

Fuzzy Control of an Autonomous Helicopter

B. Kadmiry
Dept. of Computer Sci.
Linköping University
581 83 Linköping, Sweden
bouka@ida.liu.se

D. Driankov
Dept. of Computer Sci.
Linköping University
581 83 Linköping, Sweden
ddr@ida.liu.se

Abstract

This work presents a horizontal velocity controller for the unmanned helicopter APID MK-III by Scandicraft AB in Sweden. We use a novel approach to the design consisting of two steps: first, a Mamdani-type of fuzzy rules compute for each desired horizontal velocity the corresponding desired values for the attitude angles and the main rotor collective pitch; second, a Takagi-Sugeno controller is used to regulate the attitude angles so that the helicopter achieves its desired horizontal velocities at a desired altitude. The performance of the combined linguistic/model-based controller is evaluated in simulation and shows that the proposed design method achieves its intended purpose.

1 Introduction

The Wallenberg Laboratory for Information Technology and Autonomous Systems (WITAS) at Linköping University is involved in the development of a command and control system, supporting the operation of an autonomous helicopter. One platform of choice is the APID MK-III unmanned helicopter, by Scandicraft Systems AB (www.scandicraft.se). The WITAS operational environment is over widely varying geographical terrain with traffic networks and vehicle interaction of variable complexity, speed and density. APID MK-III is capable of autonomous take-off, landing, and hovering as well as of autonomously executing pre-defined, point-to-point flight executed at low-speed. The latter is insufficient since for the above operational environment much higher speed is desired. Thus, our goal is to achieve high-speed motion through stable “aggressive” manoeuvrability at the level of attitude control (pitch, roll, and yaw) and test it on the APID MK-III simulation environment. Here we present a novel design method for horizontal velocity control based on the integration of a linguistic, Mamdani-type of fuzzy controller, and a model-based Takagi-Sugeno (TS) con-

troller. The integrated controller achieves stabilization within much larger attitude angles and horizontal velocities than the ones used in APID MK-III. The approach, as shown in simulation, enables high-speed horizontal motion (in the range of $[-36, 100]$ km/h for backward/forward motion and $[-36, 36]$ km/h for sideward motion) and altitude stabilization. The ranges for the attitude angles that allow us to achieve this are within the intervals $-\pi/4 \leq \phi, \theta \leq \pi/4$, $-\pi \leq \psi \leq \pi$. The design of the integrated controller proceeds as follows:

1. given desired horizontal velocity at certain altitude, a set of Mamdani-type of linguistic rules computes desired attitude angles that help achieve this desired velocity at the given altitude. The rules are heuristic in nature and reflect the experience of a human “pilot” who is an expert in remotely controlling the vehicle;
2. on the basis of Takagi-Sugeno (TS) model for the dynamics of both vertical motion and attitude angles, a TS control laws that achieve the desired attitude angles at the given altitude are designed.

In Sect. 2 we introduce the model of APID MK-III used for attitude/altitude control and the basic underlying assumptions used in its derivation. In Sect. 3 we present the synthesis approach to the design and analysis of the attitude/altitude TS controller. In Sect. 4 we describe the linguistic Mamdani-type of rules used in the derivation of desired set-points for the attitude angles and discuss the intuitions behind them. In Sect. 5 we describe the differences and similarities between our approach and the one by Sugeno et al. in [1]. In Sect. 6 we provide results from simulation that illustrate the performance of the combined Mamdani and TS controllers. Section 7 presents conclusions and directions for future work.

2 The APID MK-III model

The mathematical model used for attitude and altitude control of APID MK-III is of the form (for details see [2]):

$$\begin{aligned}\ddot{z} &= \frac{1}{m}(F_D + F_g - k\omega^2\theta_0 c\phi c\theta), \\ \ddot{\phi} &= -a\dot{\phi} + dk\omega^2\phi_c\theta_0, \\ \ddot{\theta} &= -b\dot{\theta} - ek\omega^2\theta_c\theta_0, \\ \ddot{\psi} &= -c\dot{\psi} + f(\alpha_{tr} - \psi_0),\end{aligned}\quad (1)$$

