehicles has increased drastically in the past decade, and they start to become mainstream in public applications. They are often referred to simply as "drones", but are now formally recognized in Canada as a RPAS (Remotely Piloted Aerial System). The use of UAVs has spread to many applications and sectors in recent years due to their versatility, cost, and flexible operation characteristics. These applications range from their extensive commercial use in photography and surveying, to industrial applications like forest fire monitoring. One application that has received interest and attention in both industry and academia is the use of UAVs for payload delivery. A UAV can carry a payload in a variety of ways, which can broadly be categorized into three methods: 1) rigidly connected to the UAV body, 2) a cable slung payload, 3) a manipulator-arm carried payload. Among these choices, the slung payload configuration is the choice of this study. The slung payload configuration allows the vehicle to safely deliver the payload to the ground without the need for the vehicle to land. This may also improve the mission efficiency by allowing the vehicle to hover over the landing zone, instead of completing a full landing and takeoff procedure. However, there are difficulties associated with the slung payload configuration. For example, the position of the payload is more difficult to control accurately, due to wind, rotor downwash, and the swinging dynamics of the payload relative to the UAV. For these reasons, the slung payload method has generated research interest, and has been extensively studied in recent years - with focus on control, payload lifting, and swing-free trajectory tracking.

Most of the study of the UAV slung payload system focus on designing controllers to flying along a desired trajectory. Palunko *et al.* specified a trajectory for the UAV that will minimize the payload swing during transportation along the path [1]. Kui *et al.* proposed a sliding mode controller to track a given trajectory with the quadrotor using no explicit payload feedback, which worked well in simulation [2]. It is likely to encounter disturbances due to wind or the swinging motion of the payload. Therefore, Nictora *et al.* proposed a controller that demonstrates some robustness to payload disturbances [3]. Qian and Liu presented a path-following controller for the slung payload system which is robust against wind disturbances on the system [4]. Guerrero *et al.* presented a passivity based approach to suppress the swinging motion of the slung payload while tracking a trajectory [5]–[7]. Their approach was not dependent on the swing angle, and they found that the swing motion was suppressed along the flight trajectory. In terms of landing, although research on quadrotor landing has been covered, such as [8]–[11], to the best knowledge of the authors of this paper, the challenge of landing of slung-payload has not been well studied. A relevant work by Goodarzi [12] uses a variable length cable to lower the payload to the ground shown by simulation, followed by experimentation by others using an additional payload-lowering mechanism [13].

This paper presents a novel guidance and control design for a slung-payload in its drone delivery mission. The proposed proportional navigation based guidance, integrated with custom built control, enable to achieve autonomous payload landing. The novelty of the proposed method is shown on two aspects: 1) a proportional navigation based guidance law is designed to allow for soft landing (zero velocity when touching down), and 2) the guidance law is integrated with a path following control, based on our previous work [4], ensures the swing-free payload transportation that sets the solid foundation for landing. The development is verified by extensive simulations and further demonstrated by flight experiments.

## II. PROBLEM STATEMENT

A representative UAV with slung payload configuration is shown in Fig. 1, where the equations of motion are derived

Longhao Qian is a PhD candidate of the Institute for Aerospace Studies, University of Toronto, 4925 Dufferin Street, Toronto, Canada

Silas Graham was a MASc student of the Institute for Aerospace Studies, University of Toronto, 4925 Dufferin Street, Toronto, Canada

Hugh H.T. Liu is a professor of the Institute for Aerospace Studies, University of Toronto, 4925 Dufferin Street, Toronto, Canada

based on the multi-body principle. The offset tether location means that the 'tether' or cable is not attached at the centre of mass of the vehicle, but rather at some offset distance on the vehicle's frame.

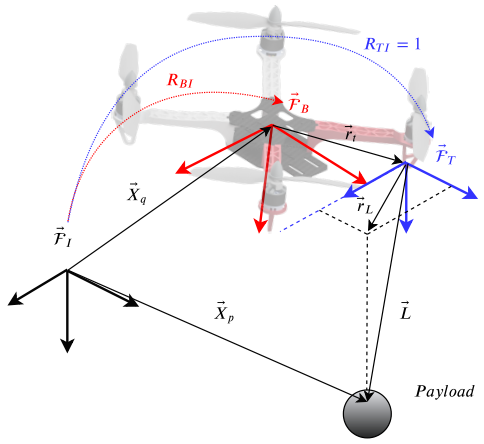



Fig. 1: Frame definition of the off-centered slung payload.

The position of the payload can be described as, $\vec{X}_p = \vec{X}_q + \vec{r}_t + \vec{L}$, leading to its expression in the inertial frame, denoted by the subscript $I$,

$$\boldsymbol{X}_{p,I} = \boldsymbol{X}_{q,I} + \boldsymbol{R}_{IB}\boldsymbol{r}_t + \boldsymbol{L}_I. \tag{1}$$

where $\boldsymbol{R}_{IB}$ is the rotation matrix from inertial frame to the body frame. We make several assumptions about the modeling for the control design purpose: 1) The tether is assumed taunted all the time. It is also rigid and of fixed length, similar to the work of Qian *et al.* in [14]. Therefore, the motion is constrained to two degrees of rotation at the fixed tether point, at the origin of $\mathcal{F}_T$. 2) The tether off-set distance $\boldsymbol{r}_t \approx \boldsymbol{0}$. In actual delivery mission, the cable is attached close to the center of mass (CM) of the quadrotor, so this is an reasonable assumption that decouples the quadrotor rotation and translation in modeling and control design. The tether is projected into the $\mathbf{t}_x - \mathbf{t}_y$ plane. We define the tether $\vec{L}$ in the inertial frame as, $\boldsymbol{L}_I = [\boldsymbol{r}_L \ L_{z,I}]^T$, $\boldsymbol{r}_L$ is the vector representing the projection of the tether into the $\mathbf{t}_x - \mathbf{t}_y$ plane, and $L_{z,I} = \sqrt{L^2 - \boldsymbol{r}_L^T\boldsymbol{r}_L}$ due to its constraint to two (rotational) degrees of freedom. The velocity state of the payload is obtained by taking the inertial time derivative,

$$\boldsymbol{v}_{p,I} = \boldsymbol{v}_{q,I} - \boldsymbol{R}_{IB}\boldsymbol{r}_t^{\times}\boldsymbol{\omega}_B + \boldsymbol{B}\boldsymbol{v}_L \tag{2}$$

where $\boldsymbol{\omega}_B = [p \ q \ r]^T$, and $\boldsymbol{B}$ is a kinematic relationship such that $\boldsymbol{B}\boldsymbol{v}_L = \dot{\boldsymbol{L}}_I$:

$$\boldsymbol{B} = \begin{bmatrix} \mathbf{1}_{2\times 2} \\ -\frac{\boldsymbol{r}_L^T}{\sqrt{L^2 - \boldsymbol{r}_L^T\boldsymbol{r}_L}} \end{bmatrix}. \tag{3}$$

Again knowing that the tether position $\boldsymbol{r}_t$ is fixed on the quadrotor, we differentiate Eq. (2) to obtain the inertial payload acceleration as,

$$\dot{\boldsymbol{v}}_p = \dot{\boldsymbol{v}}_q - \dot{\boldsymbol{R}}_{IB}\boldsymbol{r}_t^{\times}\boldsymbol{\omega}_B - \boldsymbol{R}_{IB}\boldsymbol{r}_t^{\times}\dot{\boldsymbol{\omega}}_B + \dot{\boldsymbol{B}}\boldsymbol{v}_L + \boldsymbol{B}\dot{\boldsymbol{v}}_L. \tag{4}$$

