ENOC-2008, Saint Petersburg, Russia, June, 30–July, 4 2008

ADAPTIVE TRACKING CONTROL FOR A QUAD-ROTOR

J. Cesáreo Raimúndez
Depto. Sistemas e Automática
Universidade de Vigo
España
cesareo@uvigo.es

Alejandro Fernández Villaverde
Depto. Sistemas e Automática
Universidade de Vigo
España
afernandez@uvigo.es

Abstract
This paper illustrates the application of an adaptive flight control architecture to a scale quad-rotor. For autonomous VTOL (Vertical Takeoff and Landing) flight, it is common to separate the control problem into an inner fast loop that controls attitude and an outer slow loop that controls the trajectory of the VTOL. In this paper we augment a conventional PD controller conceived mainly for hovering, with an adaptive element using a real-time tuning single hidden layer neural network in a inner-outer loop combined architecture to account for model inversion error cancellation, issued in the feedback linearization process. The results shown in simulations reveal the superior performance of the augmented controller in tracking maneuvers.

Key words
Quad - Rotors, Neural Networks, Tracking Control, Adaptive Control

1 Introduction

The potential for unmanned aerial vehicles (UAVs) in applications such as environmental monitoring or fire prevention has been well established. A quad-rotor is an underactuated, dynamic system with four input forces and six output coordinates. Its actuators are four fixed pitch angle rotors. This configuration increases payload capacity and maneuverability. The basic motions of a quad-rotor are generated by varying the rotor speeds of all four motors, thus changing the lifting forces. The quad-rotor tilts toward the direction of the slow spinning rotor, which enables acceleration along that direction. The spinning directions of the rotors are set to balance the moments, therefore eliminating the need for a tail rotor. Quad-rotors, as any other UAV's, are affected by aerodynamic forces in strong non-linear coupling, which can be considered uncertain, and also by external disturbances such as wind gusts. In order to conveniently control the quad-rotor it is required to meet the essential stability, robustness and desired dynamic performance, being able to adapt to changing parameters and environmental unmodeled disturbances. In this paper, a tracking controller is designed for the nonlinear quad-rotor model. In a first stage, the controller consists of two linear proportional plus derivative (PDs) controllers in an inner-outer loop configuration, assuring an ideal tracking capability without external perturbations. The resulting closed loop system is highly sensitive to perturbations, so the initial linear controller is augmented by an adaptive action, introduced by a single hidden layer (SHL) feed forward neural network (NN) acting also in an inner-outer additive arrangement, regarding the linear control. The performance of the augmented system is greatly improved, being capable of adapting to external unmodeled perturbations or even to internal unmodeled dynamics [Salazar, Palomino, Lozano, 2005]. The structure of this paper is as follows: section 2 presents some basic ideas on approximate feedback linearization [Nakwan, 2003; Nakwan, Calise, 2007]. Section 3 presents the quad-rotor modeling according to a Lagrangian formalism [Castillo, Lozano, Dzul, 2005]. Tracking formulation is presented in section 4. In section 5 a case study with simulation results is presented and finally, in section 6 the conclusions are presented.

2 Approximate System Linearization

One common method for controlling nonlinear dynamical systems is based on approximate feedback linearization [Isidori, 1995], which depends on the relative degree of each controlled variable. For newtonian systems like the quad-rotor in a simplified approach, the regulated variables of interest, here represented as the vector q, have relative degree two. The control variables are represented by the vector u.

$$\ddot{q} = f(q, \dot{q}, u) \tag{2.1}$$

A pseudo control ν is defined such that the dynamic relation between it and the system state is linear.

