

# Probabilistic Approach to the Synthesis of Algorithm for Solving Problems

Anatoly Aleksandrovich Mikhaylov<sup>1</sup> & Svetlana Anatolyevna Bazuyeva<sup>1</sup>

<sup>1</sup> Federal State Budget Educational Institution of Higher Professional Education "Platov South-Russian State Polytechnic University (Novocherkassk Polytechnic Institute)", Novocherkassk, Russian Federation

Correspondence: Anatoly Aleksandrovich Mikhaylov, 132 Prosveshcheniya Str., 346428 Rostov region Novocherkassk, Russian Federation. E-mail: mih01@mail.ru

Received: December 15, 2014

Accepted: December 29, 2014

Online Published: March 25, 2015

doi:10.5539/mas.v9n5p125

URL: <http://dx.doi.org/10.5539/mas.v9n5p125>

## Abstract

This paper, based on the content of the axioms for the randomized algorithm, considers the collection of using correct algorithms at synthesis for solving the problem of probabilistic hidden Markov model. Application of this model allows forming algorithm with its flexibility according to a substantial situation for ensuring structural and functional stability of the program realizing this algorithm. We found that randomization of the algorithm, increasing its flexibility and efficiency, does not improve its risk compared with the corresponding deterministic algorithm. The synthesis of the algorithm based on hidden Markov model implies that the available observed data is used to determine hidden parameters of the most likely sequence of states, determining the synthesized algorithm. At the first strategy step, the "back and forth" algorithm is used to evaluate how well the model matches with the input data of the synthesized algorithm. At the second stage, the given hidden Markov model with the space of hidden states, initial probabilities of presence in state  $i$  and probabilities of transition from state  $i$  to state  $j$ , and basing on the observed states and using the Viterbi algorithm, the Viterbi path is found. At the third strategy stage, the hidden Markov models are corrected by optimizing the parameters of the model using the Baum-Welch algorithm.

**Keywords:** hidden markov model, "back and forth" algorithm, viterbi algorithm, Baum-Welch algorithm

## 1. Introduction

At the basis of any purposeful activity, there are decision-making procedures of the problem as a sequence of actions (algorithms) that convert raw data to achieve the goal (the desired result of solving the problem) at the lowest cost. The problem can be solved by different algorithms, which have their own advantages and disadvantages at its decision. At that, the synthesis of solutions includes the construction of a qualitative model of the problem, with subsequent writing it in mathematical form, construction of the objective function variables and study of the effect of variables on the objective function.

When making decisions under the conditions of certainty, the criteria approach is used; in which each alternative is evaluated using criteria. However, the multi-factorial impact on variables of the objective function of the algorithm for the environmental conditions and controlling actions restricts the use of this approach because of a rare situation of complete certainty of the consequences of choice, and because in practice, several different criteria are often used and rarely one alternative is the best for each of them.

### 1.1 Literature Review

To overcome this limitation, currently stochastic algorithms are widely used (Fedosova, A. V. & Zavriev, S. K., 1988; Wardi, Y. 1990; Lukshin, A. V., & Smirnov, S. N., 1988; Simonov, N. A. 1995) and their varieties - probabilistic and heuristic algorithms, which are characterized by the ability to control their flexibility and efficiency that defines the use of the algorithm in the formation of a probabilistic model. At that, the randomized algorithm sets the strategy for solving the problem in several ways or methods resulting in the probability of achieving the result (Kazharov, A. A., & Kureichik, V. M., 2010; Levanova, T. V., 2004), and for heuristic algorithms the achieving of the end result is not clearly predetermined, as well as the entire sequence of actions is not defined, and all the actions of the performer are not revealed. The example of heuristic search algorithms used for solving optimization problems and modeling by random selection, combination, and variation of the

desired parameters using the mechanisms resembling biological evolution are genetic algorithms (Kureichik et al., 2006)

The purpose of this paper is to investigate the collection of using of a probabilistic model of the algorithm at syntheses of correct algorithms. That model determines the ability to control the flexibility of the algorithm to provide structural and functional stability program implementing this algorithm, during, for example, hacker at it.

