TELKOMNIKA, Vol.14, No.3, September 2016, pp. 963~973
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v14i3.3748

H-INFINITY Control for Pitch-Roll AR.DRONE

Agung Prayitno*[1], Veronica Indrawati[2], Clark Arron[3]
Electrical Engineering Department, University of Surabaya (UBAYA),
Jl. Raya Kalirungkut – Surabaya 60293, East Java – Indonesia, Tel.+62-31-2981157
*Corresponding author, e-mail: prayitno_agung@staff.ubaya.ac.id[1], veronica@staff.ubaya.ac.id[2],
clarkarronkesi319@gmail.com[3]

Abstract

This paper describes the design and implementation of H-infinity controller applied to the AR.Drone to follow a given trajectory. The trajectory will be achieved by using two control signals, pitch and roll. Pitch and roll of the AR.Drone models are obtained by assuming that the transfer function of internal control for pitch and roll is the second order system. Two schemes of H-infinity controller designed for pitch and roll. H-infinity control for x-position has exogenous input of the x-reference, xref, control input of pitch value, exogenous output in the form of x-position and process output as error x. While H-infinity control for y-position has exogenous input of y-reference, yref, control input in the form of roll value, exogenous output of y-position and process output as error y. The results of simulation and implementation show that drone can follow multiple references of trajectories given.

Keywords*: AR.Drone control, H-infinity controller, pitch control, roll control*

1. Introduction

Various applications of Unmanned Aerial Vehicle (UAV) have been widely used today for news reporting, disaster missions, expeditions, videography for tourism, businesses and others. UAVs are mostly still controlled by humans from the ground station. The existence of autonomous flight feature will be very helpful when terrain and environment restrict human movement. This paper presents the development of an automated algorithm scheme on one type of UAV, the AR.Drone quadrotor.

AR.Drone has become one of the research platforms that are widely used by researchers, especially for those who focus on the development of algorithms. Algorithm development with AR.Drone platform is faster because AR.Drone is equipped with several sensors: 3 axis gyroscope, 3 axis accelerometer, a sonar altimeter, and the front and bottom cameras. AR.Drone is also equipped with an onboard computer that can be used for basic controls, such as: vertical take off landing, hovering, forward-reverse, right and left maneuvering by giving a value between -1 to 1 on the pitch, roll, yawrate, and vertical rate input. Providing a positive pitch value (+) means ordering the drone to fly backward while a negative pitch value (-) means ordering the drone to fly forward. Positive roll value (+) means ordering the drone to fly right sideward, and left sideward for negative roll value (-). To pivot clockwise motion, positive yawrate value must be given to the drone and negative (-) for the opposite motion. Positive (+) vertical rate value is given to the drone to maneuver vertically upward and negative (-) for reverse maneuver. Range values -1 and 1 are propotional to the minimum and maximum range of the actual value of each input that is set on the configuration of its innerboard. Through Wi-fi communication, control command can be sent from a PC in the ground system to the innerboard of the AR.Drone and vice versa innerboard can send navigation data to the PC. Navigation data that can be taken include actual roll value, sideward speed, actual pitch value, forward speed, actual yaw rate value, yaw value, vertical rate value and altitude value.

Many researchers have been conducting research using the AR.Drone. Pierre-Jean Bristeau, et al., [1] describe in detail the technology used in both hardware and software AR.Drone including the hardware description, vision algorithm, sensor calibration, altitude estimation, velocity estimation, and control architecture. Krajnik, et al., [2] used the measurement data to model the internal control of the AR.Drone into four models: pitch, roll, yaw rate and vertical rate. Michael Mogenson [3] makes the AR.Drone LabVIEW toolkit which

Received April 7, 2016; Revised June 19, 2016; Accepted July 5, 2016

generally consists of the main VI to handle basic control vertical take off landing and fly forward-reverse, right and left maneuver. NavData VI transmits navigation data to PC. Video VI transmits streaming video from two cameras to drone as well as some supporting VI. Many control schemes have been developed by researchers. Emad Abbasi, et al., [4] simulated two control schemes, PID controller and fuzzyPID, to control the height of the quadrotor in turbulence situation. Santos, et al., [5] proposed fuzzy logic to control each of the four rotors using the height, roll, pitch and yaw value as inputs. Sarah Yifang [6] applied some control algorithm such as PID controller, waypoint navigation, trajectory tracking and vision-based controller for a variety of flight formation. Rabah Abbas, et al., [7] implemented the leader-follower scheme using PID controller and directed Lyapunov controller at formation tracking quadrotor. Agung, et.al., [8] implemented the algorithm fuzzy logic controller on the AR.Drone by controlling the value of pitch and yawrate. This algorithm is successfully applied to the AR.Drone to the case of multiple trajectories tracking in the x-y given. Veronica, et al., [9] successfully implemented fuzzy logic controller on drones for waypoint navigation in the field of x-y-z. Some fuzzy logic control schemes were tested in this research using a control signal pitch, roll, and vertical rate. Veronica also compared this method with control scheme used by Agung [10].

Problems experienced by [8-10] is when the drone improve the position x with a pitch resulted in a worsening of the position y, and vice versa. In this paper, H-infinity control scheme will be implemented on the AR.drone to follow a predetermined reference in the field of x-y. By using this scheme is expected to obtain the best compromise between x and y position of the drone. The main contribution of this paper is to show that the H-infinity control scheme can be used to control the position of the AR.Drone quadrotor.

Study literature of H-infinity control scheme is obtained from some researchers. Kruczek, et al., [11] implemented the H-infinity control to the active suspension using linear electric motor. The result showed better results than the passive suspension. Guilherme et.al combined predictive integral methodology to handle the reference trajectory and nonlinear Hinfinity control to stabilize the rotational movement of quadrotor. They also use a nonlinear H-infinity control scheme on quadrotor helicopter that has been modified in order to obtain coupling between longitudinal and lateral movement with the roll and pitch motions to follow a given reference path [12, 13]. Keita Mori, et al., [14] using standard H-Infinity control to reduce disturbance sensitivity of the quad-rotor helicopter.They control both the position and the altitude of the quadrotor model using a new input-output linearization method. Taesam Kang, et al., [15] designed a robust H-Infinity attitude controller of linear models quadrotor obtained from the estimated input – output data using Prediction Error Minimation.

Methodology of writing this paper has been preceded by a summary of the authors on the various researchs that have been done in the field of the AR.Drone. Next will be explained a research method that is divided into four sections which include: (1) the augmented plant that explains the H-infinity scheme to be applied; (2) pitch model that explain how to get the pitch model; (3) roll model that describes roll model of drone; (4) K value that is calculated and simulation that explains how to design a controller. At the end of this paper will be presented the result of implementation in a graph then will be analyzed and summarized.

