Journal of Intelligent & Robotic Systems (2019) 95:491–510
https://doi.org/10.1007/s10846-018-0857-x

A Study on Coaxial Quadrotor Model Parameter Estimation: an Application of the Improved Square Root Unscented Kalman Filter

Jarosław Gośliński[1] · Andrzej Kasiński[2] · Wojciech Giernacki[2] · Piotr Owczarek[3] · Stanisław Gardecki[2]

Received: 10 January 2018 / Accepted: 20 April 2018 / Published online: 17 May 2018

Abstract

The parametrized model of the Unmanned Aerial Vehicle (UAV) is a crucial part of control algorithms, estimation processes and fault diagnostic systems. Among plenty of available methods for model structure or model parameters estimation, there are a few, which are suitable for nonlinear UAV models. In this work authors propose an estimation method of parameters of the coaxial quadrotor's orientation model, based on the Square Root Unscented Kalman Filter (SRUKF). The model structure with different aerodynamic aspects is presented. The model is enhanced with various friction types, so it reflects the real quadrotor characteristics more precisely. In order to validate the estimation method, the experiments are conducted in a special hall and essential data is gathered. The research shows that the SRUKF, can provide fast and reliable estimation of the model parameters, however the classic method may lead to serious instabilities. Necessary modifications of the estimation algorithm are included, so the approach is more robust in terms of numerical stability. The resultant method allows for dynamics of selected parameters to be changed and is proved to be adequate for *on-line* estimation. The studies reveals tracking properties of the algorithm, which makes the method more viable.

Keywords Parameter estimation · Coaxial quadrotor · Mathematical model · Nonlinear filtration · Square root unscented Kalman filter

1 Introduction

In recent years Unmanned Aerial Vehicles have been widely used. The most popular UAVs configurations are fixed-wing and multi-rotor [20, 21]. They are applied in such tasks like: mapping, surveillance and reconnaissance, search and rescue, agriculture and others [30, 40]. Today one may hear about growing number of projects, where drones play the main role: from the flying cameras, through the personal observer to the sophisticated hardware simulator for the Mars lander [33]. Researchers develop improved control mechanisms as well as refined estimation techniques. Currently drone constructors are facing new problems: safety and security. The question is not about how to control the UAV or estimate its parameters, but about reliability and robustness of the proposed systems.

Among many, in literature, one can distinguish three different type of systems: remotely-guided, semi - autonomous, and fully-autonomous. Often these three are distributed into ten-point scale, namely Autonomous Control Level (ACL) [5]. In order to achieve higher level of the autonomy it is necessary to incorporate real time self-health diagnostic and mechanisms of adaptation to faults or even failures. As a consequence, specialized control algorithms with adaptive or robust structure are being applied [2, 23]. In most cases, the control scheme needs prior parametrization, usually - based on a mathematical model [3, 15, 32, 34]. Similar needs may appear during the usage of observers for fault tolerant system in quadrotors (Fig. 1) [17].

Various approaches are used in case of model parameters estimation, namely identification. Since a quadrotor model is highly nonlinear and internally-coupled system which

✉ Wojciech Giernacki
wojciech.giernacki@put.poznan.pl

[1] AISENS Sp. z o.o., F. Lubeckiego 23A, 60-348 Poznan, Poland

[2] Institute of Control, Robotics and Information Engineering, Faculty of Electrical Engineering, Poznan University of Technology, Piotrowo 3A, 60-965 Poznan, Poland

[3] Institute of Mechanical Technology, Faculty of Mechanical Engineering and Management, Poznan University of Technology, Piotrowo 3, 60-965 Poznan, Poland

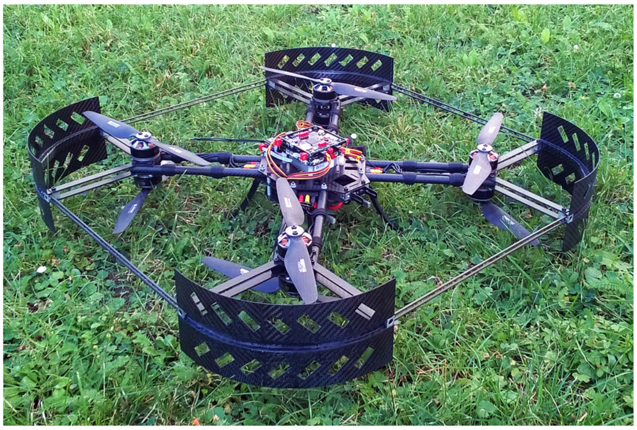


Fig. 1 The Falcon V5 coaxial quadrotor developed at the Institute of Control, Robotics and Information Engineering of Poznan University of Technology

has many physical as well as aerodynamic effects, the identification task is challenging. There are two major strategies in system identification: *online* and *offline*, both in frequency-domain (FD) and in time-domain (TD). The difference between *online* and *offline* [25, 36] methods may be understood in terms of needs or availability of onboard computational power, however the choice between time or frequency domain is not yet so trivial.

In frequency domain, e.g. in UAV, the time flight data is converted to frequency domain data. The procedure may by explained based on popular software for identification CIFER®[1] [36, 43]. Firstly, the flight data is gathered during the flight (CIFER® is an *offline*, batch method), then, the frequency response for each input-output pair is calculated together with the coherence coefficients (on this stage the model has already defined structure). Next, the multi-variable frequency analysis is carried out; which removes all cross effects in SISO models. Also partial coherence coefficients are calculated. Afterward, the optimization processes are being applied to time window lengths, so the accuracy for low and high frequencies is at a proper level. Finally, the state-space model is identified based on optimization method, driven by frequency response matching. Additionally, the parametrized model is validated at time domain.

Identification in frequency domain is claimed to have several advantages. Firstly, it stem from linear system theory, because of that it gives simple insight in itself and has strong foundations. Also, one has eventuality to choose the range of frequencies in input sequence for the identification (with accordance to an information content and signal-to-noise ratio). Furthermore, in frequency domain, the model reveals much more information about its characteristics compared to time domain. Therefore it is in general more universal. Models identified in frequency domain are suitable in a case of simulation, control law design or model structure analysis. This approach was successfully used in the case of UAV models in [11] for an insect-like micro aerial vehicle, [10] for fixed-wing UAV or in [19] in a test of flight flutter, namely: vibration, resonance and damping. The most popular methods for an identification in frequency domain are Least Square (LS), recursive Least Squares (RLS) [37, 38], Maximum Likelihood (ML) and Recursive Maximum Likelihood (RML) [6, 26].

In frequency domain the identification process is able to exploit the MIMO model structure, although each and every sub-model (pair of input-output) must be linear. This is the main feature yet disadvantage of the FD approach. Nonlinear models need to be linearized at an equilibrium point prior to identification. In a case of specialized software like CIFER®, the identified model will be automatically described by the best linear fit to the response, based on the first harmonic approximation in Fourier series [36]. That does not satisfy the requirements for the quadrotor UAV model which is, as already stated, highly nonlinear and has a coupled structure. The model cannot be simply approximated with linear model, and the linearization is only valid for a hoover state.

In some cases that problem can be solved with special techniques. To address this, Adams and Allemang [1] proposed method where model is divided into linear and non-linear parts. In terms of simplicity, the overall method boils down the non-linear state-space model to linear one with the non-linear input function (excitation). The non-linear input function depends on current input of the prior model and the non-linear function which links the subsequent derivatives of the state. As a result the FNSI gives linear model parameters as well as non-linear model part coefficients. The novel approach was successfully adopted in [37] and [38] under the name: Frequency-domain non Linear Subspace Identification (FNSI). Although the FNSI was proved to be efficient, only one work describes practical use of this method [38]. The method can be used to all models with previously mentioned structure, though a conversion from a UAV dynamic model is challenging. The most laborious part of the conversion is to obtain the linear sub-model. In the worst case scenario the linear part is given as an identity matrix, while the nonlinear part forms the overall previously defined model. The FD system identification gives many possibilities, however the cost is disproportionately high. To authors best knowledge, there are not many articles devoted to non-linear quadrotor model identification in frequency domain. In contrast, time-domain methods seem to be more promising.

[1] Comprehensive Identification from FrEquency Responses, software developed by the U.S. Army and NASA specifically for rotor-craft applications

The most popular and famous method for time-domain identification is Kalman Filter (KF) [29], a recursive algorithm, which works as a two-stages process. In a survey [22] one finds that the KF was successfully applied in various UAVs applications. Primarily, it was designed for linear systems, expressed in a space-state form. Later, its nonlinear version was developed, namely Extended Kalman Filter (EKF). Although the EKF has been used for more than 40 years [12] and has been applied successfully in many works [18, 41, 44], it is an imprecise method. It suffers from approximations, which are made during the linearization procedure. The EKF approximates mean and covariance up to the first order of the Taylor series. In case of highly non-linear problems it does not reflect the statistical values properly. It is also believed that in extreme cases it exhibits divergence and may be hard to tune when dealing with significant nonlinearities [7].

The EKF was applied for UAV parameters estimation in numerous works. In [28], authors presented results from fixed-wing UAV identification methods, based on EKF and Error Mapping Identification (EMId). In order to verify proposed algorithm, the basic 3-DOF UAV model was given with the assumption that only angular velocities are measured. The experiment was carried out in hardware in the loop (HIL) and in simulations. The obtained results indicate, that the EKF is stable and its uncertainties are at bounded levels (evaluated based on main diagonal elements of the EKF covariance matrix). The EKF proved its effectiveness at presence of partially corrupted sensory data and lower sampling rate. In [8], authors have dealt with online parameters estimation of Vertical Take Off and Landing (VTOL) UAV, by use of the EKF. Identification results were used for Riccatti equation solver in control procedure, providing optimal control even with prominent sensors noise or biases. Common and noticeable for all of the previous works is superiority of the EKF method, however, as already mentioned, the EKF may fail in case of highly non-linear and coupled models. That was partially proven in [7], where authors have proposed two Kalman Filter versions: the Extended and the Unscented.

The Unscented Kalman Filter (UKF) was firstly proposed by Julier and Uhlmann [27] as a solution for EKF problems. The main concept behind the UKF is Unscented Transformation (UT). During the Unscented Transformation a set of carefully chosen points (actually vectors) called sigma points are passed through the nonlinear function. The set of sigma points are chosen in that way, so the mean and covariance of the transformed function can be captured properly. The UT uses the re-sampling technique, so the statistic is calculated based on the set of points. Technically, if the noise distribution is Gaussian, the UKF provides approximation up to the third order, if the input is non-Gaussian the filter still holds the approximation accuracy at least to the second order of the Taylor series [35]. The UKF has been applied in different works, focusing mainly on attitude estimation [9], position estimation [42] or parameters estimation [24], though the last one remains practically unexplored. According to [22], the UKF in identification was used only in one work [4]. In [4] Brunke and Campbell have introduced the architecture for a model and state (joint) estimation of a fixed-wing plane. It is important to note that, the solution proposed by them was preliminary explained and proven. Firstly the authors have shown that the problem is non-linear and two most popular techniques including Extended Kalman Filtering and Maximum Likelihood are inadequate. Also by citing different authors, arguments for and against the EKF and the proposed UKF were presented. In conclusion Brunke and Campbell have suggested that the most reliable, viable, elegant and numerically stable is the Square Root version of the UKF (SRUKF), proposed by Merwe and Wan [35]. In addition, the authors have presented promising results obtained via the SRUKF method.