## 2. Methodology

### 2.1 General Analysis of the Decision Making Process

In accordance with (Rudakov K. V.<sup>1</sup> 1987; Rudakov K. V.<sup>2</sup> 1987; Rudakov K. V.<sup>3</sup> 1987; Rudakov K. V. 1988) let us define the problem of synthesizing correct algorithm as the process of converting primary (original) information  $J_{in} = \{X(S_i) | S_i \in U\}$  at some collection  $U = \{S_i\}$ , which elements are in the form of objects  $X(S_i)$  and are described for their observations through  $I_{in}$ , so  $J_{in} = \{I_{in}\}$ . Depending on the application,  $I_{in}$  can be a number or a non-numerical mathematical object (a symbol of the abstract alphabet, a vector, a sequence of characters, a function of a single variable (process) or a function of two variables (an image), and a function of a more complex domain).

The problem is solved using its model  $M$  defining the collection of solution algorithms  $A_M \subseteq \{a | a: J_{in} \xrightarrow{M} J_{out}\}$ ,

i.e. using the collection of algorithms  $A_M$ , the algorithm  $a \in A_M$  implements mapping from the space of initial information  $J_{in}$  to the space of final information  $J_{out}$ . As a result, the "black box" turns "white", which is characterised by the structural information of the problem  $I_s$ , which is mapping completely and correctly the essence of the problem  $J_{in}$ . Structural information of the problem  $I_s$  allocated at the subcollection of the allowable mappings, model  $M[I_s]$ , and defining the correct solution to the problem  $J_{out} = \{I_s\}$ . Algorithm  $a$ , implementing admissible mapping, which is determined by structural information  $I_s$ , is the correct solution to the problem.

The collection of stochastic algorithms contains not only the rules in the class of strict solutions (*deductive* approach), *abductive* approach (inverse deduction) is determined as well, which is used to determine the most probable initial predication of the conclusions by the inverse transformation and *inductive* to identify the most probable regularities arising from comparison of the initial data and the known results. At that, making this choice can be realized in various options:

- Collection of algorithms can be discrete or continuous.
- Selection mode can be one-time or iterative.
- Evaluation of alternatives shall be made according to the criteria of different types.
- The consequences of each alternative choice may be known (conditions of certainty), not known for sure, but the probability of effects can be estimated (conditions of statistical uncertainty), are unknown, and the probability of effects cannot be assessed (conditions of uncertainty).

### 2.2 The General Analysis of the Bayesian Approach to Problem Solving

#### 2.2.1 The Choice of Criteria for Evaluating the Quality of the Solution for the Problem

Bayesian approach is widely used at present time for solving a variety of parametric approach (Harin, A. Yu., 2013; Orlov, A. I., 2004), based on the knowledge of the distribution density of the input variables of the problem, for which the correct algorithm for this problem can be written in explicit analytical form using formalised structural information of algorithm  $I_s$ .

Bayesian problems and the basic properties of Bayesian algorithms are defined by mathematical properties of collections of input initial information  $J_{in}$ , states  $X$  and solutions  $D_{out} \in I_{out}$ , which determine the efficiency of the algorithms. The algorithm is defined at a finite collection of the observed parameters  $y \in Y$  and at finite collection of hidden states of the algorithm  $x \in X$ , in addition, it shall be consistent with the input data  $D_{in} \in J_{in}$ . At the collection  $Y \times X$  of all possible pairs of observations  $y \in Y$  and states  $x \in X$ , the distribution of joint probability  $P_{YX}(y, x): Y \times X \rightarrow \mathbf{R}$  is set.

