EXTRACTING DECK TREND TO PLAN DESCENT TRAJECTORY FOR SAFE LANDING OF UAVS

Xilin Yang , Luis Mejias Alvarez
Australian Research Center for Aerospace Automation and
Queensland University of Technology, Brisbane, Australia.
xilin.yang@qut.edu.au; luis.alvarez@qut.edu.au

Keywords: *Rotary-wing unmanned aerial vehicle; Extended Kalman filter; Prony Analysis*

Abstract

This paper outlines a feasible scheme to extract deck trend when a rotary-wing unmanned aerial vehicle (RUAV) approaches an oscillating deck. An extended Kalman filter (EKF) is developed to fuse measurements from multiple sensors for effective estimation of the unknown deck heave motion. Also, a recursive Prony Analysis (PA) procedure is proposed to implement online curve-fitting of the estimated heave motion. The proposed PA constructs an appropriate model with parameters identified using the forgetting factor recursive least square (FFRLS) method. The deck trend is then extracted by separating dominant modes. Performance of the proposed procedure is evaluated using real ship motion data, and simulation results justify the suitability of the proposed method into safe landing of RUAVs operating in a maritime environment.

1 Introduction

The RUAVs have attracted increasing interest over the past few decades due to their suitability for a variety of flight missions [11]. RUAVs also have the potential to enable a range of new maritime applications if automatic recovery operations from ships in rough seas are achieved reliably [18].

The present work is aimed at landing a RUAV on an oscillating platform in severe sea environments. For piloted helicopters, a safe landing can be completed successfully since the pilot is able to observe instantaneous relative motion, judge the level of risk, arrange a feasible approaching trajectory and trigger the proper moment for landing and touchdown. Besides, any potential threat during the landing operation can be minimized through executing necessary manipulations based on pilot experience. In contrast, the autonomy of the RUAV requires the automatic implementation of the entire process without involvement of pilots. This indicates that the flight control computer should be able to complete the guidance and control tasks with safety guarantees, which significantly exacerbates the implementation difficulties.

Safe landing of a RUAV also requires reliable navigation capability, which comes from the vulnerability of the RUAV to impact forces during touchdown in a maritime environment. Integrated navigation system makes great efforts to take advantage of complementary attributes of multiple sensors, and has been discussed in a number of papers for a wide variety of applications [3, 17]. Faruqi and Turner [3], proposed and evaluated the performance of a linearized integrated GPS/INS model with an extended Kalman filter for a typical aerospace application. The sensitivity analysis was also conducted to determine the optimal filter parameters. Jan *et al.* [17] developed an integrated navigation system to provide attitude information with sufficient accuracy for a quadrotor helicopter. The navigation system aimed to ensure the optimal usage of the GPS

measurements. The unscented Kalman filter was also applied to some scenarios to improve the navigation performance [10, 8]. In the considered application, we are targeting a feasible filtering approach which can be implemented easily at the cost of limited flight computer memory and provide sufficient estimation accuracy. Also, due to the fact that introduction of high-order system dynamics does not generally lead to an improvement in system performance [4], we use the EKF in this paper to perform the sensor fusion task.

Another requirement for automatic landing is an accurate estimation of the mean deck height so that a smooth descent trajectory can be arranged. This will help the RUAV to track the mean deck height as opposed to the instantaneous deck displacement, thus greatly reducing the risk of touching down the deck with fatal impact forces. There are several methodologies available to construct models for identifying the system trend, e.g., linear model [19], weighted exponential model [9] and polynomial model [5]. Prony Analysis is a branch of parametric curve-fitting techniques with model parameters giving magnitude, frequency and phase information about system dynamics, and has been used to analyze power systems [6, 12]. In the considered application, we are aimed at designing an online procedure to extract dominant modes in the oscillating deck.

This paper is organized as follows: in Section 2, we describe the procedure to design a multi-sensor fusion algorithm using the EKF. The proposed recursive PA is introduced in Section 3. Section 4 provides the simulation results obtained, and finally brief conclusions are presented in Section 5.

2 Sensor Fusion using the EKF

This section aims to develop a sensor fusion algorithm by using the EKF technique. Three different sensors (Inertial Navigation Sensor (INS), Global Positioning System (GPS) and visual tracking sensor) are utilized to perform the navigation task. The EKF is designed to smooth out noisy measurements from INS and GPS. It also serves to estimate the unknown heave motion of the landing deck. The EKF fuses the following groups of measurements: RUAV position (x_h, y_h, z_h) and velocity (v_{xh}, v_{yh}, v_{zh}) given by the GPS, relative motion (α_r, β_r, d_r) from the tracking sensor, accelerations (a_x, a_y, a_z) and angular rates (p, q, r) from the INS. The velocity (u, v, w) in the body frame is related to velocity (v_{xh}, v_{yh}, v_{zh}) in the navigation frame by the direction matrix C_b^n

$$[v_{xh}, v_{yh}, v_{zh}]^T = C_b^n [u, v, w]^T \tag{1}$$

with C_b^n expressed in quaternion parameters [15]

$$C_b^n = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

$$= \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix}$$

Here, quaternion elements $\boldsymbol{q} = [q_0, q_1, q_2, q_3]^T$ are employed to describe the attitude information. The singular problems encountered when attitudes are expressed in Euler forms can be avoided via adoption of the quaternion form.