$$\ddot{q} = \nu \tag{2.2}$$

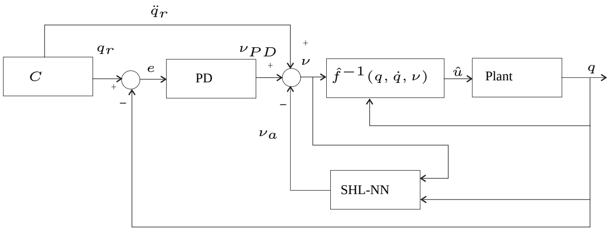

Figure 1. NN augmented adaptive control architecture

where

$$\nu = f(q, \dot{q}, u) \tag{2.3}$$

Since the function $f(q, \dot{q}, u)$ is not exactly known, an approximation is used which is invertible regarding u

$$\nu = \hat{f}(q, \dot{q}, u) \tag{2.4}$$

resulting in

$$\ddot{q} = \nu + \Delta(q, \dot{q}, u) \tag{2.5}$$

where the modeling error is represented by

$$\Delta(q, \dot{q}, u) = f(q, \dot{q}, u) - \hat{f}(q, \dot{q}, u) \tag{2.6}$$

So the effective actuator displacement can be calculated as

$$\hat{u} = \hat{f}^{-1}(q, \dot{q}, \nu) \tag{2.7}$$

Supposing in 2.5 that $\Delta(q, \dot{q}, u) = 0$ we can proceed in the stabilization problem, choosing a linear controller, a PD for instance, that will locally solve the regulation problem. A SHL neural network with conveniently adapted weights will be responsible for modeling error cancellation. Including a command path generator C, the former linear controller can be augmented through the architecture depicted in fig. 1. The pseudo control signal in 2.5 is the sum of three components

$$\nu = \ddot{q}_r + \nu_{PD} - \nu_a \tag{2.8}$$

where $\ddot{q}_r$ is generated by C, ν_{PD} is generated by the PD controller and ν_a is generated by the adaptive element introduced to compensate for the model inversion error. The tracking error is computed as

$$e = \begin{bmatrix} q_r - q \\ \dot{q}_r - \dot{q} \end{bmatrix} \tag{2.9}$$

and the PD controller can be represented by

$$\nu_{PD} = \begin{bmatrix} K_p \; K_d \end{bmatrix} e \tag{2.10}$$

2.1 Adaptive Element

The adaptive element is implemented by a SHL-NN with conveniently tuned weights V, W such that

$$\nu_a = W^\top \bar{\sigma}(V^\top \bar{q}) \tag{2.11}$$

with $\bar{q} = [\nu, q]$. Given a sufficient number of hidden layer neurons and appropriate inputs, it should be possible to train a SHL-NN [Hornik, Stinchcombe, White, 1989] on line to cancel the effect of Δ. The weight matrices are

$$\begin{aligned} V &= \begin{pmatrix} v_{0,1} & v_{0,2} & \cdots & v_{0,n_2} \\ v_{1,1} & v_{1,2} & \cdots & v_{1,n_2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n_1,1} & v_{n_1,2} & \cdots & v_{n_1,n_2} \end{pmatrix} \\ W &= \begin{pmatrix} w_{0,1} & w_{0,2} & \cdots & w_{0,n_3} \\ w_{1,1} & w_{1,2} & \cdots & w_{1,n_3} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n_2,1} & w_{n_2,2} & \cdots & w_{n_2,n_3} \end{pmatrix} \end{aligned} \tag{2.12}$$

Here n_1, n_2, n_3 are the number of inputs, hidden layer nodes and outputs. Also $\bar{\sigma}(\xi) = (1, \sigma(\xi_1), \cdots, \sigma(\xi_{n_1}))^\top$. The scalar function σ is the sigmoidal activation function

$$\sigma(\xi) = \frac{1}{1 + e^{-\alpha\xi}} \tag{2.13}$$

2.2 Contractibility

The transformation 2.11 must be contractive regarding ν_a. Note that Δ depends on ν_a through ν, whereas ν_a has to be designed to cancel Δ. Hence the existence and uniqueness of a fixed point solution for $\nu_a = \Delta(q, \dot{q}, \nu_a)$ must be assumed. A sufficient condition is to ascertain that the map $\nu_a \rightarrow \Delta(q, \dot{q}, \nu_a)$ is a contraction over the entire input domain of interest, or $\|\partial\Delta/\partial\nu_a\| < 1$. This condition is equivalent to [Nakwan, 2003].