where $c\phi = \cos\phi$ and $s\phi = \sin\phi$. The state vector is $(x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$, i.e., positions, attitude angles and their respective velocities. The control inputs, are $(\phi_c, \theta_c, \theta_0, \alpha_{tr})$, i.e., these are the usual controls in terms of lateral and longitudinal cyclic pitches, collective pitches for the main and tail rotors. The first three equations describe the dynamics of translational motion in the inertial frame where F_0 are wind forces in north, east, and vertical directions; m is the helicopter body mass, F_g is the gravity force acting on the cabin, and ω is the main rotor RPM. The last three equations describe the dynamics of the rotational motion in the body frame. The coefficients $a = 38.7072$, $b = 10.1815$, $c = 0.434$ are derived from the expression of moment equations leading to the attitude angles' equations ($\psi_0 = 0.09$ is an offset term).

The assumptions underlying the above model are: (i) the variation of the rotor speed ω is constant as a consequence of maintaining constant throttle control at the nominal part of the power curve – the constant value of ω is implicit in the gain $k\omega^2 = 1703.46$, and (ii) the variations of the main rotor angles are small enough so that the magnitude of the main rotor force can be considered equal to the thrust force.

The uncertainty or unmodeled dynamics of the above model can be categorized as follows: (i) unmodeled aerodynamics – only the wind action, e.g., F_N, F_E, F_D on the body is considered, and the action of the tail rotor force on the angular accelerations is neglected; (ii) higher order dynamics such as rotor flapping dynamics is not considered at all, while the usually highly nonlinear link between the control inputs and servos of the main and tail rotors and governing equations are linearized and are implicit in the constant gains $k\omega^2 = 1703.4$, $dk\omega^2 = 223.5824$, $ek\omega^2 = 58.3258$, and $f = 31.9065$; and (iii) servo actuators are linked to the control inputs and are modeled by first-order transfer functions of the form $\dot{\delta} = -300\delta + 300u$ where u is any one of the control inputs and δ is a pseudo state variable.

The current control system for APID MK-III does not utilize large ranges of the rotor atti-

tude angles. As a consequence this produces lower rate-of-change of the attitude angles ϕ , θ and ψ , and consequently the control is done on rather small ranges for these – all this reduces manoeuvrability w.r.t. these angles and consequently the speed of motion. In this context, the objective of our study is to design a horizontal velocity controller which acts on much larger ranges of the attitude angles, i.e., $-\pi/4 \leq \phi \leq +\pi/4$, $-\pi/4 \leq \theta \leq +\pi/4$, $-\pi \leq \psi \leq +\pi$, by utilizing the full range of the rotor attitude angles. The latter, for the purpose of this study, are in the interval $[-0.7, +0.7]$ rad.

3 Takagi-Sugeno controller

First, the nonlinearities in the control inputs of the nonlinear model from Sect. 2 are decoupled by adding first-order actuator transfer functions – as a result, these nonlinearities are moved into the state. The transformed model is then given as:

$$\begin{aligned}\dot{x}_1 &= x_5, \\ \dot{x}_2 &= x_6, \\ \dot{x}_3 &= x_7, \\ \dot{x}_4 &= x_8, \\ \dot{x}_5 &= \frac{1}{m}(F_D + F_g - 1703.4 \cos(x_6) \cos(x_7)x_{11}), \\ \dot{x}_6 &= -38.7072 x_6 + 223.5824 x_9 x_{11}, \\ \dot{x}_7 &= -10.1815 x_7 - 58.3258 x_{10} x_{11}, \\ \dot{x}_8 &= -0.434 x_8 + 31.9065 x_{12} + 0.09, \\ \dot{x}_9 &= -300 x_9 + 300 u_1, \\ \dot{x}_{10} &= -300 x_{10} + 300 u_2, \\ \dot{x}_{11} &= -300 x_{11} + 300 u_3, \\ \dot{x}_{12} &= -300 x_{12} + 300 u_4,\end{aligned}\quad (2)$$

where x_1 and x_5 are the altitude and its velocity; x_2, \dots, x_4 are the attitude angles, and x_6, \dots, x_8 are their angular rates; x_9, \dots, x_{12} are the servo states. Furthermore, u_1, u_2, u_3 , and u_4 are the commanded cyclic pitch and roll together with the main and tail rotor collective pitches. Note that $\phi_c, \theta_c, \theta_0$ and α_{tr} now are state variables.