2. Research Method

2.1. The Augmented Plant

To design the H-infinity controller, the AR.Drone control block must be converted into augmented plant. The augmented plant uses two H-infinity control schemes to control x-position and y-position of the AR.Drone separately. The design of the H-infinity control for x-position has exogenous input of the x-reference, x_{ref}, and measurement disturbance, fx, control input of pitch value, exogenous output in the form of x-position and process output as error x. While H-infinity control for y-position has exogenous input of y-reference, y_{ref}, and measurement disturbance, fy, control input in the form of roll value, exogenous output of y-position and process output as error y. In this research, it is assumed that the measurement disturbance on the pitch and roll model is zero while weighting filter and shaping filter is a constant 1. The augmented plant is shown in Figure 1.

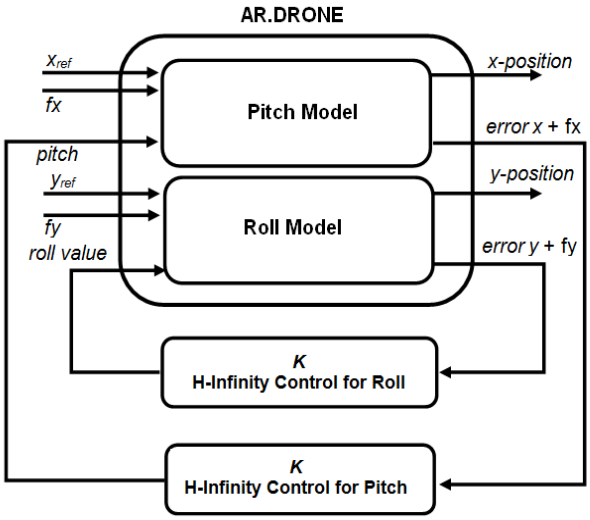


Figure 1. The augmented plant H-infinity control of AR.Drone

2.2. Pitch Model

The next step for the implementation of H-infinity controller on the AR.Drone is to get the mathematical model of the AR.Drone. The flying maneuvers of the AR.Drone in the x-y plane can be done by providing control signals to the pitch and roll. Hence the model of internal control pitch and roll should be known to design the controller

The model of internal control pitch and roll of the AR.Drone was approached by assuming that the transfer function of internal controller is a second order system, as Sarah Yifang [9] has done, where the general form is as follows:

$$G(s) = \frac{\theta(s)}{\theta_{des}(s)} = \frac{k\omega_o^2}{s^2 + 2\varsigma\omega_o s + \omega_o^2} \quad (1)$$

k, ω_o, and ς parameters can be found by identifying the characteristic of transient response and steady state second order system that includes overshoot, rise time, steady state value. The equation used to obtain the proficiency parameters is as follows:

$$\text{Overshoot} = M_p = e^{\frac{-\pi\varsigma}{\sqrt{1-\varsigma^2}}} x100\% \quad (2)$$

$$\text{Rise time} = t_p \approx \frac{1}{\omega_o} e^{\phi} \tan\phi \rightarrow \omega_o = \frac{e^{\phi} \tan\phi}{t_p} \quad (3)$$

$$\text{DC gain} = G(o) = k = \frac{ss\_value}{input\_value} \quad (4)$$

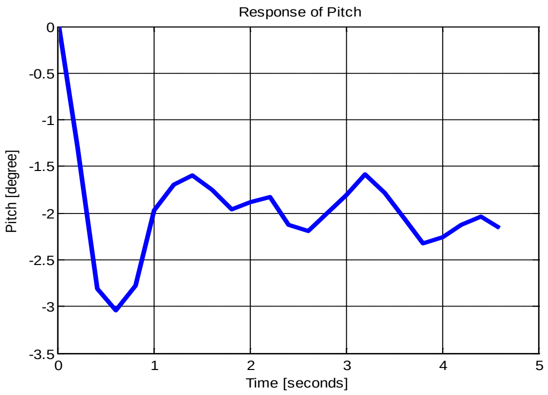
Data for modeling pitch and roll obtained by experimental procedure is as follows:

Pitch: give the desired set point pitch (eg. –0.1 equal with -1^o) and take off the drone with hover-on mode. After the AR.Drone altitude is steady at 1 metre, switch off hover menu so that the drones will move ahead with a particular pitch angle for a few seconds. Then switch on hover and land the drone. The data recorded during the flight are pitch set point (in range of -1 to 1), the measured pitch (θ)[in degrees], forward speed (u)[m/s], and estimated x-position [metre]

Roll: give the desired set point roll (eg. 0.1 equal to 1^o) and take off the drone with hover mode on. After the AR.Drone altitude is steady at 1 metre, switch off hover menu so that the drones will move sideways with a particular roll angle for a few second. Then switch on hover and land the drone. The datas recorded during the flight are roll set point (in range of -1 to 1), the measured roll (θ)[in degrees], sideward speed (u)[m/s], and estimated y-position [metre].

This section will explain the steps to get the pitch model and the state space of the augmented plant for pitch control scheme. H-infinity control scheme to roll have the same steps and will display the results later.

From the experimental result of the step response pitch with a negative value, AR.Drone provides a response as shown in Figure 2. The response was approached with second order order systems and can be calculated transient response specifications, covering %OS, peak time, steady state and DC gain.



- %OS = 55.1463%
- Peak time = 0.6 s
- Steady state = -1.9614°
- $\varsigma = 0.1862$
- $\omega_n = 5.3265$
- *k = 1.9614*

Figure 2. Step response of pitch

Therefore, the transfer function of pitch model according to the equation (1) is:

$$\frac{\theta(s)}{\theta_{des}(s)} = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{55.65}{s^2 + 1.984s + 28.37} \quad (5)$$

Representation of equation (5) in a continuous-time state space is:

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -28.37 & -1.984 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 55.65 \end{bmatrix} \theta_{des} \quad (6)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} \quad (7)$$

Representation in the discrete-time state space with 0.2s sampling time is:

$$\begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}_{t+1} = \begin{bmatrix} 0.545 & 0.1357 \\ -3.849 & 0.2758 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}_t + \begin{bmatrix} 0.8925 \\ 7.55 \end{bmatrix} \theta_{des} \quad (8)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}_t \quad (9)$$

