Universal prediction algorithm in maneuvering target tracking

Y Zemskov,1 E Koptseva,2 A Pankratova,1 S Borodulina,1 and S Kalinov1

1 Saint Petersburg State University of Civil Aviation, St. Petersburg, Russian Federation
2 Admiral Makarov State University of Maritime and Inland Shipping, St. Petersburg, Russian Federation

E-mail: george-zemskov@mail.ru, KoptsevaEP@gumrf.ru, piter00000@mail.ru, savvakalinov@gmail.com

Abstract. The paper considers the methods to increase the accuracy of trajectory processing of the results of radar measurements in automated air traffic management systems (AATMS). The recursive filters traditionally used in the secondary processing path of radar information of modern AATMS systems are briefly considered: alpha-beta filter, linear and extended Kalman filters, multi-model filter. Along with multi-model Kalman filters it is proposed to use a universal filter based on the universal prediction theorem, the accuracy of which is asymptotically no worse than any other filter in the class of continuous functions. The algorithm for the operation of a universal alpha-beta filter was developed. As a result of modeling, it was obtained that the mean square of the tracking error of the trajectory of a maneuvering target for the universal filter during the change of the motion model turned out to be lower than for traditional filters.

1. Introduction

The safety and regularity of aircraft (AC) flights, as well as the economic performance of air traffic, largely depend on the effective operation of air surveillance facilities. The increase in the density of traffic flows requires the increase in the accuracy of the information and measurement subsystems of automated air traffic management systems (AATMS), which should prevent dangerous approach of aircraft.

The estimation of coordinates and parameters of aircraft movement by the sequence of measurements (trajectory processing) refers to the subsystem of secondary processing of radar information [1]. In the first automated air traffic management systems (AATMS), an alpha-beta filter was used for these purposes, which performed a weighted summation of the forecast and the next result of measuring the coordinates of the aircraft, and the forecast was calculated based on the assumption of the constancy of the speed and direction of movement of each aircraft.

In modern AATMS, various modifications of the Kalman filter are used [1-3], assuming that disturbances and interference acting on the system are random variables with known distribution functions. In this case, the Kalman filter is tuned to a specific aircraft movement model (horizontal flight at a constant speed, coordinated turn, climb and descent along the glide path).
In most cases, due to the use of a polar or spherical coordinate system, the problem statement is non-linear and developers have to solve problems with the convergence of the corresponding extended filter [1, 4, 5].

In recent years, the theory of multi-model filters has been actively developing [6], in which the set of standard models of aircraft motion is supplemented with an algorithm for detecting maneuvers and correcting filter parameters or their structure. The most frequently cited interactive multi-model (IMM) algorithm calculates the posterior probabilities for each aircraft motion model, and the output estimate of the state vector of the observed system is the weighted sum of the estimates for each model.

The filter operates in two stages in all of the above mentioned cases: in the first stage, from various considerations, the forecast of the next state of the observed system (prior state estimate) is calculated, in the second stage, this estimate is corrected taking into account the next measurement result, i.e. the posterior estimate is calculated, which is the result of the filter. It is necessary to emphasize that in all cases the forecast is based on any assumptions about the behavior of the observed system (a certain model of aircraft movement is considered).

In this paper, we made an attempt to generalize the traditional approach to trajectory processing algorithms and proposed to use in the first stage (prediction the next state) the “universal” prediction methods developed over the past forty years, in which no prior assumptions were made about the behavior of the observed system.

The term “universal” goes back to the problem of universal coding of information generated by a source with an unknown probability distribution [7] and it will be used in this paper without quotes.

In the works [8-10], the theorems were proved that the predictions of the universal algorithm asymptotically up to a certain regret (in certain cases independent of the length of the observed sequence) turned out to be no worse than any other algorithm, including the one that was tuned to the “correct” (corresponding to the observed system) model of the source. Despite the obvious paradoxical feature of this result, numerous publications prove that universal predictions can be used in practical applications, although it should be noted that in the general case, the convergence of the algorithm may be slow and the computational complexity may be excessive for real-time processing, as a result of which the versatility in the name of the algorithm should be understood in the narrow sense outlined above (as a synonym for “modellessness”) and should not be transferred to the possibility of widespread use in any applied problems without additional analysis.

