ITHEA

International Journal INFORMATION CONTENT & PROCESSING

2015 Volume 2 Number 1

International Journal
INFORMATION CONTENT & PROCESSING
Volume 2 / 2015, Number 1

IJ ICP is official publisher of the scientific papers of the members of the ITHEA® International Scientific Society

IJ ICP rules for preparing the manuscripts are compulsory.
The **rules for the papers** for ITHEA International Journals as well as the **subscription fees** are given on www.ithea.org .
The papers should be submitted by ITHEA® Submission system http://ij.ithea.org .
Responsibility for papers published in IJ IMA belongs to authors.

International Journal "INFORMATION CONTENT AND PROCESSING" Volume 1, Number 1, 2014

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with
Institute of Mathematics and Informatics, BAS, Bulgaria,
V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politechnika de Madrid, Spain,
Hasselt University, Belgium
Institute of Informatics Problems of the RAS, Russia,
St. Petersburg Institute of Informatics, RAS, Russia
Institute for Informatics and Automation Problems, NAS of the Republic of Armenia.

Publisher: **ITHEA®**
Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: info@foibg.com
Technical editor: Ina Markova

Printed in Bulgaria

ISSN 2367-5128 (printed) **ISSN 2367-5152 (online)**

ON STRING MINING SPEECH RECOGNITION

Levon Aslanyan, Minoosh Heidari

***Abstract**: Automatic Speech Recognition (ASR) is a well developed technical area, conjoining technologies such as DSP, Machine Learning, HS Codesign, and Linguistics. For scientific research ASR is an invaluable source of ideas and solutions integrated in one important product. Unfortunately, being a business interest, advanced ASR technologies are hidden by companies. The open literature is focused round the Hidden Markov Models (HMM) which was developed earlier in 1960's and then incorporated into the HMM as an attractive technique. Open ASR systems with HMM report surprisingly low level recognition – 75%. After the beginning of HMM era many things changed in HS area – now very large memories are available, even the portable devices use multiprocessors with high computational power. It is to analyze what is achieved and what reminds to be understood with ASR in new situation and probably this helps in increasing the accuracy of recognition. Our target after this analysis will be the setting up of advanced ASR system (for Armenian language). In fact the system will be learnable, when large linguistic corpora are available, so that it can be multilingual. The feature ASR technology that is visible today is related to the Sequence Data Mining (SDM) technique. Being part of the structural data mining this technique is already studied and developed. Studies involve the topics of LCS, gene alignment and similar, but speech signal analysis provide a specific situation that is to be studied deeply, with consequent optimization and implementation. All the related thoughts are openly presented in the text below. Future work will enlarge the given ongoing prototype with means of advanced SDM and with interfaces.*

***Keywords**: hidden Markov model, string mining, and speech recognition.*

***ACM Classification Keywords**: I.2.7 Natural Language Processing.*

Introduction

Two technologies are our objective below – the Hidden Markov Models (HMM) and the String Data Mining (SDM). The former is a well known tool for many applications, including speech recognition that is our application target. SDM is specific case of Data Mining mostly used in search systems. Our discussion wants to understand the real advanced technical resource of speech recognition today. The experimental platform is presented by open tool set at [Becchetti & Ricotti, 1999].

Paper is composed in three parts. At first the HMM is considered and analyzed in order to determine the means of the best correlatedness between the state and observation sequences. Analysis shows that although probabilities are given to all pairs of states and observations, a unique observation sequence generates one dominating sequence of states moreover this sequence of states is built by constructing of individual – isolated states. We suppose that this might be the main bottleneck of the known low recognition rate in ASR with HMM. The third part of the paper considers an alternative to HMM technique for use in ASR. This is the technique, well known as the string data mining, and well developed for search purposes. Here an ASR string of observations is considered as intervals in whole. The learning set helps ASR to correspond to such intervals their classes by the ordinary means of the supervised pattern recognition. And our supposition is that by the given reason this technique fits better to the ASR needs. Finally, the section two of the work describes the open, accessible software environment of ASR, where we find all the input information tackled by HMM and SDM algorithms. [Becchetti & Ricotti, 1999] provides an open code software ASR together with the detail description of all the stages of its work. The valuable parts of the work are learning databases TIMIT (Texas Instruments and MIT) developed by a NIST project, with support of DARPA-ISTO, and ATIS (Air Travel Information System) developed by ARPA-SLS project. TIMIT provides phoneme annotated acoustic recordings to train the ASR, and ATIS is commonly used for the evaluation of word error performances. We refer to the chapter 3 “Speech signal analysis” of [Becchetti & Ricotti, 1999] that presents the DSP part of the work with signal windowing and algorithmic analysis, that outputs the multidimensional numerical vector sequences/strings. These strings correspond to the observation sequences considered by HMM and SDM.