To introduce the problem solution quality criteria let us define for the collection  $x \in X$  and possible solutions  $d_{out} \in D_{out}$  the loss function  $W(x, d_{out}): X \times D_{out} \rightarrow \mathbf{R}$ . For algorithm  $a: Y \rightarrow D_{out}$ , which assigns to each observation  $y \in Y$  a solution  $a(y) \in D_{out}$ , shall we define the risk  $R(a)$  as a mathematical expectation of the loss function for the algorithm  $a$ . At that, the Bayesian problem of statistical solutions is in the fact that for the set collections  $Y, X$ ,

$D_{out}$ , set by the algorithm  $a: Y \rightarrow D_{out}$ , which minimises the Bayesian risk

$$R(a) = \sum_{y \in Y} \sum_{x \in X} P_{YX}(y, x) W(x, a(y)). \quad (1)$$

The results of Bayesian algorithm  $a^*$  functioning is the solution of the Bayesian problem with the minimal risk  $R(a)$ . At that, the collections of states  $X$  and solutions  $D_{out}$  have different forms. Depending on the restrictions on the mathematical form from the elements of the collection of observations  $Y$ , states  $X$  and solutions  $D_{out}$ , the formulation of the Bayesian problem is specified.

### 2.2.2 Generalized Bayesian Formulation of Problems Solving

Bayesian formulation of the problem solving with model  $M$  defines the expansion of the collection of algorithms  $A_M$  in a way that it included not only algorithms of the form  $a: Y \rightarrow D_{out}$ , but all possible distributions of collections  $P_s(d_{out}|y)$ , i.e. in stochastic algorithms, each value  $y$  randomly gets a suitable solution  $d_{out}$  in accordance with the probabilities  $P_s(d_{out}|y)$ . The search for the best element is performed in the data of the stochastic algorithms, in which element at a fixed value of  $x$  the same deterministic solution  $d_{out}=a(y)$ , is taken which is in contradiction with the random nature of the state in which the algorithm is present.

For Bayesian formulation for solving the problem on finite collections shall we define  $Y, X, D_{out}$  with the distribution of probabilities  $P_{YX}: Y \times X \rightarrow \mathbf{R}$  and a loss function  $W: X \times D_{out} \rightarrow \mathbf{R}$ , the stochastic algorithm  $a_s: D_{out} \times Y \rightarrow \mathbf{R}$ , which risk is

$$R_s = \sum_{y \in Y} \sum_{x \in X} P_{YX}(y, x) \sum_{d_{out} \in D_{out}} P_s(d_{out} | y) W(x, d_{out}). \quad (2)$$

*Theorem:* For any stochastic algorithm, there is a deterministic algorithm  $a: Y \rightarrow D_{out}$ , which risk

$$R_{det} = \sum_{y \in Y} \sum_{x \in X} P_{YX}(y, x) W(x, a(y)) \quad (3)$$

is not more than  $R_s$ , i. e. the Bayesian problem can be reduced to finding a deterministic algorithm  $a: Y \rightarrow D_{out}$ .

*Proof:* Let us rewrite equation (2) in another form,

$$R_s = \sum_{y \in Y} \sum_{d_{out} \in D_{out}} P_s(d_{out} | y) \sum_{x \in X} P_{YX}(y, x) W(x, d_{out}). \quad (4)$$

Equality  $\sum_{d_{out} \in D_{out}} P_s(d_{out} | y) = 1$  holds for any  $y \in Y$ , and the inequality  $P_s(d_{out} | y) \geq 0$  holds for any  $d_{out} \in D_{out}$  and  $y \in Y$ , thus

the following equation is fair:

$$R_s \geq \sum_{y \in Y} \min_{d_{out} \in D_{out}} \sum_{x \in X} P_{YX}(y, x) W(x, d_{out}). \quad (5)$$

Let  $a(y)$  serve as a designation for any value of  $d_{out}$ , for which

$$\sum_{x \in X} P_{YX}(y, x) W(x, a(y)) = \min_{d_{out} \in D_{out}} \sum_{x \in X} P_{YX}(y, x) W(x, d_{out}). \quad (6)$$

This algorithm  $a: Y \rightarrow D_{out}$  is deterministic which is not worse than the stochastic  $a_s$

$$R_s \geq \sum_{y \in Y} \sum_{x \in X} P_{YX}(y, x) W(x, a(y)), \quad (7)$$

i.e. for a determined algorithm  $a$  the risk  $R_{det}$  is  $\leq R_s$ .

