Proceedings of the 2007 American Control Conference
Marriott Marquis Hotel at Times Square
New York City, USA, July 11-13, 2007

Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle

Carole G. Prévost[†], André Desbiens[†] and Eric Gagnon[‡]

Abstract— **The development of effective target tracking and collision avoidance algorithms is essential to the success of Unmanned Aerial Vehicle (UAV) missions. In a dynamic environment, path planning for UAVs is often based on predicted obstacle and target motion. In this paper, an extended Kalman filter (EKF) is first used to estimate the states of a moving object detected by a UAV from its measured position in space. The optimal object trajectory is then predicted from the estimated object states and using the motion model defined for Kalman filtering. Finally, the quality of the predicted trajectory is evaluated by computing the variance of the prediction error. Simulation results are presented to demonstrate the effectiveness of the proposed approach.**

I. Introduction

The global objective driving the present research project is to guide a fleet of Unmanned Aerial Vehicles (UAVs) through a series of moving targets/waypoints while avoiding a series of static and moving obstacles. The environment in which the UAVs operate is unknown and must be characterized by the vehicles. For this purpose, each UAV is equipped with sensors measuring the position of objects in space. Only objects situated within sensor range are detected by the vehicles. UAVs within communication range also exchange information they have gathered about their environment (i.e. target and obstacle locations). Obstacle and target states are then estimated and their trajectories predicted using the prediction algorithm presented in this paper. Finally, optimal UAV trajectories are selected using a particular predictive control algorithm based on the *GlobPC* [1].

The selection of adequate UAV trajectories is dependent on the ability to correctly predict target and obstacle behaviors. In a dynamic environment, path planning for UAVs often depends on estimated object trajectories [2], [3], [4]. If the predicted object trajectories differ greatly from their true trajectories, the selected UAV trajectories will be suboptimal and collisions may arise between obstacles and vehicles. In order to accurately predict the motion of an object, it is essential to correctly estimate its attitude or state at the present moment. If the dynamics of the object are uncertain and if sensors collecting data about the object are noisy and/or unreliable, estimating the true object states can become a difficult task.

Kalman filters are often employed to estimate object states. Some have used these filters to estimate the position of robots when their sensor measurements are inaccurate [5], [6], [7]. Others have employed Kalman filters to estimate the states of moving obstacles and targets subject to measurement uncertainty and model uncertainty [8], [9]. In [8], a Kalman filter is used to estimate the future position and orientation of a moving obstacle from its current position and orientation measurements. Assuming that the detected object is without mass, the author uses translational and rotational motion equations to predict its states. In [9], a Kalman filter is employed in real-time visual target tracking. In short, the filter is used to deduce target motion parameters, such as its position and velocity, by exploiting characteristics of target images taken at successive time frames. The tracked target is again modeled as an object without mass whose trajectory is solely governed by motion equations.

In this paper, an extended Kalman filter (EKF) is proposed to estimate the states of a moving object detected by a UAV. The object shall be considered as a controlled system of known dynamics and the filter will attempt to estimate its true setpoints, model states, model outputs and position in space, in spite of model uncertainty and measurement uncertainty. These estimated states together with the EKF motion model will then be used to calculate the optimal trajectory of the particular object. Finally, the quality of the predicted trajectory shall be evaluated by calculating the variance of the prediction error.

This paper begins with a detailed description of the problem at hand. In section II, the problem is stated and the set of states to be estimated by the Kalman filter is defined. In section III, the state equation and measurement equation defining our EKF are detailed. Mention of the chosen filter implementation and of an algorithm used to render the filter more numerically robust concludes this section. Section IV details the prediction algorithm used to calculate the trajectory of the object over a future horizon. Two test cases are then presented in section V: the first demonstrating the filter's ability to estimate the desired object states, the second showing optimal object trajectories computed from the estimated states and their associated prediction errors. Finally, conclusions and future work are presented in section VI.

II. Problem Statement and Variable Definition

A. Problem Statement

A UAV detects a moving object and records its position in space. The object is identified and its general dynamics are retrieved from a database. Knowing that this model is an estimation of the true object dynamics (model error) and that the UAV sensors are noisy (measurement error), one must

[†] LOOP, Université Laval, Québec, Québec, G1K 7P4, Canada
[‡] DRDC-Valcartier, Québec, Québec, G3J 1X5, Canada

attempt to estimate the object's states in order to predict its trajectory.