Here, we treat the dynamics of the quadrotor as a generic 6-DOF rigid body, and the dynamics of the payload are treated as a point mass. We also apply the assumption of $\boldsymbol{r}_t \approx \boldsymbol{0}$ in the dynamics model. Using the Newton-Euler formalism yields the following dynamic equations of motion for the system:

$$m_q\dot{\boldsymbol{v}}_q = m_q\boldsymbol{g}_I + \boldsymbol{T} + \boldsymbol{F} \tag{5}$$

$$\boldsymbol{J}_q\dot{\boldsymbol{\omega}}_B = -\boldsymbol{\omega}_B^{\times}\boldsymbol{J}_q\boldsymbol{\omega}_B + \boldsymbol{M}_B \tag{6}$$

$$m_p\dot{\boldsymbol{v}}_p = m_p\boldsymbol{g}_I - \boldsymbol{T} \tag{7}$$

$\boldsymbol{X}_T^T = \begin{bmatrix}\boldsymbol{v}_L^T & \boldsymbol{v}_q^T & \boldsymbol{r}_L^T & \boldsymbol{x}_q^T\end{bmatrix}$ is defined as the translational state. $\boldsymbol{X}_R = \{\boldsymbol{R}_{IB}, \boldsymbol{\omega}_B\}$ is defined as the attitude state. The complete state is $\boldsymbol{\mathcal{X}} = \{\boldsymbol{X}_T, \boldsymbol{X}_R\}$. Similarly, we define velocity subsets of $\boldsymbol{\mathcal{X}}$ as $\boldsymbol{V}^T = \begin{bmatrix}\boldsymbol{v}_L^T & \boldsymbol{v}_q^T\end{bmatrix}$. The input to the system is defined as $\boldsymbol{U}^T = \begin{bmatrix}\boldsymbol{F}_I^T & \boldsymbol{M}_B^T\end{bmatrix}$. We can rewrite the system equations of motion as follows:

$$\dot{\boldsymbol{\mathcal{X}}} = f(\boldsymbol{\mathcal{X}}, \boldsymbol{U}) : \begin{cases} \dot{\boldsymbol{V}} = \boldsymbol{M}^{-1}(\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{C}\boldsymbol{V}) \\ \dot{\boldsymbol{x}}_q = \boldsymbol{v}_q \\ \dot{\boldsymbol{r}}_L = \boldsymbol{v}_L \\ \dot{\boldsymbol{\omega}}_B = \boldsymbol{J}_q^{-1}(-\boldsymbol{\omega}_B^{\times}\boldsymbol{J}_q\boldsymbol{\omega}_B + \boldsymbol{M}_B) \\ \dot{\boldsymbol{R}}_{IB} = \boldsymbol{R}_{IB}\boldsymbol{\omega}_B^{\times} \end{cases} \tag{8}$$

There system matrices are defined as follows:

$$\boldsymbol{M} = \begin{bmatrix} m_p\boldsymbol{B}^T\boldsymbol{B} & m_p\boldsymbol{B}^T \\ m_p\boldsymbol{B} & (m_q + m_p)\mathbf{1}_{3\times 3} \end{bmatrix}; \boldsymbol{C} = \begin{bmatrix} m_p\boldsymbol{B}^T\dot{\boldsymbol{B}} & \boldsymbol{0} \\ m_p\dot{\boldsymbol{B}} & \boldsymbol{0} \end{bmatrix}$$
$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{F}_I \end{bmatrix}, \boldsymbol{G} = \begin{bmatrix} m_p\boldsymbol{B}^T\boldsymbol{g}_I \\ (m_q + m_p)\boldsymbol{g}_I \end{bmatrix} \tag{9}$$

where $\boldsymbol{A} = -\boldsymbol{R}_{IB}\boldsymbol{r}_t^{\times}$. The problem statement is now formulated as: given a target quadrotor position $\boldsymbol{x}_t$, design a feedback law $\boldsymbol{F} = \Phi(\boldsymbol{\mathcal{X}}, \boldsymbol{Z}), \dot{\boldsymbol{Z}} = \Theta(\boldsymbol{\mathcal{X}})$, such that as $t \to \infty$, $\boldsymbol{x}_q \to \boldsymbol{x}_t$, $\boldsymbol{v}_q \to \boldsymbol{0}$, $\boldsymbol{v}_L \to \boldsymbol{0}$, and $\boldsymbol{r}_L \to \boldsymbol{0}$. In other words, we wish to design a proper guidance and control law that ensures the UAV-Payload system fly along a prescribed path to reach the target landing location, and make sure when the payload is landed, its vertical touch down speed becomes zero for soft landing.

## III. GUIDANCE AND CONTROL LAW DESIGN

When considering a payload delivery mission, a critical mission segment is the landing procedure. This is the portion of the mission where the payload goods will be delivered to the desired location on the ground. In a practical delivery application, the payload may be fragile, or have contents that should not be subjected to harsh shocks. From this perspective, it is natural to conclude that the payload should be delivered to the ground with a soft touchdown. If the payload is rigidly connected to the UAV, this implies that the UAV should have a soft contact with the ground when landing. If the payload is carried by a tether from the UAV, it is the touchdown of the payload to the ground that must be controlled. In this section, we investigate using a guidance law to achieve an autonomous landing.

### A. The Linearized Dynamic Model

This section presents the linearized model based on Eq.(8). According to the assumption of $\boldsymbol{r}_t \approx \boldsymbol{0}$, the translational channel is decoupled from the attitude dynamics of the quadrotor. In addition, we assume that the attitude controller provided in Sec. III-D responses at a much faster rate than the translational channel containing $\boldsymbol{x}_q$ and $\boldsymbol{r}_L$. Therefore, we can firstly design the controller for the translational channel $f_T$.

$$\dot{\boldsymbol{X}}_T = f_T(\boldsymbol{X}_T, \boldsymbol{U}_T) : \begin{cases} \dot{\boldsymbol{V}} = \boldsymbol{M}^{-1}(\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{CV}) \\ \dot{\boldsymbol{x}}_q = \boldsymbol{v}_q \\ \dot{\boldsymbol{r}}_L = \boldsymbol{v}_L \end{cases} \tag{10}$$

where $\boldsymbol{U}_T = \boldsymbol{F}_I$. We present the following chain rule for determining the Jacobian matrix of a matrix-vector product:

*Lemma 1:* We define $\boldsymbol{A} \in \mathbb{R}^{N\times N}$, $x \in \mathbb{R}^{N\times 1}$, $y \in \mathbb{R}^{M\times 1}$, and $\boldsymbol{J}_y = \frac{\partial \boldsymbol{A}x}{\partial y^T}$. Then we have the following identity:

$$\boldsymbol{J}_y = \begin{bmatrix} x^T \boldsymbol{J}_{A_1,y} \\ \vdots \\ x^T \boldsymbol{J}_{A_N,y} \end{bmatrix} + \begin{bmatrix} a_1 \boldsymbol{J}_{x,y} \\ \vdots \\ a_N \boldsymbol{J}_{x,y} \end{bmatrix} \tag{11}$$

where $a_1, ..., a_N$ are the row vectors of $\boldsymbol{A}$. The matrices $\boldsymbol{J}_{A_j,y}$ and $\boldsymbol{J}_{x,y}$ are defined as:

$$\boldsymbol{J}_{A_j,y} = \begin{bmatrix} \frac{\partial a_{j,1}}{\partial y^T} \\ \vdots \\ \frac{\partial a_{j,N}}{\partial y^T} \end{bmatrix}, \boldsymbol{J}_{x,y} = \begin{bmatrix} \frac{\partial x_1}{\partial y^T} \\ \vdots \\ \frac{\partial x_N}{\partial y^T} \end{bmatrix} \tag{12}$$

Detailed derivation is shown in Appendix A.