2. Materials and Methods

2.1. Alpha beta (or g-h) filter

In the case of two Cartesian coordinates, the work of the alpha-beta filter can be written in matrix form:

\[
\begin{bmatrix}
\hat{x}_k \\
\hat{\dot{x}}_k \\
\hat{y}_k \\
\hat{\dot{y}}_k
\end{bmatrix} = \begin{bmatrix}
(1 - \alpha) & (1 - \alpha)T & 0 & 0 \\
-\beta/T & (1 - \beta) & 0 & 0 \\
0 & 0 & (1 - \alpha) & (1 - \alpha)T \\
0 & 0 & -\beta/T & (1 - \beta)
\end{bmatrix} \cdot \begin{bmatrix}
\hat{x}_{k-1} \\
\hat{\dot{x}}_{k-1} \\
\hat{y}_{k-1} \\
\hat{\dot{y}}_{k-1}
\end{bmatrix} + \begin{bmatrix}
\alpha & 0 \\
\beta/T & 0 \\
0 & \alpha \\
0 & \beta/T
\end{bmatrix} \cdot \begin{bmatrix}
x_k \\
y_k
\end{bmatrix},
\]

where \(x_k\) and \(y_k\) — the results of measurements of coordinates at the \(k\)-th moment of time; \(\hat{x}_k\) and \(\hat{\dot{x}}_k\) — posterior predictions (calculated after \(k\)-th result of measurement) for the coordinate and velocity component, respectively; \(\alpha\) and \(\beta\) — coefficients that determine the weights of the forecast and the measurement result for the coordinate and speed, respectively. When \(\alpha = 1/2\) the result is the arithmetic mean of the forecast and measurement; when decreasing \(\alpha\) prediction weight increases. For civil aviation \(\alpha = 0.3 \ldots 0.5, \beta = 0.1 \ldots 0.3\) [11] is usually chosen.
2.2. Linear discrete Kalman filter

In the simplest case of uniform rectilinear motion in Cartesian coordinates, a linear model with an additive noise is used

\[ \mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k, \]  

where \( \mathbf{x}_k \) — n-dimensional state vector (usually represented by coordinates and their derivatives); \( \mathbf{F}_k \) — an \( n \times n \) transition matrix that connects the states of the system at the previous and subsequent times, i.e. \( \mathbf{x}_{k-1} \) and \( \mathbf{x}_k \); \( \mathbf{B}_k \) — size control matrix \( n \times l \); \( \mathbf{u}_k \) — l-dimensional vector of control actions acting at the input of the system; \( \mathbf{w}_k \) — some -dimensional random process that describes the random nature of the evolution of the system.

The measurements are taken with some error \( \mathbf{v}_k \):

\[ \mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k, \]

where \( \mathbf{H}_k \) — observation matrix \( m \times n \), which connects the true state vector \( \mathbf{x}_k \) and \( m \)-dimensional vector of measurements \( \mathbf{y}_k \) (usually consisting of the observed coordinates, and if Doppler radar is used, then the velocity components are also used).

The Kalman filter tries to estimate the state of the system at the current moment in time, based on its estimate made at the previous moment in time, as well as on the new measurement results.

Prior estimate of the coordinate vector (before the next measurement \( \mathbf{x}_k \)):

\[ \hat{\mathbf{x}}_k = \mathbf{F}_k \hat{\mathbf{x}}_{k-1} + \mathbf{B}_k \mathbf{u}_k; \]

error covariance matrix estimate:

\[ \mathbf{P}_k = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k. \]

After the next measurement \( \mathbf{x}_k \) the a posteriori estimate of the state vector is calculated:

\[ \hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \]

and the estimate of the error covariance matrix:

\[ \mathbf{P}_k^+ = (I - \mathbf{G}_k \mathbf{H}_k) \mathbf{P}_k^- \],

where the filter coefficients are calculated by the formula

\[ \mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}. \]

2.3. Extended Kalman filter

In the extended Kalman filter, the system model and/or observation model are considered non-linear:

\[ \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) + \mathbf{w}_k; \]

\[ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k. \]

To synthesize an extended first-order filter, these functions are approximated by keeping only the linear term of the Taylor series:

\[ \mathbf{F} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{X} = \hat{\mathbf{x}}}; \quad \mathbf{H} = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{\mathbf{X} = \hat{\mathbf{x}}}. \]