It is important to remind, that completely different way, which adds an extension to the well-known time and frequency domain horizon also exist. The most popular among others are heuristic techniques. In their scope, two main types of algorithms are placed: evolutionary and swarm. In [45] Yang and Liu used the Particle Swarm Optimization (PSO) method in a quadrotor UAV inertia parameters estimation. The research involved simple model derivation without aerodynamic effects. Also an interesting framework for PSO-based identification was given. The authors emphasized the problem associated with higher dimensions PSO (falling into the local optimum trap). In order to prevent it, two parallel PSOs were used in the same model identification procedure. This ultimate approach seems to be adequate in case of multimodal problems. However, the authors have not mentioned about model's non-stationarity. In different work regarding the PSO in an identification of UAV parameters [16] one finds similar model, enhanced with aerodynamic effects, though limited to one axis. The authors have given comprehensive study on the PSO, with different kind of cost functions and various model and parameters structures. They proved also that only switched models can be used, since the classic models with assumption of parameters constancy cannot properly reflect quadrotor's behavior. Furthermore, when dealing with switched models, it is crucial to introduce few of them as the accuracy needs to be at appropriate level. This involves increased number of parameters to be found and significantly extends the PSO's computational time. In the literature, one will find numerous applications where evolutionary or swarm algorithms are used for identification or simply model parameters estimation, however none of them are suitable for *online* tracking in case of non-stationary models (numerous parameters, computational burden).

To conclude: in order to identify the model parameters of a quadrotor UAV the time-domain method should be choose. Among many available methods, only a few are suitable for highly non-linear and coupled models. The most popular estimation methods are within the scope of Kalman Filters, where the highest accuracy, robustness and stability are expected to be achieved by the Square Root version of the Unscented Kalman Filter (SRUKF) [35].

In this work, authors have proposed quadrotor model described in [15] (adapted to coaxial quadrotor schame), enhanced with additional effects: flapping, induced, translational and profile drags and friction model. Then the identification method based on Square Root Unscented Kalman Filter (SRUKF) is presented as well as the results obtained via this method. Furthermore, the SRUKF method, proposed by Merwe and Wan [35] is modified, so it can be more robust in terms of numerical stability - regardless of initial settings and conditions. Also the universal framework for non-linear dynamic models parameters identification is given. Studies on different model structure and identifier input-output dependency are also presented. In addition, the comparison of identification results for the same model, based on PSO method [16] and the SRUKF method is included. The novelty of the paper lies in the improved SRUKF method applied to *online* model parameters estimation, which is assumed to be non-stationary. It also lies in the presented framework, as it gives capabilities in the field of identification, especially in UAV. To our best knowledge there are no works where an issue of *online* UAV quadrotor non-stationary models parameters estimation was raised.

The article is organized as follows: in the second section the coaxial quadrotor model is derived and described. In the third section, the SRUKF method as well as its modifications are presented. In the fourth section the experiment and all assumptions undertaken prior to the experiment are described. Finally, in the last section; results, conclusions and summary are presented.

2 The Coaxial Quadrotor Model

The basic quadrotor's model has six degrees of freedom (DOFs) (zero-order derivatives). First three DOFs describe position, the rest denotes orientation (Euler or Tait-Bryan angles) expressed in global reference system $\{EF\}$ (Robot local reference system is denoted as $\{BF\}$). The global $\{EF\}$ reference is configured as the NED (North-East-Down). In the paper, authors derive the model of the coaxial quadrotor, which has eight propellers mounted co-axially on a frame of the quadrotor (Fig. 1). Compared to the classic quadrotor, this leads to the similar mechanical structure with increased payload. It is often called octarotor, though it may be confusing since octarotors have in general planar mechanical structure. In order to simplify the vocabulary, authors use the coaxial quadrotor as a name of that sort of robots. In addition, authors would like to point out that planar octarotors are more efficient, but consume more space and are larger. Coaxial quadrotors provide a reasonable trade off between size and payload, especially when applied indoors. In the paper, the discussion is narrowed to the orientation part of the coaxial quadrotor model. The basic equation, which describes the angular acceleration (for convention from the Fig. 2) has the following structure:

$$\begin{aligned}\dot{\mathbf{R}} &= \mathbf{R}[\omega^{BF}]_\times, &(1)\\ \mathbf{I}\dot{\omega}^{BF} &= -[\omega^{BF}]_\times \mathbf{I}\omega^{BF} + \sum \tau, &(2)\\ m\dot{\mathbf{v}}^{BF} &= -2m[\omega^{BF}]_\times \mathbf{v}^{BF} + m\mathbf{R}^T\mathbf{g} + \sum \mathbf{f}, &(3)\end{aligned}$$

where the $\mathbf{R} : \{BF\} \rightarrow \{EF\}$, ω^{BF} – angular velocity given in $\{BF\}$, $[]_\times$ – skew symmetric span, m – robot total mass, $\mathbf{I}$ – robot inertia matrix, τ – torques acting on the robot, $\dot{\mathbf{v}}^{BF}$ – linear velocity ($\in \{BF\}$), $\mathbf{g}$ – gravitational acceleration, $\mathbf{f}$ – forces acting on the robot.

It is important to note, that, in order to retrieve the robot's orientation in $\{EF\}$, one is forced to integrate angular acceleration ($\dot{\omega}^{BF}$) and transform the result into the Euler rates (4):

$$\dot{\Theta} = \mathbf{P}_{BF}^{EF}\omega^{BF}. \tag{4}$$

Although other methods exist, including quaternions, this one remains simple, yet intuitive. Second integration of the Euler rates gives the Euler (or Tait-Bryan) angles. Matrix $\mathbf{P}_{BF}^{EF}$ is defined in Eq. 5 (detailed in [15]):

$$\mathbf{P}_{BF}^{EF} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{cos\theta)} & \frac{\cos\phi}{\cos\theta} \end{bmatrix}. \tag{5}$$

The overall definition of $\mathbf{P}_{BF}^{EF}$ depends on the Euler or Tait-Bryan representation, here the "3-2-1" convention is used

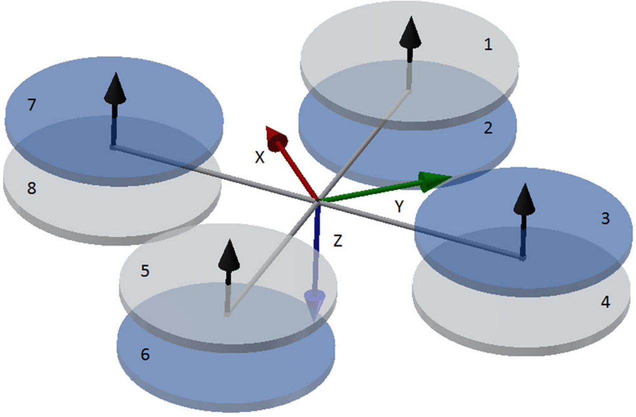


Fig. 2 The quadrotor skeleton and kinematic model (propellers marked by light gray circles have clockwise rotation)

("flight notation" with roll, pitch and yaw angles). The propellers generate forces, which (in simplest version) can be expressed as product of scalar parameter b_i and a second power of rotor's angular velocity ω_i, thus the force due to the i-th rotor may be given as in Eq. 6:

$$\mathbf{f}_i = \begin{bmatrix} 0 & 0 & b_i\omega_i^2 \end{bmatrix}^T. \tag{6}$$

Thrust is generated only in Z-axis of the local reference system $\{BF\}$. The input torque τ is a sum of torques, due to subsequent propeller. The i-th torque is a cross product of the i-th propeller position in respect to the $\{BF\}$ and the i-th force, Eq. 7.

$$\tau_i = \mathbf{p}_i \times \mathbf{f}_i, \tag{7}$$

$$\mathbf{p}_{1/2} = \begin{bmatrix} l_w \\ l_w \\ \pm h \end{bmatrix}, \mathbf{p}_{3/4} = \begin{bmatrix} -l_w \\ l_w \\ \pm h \end{bmatrix}, \tag{8}$$

$$\mathbf{p}_{5/6} = \begin{bmatrix} -l_w \\ -l_w \\ \pm h \end{bmatrix}, \mathbf{p}_{7/8} = \begin{bmatrix} l_w \\ -l_w \\ \pm h \end{bmatrix}, \tag{9}$$

where l_w is equal to $\frac{\sqrt{2}}{2}l$, l stands for arm's length in the quadrotor cross frame and h is a distance between propeller's plane and the center of mass in respect to Z axis of the $\{BF\}$. The Eq. 7 covers torques generated in X and Y axis of the $\{BF\}$, while the Z axis suffers from the reaction torques, given as:

$$\tau_z = d_1(\omega_1^2+\omega_4^2+\omega_5^2+\omega_8^2)+d_2(\omega_2^2+\omega_3^2+\omega_6^2+\omega_7^2). \tag{10}$$

In Eq. 10 parameter d_i refers to reaction torque gain. Two of them are introduced, which stems from the opposite type of propellers system. Propeller 1, 4, 5 and 8 have clockwise rotation and the rest are counter-wise.

This simple notation can be extended to additional effect like different types of drags. The better insight is presented in [3]. As it can be found, there are few types of drag-like effects: blade flapping, induced drag, translational drag, profile drag and parasitic drag (which for speeds up to 10 m/s may be ignored [3]).

Blade flapping effect is ubiquitous in rotor vehicles. It occurs when the rotor undergoes translational motion. It simply changes the rotor's tip path plane, deflecting it by the flapping angle β_{flap}. Here, the explanation is narrowed only to the final form of the force generated by the flapping effect, though one may find detailed explanation in [31]. Flapping force due to the i-th rotor is given in Eq. 11:

$$\delta_i = \mathbf{f}_i \left(\mathbf{A}_{flap}\frac{\mathbf{V}_p}{\omega_i} + \mathbf{B}_{flap}\frac{\omega^{BF}}{\omega_i} \right), \tag{11}$$

where $\mathbf{V}_p$ is a horizontal velocity in the $\{BF\}$ ($\mathbf{V}_p = \begin{bmatrix} V_x & V_y & 0 \end{bmatrix}^T$), while the $\mathbf{A}_{flap}$ and $\mathbf{B}_{flap}$ are the matrices with parameters to be identified:

$$\mathbf{A}_{flap} = \frac{1}{r}\begin{bmatrix} -A_{1c} & A_{1s} & 0 \\ -A_{1s} & -A_{1c} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{12}$$

$$\mathbf{B}_{flap} = \begin{bmatrix} -B_2 & B_1 & 0 \\ B_1 & -B_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{13}$$

where r refers to rotor radius.

Another effect is induced drag. It may occur in small quadrotors, where rotors blades are rigid or semi-rigid which cause an unbalanced in flapping forces. This effect can be simply described as a product of planar velocity and a scalar parameter:

$$\mathbf{D}_I = K_I\mathbf{V}_p. \tag{14}$$

Similarly to induced drag, it is also possible to distinguish translational drag. This effect occurs, when the air is redirected from any translational movement to the downward of the $\{BF\}$. In accordance with [3], this effect depends on a speed in the air. For slow movements, the translational drag is equal to:

$$\mathbf{D}_T = K_{T_1}\mathbf{V}_p, \tag{15}$$

while for fast maneuvers, it is necessary to incorporate the air velocity v_i induced by i-th rotor:

$$\mathbf{D}_T = K_{T_2}(-V_z + v_i)^4\mathbf{V}_p, \tag{16}$$

where the K_{T_1} and the K_{T_2} are scalar parameters.

The last two drag effects are profile drag and parasitic drag. Since the parasitic drag comes from non-lifting surfaces of a quadrotor and occurs only for high speeds ($> 10m/s$) it may be ignored [3]. Profile drag is gained with the transverse velocity of the rotor blades in the air:

$$\mathbf{D}_P = K_P\mathbf{V}_p, \tag{17}$$

where K_P is a lumped parameter.

Finally, the total drag force acting on the i-th rotor can be formulated as a sum of individual drags:

$$\mathbf{D}_i = \delta_i + \mathbf{D}_I + \mathbf{D}_T + \mathbf{D}_P, \tag{18}$$

while the torque due to drag effect, classified for i-th rotor is defined as:

$$\tau_{D_i} = \mathbf{p}_i \times \mathbf{D}_i. \tag{19}$$

The net torque τ_D, added to model (1) is a sum of all τ_{D_i}, where $i \in < 1 : 8 >$. In the model, the friction is composed of three main elements:

$$\tau_t = h_g\omega + b_w\,\mathrm{sign}(\omega) \odot \omega \odot \omega + c_w\omega \odot \omega \odot \omega, \tag{20}$$

where the h_g, b_w and c_w are friction coefficients, while $\odot$ stands for Hadamarda product.

