1. Introduction

Significant efforts have been devoted to increase the flight endurance and payload of Unmanned Aerial Vehicles (UAVs). Extended endurance has been achieved due to the progress made in propulsion and aerodynamics without the need for a pilot in the system. However, one of the major limitations of UAVs is currently the lack of an aerial-refuelling capability as is present in manned aircraft. An air force having unmanned vehicles that can refuel in the air would have the advantage of greatly extended endurance (weeks, months, and possibly even years) and the ability to immediately respond to a threat thousands of miles away. Without a man in the cockpit and with an aerial-refuelling capability, the endurance of the aircraft would only be limited by routine maintenance, allowing new payloads, giving rise to new operational capabilities (Clark, 2000; Pardesi, 2005; Withrow, 2004; Fravolini et al., 2003; Thompson, 1998; Vendra et al., 2007; Oller and Merino, 2004; Jin et al., 2006).

Autonomous command and control have been identified as one of the core areas of technology that will ensure success in future Unmanned Aerial Vehicle (UAV)/Unmanned Combat Aerial Vehicle (UCAV) development (Withrow, 2004; Clark, 2000; Hansen et al., 2004; Wong and Bill, 1998). From a control point of view, manned aerial refuelling is a difficult task for the pilot, especially in bad weather conditions (Pardesi, 2005; Wilson, 2005; Withrow, 2004). Hence Automated Aerial Refuelling (AAR) is deemed a challenging task (Withrow, 2004; Hansen et al., 2004; Khanafseh and Pervan, 2007). Without the presence of a skilled pilot in the cockpit, it becomes a control system challenge for a UAV during the refuelling manoeuvre. For AAR to be successful, the receiver aircraft must maintain an accurate position relative to the tanker aircraft (Pachter et al., 1997). This can only be achieved by a proper control system development. In the context of unmanned systems, it has been stated that the aim of aerial refuelling research is to develop control systems to reliably guide aircraft during this phase of flight (Mao and Eke, 2008).

Selection of correct gains required by a traditional controller affects the system response and stability. With...
degradation in controller performance, gains need to be re-tuned. Also, controller gains are dependent on the operating conditions of the aircraft—should these change, new gains are required. Re-tuning controller gains is impractical and time consuming (Soares et al., 2006). In addition, traditional controller design usually involves complex and extensive mathematical analysis, which implies high cost and cannot guarantee a good performance level in the whole flight envelope. Assuming linear and time invariant dynamics requires gains to be scheduled as functions of the nominal flight condition in a conventional flight controller. It could happen that, in certain flight conditions, the performance of such systems can deteriorate due to unmodelled effects and non-linearities present in the flight dynamics. Gain scheduling has been used to deal with nonlinear elements present in aircraft dynamics, provided the gain scheduling parameters vary slowly compared with the dynamic response of the aircraft. This process has worked well but, with increased capabilities and performance requirements for modern aircraft, traditional controllers often do not offer acceptable performance (Reiner et al., 1996; Gili and Battipede, 2001; Lane and Stengel, 1988).

Neural Network (NN) control schemes have been put forward as a valid means to overcome most of the typical limitations of classical control techniques and to solve complex control problems. Hagan and Demuth (1999) presented a review of various neural control techniques, such as adaptive inverse controllers, nonlinear internal model controllers, model reference controllers, adaptive critic controllers, stable direct adaptive. Adaptive control offers an alternative to completely re-designing a control system for these changes (Gili and Battipede, 2001; Dufrene, 2004; Soares et al., 2006).

Nonlinear Dynamic Inversion (NDI) control laws can also be used to overcome limitations of conventional controllers. The advantage provided by the NDI control law is the ability to directly command specific state variables. This controller has a more accurate representation of forces and moments due to large state and control perturbations. The use of dynamic inversion eliminates the need for a gain scheduled controller. Nonlinear dynamic inversion control as applied by Spaulding et al. (2005), Steinberg (2001), Lombaerts et al. (2011), Bajodah (2009) and Lee et al. (2005) has been successful. However, NDI controllers are susceptible to errors arising from the inversion of the plant dynamics.

Research has shown that neural networks can be used to overcome this issue in dynamic inversion controllers. The neural network cancels inversion errors by learning the approximate inverse plant dynamics (McFarland and Calise, 2000; Lane and Stengel, 1988).

Online neural networks have been used for controllers to be able to adapt to changes in the aircraft dynamics and to be applicable throughout the entire flight envelope (Kaneshige et al., 2000; Li et al., 2001; Battipede et al., 2003). When considering a very wide operating envelope, the online network has been shown to be effective but, as highlighted in the study of Battipede et al. (2003), intense computation is required for this option.

A Growing Radial Basis Function Network (GRBFN) was proposed by Li et al. (2001). This was done in order to capture the aircraft dynamics over a wide operating range. Kamalasadan and Ghandakly (2011) designed an online Radial Basis Function Neural Network (RBFNN)-based controller in parallel with a Model Reference Adaptive Controller (MRAC) for a fighter aircraft pitch-rate command tracking.

When modelling the receiver aircraft during aerial refuelling, there are three considerations: relative motion of receiver with respect to the tanker, the changing mass and inertia properties, and the effect of the tanker’s wake vortices on the receiver (Waishke et al., 2009). Studies involving the modelling of UAVs during aerial refuelling (Dogan et al., 2005; Ochi and Kominami, 2005; Pollini et al., 2003; Fravolini et al., 2003) have considered the aircraft centre of gravity (cog) position to be fixed. Linear models of receiver aircraft have been developed (Wang et al., 2008) in conjunction with a wake model. Much effort has been channelled to specifically research issues related to finding the relative distance between a tanker and a UAV (Vendra et al., 2007; Dell’Aquila et al., 2007; Mammarella et al., 2008).

