Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Bank-to-Turn Control for a Small UAV using Backstepping and Parameter Adaptation

Dongwon Jung and Panagiotis Tsiotras

Georgia Institute of Technology, Atlanta, GA 30332 USA
(e-mail: dongwon.jung, tsiotras@gatech.edu)

Abstract: In this research we consider the problem of path following control for a small fixed-wing unmanned aerial vehicle (UAV). Assuming the UAV is equipped with an autopilot for low level control, we adopt a kinematic error model with respect to the moving Serret-Frenet frame attached to a path for tracking controller design. A kinematic path following control law that commands heading rate is presented. Backstepping is applied to derive the roll angle command by taking into account the approximate closed-loop roll dynamics. A parameter adaptation technique is employed to account for the inaccurate time constant of the closed-loop roll dynamics during actual implementation. The path following control algorithm is validated in real-time through a high-fidelity hardware-in-the-loop simulation (HILS) environment showing the applicability of the algorithm on a real system.

1. INTRODUCTION

For decades unmanned aerial vehicles (UAVs) have been proposed as replacements of human pilots during several missions. In recent years small UAVs, in particular, are increasingly receiving attention in universities, national research institutes, etc., because of their envisioned missions in both the military and civilian sectors.

In order to accomplish these missions with minimal human intervention, the operation of a UAV should be fully automated from the top level of path planning, down to the inner control loop level. At the top level of the control hierarchy, the path planning algorithm computes a rough approximation of the optimal path toward the goal. A path generation algorithm then smooths the path yielding a dynamically feasible, flyable path, by taking into account the kinematic constraints of the UAV. Finally, the path following algorithm is responsible for guiding the UAV to stay close to the designed path.

Various control approaches in the literature have been proposed to address the path following problem: Niculescu (2001) introduced a lateral track control law, Park et al. (2004) proposed a simple, yet effective, nonlinear control logic and demonstrated it experimentally, Ren and Beard (2004) considered the problem of constrained trajectory tracking, Nelson et al. (2006) proposed a path following control using vector fields to guide the UAV on the desired path, and Rysdyk (2003) proposed a guidance law based on 'good helmsman' behavior.

Motivated by the method proposed by Micaelli and Samson (1993), kinematic control laws have been used to regulate the distance error from the reference path for unicycle-type mobile robots [Lapierre and Soetanto (2007); Lapierre et al. (2007)]. The key aspect of the proposed algorithms is that the control laws explicitly incorporate the controlled motion of a 'virtual target' along the path to be tracked. In this article, we present a nonlinear path following algorithm, which is an extension of the one by Lapierre and Soetanto (2007) for UAV path following control. We apply a standard backstepping technique to compute the roll angle command from the heading rate command of the kinematic control law. A parameter adaptation technique is then applied to compensate for the inaccurate time constant of the roll closed loop, yielding robust performance during controller implementation. Besides rigorous proofs for convergence of errors, the proposed path following algorithm has an advantage of following a parametric reference path with less stringent initial conditions, as compared to the path following algorithms in the literature. The path following control algorithm is validated through a high-fidelity hardware-in-the-loop simulation (HILS) environment to show the applicability of the presented algorithm on the actual system.

2. PROBLEM DESCRIPTION

A fixed-wing UAV is equipped with a low-level autopilot with on-board sensors that provides feedback control for attitude angles, air speed, and altitude. In a typical search mission, the air speed and the altitude are kept constant, so that the UAV stays inside the safe flying envelope. Suppose that the inertial speed (V) and the course angle (χ; inertial speed heading) are directly measured using an on-board GPS sensor. A simplified kinematic model in the two dimensional plane is utilized as follows,

$$\begin{aligned} \dot{x} &= V\cos\chi, \\ \dot{y} &= V\sin\chi, \\ \dot{\chi} &= \omega, \end{aligned} \tag{1}$$

where (x, y) denotes the inertial position, and ω is the heading rate of the UAV. By expressing the equations of motion in terms of the ground speed and the course angle, the equations of motion become independent of the wind velocity. Furthermore, by using inertial measurements for path planning control, wind disturbance is naturally rejected so that the performance of path following can be improved significantly. For a fixed-wing UAV at a banked-turn maneuver with no sideslip, and assuming that $|\phi| < \phi_{\max} < \pi/2$, the heading rate ω is induced by the roll angle ϕ with the inertial speed V as follows,

$$\omega = \frac{g}{V}\tan\phi, \tag{2}$$

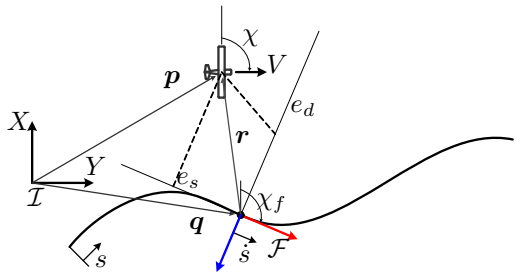


Fig. 1. Definition of the Serret-Frenet frame for the path following problem.

where the roll angle is assumed to be controlled by an inner PID loop. Properly tuned PID gains result in closed-loop roll dynamics which resemble a first-order system,

$$\dot{\phi} = \frac{1}{\lambda_\phi}(\phi_c - \phi), \tag{3}$$

where ϕ_c is the roll angle command and $\lambda_\phi > 0$ is the time constant.