Final equation for angular acceleration is given as:

$$\dot{\omega}^{BF} = \mathbf{I}^{-1}\left(-[\omega^{BF}]_{\times}\mathbf{I}\omega^{BF} + \sum_{i=1}^{8}\tau + \tau_D + \tau_{gyro} - \tau_t\right), \tag{21}$$

where τ_{gyro} is a gyroscopic torque due to the rotors and frame velocity.

3 Model Parameter Estimation – the SRUKF Method

In the Introduction comprehensive study on different methods which are adopted for model parameter estimation was given. In conclusion, authors stressed the role of the time-domain methods based on Kalman Filters by showing their superiority. Among them, probably the most interesting one is the Square Root Unscented Kalman Filter. It was firstly proposed by Merwe and Wan [35] as an extension to well known Unscented Kalman Filter method [27], though it has to be noted that some first attempts in Square-Root filters date back to 1995 [39]. As all Kalman Filter methods, the SRUKF is a recursive two-stages procedure and in case of parameter estimation it has simplified structure (compared to the state estimation variant). It is important to recall the SRUKF equations since the algorithm is partially changed. The parameter estimator or observer operates on the basic model, given as:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{r}_k, \tag{22}$$

$$\hat{\mathbf{d}}_k = \mathbf{G}(\mathbf{x}_k, \mathbf{w}_k) + \mathbf{e}_k, \tag{23}$$

where $\mathbf{w}_k$ are the parameters to be found, $\mathbf{r}_k$ refers to process noise, $\mathbf{G}(.)$ is the non-linear mapping function, $\hat{\mathbf{d}}_k$ is a desired output and $\mathbf{e}_k$ constitutes an error.

The notation (22)–(23), clarifies the roles which are assigned to the subsequent vectors and function, however it has to be explained where the non-linear UAV model (discretized by the forward Euler method) is placed. First of all, one needs to assess the measurement system applied in a UAV. Based on that it is possible to choose the observed values. The UAV is equipped with the Inertial Measurement Unit (IMU) or Attitude and Heading Reference System (AHRS) and in case of orientation these quantities are available: angular velocities, angles or quaternions. The model given in Eq. 21 has direct impact on the angular acceleration, thus it is highly important to observe or measure this quantity. In the best case scenario, the $\mathbf{d}_k$ represents the angular acceleration, while the $\mathbf{G}(.)$ transfers the previous state vector and parameters to the desired angular acceleration, however $\hat{\mathbf{d}}_k$ may also include angular velocities and angles or quaternions.

The SRUKF starts from initialization:

$$\hat{\mathbf{w}}_0 = E[\mathbf{w}], \tag{24}$$

$$\mathbf{S}_{w_0} = \text{chol}\left\{E[(\mathbf{w} - \hat{\mathbf{w}}_0)(\mathbf{w} - \hat{\mathbf{w}}_0)^T]\right\}. \tag{25}$$

During the first recursive stage, namely prediction, operations (26)–(30) are being executed.

$$\hat{\mathbf{w}}_k^- = \hat{\mathbf{w}}_{k-1}, \tag{26}$$

$$\mathbf{S}_{w_k}^- = \lambda_{RLS}^{-1/2}\mathbf{S}_{w_{k-1}}, \tag{27}$$

$$\mathbf{W}_{k|k-1} = [\hat{\mathbf{w}}_k^- \quad \hat{\mathbf{w}}_k^- + \gamma\mathbf{S}_{w_k}^- \quad \hat{\mathbf{w}}_k^- - \gamma\mathbf{S}_{w_k}^-], \tag{28}$$

$$\mathbf{D}_{k|k-1} = \mathbf{G}[\mathbf{x}_k, \mathbf{W}_{k|k-1}], \tag{29}$$

$$\hat{\mathbf{d}}_k = \sum_{i=0}^{2L} W_i^{(m)}\mathbf{D}_{i,k|k-1}, \tag{30}$$

where λ_{RLS} and γ are internal parameters (to be set based on a process knowledge). In the second stage (measurement phase), the (31)–(37) are used:

$$\mathbf{S}_{d_k} = \text{qr}\left\{\left[\sqrt{W_i^{(c)}}\left[\mathbf{D}_{1:2L,k} - \hat{\mathbf{d}}_k\right] \quad \sqrt{\mathbf{R}^e}\right]\right\}, \tag{31}$$

$$\mathbf{S}_{d_k} = \text{cholupdate}\left\{\mathbf{S}_{d_k}\ ,\ \mathbf{D}_{0,k} - \hat{\mathbf{d}}_k\ ,\ W_0^{(c)}\right\}, \tag{32}$$

$$\mathbf{P}_{w_k d_k} = \sum_{i=0}^{2L} W_i^{(c)}\left[\mathbf{W}_{i,k|k-1} - \hat{\mathbf{w}}_k^-\right]\left[\mathscr{D}_{i,k|k-1} - \hat{\mathbf{d}}_k\right]^T, \tag{33}$$

$$\mathbf{K}_k = \left(\mathbf{P}_{w_k d_k}/\mathbf{S}_{d_k}^T\right)/\mathbf{S}_{d_k}, \tag{34}$$

$$\hat{\mathbf{w}}_k = \hat{\mathbf{w}}_k^- + \mathbf{K}_k\left(\mathbf{d}_k - \hat{\mathbf{d}}_k\right), \tag{35}$$

$$\mathbf{U} = \mathbf{K}_k\mathbf{S}_{d_k}, \tag{36}$$

$$\mathbf{S}_{w_k} = \text{cholupdate}\left\{\mathbf{S}_{w_k}^-\ ,\ \mathbf{U}\ ,\ -1\right\}, \tag{37}$$

where L refers to the number of parameters, $W_i^{(c)}$ and $W_i^{(m)}$ are the method's parameters which need to be properly chosen and $\mathbf{R}^e$ is a measurement noise – according to [35] and can be set arbitrary, though it has a big influence on convergence speed (to be proven in the next section).

Behind the SRUKF, there are few different parameters that rule the algorithm (extensive description in [14, 35]). The first one is α, it determines the spread of sigma points around the estimated parameters (eg. value: 0.7). The second is β, which describes the distribution (eg. value: 2.0). Finally, there is the scaling parameter called κ (eg. value: 0 for state-estimation and $3 - L$ for parameter estimation [35]). The previously listed parameters depend on the α, β and κ in a given form:

$$\lambda = \alpha^2(L + \kappa) - L, \tag{38}$$

$$\gamma = \sqrt{L + \lambda}, \tag{39}$$

$$W_i^{(m)} = W_i^{(c)} = 1/2(L + \lambda), \tag{40}$$

$$W_0^{(m)} = \lambda/(L + \lambda), \tag{41}$$

$$W_0^{(c)} = \lambda/(L + \lambda) + (1 - \alpha^2 + \beta), \tag{42}$$

where $i = 1, ..., 2L$. The SRUKF algorithm given in Eqs. 26–37 is controlled via the λ_{RLS}, which has the same function as in case of Recursive Least Squares Method. It simply introduces the forgetting feature to the estimation process. It is commonly used and set close to 1 (e.g. 0.9995). As it came out during the first investigation, the λ_{RLS} may lead to significant numerical instabilities. In most cases, the safety could be guaranteed by setting the λ_{RLS} to 0.99999, but it was not necessarily the optimal. In order to make the method more robust and less sensitive on the initial setting, the λ_{RLS} is changed in each iteration so it can reach 1.0 in a certain time. Due to that, the algorithm has to be extended with the following formula:

$$\lambda_{RLS_k} = \lambda^0 \lambda_{RLS_{k-1}} + (1 - \lambda^0), \tag{43}$$

where the constant λ^0 change the characteristics of the λ_{RLS_k} in the subsequent samples. The influence of the λ^0 on the λ_{RLS_k} is depicted in the Fig. 3, for different λ^0.

Further modifications of the SRUKF method address the problem of the parameter convergence speed and confidence for the measurements or reference given in $\mathbf{d}_k$.

When the correction phase of the algorithm is invoked, results from the model given in $\hat{\mathbf{d}}_k$ are subtracted from the reference $\mathbf{d}_k$ (35). If the reference introduces errors, the corresponding variance in the covariance matrix $\mathbf{R}^e$ has to be increased. Although, it has strong influence on a speed of the parameters convergence, Authors of the [35] recommend to set the $\mathbf{R}^e$ arbitrarily (e.g. to $5\mathbf{I}$). That makes parameters partially immune to the errors added in the reference. However, high covariance values slow down the convergence speed. Because of this, the fast–changing parameters cannot be tracked well – one will find it difficult to properly choose the values of the $\mathbf{R}^e$. In order to find the optimal settings of $\mathbf{R}^e$, the optimization is needed.

Although the $\mathbf{R}^e$ has strong influence on the parameters as well as convergence characteristics of the algorithm, it cannot change parameter characteristic individually. The influence of the $\mathbf{R}^e$ remains implicit. For example, if $\mathbf{d}_k$, $\hat{\mathbf{d}}_k \in \mathbb{R}^{1\times 1}$, and $\mathbf{w}_k \in \mathbb{R}^{L\times 1}$, where $L > 1$, the $\mathbf{R}^e \in \mathbb{R}^{1\times 1}$ changes all the parameters equally. As it can be seen, the user cannot incorporate the prior knowledge about the parameters behavior into the estimation process. If one of the parameters is supposed to change slowly, it would be desirable to prepare the SRUKF algorithm, so it can cut off all sudden changes in the subsequent track. Unfortunately the original form of the algorithm, given in Eqs. 26–37 does not provide such tune possibilities. In order to add this, one simple operation is need to be improved. In Eq. 27, the $\mathbf{S}^-_{w_k}$ is enhanced with the square root of diagonal covariance matrix $\mathbf{Q}$, yielding the following:

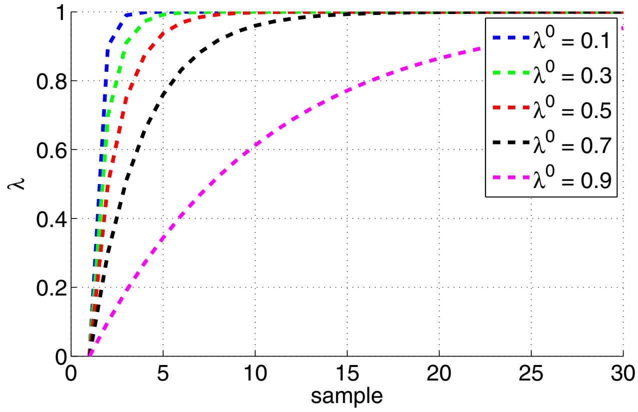


Fig. 3 The influence of the λ^0 on the λ_{RLS_k}

$$\mathbf{S}^-_{w_k} = \lambda_{RLS}^{-1/2} \mathbf{S}_{w_{k-1}} + \sqrt{\mathbf{Q}}. \tag{44}$$

This simple modification stem from the state estimation algorithm, where $\mathbf{Q}$ is used as a process noise. In that case, the $\mathbf{Q} \in \mathbb{R}^{L\times L}$. Note, that L parameters have the corresponding constant variance on the main diagonal of the $\mathbf{Q}$. By increasing subsequent variances, the user is able to set the parameters behavior to be volatile. The tracking capabilities are adjusted by $\mathbf{Q}$ and $\mathbf{R}^e$, however only its ratio explains how fast or how slow the tracking will be carried on. The settings of $\mathbf{Q}$ as well as $\mathbf{R}^e$ and the result achieved by use of the modified SRUKF method are shown and discussed in the next section.

4 The Experiment and Results

The paper concerns an orientation model of the coaxial quadrotor identified by the SRUKF method. In the previous sections the model and the method with improvements were shown. In this section, authors present experiments, restrictions (that are imposed) as well as the results achieved with the proposed method.