Analysis of Hidden Markov Models in ASR

Start with defining the basic elements of the HMM.

Hidden Markov model (HMM) is a statistical model, extended form of Markov Chain model in which the system being modeled is assumed to be a Markov process, probably with unknown parameters, and the challenge is to determine the hidden states from the observable symbols.

In regular Markov model, the stats are directly visible to the observer, and therefore the state transition probabilities are the only parameters of the model. In a HMM, unlike a regular Markov model, the states are not directly visible, but variables influenced by the state are visible. Each state has a probability distribution over the set of possible output symbols. Therefore the sequence of symbols generated by an HMM gives some information, in an intermediary way, about the sequence of the states.

Formal Definition of Hidden Markov Models. The HMM is characterized by the following elements:

1) N, the number of states of the model. Although the states are hidden, for many practical applications there is often some physical significance attached to the states or to sets of states of the model. Generally the states are interconnected in such a way that any state can be reached from any other state (e.g., an ergodic model); however, we will see later that other possible interconnections of states are often of interest. We denote the individual states set as $S = \{S_1, S_2, \dots, S_N\}$, and the state at time t as q_t.

2) M, the number of distinct observation symbols per state, i.e., the size of a finite alphabet. The observation symbols correspond to the physical output of the system being modeled. We denote the individual symbols set as $V = \{v_1, v_2, \dots, v_M\}$.

3) The state transition probability distribution

$$A = (a_{ij}) \tag{1}$$

where

$$a_{ij} = P\left[q_{t+1} = S_j \middle| q_t = S_i\right], 1 \le i, j \le N.$$

Here we see that the probabilities do not vary by the time. For the special case where any state can reach any other state in a single step, we have $a_{ij} > 0$ for all i, j. For other types of HMMs, we would have $a_{ij} = 0$ for one or more (i, j) pairs.

4) The observation symbol probability distribution in state $i, B = \{b_i(k)\}$, where

$$b_i(k) = P[v_k \; at \; t | q_t = S_i], 1 \le i \le N, 1 \le k \le M. \tag{2}$$

5) The initial state distribution $\pi = \{\pi_i\}$ where

$$\pi_i = P[q_1 = S_i], 1 \le i \le N. \tag{3}$$

Concerning (1) to (3) some relations exist among the model parameters: $\sum_{j=1}^{N} a_{ij} = 1$, $\sum_{k=1}^{M} b_j(k) = 1$, and $\sum_{i=1}^{N} \pi_i = 1$. The points 3 and 4 can be demonstrated as follows (Figure 1):

