Proceedings of the EuroGNC 2013, 2nd CEAS Specialist Conference
on Guidance, Navigation & Control, Delft University of Technology,
Delft, The Netherlands, April 10-12, 2013

Comparison of Nonlinear Dynamic Inversion and Backstepping Controls with Application to a Quadrotor

Jian Wang, Florian Holzapfel, Florian Peter[1]

Abstract **This paper presents a comparison between different control designs using Nonlinear Dynamic Inversion (NDI) and Backstepping methodologies. Most of control design variations of the two mentioned methods, if not all, are concluded here. Similarities and differences are compared not only between NDI and Backstepping, but also between different designs of the same method, where the output tracking error dynamics are used as an important criterion for the comparison. Due to the high maneuverability and agility of the quadrotor, the control bandwidth of the designs is of particular interest, which is related to the requirement on the Time Scale Separation (TSS) in the control system. Through the comparison, the related issues are clarified, e.g. if the additional Backstepping term reduces the TSS compared with the NDI designs; which control designs have the highest control bandwidth. The attitude control of a quadrotor is used as an example system to explain and verify the comparison.**

Nomenclature

$[\Phi;\ \Theta;\ \Psi]$	Euler angles, bank, pitch and azimuth angle respectively
$\vec{\boldsymbol{\omega}}$	$=[p \quad q \quad r]^T$, angular rates around x, y and z axis of frame B
$\vec{\boldsymbol{M}}$	$=[L \quad M \quad N]^T$, total moments around x, y and z axis of frame B
T	Total thrust of the quadrotor
$\boldsymbol{J}$	Moment of inertia of the C.G: point
$\boldsymbol{K_i}$	Diagonal gain matrices used in dynamic inversion designs
$\boldsymbol{A_i}$	Diagonal gain matrices used in Backstepping designs
$\boldsymbol{x_r}$	Reference command
$\boldsymbol{x_{i,des}}$	Desired state, $i = 2,3,\dots,n$
$\hat{\boldsymbol{x}}_{\boldsymbol{i,des}}$	Estimated desired state by low pass filter
$\boldsymbol{e}$	$= \boldsymbol{x_{1r}} - \boldsymbol{x_1}$, Output tracking error
$\boldsymbol{z_i}$	$= \boldsymbol{x_{i,des}} - \boldsymbol{x_i}$, Backstepping error

[1] Institute of Flight System Dynamics, TU München, Germany, 85748.
Email: jian.wang@tum.de, florian.holzapfel@tum.de. florian.peter@tum.de

1. Introduction

Quadrotor helicopters have gained increasing interest as research platforms for its VTOL and hover capabilities and extremely good robustness. Backstepping [1] and Nonlinear Dynamic Inversion [2] are two of the most popular nonlinear control methods to be applied on quadrotors. As the two methods are very similar and in some case result the same control law, a comparison is worthwhile to analyze the pros and cons of each design.

Previous effort on quadrotor control using these two methods includes cascaded NDI designs [3-5], non-cascaded NDI designs [6-7] and cascaded Backstepping design [8-10] and Backstepping designs using command filters [11]. Farrell [12] introduced an additional compensation term to the Backstepping design using command filters, named 'Command Filtered Backstepping (CFB)'. There are also studies on the comparison of some particular applications of these two methods. For instance in [13], the block Backstepping and cascaded NDI are compared. However, this paper aims to make a complete comparison of different types of control design structures and generalize the disadvantages and advantages. The control bandwidth is of particular interest, which is limited by the actuator dynamics, sensor availability and quality, and possibly reduction of bandwidth induced by the control design (e.g. TSS needed with cascaded control design). The objective is to reduce or eliminate the reduction of bandwidth due to the control design as much as possible, and to design a controller for the quadrotor with high control bandwidth as well as adequate robustness.

In total there are two types of NDI designs and five types of Backstepping designs. They are listed as follows,

a) Non-cascaded NDI design
b) Analytical Backstepping with linear state feedback
c) Analytical Backstepping with nonlinear state feedback
d) Cascaded NDI design
e) Cascaded Backstepping design
f) Backstepping using command filter
g) Command filtered Backstepping

In this work, the attitude control of a quadrotor is used to apply all the seven designs because first, all the seven designs can be realized on it; second, it is the most common and well-known system, clear comparison can be made. From the other direction, it is also make sense to use nonlinear control strategies on the attitude control of a quadrotor. Linear PID control only works well in the near hovering condition, but as soon as large attitude angle flight is demanded, nonlinear control would outperform the linear control in the nonlinear region.

In section 2, a brief introduction on the attitude dynamics of the quadrotor is given. Detailed derivations of all designs and their comparison are illustrated in section 3, followed by the simulation and experimental results in section 4. Finally, the conclusion is given in the section 5.

2. System Representations

2.1 Systems of Strict-feedback Form

System of feedback form [14-15], also called lower triangular form, is a special class of system referring to the structure of the system dynamic equations. A sub-class of system is the strict-feedback form [14-15], which has affine system inputs in each dynamic equation. For a second order dynamic system, the following general form is used,

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(x_1) + \boldsymbol{G}_1(x_1)\boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2(x_1, x_2) + \boldsymbol{G}_2(x_1, x_2)\boldsymbol{u} \end{cases} \tag{1}$$

The seven control designs can be all applied to system of strict-feedback form, but the first three non-cascaded methods might not be able to be applied on system of other forms. For the system of strict-feedback form, the derivatives of $\boldsymbol{f}_1$ and $\boldsymbol{G}_1$ normally exist and can be expressed analytically. Design c) Analytical Back-stepping with nonlinear state feedback can be directly applied, while the design a) and b) need a system transformation in the form (2) to be applicable. This system transformation is equivalent with the original system, just another representation. The nonlinear state $\boldsymbol{x}_2$ could be replaced by a linear state $\boldsymbol{x}_3$ with respect to the output state $\boldsymbol{x}_1$,

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_3 \\ \dot{\boldsymbol{x}}_3 = \ddot{\boldsymbol{x}}_1 = \frac{d}{dt}(\boldsymbol{f}_1 + \boldsymbol{G}_1\boldsymbol{x}_2) = \dot{\boldsymbol{f}}_1 + \dot{\boldsymbol{G}}_1\boldsymbol{x}_2 + \boldsymbol{G}_1\dot{\boldsymbol{x}}_2 \end{cases} \tag{2}$$

And from Eq. (1), we have,

$$\begin{cases} \boldsymbol{x}_2 = \boldsymbol{G}_1^{-1}(\dot{\boldsymbol{x}}_1 - \boldsymbol{f}_1) \\ \dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2 + \boldsymbol{G}_2\boldsymbol{u} \end{cases} \tag{3}$$

Substituting the above equation into the Eq. (2), we have the new transformed system represented as follows,