An EKF will be used to estimate the states of the object detected by the UAV. For such Kalman filtering, a non-linear state-space representation of type:

$$\begin{aligned} X(k+1) &= F\left[X(k)\right] + \omega(k) \\ Y(k) &= H\left[X(k)\right] + \nu(k) \end{aligned} \tag{1}$$

describing the object motion is required, where X is the state vector and Y is the output vector. The vectors ω and ν are stochastic processes (Gaussian, zero-mean, white noise, independent random variables) and represent the uncertainty on the model and measurements respectively.

B. Variable Definition for Kalman Filtering

Suppose that the model in Fig. 1, retrieved from a database, depicts the dynamics of the object motion.

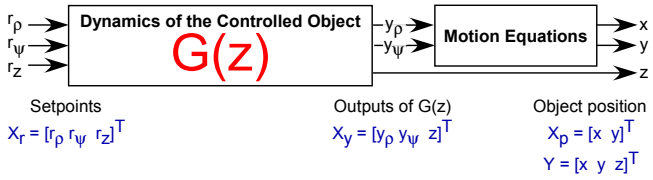


Fig. 1. Motion model of object under observation.

In Fig. 1, the transfer function $G(z)$ models the closed-loop response of the object under observation. If the detected object is an aircraft, $G(z)$ would model the dynamics of this aircraft with its autopilot. Hence, the detected object is considered a controlled and stabilized system where states X_y follow a series of setpoints X_r. Since the object has been identified by the UAV, the transfer function $G(z)$ is considered known and represents an approximation of the true underlying object dynamics.

Upon detection of the object, the UAV measures its position in space. The spatial coordinates of the object $(x,\ y,\ z)$ therefore form the measurement vector used for Kalman filtering (see Fig. 1):

$$Y = [x\ \ y\ \ z]^T. \tag{2}$$

The states to be estimated by the Kalman filter are the object setpoints (speed in the xy-plane r_ρ, heading in the xy-plane r_ψ, altitude r_z), outputs of model $G(z)$ (speed in the xy-plane y_ρ, heading in the xy-plane y_ψ, altitude z) and corresponding position (x-axis x, y-axis y, z-axis z). The vectors:

$$X_r = [r_\rho\ \ r_\psi\ \ r_z]^T \tag{3}$$
$$X_y = [y_\rho\ \ y_\psi\ \ z]^T \tag{4}$$
$$X_p = [x\ \ y]^T \tag{5}$$

shown in Fig. 1, thus describe the states of interest to be estimated by the EKF. In addition to X_r, X_y and X_p, the observer must also estimate the internal states X_m of $G(z)$ in Fig. 1.

Thus, the UAV must estimate all states in vector:

$$X = [X_r^T\ \ X_m^T\ \ X_y^T\ \ X_p^T]^T \tag{6}$$

in order to completely characterize the behavior of the detected object.

III. The Observer - Extended Kalman Filter

In this section, the state equation and measurement equation defining the EKF are presented.

A. The State Equation

The state equation describes the evolution of all states in the system. As shown in (6), the evolution of four distinct sets of states must be defined. These sets are the setpoints X_r, the internal states of model $G(z)$ X_m, the outputs of model $G(z)$ X_y and the position of the object in space X_p.

1) Setpoint Evolution: The object setpoints $(r_\rho,\ r_\psi,\ r_z)$ must be estimated. Since they are completely unknown, it is assumed that they behave like independent random walks [10]. Their temporal evolution can therefore be described as:

$$X_r(k+1) = X_r(k) + \omega_r(k) \tag{7}$$

where X_r is the state vector defined in (3) and where $\omega_r = [\omega_\rho\ \ \omega_\psi\ \ \omega_z]^T$ is a random vector composed of three independent, zero-mean, white noise, Gaussian random variables. The variances of the elements in ω_r are set to model variations in the setpoints between time samples. If the detected object is an aircraft, these variations can represent the pilot's ability or agressivity in varying the setpoints from one time instant to another. Setting $\omega_r = 0$ assumes that the object setpoints remain constant throughout time.