Design of the EKF begins with a description of the state update model given by

$$X_k = f(X_{k-1}, k-1) + \varepsilon_k \tag{2}$$

where state vector X corresponds to the following state variables

$$X = [x_h, y_h, z_h, u, v, w, x_s, y_s, z_s, v_{xs}, v_{ys}, v_{zs}, x_r, y_r, z_r, \psi_s, V_{\psi s}]^T \tag{3}$$

and system noise (mutually independent with Gaussian distributions) is $\varepsilon = [\varepsilon_1, \cdots, \varepsilon_{17}]^T$. Here, RUAV positions (x_h, y_h, z_h), ship positions (x_s, y_s, z_s) and velocities (v_{xs}, v_{ys}, v_{zs}), and relative positions (x_r, y_r, z_r) are described in navigation frame. Ship yaw and yaw rate are denoted by ψ_s and $V_{\psi s}$. In the considered application, ship heading (yaw) is known from radio signals sent by the ship. Equation (2) can be expressed in an

explicit form

$$(x_h)_k = (x_h)_{k-1} + T_s[(c_{11})_{k-1}u_{k-1} + (c_{12})_{k-1}v_{k-1} + (c_{13})_{k-1}w_{k-1}] + (\varepsilon_1)_k \quad (4)$$

$$(y_h)_k = (y_h)_{k-1} + T_s[(c_{21})_{k-1}u_{k-1} + (c_{22})_{k-1}v_{k-1} + (c_{23})_{k-1}w_{k-1}] + (\varepsilon_2)_k \quad (5)$$

$$(z_h)_k = (z_h)_{k-1} + T_s[(c_{31})_{k-1}u_{k-1} + (c_{32})_{k-1}v_{k-1} + (c_{33})_{k-1}w_{k-1}] + (\varepsilon_3)_k \quad (6)$$

$$u_k = u_{k-1} + T_s[r_{k-1}v_{k-1} - q_{k-1}w_{k-1} + (a_x)_{k-1}] + (\varepsilon_4)_k \quad (7)$$

$$v_k = v_{k-1} + T_s[-r_{k-1}u_{k-1} + p_{k-1}w_{k-1} + (a_y)_{k-1}] + (\varepsilon_5)_k \quad (8)$$

$$w_k = w_{k-1} + T_s[q_{k-1}u_{k-1} - p_{k-1}v_{k-1} + (a_z)_{k-1}] + (\varepsilon_6)_k \quad (9)$$

$$(x_s)_k = (x_s)_{k-1} + T_s(v_{xs})_{k-1} + (\varepsilon_7)_k \quad (10)$$

$$(y_s)_k = (y_s)_{k-1} + T_s(v_{ys})_{k-1} + (\varepsilon_8)_k \quad (11)$$

$$(z_s)_k = (z_s)_{k-1} + T_s(v_{zs})_{k-1} + (\varepsilon_9)_k \quad (12)$$

$$(v_{xs})_k = (v_{xs})_{k-1} + (\varepsilon_{10})_k \quad (13)$$

$$(v_{ys})_k = (v_{ys})_{k-1} + (\varepsilon_{11})_k \quad (14)$$

$$(v_{zs})_k = (v_{zs})_{k-1} + (\varepsilon_{12})_k \quad (15)$$

$$(x_r)_k = (x_r)_{k-1} + (\varepsilon_{13})_k \quad (16)$$

$$(y_r)_k = (y_r)_{k-1} + (\varepsilon_{14})_k \quad (17)$$

$$(z_r)_k = (z_r)_{k-1} + (\varepsilon_{15})_k \quad (18)$$

$$(\psi_s)_k = (\psi_s)_{k-1} + T_s(V_{\psi s})_{k-1} + (\varepsilon_{16})_k \quad (19)$$

$$(V_{\psi s})_k = (V_{\psi s})_{k-1} + (\varepsilon_{17})_k \quad (20)$$

In the state update equations mentioned above, the sampling time is denoted by T_s. System noise $\varepsilon_{(\cdot)}$ and covariance matrix of system noise $Q_{(\cdot)}$ satisfies

$$E\{\varepsilon^i_{(\cdot)}[\varepsilon^j_{(\cdot)}]^T\} = \delta(i-j)Q_{(\cdot)} \quad (21)$$