Various control systems have been proposed thus far in order to address the UAV automated aerial refuelling problem (Mao and Eke, 2008). A Linear Quadratic Regulator (LQR)-based multi input/multi output state feedback and integral control technique was considered for a tanker and receiver aircraft in an aerial refuelling manoeuvre (Dogan and Kim, 2007). Fravolini et al. (2003) developed an LQR controller for the tanker and an $H_\infty$ controller for the receiver. Pachtet et al. (1997) proposed a control system based on quantitative feedback theory for an air-to-air refuelling operation. Proportional navigation guidance, where the line of a sight angle is controlled similarly as in missile guidance, was considered (Ochi and Kominami, 2005). Khanafseh and Pervan (2007) put forward a GPS-based navigation approach for the control of UAVs during aerial refuelling. The controller developed for the linear model by Wang et al. (2008) was a novel $L_1$ adaptive neural network controller. A PID controller was considered by Pollini et al. (2003). The preliminary steps involved with setting up a multi-input multi-output feedback linearization controller were considered by Elliot and Dogan (2010). A linear controller were presented, along with an investigation within nonlinear control where the selection of various combinations of controlled (output) variables was considered.
Design of NDI-based neural controllers for unmanned combat aerial vehicles during aerial refuelling has not been considered in the surveyed literature. However, in other applications, NDI controllers have been used and found to give useful results. Spaulding et al. (2005) studied an unaugmented ducted fan UAV, having nonlinear dynamic characteristics, and both unstable and susceptible to disturbances. A nonlinear dynamic inversion controller was proposed to overcome the poor flying qualities of the UAV. The controller approach used was found to be adequate in that the complex vehicle dynamics were reduced to those of a simple integrator, which was able to cover the entire flight envelope without the need for gain scheduling. A comparative study within the field of neural control for aircraft used seven control techniques with a cover the entire flight envelope without the need for gain to those of a simple integrator, which was able to controller approach used was found to be adequate that it was difficult to develop another scheme that could exceed its performance overall.

A helicopter trajectory tracking controller was designed by Lee et al. (2005). In the implementation of feedback linearization, it is necessary to invert the plant dynamics. A simple Sigma-Pi neural network was used to this end. Further, this study considered the lack of full knowledge of the helicopter dynamics. It was found that the inversion error due to insufficient information regarding the dynamics was reduced through the use of neural networks. Deng et al. (2009) applied the combination of Model Predictive Control (MPC) with feedback linearization control to address the issues of input constraints. This work led to taking advantage of two control techniques. A reconfigurable flight control algorithm was also designed combining model identification and adaptive NDI (Lombaerts et al., 2011). Parametric uncertainty was considered along with dynamic inversion by MacKunis et al. (2010). Dynamic inversion methodologies were also considered in spacecraft attitude control (Bajodah, 2009).

The main objective of the study is to investigate the suitability of using an NDI-based neurocontroller for a UCAV during aerial refuelling in order for it to maintain an accurate position relative to a tanker aircraft. A controller is developed for the receiver aircraft in the straight leg of a refuelling manoeuvre. Control of the receiver is considered relative to the Earth-fixed inertial frame. The time-varying cog position of the UC AV is considered. The F-16 aircraft is selected as the basis for a converted UC AV concept (Pardesi, 2005; Clark, 2000).

A specific case in a possibly large operating envelope is considered; for this reason, a simple RBF network will be sufficient to use, as opposed to the one proposed by Li et al. (2001). It is unnecessary at this stage to consider using an online network as the neural networks used are trained off-line to simply invert the plant dynamics.

The rest of the paper is organized as follows. Section 2 describes the nonlinear six-DOF aircraft model with the inclusion of the effects of the cog position travel. The proposed control algorithms based on NDI with an RBFNN are formulated and discussed in Section 3. Section 4 presents model validation, NN training results, aircraft inherent behaviour and aircraft controlled behaviour. Finally, concluding remarks are given in Section 5.

2. Mathematical model of the receiver aircraft

2.1. Equations of motion. The aircraft dynamic equations of motion with the inclusion of the effects of a change in the cog position are derived from the first principles using Lagrange’s equation (Pedro, 1992):

\[
\frac{d}{dt} \left( \frac{\partial T^*}{\partial \dot{q}_i} \right) - \frac{\partial T^*}{\partial q_i} + \frac{\partial U}{\partial \dot{q}_i} = Q_i,
\]

where

\[
Q_i = \frac{\partial}{\partial q_i} (\delta W),
\]

where \(T^*\) is the total kinetic energy of the aircraft, \(U\) is the total potential energy of the aircraft, \(Q_i\) is the generalized force, \(q_i\) stands for the generalized coordinates, and \(\delta W\) is the work associated with any arbitrary displacements of the generalized coordinates. Equation (1) is consistent with the equations of motion of Etkin and Reid (1996) as well as Steven and Lewis (1992) when these effects are neglected. The twelve aircraft equations of motion are broken down into four sets:

**Force equations:**

\[
X = m (\dot{u} + qw - rv) - S_x (q^2 + r^2)
\]

\[
- S_y (\dot{r} - pq) + S_z (\dot{q} + pr) + mg \sin \theta,
\]

\[
Y = m (\dot{v} + ru - pw) + S_x (\dot{r} + q) \]

\[
- S_y (p^2 + r^2) - S_z (\dot{p} + qr) - mg \cos \theta \sin \phi,
\]

\[
Z = m (\dot{w} + pv - qu) - S_x (\dot{q} - p r)
\]

\[
+ S_y (\dot{p} + qr) - S_z (q^2 + p^2) - mg \cos \theta \cos \phi,
\]

where \(X, Y, Z\) and \(G\) are the generalized forces due to aerodynamic and propulsive sources, \(u, v, w\) are the aircraft velocity components along \(x, y,\) and \(z\)-body axes, respectively, \(p, q,\) and \(r\) are the aircraft roll, pitch, and yaw rates, respectively, \(\phi\) and \(\theta\) are the aircraft roll and
pitch angles respectively, \( m \) is the aircraft mass, \( g \) is the acceleration due to gravity, and \( S_x, S_y, \) and \( S_z \) are the static moments of inertia about \( x, y, \) and \( z \)-body axes, respectively (this is caused by the cog movement during aerial refuelling).