2) Object Dynamics: The dynamics of the object detected and identified by the UAV are represented by:

$$X_y(k) = G(z)X_r(k) \tag{8}$$

where,

$$G(z) = \begin{bmatrix} G_{11}(z) & G_{12}(z) & G_{13}(z) \\ G_{21}(z) & G_{22}(z) & G_{23}(z) \\ G_{31}(z) & G_{32}(z) & G_{33}(z) \end{bmatrix}. \tag{9}$$

The dynamics in the direct branches are represented by second-order transfer functions of the form:

$$G_{ii}(z) = Z\left\{\frac{1}{(1+T_{ii}s)^2}\right\} \tag{10}$$

where $i \in \{1,\ 2,\ 3\}$ and where $Z\{\cdot\}$ denotes the z-transform of the continuous-time transfer function. The dynamics in the indirect (coupling) branches are modeled as second-order discrete-time transfer functions of the form:

$$G_{ij}(z) = Z\left\{\frac{s}{(1+T_{ij}s)^2}\right\} \tag{11}$$

where $i,\ j \in \{1,\ 2,\ 3\}$ and $i \neq j$. The following discrete state-space representation of $G(z)$ is then computed:

$$X_m(k+1) = A_m X_m(k) + B_m X_r(k) + \omega_m(k) \tag{12}$$
$$X_y(k) = C_m X_m(k) + \nu_m(k) \tag{13}$$

where the state vector X_m is composed of all states modeling the dynamics in $G(z)$, where the input vector X_r is the state vector in (3) and where the output vector X_y is defined in (4). The random vectors ω_m and ν_m model the uncertainties on the states and outputs of model $G(z)$ respectively.

3) Motion equations: The temporal evolution of the xy-plane object position is governed by the following motion equations:

$$x(k+1) = x(k) + T_s y_\rho(k) \cos y_\psi(k) \tag{14}$$
$$y(k+1) = y(k) + T_s y_\rho(k) \sin y_\psi(k) \tag{15}$$

where x and y are the positions of the object along the x-axis and y-axis respectively. The remaining variables T_s, y_ρ and y_ψ are the sampling period and the object speed and heading in the xy-plane respectively. Note that the evolution of the object position along the z-axis has already been described by the evolution of the altitude output z.

Equations (14) and (15) are non-linear in nature:

$$X_p(k+1) = F_p\left[X_y(k), X_p(k)\right] \tag{16}$$

where X_y and X_p are defined in (4) and (5) respectively. A Taylor series approximation of the first order about the operating points X_y^* and X_p^* shall be applied to linearize these equations at each iteration of the EKF algorithm. As a result, one obtains:

$$X_p(k+1) = X_p(k) + B_p X_y(k) + \omega_p(k) \tag{17}$$

where,

$$B_p = \begin{bmatrix} T_s \cos y_\psi^*(k) & -T_s y_\rho^*(k) \sin y_\psi^*(k) & 0 \\ T_s \sin y_\psi^*(k) & T_s y_\rho^*(k) \cos y_\psi^*(k) & 0 \end{bmatrix} \tag{18}$$

and where the operating points y_ψ^* and y_ρ^* used to linearize the motion equations correspond to the states predicted/estimated by the observer at the previous sample time. The random vector $\omega_p = [\omega_x \ \omega_y]^T$ models errors due to approximating the non-linear motion equations by their linear counterparts.

4) Complete State Equation: Combining equations (7), (12), (13) and (16), the following non-linear state equation describing the object motion is obtained:

$$X(k+1) = F[X(k)] + \omega(k). \tag{19}$$

For Kalman filtering, this state equation must be linearized at each sample time. Thus, combining equations (7), (12), (13) and (17), yeilds the linear state equation governing the evolution of all states in the system:

$$X(k+1) = AX(k) + \omega(k) \tag{20}$$

where X is the complete state vector as described in (6), where,

$$A = \begin{bmatrix} I & 0 & 0 & 0 \\ B_m & A_m & 0 & 0 \\ C_m B_m & C_m A_m & 0 & 0 \\ 0 & 0 & B_p & I \end{bmatrix} \tag{21}$$

and where the random vector,

$$\omega = [\omega_r^T \ \ \omega_m^T \ \ \omega_y^T \ \ \omega_p^T]^T. \tag{22}$$

with covariance matrix W models the uncertainty on all system states.