where δ is Kronec function taking the form of

$$\delta(i-j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \quad (22)$$

Equations (4)-(6) describe relationship of velocities between body frame and navigation frame. Local velocity propagation is revealed in Eq. (7)-(9) with knowledge of accelerations (a_x, a_y, a_z). In practice, it is impossible to build up an accurate ship motion model as many parameters are required for specific ship/helicopter combinations. However, it is reasonable to assume ship speed remains constant in forward and sideways directions during landing phase, as is shown in Eq. (10)-(15). The heave motion greatly affects magnitude of the impact force during touchdown moment. In the model, deck displacement speed is tentatively set to be constant, and it will be demonstrated later that the EKF is able to estimate the displacement motion effectively as relative motion (α_r, β_r, d_r) provided by the tracking sensor can correct the estimation performance.

Nonlinearities in system model Eq. (2) stem from the quaternion components c_{ij} in direction matrix C^n_b, which should be linearized when deriving the state transition matrix. The direction matrix C^n_b can be described in terms of angular rates after using the first-order approximation method [4], and can be converted to Eq. (23) (see next page).

Substituting Eq. (23) into Eq. (4)-(6) leads to Eq. (24)-(26)(see next page). Here, the angular rates at time instant k are described by p_k, q_k, r_k. In our case, the body rate information obtained from the INS has been filtered and can be used for sensor fusion with sufficient accuracy. Angular rates (p_k, q_k, r_k) do not remain constant and keep updating when measurements from the INS change.

The measurement model can be described by

$$Z_k = h(X_k, k) + \xi_k \quad (27)$$

where measurements are

$$Z = [x_h, y_h, z_h, v_{xh}, v_{yh}, v_{zh}, \alpha_r, \beta_r, d_r, \psi_s]^T \quad (28)$$

and measurement noise ε is $\xi = [\xi_1, \ldots, \xi_{10}]^T$.

$$C_b^n = \begin{bmatrix} 1+\tilde{P}^2-\tilde{Q}^2-\tilde{R}^2 & 2(\tilde{P}\tilde{Q}-\tilde{R}) & 2(\tilde{P}\tilde{R}+\tilde{Q}) \\ 2(\tilde{P}\tilde{Q}+\tilde{R}) & 1-\tilde{P}^2+\tilde{Q}^2-\tilde{R}^2 & 2(\tilde{Q}\tilde{R}-\tilde{P}) \\ 2(\tilde{P}\tilde{R}-\tilde{Q}) & 2(\tilde{Q}\tilde{R}+\tilde{P}) & 1-\tilde{P}^2-\tilde{Q}^2+\tilde{R}^2 \end{bmatrix} \tag{23}$$

$$(x_h)_k = (x_h)_{k-1} + T_s[(1+\frac{(p_{k-1}^2-q_{k-1}^2-r_{k-1}^2)T_s^2}{4})u_{k-1} + 2(\frac{p_{k-1}q_{k-1}T_s^2}{4} - \frac{r_{k-1}T_s}{2})v_{k-1} + 2(\frac{q_{k-1}r_{k-1}T_s^2}{4} + \frac{q_{k-1}T_s}{2})w_{k-1} + (\varepsilon_1)_k] \tag{24}$$

$$(y_h)_k = (y_h)_{k-1} + T_s[2(\frac{p_{k-1}q_{k-1}T_s^2}{4} + \frac{r_{k-1}T_s}{2})u_{k-1} + (1+\frac{(q_{k-1}^2-p_{k-1}^2-r_{k-1}^2)T_s^2}{4})v_{k-1} + 2(\frac{q_{k-1}r_{k-1}T_s^2}{4} - \frac{p_{k-1}T_s}{2})w_{k-1}] + (\varepsilon_2)_k] \tag{25}$$

$$(z_h)_k = (z_h)_{k-1} + T_s[2(\frac{p_{k-1}r_{k-1}T_s^2}{4} - \frac{q_{k-1}T_s}{2})u_{k-1} + 2(\frac{q_{k-1}r_{k-1}T_s^2}{4} + \frac{p_{k-1}T_s}{2})v_{k-1} + (1+\frac{(r_{k-1}^2-p_{k-1}^2-q_{k-1}^2)T_s^2}{4})w_{k-1}] + (\varepsilon_3)_k \tag{26}$$

