Assessing The Method of State Space Determination from the Quadrotor Flight Simulation

Jemie Muliadi
Electrical Engineering Dept.
Universitas Indonesia
Depok, Indonesia
jemie.muliadi51@ui.ac.id

Benyamin Kusumoputro
Electrical Engineering Dept.
Universitas Indonesia
Depok, Indonesia
kusumo@ee.ui.ac.id

***Abstract*—This paper addresses the construction of the quadrotor State Space from its flight data. Using numerical simulation, the flight data generated to be processed then obtaining the elements of the State Space matrices. In conventional fashion, one constructs the state space from physical properties measurement such as weight, moments of inertia, vehicle dimension, etc. On the contrary, using the flight data to obtain the State Space omitted the requirement of those physical properties. The method was already described in the authors previous publication, and being assessed in this manuscript. To bring the simulation closer to the real quadrotor dynamics, the complete nonlinear attitude equation of motion applied in the vector form to be solved numerically. The simulated flight data then sampled to mimic the actual measurement process in UAV. The sampled flight data inputted to the coefficient matrix for algebraic inversion to obtain the elements of the UAV state space.**

Keywords—state-space; UAV; quadrotor; system identification; drones

I. INTRODUCTION

Drones have been developed for multi-purpose flying vehicles, from the simple task in video shooting to significant mission such as SAR purposes [1, 2]. Such significance has brought the drones, or widely known as the unmanned aerial vehicles (UAV), to be the popular flying vehicle for research on the flight mechanics sciences, from the field of flight dynamics to flight control systems [3]. Hence, the modeling of flight vehicle's dynamic also developed rapidly to recognize the UAV flight characteristics for the controlling purposes [3, 4]. In another word, to get a good flight control system, one would need the accurate model for UAV motion dynamics.

The modeling method of UAV flight dynamics has evolved into methods [5]. From those variants, the commonly used method is the first principle method which derived from the physics Newton's laws of motion and the Euler's equation of moment. Another method is the identification system that processes the UAV flight data into the parametric or the non-parametric models [6]. The first-principle based modeling is known as the Newton-Euler formalism [7], which can be manipulated and simplified into the state space form [8]. The State Space modeling is a technique of representing a system's dynamic in a set of first-order differential equations in a compact matrix form. Most of the state space technique developed the system's dynamics into a set of linear equations after linearizing the Newton-Euler equation. The linearization performed at a point of equilibrium system [8] so that the resulting State Space represents the small possible deviation of the states from its equilibrium point.

In recent times, the linear control system has grown rapidly resulting varies of methods and control techniques which have adequate reliability [9]. One of the control methods that widely developed in a linear way is the PID control system. Many UAV control systems boards are worked in the double-loop PID structure [10]. Therefore, linear state space becomes a popular technique for modeling UAVs and building its control systems.

Fig. 1. The Flying Quadrotor UAV

This paper assessing the determination of State Space UAV from its flight data. The quadrotor movement is chosen to represent the UAV flight dynamics in a numerical simulation. The flight data from the flight simulation are sampled to be inputted to the State Space determination method developed previously by Muliadi and Kusumoputro in [11]. In the conventional technique, the UAV State Space obtained by calculating the effect of control forces and torques to the physical properties of the vehicle so that the angular and translational angular occurred according to Newton's law. Thus, the conventional state space construction required the information of vehicle's dimension, weight and moment of inertia. However, the moment of inertia is a quantity that can't be measured directly and can only be estimated using tabulating [12, 13], CAD [14, 15] and pendulum method [16]. Although the pendulum method was considered as the most appropriate method for estimating the moment of inertia, the method suffers from unmodeled phenomena while oscillating the UAV, such as the not-centered of the center of gravity position, the strain dynamics, the effect of the mass of the string and rotation about non-principal axis [17]. The accuracy of the estimation also relied on the precision of the measurement besides the dimension and the construction of the pendulum [18]. The estimation error of the pendulum method arises from the exclusion of damping effect in the period measurement, and the high inertia value of the stand/holder which affected to the overall estimation of UAV moments of inertia [16]. To be concluded, the flight data is the most proper method to obtain the state space since all effect of the vehicle interact in such way and characterize the flight data.