**Moment equations:**

\[
L = I_x \ddot{\phi} - (I_y - I_z) qr - I_{xy} (q - pr) - I_{xz} (\dot{r} + p q) - I_{yz} (q^2 - r^2) + S_y (\dot{w} - u q + p v) - S_z (\dot{w} - p w + u r) - m g (y_c \cos \theta \cos \phi - z_c \cos \theta \sin \phi) - I_T \omega_T (r \sin \phi_T x_z + q \cos \phi_T z_x \sin \phi_T y_x),
\]

\[
M = I_y \ddot{\theta} - (I_x - I_z) rp - I_{xy} (\dot{q} - qr) - I_{xz} (\dot{r} - q r) - I_{yz} (p^2 - q^2) + S_x (\dot{w} - u q + p v) + S_z (\dot{w} - r v + q w) + m g (z_c \sin \theta) + x_c \cos \theta \cos \phi) + I_T \omega_T (q \cos \phi_T x_z \cos \phi_T y_x + p \cos \phi_T z_x),
\]

\[
N = I_z \ddot{\psi} - (I_x - I_y) p q - I_{xy} (\dot{q} - pr) - I_{xz} (\dot{r} - q r) - I_{yz} (p^2 - q^2) + S_x (\dot{w} - w p + r u) - S_y (\dot{w} - r v + q w) - m g (x_c \cos \theta \sin \phi) + y_c \sin \phi) - I_T \omega_T (q \cos \phi_T x_z \sin \phi_T y_x + p \sin \phi_T z_x),
\]

where \( L, M, \) and \( N \) are the generalized moments due to aerodynamic and propulsive sources, \( I_x, I_y, \) and \( I_z \) are the vehicle moments of inertia about \( x, y, \) and \( z \)-body axes, respectively, \( I_{xy}, I_{xz}, \) and \( I_{yz} \) are the vehicle products of inertia, respectively, \( I_T \) is the moment of inertia of the spinning rotor about its major axis, \( \omega_T \) is the angular velocity of the spinning rotor, and \( \phi_T, \) \( \theta_T, \) and \( \psi_T \) are the engine pitch and yaw settings, respectively.

\[
I_x = I_{x0} + m (y_c^2 + z_c^2),
\]

\[
I_y = I_{y0} + m (x_c^2 + z_c^2),
\]

\[
I_z = I_{z0} + m (x_c^2 + y_c^2),
\]

\[
I_{xy} = I_{x0} y_c + m x c y_c,
\]

\[
I_{xz} = I_{x0} z_c + m x c z_c,
\]

\[
I_{yz} = I_{y0} z_c + m y c z_c,
\]

\[
S_x = m x c, \quad S_y = m y c, \quad S_z = m z c,
\]

where \( I_{x0}, I_{y0}, I_{z0}, I_{x0}, I_{z0}, \) and \( I_{yz0} \) are the vehicle moments and products of inertia before refuelling occurs.

The cog position vector, \( r_c, \) shown in Fig. 1, is given by

\[
r_c = [x_c, y_c, z_c]^T
\]

\[
= [x_{c,\text{ref}} - x_c, y_{c,\text{ref}} - y_c, z_{c,\text{ref}} - z_c]^T,
\]

where \( x_{c,\text{ref}}, y_{c,\text{ref}}, \) and \( z_{c,\text{ref}} \) are the cog positions along the \( x, y, \) and \( z \)-body axes, respectively, and \( x_{c,\text{ref}}, y_{c,\text{ref}}, \) and \( z_{c,\text{ref}} \) are the reference cog positions along the \( x, y, \) and \( z \)-body axes, respectively.

The generalized forces and moments due to aerodynamic and propulsive sources are given as (Pedro and Bigg, 2005)

\[
X = q S [C_{xa} + C_{za} \alpha + (\dot{\epsilon}/2V) (C_{xa} (\alpha) q + C_x (\alpha) \dot{\alpha} + C_{za} \delta_e) + T \cos \phi_T x_z \cos \phi_T y_x,
\]

\[
Y = q S [C_{ya} + C_y \beta + (\dot{\beta}/2V) (C_{ya} (\alpha) p + C_y (\alpha) r + C_{ya} \delta_a + C_{yb} \delta_e) + T \sin \phi_T,
\]

\[
Z = q S [C_{za} + C_{za} \alpha + (\dot{\epsilon}/2V) (C_{za} (\alpha) q + C_z (\alpha) \dot{\alpha} + C_{za} \delta_e) - T \cos \phi_T z_x \sin \phi_T y_x,
\]

\[
L = q S [C_{la} + C_{la} \beta + (\dot{\beta}/2V) (C_{la} (\alpha) p + C_{la} (\alpha) r + C_{la} \delta_a + C_{la} \delta_e) - T (g_T \cos \phi_T x_z \sin \phi_T y_x + z_T \sin \phi_T y_x),
\]

where \( \dot{\epsilon} \) is the dynamic pressure, \( V \) is the aircraft velocity, \( \rho \) is the air density, \( S \) is the wing area, \( \dot{\epsilon} \) is the mean aerodynamic chord, \( b \) is the wing span,
\( C_x, C_y, \) and \( C_z \) are the aerodynamic force coefficients along \( x, y, \) and \( z \)-body axes, respectively, \( C_l, C_m, \) and \( C_n \) are the aerodynamic rolling, pitching, and yawing moment coefficients, respectively, \( \delta_a, \delta_e, \) and \( \delta_r \) are the aileron, elevator, and rudder deflections, respectively, \( T \) is the thrust force, \( \varphi_{T_y} \) and \( \varphi_{T_z} \) are the engine pitch and yaw settings, respectively, and \( x_T, y_T, \) and \( z_T \) are the coordinates of point of thrust application in the body frame.

