IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676, p-ISSN: 2320–3331, Volume 8, Issue 1 (Nov.-Dec.), PP 120-124
www.iosrjournals.org

An Unmanned Rotorcraft System with Embedded Design

[1]Sateesh Kumar Reddy Ch., [2]G. Anand Babu, [3]Umasankar Ch.
Dept. of ECE, Universal College of Engineering & Technology, Guntur, AP, India

Abstract: *In this paper, we present the systematic design and implementation of quick prototype design and hardware-in-the-loop real-time test of the embedded control system of Unmanned Rotorcraft. The control law is of backstepping type, the sensory system consists of a marker-based vision system outside the helicopter in radio connection with the embedded controller and a 3D inertial measurement unit (IMU) on the helicopter. Extended Kalman filters solve the state estimation problem. Brushless DC motors serve as actuators. For quick prototype design of the embedded controller MATLAB, Simulink, Real-Time Workshop and the MPC555 Target Compiler were used. More specifically, the onboard hardware system is designed to fulfill the image processing requirements by using the commercial off-the-shelf products. Real-time vision software is developed, which is running on the real-time operating system QNX.*
Keywords: *Unmanned Aerial Vehicles (UAV), Embedded Controller, Real-time realization, Marker-based vision and IMU sensors, Extended Kalman filters, Backstepping control.*

I. INTRODUCTION

The identification of the unknown nonlinear dynamical systems has received considerable attention in recent years, since it is an indispensable step toward controller design of nonlinear systems. Many systems, especially aircraft, have dynamics that vary considerably over the operating regime, effectively bringing the issue of time varying parameters (or nonlinearity) into the design [2]. Controllers for aircraft have been designed predominantly by classical control techniques [9].

While this tradition has produced many highly reliable and effective control systems, recent years have seen a growing interest in the use of robust, nonlinear adaptive control theory for flight control [3], [4], [6]. For instance, the concept of multiple models with switching, according to a change in dynamics, has been an area of interest in control theory in order to simplify both the modeling and the controller design [3], [7], [8].

System identification methods fall into two broad categories: global and local. Global approximations can be made with various function representations, e.g., polynomials, rational approximation, and multilayer perceptrons (MLPs) [9]. To approximate a function , a model should be capable of representing its many possible variations. If is complicated, there is no guarantee that any given representation will approximate equally well across all space.

The dependence on representation can be reduced using local approximation where the domain of is divided into local regions and a separate model is used for each region [5].Local modeling is based on nearest-neighbors in the operating space where a simple model is constructed using only the neighboring samples. The rationale behind this approach is basically that it is easier to develop local models (or controllers) because the dynamics are simpler locally than globally [3].

For instance, if the system phenomena or behavior changes smoothly with the operating point, then, a linear model (or controller) will always be sufficiently accurate locally, provided that the operating regime is sufficiently small, even though the system may contain complex nonlinearities when viewed globally. In these methods, the global dynamics is approximated by a preset number of local linear models that need to be specified by the user. The added difficulty in local modeling is the switching among models, but recently the approximation properties of multiple models have been examined in detail [1].

Under mild conditions, it has been shown that multiple models can uniformly approximate any system on a compact subset provided a sufficient number of local models are given [7]. Finally, with this approach, the model/controller structure is easy to understand and interpret [3].

II. ROTOTCRAFT MODEL

Multicomponent robotic systems can be well described using coordinate systems (frames). Denote $KW = KH0$ the helicopter frame in the stop state (world or base inertial frame), KH the moving frame of the helicopter center of gravity and KS the sensor frame of IMU fixed to the helicopter. For simplicity the vision frame is assumed to be equal to KW , hence the vision system measures the position and orientation of KH relative to KW . The relative position and orientation between frames can be described by a homogeneous transformation T whose orientation and positions parts are A and p , respectively. The graph of the used frames is shown in Fig. 1

$$K_W = K_{H0} \xrightarrow{T_{W,H}} K_H \xrightarrow{T_{H,S}} K_S$$

Figure1. The graph of the important frames

In order to develop the kinematic and dynamic models the results of flight control systems [6] and mini-flying machines [7] can be applied. The position and the orientation of the helicopter with respect to the base frame are denoted by ξ = (*x, y, z)T* and η = (Φ ,Θ ,Ψ)*T* respectively where Φ ,Θ ,Ψ are the Euler angles and the *A*(Φ,Θ,Ψ) orientation matrix is

$$A = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$

The angular velocity can be written in the form ω =Γη where ω =Γη+Γη and

$$\Gamma = \begin{bmatrix} 1 & 0 & -S_\Theta \\ 0 & C_\Phi & C_\Theta S_\Phi \\ 0 & -S_\Phi & C_\Theta C_\Phi \end{bmatrix}$$