## 3. Results

### 3.1 Bayesian Algorithm Problem Solving Model

Conceptual model of the deterministic algorithm is a sequence of machine instructions (operators)  $v_i$ , defined at the collection of the machine instructions  $V = \{v_i\}$ ,  $i = \overline{1, n_{mk}}$ , where  $n_{mk}$  is the number of machine instructions contained in the sample class of algorithms, which are understood as a system of instructions of the processors class in use. For the stochastic algorithm model, variables are defined by the following meaningful axioms:

- observable (hidden) events are represented as a sequence, ordered by time (Markov property, ensuring convergence of the strategy), thus  $t$ -th hidden variable  $s_t$  at known  $(t-1)$ -th variable  $s_{t-1}$  is independent from

all previous ( $t-1$ ) variable that is the transition function.

$$P(s_t | s_{t-1}, s_{t-2}, \dots, s_1, o_1) = P(s_t | s_{t-1}), \quad (8)$$

and  $t$ -th known observation  $o_t$  depends only on  $t$ -th state.

$$P(o_t | o_{t-1}, o_{t-2}, \dots, o_1) = P(o_t | s_t), \quad (9)$$

that is not time dependent.

- the two sequences  $\mathbf{S} = s_1 s_2 \dots s_T$  and  $\mathbf{O} = o_1 o_2 \dots o_T$  shall be aligned, i.e. each observed event  $o_t$  shall correspond to one hidden event  $s_t$ , i.e. the value of the observed variable  $y(t)$  depends only on the value of hidden variable  $x(t)$  (both at the point of time  $t$ ).
- the calculation of the most likely hidden sequence before time  $t$  depends only on the observed event at time  $t$ , and the most likely sequence until  $t-1$ .

These axioms (8, 9) determine the use as a model of the algorithm of Hidden Markov Model (HMM) (Figure 1) (Rabiner L. R., 1989; Mottl V. V. & Muchnik I. B., 1999)

$$\Theta = (X, Y, \mathbf{P}, \mathbf{P}^s, \Pi). \quad (10)$$

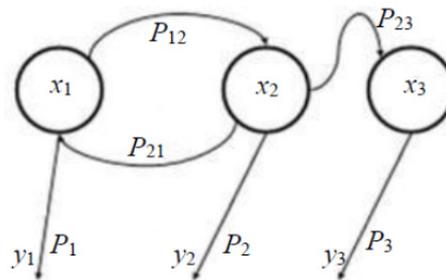


Figure 1. Transition diagram to HMM:  $x$ –hidden states;  $y$ – observed results;  $P_{ij}$ – transitions probabilities;  $P_i$ – probability of the result.

In this model, 1) The collection of hidden states of model  $X = \{x_i\}$ ,  $i = \overline{1, n}$ ,  $n$ –is the number of states of model, is used to define the hidden state of model  $s_t \in S$  at the point of time  $t$ .

2) At the collection of the observed values  $Y = \{y_i\}$ ,  $i = \overline{1, m}$ ,  $m$ –the number of various symbols of the observation by the model can deliver the sequence of the observed values represented in the form of  $\mathbf{O} = o_1 o_2 \dots o_T$ , where  $o_t$ – is the symbol of the discrete alphabet  $V = \{v_i\}$ , consisting of machine instructions identifiers  $v_i$ ,  $T$ – is the number of symbols in the observed sequence.

3) The distribution the probability for the state transition (transition probability matrix)  $\mathbf{P} = \|P_{ij}\|$ , where  $P_{ij} = P[s_{t+1} = x_j | s_t = x_i]$ ,  $1 \leq j, i \leq n$  – is the probability of the transition from the state  $s_t = x_i$  at the point of time  $t$  to the state  $s_{t+1} = x_j$  at the next point of time  $t+1$ .