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_3 \\ \dot{\boldsymbol{x}}_3 = \boldsymbol{f}_3 + \boldsymbol{G}_3\boldsymbol{u} \end{cases} \tag{4}$$

where,

$$\begin{cases} \boldsymbol{f}_3 = \dot{\boldsymbol{f}}_1 + \dot{\boldsymbol{G}}_1\boldsymbol{G}_1^{-1}(\dot{\boldsymbol{x}}_1 - \boldsymbol{f}_1) + \boldsymbol{G}_1\boldsymbol{f}_1 \\ \boldsymbol{G}_3 = \boldsymbol{G}_1\boldsymbol{G}_2 \end{cases} \tag{5}$$

If the system doesn't preserve the strict-feedback form, e.g. the system function $\boldsymbol{f}_1$ or $\boldsymbol{G}_1$ is a function of the inner state $\boldsymbol{x_2}$ or even the input $\boldsymbol{u}$. Hence, the derivative of $\boldsymbol{f}_1$ or $\boldsymbol{G}_1$ might not be available from measurements or even not exist. For this reason, the designs a), b), and c), which need the analytical expression of the function derivatives, have to apply on system of strict-feedback form. The latter four designs however can be applied to a wider range of systems, as they use low pass filters to estimate the intermediate controls instead of analytical expression containing the derivatives of $\boldsymbol{f}_1$ and $\boldsymbol{G}_1$.

It makes sense to use system of strict-feedback as an example so that all the seven designs can be applied on the same system. The attitude dynamic system is of the strict-feedback form, and it can be expressed in the above two system forms. The dynamic equations are derived in the next section.

2.2 Attitude Dynamics of the Quadrotor

The system output is the attitude $[\Phi;\ \Theta;\ \Psi]$ and the input is the moments. The control allocation after the controller maps the moments to the motor RPM commands. The mapping details can be found in [8]. Conventional Euler angles are used in the attitude dynamic formulation. The attitude propagation equation from angular rates to the attitude derivatives is,

$$\begin{bmatrix}\dot{\Phi}\\ \dot{\Theta}\\ \dot{\Psi}\end{bmatrix}=\begin{bmatrix}1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta\\ 0 & \cos\Phi & -\sin\Phi\\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta}\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{6}$$

The inverse equation is easy to calculate,

$$\begin{bmatrix}p\\ q\\ r\end{bmatrix}=\begin{bmatrix}1 & 0 & -\sin\Theta\\ 0 & \cos\Phi & \sin\Phi\cos\Theta\\ 0 & -\sin\Phi & \cos\Phi\cos\Theta\end{bmatrix}\begin{bmatrix}\dot{\Phi}\\ \dot{\Theta}\\ \dot{\Psi}\end{bmatrix} \tag{7}$$

The rotational dynamic equation of motion is well-known,

$$\dot{\vec{\boldsymbol{\omega}}}=\boldsymbol{J}^{-1}\cdot\vec{\boldsymbol{M}}-\boldsymbol{J}^{-1}\cdot\vec{\boldsymbol{\omega}}\times\boldsymbol{J}\vec{\boldsymbol{\omega}} \tag{8}$$

Similar with previous section, the attitude dynamic system can be written as Eq. (1),

$$\begin{cases}\dot{\boldsymbol{x}}_1=\boldsymbol{f}_1+\boldsymbol{G}_1\boldsymbol{x}_2\\ \dot{\boldsymbol{x}}_2=\boldsymbol{f}_2+\boldsymbol{G}_2\boldsymbol{u}\end{cases} \tag{9}$$

Where,

$$\begin{cases} \boldsymbol{x}_1 = [\Phi \quad \Theta \quad \Psi]^T \\ \boldsymbol{x}_2 = [p \quad q \quad r]^T \\ \boldsymbol{u} = \vec{\boldsymbol{M}} \\ \boldsymbol{f}_1 = \boldsymbol{0}_{1\times 3} \\ \boldsymbol{f}_2 = -\boldsymbol{J}^{-1} \cdot \vec{\boldsymbol{\omega}} \times \boldsymbol{J}\vec{\boldsymbol{\omega}} \end{cases} \tag{10}$$

$$\begin{cases} \boldsymbol{G}_1 = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta} \end{bmatrix} \\ \boldsymbol{G}_1^{-1} = \begin{bmatrix} 1 & 0 & -\sin\Theta \\ 0 & \cos\Phi & \sin\Phi\cos\Theta \\ 0 & -\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix} \\ \boldsymbol{G}_2 = \boldsymbol{J}^{-1} \\ \boldsymbol{G}_2^{-1} = \boldsymbol{J} \end{cases} \tag{11}$$

The above system can be also rewritten in the same form as Eq. (2). Further differentiating the Eq. (6), the second derivative of the Euler angles can be propagated from the angular accelerations. And the angular rates can be represented by the attitude derivative using Eq. (7). We have,

$$\begin{bmatrix} \ddot{\Phi} \\ \ddot{\Theta} \\ \ddot{\Psi} \end{bmatrix} = \begin{bmatrix} \dot{\Theta}\dot{\Phi}\tan\Theta + \frac{\dot{\Theta}\dot{\Psi}}{\cos\Theta} \\ -\dot{\Psi}\dot{\Phi}\cos\Theta \\ \frac{\dot{\Theta}\dot{\Phi}}{\cos\Theta} + \dot{\Theta}\dot{\Psi}\tan\Theta \end{bmatrix} + \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta} \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} \tag{12}$$

Summarizing the dynamic equations we have the transformed system,

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_3 \\ \dot{\boldsymbol{x}}_3 = \boldsymbol{f}_3 + \boldsymbol{G}_3\boldsymbol{u} \end{cases} \tag{13}$$

Where,

$$\begin{cases} \boldsymbol{x}_3 = [\dot{\Phi} \quad \dot{\Theta} \quad \Psi]^T \\ \boldsymbol{f}_3 = \begin{bmatrix} \dot{\Theta}\dot{\Phi}\tan\Theta + \frac{\dot{\Theta}\dot{\Psi}}{\cos\Theta} \\ -\dot{\Psi}\dot{\Phi}\cos\Theta \\ \frac{\dot{\Theta}\dot{\Phi}}{\cos\Theta} + \dot{\Theta}\dot{\Psi}\tan\Theta \end{bmatrix} - \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta} \end{bmatrix} \cdot \boldsymbol{J}^{-1} \cdot \vec{\boldsymbol{\omega}} \times \boldsymbol{J}\vec{\boldsymbol{\omega}} \end{cases} \tag{14}$$

$$\begin{cases} \boldsymbol{G}_3 = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta} \end{bmatrix} \boldsymbol{J}^{-1} \\ \boldsymbol{G}_3^{-1} = \boldsymbol{J} \begin{bmatrix} 1 & 0 & -\sin\Theta \\ 0 & \cos\Phi & \sin\Phi\cos\Theta \\ 0 & -\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix} \end{cases} \tag{15}$$

3. NDI and Backstepping Control Designs

3.1 a) Non-cascaded NDI Design

The non-cascaded NDI design structure requires the system matrix to be in companion form [2], i.e., the transformed system in Eq. (13). The relative degree is two.

