

MODELLING OF TRACKING SYSTEM USING SIX STATE KALMAN FILTER

Tomáš Vaispacher - Róbert Bréda - Pavol Lipovský

In this article is mentioned process, mathematical apparatus and graphical results of the state variable prediction simulation using Kalman filter, which is programmed in C++ platform. The content of this article is the tracker modelling using six state Kalman filter, which is based on the mentioned parts of the system model. The processes and the results demonstrate application of the algorithm in C++ platform, processing of the generated values of the state estimations and abilities of the deficiencies limitation on this modelling level. The algorithm change makes this process applicable for real dynamic systems in the terms of using real sensors.

K e y w o r d s: prediction, Kalman filter, state vector

1 INTRODUCTION

Mathematical and statistical processing of the signals and measurement results belongs to operations determining quality, information value and applicability of the obtained information. In the terms of dynamic processes is necessary to choose procedures, which are able to process results and generate values with required precision in the real time application. These results are then useable in the terms of control, statistical processing or the state prediction. The last field of application widely uses Kalman filtering for estimation of the state variables in discrete or continuous time, which differs depending on the mathematical description of the dynamic system and particular parts of the used mathematical apparatus. The characteristics, which predetermine this mathematical approach for most of the real time applications are wide ability of the mathematical application, implementation to programmes, ability of the complex statistical description of solved issue and ability of assembling to more complex systems through special (cascade) or integration architecture. The content of this article is estimation of the chosen state values using known equations, which creates the substance of Kalman filtering. For the simulation is used generating of the random numbers in the C++ platform in consideration of simulation level. The simulation consists of the predefined matrices and necessity of the noise simulation to influence values, which determine the system state.

2 DEFINITION OF PREDICTION ALGORITHM

The content of prediction are range of the vehicle and bearing of the vehicle in discrete time. These are defined on the basis of matrices, which define the system state description of the tracker. The concrete matrices for this problem are shown in [1.].

The basic mathematical apparatus of tracker, which will be used for software solution is set of the equations in discrete time space. These equations will be used by prediction algorithm. Definition of the discrete dynamic model equation (1.) and equation of the measurement model (2.) is as follows:

$$x_k = \Phi_{k-1}x_{k-1} + w_{k-1} \quad (1.)$$

$$y_k = H_k x_k + v_k \quad (2.)$$

In the equation (1.) Φ represents dynamic coefficient matrix, H in equation (2.) represents measurement sensitivity matrix, x defines the state vector and y is the measurement vector. In the equations are also included noises. In the equation (1.) w represents the process noise and in the equation (2.) represents v the measurement noise. Index k is an expression of the discrete time index.

The prediction based on recursive approach of Kalman filtering uses next set of the equations. The function of these equations can be written as cyclical prediction and correction of the state variables. The definitions of these equations are as follows:

$$\hat{x}_{k,k-1} = \Phi_{k-1} \hat{x}_{k-1} \quad (3.)$$

$$P_{k,k-1} = \Phi_{k-1} P_{k-1} \Phi_{k-1}^T + Q_{k-1} \quad (4.)$$

$$K_k = P_{k,k-1} H_k^T (H_k P_{k,k-1} H_k^T + R_k)^{-1} \quad (5.)$$

$$\hat{x}_k = \hat{x}_{k,k-1} + K_k (y_k - H_k \hat{x}_{k,k-1}) \quad (6.)$$

$$P_k = (I - K_k H_k) P_{k,k-1} \quad (7.)$$

In the prediction equations (3.) and (4.) is measured prediction of the state vector and prediction of the covariance matrix P . In the equation (4.) represent Q matrix of the process noise covariance. Equations of the correction are summarized in (5.), (6.) and (7.). Kalman gain equation (5.), which is weighting measurements in the equation (6.), contains except another defined parameters also matrix R . This matrix is defined as matrix of the measurement noise covariance. The equation (6.) is correction of the state vector in defined iteration and the equation (7.) is mathematical expression of the covariance matrix correction.

Accurate description of the principle is defined in [1] and [2]. The differences in comparison with general equation form are changes in equations (1.) and (7.), which are changed for this level of the simulation.