The above model is transformed into a TS model by a novel technique called *linear bounding* which not only approximates this nonlinear model exactly, but also drastically reduces the number of linear sub-models that constitute the TS model. In what follows we describe briefly this step of the design.

Consider again the model described in (2). The nonlinear terms to be linearized, so that the fuzzy system represents exactly the nonlinear

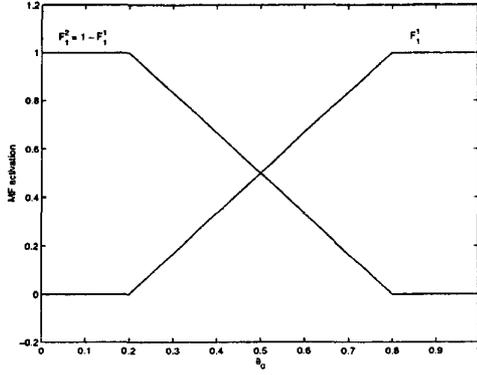


Figure 1: Membership functions for x_{11} .

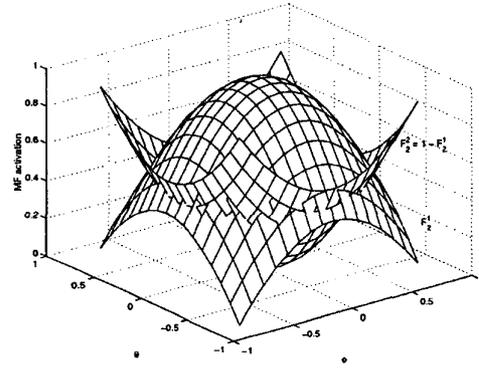


Figure 2: Membership functions for $[x_6, x_7]$.

system (2), are x_9x_{11} and $x_{10}x_{11}$ to be the terms to linearize in the attitude equations (\dot{x}_6 and \dot{x}_7), and $\cos(x_6)\cos(x_7)x_{11}$ to be linearized in the \dot{x}_5 equation. The state variables involved in these terms satisfy: $x_6, x_7 \in [-\pi/4, \pi/4]$ and $x_{11} \in [\pi/18, 5\pi/18]$. Furthermore, x_{11} is trivially bounded by: $.1745 < x_{11} < .8727$. Then $\cos(x_6)\cos(x_7)$, taking into account the bounds from above, can be bounded by two constant functions: $0.7071 < \cos(x_6) < 1$ and $0.7071 < \cos(x_7) < 1$, which gives $0.5 < \cos(x_6)\cos(x_7) < 1$.

Now the three nonlinear terms can be fully described by the upper and lower bounds derived above in the following manner:

$$x_9x_{11} = F_1^1 0.8727x_9 + F_1^2 0.1745x_9$$

$$x_{10}x_{11} = F_1^1 0.8727x_{10} + F_1^2 0.1745x_{10}$$

$$\cos(x_6)\cos(x_7)x_{11} = F_2^1 x_{11} + F_2^2 0.5x_{11},$$

where $F_1^1, F_2^1 \in [0, 1]$, $F_1^2 = 1 - F_1^1$ and $F_2^2 = 1 - F_2^1$. By solving the above equations for F_1^1, F_1^2, F_2^1 and F_2^2 we obtain the following membership functions

$$\begin{aligned} F_1^1 &= (x_{11} - 0.1745)/0.6981 \\ F_1^2 &= (0.8727 - x_{11})/0.6981 \\ F_2^1 &= 2 \cos(x_6) \cos(x_7) - 1 \\ F_2^2 &= 2 - 2 \cos(x_6) \cos(x_7) \end{aligned}$$

The graphs of the membership functions F_1^1 and F_2^1 are shown in Fig. 1 and the graphs of F_2^2 and F_1^2 in Fig. 2.