To perform the assessment for the method of the State Space determination from UAV flight data, this paper organized as follows. In this first section, the authors have introduced the UAV and its dynamic modeling together with the significance of obtaining the UAV State Space from its flight data. Section Two presents the theoretical basis for a method of the State Space determination. The UAV flight is simulated in Section Three, to be continued with the obtaining of the State Space from the flight simulation data in Section Four. Finally, Section Five resumed the concluding remark of this work.

II. Determining the State Space from UAV Flight Data

In their previous publication, Muliadi and Kusumoputro [11] proposed a direct technique to determining the UAV State Space from its flight data. The method proposed since it is costing reasonable effort to fly the UAV by a practitioner as the test pilot, and then gather the relevant flight data for analysis purposes. To assess the method, this paper presents the flight simulation of a quadrotor to have its data sampled and further processed. Thus, this section resumed the method to prepare the adequate flight simulation in the following section.

The method started with defining the state variables and input variables derived from Newton-Euler equation, so the state space structure can be constructed to accommodate these states and inputs and it can be served as an adequate dynamic model of UAV in the future analysis. The elements of the state space were obtained by rearranging and reconstructing them to separate the desired unknowns in vector-matrix from the measured quantities. These measured quantities can be grouped into a coefficient matrix attached to the vector of the unknowns i.e. the elements of the State Space to be determined. Such construction produced a set of linear equation that solvable using algebraic inversion of the coefficient matrix to obtain all the unknowns simultaneously. To implement the method, the flight data inputted into the coefficient matrix before inversion process, then after the algebraic inversion performed, the elements of the state space obtained without necessary to estimate the moment of inertia of the UAV.

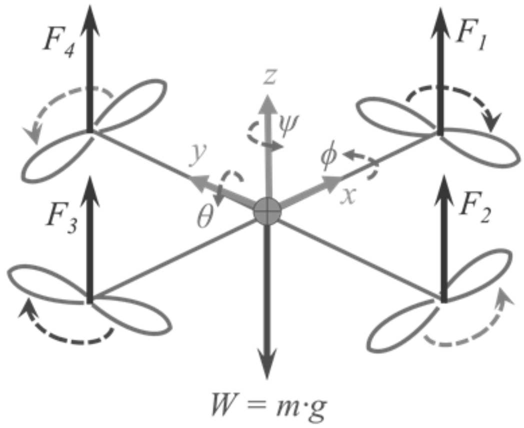

Fig. 2. Quadrotor Flight Convention

In the mathematical expression, the method presented in the following description. Consider the simplified state space form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{1}$$

With $\mathbf{x}$ denote the states vector, $\mathbf{u}$ represent the control input vector, the $\mathbf{A}$ matrix is the system matrix and $\mathbf{B}$ as the input matrix. To accommodates the attitude movement of UAV, then the matrix element composed as follows.

$$\begin{aligned} \mathbf{x} &= \{p \quad q \quad r \quad \phi \quad \theta \quad \psi\}^T \\ \mathbf{u} &= \{L \quad M \quad N\}^T \end{aligned} \tag{2}$$

The state variables are p, q, and r, which stand for the angular rates i.e. roll, pitch and yaw rate in the body axis, while ϕ, θ and ψ, denote the flight angles i.e. roll, pitch and yaw angle with respect to the inertial frame. Hence, their time derivatives denoted as $\dot{p}$, $\dot{q}$, $\dot{r}$,$\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$. The control or input variables enlisted as L, M, and N which represent the control torque in the roll, pitch and yaw direction. Thus, the state space elements treated as the unknowns consist of:

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{16} \\ A_{21} & A_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ A_{61} & \cdots & \cdots & A_{66} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ \vdots & \vdots & \vdots \\ B_{61} & B_{62} & B_{63} \end{bmatrix} \tag{3}$$

From [11], each row of the State Space are solved algebraically by the following relation:

$$\begin{aligned}&\{A_{n1} \quad \cdots \quad A_{n6} \quad B_{n1} \quad B_{n2} \quad B_{n3}\}^T \\ &= H_k^{-1}\left\{\dot{x}_{n_{(k)}} \quad \dot{x}_{n_{(k+1)}} \quad \cdots \quad \dot{x}_{n_{(k+8)}}\right\}\end{aligned} \tag{4}$$