After getting state space pitch, state space of forward velocity, u will be calculated

$$\begin{bmatrix} u \\ \dot{u} \end{bmatrix}_{t+1} = \begin{bmatrix} a & b \\ d & e \end{bmatrix} \begin{bmatrix} u \\ \dot{u} \end{bmatrix}_t + \begin{bmatrix} c \\ f \end{bmatrix} \theta_{t+1} \quad (10)$$

$$y = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} u \\ \dot{u} \end{bmatrix}_t \tag{11}$$

Value of a, b, c, d, e and f are calculated by least square estimation (12) and (13) based on data obtained from the experiment AR.Drone.

$$\boldsymbol{x}_1 = (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{y}_1 \tag{12}$$

$$\boldsymbol{x}_2 = (\boldsymbol{B}^T \boldsymbol{B})^{-1} \boldsymbol{B}^T \boldsymbol{y}_2 \tag{13}$$

$$\boldsymbol{y}_1 = \begin{bmatrix} u_{t+1} \\ \vdots \\ u_{t+1+n} \end{bmatrix} \tag{14}$$

$$\boldsymbol{y}_2 = \begin{bmatrix} \dot{u}_{t+1} \\ \vdots \\ \dot{u}_{t+1+n} \end{bmatrix} \tag{15}$$

$$\boldsymbol{A} = \begin{bmatrix} u_t & \dot{u}_t & \theta_{t+1} \\ \vdots & \vdots & \vdots \\ u_{t+n} & \dot{u}_{t+n} & \theta_{t+1+n} \end{bmatrix} \tag{16}$$

Where

u : forward velocity (metre/second)

$\dot{u}$: forward acceleration (metre/second^2)

θ : actual pitch (degree)

Results of the least square estimation obtained state space are as follows:

$$\begin{bmatrix} u \\ \dot{u} \end{bmatrix}_{t+1} = \begin{bmatrix} 0.9397 & 0.1142 \\ -0.4639 & 0.3052 \end{bmatrix} \begin{bmatrix} u \\ \dot{u} \end{bmatrix}_t + \begin{bmatrix} -0.0197 \\ -0.1499 \end{bmatrix} \theta_{t+1} \tag{17}$$

$$y = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} u \\ \dot{u} \end{bmatrix}_t \tag{18}$$

Furthermore, it is known also equation for x-position and error x based on data from the AR.Drone.

$$x_{t+1} = x_t + u_t \Delta t \tag{19}$$

$$errorx = x_{ref} - x_t \tag{20}$$

Next, the equation 8, 9, 17, 18, 19, and 20 will be combined to form a state space that represents the pitch AR.Drone system into the equations (21) and (22).

$$\begin{bmatrix} \theta \\ \dot{\theta} \\ u \\ \dot{u} \\ x \\ errorx \end{bmatrix}_{t+1} = \begin{bmatrix} 0.545 & 0.1357 & 0 & 0 & 0 & 0 \\ -3.849 & 0.2758 & 0 & 0 & 0 & 0 \\ -0.0107 & -0.0027 & 0.9397 & 0.1142 & 0 & 0 \\ -0.0817 & -0.0203 & -0.4639 & 0.3052 & 0 & 0 \\ 0 & 0 & 0.2 & 0 & 1 & 0 \\ 0 & 0 & -0.2 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ u \\ \dot{u} \\ x \\ errorx \end{bmatrix}_t + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_{ref} \\ f_x \end{bmatrix} + \begin{bmatrix} 0.8925 \\ 7.55 \\ -0.0176 \\ -0.1338 \\ 0 \\ 0 \end{bmatrix} \theta_{des} \tag{21}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ u \\ \dot{u} \\ x \\ errorx \end{bmatrix}_t + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ref} \\ f_x \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \theta_{des} \tag{22}$$

And then it is converted back into continuous-time state space form.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{u} \\ \ddot{u} \\ \dot{x} \\ err\dot{o}r\ x \end{bmatrix} = \begin{bmatrix} 0.0006546 & 1 & 0 & 0 & 0 & 0 \\ -28.37 & -1.984 & 0 & 0 & 0 & 0 \\ -0.002709 & -0.00452 & -0.03731 & 0.958 & 0 & 0 \\ -1.191 & -0.09954 & -3.892 & -5.36 & 0 & 0 \\ -0.003059 & 0.0001403 & 0.9915 & -0.1128 & 0 & 0 \\ 0.003059 & -0.0001403 & -0.9915 & 0.1128 & -80.59 & -80.59 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ u \\ \dot{u} \\ x \\ error\ x \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 80.59 & 0 \end{bmatrix} \begin{bmatrix} x_{ref} \\ fx \end{bmatrix} + \begin{bmatrix} -0.001305 \\ 55.65 \\ -0.02015 \\ -0.1318 \\ 0.0008007 \\ -0.0008007 \end{bmatrix} \theta_{des} \tag{23}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ u \\ \dot{u} \\ x \\ errorx \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ref} \\ f_x \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \theta_{des} \tag{24}$$

2.3. Roll Model

In a similar way, the roll model is obtained in the form of transfer function and state space as follows:

$$\frac{\varphi(s)}{\varphi_{des}(s)} = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{68.28}{s^2 + 3.235s + 30} \tag{25}$$

Where

φ : actual roll (degree)

φ_{des} : roll input value to the drone (degree)

Furthermore, it is converted into continuous-time state space form (26), (27) and discrete-time state space with sampling time (Δt) 0.2 s (28), (29).

$$\begin{bmatrix} \dot{\varphi} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -30 & -3.235 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ 68.28 \end{bmatrix} \varphi_{des} \tag{26}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \end{bmatrix} \tag{27}$$

$$\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix}_{t+1}=\begin{bmatrix}0.558 & 0.1197\\ -3.591 & 0.1686\end{bmatrix}\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix}_t+\begin{bmatrix}1.011\\ 8.174\end{bmatrix}\varphi_{des} \tag{28}$$

$$y=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix}_t \tag{29}$$

Where
φ : actual roll (degree)
$\dot{\varphi}$: sideward speed (degree/second)

And it will be obtained the state space sideward velocity and sideward acceleration as follows:

$$\begin{bmatrix}v\\ \dot{v}\end{bmatrix}_{t+1}=\begin{bmatrix}0.9195 & 0.0814\\ -0.394 & 0.5123\end{bmatrix}\begin{bmatrix}v\\ \dot{v}\end{bmatrix}_t+\begin{bmatrix}0.0245\\ 0.1082\end{bmatrix}\varphi_{t+1} \tag{30}$$

$$y=\begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}v\\ \dot{v}\end{bmatrix}_t \tag{31}$$

Where
v : sideward velocity (metre/second)
$\dot{v}$: sideward acceleration (metre/second^2)

Equation for y-position and error y based on data from the AR.Drone.