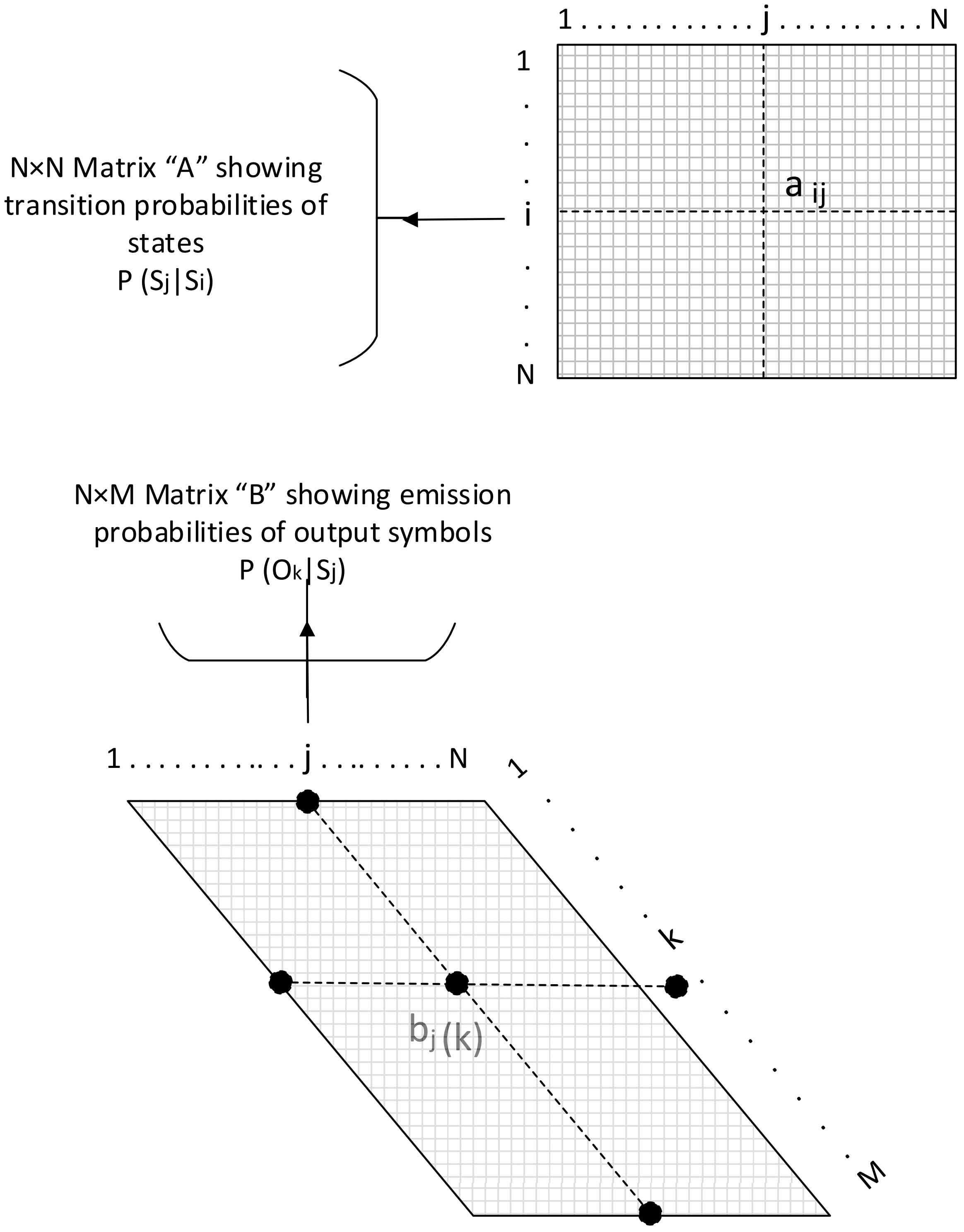

1 . . . . j . . . . N
1
.
.
.
i
.
.
.
N
a ij
N×N Matrix “A” showing transition probabilities of states
P (Sj|Si)
N×M Matrix “B” showing emission probabilities of output symbols
P (Ok|Sj)
1 . . . . j . .. . N
1
k
M
bj (k)

Figure 1. State transition probability matrix and Symbol observation probability matrix

So a complete specification of an HMM requires specification of two model parameters (N and M), specification of the state and observation symbols, and the specification of the three probability measures A, B, and π. For convenience, we use the compact notation

$$\lambda = (A, B, \pi) \tag{4}$$

to indicate the complete parameter set of the model.

An introductory example. Figure 2 shows a Hidden Markov model that represents urn-and-ball system frequently used to illustrate HMM. We use, after a correction, the example of point 4.3.3 of [Becchetti & Ricotti, 1999]. We use the simplified and the complete versions.