B. The Measurement Equation

The UAV measures the position of the object in space. The corresponding measurement equation is:

$$\begin{bmatrix} X_p^m(k) \\ z^m(k) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_r(k) \\ X_m(k) \\ y_\rho(k) \\ y_\psi(k) \\ z(k) \\ X_p(k) \end{bmatrix} + \nu(k) \tag{23}$$

where X_p^m and z^m are the xy-plane and z-axis position measurements of the object. The random vector ν with covariance matrix V is added to model the uncertainty on all measurements (sensor noise). Equation (23) can be written in a more compact form as:

$$Y(k) = CX(k) + \nu(k). \tag{24}$$

C. Extended Kalman Filter Implementation

The EKF filter presented herein has been implemented such that it estimates the object states $\hat{X}(k+1/k)$, where the time index $k+1/k$ indicates the estimation at time $k+1$ from time k and where the symbol $\hat{}$ indicates the optimal state prediction. This particular filter implementation is referred to as a one-step predicted estimator [11]. The Bierman-Thornton UD filtering algorithm [12] has also been implemented as an alternative to solving the Riccati equation, in order to improve numerical robustness.

IV. Prediction of the Object Trajectory

Once the object's states have been estimated, its future trajectory can be predicted. In this section, the trajectory of the object over a future horizon is calculated. The variance of the prediction error associated with the computed trajectory is also calculated.

A. Object Trajectory

The object motion is governed by the following non-linear model:

$$\begin{aligned} X(k+1) &= F[X(k)] + \omega(k) \\ Y(k) &= CX(k) + \nu(k) \end{aligned} \tag{25}$$

where the state and measurement equations are defined in (19) and (24) respectively. Thus, for the object motion described in (25), the optimal prediction of the object trajectory over a future horizon N is:

$$\hat{\boldsymbol{Y}}_{\boldsymbol{pred}}^{\boldsymbol{nl}}(k+N/k) = C\boldsymbol{X}_{\boldsymbol{pred}}^{\boldsymbol{nl}}(k+N/k) \tag{26}$$

where,

$$\hat{\boldsymbol{Y}}_{\boldsymbol{pred}}^{\boldsymbol{nl}}(k+N/k) = \begin{bmatrix} \hat{Y}_{pred}^{nl}(k+1/k) \\ \hat{Y}_{pred}^{nl}(k+2/k) \\ \hat{Y}_{pred}^{nl}(k+3/k) \\ \vdots \\ \hat{Y}_{pred}^{nl}(k+N/k) \end{bmatrix} \tag{27}$$

and where,

$$\boldsymbol{X^{nl}_{pred}}(k+N/k) = \begin{bmatrix} X^{nl}_{pred}(k+1/k) \\ X^{nl}_{pred}(k+2/k) \\ X^{nl}_{pred}(k+3/k) \\ \vdots \\ X^{nl}_{pred}(k+N/k) \end{bmatrix} = \begin{bmatrix} \hat{X}(k+1/k) \\ F[\hat{X}(k+1/k)] \\ F[F[\hat{X}(k+1/k)]] \\ \vdots \\ F[\cdots F[\hat{X}(k+1/k)]\cdots] \end{bmatrix}. \quad (28)$$

Note that all predictions are a function of $\hat{X}(k+1/k)$, the states predicted by the EKF.

B. Variance of the Prediction Error

The quality of the predicted trajectory is evaluated by computing the variance of the prediction error (variance of the error between the modeled and predicted object trajectories). The variance of the prediction error cannot easily be computed from the non-linear motion model (25). It is thus calculated from an approximation of the object motion, obtained by linearizing (25) at each prediction step. The resulting object motion is described by equations (20) and (24) of the EKF model:

$$\begin{aligned} X(k+1) &= AX(k) + \omega(k) \\ Y(k) &= CX(k) + \nu(k). \end{aligned} \quad (29)$$