$$0 < \frac{1}{2}\left|\frac{\partial f}{\partial u}\right| < \left|\frac{\partial \hat{f}}{\partial u}\right| < \infty \tag{2.14}$$

2.3 Tracking Error Boundedness

The tracking error dynamics is given by

$$\dot{e} = Ae + B(\nu_a - \Delta) \tag{2.15}$$

with

$$A = \begin{bmatrix} O & I \\ -K_p & -K_d \end{bmatrix}, \quad B = \begin{bmatrix} O \\ I \end{bmatrix} \tag{2.16}$$

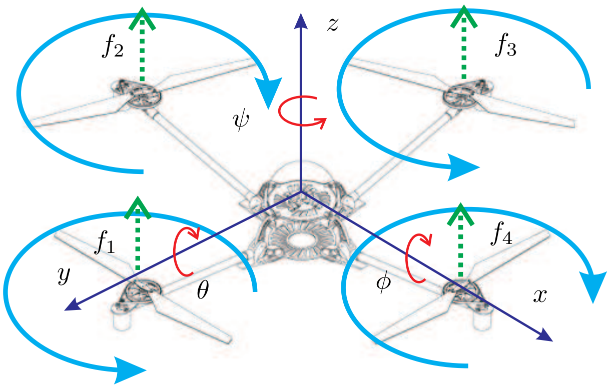

Figure 2. quadrotor representation

where I and O are a suitable identity and null matrices respectively. Consider the system 2.1, the inverse law 2.7 and the contractibility property, as well as the adaptation laws

$$\begin{aligned}\dot{W} &= -[(\bar{\sigma} - \bar{\sigma}' V^\top \bar{\eta}) r^\top + \kappa \|e\| W]\Gamma_W \\ \dot{V} &= -\Gamma_V [\bar{\eta}(r^\top W^\top \bar{\sigma}') + \kappa \|e\| V]\end{aligned} \tag{2.17}$$

where

$$\bar{\sigma}'(\hat{z}) \equiv \frac{\partial \bar{\sigma}(z)}{\partial z}\bigg|_{z=\hat{z}} \tag{2.18}$$

is the Jacobian matrix and $r = e^\top PB$. Also $P \succ 0$ is the unique positive definite solution for the Lyapunov equation

$$A^\top P + PA + Q = 0 \tag{2.19}$$

for any convenient $Q \succ 0$. A and B are defined in 2.16. Given 2.17 with $\Gamma_W \succ 0$, $\Gamma_V \succ 0$ and $\kappa > 0$, according to [Nardi, 2000; Yoonghyun, 2005] the tracking error e uniform boundedness is assured.

3 Simplified Modeling - Lagrangian Formulation

The generalized coordinates for the quad-rotor are $q = (\xi, \eta)$ where $\xi = (x, y, z)$, denote the position of the center of mass concerning the inertial frame and $\eta = (\psi, \theta, \phi)$ are the three Euler angles (yaw, pitch and roll) representing the quad-rotor pose. The total quad-rotor kinetic energy is given by T and the potential energy is given by V

$$T = \frac{1}{2} m \dot{\xi}^\top \dot{\xi} + \frac{1}{2} \omega_b^\top J \omega_b,\ V = mgz,\ \omega_b = Q(\eta)\dot{\eta} \tag{3.1}$$

Here m denotes the mass of the quad-rotor. Also

$$\omega_b = R(\eta)\dot{R}^\top(\eta) = Q(\eta)\dot{\eta} \Rightarrow Q(\eta) = \begin{pmatrix} -s_\theta & 0 & 1 \\ c_\theta s_\phi & c_\phi & 0 \\ c_\theta c_\phi & -s_\phi & 0 \end{pmatrix} \tag{3.2}$$

where $R(\eta)$ is the transformation matrix representing the quad-rotor pose.

$$J = \begin{pmatrix} J_1 & J_{12} & J_{13} \\ J_{12} & J_2 & J_{23} \\ J_{13} & J_{23} & J_3 \end{pmatrix},\quad I(\eta) = Q(\eta)^\top J Q(\eta) \tag{3.3}$$