The detailed measurement equations are

$$(x_h)_k = (x_h)_k + (\xi_1)_k \tag{29}$$
$$(y_h)_k = (y_h)_k + (\xi_2)_k \tag{30}$$
$$(z_h)_k = (z_h)_k + (\xi_3)_k \tag{31}$$
$$u_k = u_k + (\xi_4)_k \tag{32}$$
$$v_k = v_k + (\xi_5)_k \tag{33}$$
$$w_k = w_k + (\xi_6)_k \tag{34}$$
$$(\alpha_r)_k = \arctan\{\frac{(y_r)_k}{(x_r)_k}\} + (\xi_7)_k \tag{35}$$
$$(\beta_r)_k = \arccos\{\frac{(z_r)_k}{\sqrt{[(x_r)_k]^2+[(y_r)_k]^2+[(z_r)_k]^2}}\} + (\xi_8)_k \tag{36}$$
$$(d_r)_k = \sqrt{[(x_r)_k]^2+[(y_r)_k]^2+[(z_r)_k]^2} + (\xi_9)_k \tag{37}$$
$$(\psi_s)_k = (\psi_s)_k + (\xi_{10})_k \tag{38}$$

Measurement noise $\xi_{(\cdot)}$ is mutually independent with Gaussian distributions, and covariance matrix of measurement noise $R_{(\cdot)}$ satisfies $E\{\xi_{(\cdot)}^i[\xi_{(\cdot)}^j]^T\} = \delta(i-j)R_{(\cdot)}$

Given the system update model and measurement update model, an EKF can be developed to fulfill the sensor fusion task by following the procedure in [2].

3 Recursive Prony Analysis

This section starts with an introduction to the traditional PA followed by a description of the proposed recursive PA. The heave motion estimated from Section 2 is assumed to sufficiently accurate such that it can be considered as the true heave motion.

A weighted linear combination of exponential terms $\hat{y}$ is used to approximate the heave motion sequence

$$\hat{y}(kT_s) = \sum_{i=1}^{n} B_i z_i^k,\ z_i = e^{\lambda_i T_s},\ k = 0,\ldots,N-1 \tag{39}$$

where each residue B_i corresponds to its complex pole $\lambda_i, i = 1,\ldots,n$. The model order is denoted by n. the complex number z_i is termed the discrete-time system pole, and N the number of samples. Our objective is to determine B_i, λ_i and n, such

that $\hat{y}(kT_s)$ is good approximation to the heave motion $y(k)$ in the least square sense.

Procedure of the traditional PA can be divided into the following three steps.

Step 1: The PA begins with constructing the linear prediction model (LPM) [12]

$$y(k) = a_1 y(k-1) + \cdots + a_n y(k-n) \tag{40}$$

and determining coefficients $a_i,\ i = 1, ..., n$.

Step 2: A matrix representation of sequential samples is constructed by expanding the LPM at various time instants, and coefficients a_i are acquired by solving the following equations using least square (LS) method,

$$D = QA; \quad D = [y(n), y(n+1), ..., y(N)]^T \tag{41}$$

$$D = [y(n), y(n+1), ..., y(N)]^T \tag{42}$$

$$Q = \begin{bmatrix} y(n-1) & y(n-2) & \cdots & y(0) \\ y(n) & y(n-1) & \cdots & y(1) \\ \vdots & \vdots & \vdots & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-n) \end{bmatrix} \tag{43}$$

$$A = [a_1, a_2, ..., a_n]^T \tag{44}$$

The corresponding characteristic equation can be derived from coefficients a_i, which is given by

$$z^n - a_1 z^{n-1} - \cdots - a_{n-1} z - a_n = \prod_{i=1}^{n} (1 - z \cdot z_i^{-1}) \tag{45}$$

The zeros z_i appear only in the form of real numbers or complex conjugate pairs since a_i are real in Eq. (45). If z_i is completely real, then [7]

$$\lambda_i = \frac{\ln z_i}{T_s} \tag{46}$$

Otherwise, if z_i is a complex conjugate pair,

$$\lambda_i = \alpha_i \pm j\beta_i \tag{47}$$

$$\alpha_i = \frac{\ln |z_i|}{T_s},\ \beta_i = \frac{1}{T_s} \tan^{-1}\{\frac{z_{Ii}}{z_{Ri}}\} \tag{48}$$

where $z_i = z_{Ri} \pm j \cdot z_{Ii}$.

Step 3: The residues B_i are obtained through solving the following linear equations

$$Y = \Psi B \tag{49}$$

$$Y = [y(0), y(1), ..., y(N-1)]^T \tag{50}$$

$$\Psi = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_n^1 \\ \vdots & \vdots & \vdots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_n^{N-1} \end{bmatrix} \tag{51}$$

$$B = [B_1, B_2, ..., B_n]^T \tag{52}$$

Here, the Vandermonde matrix Ψ is constructed based on the zero z_i of characteristic equation (45). Normally, if zeros z_i appear in conjugate pairs, the corresponding B_i will also appear in conjugate forms.