### Kinematic equations:

\[
\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta, \quad (18)
\]
\[
\dot{\theta} = q \cos \phi - r \sin \phi, \quad (19)
\]
\[
\dot{\psi} = (q \sin \phi + r \cos \phi) \sec \theta. \quad (20)
\]

### Navigation equations:

\[
\dot{x}_E = u \cos \theta \cos \psi + v (\sin \phi \sin \theta \cos \psi - \cos \phi) = \sin \psi' + w (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi'), \quad (21)
\]
\[
\dot{y}_E = u \cos \theta \sin \psi + v (\sin \phi \sin \theta \sin \psi - \cos \phi) = \cos \psi' + w (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi'), \quad (22)
\]
\[
\dot{h} = -\dot{z}_E = u \cos \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta, \quad (23)
\]
where \( x_E, y_E, \) and \( h \) are the aircraft position components in the Earth-fixed frame. The angle of attack, \( \alpha, \) the sideslip angle, \( \beta, \) and the total speed, \( V, \) are calculated from

\[
\alpha = \tan^{-1} \left( \frac{w}{u} \right), \quad \beta = \sin^{-1} \left( \frac{v}{V} \right), \quad (24)
\]
\[
V = \sqrt{u^2 + v^2 + w^2}. \quad (25)
\]

The throttle position, \( \delta_T, \) is related to the thrust, \( T, \) as follows (Pedro and Bigg, 2005):

\[
T = \left( \frac{\rho}{\rho_0} \right)^{0.7} \left( AV^2 + BV + C \right) \delta_T / \delta_{T_{\max}}. \quad (26)
\]

The thrust component axes transformations from engine to body axes are indicated in Fig. 2. The time lag of the control surfaces was also modelled. This has been modelled as transfer functions of

\[
G(s) = \frac{\delta_i(s)}{\delta_c(s)} = \frac{1}{\tau_e s + 1}. \quad (27)
\]

where \( \delta_c, \) is the commanded control deflection, \( \delta_i, \) is the actual control deflection, and \( \tau_e \) is the time constant. The time lag of \( 0.0495 \, \text{s} \) is assumed in the movement of the aerodynamic control surfaces while the engine lag is taken to be \( 0.5 \, \text{s} \) (Kung, 2008).

Examining the equations of motion as has been described, it is observed that there is a coupling between the lateral-directional and longitudinal motions via \( S_x, S_y, \) and \( S_z. \) The force in the \( X \)-direction is seen to be dependent on the yaw acceleration and the square of the yaw rate via the cog position. The \( Y \)-direction force is a function of the roll acceleration due to the centre of gravity being misaligned with the reference cog. Similarly with the \( Z \)-direction force, a coupling exists between the longitudinal and lateral-directional variables. The moment equations also show a very interesting cross coupling of terms related via the centre of gravity position. The yaw and roll moments exhibit dependence on the rate of change of vertical and forward speed, respectively. All three moments have contributions from the mass of the aircraft due to the cog having changed the position from the reference.

The following simplifying assumptions have been applied in order to obtain the specific flight refuelling model:

- The rate of fuel used during the refuelling period was negligible compared with the rate of the fuel inflow.
- The products of inertia of the aircraft remained constant throughout the refuelling period.
- Only the variation of the cog along the \( x \)-axis is considered:
  \[
  x_c \neq 0, y_c = z_c = 0 \Rightarrow S_x \neq 0, S_y = S_z = 0. \quad (28)
  \]
- The line of action of the thrust force is assumed to be aligned in the plane of the aircraft cog, and thus gives no contribution to the force in the \( y \)-direction and \( z \)-direction; neither does it contribute to any of the aerodynamic moments, i.e.,
  \[
  \varphi_{T_y} = \varphi_{T_z} = 0, \quad y_T = z_T = 0. \quad (29)
  \]
The gyroscopic effects of engine spinning rotors are negligible, i.e.,

\[ I_T \omega_T \approx 0. \] (30)

Using body axes, the plane \( Cxz \) is a plane of symmetry i.e.,

\[ I_{xy} = I_{yz} = 0. \] (31)

After applying the assumptions given by Eqns. (28)-(31), the final dynamic equations of motion can be written as follows:

\[
m(\dot{u} + qw - rv) - S_x (q^2 + r^2) = -mg \sin \theta + T - qSC_z, \] (32)

\[
m(\dot{v} + ru - pw) + S_x (\dot{r} + qp) = mg \cos \theta \sin \phi + qSC_y, \] (33)

\[
m(\dot{w} + pv - qu) - S_x (\dot{q} - pr) = mg \cos \theta \cos \phi + qSC_z, \] (34)

\[
I_x \dot{p} - (I_y - I_z) qr - I_{xz} (\dot{r} + pq) = \dot{q}SC_l, \] (35)

\[
I_y \dot{q} - (I_z - I_x) rp - I_{xy} (r^2 - p^2) - S_x (\dot{w} - wp + ru) = -mg \cos \theta \sin \phi + qSC_m, \] (36)

\[
I_z \dot{r} - (I_x - I_y) pq - I_{xz} (\dot{q} - qr) + S_x (\dot{w} - wp + ru) = mg \cos \theta \cos \phi + qSC_n. \] (37)

The kinematic and navigation equations remain unchanged.

### 2.2. Flight refuelling model

For the UCAV configuration of the F-16, there are two 600 gallon tanks on each wing (Eshel, 2004), as depicted in Fig. [3]. This gives a total mass of 4290 kg of fuel. The fuelling rate for the F-16 is 4572.12 kg of fuel in 7 minutes, that is, 10.886 kg/s (Koch, 2005). Based on this, the total time taken to refuel the F-16 external tanks is 395 s. Note that, before refuelling, the F-16 mass is 9299 kg (Steven and Lewis, 1992). Figure [4] shows the layout of the wing/external fuel tank.