The helicopter has 4 actuators, each actuator exerts a lift force proportional with the square of the angular velocity of the rotor ($f_i = b\Omega^2{}_i$) . Each actuator is a brushless DC motor with own controller whose reference signal can be programmed in Ω_i [2]. The resulting lift force (*u*) and driving torque (τ) are defined by

$$u = f_1 + f_2 + f_3 + f_4 = b\sum_{i=1}^{4} \Omega_i^2$$

$$\tau = \begin{pmatrix} lb(\Omega_4^2 - \Omega_2^2) \\ lb(\Omega_3^2 - \Omega_1^2) \\ d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{pmatrix}$$

Where *l*,*b*,*d* are helicopter and motor constants. The gyroscopic effect can be modeled as

$$\tau_G = -I_r(\omega \times k)(\Omega_2 + \Omega_4 - \Omega_1 - \Omega_3) = \\ =: -I_r(\omega \times k)\Omega$$

Where I_r is the rotor inertia and *k* is the third unit vector. Denote *I* the inertia matrix of the helicopter then the differential equation of the helicopter is

$$m\ddot{\xi} = AF_{ext} + F_g$$

$$I\ddot{\omega} + \omega \times (I\omega) = \tau_{ext}$$

Where *Fext* and τ *ext* are the external force and torque, respectively, in *KH* and *T Fg* = (0,0,−*mg*) is the gravity force in *KW* . We assume in the sequel that $I = diag(I_x, I_y, I_z)$ and Φ ,Θ ≈ 0 , Γ ≈ I_3 (unite matrix), ω ≈ $(\Phi,\Theta,\Psi)^T$. Hence the differential equation of the helicopter can be written in the simplified form

$$\ddot{x} = (C_\phi S_\theta C_\psi + S_\phi S_\psi)\frac{u}{m} =: u_x \frac{u}{m} \qquad \ddot{\Phi} = \frac{I_y - I_z}{I_y}\dot{\Theta}\dot{\Psi} - \frac{I_r}{I_x}\dot{\Theta}\Omega + \frac{1}{I_x}\tau_x$$

$$\ddot{y} = (C_\phi S_\theta S_\psi - S_\phi C_\psi)\frac{u}{m} =: u_y \frac{u}{m} \qquad \ddot{\Theta} = \frac{I_z - I_x}{I_y}\dot{\Phi}\dot{\Psi} + \frac{I_r}{I_y}\dot{\Phi}\Omega + \frac{1}{I_y}\tau_y$$

$$\ddot{z} = -g + C_\phi C_\theta \frac{u}{m} \qquad \ddot{\Psi} = \frac{I_x - I_y}{I_z}\dot{\Phi}\dot{\Theta} + \frac{1}{I_z}\tau_z$$

From the differential equations the state equations can be easily written down if the state is chosen as

$$x := (x, y, z, \dot{x}, \dot{y}, \dot{z}, \Phi, \Theta, \Psi, \dot{\Phi}, \dot{\Theta}, \dot{\Psi})^T .$$

The back stepping controller has the form shown in Fig. 2. In the block scheme the indices *d* , *m* and *e* refer to the desired, measured and estimated values, respectively. The controller was implemented on Freescale MPC555 processor having floating point instructions.

III. Selection of Operating Regimes With A SOM

Building local mappings in the full operating space is a time and memory consuming process, which led to the natural idea of quantizing the operating regimes and building local mappings in positions given by prototype vectors obtained from running the plant. The training algorithm is simple, robust to missing values, and it is easy to visualize the map. These properties make SOM a prominent tool in data mining [4].
In most of the papers discussing local linear models for system identification, the SOM has been used with a first order expansion around each PE in the output space.

The SOM transforms an incoming signal pattern of arbitrary dimension into a one or two-dimensional discrete map, and performs this transformation adaptively in a topologically ordered fashion [8]. For an autonomous system, it is reasonable to assume that the future behavior of the system can be predicted over some finite interval from a finite number of observations of past outputs.

In contrast, for predictions of the behavior of a non autonomous system, we have to consider two different dynamics: One for the state space and the other for the control input space. Consequently, the most important difference is how to capture the dynamics in the input–output joint space, which is fundamental for identifying the unknown non autonomous system. Several options are possible and we have been investigating them.

First, we tried to find the local models by quantizing the input–output joint space by embedding not only the outputs but also the control inputs using one SOM. This modification is essential because the purpose is to characterize the system dynamics that exist in the input–output joint space.