The equilibrium point is picked as the origin, i.e. $\boldsymbol{X}_T = \boldsymbol{0}$, $\boldsymbol{F}_{I,e} = -(m_p + m_q)\boldsymbol{g}_I$. The Jacobian of $\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{CV}$ w.r.t $\boldsymbol{X}_T$ and $\boldsymbol{U}_T$ evaluated at the equilibrium are $\boldsymbol{J}_x$ and $\boldsymbol{J}_u$ respectively:

$$\begin{aligned} \boldsymbol{J}_x &= \frac{\partial(\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{CV})}{\partial \boldsymbol{X}_T} = \begin{bmatrix} \boldsymbol{0}_{2\times 5} & -m_p g/l \boldsymbol{1}_{2\times 2} & \boldsymbol{0}_{2\times 3} \\ \boldsymbol{0}_{3\times 5} & \boldsymbol{0}_{3\times 2} & \boldsymbol{0}_{3\times 3} \end{bmatrix} \\ \boldsymbol{J}_u &= \frac{\partial(\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{CV})}{\partial \boldsymbol{U}_T} = \begin{bmatrix} \boldsymbol{0}_{2\times 3} \\ \boldsymbol{1}_{3\times 3} \end{bmatrix} \end{aligned} \tag{13}$$

Note that at the equilibrum, $\boldsymbol{F} + \boldsymbol{G} - \boldsymbol{CV} = \boldsymbol{0}$. According to Lemma 1, the linearized translational subsystem is:

$$\delta \dot{\boldsymbol{X}}_T = \boldsymbol{A}_T \delta \boldsymbol{X}_T + \boldsymbol{B}_T \delta \boldsymbol{U}_T \tag{14}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{M}_1^{-1} \boldsymbol{J}_x \\ \vdots \\ \boldsymbol{M}_N^{-1} \boldsymbol{J}_x \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} \boldsymbol{M}_1^{-1} \boldsymbol{J}_u \\ \vdots \\ \boldsymbol{M}_N^{-1} \boldsymbol{J}_u \end{bmatrix} \tag{15}$$

where $\delta \boldsymbol{X}_T$ and $\delta \boldsymbol{U}_T$ are perturbations of states and inputs from the equilibrium. Hence, the overall system matices are:

$$\boldsymbol{A}_T = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{E}_0 \end{bmatrix}; \boldsymbol{B}_T = \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0}_{5\times 3} \end{bmatrix}; \boldsymbol{E}_0 = \begin{bmatrix} \boldsymbol{1}_{5\times 5} & \boldsymbol{0}_{5\times 5} \end{bmatrix} \tag{16}$$

### B. Position Stabilization Law

We define a so-called virtual control force as $\boldsymbol{F}_{I,v}$ which is the lift that should be generated by the propellers and to replace $\boldsymbol{F}_I$. Based on the second assumption, we have the time-scale separation, meaning that we could firstly design $\boldsymbol{F}_{I,v}$ as a feedback law, and then command the inner attitude control loop in Sec. III-D to tilt the quadrotor according to the direction of $\boldsymbol{F}_{I,v}$ to control the vehicle. The position stabilization law is then design based on a cascade form as a gain $\boldsymbol{K}$ as follows:

$$\begin{aligned} \boldsymbol{K} &= \begin{bmatrix} \boldsymbol{0}_{3\times 2} & \boldsymbol{K}_v & \boldsymbol{0}_{3\times 2} & \boldsymbol{K}_x \end{bmatrix} \\ \boldsymbol{F}_{I,v} &= -\boldsymbol{K}\delta \boldsymbol{X}_T - (m_p + m_q)\boldsymbol{g}_I \\ \boldsymbol{A}_c &= \boldsymbol{A}_T - \boldsymbol{B}_T \boldsymbol{K} \end{aligned} \tag{17}$$

where $\boldsymbol{A}_c$ is the closed-loop system matrix under the feedback gain $\boldsymbol{K}$. The position loop is a PD liked controller that stabilizes both the quadrotor position and payload motion. $\boldsymbol{K}_v$ and $\boldsymbol{K}_x$ are the gain matrices to be picked so that $\boldsymbol{A}_c$ is Hurwitz.

### C. The PPN Inspired Control Law

We take inspiration from a standard pure proportional navigation (PPN) approach, e.g. [9], to create a guidance law for the current mission. We have a static landing position, and we would like the velocity to be zero when the final position is reached. We follow the inspiration from the traditional approach, but instead of using an acceleration command to track a constant velocity, we use a velocity command to "track" a constant position. In a sense, this is lowering the dimension of the problem in time by one. We will also assume that the autopilot is using a position controller that we can use to control the vehicle. The intuition for this comes from the idea of reaching a zero landing velocity at the final target location. This would achieve a landing where it is required that the quadrotor or payload touches down with minimal impact. Considering the altitude axis only, the intuition here is that we want $\dot{z}_q \Rightarrow 0$ as $z_q \Rightarrow z_t$, where $\dot{z}_q, z_q$ are the velocity and height, and $z_t$ is the target height,

$$\dot{z}_{cmd} = K_{p,z}(z_t - \delta z_q) \tag{18}$$

It is further extended for three-dimensional guidance, as follows:

$$\dot{\boldsymbol{x}}_{cmd} = \begin{bmatrix} \dot{x}_{cmd} \\ \dot{y}_{cmd} \\ \dot{z}_{cmd} \end{bmatrix} = \boldsymbol{K}_p(\boldsymbol{x}_t - \delta \boldsymbol{x}_q) \tag{19}$$

In order to determinen the overall stability of the closed-loop system, we define an augmented system state as $\bar{\boldsymbol{X}}^T = \begin{bmatrix} \delta \boldsymbol{X}_T^T & \boldsymbol{x}_{cmd}^T \end{bmatrix}$. By adding the PPN guidance-inspired formula into the control law, the virtual control force should be $\boldsymbol{F}_{I,v} = \boldsymbol{K}_x \boldsymbol{x}_{cmd} - \boldsymbol{K}\delta \boldsymbol{X}_T - (m_p + m_q)\boldsymbol{g}_I$. We set $\boldsymbol{x}_t = \boldsymbol{0}$ to simplified the stability analysis. Then the corresponding augmented closed-loop system is as follows:

$$\dot{\bar{\boldsymbol{X}}} = \boldsymbol{A}_p \bar{\boldsymbol{X}} \tag{20}$$

where the system matrices are defined as:

$$\begin{aligned} \boldsymbol{A}_p &= \begin{bmatrix} \boldsymbol{A}_c & \boldsymbol{BKE}_1 \\ \boldsymbol{E}_2 & \boldsymbol{0}_{3\times 3} \end{bmatrix}; \boldsymbol{E}_1 = \begin{bmatrix} \boldsymbol{0}_{7\times 3} \\ \boldsymbol{1}_{3\times 3} \end{bmatrix} \\ \boldsymbol{E}_2 &= \begin{bmatrix} \boldsymbol{0}_{3\times 7} & -\boldsymbol{K}_p \end{bmatrix} \end{aligned} \tag{21}$$

For the above equation, we can see that $\boldsymbol{K}_p$ need to be picked so that $\boldsymbol{A}_p$ is Hurwitz. To this point, we have considered the guidance law as a means for landing the quadrotor autonomously on the ground. However, in reality we shall also consider the application of landing a slung payload on the ground, at a given target location. If a measurement of the payload position was available, it would be possible to estimate the state of the payload. The problem could be reformulated to attempt to directly drive the payload position to the target location. However, if the payload state information is not directly available (which could be the case in some practical implementations), we can investigate how we could directly apply the proposed guidance law. In this case, we must still assume that we have position information for the quadrotor, which is reasonable as most quadrotors have a GPS and barometer. The idea in this case is to drive the quadrotor to a virtual target location, as shown in Figure 2.