For the 2nd order attitude control loop, a PD controller for the linear error dynamics is used. Hence we have the following control law,

$$\begin{cases} \boldsymbol{v} = \ddot{\boldsymbol{x}}_{1r} + \boldsymbol{K}_d\dot{\boldsymbol{e}} + \boldsymbol{K}_p\boldsymbol{e} \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{16}$$

Where,

$$\begin{cases} \boldsymbol{e} = \boldsymbol{x}_{1r} - \boldsymbol{x}_1 \\ \dot{\boldsymbol{e}} = \dot{\boldsymbol{x}}_{1r} - \dot{\boldsymbol{x}}_1 = \dot{\boldsymbol{x}}_{1r} - \boldsymbol{x_3} \end{cases} \tag{17}$$

where $\boldsymbol{x}_{1r}$ is the reference attitude command and $\boldsymbol{v}$ is the pseudo control $[\ddot{\Phi}_v \quad \ddot{\Theta}_v \quad \ddot{\Psi}_v]^{\boldsymbol{T}}$. The detailed dynamic inversion equations are derived. The nonlinear dynamics are inverted in two steps. First, the commanded angular acceleration is calculated using the pseudo control signal $\boldsymbol{v}$,

$$\dot{\vec{\boldsymbol{\omega}}}_c = \begin{bmatrix} \ddot{\Phi}_v - \ddot{\Psi}_v\sin\Theta - \dot{\Theta}\dot{\Psi}\cos\Theta \\ \ddot{\Theta}_v\cos\Phi + \ddot{\Psi}_v\sin\Phi\cos\Theta + \dot{\Psi}\dot{\Phi}\cos\Phi\cos\Theta - \dot{\Theta}\dot{\Phi}\sin\Phi - \dot{\Theta}\dot{\Psi}\sin\Phi\sin\Theta \\ -\ddot{\Theta}_v\sin\Phi + \ddot{\Psi}_v\cos\Phi\cos\Theta - \dot{\Psi}\dot{\Phi}\sin\Phi\cos\Theta - \dot{\Theta}\dot{\Phi}\cos\Phi - \dot{\Theta}\dot{\Psi}\cos\Phi\sin\Theta \end{bmatrix} \tag{18}$$

Then the system input, the moments, is calculated from the commanded angular acceleration,

$$\vec{\boldsymbol{M}} = \boldsymbol{J}\dot{\vec{\boldsymbol{\omega}}}_c + \vec{\boldsymbol{\omega}} \times \boldsymbol{J}\vec{\boldsymbol{\omega}} \tag{19}$$

3.2 b) Analytical Backstepping with Linear State Feedback

In every step of Backstepping derivations, there is a virtual control, whose derivative is required in the next step. Analytical Backstepping means these derivatives are solved analytically, not estimated by low pass filters [1]. Backstepping results in slightly different control laws on the original system representation and the transformed system in companion form. It uses nonlinear state feedback when applying it on the original system and linear state feedback on the transformed system.

We look at the Backstepping case with linear state feedback first. For the transformed system, the following Lyapunov function and control law is derived [1],

$$\begin{cases} V = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\boldsymbol{z}_3^T\boldsymbol{z}_3 \\ \boldsymbol{z_3} = \boldsymbol{x_{3,des}} - \boldsymbol{x_3} \end{cases} \tag{20}$$

$$\begin{cases} \boldsymbol{x_{3,des}} = \dot{\boldsymbol{x}}_{1r} + \boldsymbol{A}_1\boldsymbol{e} \\ \boldsymbol{v} = \dot{\boldsymbol{x}}_{3,des} + \boldsymbol{e} + \boldsymbol{A}_3\boldsymbol{z_3} \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{21}$$

The Backstepping method has its own defined error dynamics (different from the tracking error dynamics),

$$\begin{cases} \dot{\boldsymbol{e}} = -\boldsymbol{A_1}\boldsymbol{e} + \boldsymbol{z_3} \\ \dot{\boldsymbol{z}}_3 = -\boldsymbol{A}_3\boldsymbol{z}_3 - \boldsymbol{e} \end{cases} \tag{22}$$

The control law of Eq. (21) can be further derived by analytically solving the $\dot{\boldsymbol{x}}_{\boldsymbol{3,des}}$ and inserting the $\boldsymbol{z_3}$ expression,

$$\dot{\boldsymbol{x}}_{\boldsymbol{3,des}} = \ddot{\boldsymbol{x}}_{1r} + \boldsymbol{A}_1\dot{\boldsymbol{e}} \tag{23}$$

$$\begin{cases} \boldsymbol{v} = \ddot{\boldsymbol{x}}_{1r} + (\boldsymbol{A}_1 + \boldsymbol{A}_3)\dot{\boldsymbol{e}} + (\boldsymbol{A}_1\boldsymbol{A}_3 + \boldsymbol{1})\boldsymbol{e} \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{24}$$

In this case the Backstepping control ends up with exactly the same dynamic inversion equation as in design a). Both designs can have the same linear error dynamics, given the same PD gains. This is the case when Backstepping end up the same with NDI design of full relative degree, giving two conditions: a) the same linear controller is used; b) the plant nonlinearities are fully feedback linearized in the inner loop, meaning no partial inversion in cascaded loops and no useful nonlinearities are kept by the Backstepping method.

The tracking error dynamics for both control designs can be derived by assuming perfect plant model and ignoring the actuator dynamics, i.e., $\boldsymbol{v} = \ddot{\boldsymbol{x}}_1$

$$\begin{cases} \ddot{\boldsymbol{e}} = -\boldsymbol{K}_d\dot{\boldsymbol{e}} - \boldsymbol{K}_p\boldsymbol{e} \\ \ddot{\boldsymbol{e}} = -(\boldsymbol{A}_1 + \boldsymbol{A}_3)\dot{\boldsymbol{e}} - (\boldsymbol{A}_1\boldsymbol{A}_3 + \boldsymbol{1})\boldsymbol{e} \end{cases} \tag{25}$$

Both control designs can achieve the optimal behavior of the tracking error $\boldsymbol{e}$ by assigning appropriate gains. For a second order system, the gains can be assigned by the natural frequency ω and relative damping ζ:

$$\begin{cases} \boldsymbol{K}_d = \boldsymbol{A}_1 + \boldsymbol{A}_3 = 2\omega\zeta \\ \boldsymbol{K}_p = (\boldsymbol{A}_1\boldsymbol{A}_3 + \boldsymbol{1}) = \omega^2 \end{cases} \tag{26}$$

$$\begin{bmatrix} \dot{\boldsymbol{e}} \\ \ddot{\boldsymbol{e}} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -\boldsymbol{K}_p & -\boldsymbol{K}_d \end{bmatrix} \begin{bmatrix} \boldsymbol{e} \\ \dot{\boldsymbol{e}} \end{bmatrix} \tag{27}$$