3 SIMULATION PARAMETERS

The state variables – range of the vehicle and bearing of the vehicle – will be predicted based on matrices, which describes chosen system. The models are introduced in [1]. The dynamic system matrix has following form:

$$\Phi = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \rho & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & r \end{bmatrix} \quad (8.)$$

The column vector of the process noise, which is part of the state vector calculation, has form:

$$w = \begin{bmatrix} 0 \\ 0 \\ w_{k-1}^1 \\ 0 \\ 0 \\ w_{k-1}^2 \end{bmatrix} \quad (9.)$$

In given matrices (8.) and (9.), which are in the equation (1.), represent variable T sampling period, ρ correlation coefficient, w^1 and w^2 are parts of w defined as the noise with zero mean value and known covariance.

The measurement sensitivity matrix H has considering on the required values of state variables the following form. By changing of this form is possible to alternate the required outputs and their number.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (10.)$$

For the solution of (2.) is required to define the measurement noise column vector as follows:

$$v = \begin{bmatrix} v_k^1 \\ v_k^2 \end{bmatrix} \quad (11.)$$

The state vector has the following form:

$$x^T = [r \quad \dot{r} \quad U_k^1 \quad \theta \quad \dot{\theta} \quad U_k^2] \quad (12.)$$

The equation (12.) represents state vector, which consists of the six basic state variables. Using the specific form of the matrix H we obtain in the each iteration of algorithm the required value. The variable r represents range of the vehicle in discrete time and following variable is derivation of the range in the same time. The variable U_k^1 defines maneuvering correlated state noise, variable θ represents bearing in discrete time and following variable is the derivation of bearing. The last state variable is U_k^2 – maneuvering correlated state noise.

After definition of all state variables and matrices, is required for complex prediction and correction to define all covariance matrices, which are

included in the algorithm. The covariance matrix P can be defined in two ways. In the first easier case we expect the form of the covariance matrix P to be equal to matrix of the process noise covariance Q . In the second case, which is customary solution, is calculation of this matrix by using the Riccati equation. This approach in necessary number of the iterations calculates the steady values of the each matrix elements. This approach optimizes the initial behavior and the filter response. The process noise covariance matrix has in the main diagonal variances of the single input noises, another elements represents the bond between another elements of the noises. On the same principle is also based the measurement noise covariance matrix. The matrices are recalculated in time and in this simulation will be used simplification in the form of constant matrices R and Q .

The forms in [1] are follows:

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_1^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_2^2 \end{bmatrix} \quad (13.)$$

$$R = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \quad (14.)$$

The values of the variances are based on description of the input noises. These are defined for this type of the prediction as noises with zero mean value and specific covariance. The last matrix for the recursive prediction algorithm is the covariance matrix P , which is solved by using the Riccati equation. The form of the covariance matrix is [1]:

$$P = \begin{bmatrix} \sigma_r^2 & \frac{\sigma_r^2}{T} & 0 & 0 & 0 & 0 \\ \frac{\sigma_r^2}{T} & \frac{2\sigma_r^2}{T^2} + \sigma_1^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_1^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_\theta^2 & \frac{\sigma_\theta^2}{T} & 0 \\ 0 & 0 & 0 & \frac{\sigma_\theta^2}{T} & \frac{2\sigma_\theta^2}{T^2} + \sigma_2^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_2^2 \end{bmatrix} \quad (15.)$$

The variances in main diagonals of (13.) and (14.) are defined in [1] as follows:

$$\sigma_r^2 = (1000)^2 [m]$$

$$\sigma_\theta^2 = (0,017)^2 [rad]$$

$$\sigma_1^2 = (103/3)^2$$

$$\sigma_2^2 = 1.3 \times 10^{-8}$$

All of these matrices are part of the recursive algorithm and the result is prediction of the state vector (or chosen elements).

4 PROGRAMMING AND OUTPUTS

For programming is used C++ platform, where are defined all of the presented matrices and constants. The equations which define the process of prediction and correction are calculated by using the values, which are the result of equations (1.) and (2.). In these simulations aren't used the real measurements, but using equations (1.) and (2.) are in each iteration generated values of the state vector elements.