$$y_{t+1}=y_t+v_t\Delta t \tag{32}$$

$$errory=y_{ref}-y_t \tag{33}$$

Furthermore, equations (28), (29), (30), (31), (32), and (33) are combined to form a state space which represents a model of the AR.Drone roll system.

$$\begin{bmatrix}\varphi\\ \dot{\varphi}\\ v\\ \dot{v}\\ y\\ errory\end{bmatrix}_{t+1}=\begin{bmatrix}0.5558 & 0.1197 & 0 & 0 & 0 & 0\\ -3.591 & 0.1686 & 0 & 0 & 0 & 0\\ 0.0136 & 0.0029 & 0.9195 & 0.0814 & 0 & 0\\ 0.0601 & 0.013 & -0.394 & 0.5123 & 0 & 0\\ 0 & 0 & 0.2 & 0 & 1 & 0\\ 0 & 0 & -0.2 & 0 & -1 & 0\end{bmatrix}\begin{bmatrix}\varphi\\ \dot{\varphi}\\ v\\ \dot{v}\\ y\\ errory\end{bmatrix}_t+\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}y_{ref}\\ f_y\end{bmatrix}+\begin{bmatrix}1.011\\ 8.174\\ 0.0248\\ 0.1094\\ 0\\ 0\end{bmatrix}\varphi_{des} \tag{34}$$

$$y=\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\varphi\\ \dot{\varphi}\\ v\\ \dot{v}\\ y\\ errory\end{bmatrix}_t+\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}y_{ref}\\ f_y\end{bmatrix}+\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}\varphi_{des} \tag{35}$$

And then it is converted back into continuous-time state space form.

$$\begin{bmatrix} \dot{\varphi} \\ \ddot{\varphi} \\ \dot{v} \\ \ddot{v} \\ \dot{y} \\ error\dot{} y \end{bmatrix} = \begin{bmatrix} -0.0005242 & 0.9999 & 0 & 0 & 0 & 0 \\ -30 & -3.235 & 0 & 0 & 0 & 0 \\ 0.07753 & 0.008759 & -0.2865 & 0.572 & 0 & 0 \\ 0.7104 & 0.06255 & -2.769 & -3.148 & 0 & 0 \\ -0.007378 & -0.0005951 & 1.024 & -0.06372 & 0 & 0 \\ 0.007378 & 0.0005951 & -1.024 & 0.06372 & -80.59 & -80.59 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \\ v \\ \dot{v} \\ y \\ error\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 80.59 & 0 \end{bmatrix} \begin{bmatrix} y_{ref} \\ fy \end{bmatrix} + \begin{bmatrix} 0.0007055 \\ 68.28 \\ 0.03046 \\ 0.1214 \\ -0.0008061 \\ 0.0008061 \end{bmatrix} \varphi_{des} \quad (36)$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \\ v \\ \dot{v} \\ y \\ errory \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{ref} \\ f_y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \varphi_{des} \quad (37)$$

2.4. Calculating *K* value

With the pitch and roll model obtained, the next step is to calculate the value of controller K for each scheme, pitch and roll. These calculations use *function hinfsyn* in Matlab. In general, matlab calculation is performed with the following command:

```
ncont = 1; % number of controlled variable
nmeas = 1; % number of measured variable
[K,CL,GAM] = hinfsyn(SystemPitch,nmeas,ncont) %calculating K
[a,b,c,d] = ssdata(K);% matrixs of A, B, C, D from K
[numPitchK,denPitchK] = ss2tf(a,b,c,d); % convert state space to transfer function
systemppitchK = tf(numPitchK,denPitchK); % transfer function of K
```

The value of K for pitch is obtained.

$$K_{Pitch} = \frac{-5.076e04s^4 - 3.135e07s^3 - 1.76e08s^2 - 1.669e08s - 3.708e07}{s^6 + 1063s^5 + 5.587e05s^4 + 5.388e07s^3 + 2.91e08s^2 + 2.782e08s + 6.469e07} \quad (38)$$

Using the same procedure, the value of K for roll is obtained.

$$K_{Roll} = \frac{6.897e04s^4 + 3.137e07s^3 + 1.151e08s^2 + 1.059e08s + 2.27e07}{s^6 + 1163s^5 + 6.696e05s^4 + 6.471e07s^3 + 2.32e08s^2 + 2.118e08s + 4.503e07} \quad (39)$$

2.5. Implementation

To implement the H-infinity controller, the transfer function K for each pitch and roll is converted into discrete-time transfer function with the sampling time (Δt) 0.2 s which is then converted into discrete-time equation with the following step:

$$K_{\theta} = \frac{-0.6101z^5 + 1.338z^4 - 0.9279z^3 + 0.1968z^2 + 8.233e^{-13}z + 3.928e^{-29}}{z^6 - 2.189z^5 + 1.517z^4 - 0.3214z^3 + 4.051e^{-11}z^2 - 1.48e^{-27}z + 1.468e^{-44}} = \frac{\theta}{ex} \quad (40)$$

$$\theta_t = -0.6101ex_{t-1} + 1.338ex_{t-2} - 0.9279ex_{t-3} + 0.1968ex_{t-4} + 8.233e^{-13}ex_{t-5} + 3.928e^{-29}ex_{t-6} + 2.189\theta_{t-1} - 1.517\theta_{t-2} + 0.3214\theta_{t-3} - 4.051e^{-11}\theta_{t-4} + 1.48e^{-27}\theta_{t-5} - 1.468e^{-44}\theta_{t-6} \quad (41)$$

$$K_\varphi = \frac{0.5z^5 - 1.19z^4 + 0.9313z^3 - 0.239z^2 + 2.18e^{-12}z - 3.889e^{-29}}{z^6 - 2.381z^5 + 1.863z^4 - 0.4781z^3 + 6.027e^{-11}z^2 + 9.88e^{-28}z - 2.792e^{-45}} = \frac{\varphi}{ey} \quad (42)$$

$$\varphi_t = 0.5ey_{t-1} - 1.19ey_{t-2} + 0.9313ey_{t-3} - 0.239ey_{t-4} + 2.18e^{-12}ey_{t-5} - 3.899e^{-29}ey_{t-6} + 2.381\varphi_{t-1} - 1.863\varphi_{t-2} + 0.4781\varphi_{t-3} - 6.027e^{-11}\varphi_{t-4} - 9.88e^{-28}\varphi_{t-5} + 2.792e^{-45}\varphi_{t-6} \quad (43)$$