However, we encountered some difficulties such as normalization of the joint space and large dimensionality of the space involved (many degrees of freedom and large dynamic range of parameters) [2].
As input feature vectors from a training set are presented to the network, unsupervised learning is used to create a topology-preserving (Kohonen) map of the input data while, at the same time, supervised learning is used to associate an appropriate output feature vector with each PE on the map.

Since the output at each PE is just the average output for all of the feature vectors that map to that point, local models might be created for better approximation using the quantization error in the input space and the average output.

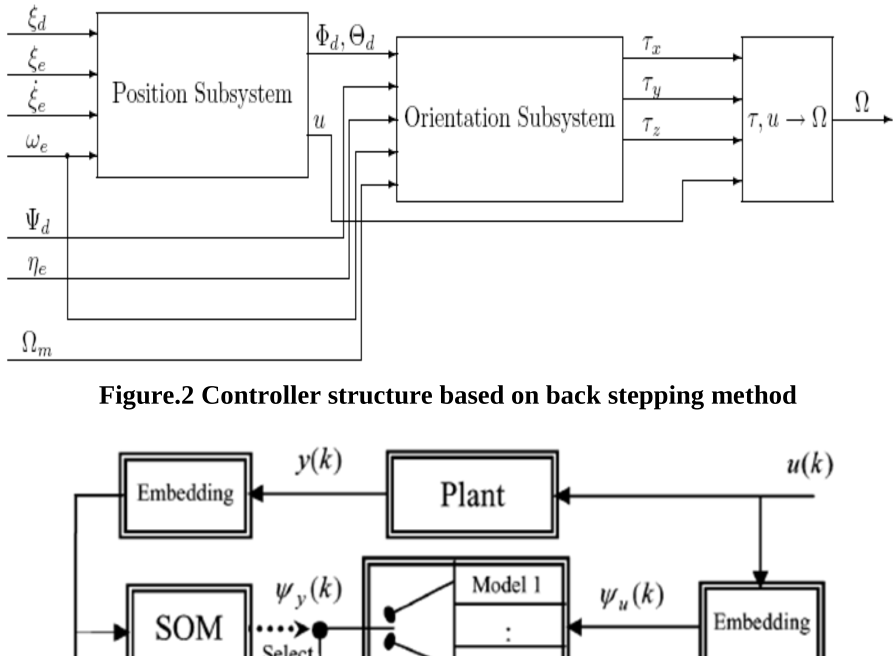

Figure.2 Controller structure based on back stepping method

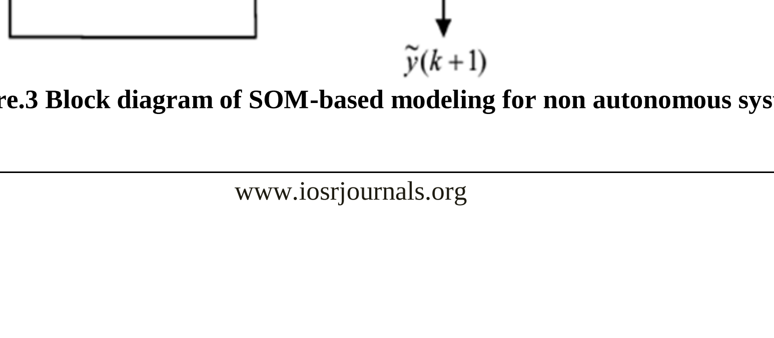

Figure.3 Block diagram of SOM-based modeling for non autonomous system.