4.1 Assumptions and Restrictions

It is important to recall that the model given in Eq. 21 is in a continuous form, which cannot be applied directly in the SRUKF. The overall discrete form (discretized by the forward Euler method) was given as in Eq. 23, however that does not suits to the general state-space model, where the state vector $\mathbf{x}_k$ has to be updated in each and every step. In order to make these two forms equally, one has to introduce the measurement system as the reference. As it has been already stated, the reference $\mathbf{d}_k$ is formed from the state vector, where all angular accelerations, velocities and angles are captured (provided by the AHRS system). The aim is to minimize the error given in Eq. 45:

$$\mathbf{e}_k = \left(\mathbf{d}_k - \hat{\mathbf{d}}_k\right)^2, \tag{45}$$

where the $\hat{\mathbf{d}}_{k-1}$ is equal to $\mathbf{x}_k$. If the number of sigma points is $2L+1$, there are $2L+1$ vectors in $\mathscr{D}_{i,k|k-1}$, where all are calculated based on $\mathbf{x}_k$ and $\mathbf{W}_{k|k-1}$. However the $\mathbf{x}_k$ has to be calculated in the previous iteration. The basic question is how to calculate the $\mathbf{x}_k$ so, that the state vector will form an optimal trajectory. The solution can be provided based upon the update procedure, which allows to compute the model state vector in $k+1$ iteration i.e. $\mathbf{x}_{k+1}$, when the $\mathbf{x}_k$ and $\mathbf{w}_k$ are known:

$$\mathbf{x}_{k+1} = \mathbf{G}[\mathbf{x}_k, \hat{\mathbf{w}}_k]. \tag{46}$$

The procedures sequence of model updating and its parameters estimation is shown in the Fig. 4. As it can be seen, the model is updated after the correction phase of the SRUKF, based on the estimated parameters vector $\hat{\mathbf{w}}_k$.

The reference consists of three main components, i.e. angular accelerations, angular velocities and angles. The most important and required is the first one. Without the angular accelerations it is not possible to perform the estimation process. It is caused by the direct influence of the angular momentum on angular acceleration. The angular velocities and angels are less important. First tests proved that the model where only velocities and angles are used (as a reference) cannot be correctly identified. The SRUKF failed to operate in all cases, where the angular acceleration was not used. The reason is simple, though disputable. The most important part of the SRUKF lies in the calculation of the covariance matrix (actually its decomposition product). If the impact of the parameters to be estimated on the output of the model is slender, the covariance matrix is minimize to close to zeros and the resulting Kalman gain is increased. Finally the SRUKF becomes unstable and spoils parameter estimation process. As a matter of fact, low influence of the parameters on the model output is mostly caused by the integration process that occurs when calculating the angular velocities from the acceleration. Another important factor is coupling that exists between the velocities and acceleration. When calculating the integral, the angular velocities are obtained, however in the next iteration those velocities are used in the accelerations equations, where the gyroscopic torques are calculated. This may lead to model instabilities as the higher derivatives are not controlled directly. To handle this, one may set the initial parameters as close to optimal as possible, unfortunately most frequently these are unknown. The second option is to control highest derivatives, i.e. angular accelerations.

The final restriction refers to traceability of the parameters. The model given in Eq. 21, cannot be fully-identified as there are infinity sets of parameters. It is why, some of them must be given prior to an identification. The first set includes the inertia tensor parameters (six elements of the symmetric matrix $\mathbf{I}$), the second group consists of the rotor-propellers gains b_i and d_i. The propellers parameter or the gain will probably change during the flight, while the inertia parameters remain constant, this is way the authors have decided to seek for the gain parameters instead of inertia parameters, which are determined before the identification.

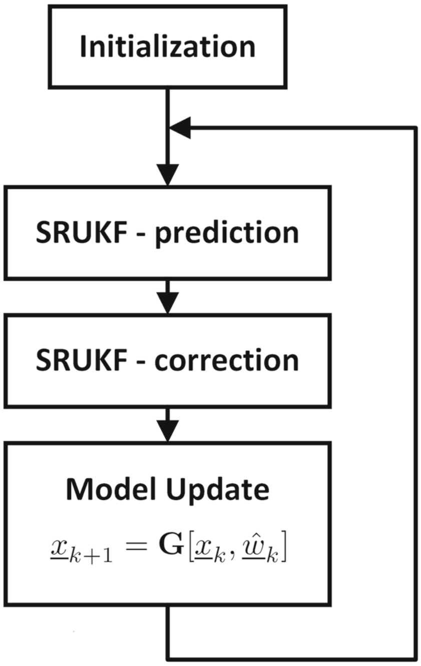

Initialization
SRUKF - prediction
SRUKF - correction
Model Update
$\underline{x}_{k+1} = \mathbf{G}[\underline{x}_k, \underline{\hat{w}}_k]$

Fig. 4 The SRUKF procedure and the model update

4.2 The Experiment, Gathered Data

The UAV used in the research (Fig. 5) was equipped with the AHRS (ADIS16488 by Analog Devices) recording samples at a frequency of 400 Hz. It provides the state estimation based on the built-in EKF estimator (400 Hz frequency), however the filter settings are not adjustable. That is why the custom algorithm that provides unbiased angular velocities

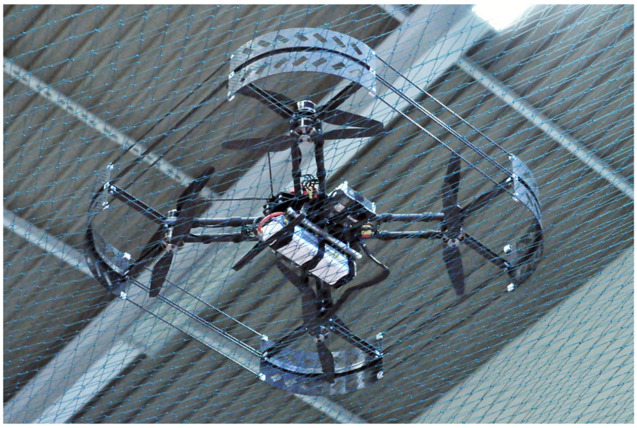


Fig. 5 The Falcon V5 coaxial quadrotor (start from a safety net)

and angles was used instead. The authors have used the sport hall (Fig. 6) as the air disturbances (assuming that the distance from the ceiling, floor and walls is sufficiently high) are lowest and the wind effects can be neglected.

It is important to notice that the research concerns estimation in open-loop system, where in the control the commands were sent with a period of 30ms. A detailed verification of the balance between the control and estimation frequencies (in a closed-loop system with observer) will be provided in next (separate) publication.

As it has already been stated, the reference consists of angular accelerations, velocities and angles (X,Y and Z axis), where accelerations are necessary. Unfortunately, in a great majority of AHRS, the angular acceleration is not available. In order to acquire it, the velocity derivative has to be calculated. This method has two disadvantages, the first one is the undesirably noise gain, and the second one is an introduced delay (at least one sample). In order to minimize the first effect, authors have applied the first-order, low-pass Butterworth FIR filter to velocities measurements (cut-off frequency equal to 0.1 of half of the normalized sampling rate). It strongly reduces the errors, though the same filter must be applied for all derivatives (same delay). As it came out, the introduced delay in angular acceleration has minor influence and could be neglected.

During the experiment the UAV coaxial quadrotor has operated in the closed area in all three axes with low linear speed (Figs. 7–8). The date gathered during the experiment consists of set values for BLDC motors (for eight propellers), estimated angles, speed in Z axis ($\in \{EF\}$), estimated height ($\in \{EF\}$), angular velocities (gyroscope-based), linear accelerations, magnetometers readings and the sample time (sample rate equal to 400Hz).

One of the most important aspects of the measurement system, is an unavailability of the propeller speed. These velocities are not measured, however they are needed as they are used in the force calculation. In order to possess the knowledge about the rotors speed, necessary calculation must be made. The rotor-propeller model can be given as the second-order inertial object [13], where the gain is included in the force (6) as b_i. It is important to note that, the second order inertia relation describes how the BLDC controller's set value (duty cycle in %) affects the propeller's speed. This simple procedure allows to estimate the rotors velocities.

Fig. 6 The sport hall and the UAV coaxial quadrotor

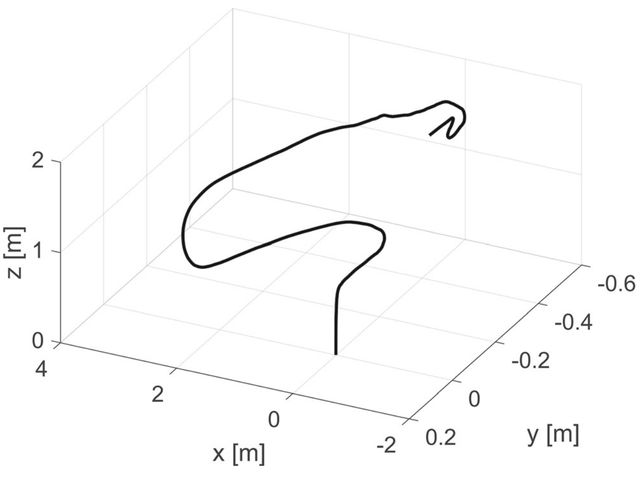


Fig. 7 3-D plot of the flight trajectory

4.3 The Results

The experiment involved coaxial quadrotor with main board equipped with STM microprocessor and internal memory. Data collected during flights was stored in the memory. After a flight the data was downloaded on a stationary computer and was used in further processing. The SRUKF estimation algorithm was used together with the discrete version of the model given in Eq. 21, fed with gathered and filtered data. The discrete version of the model was obtained via explicit forward Euler method. Similarly, the discrete integration was realized with the same approximation.

The discrete model of the coaxial quadrotor has sixteen gains b_i in X and Y axes (eight per axis, two per propeller), eight gains d_i in Z axis (one per propeller), nine friction coefficients hg_x, bw_x, cw_x, hg_y, bw_y, cw_z, hg_z, bw_z, cw_z (three per axis, for first, second and third power of the velocity in the subsequent axis), two coefficients of the flapping effect B_1, B_2 and six offsets for velocities and angles in X, Y and Z axis. The quadrotor has also constant parameters, which are calculated or measured prior to the experiment (Table 1).

After the first estimation trial it was observed that some of the parameters are redundant. The coaxial propellers have the same gain, which reduces the number of b_i to eight. Similar applies to d_i, where only four of them are necessary. The final equation for τ has a following form:

$$\tau = \mathbf{CDw} \odot \mathbf{w}, \tag{47}$$

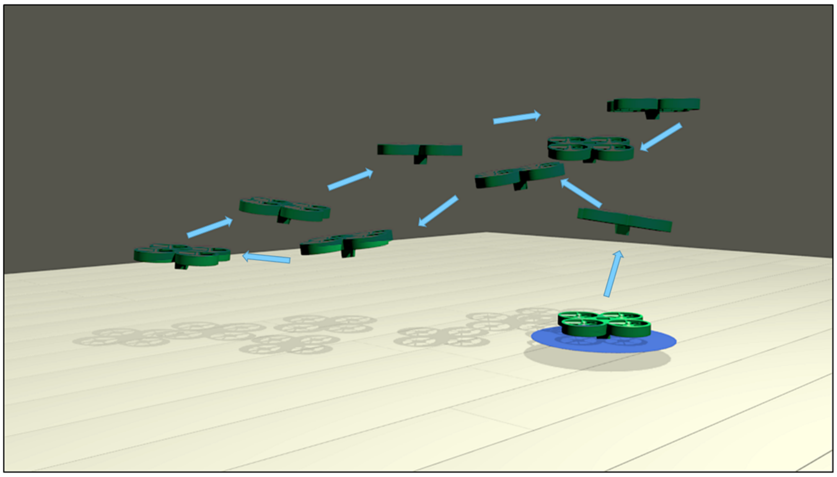


Fig. 8 Visualization of the flight trajectory in 3-D - available at http://uav.put.poznan.pl and https://youtu.be/H4KuRLZkpuM

where:

$$\mathbf{C} = \begin{bmatrix} l\frac{\sqrt{2}}{2} & 0 & 0 \\ 0 & l\frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{48}$$

$$\mathbf{w} = \begin{bmatrix} w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & w_7 & w_8 \end{bmatrix}^T, \tag{49}$$

$$\mathbf{D} = \begin{bmatrix} b_1 & b_1 & b_2 & b_2 & -b_3 & -b_3 & -b_4 & -b_4 \\ -b_5 & -b_5 & b_6 & b_6 & b_7 & b_7 & -b_8 & -b_8 \\ d_1 & d_3 & d_2 & d_4 & d_1 & d_3 & d_2 & d_4 \end{bmatrix}. \tag{50}$$

As it can be seen, the coaxial propellers pairs in X and Y axis differ with the gains. This is crucial, as the model with further simplification cannot be identified properly. The most important question is why some of the naturally or theoretically equal parameters like b_1, b_5 or b_2, b_6 cannot be simplified? The answer may lie in the side effects when maneuver in the air. Some of the drag-like effects (induced, translational) are not accounted as the linear velocity is not available. Also the propellers may deflect, which results in different torques in X and Y axis. In the Z axis, that problem did not occur. The d_i parameters cover the doublets of upper and lower propellers with the same rotary direction.