Where:
K_θ : *discreate-time transfer function* K for *pitch*
$K\varphi$: *discreate-time transfer function* K for *roll*
ex_i : *error x* for time-*i*
ey_i : *error y* for time-*i*
θ_i : control signal *pitch* for time-*i*
φ_i : control signal *roll* for time-*i*

The implementation becomes easier with LabVIEW just as shown in subVI Figure 3:

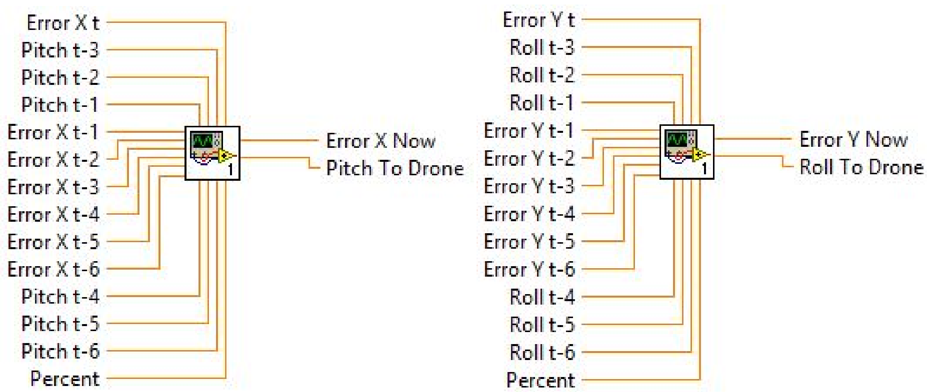


Figure 3. SubVI pitch control and roll control

3. Results and Discussion

The algorithms are implemented on the AR.Drone, then tested to fly automatically following the reference given indoor. Due to the limitation of length width and height of the room, pitch and roll control signal are restricted in the range ± 0.05 and use 30% of generated signal control.

Flying automatic test procedures are as follwos:

1. Track is used as reference inserted through a front panel that has been created
2. Drone flown to hover at 1 metre height with hover menu on the front panel
3. When hover mode is switch off, the drones will fly automatically following reference given
4. During flying automatically, x-position y-position data and other necessary data are recorded
5. After completing its work, drone will be hover switched back and landed
6. The data obtained will be plotted and analyzed

The first test was carried out on a straight path forward. Tests are carried out five times, and the results are shown in Figure 4 below.

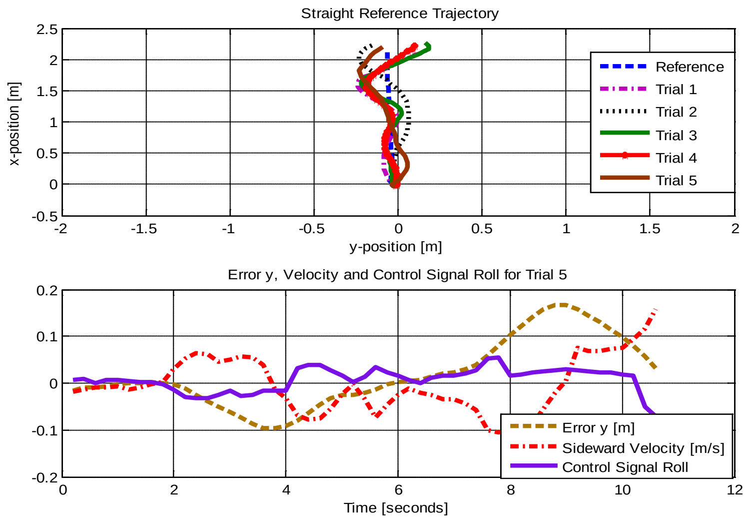


Figure 4. Experiment with Straight trajectory

The test results show that in general, the controller can do the task well, but it can be seen that the drone is oscillated around the y-axis. From the analysis of error y, sideward velocity and roll control signal is always seen to be late in anticipating error y that result in oscillation. When the error-y close to zero then the control signal generated is also close to zero, but the control signal is not sufficient to reduce the speed of the AR.Drone sideward so AR.Drone will deviate from the reference y. This problem causes the drone to oscillate around the reference y.

The next test is to provide a reference in the curve trajectory. It is also carried out five times and the results are shown in Figure 5. The test results show that the drones can approach the reference given yet are late when trying to turn following the curve. The control signal is late in responding the sensitive drone movement.

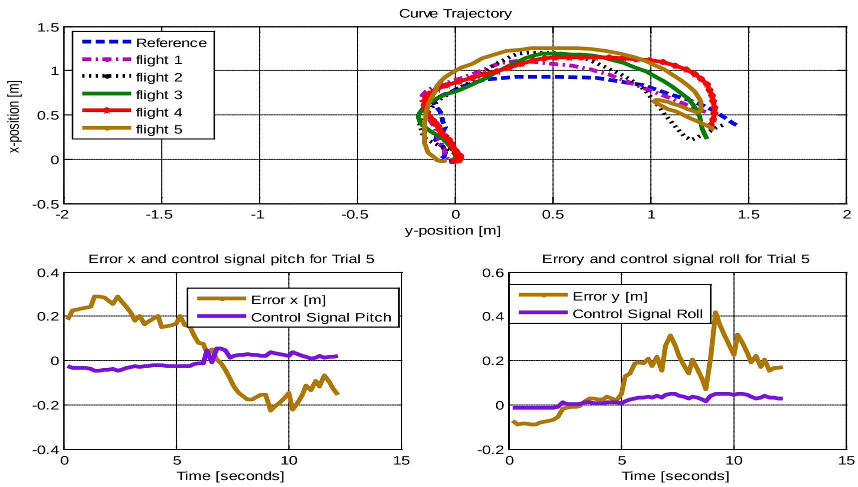


Figure 5. Experiment with Curve Trajectory

The last test is conducted on tracks with sharp turns and a box shape trajectory. Tests were also done five times and the results are shown in Figure 6.

It is seen that the drone had trouble turning on the track that has sharp curves and a box shape trajectory. Drone always seem to have oscillated in each inflection point toward the next track.