With the integer n in the subscript of the unknown As and Bs showing the row number of the State Space to be obtained, while as the subscript of state x, the n denotes the sequence of the state variable. Hence, x_1 means roll rate p, x_2 means pitch rate q and so on, $x_3 : r$, $x_4 : \phi$, $x_5 : \theta$, and $x_6 : \psi$. The integer k denotes the k-th sampled of the data, and the coefficient matrix with respect to that sample, H_k is the 9x9 matrix:

$$H_k = \left[pqr_{col} \quad \phi\theta\psi_{col} \quad LMN_{col}\right]$$

with:

$$pqr_{col} = \begin{Bmatrix} p_{(k)} & p_{(k+1)} & \cdots & p_{(k+8)} \\ q_{(k)} & q_{(k+1)} & \cdots & q_{(k+8)} \\ r_{(k)} & r_{(k+1)} & \cdots & r_{(k+8)} \end{Bmatrix}^T$$

$$\phi\theta\psi_{col} = \begin{Bmatrix} \phi_{(k)} & \phi_{(k+1)} & \cdots & \phi_{(k+8)} \\ \theta_{(k)} & \theta_{(k+1)} & \cdots & \theta_{(k+8)} \\ \psi_{(k)} & \psi_{(k+1)} & \cdots & \psi_{(k+8)} \end{Bmatrix}^T$$

$$LMN_{col} = \begin{Bmatrix} L_{(k)} & L_{(k+1)} & \cdots & L_{(k+8)} \\ M_{(k)} & M_{(k+1)} & \cdots & M_{(k+8)} \\ N_{(k)} & N_{(k+1)} & \cdots & N_{(k+8)} \end{Bmatrix}^T \tag{5}$$

The method of determining UAV State Space can be resumed as a process of solving (4), by inputting the flight data from 9 samples into the H_k matrix at (5) to compose the **A** and **B** matrices in (3), and finally constructing the State Space in (1) with the desired state and input vectors as defined in (2). Thus, to run the method, one must gather the angular rates, the flight angles, and the control torques as raw data to be processed. Continued to the following section, the quadrotor selected as UAV and its flight being simulated in the continuous time domain to provide the required flight data.

III. Simulation of Quadrotor Flight Attitude

Quadrotor has become a popular vehicle in the research of flight mechanics due to its simplicity and maintainability compared to other rotorcraft such as helicopter [3]. As previously explained in the introduction, the quadrotor UAV also performed various aerial vehicle, and most of them required attitude stabilizing, i.e. in videography or photographic mission that required the quadrotor to "stand still" in constant attitude. Thus, the attitude controlling become a significant feature of quadrotor flight control, and draw many interests of research in flight control sciences.

The flight control research required an adequate model of flying vehicle for its control design purposes. When the quadrotor studied for flight control assessment, there are various techniques to simulate its flight. Most of the quadrotor flight simulation developed from the State Space which common in modeling its flight around equilibrium in the hovering condition. The different approach also applied the nonparametric model, e.g. the artificial neural network to accommodate the nonlinear and strong cross-coupling characteristic of quadrotor flight [19].

In this section, the quadrotor flight simulation performed using the first principle derived from Newton-Euler equation of motion [7]. The equation of motion is treated as an exact model that govern the quadrotor dynamics in complete fashion without any decoupling and linearization involved. Thus, by this exact model, the state space obtained to mimic the "true characteristic" of quadrotor flight. The quadrotor movement is modeled by a compact vector equation which expresses its angular acceleration as a function of the control torque that interacts with its moment of quadrotor inertia as follows.

$$\mathbf{J}\dot{\bar{\omega}} = \bar{u} - \bar{\omega}\times\left(\mathbf{J}\bar{\omega}\right) \tag{6}$$

The eq. (6) relating the quadrotor's angular acceleration vector, $\dot{\bar{\omega}}$, as a function of control torque vector, $\bar{u}$ which overcome the cross product of its actual angular rate, $\bar{\omega}$, and its inertia tensor, **J**, as the quadrotor's angular acceleration vector, $\dot{\bar{\omega}}$, inversely proportional with the inertia tensor, **J**. The inertia tensor composed from three inertia terms in the x, y and z rotation axis, i.e. I_{xx}, I_{yy} and I_{zz}, added with six terms of product of inertia of the crossed axis i.e. I_{xy}, I_{yx}, I_{yx}, I_{yz}, I_{zx} and I_{zy}, in the following arrangement:

$$\mathbf{J} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \tag{7}$$

To use the numerical integrator tool in the simulation software the equation of motion in (6) slightly manipulated into:

$$\dot{\bar{\omega}} = \mathbf{J}^{-1}\left\{\bar{u} - \bar{\omega}\times\left(\mathbf{J}\bar{\omega}\right)\right\} \tag{8}$$