In the defined number of iterations (in this case 200) run as initial step of the iteration generating of the state vector values. These values are then used in equations for the prediction and the correction. The results are estimations of the chosen state variables. For correct operation of the program is necessary to define functions for the matrix calculations and for calculation of the inverse matrix. According to statistical properties of the process and measurement noise are presented values of the noise vectors simulated in each step using random numbers. The simulation inputs are also the sampling period $T = 1.2$ s and constant value of correlation coefficient [1] $\rho = 0.5$. The initial values of the state vector are equal to zero. This is the result of the transition phenomena in beginning estimations. For processing is defined the output to the text file and saved results are then processed in the graphical software.

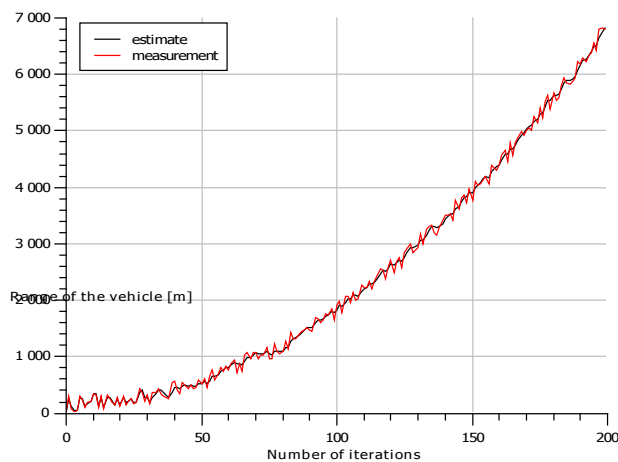


Fig. 1 Measured value and estimation of the tracker range

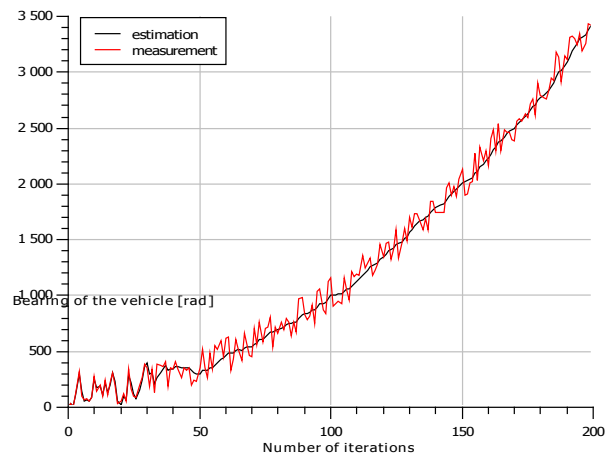


Fig. 2 Measured value and estimation of the tracker bearing

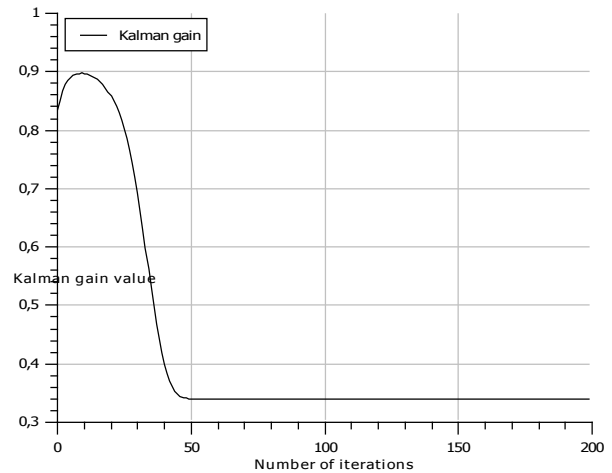


Fig. 3 Evolution of Kalman gain for range

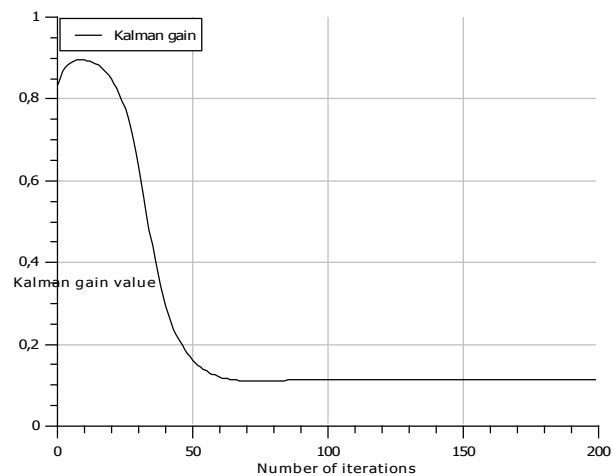


Fig. 4 Evolution of Kalman gain for bearing

If we multiply the state vector and matrix Φ and then the result of this operation in general form without including of the process noise multiply with matrix H , the result is:

$$\Phi x = \begin{bmatrix} r + T\dot{r} \\ \dot{r} + U_1 \\ \rho + U_1 \\ \theta + T\dot{\theta} \\ \dot{\theta} + U_2 \\ U_2 + r \end{bmatrix} \quad (16.)$$

$$H * \Phi x = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} r + T\dot{r} \\ \dot{r} + U_1 \\ \rho + U_1 \\ \theta + T\dot{\theta} \\ \dot{\theta} + U_2 \\ U_2 + r \end{bmatrix} = \begin{bmatrix} r + T\dot{r} \\ \theta + T\dot{\theta} \end{bmatrix} \quad (17.)$$

Solution of (17.) is expression of the output as predicted state variables. If we change values in matrix H , this change will lead to the new output definition or new number of output values.

5 CONCLUSION