Here $I(\eta)$ is the inertia matrix regarding the inertial frame. The change from an inertial to a local frame for the quad-rotor is done according to

$$R(\eta) = \begin{pmatrix} c_\theta c_\psi & s_\theta s_\psi & -s_\theta \\ c_\psi s_\theta s_\phi - s_\psi c_\phi & s_\psi s_\theta s_\phi + c_\psi c_\phi & c_\theta s_\phi \\ c_\psi s_\theta c_\phi + s_\psi s_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi & c_\theta c_\phi \end{pmatrix} \tag{3.4}$$

where c_θ, s_θ stand for $\cos\theta, \sin\theta$, respectively. The movement equations are obtained as follows,

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\xi}}\right) - \frac{\partial L}{\partial \xi} &= f_\xi \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\eta}}\right) - \frac{\partial L}{\partial \eta} &= f_\eta\end{aligned} \tag{3.5}$$

with

$$f_\xi = R(\eta) f_b,\ f_\eta = \tau_\eta \tag{3.6}$$

The movement equations have the structure observed in 2.1

$$\begin{aligned}\ddot{\xi} &= -m^{-1} f_0 + M^{-1} R(\eta) f_b \\ \ddot{\eta} &= I^{-1}(\eta)\left(\frac{1}{2}\frac{\partial}{\partial \eta}\left(\dot{\eta}^\top I(\eta)\dot{\eta}\right) - \dot{I}(\eta)\dot{\eta} + \tau_\eta\right)\end{aligned} \tag{3.7}$$

with

$$f_b = \begin{pmatrix} 0 \\ 0 \\ u \end{pmatrix},\quad f_0 = \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} \tag{3.8}$$

and $\tau_\eta = (\tau_\psi, \tau_\theta, \tau_\phi)$ being the moments regarding the local reference frame. Those moments can be modeled in a first degree of approximation, without considering rotor dynamics, as:

$$\begin{aligned} u &= \textstyle\sum_{i=1}^4 f_i \\ f_i &= B_o \omega_i^2 \\ \tau_\psi &= (D_o/B_o)(f_2 + f_4 - f_1 - f_3) \\ \tau_\theta &= l(f_4 - f_2) \\ \tau_\phi &= l(f_3 - f_1) \end{aligned} \tag{3.9}$$

where f_i are the lifting forces in each rotor, ω_i the corresponding angular velocities, l the diagonal distance between axes of the respective rotors, and D_o, B_o are *drag* and *thrust* factors, respectively. The relationship between $\{f_b, \tau_\eta\}$ and rotations ω_i is straightforward.

$$
\begin{aligned}
\omega_1^2 &= \frac{1}{4B_oD_ol}(D_olu - 2D_o\tau_\phi - B_ol\tau_\psi)\\
\omega_2^2 &= \frac{1}{4B_oD_ol}(D_olu - 2D_o\tau_\theta + B_ol\tau_\psi)\\
\omega_3^2 &= \frac{1}{4B_oD_ol}(D_olu + 2D_o\tau_\phi - B_ol\tau_\psi)\\
\omega_4^2 &= \frac{1}{4B_oD_ol}(D_olu + 2D_o\tau_\theta + B_ol\tau_\psi)
\end{aligned}
\tag{3.10}
$$

4 Tracking Controller

Being $P_r = (\xi_r, \eta_r), P = (\xi, \eta)$ the reference and final trajectories and defining $e = P_r - P$, we will be able to establish

$$
\begin{aligned}
\ddot{e}_\xi &= \ddot{\xi}_r + m^{-1}f_0 - m^{-1}R(\eta)f_b\\
&= \nu_\xi\\
&= -k_{p\xi}e_\xi - k_{d\xi}\dot{e}_\xi\\
\ddot{e}_\eta &= \ddot{\eta}_r - I^{-1}(\eta)\left(\frac{1}{2}\frac{\partial}{\partial\eta}\left(\dot{\eta}^\top I(\eta)\dot{\eta}\right) - \dot{I}(\eta)\dot{\eta} + \tau_\eta\right)\\
&= \nu_\eta\\
&= -k_{p\eta}e_\eta - k_{d\eta}\dot{e}_\eta
\end{aligned}
\tag{4.1}
$$