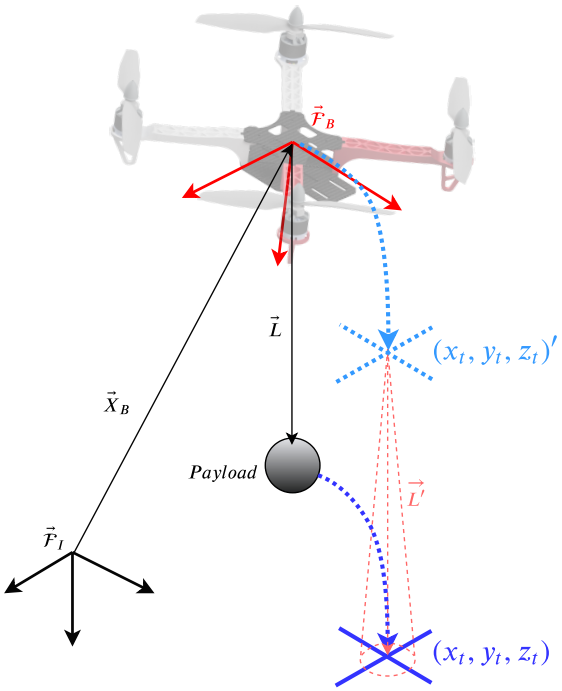



Fig. 2: Definition for modified PPN Law with a slung payload.

In this figure, we use the annotation ($'$) to denote the virtual target for the UAV. What we mean by a virtual target is a desired location for the quadrotor where the payload will also be at it's desired location. However, we must also allow the payload to swing at the final location, which is denoted by $\vec{L}'$ as opposed to the current tether $\vec{L}$. If we want the payload to reach the final location, we command the virtual target for the UAV to be directly above the desired payload target, shown in Eq. (22). Here, $L$ represents the scalar value of the tether length. This is under defined in the inertial reference frame, and assumes that the payload has no swinging motion.

$$x_t' = x_t \tag{22a}$$
$$y_t' = y_t \tag{22b}$$
$$z_t' = z_t + L \tag{22c}$$

If the payload has zero or sufficiently small swing angle, then it is trivial to see that this behaves the same as the quadrotor landing case. However, we must consider what effect a non-zero swing angle would have on this payload landing. Using simple trigonometric functions, we can obtain relations for the offset in the z and x-y positions caused by a given swing angle. Although there are two swing angles (along the x and y axes), we use $\theta$ to represent the magnitude of the swing angle in relation to the z-axis (i.e. with components along the x and y axes). Here, $\Delta z = L - L\cos(\theta)$ and $\Delta xy = L\sin(\theta)$. $\Delta z$ represents the error from the desired z position, $\Delta xy$ represents the error in the x-y plane. The results are tabulated as a percentage of the total length, $L$, and shown in Table I. From these result, we can see that the x-y position accuracy is affected the most by a swing angle. For small swing angles, the z position accuracy is high. For example, if the tether length is 1 m, the error in the z position is less than 5 cm, which is also within the error of the position controller under lab conditions. It is possible to conclude that for small swing angles, the landing of the payload should be relatively unaffected, with small inaccuracy in the xy landing position. Therefore, during the landing phase, it is desirable to maintain a small swing angle, to preserve landing accuracy. We will show later in the experimental investigation that the modified PPN maintains a swing angle less than 4 degrees during the landing. This is in contrast to the unguided controller, which induces a swing angle around 14 degrees during the landing phase.

TABLE I: Swing Angle Effect on Payload Accuracy

| $\theta$ [$^o$] | $\Delta z$ [$\%$ $of$ $L$] | $\Delta xy$ [$\%$ $of$ $L$] |
|---|---|---|
| 5 | 0.381 | 8.72 |
| 10 | 1.52 | 17.3 |
| 15 | 3.41 | 25.8 |

In summary, the proposed guidance law required a solid foundation for good tracking control for payload transportation with small swing angles. It has been achieved by path following control study of our research work, e.g. [4]. The overall guidance and control framework is presented by a block diagram, shown in Fig. 3.

### D. Attitude Controller

This section gives the target torque and lift the quadrotor should generate based on the virtural control force $\boldsymbol{F}_{I,v}$. For this type of payload manipulation control, we directly adopt an AGAS attitude tracker as suggested in [4]. The total lift from the propellers is $f = ||\boldsymbol{F}_{I,v}||$. We pick a command yaw angle $\psi$ for each quadrotor. We assume lift is along the -z axis of the quadrotor, i.e. $n_z = -\boldsymbol{F}_{I,v}/f$. The reference attitude

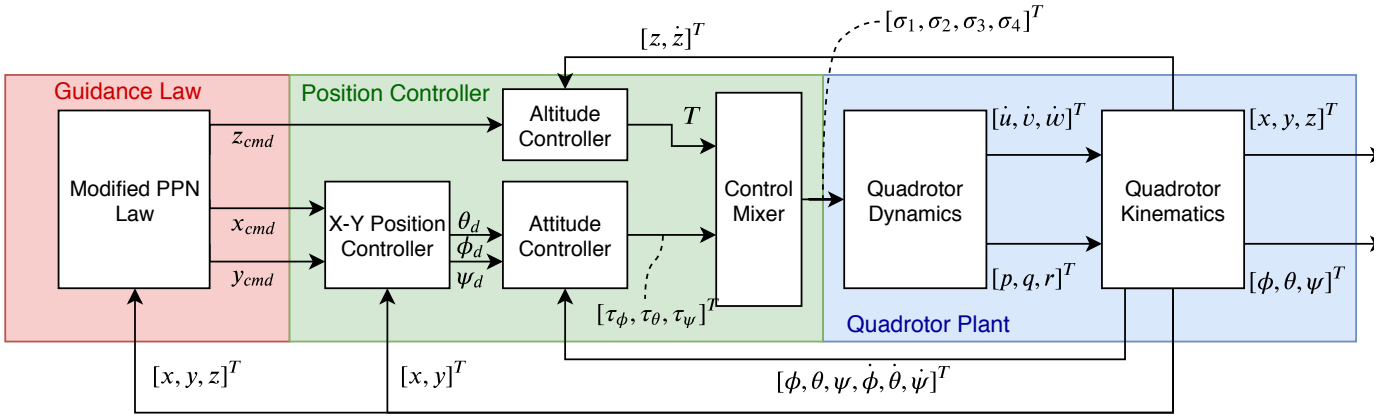



Fig. 3: Block diagram for modified PPN law implementation on a quadrotor.

trajectory of each quadrotor based on $\boldsymbol{F}_{I,v}$ is $\boldsymbol{R}_{IB,d}$ obtained in the following way:

$$\begin{aligned}&\tilde{n}_x = \begin{bmatrix}\cos\psi & \sin\psi & -(\cos\psi n_{z,x} + \sin\psi n_{z,y})/n_{z,z}\end{bmatrix}^T;\\&n_x = \tilde{n}_x/||\tilde{n}_x||; n_y = n_z^\times n_x/||n_z^\times n_x||;\\&\boldsymbol{R}_{IB,d} = \begin{bmatrix}n_x & n_y & n_z\end{bmatrix}\end{aligned} \tag{23}$$

where $n_{z,x}$ and $n_{z,y}$ are the x and y component of $n_z$ respectively. $\boldsymbol{F}_{I,v}$ only provides two degrees of freedom, i.e. $n_z$, so $\psi$ is an additional constraint to determine $\boldsymbol{R}_{IB,d}$. We define $\boldsymbol{\omega}_d$ as the desired angular velocity. Once $\boldsymbol{R}_{IB,d}$, $\boldsymbol{\omega}_d$, and $\dot{\boldsymbol{\omega}}_d$ are calculated based on $\boldsymbol{F}_{I,v}$, the control torque $\boldsymbol{M}_B$ is as follows:

$$\begin{aligned}\boldsymbol{M}_B = &-b_\omega\tilde{\boldsymbol{\omega}} - b_r\boldsymbol{e}_r - \tilde{\boldsymbol{\omega}}^\times\boldsymbol{J}_q\tilde{\boldsymbol{\omega}} + \boldsymbol{\omega}_B^\times\boldsymbol{J}_q\boldsymbol{\omega}_B\\&- \boldsymbol{J}_q(\tilde{\boldsymbol{\omega}}^\times\tilde{\boldsymbol{R}}^T\boldsymbol{\omega}_d - \tilde{\boldsymbol{R}}^T\dot{\boldsymbol{\omega}}_d)\end{aligned} \tag{24}$$

where $\boldsymbol{e}_r = \sum_{i=1}^3 e_i^\times\tilde{\boldsymbol{R}}e_i$, $\tilde{\boldsymbol{R}} = \boldsymbol{R}_{IB,d}^T\boldsymbol{R}_{IB}$, $\boldsymbol{\omega}_d = (\boldsymbol{R}_{IB,d}^T\dot{\boldsymbol{R}}_{IB,d})^\vee$, and $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}_B - \boldsymbol{R}_{IB}^T\boldsymbol{\omega}_d$. The propeller rotation speed can be obtained by a motor mixing algorithm in the on-board FCU.

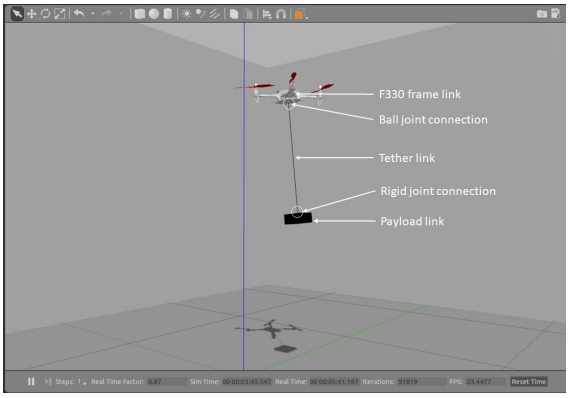



Fig. 4: Simulation in Gazebo environment running PX4 SITL and ROS.

## IV. GAZEBO SIMULATIONS

In order to evaluate design to support experimental implementation, where a PX4-based autopilot and ROS (Robot Operating System) would be in the development framework for the UAV, Gazebo was selected due to the ease of transition from simulation to experiment. Gazebo is the natural simulation tool to use due to the compatibility and integration. The PX4 firmware that serves as the UAV's autopilot and flight controller can be run in software-in-the-loop (SITL) mode and connected to the Gazebo physics. This means that all of the code and algorithms executed on the hardware can be run within the simulation. This allows for easy code debugging and testing in the simulation environment prior to deployment on the experimental UAV. For this work, a custom model for the Gazebo simulator was developed based on our experimental quadrotor (F330). This model was based on a pre-built and available model (3DR Iris), however it was modified to match our experimental platform (from CAD) and other measurements (mass, inertia), seen in Figure 4. The model was also extended to include our laboratory room as the flight zone for testing, and a slung tether payload.

The principal moments of inertia around the x, y and z axes were then calculated using the parallel axis theorem. The results from this component build-up method is $\boldsymbol{J}_q = diag(\begin{bmatrix}0.0115 & 0.0116 & 0.0128\end{bmatrix})[kg \cdot m^2])$. The total mass of the experimental F330 quadrotor with all components was 1.00 kg. These values were entered into the Gazebo SDF file for the F330.

### A. Parameter Selection and Close Loop Eigenvalues

After plugging in the system parameters, we have the following matrix for the linearized system:

$$\boldsymbol{A} = \begin{bmatrix}\boldsymbol{0}_{5\times1} & -14.7150 & 0 & 0 & 0 & 0\\ \boldsymbol{0}_{5\times1} & 0 & -14.7150 & 0 & 0 & 0\\ \boldsymbol{0}_{5\times1} & 4.9050 & 0 & 0 & 0 & 0\\ \boldsymbol{0}_{5\times1} & 0 & 4.9050 & 0 & 0 & 0\\ \boldsymbol{0}_{5\times1} & 0 & 0 & 0 & 0 & 0\end{bmatrix}$$

$$\boldsymbol{B} = \begin{bmatrix}-1.0000 & 0 & 0\\ 0 & -1.0000 & 0\\ 1.0000 & 0 & 0\\ 0 & 1.0000 & 0\\ 0 & 0 & 0.6667\end{bmatrix} \tag{25}$$

For the control and PPN gains, we pick them as:

$$\boldsymbol{K} = \begin{bmatrix} \mathbf{0}_{3\times2} & 0.5 \times \mathbf{1} & \mathbf{0}_{3\times2} & 0.4 \times \mathbf{1} \end{bmatrix} \tag{26}$$

$\boldsymbol{K}_p = k \times \mathbf{1}_{3\times3} = 0.3 \times \mathbf{1}_{3\times3}$ is picked for the following simulation and experiments. The eigenvalues of $\boldsymbol{A}_p$ are listed in Table II. From the table, we can see that all eigenvalues are on LHS of the imaginary plane. The $\boldsymbol{A}_p$ is Hurwitz so that the overall closed-loop system is asymptotically stable. The virtual control force $\boldsymbol{F}_{I,v}$ is converted into a target quaternion and thrust command to the drone. For the simulation and experiment, we directly rely on the default PX4 firmware for attitude stabilization in OFFBOARD mode.

TABLE II: Eigenvalues of $\boldsymbol{A}_p$

| | |
|---|---|
| $\lambda_{1,2}$ | $-0.0844 \pm 3.8455i$ |
| $\lambda_{3,4}$ | $-0.0111 \pm 0.5073i$ |
| $\lambda_5$ | $-0.3091$ |
| $\lambda_{6,7}$ | $-0.0844 \pm 3.8455i$ |
| $\lambda_{8,9}$ | $-0.0111 \pm 0.5073i$ |
| $\lambda_{10}$ | $-0.3090$ |
| $\lambda_{11,12}$ | $-0.0123 \pm 0.5089i$ |
| $\lambda_{13}$ | $-0.3088$ |

### B. Case 1: Vertical Descent

For the first case, we perform a simple vertical landing maneuver. This represents a case where the quadrotor is hovering directly over the desired landing target, and allows us to evaluate the touchdown. For the sample mission, we begin hovering at a height of 4 m above the target. It should also be noted that although the x and y PPN guidance is active, they will not be discussed here as they only keep the quadrotor at the current x and y position during the landing.

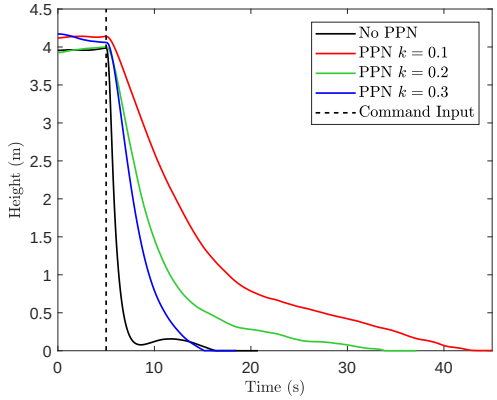



Fig. 5: Height of the quadrotor during the landing command.