The simulation started with all zero in initial-conditions, then the instantaneous angular rate $\bar{\omega}$ obtained by numerically integrating the right-hand side of (8). The integration executed using the Matlab's ode 113 or the Adams-Bashforth-Moulton predictor-corrector method [20]. The Matlab's ode 113 is a variable step and variable order that monitored the integration very closely so it can solve the differential equation in shorter time and less of computational cost [20].

After completing the quadrotor flight simulation, the next step is simulating the flight data sampling by mimicking the work of data logger on the real flight. The data logger sampled the flight data at regular intervals and saves them for analysis purposes. In the next section, the data sampling explained and calculating the state space quadrotor.

IV. Quadrotor State Space from Flight Data

As resumed earlier in Section 2 the method for determining the state space has been derived for generic UAV. Thus, several modifications were done for specific application in the quadrotor as follows. Instead of using the generic form, the author using the quadrotor state space derived in [12].

$$\begin{Bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{Bmatrix} pq \\ qr \\ rq \end{Bmatrix} + \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \begin{Bmatrix} L \\ M \\ N \end{Bmatrix} \tag{9}$$

Hence, the first row is solved using the following equation using the flight data around the k-th sampled:

$$\begin{aligned} &\{A_{11} \quad A_{12} \quad A_{13} \quad B_{11} \quad B_{12} \quad B_{13}\}^T \\ &= H_k^{-1}\{\dot{p}_{(k)} \quad \dot{p}_{(k+1)} \quad \cdots \quad \dot{p}_{(k+5)}\} \end{aligned} \tag{10}$$

With the coefficient matrix is the 6x6 matrix, H_k:

$$H_k = [pqr_{col} \quad LMN_{col}]$$

with:

$$pqr_{col} = \begin{Bmatrix} qr_{(k)} & qr_{(k+1)} & \cdots & qr_{(k+5)} \\ pr_{(k)} & pr_{(k+1)} & \cdots & pr_{(k+5)} \\ rq_{(k)} & rq_{(k+1)} & \cdots & rq_{(k+5)} \end{Bmatrix}^T \tag{11}$$

$$LMN_{col} = \begin{Bmatrix} L_{(k)} & L_{(k+1)} & \cdots & L_{(k+5)} \\ M_{(k)} & M_{(k+1)} & \cdots & M_{(k+5)} \\ N_{(k)} & N_{(k+1)} & \cdots & N_{(k+5)} \end{Bmatrix}^T$$

To obtain these data, the quadrotor flight simulation carried out with the following scheme:

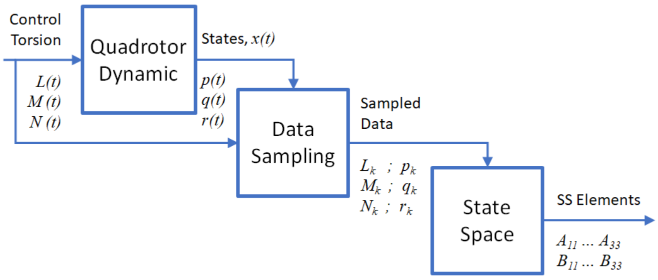

Fig. 3. Block Diagram for Flight Data Sampling Simulation

The "Quadrotor Dynamic" in Fig. 1 modeled the quadrotor flight according to the Newton-Euler equation from (8), processing the continues data, the control torsion *L(t)*, *M(t)*, and *N(t)* as the input to produce the angular rates *p(t)*, *q(t)*, and *r(t)* to be sampled in the following block. The "Data Sampling" block simulates the sampling of the continues-based data every 200 ms and logs the sampled *L(k)*, *M(k)*, *N(k)*, *p(k)*, *q(k)*, and *r(k)* for further process in obtaining the State Space element. In the final block "State Space", the sampled data inputted to (11) to solve (10) and completing the elements listed in (9).