with (ν_ξ, ν_η) the pseudo-control components and $k_{p\xi}, k_{p\eta}, k_{d\xi}, k_{d\eta}$ positive matrices. From

$$
\ddot{\xi}_r + m^{-1}f_0 - m^{-1}R(\eta_o)f_b = \nu_\xi \tag{4.2}
$$

follows that

$$
\begin{aligned}
\eta_o &= \begin{cases} \psi_o = \psi_r \\ \theta_o = \arcsin\left(\dfrac{\ddot{x}_r - \nu_{\xi_x}}{u}\right) \\ \phi_o = \arctan\left(\dfrac{\ddot{y}_r - \nu_{\xi_y}}{\ddot{z}_r + g - \nu_{\xi_z}}\right) \end{cases}\\
u &= m\left\|\ddot{\xi}_r + m^{-1}f_0 - \nu_\xi\right\|
\end{aligned}
\tag{4.3}
$$

Calling now

$$
\Delta\eta = \begin{pmatrix} 0 \\ \theta_r - \theta_o \\ \phi_r - \phi_o \end{pmatrix} \tag{4.4}
$$

the error on the η coordinates is corrected, resulting in

$$
\nu_\eta = -k_{p\eta}(e_\eta - \Delta\eta) - k_{d\eta}\dot{e}_\eta
$$

which defines the control law

$$
\begin{aligned}
\tau_\eta &= \dot{I}(\eta)\dot{\eta} - \frac{1}{2}\frac{\partial}{\partial\eta}\left(\dot{\eta}^\top I(\eta)\dot{\eta}\right)\\
&\quad + I(\eta)\left(\ddot{\eta}_r + k_{p\eta}(e_\eta - \Delta\eta) + k_{d\eta}\dot{e}_\eta\right)
\end{aligned}
\tag{4.5}
$$

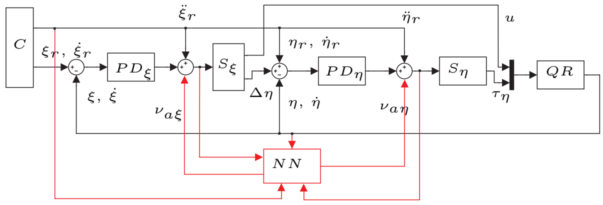

Figure 3. Augmented Linear Controller with an Adaptive SHL-NN

which will stabilize the P_r trajectory tracking with a bounded error. In figure (3) the structure of the controller is shown, consisting of two proportional-derivative terms, namely PD_ξ, PD_η where S_ξ, S_η represent the operations described in equations 4.2 and 4.5 respectively. QR represents the plant (quad-rotor) and C the generator of trajectory commands.

4.1 Adaptive Element

In order to cancel the presence of unmodeled dynamics, two corrective components are added to the control loops presented in figure (3), which are generated by the adaptive element SHL-NN = (NN_ξ, NN_η). Let $\Delta = (\Delta_\xi, \Delta_\eta)$ be the vector of modeling errors. Equations (4.1) can be written as:

$$
\begin{aligned}
\ddot{e}_\xi &= \ddot{\xi}_r - (\ddot{\xi} + \Delta_\xi)\\
\ddot{e}_\eta &= \ddot{\eta}_r - (\ddot{\eta} + \Delta_\eta)
\end{aligned}
\tag{4.6}
$$