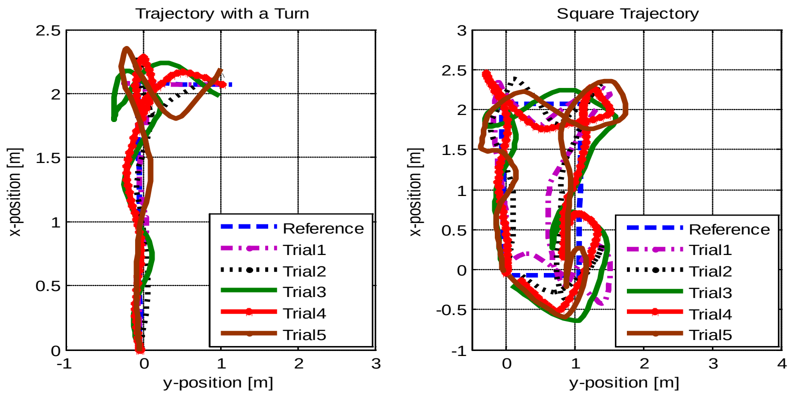


Figure 6. Experiment with sharp turn and box shape Trajectory

4. Conclusion

Generally, H-infinity control for pitch and roll can be used to control the AR.Drone to fly autonomously following various path given. The test results indicate that the controller still has a weakness when there is a sharp turn on the track, and the drone is very sensitive to small changes in control signal. To resolve this problem, use the sideward and forward velocity of drones as inputs of controller might give better results.

References

[1] Pierre-Jean B, Francois C, David V, Nicolas P. *The Navigation and Control Technology Inside the AR.Drone Micro UAV*. 18th IFAC World Congress. Milano, Italy. 2011.

[2] Krajnik T, Vonasek V, Fiser D, Faigl J. *AR-Drone as a Platform for Robotic Research and Education*. Research and Eductation in Robotics: EUROBOT. Heidelberg. 2011.

[3] Michael M. The AR.Drone LabVIEW Toolkit: A Software Framework for the Control of Low Cost Quadrotor Aerial Robots. Master of Science Thesis. TUFTS University; 2012.

[4] Emad Abbasi Seidabad, Saeed Vandaki, Ali Vahidin Kamyad. Designing Fuzzy PID Controller for Quadrotor. *International Journal of Advanced Research in Computer Science & Technology (IJARCST)*. 2014; 2(4): 221-227.

[5] Matilde Santos, Victoria Lopez, Franciso Morata. *Intelligent Fuzzy Controller of a Quadrotor*. IEEE Intelligent Systems and Knowledge Engineering Conf (ISKE). 2010.

[6] Agung Prayitno, Veronica Indrawati, Gabriel Utomo. Trajectory Tracking of AR.Drone Quadrotor Using Fuzzy Logic Controller. *Journal Telkomnika.* 2014; 12(4): 819-828.

[7] Veronica Indrawati, Veronica Indrawati, Thomas Ardi Kusuma. Waypoint Navigation of AR.Drone Quadrotor Using Fuzzy Logic Controller. *Journal Telkomnika.* 2015; 13(3): 930-939.

[8] V Indrawati, A Prayitno, G Utomo. *Comparison of Two Fuzzy Logic Controller Schemes for Position Control of AR.Drone.*7th International Conference on Information Technology and Electrical Engineering (ICITEE). Chiangmai, Thailand. 2015.

[9] Sarah Yifang Tang. Vision-Based Control for Autonomous Quadrotor. Final Report: Undergraduated Senior Thesis. Department of Mechanical and Aerospace Engineering. Princeton University; 2013.

[10] Rabah Abbas, Qinghe Wu. Improved Leader Follower Formation Control for Multiple Quadrotors Based AFSA. *Journal Telkomnika*. 2015; 13(1): 85-92.

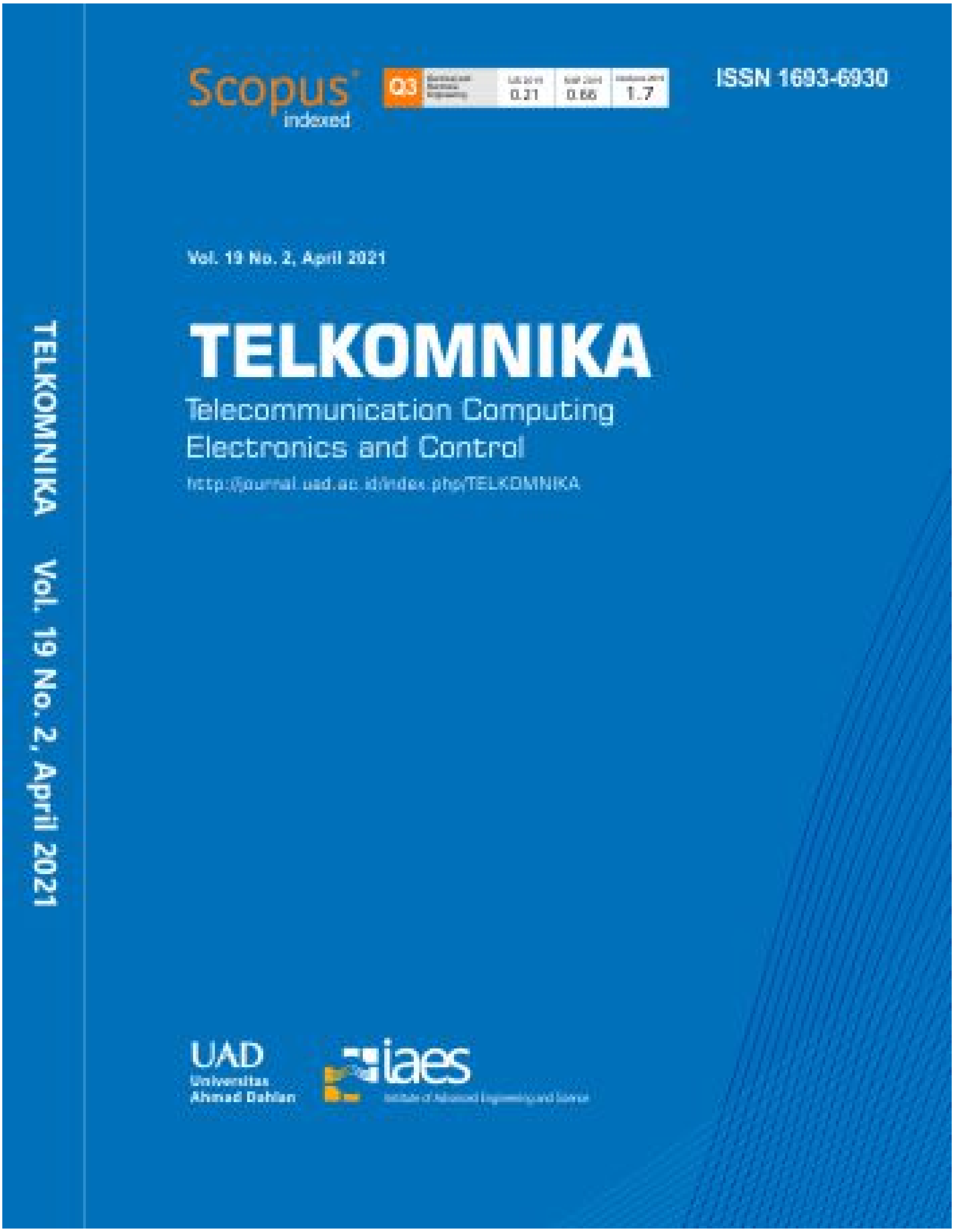
[11] Kruczek A, Stribrsky A. *H∞ Control of Automotive Active Suspension with Linear Motor*. Proceedings of the 3rd IFAC Symposium on Mechatronic Systems. Sydney, Australia. 2004.