2.3. Trim conditions. “An aircraft is in trimmed, equilibrium flight when its velocity is fixed and its pitch and roll angles are unchanging” (Stengel, 2004).

The trim problem is to find the control settings that yield a steady flight condition. The objective is to set the vector equation:

\[ \dot{x}_d = f_t (x_d, u, w), \] (39)

by a proper choice of control \( u \) subject to the constant values of \( x_d \) and disturbance \( w \), where

\[ x_d = [u, v, w, p, q, r]^T. \] (40)

The remaining components,

\[ x_k = [x_E, y_E, h, \phi, \theta, \psi]^T, \] (41)

may be treated as fixed or free parameters.

The desired trim condition can be specified by various combinations of velocity and angle components. With conventional \( \delta_e, \delta_a, \delta_r, \) and \( \delta_T \), a typical specification would be \( V, \gamma = \theta - \alpha \) (flight path angle), \( \beta, \psi, \theta, \) and \( \phi \) would be free to take the necessary values. The trim variables and attitude angles specify the corresponding \( u, v, w, p, q, \) and \( r \). Although they are free variables, \( \theta \) and \( \phi \) do not change unless forced to do so,
i.e., an adjustment rule is required. That rule is provided by incorporating the free variables in the control vector, hence \( u \) is (Stengel, 2004):

\[
u = [\delta_a, \delta_e, \delta_r, \theta, \phi]^T. \tag{42}
\]

### 3. Controller implementation

#### 3.1. Nonlinear dynamic inversion.

The equations of motion as described by Eqns. (32)–(37) and (38)–(39) can be written in a matrix form as

\[
\dot{x} = F(x, u) = f(x) + g(x)u, \tag{43}
\]

\[
y = h(x), \tag{44}
\]

where \( f \) and \( g \) are nonlinear state and control distribution functions, respectively. The state vector, \( x \), the control vector, \( u \), and the output vector, \( y \), are given as

\[
x = [x_1, \ldots, x_{12}]^T
\]

\[
= \begin{bmatrix}
(u, v, w)^T \\
(p, q, r)^T \\
(\phi, \theta, \psi)^T \\
(x_E, y_E, h)^T
\end{bmatrix}
\in \mathbb{R}^{12},
\tag{45}
\]

\[
u = [u_1, \ldots, u_4]^T = \begin{bmatrix}
\delta_a \\
\delta_e \\
\delta_r \\
\theta \\
\phi
\end{bmatrix}
\in \mathbb{R}^5,
\tag{46}
\]

\[
y = [y_1, \ldots, y_4]^T = \begin{bmatrix}
\beta \\
p \\
q \\
r
\end{bmatrix}
\in \mathbb{R}^4.
\tag{47}
\]

Equation \(43\) can be rewritten as

\[
\dot{X}_1 = F_1(V, \alpha, \beta, p, q, r, h, \theta, \phi, \delta, \delta_T),
\tag{48}
\]

\[
\dot{X}_2 = F_2(V, \alpha, \beta, p, q, r, h, \delta, \delta_T),
\tag{49}
\]

\[
\dot{X}_3 = F_3(p, q, r, \phi, \psi),
\tag{50}
\]

\[
\dot{X}_4 = F_4(V, \phi, \theta, \psi).
\tag{51}
\]

The control law can be found from Eqn. \(43\) as

\[
u = g^{-1}(x)[\dot{x} - f(x)],
\tag{52}
\]

under the assumption that \( g(x) \) is invertible for all values of \( x \). The aircraft is to be commanded to specified states, \( x \). But the rates of the state vector will be specified as \( \dot{x}_{com} \). This is substituted in Eqn. \(48\) to obtain

\[
u = g^{-1}(x)[\dot{x}_{com} - f(x)].
\tag{53}
\]

It is important to note that perfect inversion is not always possible. The assumption that \( g(x) \) may be invertible for all \( x \) may not necessarily be true. When there are more states than controls, inversion is not possible. On the other hand, if \( g(x) \) is invertible and small, the control inputs become large. Thus actuator saturation must be taken into account.

### 3.2. RBFNN training and optimization.

The radial basis function has originated from techniques for
performing exact interpolation of a set of data points in a multi-dimensional space. This problem requires every input vector to be mapped exactly onto the corresponding target vector (Bishop, 1995; Haykin, 1999). A schematic representing the network is shown in Fig. 6. The desired inverse functions can be approximated by a general RBFNN of the form

\[
y_k(x) = w_{k0} + \sum_{j=1}^{M} w_{kj} \phi_j(x) = w_{k0} + w^T \Phi(x),
\]

where \(w_{k0}\) is the network bias, \(\phi_j\) are the basis functions, \(w_{kj}\) are the output layer weights, \(M\) is the number of neurons in the hidden layer, \(w = (w_1, w_2, \ldots, w_M)\) is the weight vector and \(\Phi\) is the vector of basis functions. We use the Gaussian basis functions in the present work for the RBFNN, i.e.,

\[
\phi_j(x) = \exp\left(-\frac{\|x - \mu_j\|^2}{2\sigma_j^2}\right),
\]

where \(\mu_j\) and \(\sigma_j\) are the centre and width of the \(j\)-th kernel unit, respectively, and \(x\) is the \(d\)-dimensional input vector with elements \(x_i\) (Bishop, 1995).

When training the RBFNN, initially, the parameters governing the basis functions are determined \((\mu_j, \sigma_j)\) using only the input data \((x^n)\). The basis functions are then kept fixed while the second layer weights are found in the second part of training. Since the basis functions are fixed, the network is equivalent to a single layer network.