For the object motion described in (29), the optimal object trajectory over a future horizon N is:

$$\boldsymbol{\hat{Y}^{l}_{pred}}(k+N/k) = \boldsymbol{G}\hat{X}(k+1/k) \quad (30)$$

where,

$$\boldsymbol{\hat{Y}^{l}_{pred}}(k+N/k) = \begin{bmatrix} \hat{Y}^{l}_{pred}(k+1/k) \\ \hat{Y}^{l}_{pred}(k+2/k) \\ \hat{Y}^{l}_{pred}(k+3/k) \\ \vdots \\ \hat{Y}^{l}_{pred}(k+N/k) \end{bmatrix} \quad (31)$$

where $\hat{X}(k+1/k)$ are the states predicted by the EKF and where the prediction matrix $\boldsymbol{G}$ is:

$$\boldsymbol{G} = \begin{bmatrix} C \\ CA_{k+1} \\ CA_{k+2}A_{k+1} \\ \vdots \\ CA_{k+N-1}A_{k+N-2}\cdots A_{k+1} \end{bmatrix}. \quad (32)$$

Elements A_{k+j} where $j = \{1, 2, \ldots, N-1\}$ represent the evolution of matrix A in (29) over the prediction horizon N. More precisely, A_{j+1} is A evalutated at the operating point $X^{nl}_{pred}(k+j/k)$.

The error on the predictions in (30) is obtained by calculating:

$$\epsilon(k+h/k) = Y(k+h) - \hat{Y}^{l}_{pred}(k+h/k) \quad (33)$$

for $h \in \{1, 2, \ldots, N\}$. The quality of the predicted trajectory is then evaluated by calculating the variance of (33). One obtains:

$$\boldsymbol{\sigma^2_{\epsilon}}(k+N/k) = \boldsymbol{S} \quad (34)$$

where,

$$\boldsymbol{\sigma^2_{\epsilon}}(k+N/k) = \begin{bmatrix} \sigma^2_{\epsilon}(k+1/k) \\ \sigma^2_{\epsilon}(k+2/k) \\ \sigma^2_{\epsilon}(k+3/k) \\ \vdots \\ \sigma^2_{\epsilon}(k+N/k) \end{bmatrix} \quad (35)$$

and where,

$$\boldsymbol{S} = \begin{bmatrix} CP_{k+1/k}C^T + V \\ CM_{1,1}P_{k+1/k}M^T_{1,1}C^T + CWC^T + V \\ CM_{2,2}P_{k+1/k}M^T_{2,2}C^T + \\ CM_{1,2}WM^T_{1,2}C^T + CWC^T + V \\ \vdots \\ CM_{N-1,N-1}P_{k+1/k}M^T_{N-1,N-1}C^T + \\ CM_{N-2,N-1}WM^T_{N-2,N-1}C^T + \cdots + \\ CWC^T + V \end{bmatrix} \quad (36)$$

$$\begin{aligned} M_{1,j} &= A_{k+j} \\ M_{2,j} &= A_{k+j}A_{k+j-1} \\ M_{3,j} &= A_{k+j}A_{k+j-1}A_{k+j-2} \\ &\vdots \\ M_{N-1,j} &= A_{k+j}A_{k+j-1}\cdots A_{k+j-N+2} \end{aligned}$$

for $j \in \{1, 2, \ldots, N-1\}$. Matrices W and V are the model and measurement covariance matrices used in the EKF and $P_{k+1/k}$ is the covariance matrix of estimation uncertainty for states $\hat{X}(k+1/k)$ estimated by the EKF.

V. Simulations and Results

In this section, two test cases are shown. The first demonstrates the performance of the EKF of section III. The second demonstrates the performance of the trajectory prediction algorithm detailed in section IV. In the first test case, the EKF is used to estimate the states of an object in spite of process and measurement noises. In the second test case, trajectories of the object presented in the first test case are predicted from the states estimated by the EKF.