During the first stage of the research, data gathered in the experiment was filtrated and used in the standard SRUKF algorithm. The results are shown in Figs. 9, 10, 11 and 12.

Table 1 Parameters of the coaxial quadrotor

Constant	Value	Unit
Propeller distance in Z axis (h)	0.02	m
Frame arm length (l)	0.2	m
Rotor-propeller moment of inertia I_w	0.000001	$kg \cdot m^2$
Robot moment of inertia I_{xx}, I_{yy}, I_{zz}	0.008	$kg \cdot m^2$
Robot product of inertia I_{xy}, I_{yz}, I_{zx}	0.00001	$kg \cdot m^2$

The angular acceleration is well reconstructed. The difference between the model and the reference can be captured only during the first 20 ms. However these 20 ms has a huge impact on angular velocity which results in the offset.[2] Similar situation occurred in case of the lowest derivations, namely Tait-Bryan angles. Although the offsets were estimated, the SRUKF could not follow its changes sufficiently fast. Authors have tested the algorithm with different settings for $\mathbf{R}^e$ without positive results. Moreover, the algorithm was stable only for $\mathbf{R}^e_{ii} \in < 0.01; 9 >$. In order to intentionally change the convergence for offsets and to boost the robustness of the proposed method, modification of the SRUKF was added. The $\sqrt{\mathbf{Q}}$ was modified, so its main diagonal elements responsible for the noise of the offset parameters were set arbitrary to 0.01. The results are shown in Figs. 13, 14, 15 and 16.

The applied modifications have allowed to follow the offset properly. The estimation process takes less than 0.8s to fully reconstruct the reference. It also reveals high stability, as the $\mathbf{R}^e$ was changed from small to large values (tested within the range: $< 0.001; 50 >$). As the $\mathbf{R}^e$ was decreased, the method started to be sensitive for small changes in the reference signal. The advantage of the low values in $\mathbf{R}^e$ can be visible as fast convergence, however this may lead to errors while some sudden changes occur. For example, when the reference is distorted by state estimation or measurement noise, the SRUKF apace follows this noise by appropriate changes in parameters, which definitely corrupts the estimation. Although reliable method for assessment of the hyperactive behavior of parameters estimator do not exist, one is able to intentionally introduced noise which should not have influence on parameters. Authors have added the noise as a peak value in the

[2] the offset is also increased by the integration errors

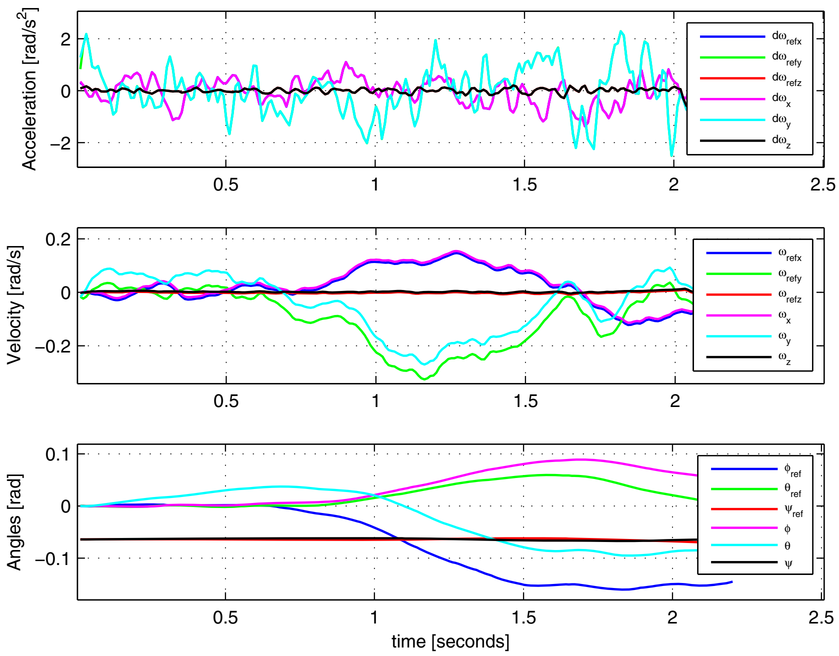


Fig. 9 Angular acceleration, velocity and Tait-Bryan angles

reference of the velocity in the X axis. Recorded results for different $\mathbf{R}^e$ are shown in Figs. 17, 18 and 20.

An adverse effect is clearly observed for $\mathbf{R}^e_{ii} = 0.005$. The most interesting case is depicted in the Figs. 19 and 20, where in 150-th sample of data, sudden reference noise occurred. It shows how simple data corruption changes the overall characteristics of the estimation. As it can be deduced the process of the algorithm tuning requires optimization. Authors have decided to use the PSO algorithm with only two parameters to be tuned-optimized. The first one is the $\mathbf{R}^e_{ii}$, the second is $\mathbf{Q}_{ii}$. An optimization must be performed in presence of noise, that have significant influence on parameters estimation for small $\mathbf{R}^e_{ii}$. A trade off is made between the SRUKF's hyperactivity and the

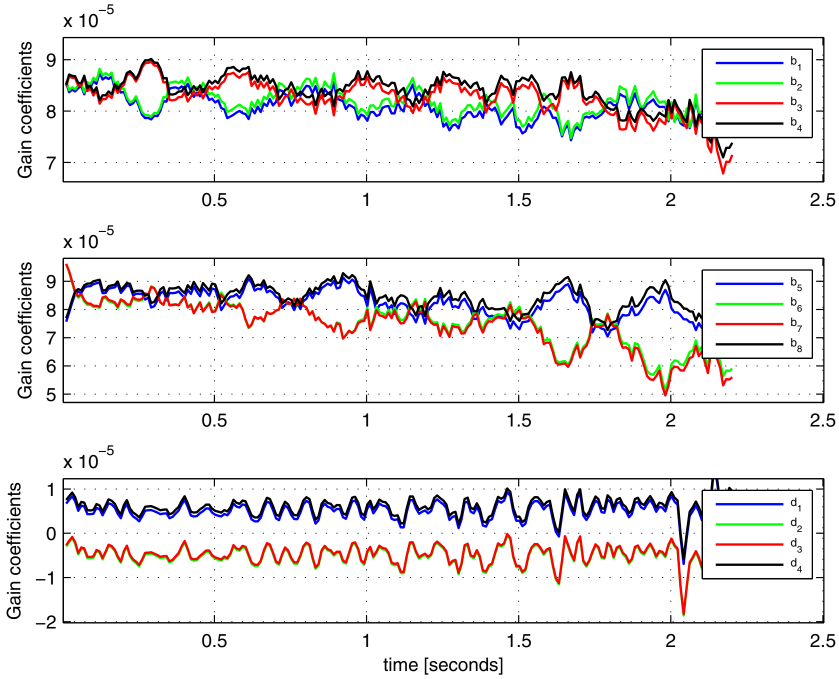
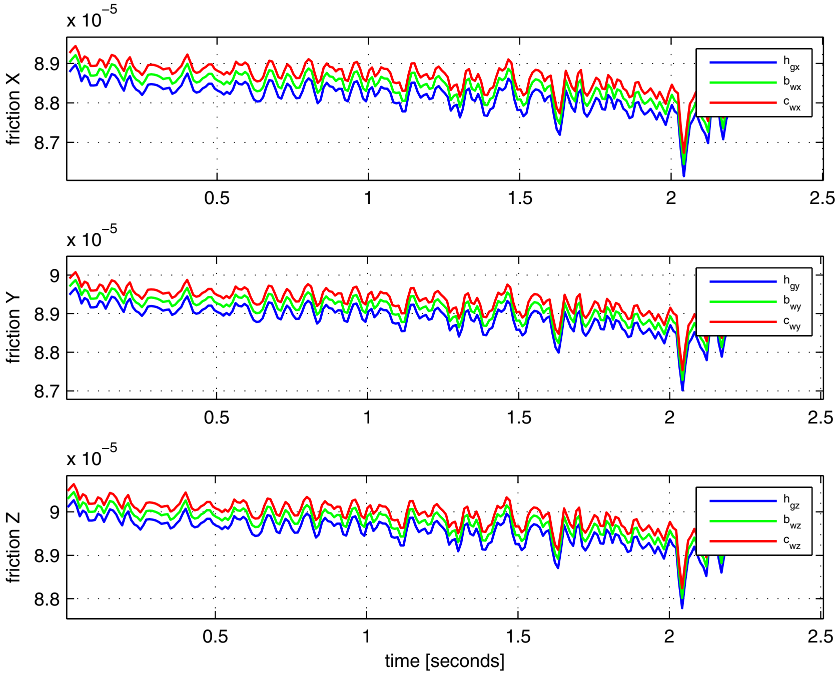


Fig. 10 Motor-Propellers gains b_i, d_i

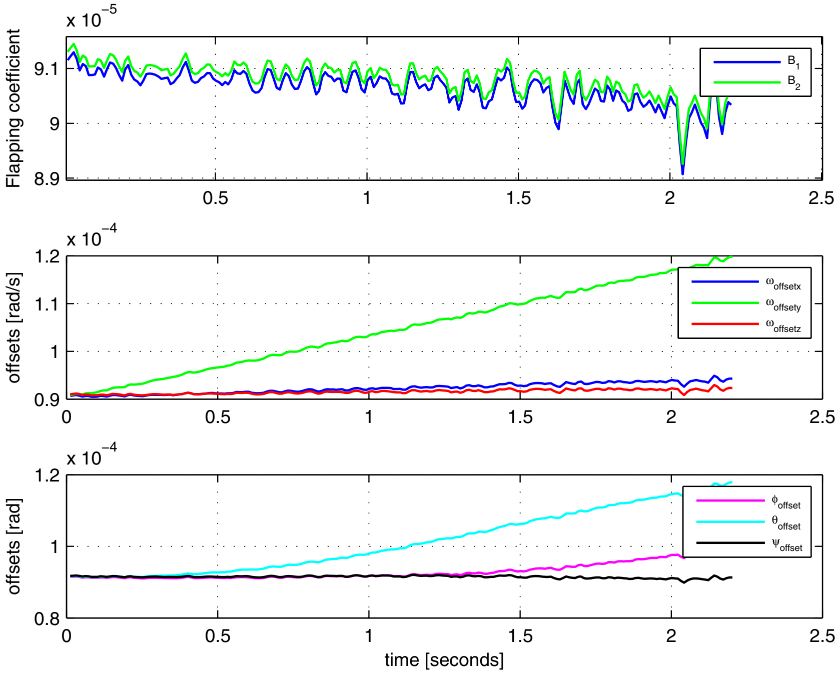
Fig. 11 Friction coefficients



convergence speed. The PSO's settings are described in [16]. In order to introduce fair evaluating factor, the cost function based on sum of the squared errors (SSE) for acceleration, velocity and angles is adopted. One of the PSO simplification is a length of input data, which was shortened to 200 samples. This not only reduces an optimization time, but also gives possibility to choose the most interesting scope of flight data. The obtained trajectory of particles in a chase for the global optimum is shown in the Figs. 21 and 22.