Remark 1 *The fundamental limitation of the PA lies in inverting the large matrices in step 1 and 3. In our case, this involves dealing with a large number of samples, and greatly exacerbates the difficulties in real-time implementation. Also, manipulation of matrix inversion may suffer from singularity issues. Ill-conditioned matrices may occur when inverting large matrices, which would cause the PA to fail.*

Remark 2 *The LS only deals with the measurements for a separate sliding window, and starts estimation without consideration of information in the previous data window. Therefore, estimation of instantaneous mean is subject to great changes when successive data windows are processed.*

To remedy the weakness of the PA, the following factors are considered in the recursive PA:

1. How to obtain accurate and reliable model parameters when new measurements are collected?

2. How to carry forward system information for successive sliding windows to achieve an accurate estimation?

3. How to reduce computational burden to accomplish a rapid online estimation of monotonous trend?

Remark 3 *Regarding the first question, the FFRLS is employed which introduces a forgetting factor to discard the effect of old measurements and highlight contributions of the most recent measurements to system dynamics [16].*

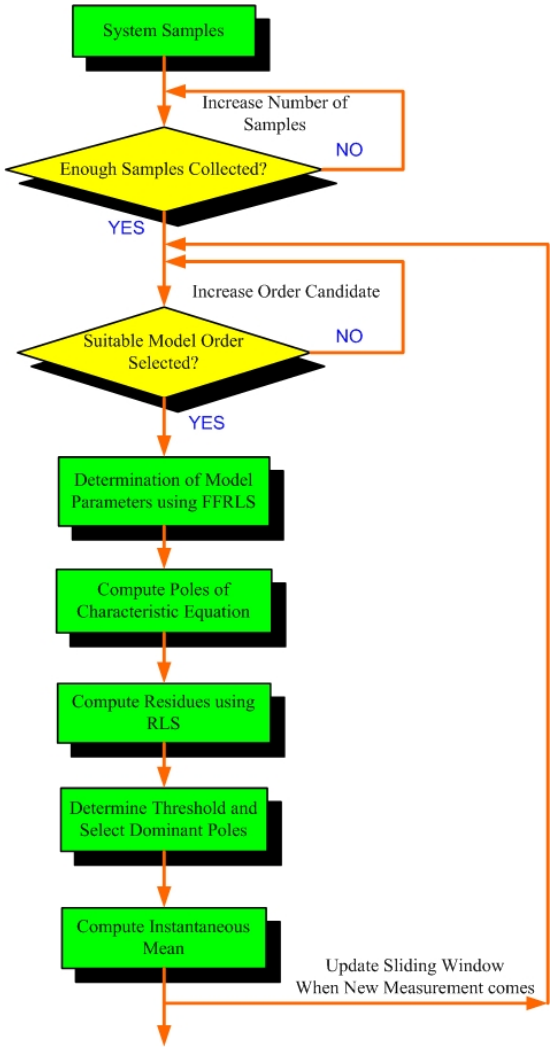

Fig. 1 Flowchart for extracting instantaneous mean

To implement the FFRLS, the vector of lagged sampled data

$$\varphi(t) = [y(t-1), ..., y(t-n)]^T \tag{53}$$

and coefficient vector $\hat{\theta}(t) = [\hat{a}_1(t), ..., \hat{a}_n(t)]^T$ up to time instant t is introduced. Coefficients $\hat{a}_1(t), ..., \hat{a}_n(t)$ are updated recursively to approach the real values $a_1, ..., a_n$. The LPM can be written in a compact form

$$y(t) = \hat{\theta}^T(t)\varphi(t) \tag{54}$$

The estimation of model coefficients $\hat{\theta}(t)$ can be obtained after minimizing the loss function [14]

$$\hat{\theta}(t) = [\sum_{j=1}^{t} \gamma^{t-j}\varphi(j)\varphi^T(j)]^{-1}[\sum_{j=1}^{t} \gamma^{t-j}\varphi(j)y(j)] \tag{55}$$

The FFRLS can be implemented recursively by

$$\hat{\theta}(t+1) = \hat{\theta}(t) + K(t+1)[y(t+1) - \varphi^T(t+1)\hat{\theta}(t)] \tag{56}$$

$$K(t+1) = P(t)\varphi(t+1)[\gamma + \varphi^T(t+1)P(t)\varphi(t+1)]^{-1} \tag{57}$$

$$P(t+1) = [P(t) - K(t+1)\varphi^T(t+1)P(t)]/\gamma \tag{58}$$

$$\hat{\theta}(0) = 0 \quad P(0) = \alpha I \tag{59}$$

Here, matrix $P(t+1)$ is referred to as error covariance matrix, matrix $K(t+1)$ denotes the update matrix, and α is a large positive number.