By adding to the control effort the adaptive terms $\nu_{a\xi}, \nu_{a\eta}$ the following representation of the error dynamics is obtained:

$$
\begin{aligned}
\ddot{e}_\xi + k_{p\xi}e_\xi + k_{d\xi}\dot{e}_\xi + \nu_{a\xi} - \Delta_\xi = 0\\
\ddot{e}_\eta + k_{p\eta}e_\eta + k_{d\eta}\dot{e}_\eta + \nu_{a\eta} - \Delta_\eta = 0
\end{aligned}
\tag{4.7}
$$

which can also be written as

$$
\frac{d}{dt}\begin{pmatrix} e \\ \dot{e} \end{pmatrix} = \begin{pmatrix} O & I \\ -K_p & -K_d \end{pmatrix}\begin{pmatrix} e \\ \dot{e} \end{pmatrix} + B(\nu_a - \Delta) \tag{4.8}
$$

with

$$
\begin{aligned}
K_p &= \begin{pmatrix} k_{p\xi} & O \\ O & k_{p\eta} \end{pmatrix}, \quad K_d = \begin{pmatrix} k_{d\xi} & O \\ O & k_{d\eta} \end{pmatrix}\\
B &= \begin{pmatrix} O \\ I \end{pmatrix}, \quad \nu_a = \begin{pmatrix} \nu_{a\xi} \\ \nu_{a\eta} \end{pmatrix}, \quad \Delta = \begin{pmatrix} \Delta_\xi \\ \Delta_\eta \end{pmatrix}
\end{aligned}
\tag{4.9}
$$

and with $e = (e_\xi, e_\eta)$. Here again, O, I are suitable null and identity matrices respectively. If the SHL-NN output signal ν_a perfectly cancels Δ, then we have asymptotically stable error dynamics. ν_a has the structure

$$
\nu_a = \left(W_\xi^\top\bar{\sigma}(V_\xi^\top\bar{\xi}), W_\eta^\top\bar{\sigma}(V_\eta^\top\bar{\eta})\right) \tag{4.10}
$$

Weight propagation for $W_{\{\xi,\eta\}}, V_{\{\xi,\eta\}}$, is done according to the adaptation laws

$$\begin{aligned}\dot{W}_i &= -[(\bar{\sigma} - \bar{\sigma}' V_i^\top \bar{q}) r^\top + \kappa \|e\| W_i]\Gamma_{W_i} \\ \dot{V}_i &= -\Gamma_{V_i}[\bar{q}(r^\top W_i^\top \bar{\sigma}') + \kappa \|e\| V_i]\end{aligned} \tag{4.11}$$

with $r = (e^\top PB)^\top$, and $i = \{\xi, \eta\}$. The representation of $\bar{\sigma}(V_\xi^\top \bar{q})$ as $\bar{\sigma}$, as well as that of $\bar{\sigma}'$, is done for the sake of clarity. $\Gamma_{V_i} \succ 0$, $\Gamma_{W_i} \succ 0$ are definite positive matrices and $\kappa > 0$ is a real constant, being $\bar{q}$ the extended input vector, this is, $\bar{q} = (1, q)$ where q is the input vector.

4.2 Obtaining the Adaptation Laws

Let us consider the Lyapunov function

$$\begin{aligned}\mathfrak{V}(e, \tilde{V}, \tilde{W}) = \frac{1}{2}\big(e^\top Pe \\ + \operatorname{tr}\left(\tilde{W}^\top \Gamma_W^{-1} \tilde{W}\right) + \operatorname{tr}\left(\tilde{V}^\top \Gamma_V^{-1} \tilde{V}\right)\big)\end{aligned} \tag{4.12}$$

where P solves the equation

$$A^\top P + PA + Q = 0, \quad A = \begin{pmatrix} O & I \\ -K_p & -K_d \end{pmatrix} \tag{4.13}$$

with $-Q$ and P definite positive. In order to obtain the adaptation equations (4.11) we must follow the steps required to proof that, on the error orbits, the following condition is satisfied:

$$\dot{\mathfrak{V}} \leq 0 \tag{4.14}$$

as explained in [Johnson, Kannan, 2002]. The following steps are given in order to show the parameters regarding an adequate tuning of the controller. The details of the proof of convergence follow the above mentioned reference. Let us consider

$$\epsilon = \nu_a^* - \Delta = W^{*\top} \bar{\sigma}(V^{*\top} \bar{q}) - \Delta \tag{4.15}$$

where W^*, V^* are the optimum values that best approximate Δ. The error dynamics is

$$\dot{e} = Ae + B\left(W^{*\top}\bar{\sigma}(V^\top \bar{q}) - W^\top \bar{\sigma}(V^{*\top}\bar{q}) + \epsilon\right) \tag{4.16}$$