3.3 c) Analytical Backstepping with Nonlinear State Feedback

Applying Backstepping on the original system representation, the nonlinear states $[x_1 \quad x_2]$ will be used as feedback. Following the standard backstepping procedure [1], we can obtain the Lyapunov function and the control law,

$$\begin{cases} V = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\boldsymbol{z}_2^T\boldsymbol{z}_2 \\ \boldsymbol{z}_2 = \boldsymbol{x}_{2,des} - \boldsymbol{x}_2 \end{cases} \tag{28}$$

$$\begin{cases} \boldsymbol{x}_{2,des} = \boldsymbol{G}_1^{-1}(-\boldsymbol{f}_1 + \dot{\boldsymbol{x}}_{1r} + \boldsymbol{A}_1\boldsymbol{e}) \\ \boldsymbol{u} = \boldsymbol{G}_2^{-1}\left(-\boldsymbol{f}_2 + \dot{\boldsymbol{x}}_{2,des} + \boldsymbol{G}_1^T\boldsymbol{e} + \boldsymbol{A}_2\boldsymbol{z}_2\right) \end{cases} \tag{29}$$

The resulting Lyapunov function derivative and the Backstepping defined error dynamics are,

$$\dot{V} = -\boldsymbol{e}^T\boldsymbol{A}_1\boldsymbol{e} - \boldsymbol{z}_2^T\boldsymbol{A}_2\boldsymbol{z}_2 \tag{30}$$

$$\begin{cases} \dot{\boldsymbol{e}} = -\boldsymbol{A}_1\boldsymbol{e} + \boldsymbol{G}_1\boldsymbol{z}_2 \\ \dot{\boldsymbol{z}}_2 = -\boldsymbol{A}_2\boldsymbol{z}_2 - \boldsymbol{G}_1^T\boldsymbol{e} \end{cases} \tag{31}$$

It is difficult to compare the control designs if the errors are defined differently, so the tracking error dynamics is derived. From Eq. (31), we have,

$$\boldsymbol{z}_2 = \boldsymbol{G}_1^{-1}(\dot{\boldsymbol{e}} + \boldsymbol{A}_1\boldsymbol{e}) \tag{32}$$

Differentiating the first error equation in Eq. (31)

$$\ddot{\boldsymbol{e}} = -\boldsymbol{A}_1\dot{\boldsymbol{e}} + \dot{\boldsymbol{G}}_1\boldsymbol{z}_2 + \boldsymbol{G}_1\dot{\boldsymbol{z}}_2 \tag{33}$$

Inserting $\dot{\boldsymbol{z}}_{\boldsymbol{2}}$ into Eq. (33),

$$\ddot{\boldsymbol{e}} = -\boldsymbol{A_1}\dot{\boldsymbol{e}} + \dot{\boldsymbol{G}}_{\boldsymbol{1}}\boldsymbol{z_2} + \boldsymbol{G_1}(-\boldsymbol{A}_2\boldsymbol{z}_2 - \boldsymbol{G}_1^T\boldsymbol{e}) \tag{34}$$

Substituting $\boldsymbol{z_2}$ from Eq. (32),

$$\ddot{\boldsymbol{e}} = -\boldsymbol{A_1}\dot{\boldsymbol{e}} + \dot{\boldsymbol{G}}_{\boldsymbol{1}}\boldsymbol{G}_{\boldsymbol{1}}^{-\boldsymbol{1}}(\dot{\boldsymbol{e}} + \boldsymbol{A_1}\boldsymbol{e}) + \boldsymbol{G_1}\big(-\boldsymbol{A}_2\boldsymbol{G}_{\boldsymbol{1}}^{-\boldsymbol{1}}(\dot{\boldsymbol{e}} + \boldsymbol{A_1}\boldsymbol{e}) - \boldsymbol{G}_1^T\boldsymbol{e}\big) \tag{35}$$

Re-organize the above equation,

$$\ddot{\boldsymbol{e}} = \big(-\boldsymbol{A_1} - \boldsymbol{A_2} + \dot{\boldsymbol{G}}_{\boldsymbol{1}}\boldsymbol{G}_{\boldsymbol{1}}^{-\boldsymbol{1}}\big)\dot{\boldsymbol{e}} + \big(-\boldsymbol{A_2}\boldsymbol{A_1} + \dot{\boldsymbol{G}}_{\boldsymbol{1}}\boldsymbol{G}_{\boldsymbol{1}}^{-\boldsymbol{1}}\boldsymbol{A_1} - \boldsymbol{G_1}\boldsymbol{G}_{\boldsymbol{1}}^{\boldsymbol{T}}\big)\boldsymbol{e} \tag{36}$$

$$\begin{bmatrix}\dot{\boldsymbol{e}}\\ \ddot{\boldsymbol{e}}\end{bmatrix} = \begin{bmatrix}0 & I\\ -\boldsymbol{A}_1\boldsymbol{A}_2 + \dot{\boldsymbol{G}}_1\boldsymbol{G}_1^{-1}\boldsymbol{A}_1 - \boldsymbol{G}_1\boldsymbol{G}_1^T & -\boldsymbol{A}_1 - \boldsymbol{A}_2 + \dot{\boldsymbol{G}}_1\boldsymbol{G}_1^{-1}\end{bmatrix}\begin{bmatrix}\boldsymbol{e}\\ \dot{\boldsymbol{e}}\end{bmatrix} \tag{37}$$

Though in Eq. (30) the Lyapunov function derivative is made negative semi-definite, the control law doesn't cancel the plant nonlinearities completely. The nonlinear function $\boldsymbol{G}_1$ and its derivative still appear in the tracking error dynamic equations. Hence the tracking error convergence behavior does not purely depend on the feedback gains, but also on the function $\boldsymbol{G}_1$. So overshoot or over damped situation may still occur even optimal gains are assigned as in Eq. (26).

The performance of this backstepping approach can vary dramatically depending on the magnitude and dynamics of the function $\boldsymbol{G}_1$. For the presented attitude dynamics, the function $\boldsymbol{G}_1$ as in Eq. (11) has small contribution in the error dynamics: 1. the mean determinate of $\boldsymbol{G}_1$ is one, which is the strap-down matrix and is rather small compared with the feedback gains; 2. It is a function of the Euler angle so it has slower dynamics compared with the rate dynamics and thus its derivative is also relative small. There are state space models where $\boldsymbol{G}_1$ has a significant magnitude and could ruin the performance or even destabilize the closed-loop. For example, in the longitudinal dynamics when the load factor is used as outer state and pitch rate as the inner state, the function $\boldsymbol{G}_1$ is of a magnitude of 10. In the error dynamic equation, the term $\boldsymbol{G}_1\boldsymbol{G}_1^T$ is of a magnitude of 100.