The SRUKF settings found via PSO satisfy the estimation performance, though the optimal performance depends on many variables, which cannot be provided

Fig. 12 Flapping and offsets coefficients



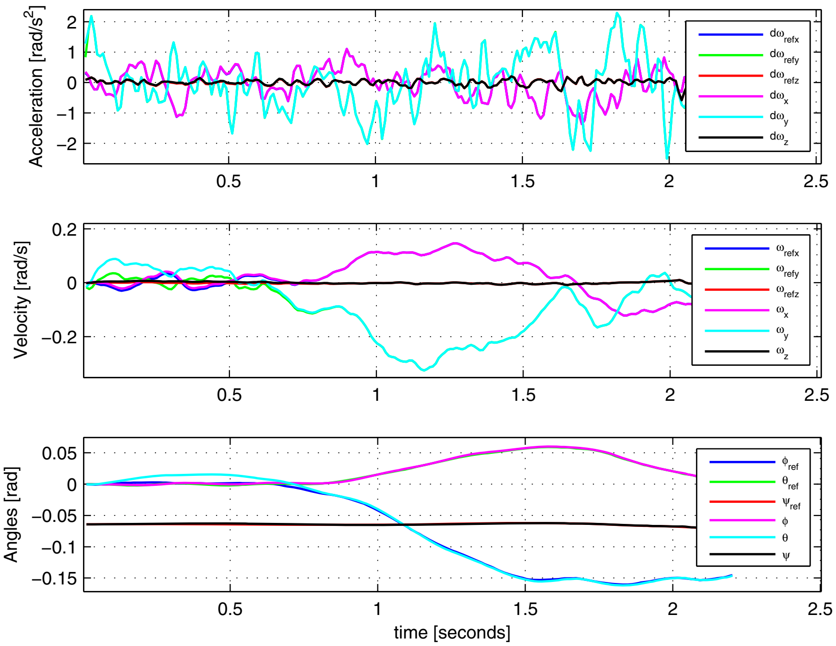


Fig. 13 Angular acceleration, velocity and Tait-Bryan angles

in advance. The results for all tests are given in the Table 2. The best performance is achieved for $\sqrt{\mathbf{R}^e_{ii}} = 5.0658$, $\sqrt{\mathbf{Q}_{ii}} = 0.0978$ and $\sqrt{\mathbf{R}^e_{ii}} = 6.4941$, $\sqrt{\mathbf{Q}_{ii}} = 0.0077$ (data disrupted). Interestingly, the best settings for disrupted reference data, which were found via PSO, have low sum of errors (22.884) for undistorted data. Therefore settings: $\sqrt{\mathbf{R}^e_{ii}} = 6.4941$, $\sqrt{\mathbf{Q}_{ii}} = 0.0077$ become optimal for SRUKF parameters estimator. Note that the cost function given in Figs. 21 and 22 are significantly different as the PSO optimization operated only on first two hundred samples, while the err_{sum} was calculated for one thousand input sequences.

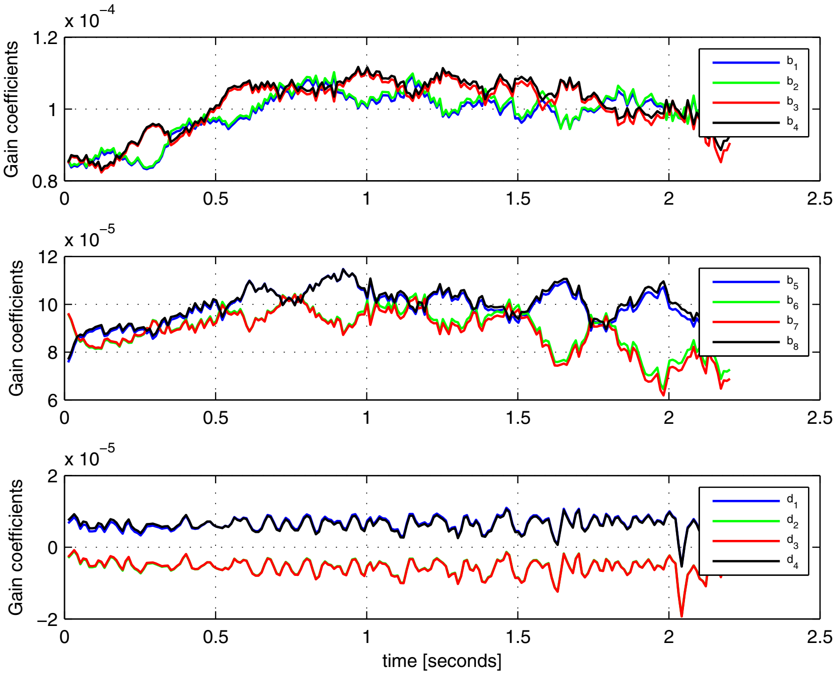


Fig. 14 Motor-Propellers gains b_i, d_i

Fig. 15 Frictions coefficients

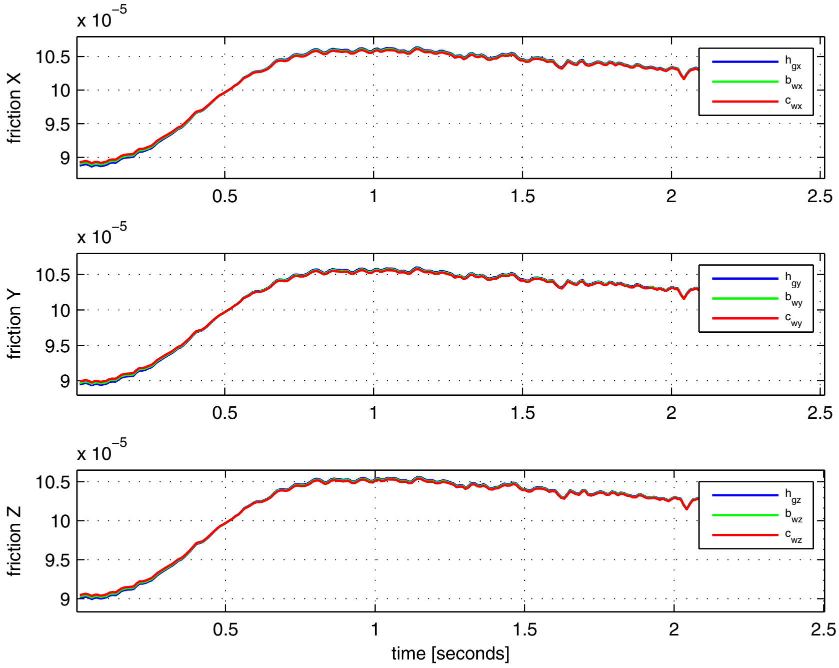


Fig. 16 Flapping and offsets coefficients

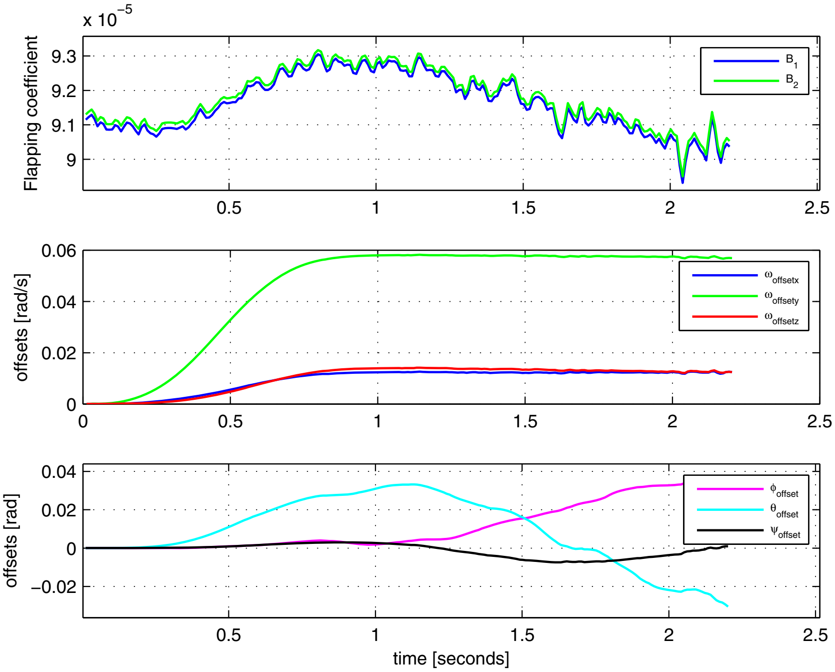


Fig. 17 Angular acceleration, velocity and Tait-Bryan angles, for $\mathbf{R}_{ii}^{e} = 150$ with distorted velocity reference

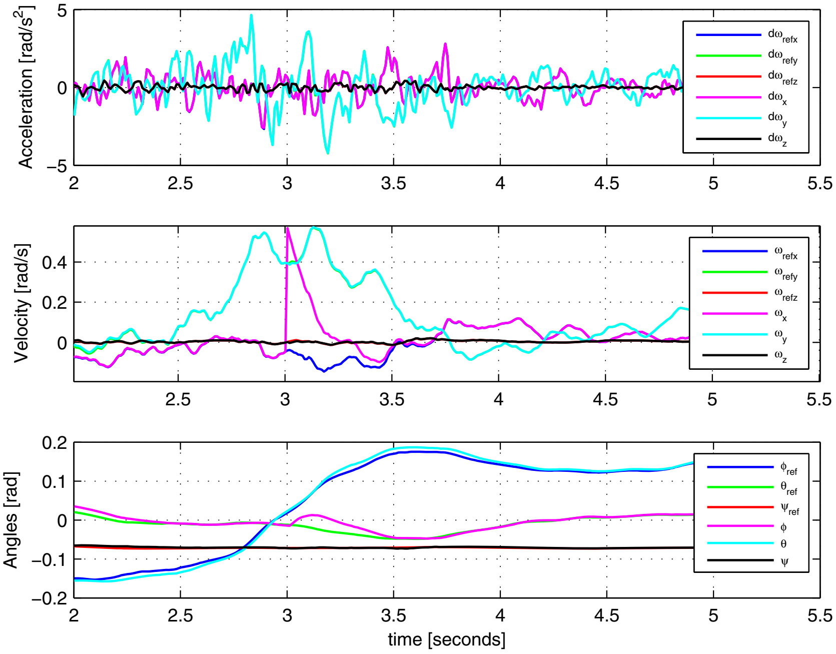


Fig. 18 Motor-Propellers gains b_i, d_i, for $\mathbf{R}_{ii}^{e} = 150$ with distorted velocity reference

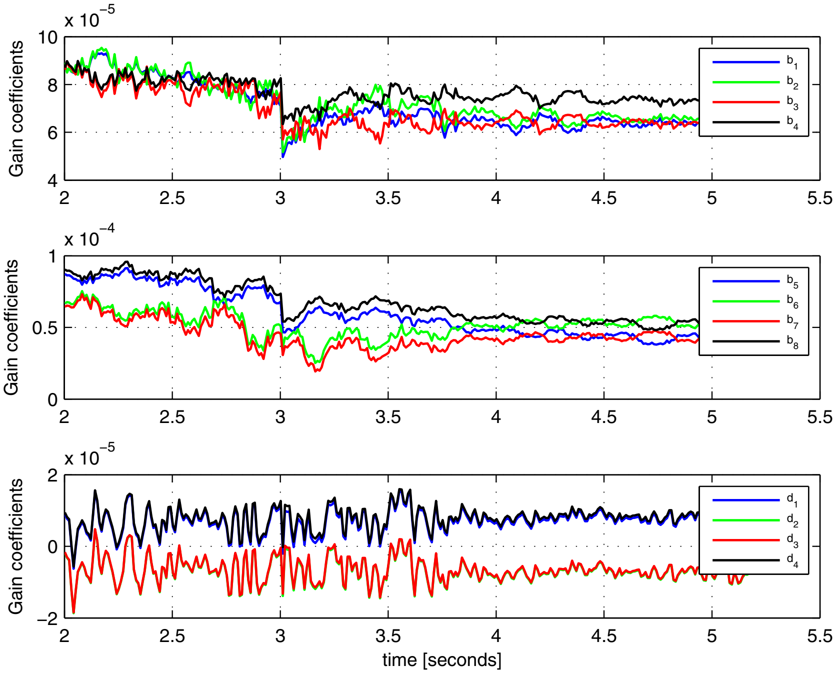


Fig. 19 Angular acceleration, velocity and Tait-Bryan angles, for $\mathbf{R}_{ii}^{e} = 0.005$ with distorted velocity reference