For example, for the “coordinated turn” trajectory, the model of system (2) is represented in the following form:
\[
\begin{bmatrix}
\dot{x}_k \\
\dot{\theta}_k \\
y_k \\
\omega_k
\end{bmatrix}
= \begin{bmatrix}
\sin(\omega T) & 0 & -(1-\cos(\omega T)) & 0 \\
\omega & 0 & \omega & 0 \\
0 & \cos(\omega T) & 0 & -\sin(\omega T) \\
0 & 1-\cos(\omega T) & \sin(\omega T) & 0 \\
0 & 0 & \cos(\omega T) & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_k \\
\theta_k \\
y_k \\
\omega_k
\end{bmatrix}
+ \begin{bmatrix}
T^2/2 \\
T \\
T^2/2 \\
T
\end{bmatrix} \cdot w_k,
\]

where \( \omega \) – angular velocity, rad/s; \( T \) – time step, s. Observation model (if only coordinates are measured):

\[
y_k = \begin{bmatrix}
1 \\
0 \\
0 \\
0
\end{bmatrix}
\begin{bmatrix}
x_k \\
\dot{x}_k \\
y_k \\
\dot{y}_k
\end{bmatrix}
+ v_k.
\]

2.4. Universal filter

Let us consider a sequence of real numbers \( x_1, x_2, \ldots, x_{k-1}, \) generated by some source. For simplicity, we assume that all numbers \( x_k \) normalized to the unit segment \([0; 1]\). The probabilistic model of the source of the sequence is unknown and no restrictions are imposed on it.

The prediction algorithm calculates the corresponding sequence of predictions \( \hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{k-1}, \hat{x}_k \). Prediction \( \hat{x}_k \) is done after receiving the measurement result on the \((k - 1)\)-th step, before \( x_k \). For definiteness, let us take \( \hat{x}_1 = 0 \) (prediction generated before the arrival of the very first \( x_1 \)). Following [9], we require that the forecasts satisfy the condition of calibratability, i.e. for any subinterval \( I = [a, b] \), \( (a, b) \), \( (a, b) \) of considered segment \([0, 1]\) with any choice of indicator function

\[
I(p) = \begin{cases}
1, & \text{if } p \in I; \\
0, & \text{if } p \notin I
\end{cases}
\]

(11)

calibration error tended to zero:

\[
\frac{\sum_{i=1}^{k} I(x_i)(x_i - \hat{x}_i)}{\sum_{i=1}^{k} I(x_i)} \to 0,
\]

(12)

if the denominator (12) tends to infinity at \( k \to \infty \). The indicator function (11) defines the rule according to which the values \( i \), for which the difference between the predictions \( \hat{x}_i \) and corresponding outcomes \( x_i \) is taken into account.

For the simplified case of predicting a binary sequence \( x_k \in \{0, 1\} \) the value \( \hat{x}_k \) may be considered as an estimate of the likelihood that an event will occur in the next step \( x_k = 1 \); then the condition of calibratability means that the algorithm estimates these probabilities adequately: roughly, for those values \( k \), when the estimate of the probability of completion was about 0.7, actually an event \( x_k = 1 \) occurred in about 70% of cases.

For the sequence of real numbers considered here, the prediction values \( \hat{x}_k \) are estimates of the mathematical expectation \( x_k \).

In the work [10], it was proved that the above mentioned condition of calibratability can not be achieved in the class of deterministic prediction algorithms when performing analysis for the worst case, in practice corresponding to the adaptively hostile behavior of the observed system (note that the prediction algorithm can be considered as an online algorithm and, therefore, can be subjected to competitive analysis, i.e. instead of the traditional worst-case analysis, the effectiveness of an online prediction algorithm can be compared with the optimal offline algorithm, but this issue can be the topic of a separate study).
The problem of calibratability is solved by the randomization of predictions: for any $\Delta > 0$ it is possible to propose such a randomized algorithm $f(x^k) = \text{Pr}_k(x)$, that for any infinite sequence $x^k = x_1, x_2, \ldots, x_k$ ($k \to \infty$) with probability 1 the inequality is presented:

$$\lim_{k \to \infty} \sup \frac{1}{k} \sum_{i=1}^{k} I(x_i) (x_i - \hat{x}_i) \leq \Delta,$$

Where predictions $\hat{x}_1, \hat{x}_2, \ldots$ are distributed according to the probability measure $\text{Pr}$.