[12] G Raffo, M Ortega, F Rubio. An integral predictive/nonlinear control structure for a quadrotor helicopter. *Automatica*. 2010; 46(1): 29-39.

[13] GV Raffo, MG Ortega, FR Rubio. *Nonlinear H ∞ Controller for the Quad-Rotor Helicopter with Input Coupling*. 18th World Congress. IFAC 2011. Milano, Italy. 2011: 13834-13839.

[14] Keita Mori, Katsuya Hotta, Manabu Yamada. Adaptive H ∞ Control for Quad-rotor Helicopters Based on Input Output Linearization. *Transactions of the Society of Instrument and Control Engineers*. 2014; 50(11): 784-791.

[15] Taesam Kang, Kwang Joon Yoon, Tae-Hyun Ha, Gigun Lee. H-infinity Control System Design for a Quad-rotor. *Journal of Institute of Control*. 2015; 21(1): 14-21.


TELKOMNIKA Vol. 19 No. 2, April 2021
Scopus indexed
Q3
ISSN 1693-6930
Vol. 19 No. 2, April 2021
TELKOMNIKA
Telecommunication Computing
Electronics and Control
http://journal.uad.ac.id/index.php/TELKOMNIKA
UAD
Universitas
Ahmad Dahlan
iaes

Editorial Team

TELKOMNIKA Telecommunication, Computing, Electronics and Control
ISSN: 1693-6930, e-ISSN: 2302-9293
Universitas Ahmad Dahlan, 4th Campus
Jl. Ringroad Selatan, Kragilan, Tamanan, Banguntapan, Bantul, Yogyakarta, Indonesia 55191
Phone: +62 (274) 563515, 511830, 379418, 371120
Fax: +62 274 564604

Vol 14, No 3

September 2016

DOI: http://dx.doi.org/10.12928/telkomnika.v14i3

Table of Contents

TELKOMNIKA Telecommunication, Computing, Electronics and Control
ISSN: 1693-6930, e-ISSN: 2302-9293
Universitas Ahmad Dahlan, 4th Campus
Jl. Ringroad Selatan, Kragilan, Tamanan, Banguntapan, Bantul, Yogyakarta, Indonesia 55191
Phone: +62 (274) 563515, 511830, 379418, 371120
Fax: +62 274 564604

Telkomnika

COUNTRY

Indonesia

Universities and research institutions in Indonesia

SUBJECT AREA AND CATEGORY

Engineering
Electrical and Electronic Engineering

PUBLISHER

Institute of Advanced Engineering and Science (IAES)

H-INDEX

23

PUBLICATION TYPE

Journals

ISSN

16936930, 2087278X

COVERAGE

2011-2020

INFORMATION

Homepage

How to publish in this journal

telkomnika@ee.uad.ac.id

SCOPE

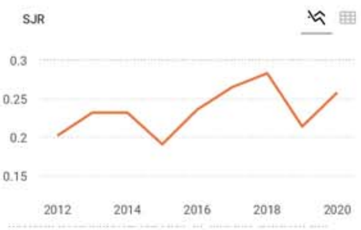
TELKOMNIKA (Telecommunication Computing Electronics and Control) is a peer reviewed International Journal in English published four issues per year (March, June, September and December). The aim of TELKOMNIKA is to publish high-quality articles dedicated to all aspects of the latest outstanding developments in the field of electrical engineering. Its scope encompasses the engineering of signal processing, electrical (power), electronics, instrumentation & control, telecommunication, computing and informatics which covers, but not limited to, the following scope: Signal Processing[...] Electronics[...] Electrical[...] Telecommunication[...] Instrumentation & Control[...] Computing and Informatics[...]

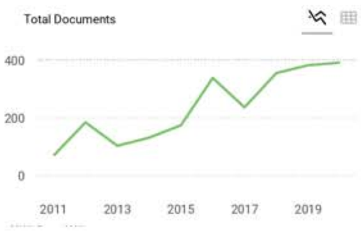
Join the conversation about this journal

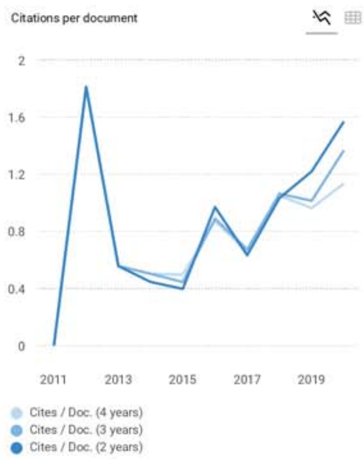
Quartiles

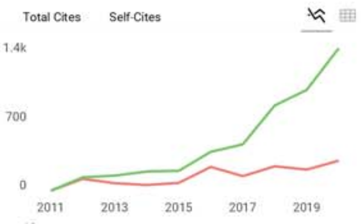
FIND SIMILAR JOURNALS options

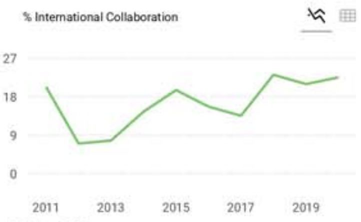
1	2	3	4	5
Indonesian Journal of Electrical Engineering and	**Indonesian Journal of Electrical Engineering and**	**Bulletin of Electrical Engineering and Informatics**	**International Journal of Electrical and Computer**	**Journal of Telecommunication,**
IDN	IDN	IDN	IDN	MYS
48% similarity	46% similarity	44% similarity	41% similarity	22% similarity

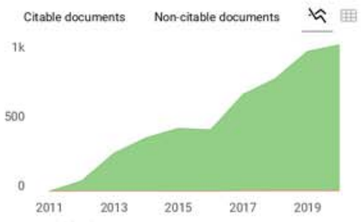


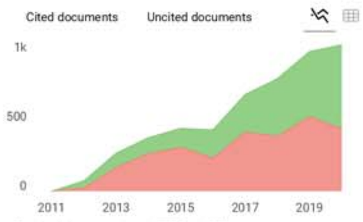












← Show this widget in your own website

Just copy the code below and paste within your html code:

<a href="https://www.scim

SCImago Graphica

Explore, visually communicate and make sense of data with our **new free tool**.