Consider now a UAV flying along a planar path as shown in Fig. 1. Denote the inertial position of the UAV by $\boldsymbol{p} = [x\ y]^\mathsf{T} \in \mathbb{R}^2$. The geometric path is defined in terms of the arc-length parameter s. For any given s, the inertial position of the point associated with s is denoted by $\boldsymbol{q}(s) \in \mathbb{R}^2$, at which the Serret-Frenet frame is attached, and moves along the path with speed $\dot{s}$. The x-axis of the Serret-Frenet frame is aligned with the tangent vector to the path at $\boldsymbol{q}(s)$ and has an angle $\chi_f(s)$ with respect to the inertial frame $\mathcal{I}$. Let the error vector $\boldsymbol{e}$ in the Serret-Frenet frame be decomposed in the along-track error e_s and the cross-track error e_d. Then the inertial error vector $\boldsymbol{r} = \boldsymbol{p} - \boldsymbol{q}(s)$ expressed in the Serret-Frenet frame is obtained by

$$\boldsymbol{e} = \mathbf{R}^\mathsf{T}(\chi_f)\big(\boldsymbol{p} - \boldsymbol{q}(s)\big), \tag{4}$$

where,

$$\mathbf{R}(\chi_f) = \begin{bmatrix} \cos\chi_f & -\sin\chi_f \\ \sin\chi_f & \cos\chi_f \end{bmatrix}, \tag{5}$$

is the rotation matrix from the Serret-Frenet frame to the inertial frame.

By differentiating Eq. (4) with respect to time, it follows that

$$\dot{\boldsymbol{e}} = \dot{\mathbf{R}}^\mathsf{T}(\chi_f)\big(\boldsymbol{p} - \boldsymbol{q}(s)\big) + \mathbf{R}^\mathsf{T}(\chi_f)\big(\dot{\boldsymbol{p}} - \dot{\boldsymbol{q}}(s)\big), \tag{6}$$

where,

$$\dot{\mathbf{R}}(\chi_f) = \mathbf{R}(\chi_f)\mathbf{S}(\dot{\chi}_f), \tag{7}$$

where, $\mathbf{S}(\dot{\chi}_f)$ is the skew-symmetric matrix,

$$\mathbf{S}(\dot{\chi}_f) = \begin{bmatrix} 0 & -\dot{\chi}_f \\ \dot{\chi}_f & 0 \end{bmatrix}. \tag{8}$$

It follows from Eq. (1) that

$$\dot{\boldsymbol{p}} = \mathbf{R}(\chi)\begin{bmatrix} V \\ 0 \end{bmatrix}. \tag{9}$$

Notice that $\dot{\boldsymbol{q}}$ is the time derivative of the point $\boldsymbol{q}(s)$, whose speed is represented by $\dot{s}$ when expressed in the Serret-Frenet frame. It follows that

$$\dot{\boldsymbol{q}} = \mathbf{R}(\chi_f)\begin{bmatrix} \dot{s} \\ 0 \end{bmatrix}. \tag{10}$$

Subsequently, by substituting Eqs. (9) and (10) in Eq. (6), we obtain,

$$\dot{\boldsymbol{e}} = -\mathbf{S}(\dot{\chi}_f)\boldsymbol{e} + \mathbf{R}(\chi - \chi_f)\begin{bmatrix} V \\ 0 \end{bmatrix} - \begin{bmatrix} \dot{s} \\ 0 \end{bmatrix}, \tag{11}$$

where $\mathbf{R}^\mathsf{T}(\chi_f)\mathbf{R}(\chi) = \mathbf{R}(\chi - \chi_f)$.

Let now the error course angle $\widetilde{\chi}$ be defined by

$$\widetilde{\chi} \triangleq \chi - \chi_f. \tag{12}$$

Hence,

$$\dot{\widetilde{\chi}} = \dot{\chi} - \dot{\chi}_f = \omega - \dot{\chi}_f, \tag{13}$$

and

$$\dot{\chi}_f \triangleq \frac{\mathrm{d}\chi_f}{\mathrm{d}t} = \frac{\mathrm{d}\chi_f}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}t} = \kappa(s)\dot{s}, \tag{14}$$

where $\kappa(s)$ is the curvature of the path at $\boldsymbol{q}(s)$.

The error kinematic model of a fixed-wing UAV for the path following problem with respect to the Serret-Frenet frame is summarized as follows,

$$\dot{e}_s = V\cos\widetilde{\chi} - (1 - \kappa(s)e_d)\dot{s}, \tag{15a}$$
$$\dot{e}_d = V\sin\widetilde{\chi} - \kappa(s)e_s\dot{s}, \tag{15b}$$
$$\dot{\widetilde{\chi}} = \omega - \kappa(s)\dot{s}. \tag{15c}$$

From Eqs. (15), the path following problem reduces into a problem of driving the errors to zero as the UAV approaches the given path. In Micaelli and Samson (1993), the point $\boldsymbol{q}$ is found by projecting $\boldsymbol{p}$ on the path, assuming the projection is well defined. Hence, the control law derived by Micaelli and Samson (1993) requires a stringent restriction on the initial position of $\boldsymbol{q}$ in order to avoid singularities. In contrast, Lapierre and Soetanto (2007) proposed to employ a moving Serret-Frenet frame along the path, which effectively provides an extra control parameter $\dot{s}$ allowing $\boldsymbol{q}(s)$ to evolve along the error states. This control parameter then mitigates the stringent restriction on the initial condition arose in Micaelli and Samson (1993).