4) Distribution of probability for appearing observation symbols in state  $j$ ,  $\mathbf{P}^s = \|P_j(k)\|$ , where  $P_j(k) = P[v_k | s_t = x_j]$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq m$ .

5) Initial distribution of collections of state  $\Pi = \{\pi_i\}$ ,  $\pi_i = P[s_1 = x_i]$ , where  $i = \overline{1, n}$ .

In short form HMM has the form  $\Theta = (\mathbf{P}, \mathbf{P}^s, \Pi)$  and represents a doubly stochastic process consisting of a pair of random variables  $\{o_1, \dots, o_t, s_1, \dots, s_t\}$ , where  $o_t$ – is known discrete observations describing the appearance of the observation symbols (machine instructions)  $s_t$ –are "hidden" discrete values, determining changes in the state of the model, but it is not known how many and what state connection there are between them (unknown parameters of the model).

### 3.2 The Strategy of Solving Algorithm Synthesis

The accepted model of the algorithm determines the iterative strategy of algorithms synthesis, which is reduced to the transformation of the original model  $\Theta = (\mathbf{P}, \mathbf{P}^s, \Pi)$ , leading to the goal in the form of an optimal model of correct algorithm  $\overline{\Theta} = (\overline{\mathbf{P}}, \overline{\mathbf{P}}^s, \overline{\Pi})$ , by iterative recalculation of the model parameters  $\overline{\Theta} = (\overline{\mathbf{P}}, \overline{\mathbf{P}}^s, \overline{\Pi})$  until convergence.

**Statement of the problem.** The observed value  $y(t)$  is known in the form of sequence  $Y_T = y_0, y_1, \dots, y_{T-1}$  of length  $T$ , produced by the sequence  $O = o_1 o_2 \dots o_T$ . The probability to observe  $y(t)$  is equal to  $P(Y_T) = \sum P(Y_T | X_T) P(X_T)$ , where sum is defined over all possible hidden variables  $x(t)$  in the form of sequence of hidden nodes  $X_T = x_0, x_1, \dots, x_{T-1}$ . According to the available data  $y(t)$ , it is required to determine the hidden parameters of the most likely sequence of states of the Markov chain  $S = s_{i1} \dots s_{iT}$ .

The solution of the formulated problem can be achieved in three stages, each of which implements by a known algorithm.

**First stage:**  $T$  steps in model  $\Theta = (\mathbf{P}, \mathbf{P}^s, \Pi)$  give the sequence of observations  $O_{1,T} = o_1, \dots, o_T$ , while at the point in time  $t$  for the state  $s: X_t = i$  probability  $P(O_{1,t-1} | X_t = s)$  of the fact that during the transitions there was formed the sequence of observations  $O_{1,t-1}$ , and  $P(O_{t,T} | X_t = s)$  – is the probability that the sequence of observations  $O_{t,T}$  is observed after that state.

There is the search for the probability  $P(X_t = s | O) = P(X_t = s | O_{1,t-1} \cap O_{t,T})$  of the fact that at the point of time  $t$  the chain will be in the state  $s$ .

1. For the random state  $s$  at random step  $t$ , the probability  $P(O_{1,t} | X_t = s_i)$  that the fact the the sequence  $O_{1,t}$  was produced on its way for the following  $t$  can be made recursively:

$$P(O_{1,t} | X_t = s_i) = \sum_{j \in S} P(O_{1,t} | X_t = s \cap X_{t-1} = j) = \sum_{j \in S} P(O_{1,t-1} | X_{t-1} = j) P(X_t = s | X_{t-1} = j) P(O_t = o_t | X_t = s). \quad (11)$$

The probability to get into the state  $s$  at the  $t$ -th step, taking into account that the event  $o_t$  will happen after the transition will be equal to be in the state  $j$  at the  $t$ -th step multiplied by the probability to transit from the state  $j$  to  $s$ , having performed the event  $o_t$  for all  $j \in S$ .