Let us divide the unit interval $[0, 1]$ in $K$ segments of length $\Delta = 1/K$, in this case, the coordinates of the ends of each segment are equal

$$v_i = i\Delta, \quad (13)$$

where $i = 0, 1, ..., K$. Any number $x \in [0, 1]$ either matches one of $v_i$, or it can be represented as a linear combination of two boundary points of the segment to which $x$:

$$x = w_l(x_i) \cdot v_l + w_{l+1}(x) \cdot v_{l+1}, \quad (14)$$

where $i = \lfloor x/\Delta \rfloor = \lfloor xK \rfloor$ (semi-square brackets mean rounding down), and the weight function

$$w_l(x) = 1 - \frac{|v_l - x|}{\Delta}, \quad (15)$$

Assuming that $w_l(x) = 0$ for all other values $i$ (corresponding to all other segments of the partition), we can write (14) in vector form:

$$x = \sum_{l=0}^{K} w_l(x) v_l = w_{l-1}(x) v_{l-1} + w_l(x) v_l = \mathbf{w}(x) \cdot \mathbf{v}. \quad (16)$$

The deterministic prediction obtained at each step (according to the algorithm described below) we will randomize, randomly rounding it to the left boundary of the segment of the partition $v_{l-1}$ with probability $w_{l-1}(x)$ or to the right limit $v_{l}$ with probability $w_{l}(x)$. A prediction randomized according to this rule is denoted $w(x)$.

Next, we consider the value

$$\mu_{k-1}(\mathbf{v}) = \sum_{l=1}^{k-1} \mathbf{w} (\hat{x}_l)(x_l - \hat{x}_l), \quad (17)$$

where vector $\mathbf{w}$ depends on the method of dividing the segment $[0, 1]$, i.e. on $\mathbf{v}$; vector components $\mathbf{w}$ are calculated by the formula (15); components $\mathbf{v}$ — by the formula (13).

Following [8] and writing $\left(\mu_k(\mathbf{v}) \right)^2$ as the square of the sum of the previous value $\mu_{k-1}(\mathbf{v})$ and the last term according to the formula (17), we change the order of summation by time counts and division segments. Next, we find the mathematical expectation of a random variable $I(\hat{x}_l)(x_l - \hat{x}_l)$ by the formula

$$E[I(\hat{x}_l)(x_l - \hat{x}_l)] = \sum_{l} w_l(x_l) I(i)(x_l - \hat{x}_l),$$

and apply the Cauchy-Bunyakovsky inequality to the vectors $\mathbf{u}$ and $\mathbf{I}$. As a result, we get that if we choose the next deterministic prediction $\hat{x}_k$ as the root of the equation

$$\sum_{l=1}^{k-1} F(x_l, \hat{x}_l) \cdot (x_l - \hat{x}_l) = 0, \quad (18)$$

Where kernel:

$$F(x, \hat{x}) = \mathbf{w}(x) \cdot \mathbf{w}(\hat{x}), \quad (19)$$

(dot product of vectors), then for any $\Delta$ the upper estimate will be satisfied almost everywhere:

$$\lim_{k \to \infty} \sup \frac{1}{k} \sum_{l=1}^{k} I(\hat{x}_l)(x_l - \hat{x}_l) \leq \Delta \quad (20)$$

(where $\hat{x}_l$ — randomization result of deterministic prediction $\hat{x}_l$). In other words, the requirement of calibrating randomized predictions will be fulfilled.
In practice, to calculate the root of equation (18), it is easier to use a smooth approximation of the kernel (19) - the Gaussian kernel

$$F(x, x') = e^{-\alpha(x-x')^2}$$

(21)

For some $\alpha > 0$ or the kernel of the form

$$F(x, x') = \cos\left(\frac{\pi}{2}(x - x')\right)$$

(22)

when $x, x' \in [0, 1]$.

In the work [13, p. 195], a stronger result is given: the sequence of randomized predictions constructed in the described manner turns out to be asymptotically no worse than the result of any other algorithm presented as a continuous function of the input information.

Thus, the algorithm of the universal filter is as follows:

1. Suppose that $\tilde{x}_1 = 0, \tilde{x}_3 = 0$. Choose the value $K$ (note B).

2. We get the first sample of the measured coordinate $x_1$ (generalization to the case of multiple measurements is given below, note C).

3. Repeatedly calculating the left side of equation (18), using the kernel (19), (21) or (22), we find one of the roots (for example, by the method of half division). We take the found root as a deterministic prediction $\hat{x}_2$.

4. We find $\tilde{w}_2$, by randomizing the prediction according to (13), (14) and (15). We will take the resulting value as prior estimate of the coordinate.