Defining now $\tilde{W} = W - W^*$, $\tilde{V} = V - V^*$ and using the Taylor series expansion of σ with respect to V in the neighborhood of V^*, which is the optimum value, we obtain

$$\dot{e} = Ae + B\left(\tilde{W}^\top(\sigma - \sigma' V^\top \bar{q}) + W^\top \sigma' \tilde{V}^\top \bar{q} + w\right) \tag{4.17}$$

with

$$w = \epsilon - W^{*\top}\left(\sigma^* - \sigma + \sigma' \tilde{V}^\top \bar{q}\right) + \tilde{W}^\top \sigma' V^{*\top} \bar{q} \tag{4.18}$$

Substituting now (4.11) and (4.17) in the expression of $\dot{\mathfrak{V}}$ we have

$$\dot{\mathfrak{V}} = -\frac{1}{2} e^\top Qe + e^\top PBw - \kappa \|e\| \operatorname{tr}\left(\tilde{Z}^\top Z\right) \tag{4.19}$$

where

$$Z = \begin{pmatrix} V & 0 \\ 0 & W \end{pmatrix}, \quad \tilde{Z} = Z - Z^* \tag{4.20}$$

Using $\operatorname{tr}(\tilde{Z}^\top Z) \leq \left\|\tilde{Z}\right\| \|Z^*\| - \left\|\tilde{Z}\right\|^2$ and following [Johnson, Kannan, 2002] there exist $a_0, a_1, c_3, \kappa > \|PB\| c_3$ such that

$$\begin{aligned}\dot{\mathfrak{V}} = -\frac{1}{2}\lambda_{min}(Q)\|e\|^2 - (\kappa - \|PB\| c_3)\|e\| \left\|\tilde{Z}\right\|^2 + \\ + a_0\|e\| + a_1\|e\| \left\|\tilde{Z}\right\|\end{aligned} \tag{4.21}$$

and, with $Z_m = \frac{a_1 + \sqrt{a_1^2 + 4a_0(\kappa - \|PB\| c_3)}}{\kappa - \|PB\| c_3}$,

$$\|e\| \geq \frac{a_0 + a_1 Z_m}{\frac{1}{2}\lambda_{min}(Q)} \Rightarrow \dot{\mathfrak{V}} \leq 0 \tag{4.22}$$

Thus for convenient initial conditions, the tracking error e is ultimately uniformly bounded.

5 A Case Study

Adopting a reference path given by

$$P_r = \left\{\frac{V_r}{\Omega_r}\cos(\Omega_r t),\ \frac{V_r}{\Omega_r}\sin(\Omega_r t),\ h,\ \Omega_r t,\ 0,\ 0\right\} \tag{5.1}$$

and the parametric values $V_r = 0.2, \Omega_r = 0.2, m = 2, l = 0.4, J_1 = 0.5, J_2 = 0.1, J_3 = 0.1, n_1 = 9, n_2 = 3, n_3 = 3, \Gamma_{V_\xi} = 20I, \Gamma_{W_\xi} = 20I, \Gamma_{V_\eta} = 10I, \Gamma_{W_\eta} = 10I, K_{p_\xi} = K_{d_\xi} = 1I, K_{p_\eta} = 18I, K_{d_\eta} = 2I, \kappa_\xi = \kappa_\eta = 0.1$ and also considering a perturbation δ_η such as

$$\delta_\eta = (\cos(0.5t)),\ \sin(0.7t),\ \cos(0.2t)) \tag{5.2}$$

added to τ_η in 4.5, the validity of the proposed controller can be noticed in figures (4,5,6). In figures (7,8) the neural net weights evolution is shown.