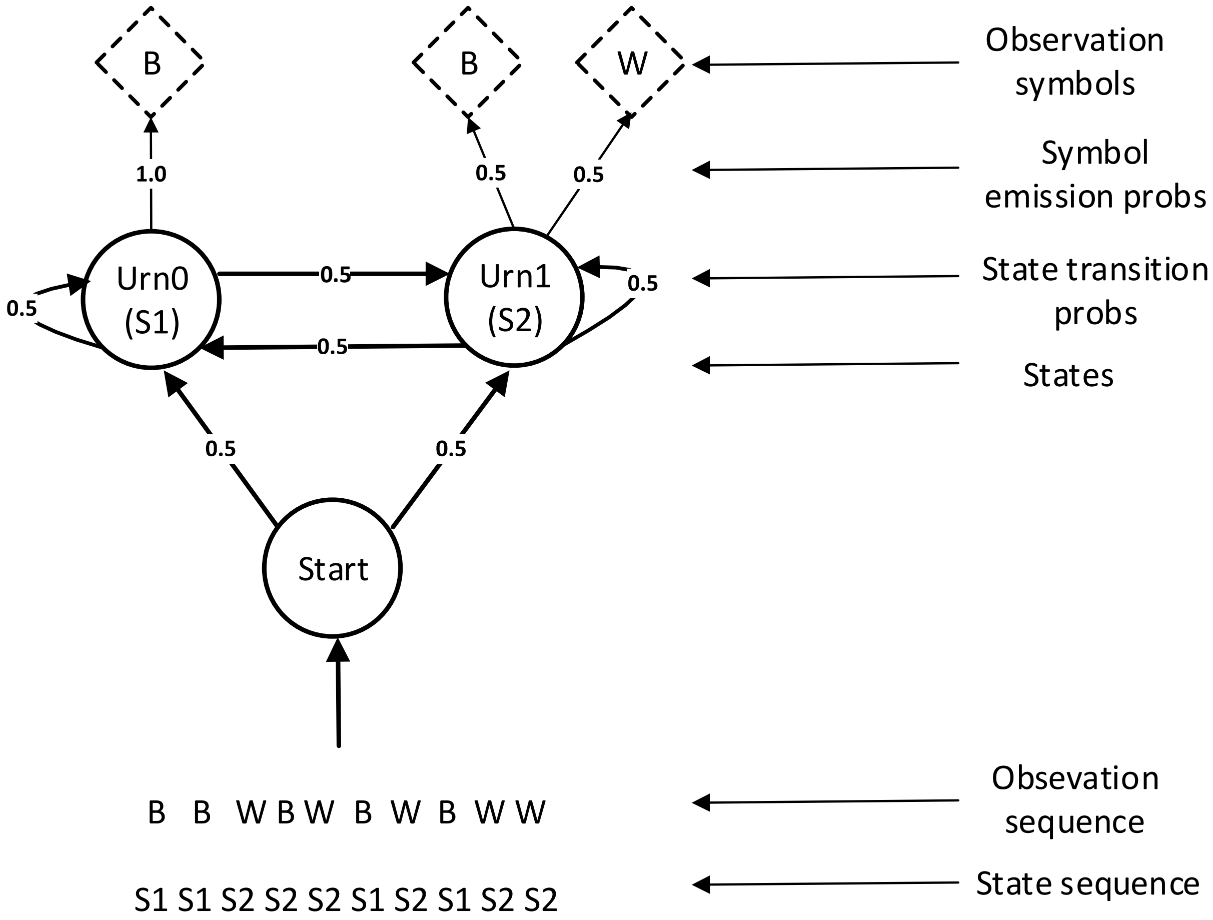


Figure 2. An HMM that models two urns containing Black and White balls.

Let us assume that there are 2 ($N = 2$ states) urns with black and white balls inside, i.e. 2 distinct colors ($M = 2$ observations as output symbols). Within each urn there is a large quantity of different balls. From urn, a ball is chosen at random, and its color is recorded as an observation. The ball is then replaced in the urn form which it was selected. A new urn is then selected according to the random selection procedure associated with the current urn. Ball selection process is repeated. This entire process generates a finite observation sequence of colors, which we would like to model as the observable output of an HMM.

Consider in particular the example when the first urn 0 is filled only with black balls (symbol 1), and the second urn 1 is filled with an equal number of the two color balls. Each extraction from urn 0 is followed by an extraction from urn 0 or 1 with equal probabilities. And the extraction from urn 1 determines a successive extraction from urn 0 and 1, with the same equal probability ½. The initial urn probabilities are supposed to be equal.

A physical process for obtaining observations is as shown on Figure 2.

Revisiting Hidden Markov Model for Figure 2, the formal definition is (S, V, N, M, A, B, π), where

1. $S = \{Urn0(s1), Urn1(s2)\}$
2. $V = \{white/(0), black/(1)\}$

1. $A = (a_{ij})$ is given as

	S1	S2
Start	0.5	0.5
S1	0.5	0.5
S2	0.5	0.5

2. $B = \{b_j(k)\}$ is given as

	B	W
S1	1.0	0.0
S2	0.5	0.5

3. $\pi_i = 0.5$

What is the probability of occurring of the sequence B W B?