3. PATH FOLLOWING CONTROLLER DESIGN

In this section we present a nonlinear path following control logic, which steers the UAV to the reference path with an inaccurately known time constant λ_ϕ. Beginning with the derivation of a kinematic control law for the heading rate command, we apply backstepping to derive a roll control command for a fixed-wing UAV, which, in turn, induces an equivalent control effort to the kinematic control law. In addition, we apply a parameter adaptation technique to deal with an inaccurately known λ_ϕ.

3.1 Kinematic controller design

Following an approach similar to Lapierre and Soetanto (2007) we first derive a kinematic control law for the heading rate. We introduce a bounded differentiable function with respect to the cross-track error e_d, as follows

$$\delta(e_d) \triangleq -\chi^\infty \frac{e^{2ke_d} - 1}{e^{2ke_d} + 1} \tag{16}$$

where, $k > 0$ and $\chi^\infty \in (0, \pi/2)$ [Micaelli and Samson (1993)]. This function is so-called the *approach angle*, since it provides the desired relative course transition of the UAV to the path as a function of the cross-track error e_d. When the cross-track error e_d is zero, the approach angle vanishes, thus imposing the condition that the course angle of the UAV must be tangential to the path. The positive constant k sets the effectiveness of the transient maneuver during approach. The approach angle provides a behavior similar to that of a 'good helmsman' [Rysdyk (2003)] since it satisfies the following condition:

$$e_d \sin\big(\delta(e_d)\big) \leq 0, \qquad \text{for all } e_d. \tag{17}$$

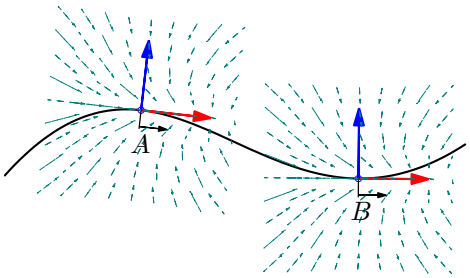


Fig. 2. Local vector fields generated with respect to two distinct points on the path.

This condition steers the UAV to the path along the correct direction (turn left when the UAV is on the right side of the path, and turn right in the opposite situation).

The kinematic controller for path following using the error kinematic model in Eq. (15) is given below.

Proposition 1. [Lapierre et al. (2007)] Consider the kinematic error model of the UAV described in Eq. (15) and the approach angle $\delta(e_d)$ defined in Eq. (16). Assume that the speed of the UAV is non-negative. Suppose that the reference path is parameterized by the arc-length s, and at each s the variables κ, e_s, e_d, and χ_f are well defined. Then, the following kinematic control law,

$$\omega = -k_\omega\big(\widetilde{\chi} - \delta(e_d)\big) + \kappa(s)\dot{s} + \delta'(e_d)\big(V\sin\widetilde{\chi} - \kappa(s)e_s\dot{s}\big) - \frac{e_d V}{\gamma}\left(\frac{\sin\widetilde{\chi} - \sin\big(\delta(e_d)\big)}{\widetilde{\chi} - \delta(e_d)}\right), \tag{18a}$$

$$\dot{s} = k_s e_s + V\cos\widetilde{\chi}, \tag{18b}$$

where k_s, k_ω and γ are positive constants, asymptotically drives e_s, e_d, and $\widetilde{\chi}$ towards zero.

Remarks. The objective of the closed loop system is to steer the error states towards zero, so that the UAV follows the virtual target moving on the path. It should be noted that the approach angle plays an important role in steering the UAV to the reference path. By constructing a vector field with respect to a reference point on the path the UAV is guided to the path, as shown in Fig. 2. As mentioned earlier, the moving reference frame in conjunction with the extra control command $\dot{s}$ helps the errors converge to zero. In particular, the term $k_s e_s$ in Eq. (18) ensures the convergence of e_s to zero, providing the momentum induced by the virtual target to reduce e_s. Figure 2 shows the vector fields with respect to the corresponding reference points on the path, which takes into account the relative momentum induced by the virtual target (drawn by dashed arrows). During implementation, the new location of the virtual target on the path is propagated using $\dot{s}$ obtained via a numerical integration scheme, such as the forward Euler method. Hence, if the reference path is well defined in terms of the arc-length parameter s, then the corresponding variables e_s, e_d, χ_f, and κ as also well defined, and thus the proposed path following algorithm is well-posed.