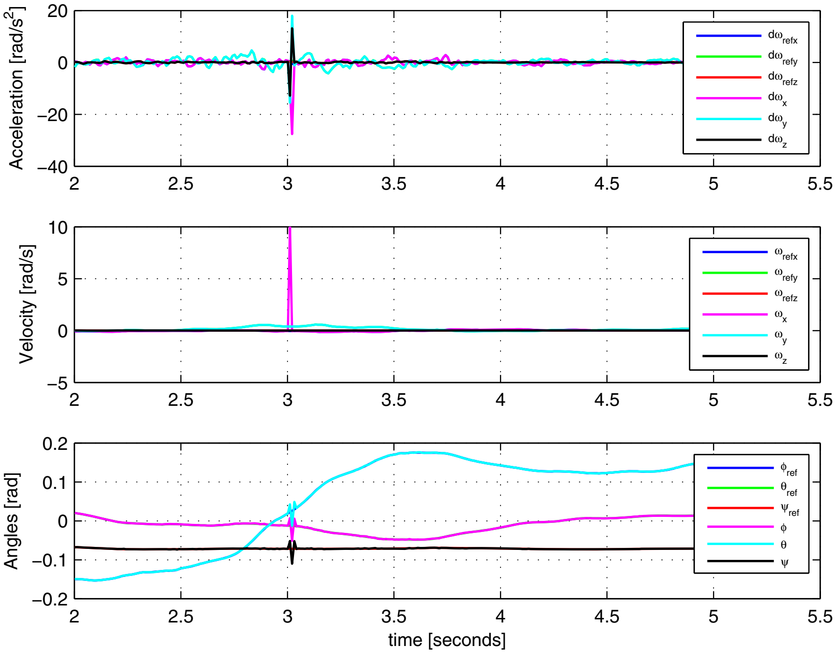
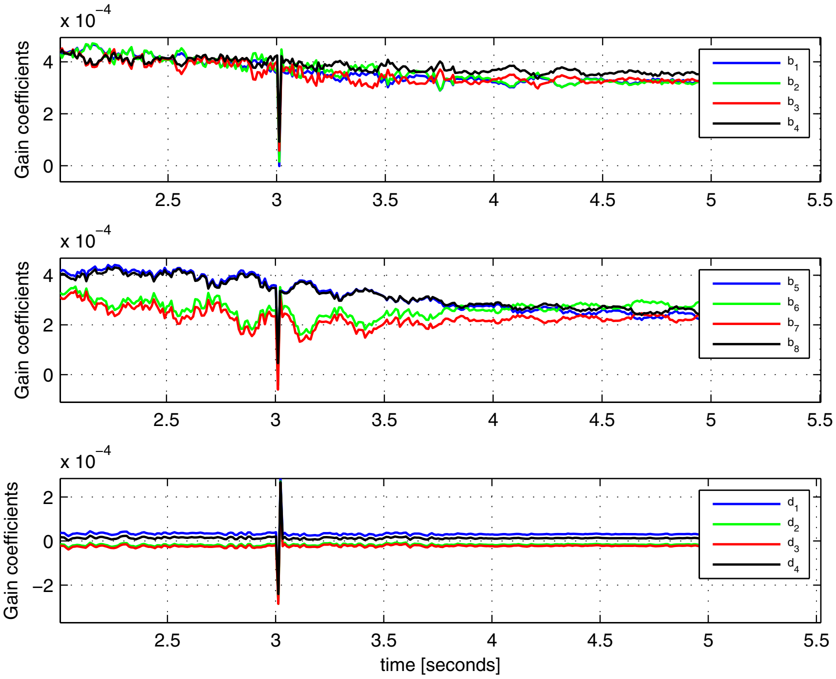


Fig. 20 Motor-Propellers gains b_i, d_i, for $\mathbf{R}_{ii}^{e} = 0.005$ with distorted velocity reference



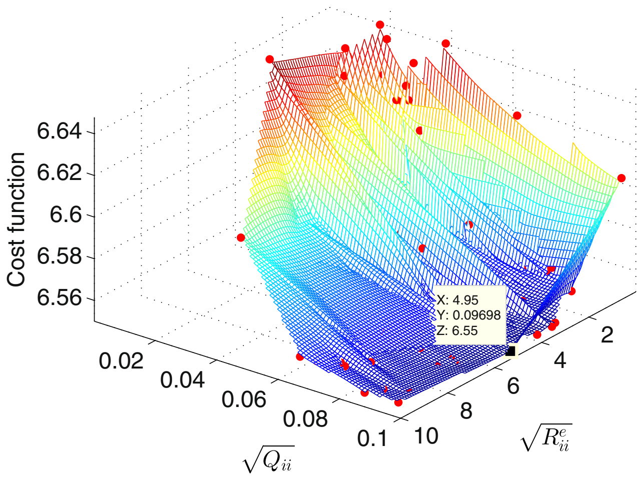


Fig. 21 Cost function versus $\mathbf{R}^e_{ii}$ and $\mathbf{Q}_{ii}$

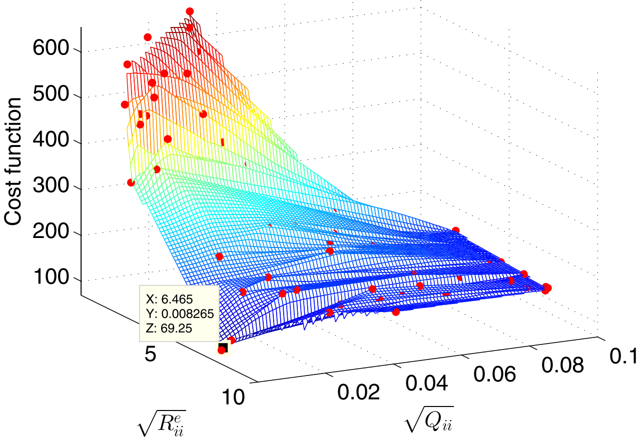


Fig. 22 Cost function versus $\mathbf{R}^e_{ii}$ and $\mathbf{Q}_{ii}$, for disrupted reference

Table 2 SRUKF results

$\sqrt{\mathbf{R}^e_{ii}}$	$\sqrt{\mathbf{Q}^e_{ii}}$	$err_{d\omega}$	err_{ω}	err_{ang}	err_{sum}	AI
5.0	0.00	0.4565	3.1548	141.90	145.51	–
5.0	0.01	0.4719	0.1058	22.164	22.742	–
0.005	0.01	1765.4	100.30	22.075	1887.8	*
5.0	0.01	111.32	18.816	22.249	152.39	*
150	0.01	2.3000	3.3209	22.600	28.221	*
5.0658	0.0978	0.4349	0.0155	22.072	22.523	**
6.4941	0.0077	53.999	9.3415	22.440	85.781	***
6.4941	0.0077	0.4732	0.1436	22.268	22.884	**

* disrupted

** PSO

*** PSO, disrupted

5 Discussion and Conclusions

In the article the method of coaxial quadrotor model parameter estimation was presented. The method is based on Square Root Unscented Kalman Filter which has better stability compared to Unscented Kalman Filter (UKF). Authors have shown the orientation model of coaxial quadrotor, extended with additional drag-like effects. In the experiment the data was gathered at a sport hall, from the AHRS system, applied in onboard measurement system. The AHRS provided an unbiased estimation of the angular velocity, angular acceleration and Tait-Bryan angles. The coaxial quadrotor has flown within the area, where the distance from the walls ceiling and floor was sufficiently high (the risk of reflected air was minimized), the drone operated in three axes, i.e. X, Y and Z.

In the paper, a complex analysis of the presented method was introduced. The main modifications provided by the authors concerned the λ_{RLS} and $\mathbf{S}^-_{w_k}$ upgrade. By introduction of the λ^0, the algorithm acts more stable, even though the input settings are within a wide range. On the other hand, carefully chosen and modified elements of the $\sqrt{\mathbf{Q}}$ allows to change the characteristics of the subsequent parameters. During the research, twenty-nine model parameters were estimated (based on nine reference signals). Into the reference signals, acceleration, velocity and angles for X, Y and Z axes signals were included. The basic SRUKF algorithm (without proposed modifications) has given poor results, as the offsets added in the model could not be followed fast enough. Later, dynamics of the offsets parameters were changed by increasing subsequent values in the $\sqrt{\mathbf{Q}}$. The results of the modified and improved SRUKF method were considerably better than in case of classic SRUKF. As it was revealed in a track of studies, the best performance is achieved when the $\sqrt{\mathbf{R}^e_{ii}}$ is substantially decreased. The SRUKF could get on a track of the reference signal fast and the errors were smallest, however the cost was extremely high, as small values of $\sqrt{\mathbf{R}^e_{ii}}$ lead to estimator's hyperactivity. This feature manifests as a sudden overreact on the noise or disruption in the reference data. The shown figures present how the estimator reacts on data disruption. When authors have introduced a peak value in the reference value, the parameters were dramatically changed, even though the changes were physically impossible (e.g. gains). Authors have distinguished two settings to be optimized. The first one was $\sqrt{\mathbf{Q}_{ii}}$, the second was $\sqrt{\mathbf{R}^e_{ii}}$. When increasing the $\sqrt{\mathbf{Q}_{ii}}$, the offsets parameters are estimated faster. On the other hand when $\sqrt{\mathbf{R}^e_{ii}}$ is decreased the algorithm is getting more sensitive on reference. Parameters, for which the $\sqrt{\mathbf{Q}_{ii}}$ is equal to zero, are influenced only by the $\sqrt{\mathbf{R}^e_{ii}}$. For parameters, which has $\sqrt{\mathbf{Q}_{ii}} > 0$ the behavior is determined by the ratio $\sqrt{\mathbf{Q}_{ii}}/\sqrt{\mathbf{R}^e_{ii}}$. The tuning process remains difficult as those parameters cannot be changed independently.

In order to find optimal settings, authors have used the PSO method. Two different cases were studied. The first one is a standard estimation, the second has intentional peak value in reference signal acting as a data disruption. The optimization has provided sets of tuning parameters, where the optimal reduced cost function markedly. Among many SRUKF setting parameters, the most important are λ_{RLS}, $\sqrt{\mathbf{R}^e}$ and $\sqrt{\mathbf{Q}}$, the rest like sigma points spread α, β and κ are less important. Authors could not record their influence on algorithm performance, as they play minor role (the actual influence was observed in the first milliseconds of the estimation, though the overall performance was similar).

One of the major advantages of the presented method is the natural ability of data tracking. The SRUKF can be applied in case of non-stationary models. The proved study shows how the estimation process can be set to individual parameters, where feature like rate of change are crucial in dynamic simulation. Furthermore, the given modified estimation method is suitable either in *offline* or *online* application. The tested batch method was validated in terms of parameters estimation accuracy, where the computational complexity factor was not essential, however this may become most important for *online* application. Two main operations which are executed in the SRUKF are: QR decomposition (complexity $\mathcal{O}(NL^2)$) and Cholesky factor update ($\mathcal{O}(L^2)$). The algorithm is in general $\mathcal{O}(L^2)$, which is less than in case of the EKF ($\mathcal{O}(L^3)$) [35]. As it has already been stated, presented coaxial quadrotor model had twenty-nine model parameters to be found. This lead to significant computational load, which for most of the on-board single-core computers may cause difficulties. Also, authors would like to emphasize the role of an implementation of the decomposition algorithm as well as the Cholesky factorization, which should be prepared to close to non-positive-definite matrix forms.