2. The probability that after a random state  $s$  there will be produced a sequence  $O_{t+1,T}$  is defined recursively:

$$P(O_{t+1,T} | X_t = s) = \sum_{j \in S} P(O_{t+1,T} | X_{t+1} = j) P(X_{t+1} = j | X_t = s) P(o_{t+1} | X_t = s). \quad (12)$$

3. To find the probability that the chain of events will be made,  $P(O)$ , there shall be got the sum of the product for the two probabilities for all states at a random step  $t$ :

$$P(O) = \sum_{s \in S} P(O_{1,t} | X_t = s) P(O_{t,T} | X_t = s), \quad (13)$$

as the future of the Markov chain does not depend on the past and the probability of observation of the event  $O_t$  does not depend on the past observations of the event  $O_{1,t-1}$ , then the probability that at the point in time of time  $t$  the chain will be in the state  $s$ :

$$P(X_t = s | O) = P(X_t = s | O_{1,t-1} \cap O_{t,T}) = \frac{P(X_t = s | O_{1,t-1}) P(X_{t+1} = s | O_{t,T})}{P(O)}. \quad (14)$$

This problem is solved by the "back and forth" algorithm (Binder et al., 1997; Lawrence, R., & Rabiner, A., 1989; Lawrence et al., 1986), which allows to find in the hidden Markov model the probability of getting into the state  $s$  at the  $t$ -th step at the sequence of observations  $O$  and (hidden) sequence of states  $X$ .

**Second stage:** For the given HMM with the space of states  $S = \{s_1, s_2, \dots, s_K\}$ , the initial probabilities  $\pi_i$  of being in the state  $i$  and probabilities  $P_{ij}$  of transition from the state  $i$  to the state  $j$ , according to the observed  $y_1, \dots, y_T$  and the majority of the initial information  $J_{in}$  are used to find the most plausible sequence of the states for the hidden nodes  $S = s_{1..T}$  (the Viterbi path) which describes the given model better. Then the most likely sequence of states  $x_1, \dots, x_T$  is defined by the recursive relations:

$$V_{1,k} = P(y_1 | k) \pi_k, \quad V_{t,k} = P(y_t | k) \max_{x \in S} (P_{x,k} V_{t-1,x}), \quad (15)$$

where  $V_{t,k}$  – is the probability of the most likely sequence of states responsible for appearing of the first  $t$  of the observed symbols ending in the state  $k$ . Viterbi path is searched based on the states  $x$ , satisfying the equation (15), i.e.,

$$x_T = \arg \max_{x \in S} (V_{T,x}) \quad (16)$$

$$x_{t-1} = \begin{cases} \arg(V_{t,k}), & \text{если } t > 1, \\ k, & \text{если } t = 1. \end{cases} \quad (17)$$

The solution of this problem uses the Viterby algorithm (*Andrew Viterbi*), allowing to obtain the most likely sequence of hidden states (*Viterbi path*) of the Hidden Markov model on the basis of a sequence of observations (Forney, G. D. Jr., 2005; Forney, G. D. Jr., 1973; Viterbi A. D., & Omura, J. K., 1982; Zolotariov, V. V., & Ovechkin, G. V., 2004; Morelos-Zaragoza, R., 2006.).

**Third stage:** The given output sequence of observations  $O$  is used to define unknown parameters of HMM  $\Theta$ , maximising the collection of observations  $O$

$$\Theta^* = \max_{\Theta} (P(O|\Theta)). \quad (18)$$

At that, the number of point in times  $r$ , where the observations are made, is set beforehand and comprises of the following steps:

- 1) definitions of all the model's states sequence  $S_i = \{s_{i1}, \dots, s_{ir}\}$ ,  $i=1, \dots, r$  of the problem-solving system at the specified point in times;
- 2) evaluation of probabilities  $P(S_i)$  for appearing of each sequence  $S_i$ ,  $i=1, \dots, r$ , identified at the previous step, by calculating the probabilities of transitions between the works of the model state within the range of the established control points in time, namely:

$$P(S_i) = \prod_{t=1}^{r-1} P_{s_{i,t}, t+1}, \quad (19)$$

where  $P_{s_{i,t}, t+1}$  is the collection of transition from the state  $s_{it}$ , in which the system was at the point in time  $t$ , into the state  $s_{i,t+1}$ , occupied at the point in time  $t+1$ ;