3.4 d) Cascaded NDI Design

Cascaded structure design is simple and straightforward, but TSS of 3-5 times is required between the cascaded loops. The control design is applied on the transformed system. The outer loop control is straightforward,

$$\boldsymbol{x}_{\boldsymbol{3},\boldsymbol{des}} = \dot{\boldsymbol{x}}_{1r} + \boldsymbol{K}_1\boldsymbol{e} \tag{38}$$

Then a low pass filter, named inner loop reference model, is used on the outer loop output $\boldsymbol{x_{3,des}}$ to generate smooth inner loop command,

$$\dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} = \boldsymbol{K}_f \left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \tag{39}$$

Lastly the inner loop control law can be designed,

$$\begin{cases} \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{K}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{40}$$

The control law is manipulated to have similar form as the previous control designs,

$$\boldsymbol{v} = \boldsymbol{K}_f \left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) + \boldsymbol{K}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \tag{41}$$

$$= \left(\boldsymbol{K}_f - \boldsymbol{K}_2\right)\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) + \boldsymbol{K}_2\left(\boldsymbol{x_{3,des}} - \boldsymbol{x_3}\right) \tag{42}$$

$$= \left(\boldsymbol{K}_f - \boldsymbol{K}_2\right)\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) + \boldsymbol{K}_2(\dot{\boldsymbol{e}} + \boldsymbol{K}_1\boldsymbol{e}) \tag{43}$$

Tracking error dynamics can be derived,

$$\ddot{\boldsymbol{e}} + \boldsymbol{K}_2\dot{\boldsymbol{e}} + \boldsymbol{K}_1\boldsymbol{K}_2\boldsymbol{e} = \ddot{\boldsymbol{x}}_{1r} - \left(\boldsymbol{K}_f - \boldsymbol{K}_2\right)\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \tag{44}$$

So the tracking error dynamics have additional term compared with the first two designs a)&b), the differential equation is not homogenous any more. Performance degradation is expected.

The bandwidth of the inner loop reference model and the error dynamics can be the same, i.e. $\boldsymbol{K}_f = \boldsymbol{K}_2$. In that case, we have

$$\boldsymbol{v} = \boldsymbol{K}_2(\dot{\boldsymbol{e}} + \boldsymbol{K}_1\boldsymbol{e}) \tag{45}$$

$$\ddot{\boldsymbol{e}} + \boldsymbol{K}_2\dot{\boldsymbol{e}} + \boldsymbol{K}_1\boldsymbol{K}_2\boldsymbol{e} = \ddot{\boldsymbol{x}}_{1r} \tag{46}$$

The feedback gains can be calculated using the natural frequency and relative damping for this second order system,

$$\boldsymbol{K}_2 = 2\omega\zeta;\ \boldsymbol{K}_1 = \frac{\omega}{2\zeta} \tag{47}$$

The TSS is normally set to 3-5 times. This requirement can be directly seen from the inner loop/outer loop gain ratio $\frac{\boldsymbol{K}_2}{\boldsymbol{K}_1} = 4\zeta$, for instance, with critical damping of 1, the TSS is 4 times.

As we have a linear outer loop, the resulting equation is not so complicated. If the outer loop is nonlinear the error dynamics will have more residual terms. However, the concept of TSS is the same for both cases.

3.5 e) Backstepping with Cascaded Structure

Appling the Backstepping with cascaded structure on the same system representation, exactly the same control law as the cascaded NDI design can be derived, given the same feedback gains. Hence the derivation is omitted here.

3.6 f) Backstepping with Command Filter

The same system representation is also used for this design.

Backstepping with command filter use the same derivation steps as in design b), however, the virtual control derivative $\boldsymbol{x_{3,des}}$ is not calculated analytically but estimated as $\hat{\boldsymbol{x}}_{\boldsymbol{3,des}}$ by a command filter.

$$\begin{cases} \boldsymbol{x_{3,des}} = \dot{\boldsymbol{x}}_{1r} + \boldsymbol{A}_1\boldsymbol{e} \\ \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} = \boldsymbol{A}_f\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \\ \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{e} + \boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{48}$$

Due to the estimation error of the filter, the Lyapunov function is violated. With the above control law, the Lyapunov function derivative is not negative semi-definite any more,

$$\begin{cases} V = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right)^T\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \\ \dot{V} = -\boldsymbol{e}^T\boldsymbol{A}_1\boldsymbol{e} - \left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right)^T\boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) + \boldsymbol{e}^T\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \end{cases} \tag{49}$$

Similar tracking error dynamics can be derived and they are almost the same as the design d) and e),

$$\ddot{\boldsymbol{e}} + \boldsymbol{A}_2\dot{\boldsymbol{e}} + (\boldsymbol{A}_1\boldsymbol{A}_2 + 1)\boldsymbol{e} = \ddot{\boldsymbol{x}}_{1r} - \left(\boldsymbol{A}_f - \boldsymbol{A}_2\right)\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \tag{50}$$

We can see the error dynamics can be exactly the same as in the cascaded designs d) and f), given the same feedback gains and filter gain

$$\begin{cases} \boldsymbol{A_2 = K_2} \\ \boldsymbol{A_1 A_2 + 1 = K_1 K_2} \\ \boldsymbol{A_f = K_f} \end{cases} \tag{51}$$

To better analysis the effect of filter bandwidth, the error dynamics are written in Laplace domain as below.

$$\boldsymbol{x_{3,des} - \hat{x}_{3,des}} = \frac{s}{s+\boldsymbol{K_f}} \boldsymbol{x_{3,des}} = \frac{s}{s+\boldsymbol{K_f}} (s\boldsymbol{x_{1r}} + \boldsymbol{K_1 e}) \tag{52}$$

$$(s^2 + \boldsymbol{A_2} s + \boldsymbol{A_1 A_2} + 1)\boldsymbol{e} = s^2 \boldsymbol{x_{1r}} - \frac{\boldsymbol{A_f - A_2}}{s+\boldsymbol{A_f}} (s^2 \boldsymbol{x_{1r}} + s\boldsymbol{A_1 e}) \tag{53}$$

$$[s^2 + (\boldsymbol{A_1} + \boldsymbol{A_2})s + \boldsymbol{A_1 A_2} + 1]\boldsymbol{e} = \frac{s+\boldsymbol{A_2}}{s+\boldsymbol{A_f}} (s^2 \boldsymbol{x_{1r}} + s\boldsymbol{A_1 e}) \tag{54}$$

If the filter bandwidth is the same as the intermediate feedback, i.e.$\boldsymbol{A_f = A_2}$, the tracking error dynamics become the same as the cascaded control designs d) and e). If the filter bandwidth is extremely large, the error dynamics can be rendered to be the same as in design a) and b). However, the filter bandwidth is limited by the sensor noise and actuator dynamics. Hence the performance of this control design is expected to be similar with designs d) and e). the TSS is needed and control bandwidth is compromised.

3.7 g) Command Filtered Backstepping

In design f), the Lyapunov function derivative can only be rendered negative semi-definite by neglecting the estimation error, i.e. $\boldsymbol{x_{3,des}} \approx \boldsymbol{\hat{x}_{3,des}}$. However, depending on the filter bandwidth, there is always an unachieved portion of $\boldsymbol{x_{3,des}}$. Command filtered Backstepping [12] introduces a modification trying to compensate the unachieved portion.