Figure 5 shows the z-position as a function of time during the landing maneuver. We also test a PID setpoint controller as our baseline (the "No PPN" case). In each case, the landing is initiated by switching to the desired target position located at 0 m (i.e. a step-input). It should be noted that the PID controller is also used as the "inner-loop" position controller for the PPN guidance law - without modification. From this figure, we see that each law will eventually drive the quadrotor to land. The step input for the unguided controller results in a profile where the quadrotor drops from the sky and crash-lands (visually observed in simulation). However, it does recover and shows a small oscillation before eventually landing. Although the controller gains might be changed to improve this landing condition, those gains may not be suitable for general position holding functionality. On the other hand, we see that for the PPN gains tested, the landing maneuver is more docile and smooth.

Figure 6 shows the vertical velocity during the landing. Here we notice the potential concern with directly using a step-input for the unguided position controller when landing the vehicle. For the landing maneuver, we would like the quadrotor to move smoothly, which is not observed due to the sharp velocity profile (which corresponds to a high acceleration). Using the PPN law, we find that the velocity profile is more controlled, and the maximum velocity is kept at less than one-third that of the unguided controller.

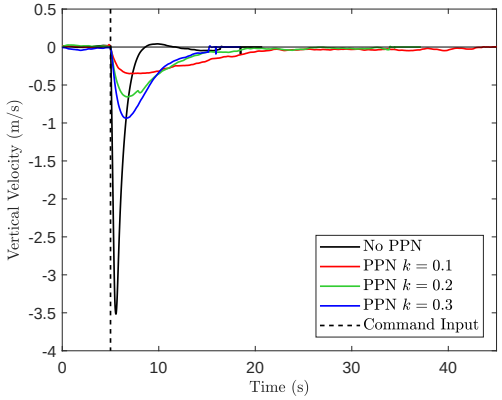



Fig. 6: Z-velocity of the quadrotor during the landing command.

### C. Case 2: Translational Descent

For the second case, we will consider a translational descent landing maneuver for the quadrotor. This would represent a case where the landing maneuver is initiated once the UAV is within a certain vicinity of the landing target. This allows us to evaluate the x-y performance of the PPN law. The initial position of the quadrotor is set at $(x, y, z) = (4, 4, 4)\ m$, and the target location is placed at $(x, y, z) = (0, 0, 0)\ m$.

For this case, we show the x, y, and z, positions of the quadrotor as a function of time in Figure 7. The results from this figure demonstrate similar behaviour to that observed during Case 1. Specifically, the unguided controller is fast to react to the step-input, but introduces some oscillation and overshoot in the x and y axes. It should be noted that for the x and y axes we also notice some overshoot in the PPN law, primarily at the largest gain of $k = 0.3$. In each case, there is some relatively small errors in the final x-y position of the quadrotor, however, all arrive within close proximity to the desired target.

The velocities of the vehicle are shown in Figure 8. Once again, we observe the large spike in the unguided controller velocity, indicating a very aggressive maneuver with high

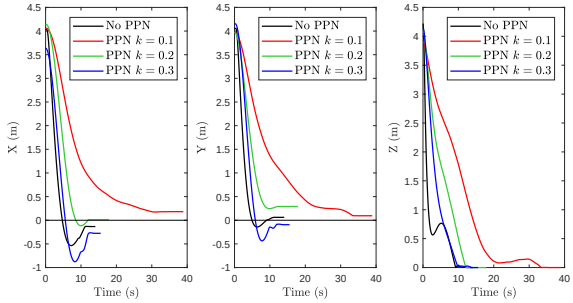



Fig. 7: X,Y,Z positions of the quadrotor during landing command.

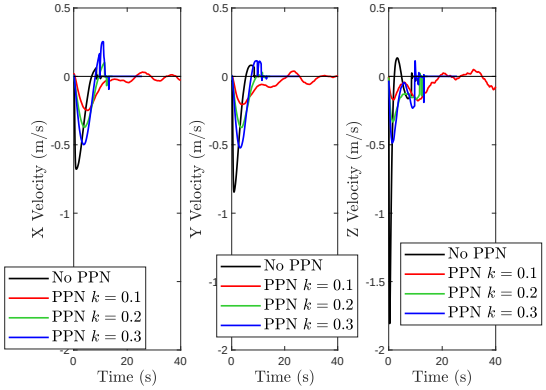



Fig. 8: X,Y,Z-velocities of the quadrotor during the landing command.

acceleration. This is not our desired intent for the landing maneuver, as we would like a smooth trajectory. We notice that the PPN velocities are less than one-third of the unguided controller velocities. This is again achieved with a near identical landing time as the PID control, with the PPN at $k = 0.3$. The noisy velocity behaviour at the end of the trials is explained by the vehicle coming into contact with the ground. The motors are not disarmed immediately upon contact with the ground, and therefore due to the small offsets in the x-y landing positions from the desired target, the PID and PPN continue to try to reach the final desired x-y position. This results in the vehicle skimming in contact with the ground, creating a noisy velocity profile around zero. To prevent this, once the quadrotor has touched down, it should be disarmed immediately (either manually or automatically through detection of this state).

## V. EXPERIMENTAL INVESTIGATION

The assembled quadrotor with labeled components can be seen in Figure 9. For the autopilot system on this quadrotor, we use a Pixhawk running the PX4 flight stack firmware. The Odroid XU-4 is used as an on-board computer, which runs Ubuntu 18.04 and ROS. The XU-4 is a single board computer, equipped with Samsung Exynos5422 Cortex™-A15 2Ghz and Cortex™-A7 Octa core CPUs. This computer is used to communicate with the ground station and Pixhawk, while providing a computational platform to implement high-level controls and algorithms. We use this onboard computer to run guidance law and position controller for this landing mission.

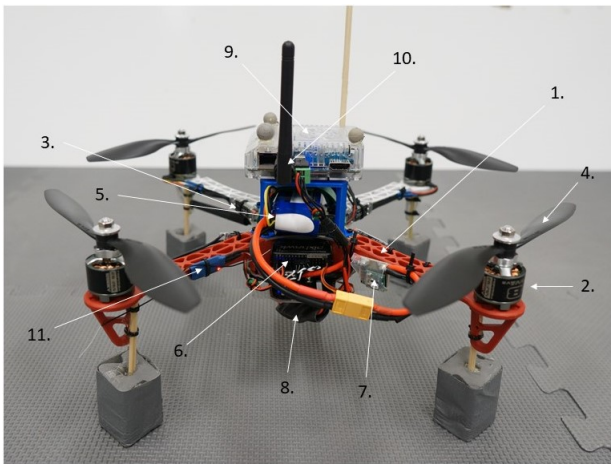



Fig. 9: The F330 Quadcopter platform. 1. - F330 frame, 2. - NTM PropDrive 1200kV motors, 3. - LittleBee 20A ESCs, 4. - 8x4.5 Propellers, 5. Turnigy 3S 2200 25C LiPo battery, 6. - Pixhawk autopilot, 7. - Pixhawk power module, 8. - 5V/5A external BEC, 9. - Odroid XU4, 10. - WiFi adapter, 11. - FrSky XM Plus receiver.

The experimental investigation for this work was completed in a laboratory environment, shown in Figure 10. 14 Flex13 OptiTrack cameras [1] are used to cover the room, giving us a flyable volume of approximately 5.5 m $\times$ 3.5 m $\times$ 2.5 m. The Flex13 cameras then connect to a computer via USB, which is running OptiTrack's Motive software. This software can then be used to stream the pose information of the trackable objects over a network interface using the `VRPN` interface. In our particular case presented here, we connect the computer to an ASUS AC5300 WiFi router via an Ethernet cable, for reliable connection. The OptiTrack data can then be wireless streamed from the Odroid using a 5 GHz WiFi adapter plugged into a USB port on the Odroid. A 2.4 GHz connection can also be used, however, since the transmission distance is short a 5 GHz connection is preferable.