5. We get the following measurement result $x_2$. Let us calculate the posterior estimate of the coordinate in the form of the weighted sum of the prior estimate $\tilde{x}_2$ and the measured coordinate $x_2$ by the formula $\hat{x}_2^+ = (1 - \alpha)\tilde{x}_2 + \alpha x_2$. Let us take the posterior estimate as the output of the universal filter.

6. We will repeat steps 3–5, calculating the next prior estimate $\tilde{x}_k$, obtaining the measurement result $x_k$ and calculating the posterior estimate $\hat{x}_k^+$. As the number $k$ increases, we will increase the number of segments of the partition $K$ according to the expression $K = C \cdot k^{-1/3}$, where $C$ is a constant.

Algorithm Notes:

A) The prediction randomization is important for the mathematical proof of estimate (20) in the worst case - adaptively hostile behavior of the observed system. Since, in the case of trajectory processing of air monitoring data, the result of the next measurement $x_k$ can not change after calculating the prior estimate $\tilde{x}_k$ (i.e., the observed system is indifferent to our forecasts), then item 4 of the algorithm can be omitted, i.e. the deterministic forecast $\hat{x}_k$ can be taken as prior estimate of the universal filter $\tilde{x}_k$.

B) If the previous amendment is accepted and when kernels of the form (21) or (22) are used, the parameter $K$ is not required and the stage of dividing it into segments can be omitted.

C) The generalization in case of several measurements. When observing the state vector (several coordinates and, possibly, their time derivatives), each component of the state vector is processed separately (as in the traditional alpha-beta filter).

D) With the increase in the number of obtained measurements and accumulated forecasts, the computational complexity for item 3 unjustifiably increases, while there is no need to take into account the past readings, which are far enough from the current moment in time. Therefore, when the report number $k$ approaches a certain threshold value $k_m$, it is necessary to initialize the operation of the new filter (starting from item 1) and after several steps, when the difference in the output signals of new and old filter becomes less than the specified value $\delta_m$, only the results of the new filter can be used.

E) If we take into account the physical limitations of the problem and the aircraft performance characteristics (the observed target can not change the direction of movement too sharply and its speed is limited), the search area for the root of the equation for item 3 will turn out to be rather narrow, which will significantly reduce the computational complexity of the algorithm. Note that this aspect does not negate
the fact that the universal filter does not use any assumptions about the nature of the target's motion, since the result of the algorithm's operation (coordinate estimate) will not change when the search area for the root of the equation is narrowed.

F) The choice of the kernel in the form (21) is essentially equivalent to the approximation of the left side of equation (18) using neural networks of a radial basis, which opens up wide opportunities for the solution of the problem on parallel computing systems.

3. Results and discussion
The authors simulated the operation of the alpha-beta filter (1), the linear Kalman filter (2–8) and the proposed universal filter in the case of a target moving along a “horizontal figure eight” trajectory (Fig. 1), similar to that used to test the trajectory algorithms processing in the EKF/UKF toolbox library of the MATLAB system [14]. The measurement errors were modeled as additive white Gaussian noise with zero average value. For illustrative purposes, the noise power was chosen somewhat overestimated (in comparison with the values typical for modern radar measurements).

To compare the efficiency of trajectory processing algorithms, the root-mean-square error (RMSE) was calculated, i.e. square root of the mean square of the difference between the true coordinates $x_k$ and the corresponding filter output $\hat{x}_k$:

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} |x_k - \hat{x}_k|^2},$$

where $N$ — the number of samples along the considered trajectory.

Figure 2 shows one of a hundred realizations of each random process: the dotted line shows the true (unknown to the filtering algorithms) trajectory of the target movement, the points correspond to the results of radar measurements, the output signal of the Kalman filter (2–8) is shown by the dash-dot line, the result of the universal filter is shown by the solid line. In order not to clutter up the figure, the output signal of the traditional alpha-beta filter (1) and the universal alpha-beta filter is not shown (it is practically indistinguishable from the results of prior predictions of the universal algorithm).

The numerical values of the root mean square errors along the entire trajectory (averaged results over 100 tests) are presented in the table.