A new Lyapunov function candidate is defined as

$$\begin{cases} V = \frac{1}{2}(\boldsymbol{e} + \boldsymbol{\xi})^T (\boldsymbol{e} + \boldsymbol{\xi}) + \frac{1}{2}\left(\boldsymbol{\hat{x}_{3,des}} - \boldsymbol{x_3}\right)^T \left(\boldsymbol{\hat{x}_{3,des}} - \boldsymbol{x_3}\right) \\ \dot{\boldsymbol{\xi}} = -\boldsymbol{A_1 \xi} - \left(\boldsymbol{x_{3,des}} - \boldsymbol{\hat{x}_{3,des}}\right) \end{cases} \tag{55}$$

In the control law, the compensated tracking error provide one additional term $\boldsymbol{\xi}$ in the final control law compared with design f).

$$\begin{cases} \boldsymbol{x_{3,des}} = \dot{\boldsymbol{x}}_{1r} + \boldsymbol{A}_1\boldsymbol{e} \\ \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} = \boldsymbol{A}_f\left(\boldsymbol{x_{3,des}} - \hat{\boldsymbol{x}}_{\boldsymbol{3,des}}\right) \\ \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{e} + \boldsymbol{\xi} + \boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \\ \boldsymbol{u} = \boldsymbol{G}_3^{-1}(-\boldsymbol{f}_3 + \boldsymbol{v}) \end{cases} \tag{56}$$

By the above modification, the Lyapunov function derivative can be rendered negative semi-definite,

$$\dot{V} = -(\boldsymbol{e} + \boldsymbol{\xi})^T\boldsymbol{A}_1(\boldsymbol{e} + \boldsymbol{\xi}) - \left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right)^T\boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \tag{57}$$

The Backstepping defined error dynamics is modified accordingly,

$$\begin{cases} \left(\dot{\boldsymbol{e}} + \dot{\boldsymbol{\xi}}\right) = -A_1(\boldsymbol{e} + \boldsymbol{\xi}) + \left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) \\ \left(\dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} - \dot{\boldsymbol{x}}_{\boldsymbol{3}}\right) = -\boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) - (\boldsymbol{e} + \boldsymbol{\xi}) \end{cases} \tag{58}$$

To derive the tracking error dynamics, we can differentiate the first equation,

$$\ddot{\boldsymbol{e}} + \ddot{\boldsymbol{\xi}} = -A_1\left(\dot{\boldsymbol{e}} + \dot{\boldsymbol{\xi}}\right) + \left(\dot{\boldsymbol{x}}_{\boldsymbol{3f}} - \dot{\boldsymbol{x}}_{\boldsymbol{3}}\right) \tag{59}$$

Replacing the $\left(\dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} - \dot{\boldsymbol{x}}_{\boldsymbol{3}}\right)$ and $\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right)$ terms,

$$\ddot{\boldsymbol{e}} + \ddot{\boldsymbol{\xi}} = -A_1\left(\dot{\boldsymbol{e}} + \dot{\boldsymbol{\xi}}\right) - \boldsymbol{A}_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}\right) - (\boldsymbol{e} + \boldsymbol{\xi}) \tag{60}$$

$$= -A_1\left(\dot{\boldsymbol{e}} + \dot{\boldsymbol{\xi}}\right) - \boldsymbol{A}_2\left[\left(\dot{\boldsymbol{e}} + \dot{\boldsymbol{\xi}}\right) + A_1(\boldsymbol{e} + \boldsymbol{\xi})\right] - (\boldsymbol{e} + \boldsymbol{\xi}) \tag{61}$$

Reorganize it,

$$\ddot{\boldsymbol{e}} + (\boldsymbol{A}_1 + \boldsymbol{A}_2)\dot{\boldsymbol{e}} + (\boldsymbol{A}_1\boldsymbol{A}_2 + \boldsymbol{1})\boldsymbol{e} = \ddot{\boldsymbol{\xi}} + (\boldsymbol{A}_1 + \boldsymbol{A}_2)\dot{\boldsymbol{\xi}} + (\boldsymbol{A}_1\boldsymbol{A}_2 + \boldsymbol{1})\boldsymbol{\xi} \tag{62}$$

In Laplace domain,

$$[\mathrm{s}^2 + (\boldsymbol{A}_1 + \boldsymbol{A}_2)\boldsymbol{s} + (\boldsymbol{A}_1\boldsymbol{A}_2 + \boldsymbol{1})](\boldsymbol{e} - \boldsymbol{\xi}) = 0 \tag{63}$$

The tracking error dynamics is not as good as in design a) and b), but it gives a homogenous form. The effect of $\boldsymbol{\xi}$ is analyzed in the next section

4. Comparison in Simulation and Experiments

As all the first three designs use analytical solutions, they can be analytically compared together. The first two design a) and b) gives the perfect linear error dy-

namics, which should result the best performance. Design c) gives state dependent error dynamics; the performance of the tracking error $\boldsymbol{e}$ depends on the nonlinear function $\boldsymbol{G_1}$ hence not consistent, esp. when the $\boldsymbol{G_1 G_1^T}$ or $\dot{\boldsymbol{G}}_1 \boldsymbol{G}_1^{-1}$ are comparatively large.

$$\begin{cases} a)\&b): \ddot{\boldsymbol{e}} + \quad \boldsymbol{K}_d \dot{\boldsymbol{e}} + \quad \boldsymbol{K}_p \boldsymbol{e} = \boldsymbol{0} \\ c): \quad \ddot{\boldsymbol{e}} + (\boldsymbol{A}_1 + \boldsymbol{A}_2)\dot{\boldsymbol{e}} + \boldsymbol{A}_1 \boldsymbol{A}_2 \boldsymbol{e} = \dot{\boldsymbol{G}}_1 \boldsymbol{G}_1^{-1}(\dot{\boldsymbol{e}} + \boldsymbol{A}_1 \boldsymbol{e}) - \boldsymbol{G}_1 \boldsymbol{G}_1^T \boldsymbol{e} \end{cases} \tag{64}$$

The last four designs all have low pass filter in the control law design, i.e. additional dynamics is introduced, so analytically comparisons are not so straightforward. Nevertheless, there are still a few obvious observations: 1) design d) & e) give the same control law; 2) The low pass filter and the output state control are the same for all designs; 3) There is only one term difference between the designs, as shown in the following equation,

$$\begin{cases} d)\&e): \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{A}_2(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}) \\ f): \quad \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{e} + \boldsymbol{A}_2(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}) \\ g): \quad \boldsymbol{v} = \dot{\hat{\boldsymbol{x}}}_{\boldsymbol{3,des}} + \boldsymbol{e} + \boldsymbol{\xi} + \boldsymbol{A}_2(\hat{\boldsymbol{x}}_{\boldsymbol{3,des}} - \boldsymbol{x_3}) \end{cases} \tag{65}$$

In order to clearly observe the differences of the control designs, the sensor noise is turned off and only actuator dynamics is considered in the simulation. The same reference signals are used for all designs, i.e. the reference signal demand the same control bandwidth. Second order low pass filter is used to generate smooth reference command with relative damping of $\xi = 1$ and natural frequency of $\omega = 10$. The control designs with larger control bandwidth should have better performance in terms of overshoot and rise time. So the performance differences can be directly compared.