The TS rule base is then expressed as follows:

- 1 : IF x_{11} IS F_1^1 and $[x_6, x_7]$ IS F_2^1
THEN $\dot{x} = A_1x + Bu$
- 2 : IF x_{11} IS F_1^1 and $[x_6, x_7]$ IS F_2^2
THEN $\dot{x} = A_2x + Bu$
- 3 : IF x_{11} IS F_1^2 and $[x_6, x_7]$ IS F_2^1
THEN $\dot{x} = A_3x + Bu$
- 4 : IF x_{11} IS F_1^2 and $[x_6, x_7]$ IS F_2^2
THEN $\dot{x} = A_4x + Bu$

In the above rules the matrix A_i is the Jacobian obtained by Taylor series expansion of (2) for values of x_6, x_7 , and x_{11} such that $F_1^1(x_{11}) = 1$, and $F_2^1(x_6, x_7) = 1$. The rest of A_2, A_3 , and A_4 are obtained in the same manner. The B matrix is identical for all rules and contains the gains for the servo actuators connected to the control inputs. The global model is then represented as:

$$\dot{x} = \sum_{i=1}^4 w_i(x_6, x_7, x_{11})(A_i x + Bu), \quad (3)$$

where w_i is the total activation for each rule: $w_1 = F_1^1 \cdot F_2^1$, $w_2 = F_1^1 \cdot F_2^2$, $w_3 = F_1^2 \cdot F_2^1$ and $w_4 = F_1^2 \cdot F_2^2$, with $\sum_{i=1}^4 w_i = 1$. Given the TS fuzzy model, we obtain a fuzzy gain scheduled dynamic output feedback \mathcal{H}_∞ controller of the form:

$$\begin{bmatrix} \dot{x}_c \\ u \end{bmatrix} = \sum_{i=1}^4 w_i \begin{bmatrix} A_c^i & B_c^i \\ C_c^i & D_c^i \end{bmatrix} \begin{bmatrix} x_c \\ y \end{bmatrix} \quad (4)$$

using the results in [3] for self-scheduled output feedback controllers. Although this technique only is applicable to linear subsystems we have extended it to affine linear subsystems, see [4] for more details. The controller was designed

to track desired values in altitude and attitude angles. Integral action was introduced to avoid steady state errors. The actuator states are limited to a certain range and this is accounted for in the controller design. The outputs from the system which are fed into the controller are taken to be x_1, \dots, x_8 . The servo state x_{11} must of course also be measured because of its use in the scheduling.

4 The Mamdani-type controller

As already mentioned, the Mamdani-type of linguistic controller is used to generate desired values for attitude angles given desired horizontal velocities at a given altitude. This type of controller has a heuristic nature which reflects the experience of a human "pilot" who is an expert in remotely controlling the vehicle. The motivation for resorting to such a heuristic approach is as follows:

- The available equations describing the dynamics of horizontal motion do not take into account aerodynamic effects related to the main rotor. Also the the contributions of the tail rotor torque and force are neglected. Thus using these equations to derive desired attitude angles, given desired horizontal velocities, is not a reliable option. Instead, the Mamdani-type of linguistic controller uses the magnitude of acceleration and velocity-error to infer attitude angles that if achieved will reduce the velocity error to zero. Thus they "mimic" a human "pilot's" behavior when trying to achieve certain desired velocities via remote control.

In this context, the rules used to compute desired values for pitch are of the form:

IF e_{v_x} is Neg and \dot{e}_{v_x} is Neg **THEN** desired pitch is Pos,

where e_{v_x} is the longitudinal velocity-error and \dot{e}_{v_x} is the longitudinal acceleration. The "heuristic" interpretation of this particular rule is as follows: if the current longitudinal velocity is higher than the desired one and we are accelerating, i.e., we are moving further away from the desired velocity which is caused by a negative pitch angle. In order to bring the current velocity back to the desired one we have to slow down the longitudinal motion and reverse the acceleration. This is done by bringing the pitch from a negative to a positive angle. Furthermore, Neg and Pos are

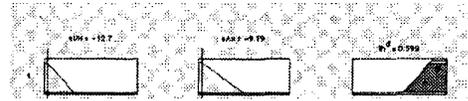


Figure 3: Rule for longitudinal speed with membership functions.