3.2 Roll angle command via backstepping

For a fixed-wing UAV flying at a constant altitude, a roll angle command is often used for heading control. A simple way is to compute the roll angle command directly from the heading rate command ω_d using the kinematic relationship in Eq. (2),

$$\phi_d = \tan^{-1}\left(\frac{V\omega_d}{g}\right). \tag{19}$$

Note that the actual response of the roll angle differs from the commanded value of Eq. (19), but it can be approximated by a first-order system described as in Eq. (3). Taking into account the approximated model of Eq. (3), one can design a steering logic for the roll angle command ϕ_c in order to achieve $\omega \to \omega_d$ through $\phi \to \phi_d$. Following the standard backstepping technique [Spooner et al. (2002)], we introduce an auxiliary control input ν for the roll angle by augmenting the error model in Eq. (15) with $\dot{\phi} = \nu$. Thus, the system is given as follows,

$$\dot{e}_s = V\cos\widetilde{\chi} - (1 - \kappa(s)e_d)\dot{s}, \tag{20a}$$

$$\dot{e}_d = V\sin\widetilde{\chi} - \kappa(s)e_s\dot{s}, \tag{20b}$$

$$\dot{\widetilde{\chi}} = \frac{g}{V}\tan\phi - \kappa(s)\dot{s}, \tag{20c}$$

$$\dot{\phi} = \nu, \tag{20d}$$

where the auxiliary control input ν is associated with the roll command by

$$\nu = \frac{1}{\lambda_\phi}\big(\phi_c - \phi\big). \tag{21}$$

Suppose that the desired heading rate ω_d is obtained from Eq. (18a) and let $\omega_e \triangleq \omega - \omega_d = (g/V)\tan\phi - \omega_d$ be an error state for the heading rate, whose time derivative is calculated using $(\tan\phi)' = \sec^2\phi$, as follows

$$\dot{\omega}_e = \frac{g}{V}\dot{\phi}\sec^2\phi - \dot{\omega}_d. \tag{22}$$

Proposition 2. Consider the kinematic error model of the UAV described in Eqs. (20) and the approach angle $\delta(e_d)$ defined in Eq. (16). Assume that the speed of the UAV is non-negative. Suppose that the reference path is parameterized by the arc-length s, and at each s the variables κ, e_s, e_d, and χ_f are well defined. Then, the control input,

$$\nu = \frac{V}{g\sec^2\phi}\Big(-k_e\omega_e - \widetilde{\chi} + \delta(e_d) + \dot{\omega}_d\Big), \tag{23}$$

where k_e is a positive constant and ω_d is given in Eq. (18a), asymptotically drives e_s, e_d, and $\widetilde{\chi}$ towards zero, while $\omega \to \omega_d$. It should be noted that $\dot{\omega}_d$ can be calculated by numerical differentiation of ω_d. However, in order to avoid any high frequency noise effect due to differentiation, $\dot{\omega}_d$ in the actual implementation is obtained by using a pseudo differentiation filter in conjunction with saturation, as a transfer function $D(s) = \mathrm{sat}\big(s/(\tau s + 1)\big)$ where $\tau > 0$ is employed.

Proof. Let V_1 be a candidate Lyapunov function given by,

$$V_1 \triangleq \frac{1}{2\gamma}\big(e_s^2 + e_d^2\big) + \frac{1}{2}\big(\widetilde{\chi} - \delta(e_d)\big)^2 + \frac{1}{2}\omega_e^2. \tag{24}$$

Differentiating with respect to time one obtains,

$$\begin{aligned}\dot{V}_1 &= \frac{1}{\gamma}\big(e_s\dot{e}_s + e_d\dot{e}_d\big) + \big(\widetilde{\chi} - \delta(e_d)\big)\big(\dot{\widetilde{\chi}} - \dot{\delta}(e_d)\big) + \omega_e\dot{\omega}_e \\ &= \frac{1}{\gamma}e_s(V\cos\widetilde{\chi} - \dot{s}) + \frac{1}{\gamma}e_d V\sin\big(\delta(e_d)\big) \\ &\quad + \big(\widetilde{\chi} - \delta(e_d)\big)\Big\{\frac{g}{V}\tan\phi + \omega_d - \omega_d - \kappa(s)\dot{s} \\ &\quad - \delta'(e_d)\big(V\sin\widetilde{\chi} - \kappa(s)e_s\dot{s}\big) + \frac{e_d V}{\gamma}\frac{\sin\widetilde{\chi} - \sin\big(\delta(e_d)\big)}{\widetilde{\chi} - \delta(e_d)}\Big\} \\ &\quad + \omega_e\Big(\frac{g}{V}\dot{\phi}\sec^2\phi - \dot{\omega}_d\Big).\end{aligned} \tag{25}$$

By the definition of ω_e, and substituting the desired ω_d from Eq. (18a) and $\dot{s}$ from Eq. (18b), the term inside of the bracket collapses to

$$\dot{V}_1 = -\frac{k_s}{\gamma}e_s^2 + \frac{e_d V}{\gamma}\sin\big(\delta(e_d)\big) - k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)^2 + \omega_e\bigg(\frac{g}{V}\dot{\phi}\sec^2\phi + \widetilde{\chi} - \delta(e_d) - \dot{\omega}_d\bigg). \tag{26}$$