To compute this probability, all the possible paths of the states to produce the output sequence (B W B) should be taken into account. The four possible paths are:

$s1 \rightarrow s2 \rightarrow s1$		$s1 \rightarrow s2 \rightarrow s2$		$s2 \rightarrow s2 \rightarrow s1$		$s2 \rightarrow s2 \rightarrow s2$	
↓		↓		↓		↓	
.5×.5×.5×.5	+	.5×.5×.5×.5×.5	+	.5×.5×.5×.5×.5	+	.5×.5×.5×.5×.5×.5	= .140925

With this simple example, we tried to understand the capability of HMM modeling which is larger in reality. The theory differs the output symbol generation – being it related to the states or to the state transitions correspondingly. The former (that we considered above) is called state-output HMM while the output generation in state transitions is known as the edge-output HMM. In this form, the output symbol is produced by the edges. So, it is called the edge-output HMM which is defined as a quadruple $\Theta = (S, Y, \{T^k\}, \pi)$, where:

S is the set of N states, Y is the set of M output symbols.

$\{\mathrm{T^k}\} = \{\mathrm{T^k} | \mathrm{k} = 1, \dots, \mathrm{M}\}$ is a set of N×N (and implicitly N×M) matrices, with the elements $\mathrm{t_{ij}(k)}$ of the joint probability distribution of states and output symbols. $\{\mathrm{T^k}\}$ must satisfy:

- For all i, j such that $1 \leq \mathrm{i}, \mathrm{j} \leq \mathrm{N}$, $a_{ij} = \sum_{\mathrm{k}} \mathrm{t_{ij}(k)}$;
- For all i such that $1 \leq \mathrm{j} \leq \mathrm{N}$, $b_i(k) = \sum_{\mathrm{j}} \mathrm{t_{ij}(k)}$;
- For all i such that $1 \leq \mathrm{i} \leq \mathrm{N}$, $\sum_{\mathrm{j,k}} \mathrm{t_{ij}(k)} = 1$;
- For all i, j such that $1 \leq i, j \leq N$ and $1 \leq k \leq M$, $t_{ij}(k) \geq 0$.

And π is the initial state probability distribution.

The joint matrices of this form of HMM can be demonstrated as shown on(Figure 3):

Consider again the example above. Replace the part of description “And the extraction from urn 1 determine a successive extraction from urn 0 and 1, with the same equal probability ½.” by the new one “The extraction of a white ball (symbol 0) or a black ball (symbol 1) from urn 1 determine a successive extraction from urn 0 and 1 correspondingly, with the equal probability ½”. Model is the same. Now also change the equal probabilities. It is easy to see that the state output model is unable to model this case which is to consider as an edge-output model.