- 3) collection of appearing of the observed sequence  $X = \{x_1, \dots, x_r\}$  for states sequence  $S_i$ ,  $i=1, \dots, r$

$$P_{ix} = P(S_i) \prod_{j=1}^r P_{x_i, j}, \quad (20)$$

where  $P_{x_i, j}$  is the collection of getting the observed characteristic  $x_j$  at state  $s_{ij}$ ;

- 4) the choice of the most likely state sequence  $S_{\max} \in \{S_i\}_{i=1, \dots, r}$ , of the corresponding biggest collection.

$$\{P\}_{x, \max} = \max_i P_{ix} \quad i = 1, \dots, r. \quad (21)$$

This strategy step is fully implemented by the Baum – Welch algorithm (Lawrence, R., & Rabiner, A., 1989; Baum, L. E., 1972).

#### 4. Discussion and Conclusions

The conducted analysis of the probabilistic approach to the synthesis algorithm for solving the problem allows formulation of the following conclusions:

1. Meaningful axioms for the model of randomized algorithms allow for the collection of using them as a model of the algorithm of Hidden Markov Model, HMM, which is applicable not only to the original types of algorithms, but also to the correction algorithms, as well as for synthesized compositions.
2. Using a probabilistic model of the algorithm, it is possible to control the relationship between the algorithm variables by setting the flexibility of the algorithm to achieve the desired structural and functional stability of the software implementing the algorithm.
3. Randomization of the algorithm, increasing its flexibility and efficiency, does not improve its risk compared with the corresponding deterministic algorithm.
4. The problem of synthesis of algorithm based on HMM is that the available observed data is used to determine the hidden parameters of the most likely sequence of states.
5. At the first strategy stage for the given model parameters  $\Theta = (\mathbf{P}, \mathbf{P}^s, \Pi)$  and sequence of hidden  $S$  and relevant

observable states  $O$ , the probability of occurrence of these sequences is determined. Thus on the first step, the model  $\Theta$  and collection of initial data  $D_{in}$  get the determination of  $\pi_i = P(D_{in} | \Theta)$  and the evaluation of how well the  $\Theta$  model is agreed to the initial data. This problem in such a formulation is solved using the "back and forth" algorithm which allows finding in HMM the collection of getting into state  $s_i$  at  $t$ -th step at the sequence of observation  $O$  and (hidden) sequence of states  $S$ .

6. At the second stage, at the given hidden Markov model  $\Theta$  with the space of hidden states  $S = \{s_1, s_2, \dots, s_K\}$ , initial collections  $\pi_i$  presence in state  $i$  and collections  $P_{ij}$  of transition from the state  $i$  to the state  $j$ , and basing on the observed states,  $o_1, \dots, o_T$  the Viterbi path can be found  $S_V = s_1 \dots s_T$ , which described this model in the fullest way. It is natural to use the Viterbi algorithm for this purpose.

7. The third stage of the strategy implies correcting the hidden Markov models by optimizing the model parameters  $\Theta = (\mathbf{P}, \mathbf{P}^s, \Pi)$  in a way to maximise  $P(O | \Theta)$  at the observed data  $O$ . This strategy step is fully implemented by the Baum – Welch algorithm.

The collection of using the synthesis algorithm, grounded in the article, for solving the HMM problem determines the further research of control by flexibility of the synthesized algorithm to provide structural and functional stability of the software implementing this algorithm.