The weights can be optimized by the minimization of a suitable error function. It is convenient to consider the sum-of-squares error (Bishop, 1995; Haykin, 1999):

\[
E = \frac{1}{2} \sum_n \sum_k \{y_k(x^n) - t^n_k\}^2,
\]

where \(t^n_k\) is the target value for output unit \(k\) when the network is presented the input vector \(x^n\). The error function is a quadratic function of the weights, and so its minimum can be found in terms of a set of linear equations. The weights are determined by

\[
\Phi^T \Phi W^T = \Phi^T T,
\]

where \(\Phi^T \Phi\) can be approximated by a neural network to obtain the neurocontroller architecture:

\[
[W^T] = \Phi^T \Phi^{-1} \Phi^T T,
\]

where \(\Phi^*\) is the pseudo-inverse of \(\Phi\).

3.3. Combined NDI-RBFNN. The desired inverse functions, \(K_2\), in Eqn. \((58)\) can be approximated by a neural network to obtain the neurocontroller architecture:

\[
\begin{bmatrix} \delta_a, \delta_c, \delta_r \end{bmatrix}^T = NN_2 \begin{bmatrix} V, \alpha, \beta, p, q, r, h, \dot{X}_2, \delta_T \end{bmatrix} = \hat{\Phi}_2 \begin{bmatrix} V, \alpha, \beta, p, q, r, h, \dot{X}_2, \delta_T \end{bmatrix},
\]

where \(\hat{\Phi}_2\) represent estimates of the network bias terms, the RBF weight vector and the basis functions vector, respectively.

A global view of the controller can be seen in Fig. 7. Here \(x_\text{d}\) stands for the commanded states (input to reference model) of the aircraft. This is compared with the state variables in the feedback loop, and the error is sent to the dynamic inversion neural network controller. Manipulations are performed in the controller block (command inversion, desired dynamics and nonlinear inversion), and the input signal, \(u\), is fed into the nonlinear plant.

4. Numerical simulations and results

The equations of motion given in Section 2.1 were modelled in the Matlab and Simulink environment. The full nonlinear model was simulated, and then, the Jacobian matrices were extracted. Once these were obtained, the eigenvalues and other relevant parameters were determined for various points in the refuelling envelope. The modes of motion were identified across the refuelling range. The F-16 aircraft was chosen for simulation. The
Table 1. Geometrical, mass and inertial properties of the F-16.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Values</th>
<th>Symbols</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_x )</td>
<td>12874.84 kg m²</td>
<td>( b )</td>
<td>9.14 m</td>
</tr>
<tr>
<td>( I_y )</td>
<td>75673.58 kg m²</td>
<td>( c )</td>
<td>3.45 m</td>
</tr>
<tr>
<td>( I_z )</td>
<td>85564.4 kg m²</td>
<td>( m )</td>
<td>9900 kg</td>
</tr>
<tr>
<td>( I_{xy} )</td>
<td>1331.41 kg m²</td>
<td>( S )</td>
<td>27.87 m²</td>
</tr>
<tr>
<td>( I_{yz} )</td>
<td>0 kg m²</td>
<td>( I_{xz} )</td>
<td>0 kg m²</td>
</tr>
</tbody>
</table>

The basic geometrical, mass and inertial properties of this aircraft are presented in Table 1 (Morelli, 1998).

The aircraft model was validated by comparing its trim conditions for two cruise cases with relevant and compatible data from the work of Steven and Lewis (1992). The aircraft cruise speed in both cases was 153 m/s and the altitude was 0 m, i.e., at sea level. The distinguishing factor between the two cases was the cog position, the first case having a cog position of 0.3\( c \), while the second cog position was 0.38\( c \). The validation data are presented in Table 2.

Table 2 shows that the results of the Matlab simulation and the cited values are the same up to three decimal places in some cases. Some of the values (viz. the sideslip angle, aileron and rudder deflections) are extremely small, and can be considered to be zero. Thus, the model is accurate in estimating the trim conditions.

The eigenvalues were found and compared for the single case of the aircraft in cruise at a speed of 153 m/s, with the forward cog of 0.3\( c \). The eigenvalues, damping coefficients, and the periods are given in Table 3. The discrepancies are not large enough to cause the model to be inaccurate, and the model will be deemed sufficient to analyse the behaviour of the F-16 accurately.

![Fig. 8](image)

**Fig. 8.** Cases considered in refuelling inherent behaviour analysis.

The Dutch roll movement in Table 4 shows that, as the mass increases, the mode gets more stable. The mode becomes less oscillatory until the cog passes the turning point and starts to decrease again, then the mode gets most, while the spiral and the Phugoid modes are in the same region of values. The Dutch roll mode is the most accurate, and the Phugoid is the least accurate in terms of the damping coefficients and periods. The discrepancies are not large enough to cause the model to be inaccurate, and the model will be deemed sufficient to analyse the behaviour of the F-16 accurately.

Table 3. Comparison of trimmed aircraft conditions: model validation.

| \( V \) | 153 | 153 | 153 | 153 |
| \( \alpha \) | 2.26 | 2.26 | 2.03 | 2.03 |
| \( \beta \) | -6.2e-15 | 2.4e-17 | 3.3e-11 | 1.8e-8 |
| \( \phi \) | 0 | 0 | 0 | 0 |
| \( \theta \) | 2.26 | 2.26 | 2.03 | 2.03 |
| \( \rho \) | 0 | 0 | 0 | 0 |
| \( q \) | 0 | 0 | 0 | 0 |
| \( r \) | 0 | 0 | 0 | 0 |

Table 3. Comparison of eigenvalues, damping coefficients, and periods.