For design d)-g), which use low pass filter in the design, two set of gains has been used, I. assume no TSS needed, and II provide 4 times TSS. The control gains for all designs are listed as follows,

Actuator Dynamics	$G_A(s) = \frac{40}{s+40}$
Design a) & b)	$K_d = 2\omega = 30; K_p = \omega^2 = 225$
Design c)	$A_1 = 15; A_2 = 15$
Design d)-g) – I	$A_1 = 15; A_2 = 15; A_f = K_f = 40$
Design d)-g) – II	$A_1 = 7.5; A_2 = 30; A_f = K_f = 40$

Table 1, gain design for the simulations

The simulated tracking performance for a step command is shown in the Fig. 1 and Fig. 2. It can be seen that design a), b), and c) gives almost the same best performance; while design d)-g) gives worse performance as expected. The design c) in this case is only a bit worse than design a), b). The differences are rather small, but design c) performance can get worse if $\boldsymbol{G_1 G_1^T}$ and $\dot{\boldsymbol{G}}_1 \boldsymbol{G}_1^{-1}$ get bigger.

What is also interesting is that the additional terms in design f) and g) didn't contribute much, and their performance is almost the same as the cascaded designs d) and e). Their step responses almost coincide in the same curve in Fig. 1&2. The common feature of the four designs is the low pass filter used in the control structure. Its bandwidth is limited by the sensor noise, so it introduce additional phase lag in the system, which slow down the control action. Hence performance is degraded and TSS is needed between the outer and inner loop.

Further simulation is performed on design d)-g) to analyze the TSS. In simulation I, the gains are kept the same as in design a), i.e. no TSS between the intermediate state and output state. And in simulation II, the TSS concept is introduced by increasing the intermediate state feedback gain and decreasing the output state feedback gain. The performance of design d)-g) get improved by considering the TSS, though they are still worse than the design a) and b).

The Asctec 'Hummingbird' [16] is used as experiment platform. An AHRS Kalman filter is used for the attitude estimation. Both data fusion and control loop are running with 1k Hz. The experimental results, shown in Fig. 3, give similar results as the simulations. Non-cascaded designs a) –c) gives very good tracking performance. The last four designs have a bit worse performance but they are rather similar with each other.

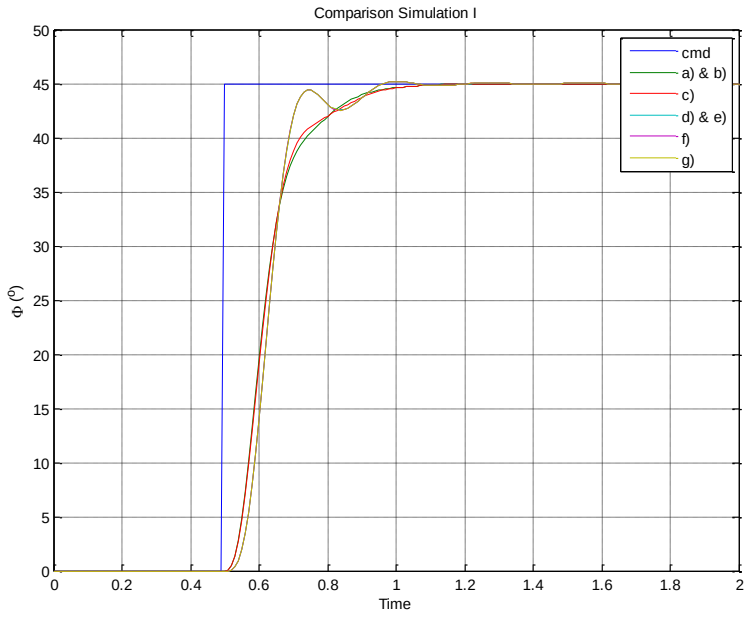

Fig. 1a. Simulation performance for step commands of all seven control designs. Design d)-g) assume no TSS needed

16

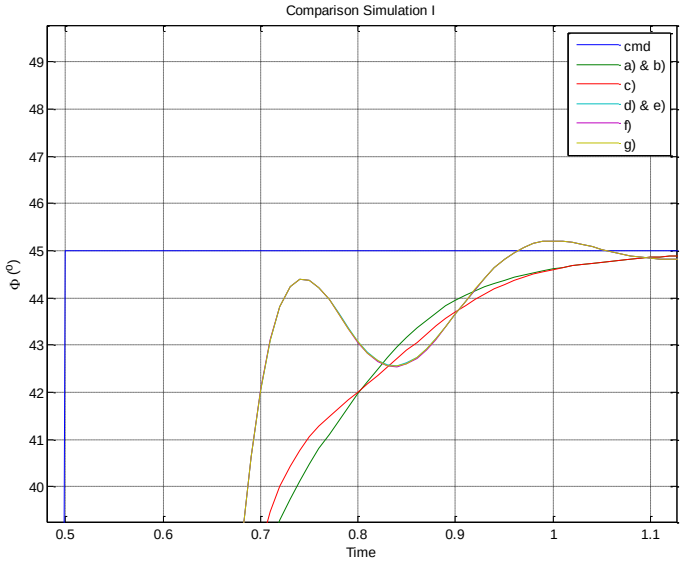

Fig. 1b. Close in view of the Figure 1a.

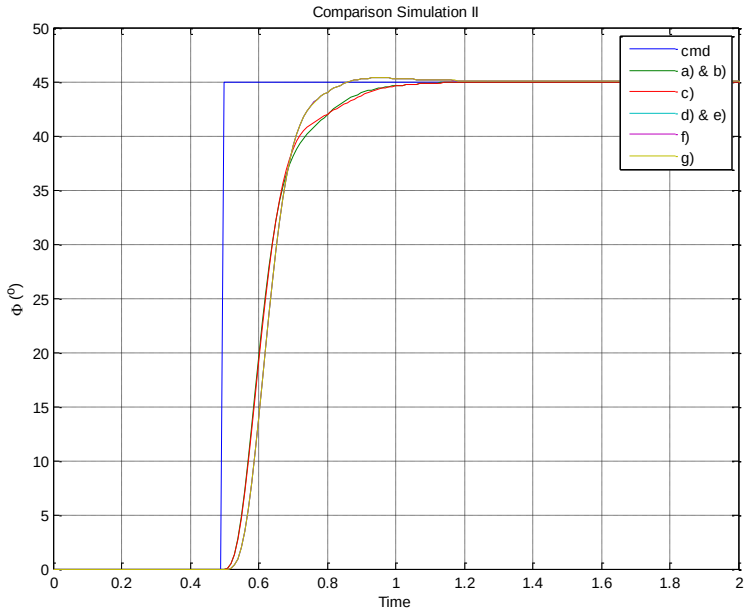

Fig. 2. Simulation performance for step commands of all seven control designs. Design d)-g) provide 4 times scale separation between inner/outer loops