Remark 4 *Regarding the second question in Remark 2, the error covariance $P(t)$ and model coefficients $\hat{\theta}(t)$ are initialized once for the first sliding window, then the FFRLS carries them forward as the sliding window moves to the next. This implies the model coefficient vector is slow-varying and its components for adjacent data windows are closely related.*

Remark 5 *Step 3 of the PA can be followed according to Eq. (49)-(52). The recursive least square (RLS) is used to estimate the magnitude B_i, and the vector of sampled data $\varphi(t)$ in Eq. (53) should be replaced with $\rho(t) = [z_1^{t-1}, ..., z_n^{t-1}]^T$ which corresponds to row components in Eq. (51). It should be noticed that the vector $\rho(t)$ at different time instants t varies greatly. Carrying forwards error covariance matrix and coefficient vector is not proper in this step, as vector $\rho(t)$ for different time instants are not slow-varying.*

3.1 Determination of model order

Some available information criteria for order determination are Akaike Information Criterion (AIC) [1] and Bayes Information Criterion (BIC) [13]. AIC and BIC aim to make a trade-off between estimation errors accumulated and model complexity, and the optimal order is determined when they reach the minimum. However, when used to determine the model order, AIC and BIC consistently decrease when model order becomes large, and the estimation performance does not

deteriorate. Underlying this fact is that the extra exponential terms are trying to fit the noise effect.

Practically, the model order of PA should be selected such that a trade-off is achieved between estimation accuracy and computational burden. The proper model order can be found out by checking the summed squared error (SSE)

$$SSE = \sum_{k=0}^{N-1} [y(k) - \sum_{i=1}^{n} B_i e^{\lambda_i k}]^2 \qquad (60)$$

The order selection procedure consists of three steps:

1. Set the predicted model order R_N which is larger than the maximum number of model order which is expected;

2. Determine the model order n_l out of predicted order R_N such that there is a significant drop in SSE when the LPM is constructed by n_l exponential terms. This gives a lower bound of acceptable model order;

3. Calculate the computational burden when order is larger than n_l. The proper model order is chosen when a balance between the match accuracy and the computational burden is achieved.

Remark 6 *In practice, if the predefined curve-fitting match accuracy is satisfied, the Prony model with small order is preferred to reduce the computational burden. In situations where computational burden is a minor factor, the proper model order can be found out when there is a significant drop in SSE.*

3.2 Dominant mode selection criterion

The proposed dominant mode selection criterion begins with defining a suitable threshold. The coefficients B_i with respect to the poles within the threshold are taken as dominant residues. The threshold is chosen according to the following criterion:

1. Choose the pole with its negative real part closest to the imaginary axis, which corresponds to the smallest horizontal distance d;

2. The threshold L_p is 5 times of the horizontal distance d;

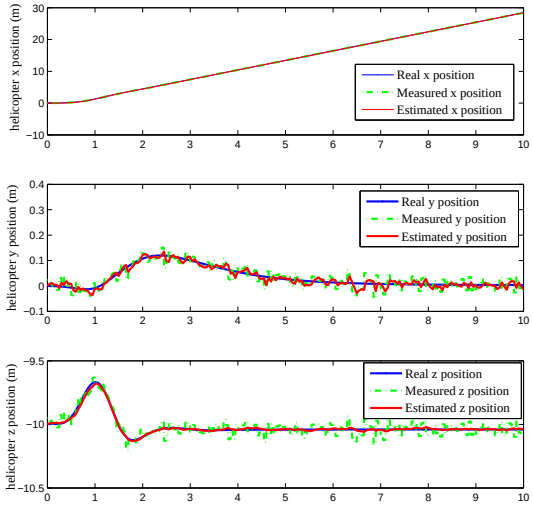

Fig. 2 Estimation of RUAV positions using the EKF

3. The width of the box threshold W_p depends on the magnitude of rounding errors, which takes a very small value $(O(e^{-8}))$.

The system trend $\bar{y}_{ins}$ can be expressed as

$$\bar{y}_{ins} = \sum_{i=1}^{W} B_{Di} e^{\lambda_{Di}(N-1)T_s} \qquad (61)$$

where λ_{Di} is dominant pole, and B_{Di} is the corresponding residue. The number of dominant poles is denoted by W.

The flowchart for online estimation of monotonous trend of an oscillating deck is depicted in Fig. 1. The proposed approach firstly collects enough samples. Then, model order is found out based on the SSE in consideration of the computational burden. Model parameters are specified using the FFRLS. Also, poles of characteristic equation are computed. Afterwards, the corresponding residues B_i are calculated using the RLS. The system trend is obtained after selecting the dominant poles and residues using Eq. (61).