6 Conclusions

This paper presents the adaptive augmentation of a linear tracking controller. This augmentation prevents

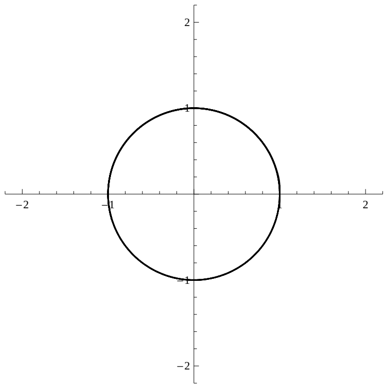

Figure 4. Circular path tracking without adaptive augmentation without external perturbations

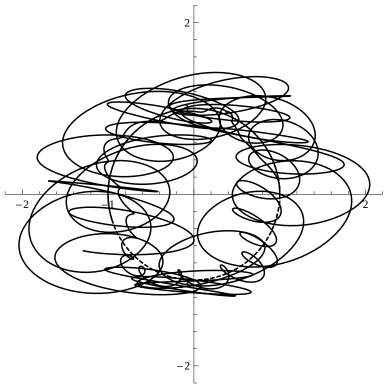

Figure 5. Circular path tracking without adaptive augmentation with external perturbation δ_η

and cancels unmodeled perturbations, making possible the adoption of a simplified plant model. This is specially worth in UAVs and particularly in quad-rotors. The simulations confirm the robustness of this methodology.

7 Acknowledgments

This work was supported by the Spanish Ministry of Science and Technology (Research Project DPI2007-66455-C02-02)

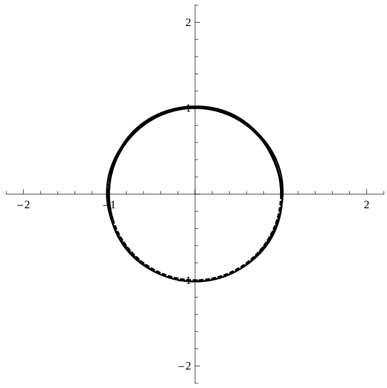

Figure 6. Circular path tracking with adaptive augmentation with external perturbation δ_η

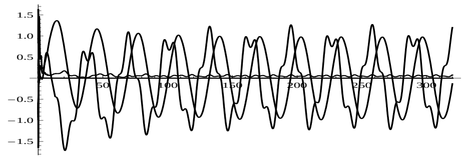

Figure 7. Weights W_ξ evolution along the path

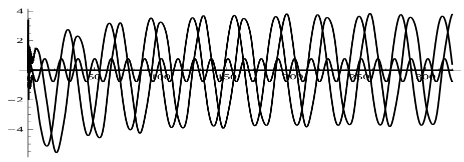

Figure 8. Weights W_η evolution along the path

References

Nakwan Kim, *Improved Methods in Neural Network-Based Adaptive Output Feedback Control, with Applications to Flight Control*, Doctoral Thesis. Georgia Institute of Technology November 2003.

N. Kim, A. J. Calise, *Neural Network Based Adaptive Output Feedback Augmentation of Existing Controllers,* Aerospace Science and Technology (2007), doi: 10.1016/.ast.2007.07.001.

Suresh K. Kannan and Eric N. Johnson, *Adaptive Flight Control for an Autonomous Unmanned Helicopter, AIAA*, 2002.

Pedro Castillo, Rogelio Lozano, Alejandro E. Dzul, *Modeling and Control of Mini-Flying Machines, Springer - Advances in Industrial Control*, (2005),ISBN-1852339578.

S. Salazar-Cruz, A. Palomino, R. Lozano, *Trajectory tracking for a four rotor mini-aircraft, 44th Conf. on Decision and Control*, Seville, Spain Dec. 12-15, 2005.

A. Isidori, *Nonlinear Control Systems*, third edition, Springer-Verlag ed., London, 1995.

Yoonghyun Shin, *Neural Network Based Adaptive Control for Nonlinear Dynamic Regimes, P. D thesis*, Georgia Institute of Technology, 2005.

K. Hornik, M. Stinchcombe and H. White, *Multilayer Feedforward Networks are Universal Approximators, in IEEE Transactions on Neural Networks*, 2(1989), 359-366.

Flavio Nardi, *Neural Network Based Adaptive Algorithms for Nonlinear Control, Ph. D. thesis*, Georgia Institute of Technology, 2000.