![Figure 1. “Horizontal figure eight” trajectory](image-url)
Figure 2. Comparison of the results of the universal filter and the Kalman filter for the horizontal figure eight trajectory (only the x coordinate is shown)
Table 1. RMSE of tracking along “horizontal figure eight” trajectory for different filters

| Filter                      | RMSE, m |
|-----------------------------|---------|
| Raw measurements            | 393     |
| Universal filter            | 128     |
| Universal alpha-beta filter | 105     |
| Linear Kalman filter        | 197     |
| IMM                         | 146     |

4. Conclusion

Thus, the simulation results prove that in the case of a maneuvering target, at the moments of the movement model change, when the root-mean-square error of the traditional filter tracking is the largest, the universal algorithm shows a faster adaptation to the changed system parameters. While on quiet parts of the trajectory corresponding to a constant motion model, the RMSE of the universal algorithm, as expected, turns out to be no worse than traditional trajectory processing algorithms.

For further research, the authors set the task to embed a universal prediction algorithm into the general multi-model filter, as well as to develop an algorithm for the aggregation of predictions of a group of models based on the method [15].

At the same time, it is impossible to ignore the obvious disadvantages of the proposed method associated with the increased computational complexity of the algorithm and the absence of a recursive implementation typical for the operation of traditional filters.

Nevertheless, the authors hope that the proposed method can be useful in the solution of problems of trajectory processing of rapidly maneuvering targets, with a large prior uncertainty of the motion parameters, as well as in the composition of a multi-model filter.

References

[1] You H, Jianjuan X, Xin G 2016 Radar Data Processing with Applications (Wiley) DOI 10.1002/9781118956878.
[2] Bar-Shalom Y, Li X, Kirubarajan T 2008 Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software (Wiley) Available at: http://onlinelibrary.wiley.com/doi/book/10.1002/0471221279
[3] Aditya P P, Apriliani E, Arif D K, Baihaqi K 2018 Estimation of three-dimensional radar tracking using modified extended kalman filter Journal of Physics: Conference Series 974 http://iopscience.iop.org/article/10.1088/1742-6596/974/1/012071
[4] Duan Z, Han C, Li X R 2007 Sequential Nonlinear Tracking Filter with Range-rate Measurements in Spherical Coordinates IEEE Trans. Aerospace and Electronic Systems 43(1) 239–250 http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7375744
[5] Pudovkin A, Panasyuk Yu, Belyaev M, Danilov S, Moskvitin S, Varepo L, Nagornova I 2021 Development and research of the rangefinder of the information and measurement system of air traffic control based on data from on-board sensors of the aircraft Journal of Physics: Conference Series 1901 DOI 10.1088/1742-6596/1901/1/012020
[6] Pitre R R, Jilkov V P, Li X R 2005 A comparative study of multiple-model algorithms for maneuvering target tracking Proc. SPIE 5809, Signal Processing, Sensor Fusion, and Target Recognition http://doi.org/10.1117/12.609681
[7] Shtarkov Yu M 2013 *Universal coding. Theory and algorithms* (Moscow: Fizmatlit) ISBN 978-5-9221-1517-9.
[8] V’yugin V 2011 On universal algorithms for adaptive prediction *Problems of Information Transmission* 47(2) 166–189 DOI 10.1134/S0032946011020074
[9] Dawid A P 1985 Calibration-based empirical probability [with discussion] *Ann. Statist.* 13 1251–1285 DOI 10.1214/aos/1176349736
[10] Kakade S M, Foster D P 2004 Deterministic calibration and Nash equilibrium *Lecture Notes in Computer Science* (Berlin: Springer) 33–48 http://doi.org/10.1016/j.jcss.2007.04.017
[11] Pyatko S G, Krasov A I (eds.) 2004 Automated air traffic control systems: new information technologies in aviation 2004 (SPb: Polytechnic) ISBN 5-7325-0779-5
[12] Albers S 2006 Online Algorithms. In: Goldin D, Smolka S A, Wegner P (eds) *Interactive Computation* (Springer, Berlin, Heidelberg) http://doi.org/10.1007/3-540-34874-3_7
[13] Vyugin V V 2020 Mathematical foundations of the theory of machine learning and prediction (Moscow: MTsNMO)
[14] Hartikainen J, Solin A, Särkkä S 2021 Optimal Filtering with Kalman Filters and Smoothers a Manual for the Matlab toolbox EKF/UKF. Version 1.3. Available at: http://citeseerx.ist.psu.edu/viewdoc/versions?doi=10.1.1.331.150
[15] Vovk V 1990 Aggregating strategies *Proceedings of the 3rd Annual Workshop on Computational Learning Theory* (M. Fulk and J. Case, editors) (San Mateo, CA: Morgan Kaufmann) 371–383 http://doi.org/10.1016/B978-1-55860-146-8.50032-1