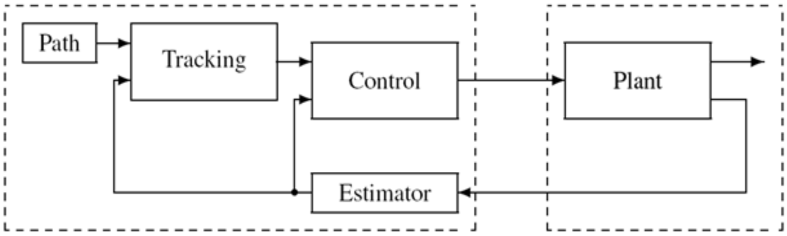

Figure.4 The simplified scheme of the distributed system

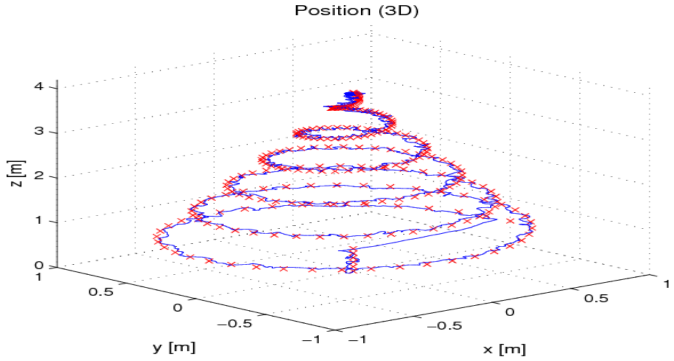

Figure.5 Test results during spiral-formed path

IV. EMBEDDED DESIGN

An embedded system is a special-purpose computer system designed to perform some dedicated functions, often with real-time constraints. It is usually embedded as part of a complete device, in our case the indoor UAV, including hardware and mechanical parts. Embedded controller of a rotorcraft integrates microcontrollers, actuators, vision and inertial sensors. Earlier small indoor UAVs were realized using 8-bit microcontrollers, limited degree of freedom gyroscope and accelerometer (MEMS), serial servo controllers, R/C transmitter and receiver and lightweight rechargeable battery allowing approximately 15 minutes flight duration [8].

The development of the control algorithm has been carried out in MATLAB Simulink environment, since code generation to the target processor can be done in a convenient way with its additional components (Real-Time Workshop and Target Language Compiler).

In order to validate the control algorithm, i.e. check if it can operate at the desired frequency, hardware-in-the-loop tests were performed that involved the emulation of the Rotorcraft helicopter's model and the flow of the sensory data.

V. SOFTWARE REALIZATION

Since the backstepping based algorithm realizes a point-to-point control, the rotorcraft is able to follow a path defined by a series of points in the Cartesian coordinate system using an additional path tracking algorithm. The tracking algorithm is responsible for providing the control algorithm with the actual reference signals (3D position coordinates and yaw angle) and for keeping the helicopter in continuous motion. Sudden changes in the reference signals may cause numerical instability, therefore the reference signals are smoothed by third order filters, which ensure that the second derivatives of the state variables remain smooth.

Due to the high complexity of the control algorithm the planned 10 ms sampling time couldn't be realized, 30 ms was used during the tests instead.

VI. TEST RESULTS

Several tests have been performed to investigate the capabilities of the proposed control algorithm for paths of different kind including maneuvers in horizontal plane consisting of straight lines and special spiral-formed paths.

Fig. 5. shows the results of a test of the latter type of path. The crosses show the waypoint coordinates while the helicopter's trajectory is signed with continuous line.

VII. CONCLUSION

In this paper the theoretical foundations and the real-time realization of the embedded control system of a Rotorcraft (UAV) were presented. The control law is of backstepping character, the state estimation is based on two stage extended Kalman filter. The hardware-in-the-loop real-time test was presented which emulated the helicopter and the sensory system of the Rotorcraft during the test. Communication happened using CAN protocol among the system components.

REFERENCES

[1] S. Bouabdallah and R. Siegwart, "Backstepping and Sliding-Mode Techniques Applied to an Indoor Micro Quadrotor", Proc. Int. Conf. Robotics and Automation , Barcelona, Spain, pp. 2247-2252, 2005.

[2] D. M. Wolpert and M. Kawato, "Multiple paired forward and inverse models for motor control," *Neural Netw.*, vol. 11, pp. 1317–1329, 1998.

[3] T.Kohonen, *Self-Organizing Maps*. NewYork: Springer-Verlag, 1995.

[4] J. Walter, H. Ritter, and K. Schulten, "Nonlinear prediction with selforganizing maps," in Proc. Int. Joint Conf. Neural Networks (IJCNN) , vol. 3, 1990, pp. 589–594.

[5] E. A. Wan and A. A. Bogdanov, "Model predictive neural control with applications to a 6 DoF helicopter model," in Proc. American Control Conf. (ACC), vol. 1, 2001, pp. 488–493.

[6] Sz. Matusik, "Real-Time Position and Orientation Determination Module for the Indoor Work of a Quadrotor Helicopter", MSc Thesis, Budapest University of Technology and Economics, Department of Control Engineering and Information Technology, 2007.

[7] S.D. Hanford, L.N. Long and J.F. Horn, "A Small Semi-Autonomous Rotary-Wing Unmanned Air Vehicle (UAV)", AIAA InfoTech Aerospace Conference, Washington D.C., 2005, AIAA Paper N0. 2005-7077.

[8] T. Madani and A. Benallegue, "Control of Quadrotor Mini-Helicopter via Full State Backstepping Technique", Proc. IEEE Conference on Decision and Control, pp. 1515-1520, 2006.

[9] J. Cho, J. Lan, G. Thampi, J. C. Principe, and M. A. Motter, "Identification of aircraft dynamics using a SOM and local linear models," in Proc. IEEE Int. Midwest Symp. Circuits Systems (MWSCAS), vol. 2, 2002, pp. 148–151.