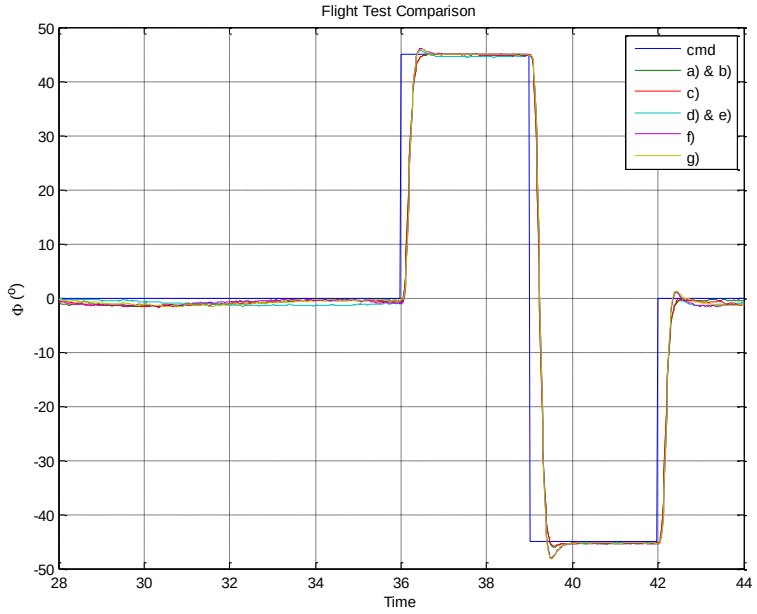

Fig. 3a. Experimental results of the seven designs

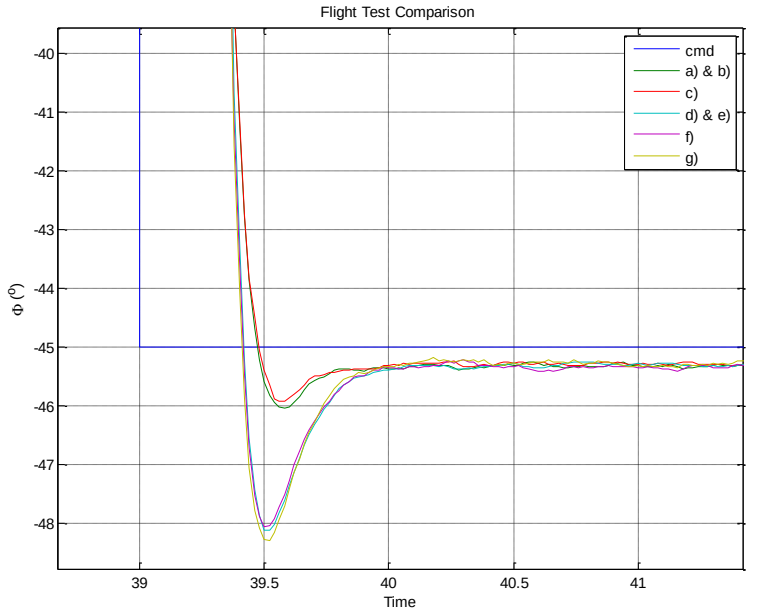

Fig. 3b. Close in view of the Figure 2a.

5. Conclusion

NDI and Backstepping could result in the same control law and give the best control performance, like in design a) and b). Both are non-cascaded design, they completely cancel the plant nonlinearities and result in perfect linear tracking error dynamics. Other non-cascaded Backstepping design as in design c) results a little bit worse performance. The tracking error dynamics is nonlinear and depend on the state dependent matrix $\boldsymbol{G_1}$ and its derivative.

The last four designs can be all categorized as cascaded designs due to the usage of the low pass filter. It is worth noting that the Backstepping with command filter and the command filter Backstepping designs appears like non-cascaded design, but they give no observable advantage over the cascaded designs. They result very similar control laws and thus similar performance. As expected, their performance is worse compared with designs a) and b) as TSS is needed.

It is clear for system of strict-feedback form, non-cascaded structure design as in design a) and b) are the best choices in terms of control bandwidth. For system of other forms, if non-cascaded structure design is not possible, design d) is the best choices as it is the simplest among the last four designs yet provides the same performance. However, with proper relaxations, it is possible to extend them into strict-feedback form, then design a) and b) can be applied on the modified system. The position control of quadrotor is one of the examples. In the companion paper [17], such design is derived.

6. Acknowledgments

The Author gratefully acknowledges the support of the Technische Universität München – International Graduate School of Science and Engineering (IGSSE) and Institute for Advanced Study, funded by the German Excellence Initiative, project 4.03 Image Aided Flight Control.

Reference

1. Krstic M., Kanellakopoulos I., Kokotovic P. (1995) Nonlinear and adaptive control design, vol. 222. Wiley New York
2. Khalil H. (2002) Feedback linearization in: Nonlinear Systems, 3rd ed. Prentice Hall
3. Klose S., Wang, J. et al (2010) Markerless Vision Assisted Flight Control of a Quadrocopter. IEEE 2010, RSJ International Conference on Intelligent Robots and Systems
4. Wang J., Bierling T. et al (2011) Novel Dynamic Inversion Architecture Design for Quadrocopter Control. 'Advances in Aerospace Guidance, Navigation and Control', F. Holzapfel and S. Theil, Eds. Springer Berlin Heidelberg, pp. 261-272
5. Voos, H. (2009) Nonlinear Control of a Quadrotor Micro-UAV using Feedback-Linearization, IEEE International Conference on Mechatronics

6. Fritsch O., Monte P. De, et al (2012) Quasi-static feedback linearization for the translational dynamics of a quadrotor helicopter, American Control Conference
7. Mistler V., Benallegue A., M'Sirdi N. K. (2001) Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," Proceedings 10th IEEE International Workshop on Robot and Human Interactive Communication. ROMAN (Cat. No.01TH8591), pp. 586-593
8. Wang J., Raffler T., and Holzapfel F. (2012) Nonlinear Position Control Approaches for Quadcopters Using a Novel State Representation. AIAA Guidance, Navigation and Control Conference, Minneapolis, 2012
9. Bouabdallah S. and Siegwart R. (2005) Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor, IEEE International Conference on Robotics and Automation, pp. 2259-2264
10. Pollini L. and Metrangolo A. (2008) Simulation and robust Backstepping control of a quadrotor aircraft, in Proc. of the AIAA Modeling and Simulation technologies Conference and Exhibit
11. Madani T. and Benallegue A. (2006) Backstepping control for a quadrotor helicopter, in Proc. of the IEEE International Conference on Intelligent Robots and Systems
12. Farrell J. A., Polycarpou M. et al (2008) Command Filtered Backstepping, at American Control Conference
13. Thunberg J. and Robinson J. W. C. (2008) Block Backstepping, NDI and related cascade designs for efficient development of nonlinear flight control laws, AIAA Guidance, Navigation and Control Conference and Exhibit
14. Jakubczyk B. and Respondek W. (1980), On Linearization of Control Systems, Bull, Acad. Sci. Polonaise Ser. Sci. Math. 28, 517-522.
15. Nijmeijer H. and Van der Scaft A. J. (1990), Nonlienar Dynamical Control Systems, Springer, New York
16. Ascending technologies GmbH, Hummingbird Quadrotor. http://www.asctec.de. Accessed 27 July 2010
17. Wang, J., Holzapfel F. et al (2013) Non-cascaded Dynamic Inversion Design for Quadrotor Position Control with L1 Augmentation. CEAS EuroGNC, Delft