Choosing $\dot{\phi}$ through the auxiliary control input in Eq. (23), that is,

$$\dot{\phi} = \nu = \frac{V}{g\sec^2\phi}\bigg(-k_e\omega_e - \widetilde{\chi} + \delta(e_d) + \dot{\omega}_d\bigg), \tag{27}$$

results in,

$$\dot{V}_1 = -\frac{k_s}{\gamma}e_s^2 + \frac{e_d V}{\gamma}\sin\big(\delta(e_d)\big) - k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)^2 - k_e\omega_e^2 \leq 0. \tag{28}$$

The last inequality implies, in particular, that $e_s, e_d, \tilde{\chi}$ and ω_e are bounded. Furthermore, $\delta(e_d)$ is also bounded from (16) since e_d is bounded. It is also easy to show, using (18a), that ω_d is bounded. To this end, notice that all signals in the rhs of (18a) except the last are bounded. Moreover, since

$$\frac{\sin\widetilde{\chi} - \sin\big(\delta(e_d)\big)}{\widetilde{\chi} - \big(\delta(e_d)\big)} = \operatorname{sinc}\bigg(\frac{1}{2}\big(\widetilde{\chi} - \delta(e_d)\big)\bigg)\cos\bigg(\frac{1}{2}\big(\widetilde{\chi} + \delta(e_d)\big)\bigg) \tag{29}$$

and since the sinc function is bounded, the last term in the rhs of (18a) is also bounded. Since ω_d and ω_e are bounded, it follows that $|\phi| < \pi/2$ and the control (23) is well defined.

To complete the proof, note that V_1 is radially unbounded, hence the set $\Omega_c = \{V_1(e_s, e_d, \widetilde{\chi}, \omega_e) \leq c\}$ is a compact, positively invariant set. Let E_1 be the set of all points in Ω_c where $\dot{V}_1 = 0$. The set E_1 is given by $E_1 = \{e_s = e_d = 0,\ \widetilde{\chi} = \delta, \text{and } (g/V)\tan\phi = \omega_d\}$. Noticing that $\delta(\cdot)$ is a function of the cross-track error e_d, one can easily verify that any point starting from the set E_1 will remain in the set, i.e. E_1 is an invariant set. By the LaSalle's invariance principle, we have that every trajectory starting inside the set Ω_c approaches E_1 as $t \to \infty$. Therefore, $\lim_{t\to\infty} e_s = 0$, $\lim_{t\to\infty} e_d = 0$, and $\lim_{t\to\infty}\widetilde{\chi} = 0$ since $\delta(e_d) \to 0$. We also have $(g/V)\tan\phi \to \omega_d$ as $t \to \infty$. □

3.3 Parameter adaptation

Note that the actual roll angle command is computed from Eq. (3) and (23),

$$\phi_c = \lambda_\phi\nu + \phi, \tag{30}$$

where, λ_ϕ is the time constant of the system.

In practice, λ_ϕ is determined by the characteristics of the UAV airframe and PID gains, thus it is not known accurately. The actual roll command $\widehat{\phi}_c$ is then given by

$$\widehat{\phi}_c = \widehat{\lambda}_\phi\nu + \phi, \tag{31}$$

where $\widehat{\lambda}_\phi$ is an estimate of λ_ϕ.

In order to compensate for any uncertainty in λ_ϕ, we apply a parameter adaptation technique. To this end, let V_2 be a candidate Lyapunov function defined by

$$V_2 \triangleq V_1 + \frac{1}{2}\frac{1}{k_a\lambda_\phi}\widetilde{\lambda}_\phi^2, \tag{32}$$

where V_1 is given in Eq. (24) and $\widetilde{\lambda}_\phi \triangleq \lambda_\phi - \widehat{\lambda}_\phi$ is the estimate error. Differentiating with respect to time, we get

$$\dot{V}_2 = \dot{V}_1 + \frac{1}{k_a\lambda_\phi}\widetilde{\lambda}_\phi\dot{\widetilde{\lambda}}_\phi = -\frac{k_s}{\gamma}e_s^2 + \frac{e_d V}{\gamma}\sin\big(\delta(e_d)\big) - k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)^2 + \omega_e\bigg(\frac{g}{V}\dot{\phi}\sec^2\phi + \widetilde{\chi} - \delta(e_d) - \dot{\omega}_d\bigg) + \frac{1}{k_a\lambda_\phi}\widetilde{\lambda}_\phi\dot{\widetilde{\lambda}}_\phi. \tag{33}$$

The actual value of $\dot{\phi}$ is calculated from Eq. (3) in conjunction with the actual roll command in Eq. (31) and ν given in Eq. (23). Subsequently, substituting the actual $\dot{\phi}$ in Eq. (33), we get

$$\dot{V}_2 = -\frac{k_s}{\gamma}e_s^2 + \frac{e_d V}{\gamma}\sin\big(\delta(e_d)\big) - k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)^2 - \frac{\widehat{\lambda}_\phi}{\lambda_\phi}k_e\omega_e^2 + \omega_e\frac{\widetilde{\lambda}_\phi}{\lambda_\phi}\big(\widetilde{\chi} - \delta(e_d) - \dot{\omega}_d\big) + \frac{1}{k_a\lambda_\phi}\widetilde{\lambda}_\phi\dot{\widetilde{\lambda}}_\phi. \tag{34}$$