## References

- Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process. *Inequalities*, 3, 1–8.
- Binder, J., Murphy, K., & Russell, S. (1997). Space-Efficient Inference in Dynamic Probabilistic Networks. *Int'l. Joint Conf. on Artificial Intelligence*, 1(5), 1292–1296.
- Fedosova, A. V., & Zavriev, S. K. (1998). A Stochastic Algorithm for Solving Convex Semi-Infinite Programming Problems. Abstracts The 2nd Moscow International Conference On Operations Research (Moscow, November 17–20), 14.
- Forney, G. D. Jr. (2005). The Viterbi Algorithm: A Personal History. Viterbi Conference, University of Southern California, Los Angeles, March 8.
- Forney, G. D. Jr. (1973). The Viterbi Algorithm. *Proceedings of the IEEE*, 61(3), 268–277. <http://dx.doi.org/10.1109/PROC.1973.9030>
- Kazharov, A. A., & Kureichik, V. M. (2010). Ant algorithms for solving transport problems. Bulletin of the Russian Academy of Sciences. *Theories and system of control*, 1, 32–45.
- Kharin, A. Y. (2013). Bayesian robustness and consistency of statistical decisive rules.–Minsk: BSU. Pages 207.
- Kureichik, V. M., Gladkov, L. A., & Kureichik, V. V. (2006). Genetic algorithms.–M.: Fizmatlit Pages 320.
- Lawrence, R., Rabiner A., & Juang, B. H. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*. January: 4–15.
- Lawrence, R., & Rabiner, A. (1989). Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), 257–286. <http://dx.doi.org/10.1109/5.18626>
- Levanova, T. V. (2004). Ant colony and simulated annealing algorithms for the problem of the p-median. *Automation and telemetry*, 3, 80–88.
- Lukshin, A. V., & Smirnov, S. N. (1988). Stochastic algorithm for solving the Boltzmann equation. *J. of comp. math. and math. phys.*, 28(2), 400–405.
- Morelos, Z. R. (2006). The art of noiseless coding. Methods, algorithms, application/transl. from English. V. B. Afanasieva. M.: Tekhnosfera. Pages 320.
- Mottl, V. V., & Muchnikov, I. B. (1999). Hidden Markov models in structural analysis of signals.–M.: Fizmatlit. Pages 351.
- Orlov, A. I. (2004). Decision theory. Study guide.–M.: Ed. Mart. Pages 656.
- Rabiner, L. R. (1989). Hidden Markov models and their use in selected applications for speech recognition. Overview TIER. *M: Nauka*, 2(77), 86–102.
- Rudakov, K. V. (1988). On the application of universal constraints in the study of classification algorithms. *Cybernetics*, 1, 1–5. <http://dx.doi.org/10.1007/BF01069519>
- Rudakov, K. V. (1987a). Universal and local restrictions on the problem of heuristic algorithms correction.

*Cybernetics*, 2, 30–35.

- Rudakov, K. V. (1987b). Completeness and universal constraints in the problem of heuristic classification algorithms correction. *Cybernetics*, 3, 106–109.
- Rudakov, K. V. (1987c). Symmetric and functional limitations in correcting the problem of heuristic classification algorithms. *Cybernetics*, 4, 73–77.
- Simonov, N. A. (1995). Stochastic algorithms for solving the first boundary problem and the Cauchy problem for convective diffusion. Novosibirsk, 35.
- Viterbi, A. D., & Omura, J. K. (1982). Principles of Digital Communication and Coding/Trans. from English. In K. Sh. Zigangirova (Ed.), M.: Radio and communication., 536.
- Wardi, Y. (1988). A Stochastic Algorithm for Optimization Problems with Continua of Inequalities. *Journal of Optimization Theory and Applications*, 56, 285–311. <http://dx.doi.org/10.1007/BF00939413>
- Wardi, Y. (1990). Stochastic Approximation Algorithm for Minimax Problems. *Journal of Optimization Theory and Applications*, 64, 615–640. <http://dx.doi.org/10.1007/BF00939427>
- Zolotariov, V. V., & Ovechkin, G. V. (2004). Noiseless coding. Methods and Algorithms: Handbook. In RAS corresponding member Yu. B. Zubareva (Ed.), M.:Hotline-Telecom, 126.

### Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).