### A. *Case 3: Slung Payload - Vertical Landing*

Now, we will move on to the slung payload delivery mission. In this case and the following case, we use a mass of 500g for the payload, and a tether length of 1 meter. The payload mass is half the mass of the experimental quadrotor, therefore this is considered a heavy payload for the UAV. We use a heavy payload mass, as it is more likely to cause the system to be unstable. Any payload swing will have a greater effect on the quadrotor motion, due to the larger reaction forces generated through the tether. We again begin with a case where we perform a vertical landing mission. This test will evaluate the vertical landing performance of the PPN guidance law. It is important to note that for the slung payload cases, we only evaluate the PPN with a gain of $k = 0.3$, which was observed

[1] https://v22.wiki.optitrack.com/index.php?title=Calibration#Calibration_Summary

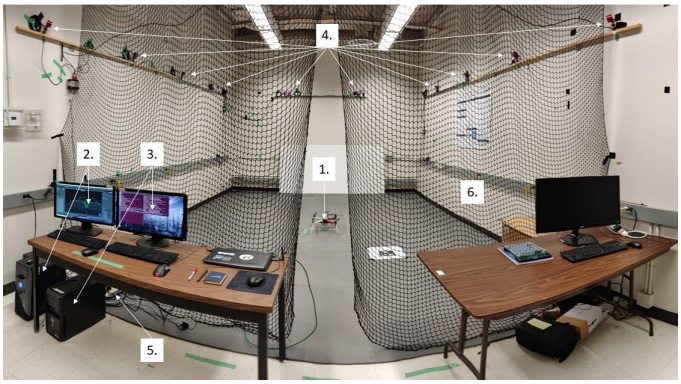



Fig. 10: Lab environment and setup for experiments. 1. - Experimental quadcopter, 2. - Computer running Motive software, 3. - Ground station computer, 4. - OptiTrack Flex13 cameras, 5. (hidden) - WiFi router, 6. - Safety netting.

to provide the best performance from the results in Case 1 and Case 2.

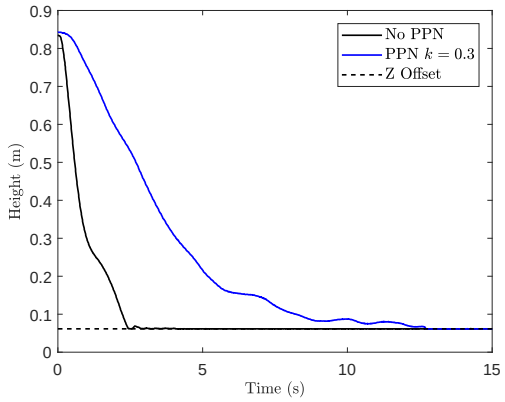



Fig. 11: Height of the payload during the landing command.

The time history of the payload height during the landing procedure is shown in Figure 11. Here we observe that the unguided controller reaches the final target faster than the PPN guided law. The steep slope for the unguided landing indicates that the payload touched down with a higher velocity when compared to the PPN landing. This resulted in a small rebound in height resulting from the ground contact. The magnitude of this rebound is small, however, this could be due to the low initial altitude of the payload. If the payload could be flown higher, we might expect to see a more significant rebound upon touchdown. The PPN landing takes longer to finally touchdown, however, the slope approaching this target is low, indicating a softer touchdown. The experiment result with PPN law is also shown in a video. [2]

### *B. Case 4: Slung Payload - Translational Landing*

In the final case, we perform a mission where we deliver a slung payload by performing a translational landing. This is a combination of Cases 2 and 3 to evaluate the PPN modified guidance law for a more realistic payload delivery mission. We will again fly the quadrotor at an initial position of $(x, y, z) = (1, 1, 2)m$ . This results in a payload initial location of $(x, y, z) = (1, 1, 0.85)m$. The quadrotor is given a virtual landing target as the goal location for delivering the payload. The virtual target location for the quadrotor is given as $(x, y, z) = (-1, -1, 1)$ m. The position of the payload during the landing is shown in Figure 12. The results show that the PPN achieved equal or better final accuracy in the x and y axes. In addition, the PPN law slowed the payload down in the z axis and achieved a soft touchdown. However, the unguided controller essentially dropped the payload on the ground and did not achieve a soft touchdown. During the unguided controller test, touchdown occurred before the payload had reached the final goal location, which is shown more clearly in Figure 13.

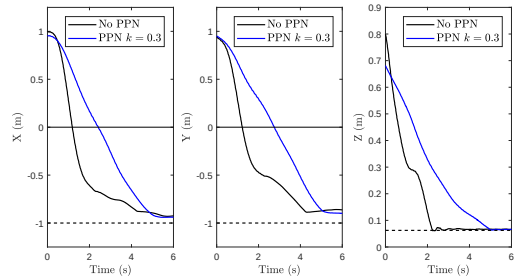



Fig. 12: X,Y,Z positions of the payload during the landing command.

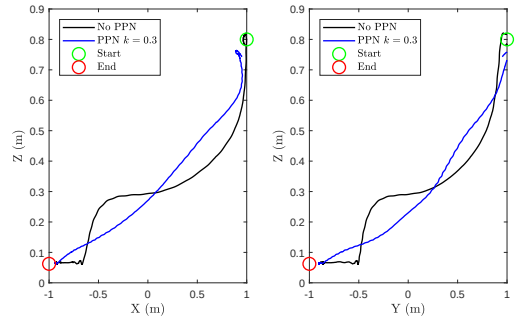



Fig. 13: XZ and YZ position plots of the payload.

In Figure 13, we can observe the path of the payload in the x-z and y-z planes respectively. The PPN guided landing follows a more direct path to the goal, arriving at the target along a nearly straight line. It is in these plots however, that we clearly observe an issue with the unguided controller for this mission. As mentioned, the payload not only has a steep descent directly into the ground (resulting in a hard landing), it also touches down before the target location has been reached in the x and y directions. This results in the quadrotor dragging the payload along the ground until it reaches the desired x and y target. This is clearly not what could be classified as a desirable autonomous landing, with a smooth and precise touchdown.

Figure 14 shows the x and y position of the payload throughout the flight. In this case, we see that both the

[2]https://youtu.be/lnX3RhOztgg

unguided and PPN guided landings followed a reasonably straight line path to the target location along the x and y axes.

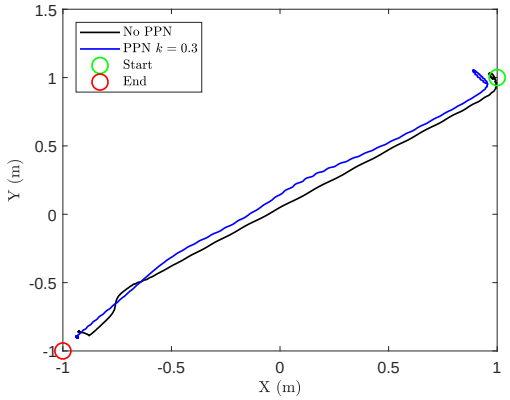



Fig. 14: XY position plot of the quadrotor.

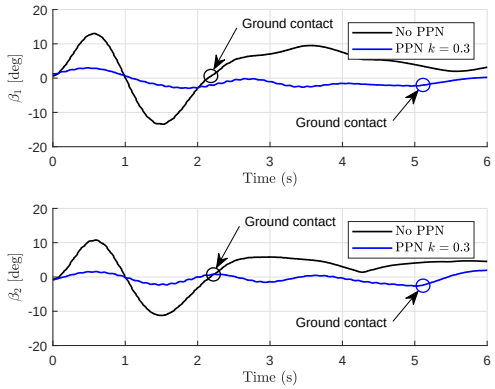



Fig. 15: Swing angles of the payload during the landing.