In this article are mentioned results of the Kalman filtering simulation of the chosen state variables. In article is used particular model of the dynamic system, which is used for simulation of chosen variables. The simulation outputs illustrated in Fig. 1 and Fig. 2 shows the transition phenomena in the beginning simulation. This phenomena in range of 0 – 50 iterations is caused by used approach of modelling, which is included in C++ program. The algorithm uses constant values of the Q and R matrices elements. Although is used the Riccati equation for calculating of the covariance matrix, there aren't calculations for determination of the Kalman gain matrix elements, which should be used as initial for the algorithm. This is the main reason of the transition phenomena, which is also proven by Fig. 3 and Fig. 4. Removing of this phenomena and increasing of the processing accuracy can be achieved by performance analysis of the filter. In this case, equations (4.), (5.) and (7.) will be respectively recalculated before algorithm using initial value of matrix P , which is result of the Riccati equation. This process leads to initial values of the Kalman gain matrix, which will reduce the transition phenomena. Real application of this prediction algorithm is also based on the statistical sensor description, which is initial information for creating of Q and R covariance.

BIBLIOGRAPHY

- [1] GERWAL, M.S. – ANDREWS, A.P.: Kalman filtering: Theory and practise using MATLAB, ISBN: 0-471-39254-5
- [2] SOTÁK, M. – SOPATA, M. – BRÉDA, R. – ROHÁČ, J. – VÁCI, L.: Integrácia navigačných systémov – Monografia, ISBN: 80-969619-9-3
- [3] WELCH, G. – BISHOP, G.: An introduction to the Kalman filter, Course 8, SIGGRAPH 2001
- [4] KLEINBAUER, R.: Kalman filtering implementation with Matlab, Institute of Geodesy, Universität Stuttgart, Helsinki 2004
- [5] STORMS, W.F.: Magnetic field aided indoor navigation, Air Force Institute of technology, March 2009

AUTHORS' ADDRESSES

VAISPACHER Tomas, Bc.
Faculty of Aeronautics, TUKE
Rampova 7, 04121
Kosice, Slovakia
e-mail: tomas.vaispacher@gmail.com

BREDA Robert, Ing., PhD.
Faculty of Aeronautics, TUKE
Rampova 7, 04121
Kosice, Slovakia
e-mail: robert.breda@tuke.sk

LIPOVSKY Pavol, Ing., PhD.
Faculty of Aeronautics, TUKE
Rampova 7, 04121
Kosice, Slovakia
e-mail: pavol.lipovsky@tuke.sk