Figure 4: Rule for lateral speed with membership functions.

linguistic labels for the magnitudes of e_{v_x} , \dot{e}_{v_x} , and the pitch. The meaning of these linguistic labels is given by fuzzy sets defined on the physical domains of e_{v_x} , \dot{e}_{v_x} , and the pitch. Figure 3 illustrates the above rule in terms of these membership functions. All in all there are 9 rules describing the relationship between e_{v_x} , \dot{e}_{v_x} and the pitch.

The rules used to compute desired values for roll are of the form:

IF e_{v_y} is Neg and \dot{e}_{v_y} is Neg **THEN** desired roll is Neg,

where e_{v_y} is the lateral velocity-error and \dot{e}_{v_y} is the lateral acceleration. The "heuristic" interpretation of this particular rule is as follows: if the current lateral velocity is higher than the desired one and we are accelerating, i.e., we are moving further away from the desired velocity which is caused by a positive roll angle. In order to bring the current velocity back to the desired one we have to slow down the lateral motion and reverse the acceleration. This is done by bringing the roll from a positive to a negative angle. Furthermore, Neg and Pos are linguistic labels for the magnitudes of e_{v_y} , \dot{e}_{v_y} , and the roll. Figure 4 illustrates the above rule in terms of membership functions corresponding to these linguistic labels. All in all there are 9 rules describing the relationship between e_{v_y} , \dot{e}_{v_y} and the roll.

The desired value for the yaw is computed by rules as:

IF e_χ is Pos and \dot{e}_χ is Neg **THEN** desired yaw is Zero.

where e_χ is the heading-error and \dot{e}_χ is its rate of change. The "heuristic" interpretation of this particular rule is as follows: if the current head-

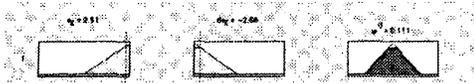


Figure 5: Rule for heading with membership functions.

ing is higher than the desired one and we are reducing it, i.e., we are moving closer to the desired heading which is caused by certain orientation of the horizontal velocity. In this case we maintain the current yaw. Furthermore, Neg, Pos, and Zero are linguistic labels for the magnitudes of e_x , \dot{e}_x , and the current yaw. Figure 5 illustrates the above rule in terms of membership functions corresponding to these linguistic labels. All in all there are 9 rules describing the relationship between e_x , \dot{e}_x and the yaw.

The first two types of rules neglect the cross-couplings between pitch and roll angles in the dynamics of longitudinal and lateral motions. However, these couplings are taken care by the heading rules that in addition also prevent sideslip by restricting the yaw to be always equal to the heading. Furthermore, the pitch and roll angles affect the dynamics of vertical motion so that they cause a drop in altitude. Preventing this is taken care of at the level of the TS controller. The control scheme computing desired attitude angles given desired horizontal velocities at a given altitude is presented in Fig. 6.

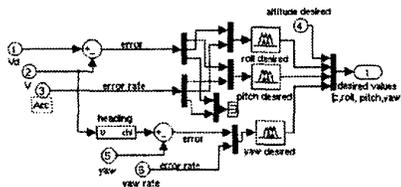


Figure 6: The Mamdani controller.

5 Related work

The work by Sugeno [1] reports a hierarchical, Mamdani-type of a controller for the unmanned helicopter Yamaha R-50 by Yamaha Motors. The lower layer contains a number of Mamdani-type control modules: longitudinal (pitch control), lateral (roll control), collective (vertical control), rudder (yaw control), and coupling compensation modules. Furthermore, within each such module there is a number of sub-modules only some of which correspond di-

rectly to our Mamdani-type controller from Sect. 4. These are as follows:

- Longitudinal: this module includes a \dot{x} Mamdani-type controller. The \dot{x} controller infers a desired pitch angle using a velocity-error and its derivative and is identical to the one used by us. However, in our case the desired pitch angle is sent to the TS controller from Sect. 3 while in Sugeno's case it is sent to yet another controller from the same module;
- Lateral: this module includes a \dot{y} Mamdani-type controller. The \dot{y} controller infers a desired roll angle using a velocity-error and its derivative and is identical to the one used by us. However, in our case the desired roll angle is sent to the TS controller from Sect. 3 while in Sugeno's case it is sent to yet another controller from the same module;
- Collective: this module includes a \dot{z} Mamdani-type controller. The \dot{z} controller infers a control value for the main collective using altitude, velocity-error and its derivative. In our case, the control value for the main collective is obtained by the TS controller given a desired altitude;
- Rudder: this module, given a desired heading, infers a control input for the tail collective using yaw angle error and its rate of change. In our case we infer a desired yaw angle given a desired heading, yaw angle error and its rate of change. The desired yaw angle is sent to the TS controller which in turn gives a control input for the tail collective.
- Coupling compensation: the use of this module is twofold: i) it takes into account cross-couplings between longitudinal/lateral and vertical motion; ii) it takes into account cross-couplings between yaw and roll during a turn. In our case the first type of cross-couplings are taken care of by the TS controller. The second type are partially solved by guaranteeing that a turn is always performed in a directional manner.