A. Test Case 1: Estimation of Object States

Suppose that a UAV detects a moving object whose true dynamics are:

$$X^{\star}_{y}(k) = G^{\star}(z)X^{\star}_{r}(k) \quad (37)$$

where,

$$G^{\star}(z) = \begin{bmatrix} G^{\star}_{11}(z) & 0 & G^{\star}_{13}(z) \\ 0 & G^{\star}_{22}(z) & 0 \\ G^{\star}_{31}(z) & 0 & G^{\star}_{33}(z) \end{bmatrix} \quad (38)$$

and where,

$$G^{\star}_{ii}(z) = Z\left\{\frac{1}{(1+5s)^2}\right\}$$
$$G^{\star}_{ij}(z) = Z\left\{\frac{5s}{(1+8s)^2}\right\} \quad (39)$$

for $i \in \{1, 2, 3\}$, $j \in \{1, 3\}$ and $i \neq j$. Suppose also that the UAV incorrectly identifies the object under observation and that the dynamics of the model are:

$$X_y(k) = G(z)X_r(k) \quad (40)$$

where,

$$G(z) = \begin{bmatrix} G_{11}(z) & 0 & G_{13}(z) \\ 0 & G_{22}(z) & 0 \\ G_{31}(z) & 0 & G_{33}(z) \end{bmatrix} \quad (41)$$

and where,

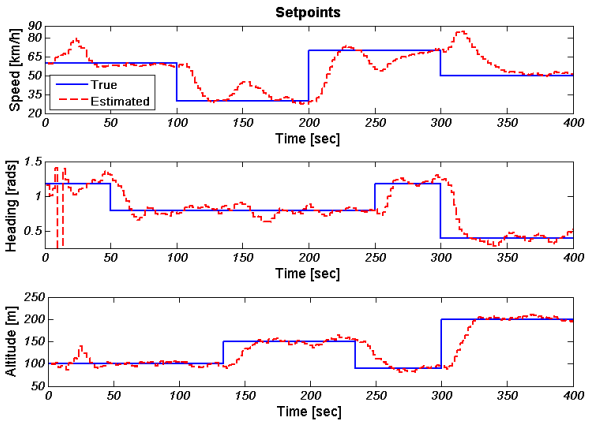
$$G_{ii}(z) = Z\left\{\frac{1}{(1+10s)^2}\right\}$$
$$G_{ij}(z) = Z\left\{\frac{5s}{(1+6s)^2}\right\} \quad (42)$$

for $i \in \{1, 2, 3\}$, $j \in \{1, 3\}$ and $i \neq j$. The sampling period for both process and model is $T_s = 2$ sec. Finally, suppose that UAV sensors measuring the position of the object in space are noisy.

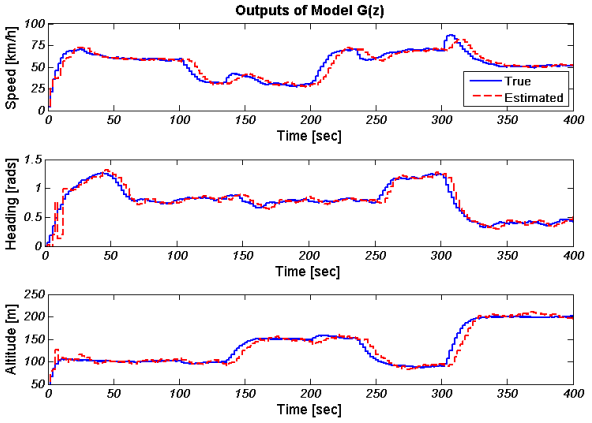
The Kalman filter of section III is used to estimate the true object setpoints, internal states of model $G(z)$, outputs of model $G(z)$ and object position. In the filter's implementation, setpoints are assumed to vary like random walks ($\omega_r \neq 0$) and the object dynamics identified by the UAV ($G(z)$) are considered uncertain ($\omega_m \neq 0$ and $\nu_m = 0$). The observer also considers error on the linear motion equations ($\omega_p \neq 0$) and considers noisy and/or unreliable UAV sensors ($\nu \neq 0$).

Test results are illustrated in Fig. 2. In Fig. 2(a), the true object speed, heading and altitude setpoints are shown in solid lines while the setpoints estimated with the EKF are represented by dashed lines. Note that the true object setpoints are a series of step functions and that the dynamics of these setpoints have been modeled as random walks. Fig. 2(b) depicts the true (solid) and estimated (dash) model outputs. Finally, Fig. 2(c) illustrates the true (solid), measured (dot-dash) and estimated (dash) object position.