<table>
<thead>
<tr>
<th>Mode types</th>
<th>Simulated results</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda )</td>
<td>( \zeta )</td>
</tr>
<tr>
<td>Short period</td>
<td>-1.29 ± 1.274</td>
</tr>
<tr>
<td>Phugoid</td>
<td>-0.0097 ± 0.0084</td>
</tr>
<tr>
<td>Dutch roll</td>
<td>-0.43 ± 3.256</td>
</tr>
<tr>
<td>Spiral</td>
<td>-0.0021</td>
</tr>
<tr>
<td>Roll</td>
<td>-3.62</td>
</tr>
</tbody>
</table>
more oscillatory. The Phugoid mode is less stable as the mass is increased until the cog position reaches the turning point and starts to decrease again, then the mode gets more stable. The mode becomes more oscillatory as the aircraft mass increases. The Short Period Oscillatory (SPO) mode decomposed into two first-order modes: one highly damped and the other very poorly damped, and remains largely unchanged throughout the flight refuelling process. The decomposition of SPO can be attributed to the addition of mass into the system, as well as an aft cog. When mass is introduced steadily and the cog moves further aft, the results show that the longitudinal dynamics remains largely unchanged throughout the flight refuelling.

Three RBFNNs were set up in Matlab using the Netlab Toolbox developed by Nabney (2002). One NN each was required for determining the commanded elevator, aileron and rudder deflections. The individual networks consisted of 75 neurons and were trained over 5000 cycles each. The training data was generated in Simulink using a random number generator set within the network's characteristics are summarized in Table 4. Eigenvalues changes during refuelling.

Table 4. Eigenvalues changes during refuelling.

<table>
<thead>
<tr>
<th>Case 1</th>
<th>Mode types</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitudinal mode</td>
<td>$-1; -0.0014$</td>
<td></td>
</tr>
<tr>
<td>Phugoid</td>
<td>$-0.0193 \pm 0.118\iota$</td>
<td></td>
</tr>
<tr>
<td>Dutch roll</td>
<td>$-0.271 \pm 2.45\iota$</td>
<td></td>
</tr>
<tr>
<td>Spiral</td>
<td>$-0.00491$</td>
<td></td>
</tr>
<tr>
<td>Roll</td>
<td>$-1.8$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Case 2</th>
<th>Mode types</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitudinal mode</td>
<td>$-1; -0.0015$</td>
<td></td>
</tr>
<tr>
<td>Phugoid</td>
<td>$-0.00855 \pm 0.1\iota$</td>
<td></td>
</tr>
<tr>
<td>Dutch roll</td>
<td>$-0.278 \pm 2.37\iota$</td>
<td></td>
</tr>
<tr>
<td>Spiral</td>
<td>$-0.00536$</td>
<td></td>
</tr>
<tr>
<td>Roll</td>
<td>$-2.18$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Case 3</th>
<th>Mode types</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitudinal mode</td>
<td>$-1; -0.0015$</td>
<td></td>
</tr>
<tr>
<td>Phugoid</td>
<td>$-0.0122 \pm 0.056\iota$</td>
<td></td>
</tr>
<tr>
<td>Dutch roll</td>
<td>$-0.347 \pm 2.67\iota$</td>
<td></td>
</tr>
<tr>
<td>Spiral</td>
<td>$-0.00172$</td>
<td></td>
</tr>
<tr>
<td>Roll</td>
<td>$-1.5$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Case 4</th>
<th>Mode types</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitudinal mode</td>
<td>$-1; -0.0015$</td>
<td></td>
</tr>
<tr>
<td>Phugoid</td>
<td>$-0.0083 \pm 0.093\iota$</td>
<td></td>
</tr>
<tr>
<td>Dutch roll</td>
<td>$-0.287 \pm 2.38\iota$</td>
<td></td>
</tr>
<tr>
<td>Spiral</td>
<td>$-0.00488$</td>
<td></td>
</tr>
<tr>
<td>Roll</td>
<td>$-2.21$</td>
<td></td>
</tr>
</tbody>
</table>

The networks’ characteristics are summarized in Table 5. Limits of the state and control variables used in RBFNN training.

Table 5. Limits of the state and control variables used in RBFNN training.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Lower limits</th>
<th>Upper limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$ (deg)</td>
<td>$-10$</td>
<td>$45$</td>
</tr>
<tr>
<td>$\beta$ (deg)</td>
<td>$-3$</td>
<td>$3$</td>
</tr>
<tr>
<td>$p$ (deg/s)</td>
<td>$-28.65$</td>
<td>$28.65$</td>
</tr>
<tr>
<td>$q$ (deg/s)</td>
<td>$-28.65$</td>
<td>$28.65$</td>
</tr>
<tr>
<td>$r$ (deg/s)</td>
<td>$-28.65$</td>
<td>$28.65$</td>
</tr>
<tr>
<td>$\delta_T$ (-)</td>
<td>$0$</td>
<td>$1$</td>
</tr>
<tr>
<td>$\delta_\alpha$ (deg)</td>
<td>$-25$</td>
<td>$25$</td>
</tr>
<tr>
<td>$\delta_\delta$ (deg)</td>
<td>$-21.5$</td>
<td>$21.5$</td>
</tr>
<tr>
<td>$\delta_\gamma$ (deg)</td>
<td>$-30$</td>
<td>$30$</td>
</tr>
</tbody>
</table>

The networks were tested after training and predictions were compared with the actual values. The neural network errors are plotted in Figs. 9–11 for the elevator, aileron and rudder, respectively. The elevator prediction was the most accurate. The error in prediction of the elevator deflection reached a maximum of just over 1%. However, for the aileron and rudder, excessively high errors were found in certain instances. Errors of 6000% in the aileron prediction and 200% in the rudder prediction can be considered as single outliers. The average of the errors in the aileron prediction was 5.8% and in the rudder prediction 0.5%. This means that the majority of the results are reasonable.

Before the neural network results can be analysed,
the conventional NDI (without an NN for plant inversion) controller results are presented. Figure 12 shows the UCAV state variables over the refuelling period.

The aircraft angle of attack is not constant, nor can the controller maintain the aircraft at a constant altitude and speed during the refuelling period. No change is observed in the pitch rate, sideslip angle, and roll angle. The commanded control surface deflections are shown in Fig. 13. These are found to be very small and insufficient to keep the aircraft in steady flight.