Get it

Metrics based on Scopus® data as of April 2021

J **jancokers** 4 weeks ago

Why is this journal still indexed by Scopus? even though it still contains unqualified journals

reply

SCImago Team

Melanie Ortiz 4 weeks ago

Dear Jancokers, thank you very much for your comment. We suggest you consult the Scopus database directly. Keep in mind that the SJR is a static image (the update is made one time per year) of a database (Scopus) which is changing every day.
Best Regards, SCImago Team

I **Indonesia Author** 4 months ago

Very bad Journal. The editor is very low response. Every email question is never answered by the editor. I really strongly don't recommend submitting on Telkomnika.

reply

SCImago Team

Melanie Ortiz 4 months ago

Dear Sir/Madam, thanks for your participation! Best Regards, SCImago Team

A **ali** 1 year ago

I am really sad because I have been trying to reach this journal in order to assist me in the publication payment process. but unfortunately no response. I have sent them many emails, No feedback.

reply

Melanie Ortiz 1 year ago SCImago Team

Dear Ali,
thank you for contacting us.
Unfortunately, we cannot help you with your request.
Best Regards, SCImago Team

W **Wahiba** 1 year ago

Hello

I have found two journals with the same name, one is from UEA and the second one is from Indonesia is it the same journal, can you send their websites.

reply

Melanie Ortiz 1 year ago SCImago Team

Dear Wahiba,
Thank you for contacting us. Could you please provide us more details about these journals?
Best Regards, SCImago Team

W **Wisnu Ananta Kusuma** 1 year ago

Hello,

Why my all paper published in this journal (telkomnika) before 2017 did not indexed again in scopus?

reply

Melanie Ortiz 1 year ago SCImago Team

Dear Wisnu,
thank you very much for your comment, unfortunately we cannot help you with your request. We suggest you contact Scopus support: https://service.elsevier.com/app/answers/detail/a_id/14883/kw/scimago/supporthub/scopus/
Best Regards, SCImago Team

abdelkbir 2 years ago

what is the Impact Factor for Telkomnika

reply

Melanie Ortiz 2 years ago SCImago Team

Dear Abdelkbir, thank you very much for your comment.
SCImago Journal and Country Rank uses Scopus data, our impact indicator is the SJR. Check out our web to localize the journal. We suggest you to consult the Journal Citation Report for other indicators (like Impact Factor) with a Web of Science data source. Best Regards, SCImago Team

S **Siddhant Karmacharya** 2 years ago

Hi, what is the word limit of this journal ?

reply

Melanie Ortiz 2 years ago SCImago Team

Dear user,
thank you for contacting us.
Sorry to tell you that SCImago Journal & Country Rank is not a journal. SJR is a portal with scientometric indicators of journals indexed in Elsevier/Scopus.
Unfortunately, we cannot help you with your request, we suggest you to visit the journal's homepage or contact the journal's editorial staff , so they could inform you more deeply. You can see the updated journal's information just above .
Best Regards, SCImago Team

mulyadi rusli 2 years ago

pardon me, can I know how long the reviewing process?

reply

SCImago Team

Melanie Ortiz 2 years ago

Dear user,

thank you for contacting us.
Sorry to tell you that SCImago Journal & Country Rank is not a journal. SJR is a portal with scientometric indicators of journals indexed in Elsevier/Scopus.
Unfortunately, we cannot help you with your request, we suggest you to visit the journal's homepage or contact the journal's editorial staff , so they could inform you more deeply.
Best Regards, SCImago Team

M **Muthna** 3 years ago

I want answer, why Journal website appear error? how can check my research status? please inform?

reply

M **muthna** 3 years ago

why journal website close? appear error page 404.

reply

R **Rajni Bhalla** 3 years ago

Hello sir,
My paper suppose to publish in October. But still full text is not coming for this paper.As i have sent word file through mail also and uploaded on website also.Kindly let me know to whom should i contact.I have already sent mail to editor number of times.Kindly do the needful.
Thanking you
Regards
Rajni

reply

M **Mohammed Al-obaidi** 3 years ago

Hi, can I know how long the reviewing process?

SCImago Team

Elena Corera 3 years ago

Dear Rajni,

thank you very much for your comment. Unfortunately, we cannot help you with your request, we suggest you contact journal's editorial staff so they could inform you more deeply. You can find contact information in SJR website https://www.scimagojr.com

Anyway, if there is any user who has already published in the journal, maybe could help us with your request.

Best Regards,
SCImago Team

S **shahd** 4 years ago

I want to Know how I can get the Impact Factor of any Journal

reply

SCImago Team

Elena Corera 4 years ago

Dear Ahahd,

thank you very much for your request. You can consult that information in SJR website.

Best Regards,

SCImago Team

Leave a comment

Name

Email
(will not be published)

I'm not a robot
reCAPTCHA
Privacy - Terms

Submit

The users of Scimago Journal & Country Rank have the possibility to dialogue through comments linked to a specific journal. The purpose is to have a forum in which general doubts about the processes of publication in the journal, experiences and other issues derived from the publication of papers are resolved. For topics on particular articles, maintain the dialogue through the usual channels with your editor.

Source details

Telkomnika (Telecommunication Computing Electronics and Control)

Scopus coverage years: from 2011 to Present

Publisher: Universitas Ahmad Dahlan

ISSN: 1693-6930 E-ISSN: 2087-278X

Subject area: Engineering: Electrical and Electronic Engineering

Source type: Journal

View all documents > Set document alert Save to source list Source Homepage

CiteScore 2020
2.2

SJR 2020
0.258

SNIP 2020
0.746

CiteScore CiteScore rank & trend Scopus content coverage

i Improved CiteScore methodology

CiteScore 2020 counts the citations received in 2017-2020 to articles, reviews, conference papers, book chapters and data papers published in 2017-2020, and divides this by the number of publications published in 2017-2020. Learn more >

CiteScore 2020

2.2 = 2,987 Citations 2017 - 2020 / 1,360 Documents 2017 - 2020

Calculated on 05 May, 2021

CiteScoreTracker 2021

2.8 = 3,927 Citations to date / 1,385 Documents to date

Last updated on 06 March, 2022 • Updated monthly

CiteScore rank 2020

Category	Rank	Percentile
Engineering — Electrical and Electronic Engineering	#373/693	46th

View CiteScore methodology > CiteScore FAQ > Add CiteScore to your site