4 Simulation Results

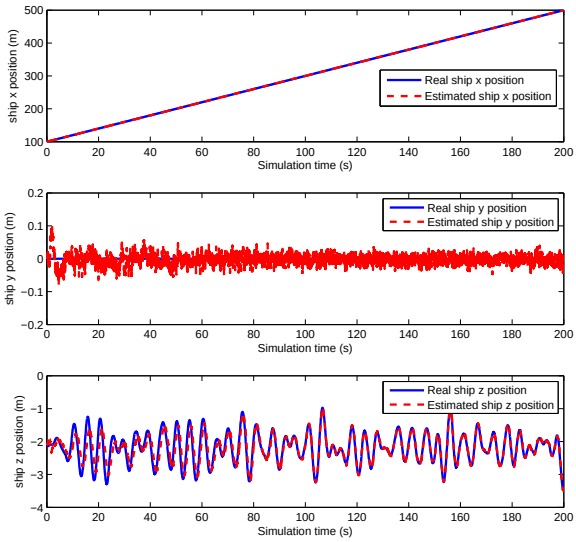

Fig. 3 Estimation of ship positions using the EKF

4.1 Evaluation of the Sensor Fusion Algorithm

The EKF algorithm is tested using real-time deck motion data. In simulations, the RUAV is commanded to follow the middle line of the ship, approach the deck at a constant speed of $3m/s$, and hover at a height of 10 m. For the NovAtel GPS receiver on our helicopter, the distance accuracy is 2 cm in the longitudinal-lateral plane and 4 cm in the elevation. Thus, white noise with standard deviations of $2cm$, $2cm$ and $4cm$ were added to real positions of the RUAV to test the performance of the EKF. Also, azimuth angle α_r and elevation angle β_r were contaminated by white noise with standard deviations of 0.18^o. This agrees with the noise levels in our visual tracking sensor.

Performance of the EKF when applied to estimate positions of the RUAV is shown in Fig. 2. For the sake of observation convenience, estimation results for the first 10 s are plotted. It is noticed that noise effects in positions are attenuated efficiently. Also, the unknown ship positions are estimated accurately, as shown in Fig. 3. It takes around 80 s for the EKF to capture the system dynamics accurately. In particular, deck displacement is estimated smoothly, which greatly contributes to extracting instantaneous mean deck position for landing operations.

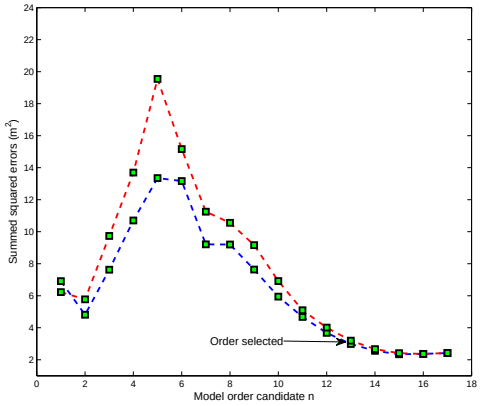

Fig. 4 Summed squared errors for different model orders

4.2 Extracting the trend of real deck displacement

We firstly seek to collect adequate length of estimated displacement data. The length of the data window is chosen based on the SNR in Eq. (**??**). Given the predefined SNR level SNR=35 dB, it is found that 600 samples are required to extract the slow-varying trend. Therefore, the window width is set $N = 600$. The model order is chosen based on the SSE shown in Fig. 4. The order $n = 13$ is selected to make a trade-off between match accuracy and computational burden.

The deck trends are given in Fig. 5 and Fig. 6 for two groups of real deck data (red dotted). Since measurement noise is always present, an EKF is designed to smooth out the deck motion measurements. This enables the proposed method to deal with measurements with large noise level, thus improving the robustness. It is seen that the estimated deck motion (green solid) is smoother than the noisy measurements (black solid) and makes it easier for the PA to handle. The estimated deck motion using the proposed PA is shown on the same graphs (blue dotted), it is seen that data produced by the Prony model match the measurements well. The standard deviations are 0.82 cm and 0.91 cm for Fig. 5 and Fig. 6. Based on the proposed PA, the deck

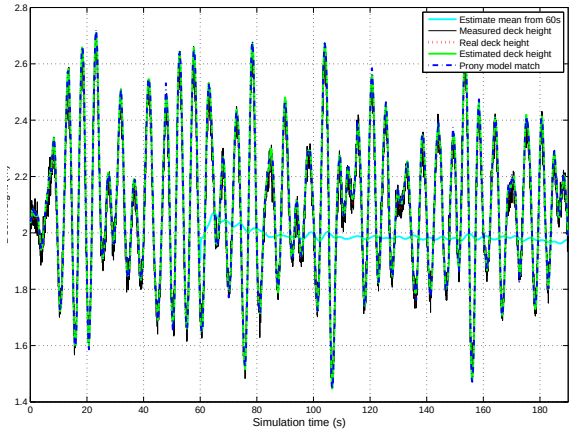

Fig. 5 Extracting monotonous trend of real deck displacement (group 1)