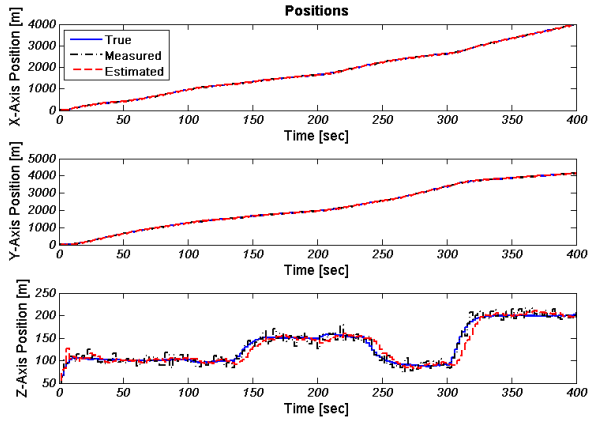
In Fig. 2(c), sensor noises can be seen affecting the altitude measurement (dot-dash line). One can also see how the observer manages to filter these noises and estimate an adequate object position. In spite of measurement uncertainty and model uncertainty, the EKF also succeeds in estimating accurate model outputs (Fig. 2(b)) and adequate object setpoints (Fig. 2(a)).



(a) True and estimated setpoints.



(b) True and estimated model outputs.



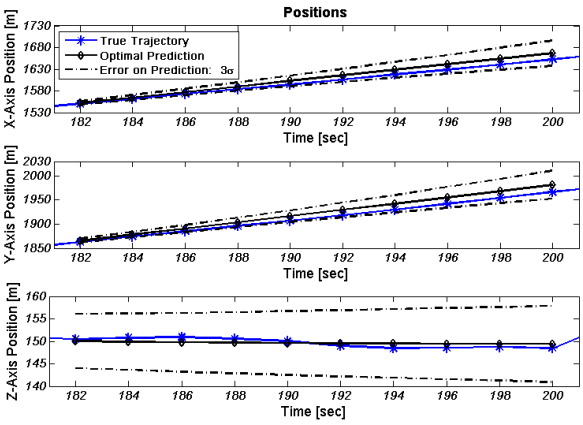
(c) True, measured and estimated Positions.

Fig. 2. Simulation: estimation of an object's states.

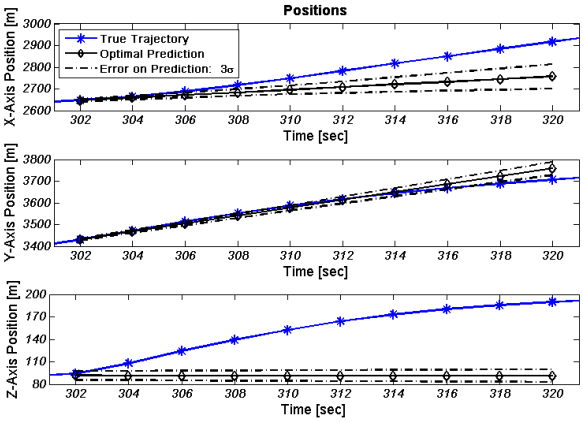
B. Test Case 2: Prediction of Object Trajectories

In this section, trajectories of the object in V-A are predicted from the states estimated by the EKF. The model employed to predict the object trajectory at each sample time is (25), adapted to the object dynamics stated in (40)-(42). The optimal object trajectory at each sample time is thus computed from (26). Each predicted trajectory is later compared with the true object trajectory.

Fig. 3 depicts two predicted object trajectories at two particular sample times. For each case, the predicted object trajectory (solid line, diamond markers), variance of the prediction errors (dot-dash lines) and true object trajectory (solid line, asterisk markers) along the x-axis, y-axis and z-



(a) True and estimated object dynamics are comparable.



(b) True and estimated object dynamics differ.

Fig. 3. Simulation: True and predicted object trajectories.

axis are drawn.

In Fig. 3(a), the predicted trajectory calculated at time $t = 182\,s$ is a good approximation of the true object trajectory. The latter closely follows the predicted optimal trajectory and does not deviate outside the error margins. Note that from $t = 134\,s$, all object setpoints were constant. The hypothesis that they behave like random walks is thus acceptable and the results are good.