TABLE I. The Quadrotor Tensor od Inertia Elements

Parameters	Value	Unit
I_{xx}	0.008	kg.m^2
I_{yy}	0.008	kg.m^2
I_{zz}	0.01	kg.m^2
I_{xy}, I_{yx}, I_{yx}, I_{yz}, I_{zx} and I_{zy},	0	kg.m^2

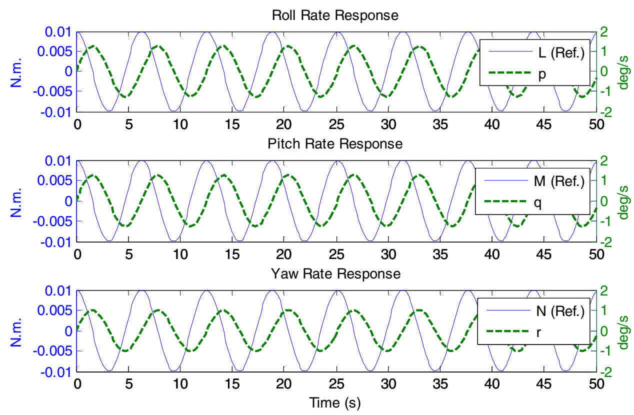

Fig. 4. Control Torque and Attitude Responses

In running the flight simulation, the quadrotor is considered to have tensor of inertia with the values listed in Table 1. Then the quadrotor excited with the sinusoidal input of 1 rad/s of frequency and 0.01 Nm of amplitude. From (9), the rolling torsion L directly influenced the Quadrotor via the first line elements of the State Space. Thus, to obtain the state space elements, the control torque is separately excited for each of respective row, i.e. excite the L for the first row of state space elements, M for the second row and N for the third row. Each of these excitations produce the angular rate responses as displayed in Fig. 4.

The continues data illustrated in Fig. 4, undergone sampling processed in every 200 ms and the state space determination yield the following matrices A and **B** elements:

$$\mathbf{A}=\begin{bmatrix} -0.1809 & -0.2026 & -0.1542 \\ -1.5702 & -1.9921 & -1.9514 \\ -0.2165 & -0.3004 & -0.2418 \end{bmatrix}$$
$$\mathbf{B}=\begin{bmatrix} 25.0543 & -0.2605 & -0.3733 \\ -0.8473 & 26.1247 & -0.8268 \\ -0.2418 & -0.0799 & 24.1748 \end{bmatrix} \quad (12)$$

The completion of flight data processing has yield the matrices element as listed in (12) to equip the defined quadrotor's state space in (9). The main diagonal elements are dominant in each row which validated the main effect of each control torsion to its respective rotation, i.e. the L dominates rolling rates, the M dominates the pitching rates and the N dominates the yawing rates. Since every element is non-zero, it enhanced the fact that quadrotor is a strongly crossed-coupled system.

The validation of the model estimated were depend on the accuracy of the numeric integrator to solve (8) and the accuracy of the inversion of the matrix in solving (11). The Adams-Bashforth-Moulton predictor-corrector method ensure the convergent integration [20] to mimic the realistic dynamic of the quadrotor movement. The pseudo-inverse algorithm used to solve (11) becomes the crucial factor for the accuracy of the state-space obtained. The better accuracy of the pseudo-inverse method yields the more valid state space obtained.

V. Conclusion

The method of obtaining the state space from flight data has been assessed and showing proper results. Since quadrotor flies in a nonlinear dynamic, strong coupling and also underactuated [5], the non-zero elements of the State Space located in the cross-coupling position validated that flight characteristic of the quadrotor.The methods validation depend with the accuracy of the inversion algorithm used to invert the coefficient matrix. Thus, the method is capable to approaches the UAV flight characteristic in a more realistic way than the linearized model.

Acknowledgment

This work was supported by the PITTA (*Publikasi Internasional Terindeks untuk Tugas Akhir Mahasiswa UI*) 2017 of the DRPM UI (*Direktorat Riset dan Pengabdian Masyarakat - Universitas Indonesia*).