Choosing $\dot{\widetilde{\lambda}}_\phi$ as,

$$\dot{\widetilde{\lambda}}_\phi = -k_a\omega_e\big(\widetilde{\chi} - \delta(e_d) - \dot{\omega}_d\big), \tag{35}$$

where k_a is a positive constant. It follows that

$$\dot{V}_2 = -\frac{k_s}{\gamma}e_s^2 + \frac{e_d V}{\gamma}\sin\big(\delta(e_d)\big) - k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)^2 - \frac{\widehat{\lambda}_\phi}{\lambda_\phi}k_e\omega_e^2 \leq 0. \tag{36}$$

Assuming λ_ϕ is constant, the parameter update law is readily obtained from Eq. (35) as follows,

$$\dot{\widehat{\lambda}}_\phi = k_a\omega_e\big(\widetilde{\chi} - \delta(e_d) - \dot{\omega}_d\big). \tag{37}$$

Proposition 3. Let the control law in Eqs. (18) and (23). With the parameter update law given by Eq. (37) the actual roll command of Eq. (31) guarantees that the signals e_s, e_d, and $\widetilde{\chi}$ asymptotically tend to zero, while $(g/V)\tan\phi \to \omega_d$.

Proof. In order to prove the proposition notice that $\lim_{t\to\infty} V_2(t)$ exists since V_2 is bounded from below and is non-increasing. It therefore suffices to show that $\dot{V}_2$ is uniformly continuous, in which case we can invoke Barbalat's lemma to ensure that $\lim_{t\to\infty}\dot{V}_2(t) = 0$.

To this end, consider the expression of $\dot{V}_2$ in Eq. (36), from which it follows that e_d, e_s, $\widetilde{\chi}$, ω_e, $\hat{\lambda}_\phi$, and $\widetilde{\lambda}_\phi$ are all bounded. As with the proof of Proposition 2, it follows from (18b) that $\dot{s}$ is bounded, and from (18a) that ω_d is bounded as well. The boundedness of ω_d and ω_e imply that $|\phi| < \pi/2$. Using (20) we conclude that $\dot{e}_s, \dot{e}_d, \dot{\widetilde{\chi}}$ are also bounded. Differentiating Eq. (18a) with respect to time, one can show that all terms are bounded, while the last term is also bounded since the derivative of the sinc function is bounded (see also Eq. (29)). Hence, it can be shown that $\dot{\omega}_d$ is bounded. Furthermore, $\delta(e_d)$ and its time derivative $\dot{\delta}(e_d) = \delta'(e_d)\dot{e}_d$ are bounded. A straightforward calculation shows that

$$\dot{\omega}_e = \frac{\widehat{\lambda}_\phi}{\lambda_\phi}\big(-k_e\omega_e - \widetilde{\chi} + \delta(e_d)\big) - \frac{\tilde{\lambda}_\phi}{\lambda_\phi}\dot{\omega}_d. \tag{38}$$

From the previous equation it follows that $\dot{\omega}_e$ is bounded. Furthermore, $\dot{\widehat{\lambda}}_\phi$ is also bounded from (37). Consider now the expression for $\ddot{V}_2$, given by

$$\ddot{V}_2 = -\frac{k_s}{\gamma}e_s\dot{e}_s + \frac{V\dot{e}_d}{\gamma}\sin\big(\delta(e_d)\big) + \frac{Ve_d}{\gamma}\dot{\delta}(e_d)\cos\big(\delta(e_d)\big) - 2k_\omega\big(\widetilde{\chi} - \delta(e_d)\big)\big(\dot{\widetilde{\chi}} - \dot{\delta}(e_d)\big) - \frac{\widehat{\lambda}_\phi}{\lambda_\phi}k_e\omega_e\dot{\omega}_e, \tag{39}$$

Table 1. Simulation parameters.

V = 20 [m/s]	$k = 0.01$	$k_s = 0.4$	$k_\omega = 0.001$	$\lvert\phi_c^{\max}\rvert = \pi/6$
$k_e = 1.1$	$k_a = 0.7$	$\gamma = 4000$	$\lambda_\phi \approx 1.1$	

where, all expressions in the rhs of the equation have been shown to be bounded. It follows that $\ddot{V}_2$ is bounded and hence $\dot{V}_2$ is uniformly continuous. Applying Barbalat's lemma, it follows that $\dot{V}_2 \to 0$ as $t \to 0$. □

4. HARDWARE IN-THE-LOOP SIMULATION RESULTS

4.1 Simulation environment

A realistic hardware-in-the-loop simulation (HILS) environment has been developed to validate the UAV autopilot hardware and software development utilizing Matlab® and Simulink®. A full 6-DOF nonlinear aircraft model is used in conjunction with a linear approximation of the aerodynamic forces and moments, along with Earth gravitational (WGS-84) and magnetic field models. Detailed models for the sensors and actuators have also been incorporated. Four independent computer systems are used in the hardware-in-the-loop simulation (HILS) as illustrated in Fig. 3. A 6-DOF simulator, the flight visualization computer, the autopilot micro-controller, and the ground station computer console are involved in the simulation. Further details about the UAV platform, autopilot and HILS set-up can be found in Jung et al. (2005); Jung and Tsiotras (2007a) and Jung and Tsiotras (2007b).