The time domain results presented show that the performance of this controller is inadequate. This can be attributed to the use of the aerodynamic model uncertainties during the flight refuelling process. This led to some inversion error. The NDI controller coupled with the NN shows excellent results. All states are maintained as constant during the 400 seconds duration of the flight refuelling (see Fig. 14). The elevator deflection required to maintain steady flight is shown in Fig. 15. This variation is well within the physical limits of the elevator actuator. The results here can be attributed to the low inversion error of the trained neural network. The results also show that the controller with the NN is less sensitive to the aerodynamic model uncertainties during the flight refuelling process. The NN is trained using the model of Ito et al. (2002) and is implicitly used in this controller.

This result compared to the basic NDI controller with no NN ties up with the conclusions of Li et al. (2001). The neural controller was able to overcome modelling errors which posed a challenge to the NDI controller without the NN.

The results of the lateral controller are promising. The sideslip angle shown in Fig. 16 does not stay steady—a small deviation is observed. The sideslip angle quickly steadies to a fixed value thereafter. However, all other states are steady throughout the refuelling. The aileron and rudder deflections are shown in Fig. 17. These are within the physical limits. The aileron deflection was observed to be unaffected by the refuelling process/mass increase of the aircraft. The rudder deflection followed a similar trend to that of the elevator in Fig. 15. It is also observed that the commanded aileron and rudder deflections are non-zero. This is to maintain the sideslip angle that was observed in Fig. 16. The adequate performance of the lateral neural controller over the plain NDI controller can also be attributed to the addition of the RBF network, which is less sensitive to model errors as described earlier.

5. Conclusion

For the aerial-refuelling model, a nonlinear six-degree-of-freedom aircraft model including the centre of gravity movement during refuelling was built. Published data verified the accuracy of the aircraft model. When refuelling, the centre of gravity variation with
Fig. 12. Time history of controlled state variables for the NDI-based controller without an NN.

Fig. 14. Time history of controlled state variables for the longitudinal NDI-based neurocontroller.
increasing mass was found to be parabolic. Modal analysis of the unmanned vehicle during refuelling was done. For the Dutch roll mode, as refuelling occurs, the mode is initially less oscillatory, but as the mass increases further, the cog starts to move forward again and the Dutch roll mode becomes more oscillatory.

Controllers were designed to maintain the aircraft path in the straight leg of a racetrack manoeuvre. A dynamic inversion controller with no neural network was used initially. The performance of this controller was not satisfactory. This is attributed to the difference in aerodynamic models used to evaluate the aerodynamic forces in the six-degree-of-freedom model and in the inversion. This controller was sensitive to model uncertainties.

When training the dynamic inversion controller with
the radial basis function network, it was found that the error in the elevator prediction was the lowest. The longitudinal nonlinear dynamic inversion-based neurocontroller gave excellent results. The control surface deflections were within the physical limits of the aircraft. The lateral-directional nonlinear dynamic inversion-based neurocontroller showed promising results. However, steady flight was not able to be maintained initially. The results were dependent on the training of the neural network via the number of neurons used. Finally, it can be concluded that a nonlinear dynamic inversion controller with a neural network can be applied to control an unmanned combat aerial vehicle during aerial refuelling.

As this field of study has not yet been exhausted, many avenues for future work are still available. Other neural network architectures (other than radial basis functions) can be used. This can extend to the well known multi-layered perceptron, where results using this, or any other architecture, can be compared with existing radial basis function based results. The use of dynamic or recurrent neural networks can be explored. Moving away from the neural implementation, recommendations can be made with respect to modelling sources of uncertainty, thereby giving robust and reliable simulation results. Further considering robustness and reliability applied to modelling and simulation, aircraft parameters (such as inertia/mass properties, trimmed state variables) can be statistically varied and subjected to Monte Carlo analysis. The changing inertia tensor and aerodynamic effects of the tanker on the receiver can also be added to the dynamic model. Another area where this work could be improved is to explore the possibility of a combined longitudinal and lateral-directional controller.

References

A nonlinear dynamic inversion-based neurocontroller for unmanned combat aerial vehicles.


Jimoh Olarewaju Pedro received the M.Sc. and Ph.D. degrees in aeronautical engineering from the Warsaw University of Technology in 1986 and 1992, respectively. He was a post-doctoral associate at the Institute of Aviation in Warsaw (1993–1994). He is currently an associate professor at the School of Mechanical, Industrial and Aeronautical Engineering, University of the Witwatersrand, Johannesburg, South Africa. He has published over 50 articles in refereed journals and conference proceedings. His research interests include applications of soft computing techniques to mechatronic systems and aerospace vehicles, trajectory optimization, and applications of robust control theory to rigid/flexible aerospace vehicles.

Aarti Panday received a B.Sc.Eng. and an M.Sc.Eng. from the University of the Witwatersrand in 2004 and 2009, respectively. She currently works as an aircraft certification engineer at the South African Civil Aviation Authority and is pursuing a Ph.D. degree in aeronautical engineering at the University of the Witwatersrand. Her main research interest is aerospace control system design.

Laurent Dala is the head of Aeronautical Engineering at the University of the Witwatersrand. He graduated from French Grandes Ecoles, ESTP and ESTA, and received a Ph.D. in aerodynamics from the University of Manchester (UK). He is a fellow of the Royal Aeronautical Society (FRAeS), a chartered engineer and a member of the Aerodynamics Specialist Group of the Royal Aeronautical Society. In the last 20 years, he has developed international experience in aerospace industry and research working at ONERA (France), Cargo Lifter (Germany), Institut AroTechnique (IAT/CNAM) and the Aircraft Research Association Ltd (UK). This has allowed him to have close association with notable eastern European institutes like TsAGI (Russia) and the Warsaw University of Technology. He has also served as the chairman of the Advisory Group for the European project New Aircraft Concepts Research (NACRE) within the European Commission Framework Programme for the past five years.

Received: 15 December 2011
Revised: 25 May 2012