Finally authors would like to compare the obtained estimation results with the results from the PSO, presented in [16]. One of the most fundamental difference between the SRUKF and the PSO lies in the model structure. In the case of PSO, one is obliged to follow the rule of stationarity of the parameters, otherwise the method will not be able to find the solution. Technically, the PSO for parameter estimation is an off-line method, though one is able to introduce finite number of sets of parameters forming a switched model (with conditions for parameters to be switched). The second disadvantage of the PSO is computational inefficiency. As it was already stated, the method for large problems suffers from the local optima and in order to find best parameters, at least few swarms must be incorporated. The SRUKF uses the non-stationary model which is definitely better, especially in case of the coaxial quadrotor. As it was proved, the model parameters are fast-changing, which makes the SRUKF suitable for this group of models. It is important to note, that all of the parameters are required in model and further redundancy cannot be introduced. In the paper, authors have also demonstrated that the proposed framework (subsequent derivatives, model structure, data filtration, added offsets) is universal and can be applied in most of the multirotor UAVs. The improved SRUKF in coaxial quadrotor model parameters estimation has proved to have good performance.

Comparison of the proposed observer to other existing methods is essential for the future investigation, as this will help to decide whether the SRUKF is the most effective method for the given model and data. Authors would like to raise the issue in the upcoming paper. In the future, authors will also apply this method in a dual estimation, in simultaneous estimation of the quadrotor state vector and model parameters.

References

1. Adams, D.E., Allemang, R.J.: A new derivation of the frequency response function matrix for vibrating non-linear systems. J. Sound Vib. **227**(5), 1083–1108 (1999)
2. Alexis, K., Nikolakopoulos, G., Tzes, A.: Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances. Control Eng. Pract. **19**, 1195–1207 (2011)
3. Bangura, M., Mahony, R.: Nonlinear dynamic modeling for high performance control of a quadrotor. In: Australasian Conference on Robotics and Automation, pp. 1–10 (2012)
4. Brunke, S., Campbell, M.: Estimation architecture for future autonomous vehicles. In: Proceedings of the 2002 American Control Conference, vol. 2, pp. 1108–1114 (2002)
5. Chen, H., Wang, X., Li, Y.: A survey of autonomous control for UAV. In: International Conference on Artificial Intelligence and Computational Intelligence, vol. 2, pp. 267–271 (2009)
6. Chen, R.T., Eulrich, B.J., Lebacqz, J.V.: Development of advanced techniques for the identification of V/STOL aircraft stability and control parameters. Technical Report Cornell Aeronautical Laboratory (1971)
7. Chowdharya, G., Jategaonkarb, R.: Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter. Aerosp. Sci. Technol. **14**(2), 106–117 (2010)
8. Chowdharya, G., Lorenz, S.: Control of a vtol uav via online parameter estimation. In: Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, pp. 1—15 (2005)
9. Crassidis, J., Markley, F.L.: Unscented filtering for spacecraft attitude estimation. J. Guid. Control Dyn. **26**(4), 536–542 (2003)
10. Dorobantu, A., Murch, A., Mettler, B., Balas, G.: System identification for small, low-cost, fixed-wing unmanned aircraft. J. Aircraft. **50**(4), 1118–1130 (2013)

11. Finio, B.M., Perez-Arancibia, N.O., Wood, R.J.: System identification and linear time-invariant modeling of an insect-sized flapping-wing micro air vehicle. In: Proc. of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1107–1114 (2011)
12. Gelb, A., Kasper, J.F., Nash, R.A., Price, C.F., Sutherland, A.A.: Applied Optimal Estimation. MIT Press, Massachusetts (1974)
13. Giernacki, W., Gardecki, S., Goslinski, J.: Speed control of drive unit in four-rotor flying robot. In: Proc. of the 10th International Conference on Informatics in Control, Automation and Robotics, vol. 1, pp. 245–250 (2013)
14. Gośliński, J.: Application of Selected Methods of Kalman Filtration to Estimation of State and Model Parameters of the Flying Robot. PhD thesis. Poznan University of Technology, Poland (2016)
15. Goslinski, J., Gardecki, S., Giernacki, W.: An adequate mathematical model of four-rotor flying robot in the context of control simulations. J. Autom. Mob. Robot. Intell. Syst. **8**(2), 9–16 (2014)
16. Goslinski, J., Gardecki, S., Giernacki, W.: An efficient PSO-based method for an identification of a Quadrotor model parameters. In: Progress in Automation, Robotics and Measuring Techniques Advances in Intelligent Systems and Computing, vol. 351, pp. 95–104 (2015)
17. Goslinski, J., Gardecki, S., Giernacki, W.: Introduction of the flying robots into the human environment: An adaptive square-root unscented Kalman filter for a fault tolerant state estimation in a quadroto. Int. Conf. Intell. Environ., 117–123 (2014)
18. Goslinski, J., Nowicki, M., Skrzypczynski, P.: Performance comparison of EKF-based algorithms for orientation estimation on android platform. IEEE Sensors J. **15**(7), 3781–3792 (2015)
19. Guillaume, P., Verboven, P., Vanlanduit, S.: Frequency-domain maximum likelihood identification of modal parameters with confidence intervals. In: Proc. of the Int. Seminar on Modal Analysis, vol. 1, pp. 359–366 (1998)
20. Gupta, S.G., Ghonge, M.M., Jawandhiya, P.M.: Review of unmanned aircraft system (UAS). International Journal of Advanced Research in Computer Engineering and Technology (IJARCET). **2**(4), 1647–1658 (2013)
21. Gupte S., Mohandas, P.I.T., Conrad J.M.: A survey of quadrotor unmanned aerial vehicles. In: Proceedings of IEEE Southeastcon, pp. 1–6 (2012)
22. Hoffer, N.V., Coopmans, C., Jensen, A.M., Chen, Y.: A survey and categorization of small low-cost unmanned aerial vehicle system identification. J. Intell. Robot. Syst. **74**(1–2), 129–145 (2014)
23. Hoffmann, G.M., Huang, H., Waslander, S.L., Tomlin, C.J.: Quadrotor helicopter flight dynamics and control: theory and experiment. In: Proceedings of the AIAA Guidance, Navigation, and Control Conference, pp. 1–17 (2007)
24. Huang, H., Yang, M., Zang, W., Wu, S., Pang, Y.: In vitro identification of four-element Windkessel models based on iterated unscented Kalman filter. IEEE Trans. Biomed. Eng. **58**(9), 2672–2680 (2011)
25. Jategaonkar, R.V., Plaetschke, E.: Algorithms for aircraft parameter estimation acconting for process and measurement noise. J. Aircraft. **26**(4), 360–372 (1989)
26. Jategaonkar, R.V., Thielecke, F.: Aircraft parameter estimation – a tool for development of aerodynamic databases. Sadhana **25**(2), 119–135 (2000)
27. Julier, S.J., Uhlmann, J.K.: A new extension of the Kalman filter to nonlinear systems. In: Proc. SPIE 3068, Signal Processing, Sensor Fusion, and Target Recognition VI, vol. 182, pp. 1–12 (1997)
28. Kallapur, A.G., Ali, S.S., Anavatti, S.G.: Application of extended Kalman filter towards UAV identification, autonomous robots and agents. In: Studies in Computational Intelligence (SCI), vol. 76, pp. 199–207. Springer (2007)
29. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. ASME — J. Basic Eng. D **82**, 35–45 (1960)
30. Kanistras, K., Martins, G., Rutherford, M.J., Valavanis, K.P.: A survey of unmanned aerial vehicles (UAVs) for traffic monitoring. In: Proceedings of International Conference on Unmanned Aircraft Systems (ICUAS), pp. 221–234 (2013)
31. Kumar, V., Mahony, R., Corke, P.: Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. IEEE Robot. Autom. Mag. **19**(3), 20–32 (2012)
32. Kwanghee, N., Araposthathis, A.: A model reference adaptive control scheme for pure-feedback nonlinear systems. IEEE Trans. Autom. Control. **33**(9), 803–811 (1988)
33. Lutz, T., Gu, C., Gardecki, S., Cordes, F., Homingo, T., Böljes, A., Maagt, P.: StarTiger dropter project: Development and flight experiment of a skycrane-like terrestrial lander demonstrator. In: The 9th International ESA Conference on Guidance, Navigation and Control Systems (2014)
34. Mahony, R., Kumar, V., Corke, P.: Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. IEEE Robot. Autom. Mag. **19**(3), 20–32 (2013)
35. Merwe, R., Wan, E.: The square-root unscented Kalman filter for state and parameters-estimation. In: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 6, pp. 3461–3465 (2001)
36. Mettler, B., Tischler, M.B., Kanade, T.: System identification of small-size unmanned helicopter dynamics. In: Annual Forum Proceedings – American Helicopter Society, pp. 1706–1717 (1999)
37. Noel, J.P., Kerschen, G.: Frequency-domain subspace identification for nonlinear mechanical systems. Mech. Syst. Signal Process. **40**, 701–717 (2013)
38. Noel, J.P., Marchesiello, S., Kerschen, G.: Subspace-based identification of a nonlinear spacecraft in the time and frequency domains. Mech. Syst. Signal Process. **43**, 217–236 (2014)
39. Park, P.G., Kailath, T.: New square-root algorithms for Kalman filtering. IEEE Trans. Autom. Control. **40**(5), 895–899 (1995)
40. Rathinam, S., Almeida, P., Kim, Z., Jackson, S., Tinka, A., Grossman, W., Sengupta, R.: Autonomous searching and tracking of a river using an UAV. In: Proceedings of the 2007 American Control Conference, pp. 359–364 (2007)
41. Speyer, J.L., Crues, E.Z.: On-line aircraft state and stability derivative estimation using the modified-gain extended Kalman filter. J. Guida. Control. **10**(3), 262–268 (1987)
42. St-Pierre, M., Gingras, D.: Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system. In: Proccedings of the IEEE Intelligent Vehicles Symposium, pp. 831–835 (2004)
43. Tischler, M.B.: System identification methods for aircraft flight control development and validation. NASA Technical Memorandum 110369 USAATCOM Technical Report 95-A-007 (1996)
44. Wendel, J., Maier, A., Metzger, J., Trommer, G.F.: Comparison of extended and sigma-point Kalman filters for tightly coupled GPS/INS integration. In: AIAA Guidance, Navigation, and Control Conference and Exhibit, pp. 1–15 (2005)
45. Yang, L., Liu, J.: Parameter identification for a quadrotor helicopter using (PSO). In: Proc. of the 52nd IEEE Conference on Decision and Control, pp. 5828—5833 (2013)

Jarosław Gośliński received the M.Sc. degree in control engineering and robotics from Poznan University of Technology (Poland) in 2011 and Ph.D. degree in 2016. His research interests include signal filtering and processing, MEMS sensors, state observers, nonlinear state estimation and identification, swarm robotics and artificial intelligence methods. Currently working with Polish Aerial Robotics Team (PART) on UAVs, especially quadrotors.

Andrzej Kasiński graduated from the Poznan University of Technology (1973) and Adam Mickiewicz University (1974). He received the Ph.D. and D.Sc. degrees in Robotics from the same University in 1979 and 1998, respectively. Director of Institute of Control, Robotics and Information Engineering. Lectured at several foreign Universities: TU Delft, Universidad de Murcia, among others. He is specializing in computer vision systems, spiking neural networks and control theory. A. Kasiński is the author or co-author of over 150 technical papers in the fields of robotics.

Wojciech Giernacki received Ph.D. degree in control engineering and robotics from Poznan University of Technology in 2011 and master degree in 2006. Assistant professor at the Institute of Control, Robotics and Information Engineering of the same university. His scientific interests are focused around the issues of UAVs, especially robust and adaptive control, optimization techniques, as well as data fusion from sensors.

Piotr Owczarek is a Ph.D. student and assistant in Division of Mechatronics Devices at Poznan University of Technology. He graduated from electrical engineering, robotics and control engineering in 2011. His research interests include modern methods of digital image processing, artificial intelligence methods, design of electronic and mechatronic devices, mobile robots and industrial controllers.

Stanislaw Gardecki received the M.Sc. degree in control engineering and robotics from Poznan University of Technology (Poland) in 2009 and Ph.D. degree in 2017. Assistant at Institute of Control, Robotics and Information Engineering at the same University. Winner of numerous robots competitions. His research interests are within the scope of mobile robots constructions and their control algorithms. Currently working on quadrotors, designed for different purposes in urban spaces.