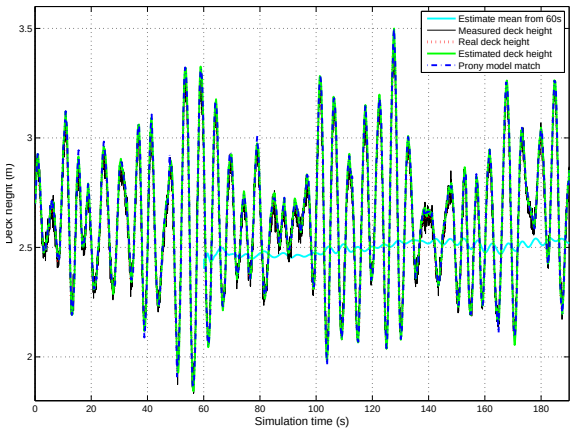

Fig. 6 Extracting monotonous trend of real deck displacement (group 2)

trend can be estimated about 1 minute, and sufficient time is given for trajectory planning and controller design.

5 Conclusion

In this paper we focus on building a proper procedure for extracting the trend of an oscillating deck. A practical EKF is designed to estimate the unknown ship displacement motion. A modified PA was proposed with model order identified by minimizing squared estimation errors and model coefficients determined using the FFRLS. Also, the dominant modes are found out based on a box threshold selection criterion. Simulation results justify the suitability of our procedure for extracting the deck trend.

6 Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS2012 proceedings or as individual off-prints from the proceedings.

References

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, AC-19(6):716–723, 1974.

[2] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Englewood Cliffs, N.J : Prentice-Hall, 1979.

[3] F. A. Faruqi and K. J. Turner. Extended kalman filter synthesis for intergrated global positioning/inertial navigation systems. *Applied Mathematics and Computation*, 115(2-3):213–227, 2000.

[4] M. A. Garratt. *Biologically Inspired Vision and Control for an Autonomous Flying Vehicle*. PhD thesis, Australian National University, Australia, Oct. 2007.

[5] S. GreenLand and M. P. Longnecker. Methods for trend estimation from summarized dose-response data with applications to meta-analysis. *American Journal of Epidemiology*, 135(11):1301–1309, 1992.

[6] J. F. Hauer. Application of Prony analysis to the determination of modal content and equivalent models for measured system response. *IEEE Transactions on Power Systems*, 6(3):1062–1068, 1991.

[7] M. A. Johnson, I. P. Zarafonitis, and M. Calligaris. Prony analysis and power system stability-some recent theoretical and applications research. In *IEEE Power Engineering Society Summer Meeting*, volume 3, pages 1918–1923, 2000.

[8] S. J. Julier. The scaled unscented transformation. In *Proceedings of the American Control Conference*, number 6, pages 4555–4559, 2002.

[9] P. B. Kenny and J. Durbin. Local trend estimation and seasonal adjustment of economic and social time series. *Journal of the Royal Statistical Society. Series A*, 145(1):1–41, 1982.

[10] R. V. D. Merwe and E. A. Wan. Sigma-point kalman filters for integrated navigation. In *Proceedings of the 60th Annual Meeting of the Institute of Navigation*, pages 641–654, Dayton, OH, USA, 2004.

[11] R. R. Murphy, K. S. Pratt, and J. L. Burke. Crew roles and operational protocols for rotary-wing micro-uavs in close urban environment. In *Proceedings of the 3rd ACM/IEEE International Conference on human robot interaction*, 2008.

[12] L. Qi, L. Qian, S. Woodruff, and D. Cartes. Prony analysis for power system transient harmonics. *EURASIP Journal of Advances in Signal Processing*, 2007:Article ID:48406, 2007.

[13] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

[14] T. Soderstrom and P. Stoica. *System Identification*. Prentice-Hall, 1989.

[15] D. H. Titterton and J. L. Weston. *Strapdown Inertial Navigation Technology*, volume 207. American Institute of Aeronautics and Astronautics, 2 edition, 2004.

[16] A. Vahidi, A Stefanopoulou, and H. Peng. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments. *Vehicle System Dynamics*, 43(1):31–55, 2005.

[17] J. Wendel, O. Meister, C. Schlaile, and G. F. Trommer. An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter. *Aerospace Science and Technology*, 10(6):527–533, 2006.

[18] L.A. Young. Small autonomous air/sea system concepts for coast guard missions. In *U.S. Coast Guard Maritime Domain Awareness Requirements, Capabilities, and Technology Forum, Santa Clara, CA*, Santa Clara, USA, 2005.

[19] P. C. Young. Time-variable parameter and trend estimation in non-stationary economic time series. *Journal of Forecasting*, 13(2):179–210, 1994.