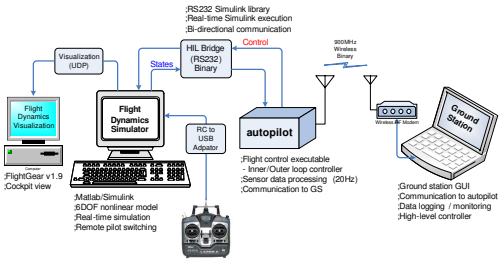


Fig. 3. High fidelity hardware-in-the-loop simulation (HILS) environment for validating the path following algorithm

4.2 Simulation results

This section illustrates the performance of the derived path-following control law. The reference path is given by a quartic B-spline over a non-decreasing knot parameter u which is monotonic to the arc-length s. Hence, we can compute $\boldsymbol{q}(s)$, $\chi_f(s)$, and $\kappa(s)$ in terms of the knot parameter as follows,

$$\chi_f(s) = \tan^{-1}\frac{y'}{x'}, \quad \kappa(s) = \frac{x'y'' - x''y'}{(x'^2 + y'^2)^{3/2}}. \tag{40}$$

where $(\cdot)'$ and $(\cdot)''$ are the derivatives with respect to u. The knot parameter is propagated along with Eq. (18) as follows,

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \frac{\mathrm{d}s}{\mathrm{d}t}\Big/\frac{\mathrm{d}s}{\mathrm{d}u} = \dot{s}/\sqrt{x'^2 + y'^2}. \tag{41}$$

The parameters used in the simulations are shown in Table 1. We present the results from two simulations. In the first case, we calculate the roll angle command from Eq. (30) using a time constant $\lambda_\phi = 0.4$ which is inaccurately known. Without parameter adaptation, as shown in Fig. 4, we observe a sluggish and low damped response of the actual trajectory. The error variables are shown in Fig. 5. In the second case, we calculate the roll angle command from Eq. (31) with the parameter update law of Eq. (37). Figures 6-8 show the results. The error states tend to zero asymptotically as shown in Fig. 7. Because the roll angle command is limited within $\pm\pi/6$, the UAV is unable to exactly follow the path where the curvature exceeds the maximum curvature achievable by the UAV at the speed of 20 [m/sec]. Nevertheless, the path following control law forces the UAV converge to the path asymptotically after a short transient. It should be noted that the simulation was carried out in conjunction with realistic sensor noises and constant wind disturbance, showing that the adaptation law is more or less robust to the presence of such measurement noises and disturbances. Fig. 8 displays the time history of the estimate of λ_ϕ, which shows that from the initial guess of $\widehat{\lambda}_\phi = 0.4$, the parameter estimate converges to the actual value of $\lambda_\phi \approx 1.1$.

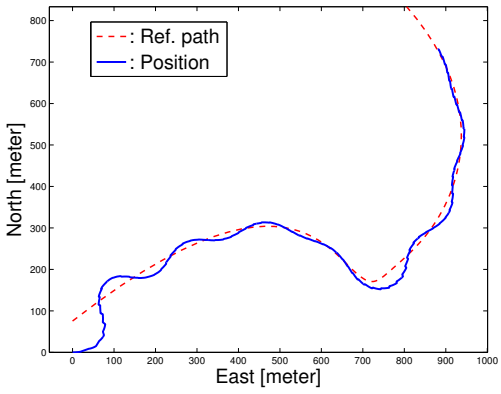
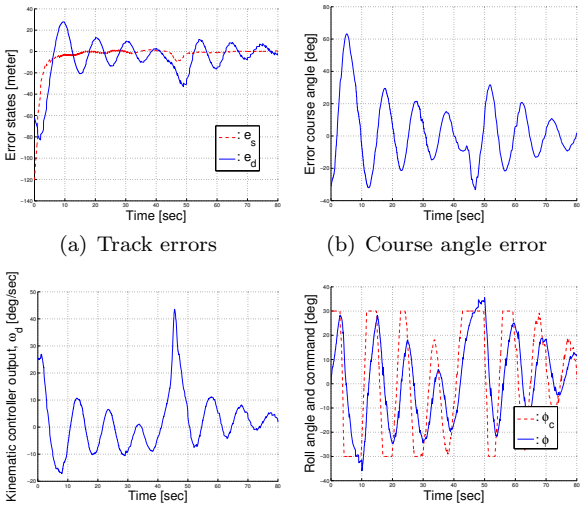


Fig. 4. Reference path and actual trajectory of the UAV without parameter adaptation.



(a) Track errors (b) Course angle error

(c) Heading rate command (d) ϕ v/s ϕ_c

Fig. 5. Error states and command inputs without parameter adaptation.