Fig. 3(b) depicts the predicted trajectory from time $t = 302\,s$. Note that all setpoints were changed at time $t = 300\,s$ by a considerable amount, thus violating the assumptions about the future inputs. More precisely, the covariance of the random walk was selected too small and does not well represent the true magnitude of the possible changes in the setpoints. As a result, the prediction is degraded and is no longer a good approximation of the true object trajectory.

VI. Concluding Remarks

In this paper, an EKF was first designed to estimate the states of a moving object detected by a single UAV. Based on assumptions governing the object motion, adequate estimations of its setpoints, model outputs and positions were made, in spite of model and measurement uncertainties. The optimal object trajectory was then predicted from the states estimated by the EKF and using the motion model of the EKF. It was shown that accurate object trajectories can be predicted when the EKF model well represents the dynamics of the true object.

Future work will involve modifying the existing EKF structure to include the estimation of multiple objects' states detected by multiple UAVs. Fault detection will also be added to the existing observer to eliminate sensor biases and disregard measurements generated by defective or damaged sensors. Finally, the resulting observer and target/obstacle trajectory prediction algorithm will be paired with a predictive control algorithm based on the *GlobPC* to generate optimal UAV trajectories: trajectories that will guide the UAV through a series of targets/waypoints while avoiding a series of static and moving obstacles.

VII. Acknowledgment

Thanks are accorded to the Natural Sciences and Engineering Research Council of Canada (NSERC), to Numérica Technologies Inc. and to La Fondation Baxter & Alma Ricard. Their financial support has made this work possible. Thanks are also accorded to Defence Research and Development Canada (DRDC) for their technical support.

References

[1] A. Desbiens, D. Hodouin and É. Plamondon, "Global predictive control: A unified control structure for decoupling setpoint tracking, feedforward compensation and disturbance rejection", *IEE Proc. on Control Theory and Applications*, vol. 147, 2000, pp 465-475.

[2] H. Yu, C. Chi and R. Thompson, "Pursuing a dynamic target through motion prediction and follow boundary repair", *Proc. of the IASTED Inter. Conf. of Robotics and Applications*, 2000, pp 58-62.

[3] A. Elnagar, "Motion prediction of moving objects", *Proc. of the Second IASTED Inter. Conf. of Control and Applications*, 1999, pp 448-451.

[4] K. Kyriakopoulos and G. Saridis, "An integrated collision prediction and avoidance scheme for mobile robots in non-stationary environments", *Proc. of the IEEE Inter. Conf. on Robotics and Automation*, vol. 1, 1999, pp 219-229.

[5] L. Jetto, S. Longhi and G. Venturini, "Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots", *IEEE Trans. on Robotics and Automation*, vol. 2, 1999, pp 219-229.

[6] G. Garcia and P. Bonnifait, "A multisensor localization algorithm for mobile robots and its real-time experimental validation", *Proc. of the IEEE Intern. Conf. on Robotics and Automation*, vol. 2, 1996, pp 1395-1400.

[7] A. Curran and K. Kyriakopoulos, "Sensor-based self-localization for wheeled mobile robots", *in IEEE Inter. Conf. on Robotics and Automation*, Atlanta, GA, vol. 1, 1993, pp 8-13.

[8] A. Elnagar, "Prediction of moving objects in dynamic environments using Kalman filters", *Proc. of the IEEE Inter. Symp. on Computational Intelligence in Robotics and Automation*, 2001, pp 414-419.

[9] J. Lee, M. Kim and I. Kweon, "A Kalman filter based visual tracking algorithm for an object moving in 3D", *Proc. of the IEEE/RSJ Inter. Conf. on Intelligent Robots and Systems*, vol. 1, 1995, pp 342-347.

[10] Y. Bar-Shalom, X. Rong Li and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*, John Wiley & Sons, Inc., New York, NY; 2001.

[11] T. Kailath, A. Sayed and B. Hassibi, *Linear Estimation*, Prentice Hall, Upper Saddle River, NJ; 2000.

[12] G. Bierman, *Factorization Methods for Discrete Sequential Estimation*, Academic Press, New York, NY; 1977.