6 Simulation results

The integration between the Mamdani and TS controllers is illustrated in Fig. 7.

The numerical experiments are performed with the controllers designed in the previous sections and acting on the nonlinear model from Sect. 3.

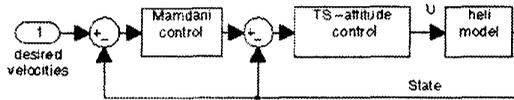


Figure 7: The integrated controller.

The first experiment, depicted in Fig. 8, shows the results from set-point regulation around a desired low and high longitudinal velocities.

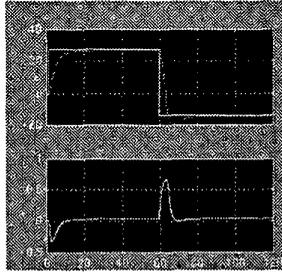


Figure 8: Upper-part: Low and high \dot{x} -velocity set-point regulation. Lower-part: Corresponding desired pitch values.

The second experiment, depicted in Fig. 9, shows the results from set-point regulation around a desired low and high lateral velocities.

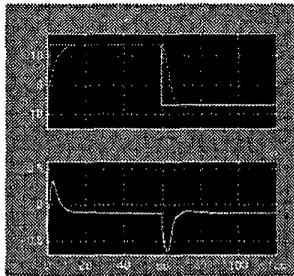


Figure 9: Upper-part: Low and high \dot{y} -velocity set-point regulation. Lower-part: Corresponding desired roll values.

The last experiment, depicted in Fig. 10, shows the results from tracking a desired heading computed from desired horizontal velocities.

7 Conclusions

This work has shown the applicability of our approach, using a combination of linguistic and model-based fuzzy control of an unmanned helicopter. The performance of the controller when evaluated in simulation achieves stabilization of

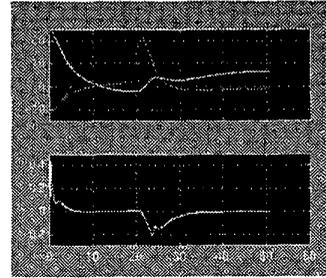


Figure 10: Upper-part: Tracking error for velocities. Lower-part: Corresponding tracking error for heading and yaw.

horizontal high-speed velocities and altitude using attitude angles within much larger ranges than the ones currently available on the APID MK-III platform.

Future work will address the use of the approach presented here for position control and for the purpose of behavior-based helicopter control.

8 Acknowledgements

This work was made possible by a research grant provided by the Wallenberg Foundation in Sweden.

References

- [1] M. Sugeno, Development of an intelligent unmanned helicopter, *Fuzzy Modeling and Control: Selected Works of M. Sugeno*, H.T. Nguyen and N. R. Prasad (eds.), CRC Press, Boca Raton, 1999.
- [2] B. Kadmiry, P. Bergsten, and D. Driankov, Autonomous helicopter control using fuzzy gain scheduling, accepted at ICRA 2001, Seoul, Korea, May 21–26, 2001.
- [3] P. Apkarian, P. Gahinet and G. Becker, Self-Scheduled \mathcal{H}_∞ Control of Linear Parameter Varying Systems: A Design Example, *Automatica*, 31(9), pp 1251–1261, 1995.
- [4] P. Bergsten, M. Persson and B. Iliev, Fuzzy gain scheduling for flight control, *Proceedings IEEE Conference on Industrial Electronics, Control and Instrumentation*, Nagoya, Japan, 2000. Accepted for publication.