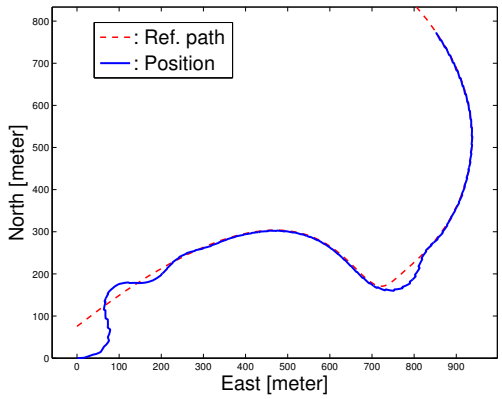
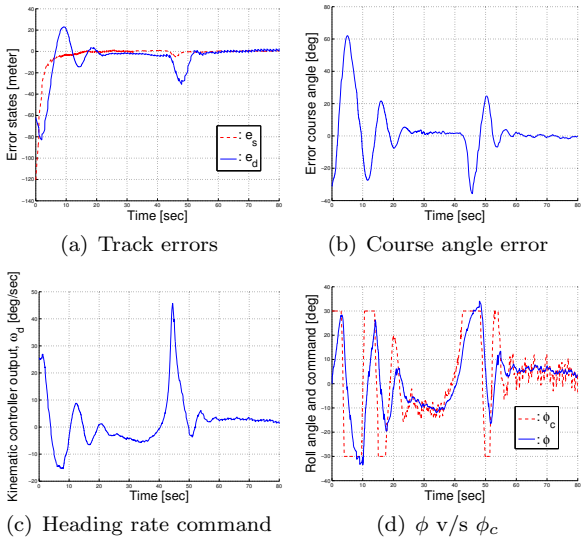


Fig. 6. Reference path and actual trajectory of the UAV with parameter adaptation.



(a) Track errors (b) Course angle error

(c) Heading rate command (d) ϕ v/s ϕ_c

Fig. 7. Error states and command inputs with parameter adaptation.

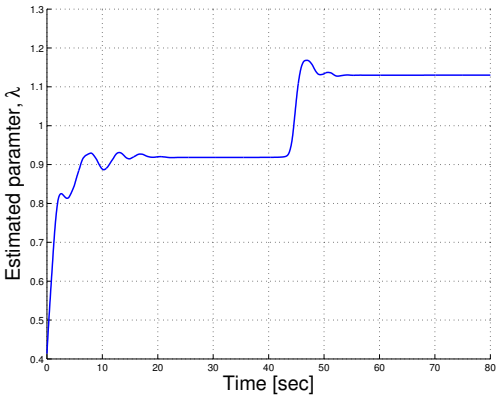


Fig. 8. Parameter estimate of λ_ϕ.

5. CONCLUSION

In this research a nonlinear path following control law has been developed for a small UAV using backstepping of the heading rate command. The kinematic control law realizes cooperative path following so that the motion of a virtual target is controlled by an extra control input to help the convergence of the error variables. A roll command that gives rise to the desired heading rate has been derived by taking into account the inaccurate system time constant. The path following control algorithm is validated through a high-fidelity hardware-in-the-loop simulation (HILS) environment, which verifies the applicability of the presented algorithm to the actual UAV.

ACKNOWLEDGEMENTS

Partial support for this work has been provided by NSF award CMS-0510259.

REFERENCES

D. Jung, E. J. Levy, D. Zhou, R. Fink, J. Moshe, A. Earl, and P. Tsiotras. Design and Development of a Low-Cost Test-Bed for Undergraduate Education in UAVs. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2739–2744, Seville, Spain, December 2005.

D. Jung and P. Tsiotras. Inertial Attitude and Position Reference System Development for a Small UAV. In *AIAA Infotech at Aerospace*, Rohnert Park, CA, May 2007a. AIAA Paper 07-2768.

D. Jung and P. Tsiotras. Modelling and Hardware-in-the-loop Simulation for a Small Unmanned Aerial Vehicle. In *AIAA Infotech at Aerospace*, Rohnert Park, CA, May 2007b. AIAA Paper 07-2763.

L. Lapierre and D. Soetanto. Nonlinear Path-following Control of an AUV. *Ocean Engineering*, 34:1734–1744, 2007.

L. Lapierre, R. Zapata, and P. Lepinay. Combined Path-following and Obstacle Avoidance Control of a Wheeled Robot. *The International Journal of Robotics Research*, 26(4):361–375, April 2007.

A. Micaelli and C. Samson. Trajectory Tracking for Unicycle-type and Two-Steering-Wheels Mobile Robots. Technical Report 2097, INRIA, Sophia-Antipolis, November 1993.

D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard. Vector Field Path Following for Small Unmanned Air Vehicles. In *Proceedings of the 2006 American Control Conference*, pages 5788–5794, Minneapolis, MN, June 2006.

M. Niculescu. Lateral Track Control Law for Aerosonde UAV. In *39th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 2001. A01-16013.

S. Park, J. Deyst, and J. P. How. A New Nonlinear Guidance Logic for Trajectory Tracking. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, August 2004. AIAA-2004-4900.

W. Ren and R. W. Beard. Trajectory Tracking for Unmanned Air Vehicles with Velocity and Heading Rate Constraints. *IEEE Transactions on Control Systems Technology*, 12(5):706–716, September 2004.

R. Rysdyk. UAV Path Following for Constant Line-Of-Sight. In *2nd AIAA Unmanned Unlimited Conference and Workshop and Exhibit*, San Diego, CA, September 2003. AIAA-2003-6626.

J. T. Spooner, M. Maggiore, R. Ordóñez, and K. M. Passino. *Stable Adaptive Control and Estimation for Nonlinear Systems*. A John Wiley & Sons, Inc., New York, NY, 2002.