Finally, we plot the swing angles of the payload during this landing segment, shown in Figure 15. We use two angles $\beta_1$ and $\beta_2$ to describe the two axes of swing motion for the payload. In this figure, we observe the concern with the unguided landing as it results in a large swing angle relative to the PPN guided landing. This results from the large initial acceleration of the quadrotor, which induces the initial swing angle. The swing angle in the unguided case reaches close to 14 degrees, whereas the PPN guided landing keeps the swing angle below 4 degrees. By keeping the swing angle low, the dynamics of the UAV are less disturbed by the payload motion. This also helps in maintaining an accurate landing. We note the points at which the payload came into contact with the ground, as this stopped the free-swinging motion of the payload. The experiment result with PPN law is also shown in a video. [3]

## VI. CONCLUSION

In this paper, we explored the emerging application of delivering payloads using a UAV. There are various methods in which this type of delivery can be performed, however, in this paper we investigated the concept of slung payload delivery. This is where the payload is attached like a pendulum to the UAV through the use of a cable, or 'tether'. The proposed guidance and control framework focused on the landing portion of the mission. We described the autonomous landing, where the aim is to deliver the payload (or UAV) to the ground with close to zero velocity at touchdown. The pure proportional inspired guidance law, together with custom path following control, enable the soft landing. The comparative study through extensive simulations and representative experimental investigation verified the effectiveness of the prosed approach.

[3]https://youtu.be/fGq2gfyumcU

## ACKNOWLEDGEMENT

The research work presented in this paper is sponsored by Natural Science and Engineering Research Council of Canada (NSERC), under its Collaborative Research Program (CRDPJ 508381 - 16), in collaboration with Drone Delivery Canada (www.dronedeliverycanada.com). The authors would like express the appreciation to the following DDC technical and management staff for their continuous support: Paul Di Benedetto, Greg Colacitti, Erik Chau, Rikky Duivenvoorden, and Yih Tang Yeo.

## APPENDIX A
## A SKETCH OF THE PROOF OF LEMMA 1

The Jacobian $\boldsymbol{J}_y$ of the product $Ax$ is:

$$\boldsymbol{J}_y = \begin{bmatrix} \frac{\partial a_1 x}{\partial y^T} \\ \vdots \\ \frac{\partial a_N x}{\partial y^T} \end{bmatrix} \tag{27}$$

where $a_j$, $j = 1, 2, .., N$ are the row vectors of $\boldsymbol{A}$. We define $J_{y,j}$ as the $j^{th}$ row vector of $\boldsymbol{J}_y$. Hence we have:

$$J_{y,j} = \frac{\partial a_j x}{\partial y^T} = \sum_{i=1}^{N} \frac{\partial a_{j,i} x_i}{\partial y^T} = \sum_{i=1}^{N} x_i \frac{\partial a_{j,i}}{\partial y^T} + a_{j,i} \frac{\partial x_i}{\partial y^T} \tag{28}$$

rearranging the above equation, we have:

$$J_{y,j} = \begin{bmatrix} x_1, \ldots, x_N \end{bmatrix} \begin{bmatrix} \frac{\partial a_{j,1}}{\partial y^T} \\ \vdots \\ \frac{\partial a_{j,N}}{\partial y^T} \end{bmatrix} + \begin{bmatrix} a_{j,1}, \ldots, a_{j,N} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1}{\partial y^T} \\ \vdots \\ \frac{\partial x_N}{\partial y^T} \end{bmatrix} \tag{29}$$

Then the conclusion is verified if we write $\boldsymbol{J}_y$ in matrix format.

## REFERENCES

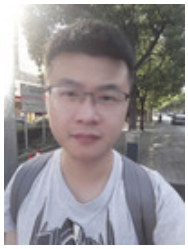
[1] Ivana Palunko, Rafael Fierro, and Patricio Cruz. Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: A dynamic programming approach. In *2012 IEEE International Conference on Robotics and Automation*, pages 2691–2697. IEEE, 2012.

[2] Yi Kui, Gu Feng, Yang Liying, He Yuqing, and Han Jianda. Sliding mode control for a quadrotor slung load system. In *2017 36th Chinese Control Conference (CCC)*, pages 3697–3703. IEEE, 2017.

[3] Marco M Nicotra, Emanuele Garone, Roberto Naldi, and Lorenzo Marconi. Nested saturation control of an UAV carrying a suspended load. In *2014 American Control Conference*, pages 3585–3590. IEEE, 2014.

[4] Longhao Qian and Hugh HT Liu. Path following control of a quadrotor UAV with a cable suspended payload under wind disturbances. *IEEE Transactions on Industrial Electronics*, 2019.

[5] ME Guerrero, DA Mercado, R Lozano, and CD García. Ida-pbc methodology for a quadrotor UAV transporting a cable-suspended payload. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 470–476. IEEE, 2015.
[6] Maria Eusebia Guerrero, DA Mercado, Rogelio Lozano, and CD García. Passivity based control for a quadrotor UAV transporting a cable-suspended payload with minimum swing. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 6718–6723. IEEE, 2015.
[7] M Eusebia Guerrero-Sánchez, D Alberto Mercado-Ravell, Rogelio Lozano, and C Daniel García-Beltrán. Swing-attenuation for a quadrotor transporting a cable-suspended payload. *ISA transactions*, 68:433–449, 2017.
[8] Zhongjiao Shi and Liangyu Zhao. On the autopilot design for a quadrotor during landing phase. In *2016 35th Chinese Control Conference (CCC)*, pages 10875–10879. IEEE, 2016.
[9] Alvika Gautam, PB Sujit, and Srikanth Saripalli. Application of guidance laws to quadrotor landing. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 372–379. IEEE, 2015.
[10] Cai Luo, Weikang Zhao, Zhenpeng Du, and Leijian Yu. A neural network based landing method for an unmanned aerial vehicle with soft landing gears. *Applied Sciences*, 9(15):2976, 2019.
[11] Mohammad Fattahi Sani, Maryam Shoaran, and Ghader Karimian. Automatic landing of a low-cost quadrotor using monocular vision and kalman filter in gps-denied environments. *Turkish Journal of Electrical Engineering & Computer Sciences*, 27(3):1821–1838, 2019.
[12] Farhad A Goodarzi. Autonomous aerial payload delivery with quadrotor using varying length cable. In *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*, pages 394–399. IEEE, 2016.
[13] William Graham Patrick, James Ryan Burgess, and Andrew Conrad. Mechanisms for lowering a payload to the ground from a uav, October 10 2017. US Patent 9,783,297.
[14] Longhao Qian and Hugh HT Liu. Dynamics and control of a quadrotor with a cable suspended payload. In *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 1–4. IEEE, 2017.



**Longhao Qian** Longhao Qian is a PhD candidate at the University of Toronto Institute for Aerospace Studies. He received his Bachelor's degree from Shanghai Jiao Tong University and started his PhD study at University of Toronto since 2015. His research interests include cooperative drone stability and control.

**Silas Graham** Silas Graham received his Master of Applied Science degree at the University of Toronto Institute for Aerospace Studies in 2019. His research topic was on drone delivery project.

**Hugh H.T. Liu** Hugh H.T. Liu is a Professor of the University of Toronto Institute for Aerospace Studies. He received his Bachelor's degree from Shanghai Jiao Tong University (1991), Master's degree from Beijing University of Aeronautics and Astronautics (1994), and Ph.D. from the University of Toronto (1998). His research interests include autonomous unmanned systems, cooperative and formation control, fault tolerant control, active control on advanced aircraft systems, as well as integrated modeling and simulations. Dr. Liu has served many years as a member of the AIAA Guidance, Navigation, and Control Technical Committee. He currently serves as the Associate Editor of AIAA Journal of Guidance, Control and Dynamics. He is also an Associate Editor of the Canadian Aeronautics and Space Journal, an Associate Fellow of AIAA, a Fellow of Canadian Society of Mechanical Engineers, and a registered Professional Engineer in Ontario, Canada.