References

[1] T. Prucksakorn, K. Wachirarattanakornkul, and I. Nilkhamhang, "Unmanned aerial vehicle for observing landslide with iterative feedback tuning," in The 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2013, pp. 1-5.

[2] "Drone tested by Albany sea rescuers at Salmon Holes set to help in emergencies," *ABC News*, A. Sargent, ed., 2017.

[3] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment," *AIAA Guidance, Navigation and Control Conference and Exhibit*, Guidance, Navigation, and Control and Co-located Conferences: American Institute of Aeronautics and Astronautics, 2007.

[4] Z. Jin-Liang, and Z. Jian, "Research on control method and Controller design for micro quadrotor aircraft," in The 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS), 2016, pp. 317-320.

[5] X. Zhang, X. Li, K. Wang, and Y. Lu, "A Survey of Modelling and Identification of Quadrotor Robot," *Abstract and Applied Analysis,* vol. 2014, pp. 16, 2014.

[6] M. B. Tischler, and R. K. Remple, *Aircraft and rotorcraft system identification*, Virginia, USA: American Institute of Aeronautics and Astronautics, Inc., 2006.

[7] A. Honglei, L. Jie, W. Jian, W. Jianwen, and M. Hongxu, "Backstepping-Based Inverse Optimal Attitude Control of Quadrotor," *International Journal of Advanced Robotic Systems,* vol. 10, no. 5, pp. 223, 2013.

[8] K. Ogata, *Modern control engineering*: Prentice-Hall, 2002.

[9] N. S. Nise, *Control system engineering*, New York: John-Wiley and Sons Inc., 2011.

[10] A. Julkananusart, and I. Nilkhamhang, "Quadrotor tuning for attitude control based on double-loop PID controller using fictitious reference iterative tuning (FRIT)," in IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society, 2015, pp. 004865-004870.

[11] J. Muliadi, and B. Kusumoputro, "Determining the UAV State Space Rotational Dynamics Model Using Algebraic Inversion Technique," in Proceedings of

the 8th International Conference on Computer Modeling and Simulation, Canberra, Australia, 2017, pp. 52-56.
[12] R. Beard, "Quadrotor dynamics and control," *All Faculty Publications*, http://scholarsarchive.byu.edu/facpub/1325, 2008.
[13] M. Y. Amir, and V. Abbass, “Modeling of Quadrotor Helicopter Dynamics,” in 2008 International Conference on Smart Manufacturing Application, Gyeonggi-do, Korea, 2008, pp. 100-105.
[14] W. Hussein, M. El-khatib, A. Elruby, and H. Haleem, “Quad Rotor Design, Simulation And Implementation,” in International Conference on Computer Science from Algorithms to Applications, (CSAA09) JW Marriott, Mirage City, Cairo, Egypt, 2009.
[15] H. Oh, D.-Y. Won, S.-S. Huh, D. H. Shim, M.-J. Tahk, and A. Tsourdos, “Indoor UAV Control Using Multi-Camera Visual Feedback,” *Journal of Intelligent & Robotic Systems,* vol. 61, no. 1, pp. 57-84, 2011.
[16] A. Kotikalpudi, B. Taylor, C. Moreno, H. Pfifer, and G. Balas, “Swing Tests for Estimation of Moments of Inertia,” *Unpublished notes, University of Minnesota Dept. of Aerospace Engineering and Mechanics*, 2013.
[17] M. Jardin, and E. Mueller, “Optimized Measurements of UAV Mass Moment of Inertia with a Bifilar Pendulum,” in AIAA Guidance, Navigation and Control Conference and Exhibit, 2007.
[18] A. Teimourian, and D. Firouzbakht, “A Practical Method for Determination of the Moments of Inertia of Unmanned Aerial Vehicles,” in The XXII Conference of Italian Association of Aeronautics and Astronautics, Napoli, Italy, 2013.
[19] M. A. Heryanto, W. Wahab, and B. Kusumoputro, “Optimization of a neural network based direct inverse control for controlling a quadrotor unmanned aerial vehicle,” *MATEC Web of Conferences,* vol. 34, pp. 04003, 2015.
[20] R. Ashino, M. Nagase, and R. Vaillancourt, “Behind and beyond the Matlab ODE suite,” *Computers & Mathematics with Applications,* vol. 40, no. 4, pp. 491-512, 2000/08/01/, 2000.