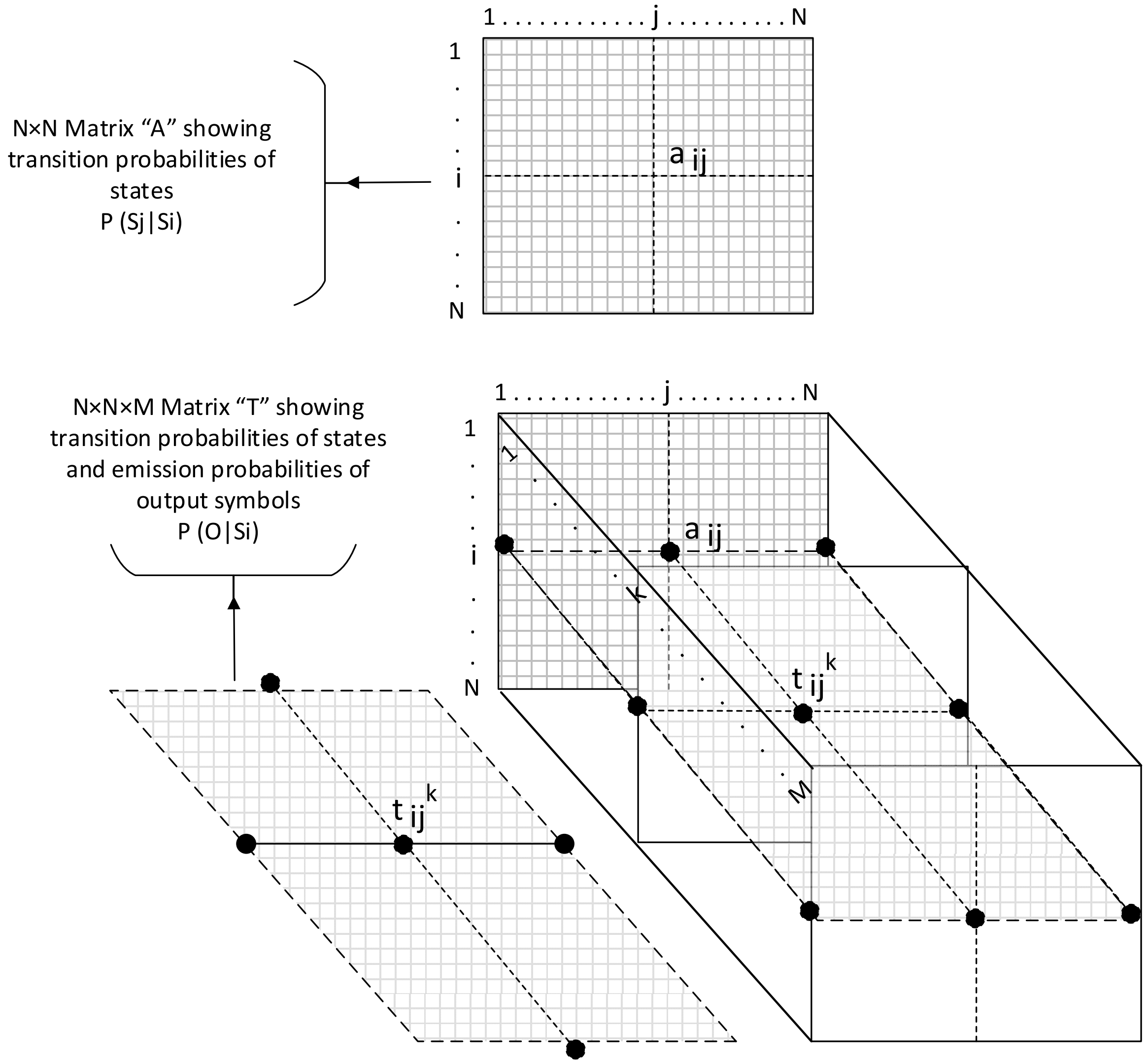

N×N Matrix "A" showing transition probabilities of states
P (Sj|Si)
1 . . . . j . . . . N
1
i
N
a ij
N×N×M Matrix "T" showing transition probabilities of states and emission probabilities of output symbols
P (O|Si)
1 . . . . j . . . . N
1
i
N
a ij
1
k
M
t ij k
t ij k

Figure 3. Joint Matrices Demonstration

The Three Basic Problems that Serve the HMM Applications. Given the HMM formalism of the previous section, mention the three basic problems of interest that must be solved for the model to be useful in real-world applications. These problems are the following:

P1. Observation Sequence Probability Computation

Given the observation sequence $O = O_1, O_2, \dots, O_T$ and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $P(O|\lambda)$, the probability of the observation sequence, given the model?

It is well known that the direct calculation of $P(O|\lambda)$ by its definition requires in its order $2T \cdot N^T$ operations. But fortunately, by the developed Forward-Backward Procedure this computation requires only $T \cdot N^2$ operations.

We will use in our analysis some internal objects and definitions of this Procedure, so we have to bring them again.

That is the so called forward variable

$$\alpha_t(i) = P(O_1, O_2, \dots, O_t, q_t = S_i|\lambda), \tag{5}$$

And the similar backward variable, that is

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \dots, O_T, q_t = S_i|\lambda). \tag{6}$$

P2. Optimal State Sequence Associated to the Acquired Observation Sequence

Given the observation sequence $O = O_1, O_2, \dots, O_T$, and the model $\lambda = (A, B, \pi)$, how do we choose a corresponding state sequence $Q = q_1, q_2, \dots, q_T$ which is optimal in some meaningful sense (i.e., best “explains” the observations)?

P3. Model Parameter Estimation that Maximize the Acquired Observation Sequence Probability

How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

Among the basic HMM problems P2. is the most meaningful. Problems P1 and P3 are computational, algorithmic, optimization tasks. Indeed these tasks are very much important to set up the final effective HMM application environment but the P2 is the place where the states and observations meet each other.

Analysis of the P2.

Here, as mentions [Becchetti & Ricotti, 1999], the difficulty lies with the definition of the optimality of a state sequence, i.e., there are several possible optimality criteria. For example, one possible optimality criteria is to choose the states, which are individually and independently most likely.

To implement this solution define the variable

$$\gamma_t(i) = P(q_t = S_i|O, \lambda) \tag{7}$$

that is the probability of being in state S_i at time step t, given the observation sequence O, and the model λ. Equation (7) can be expressed simply in terms of the forward-backward variables, i.e.

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)} \tag{8}$$

since $\alpha_t(i)$ counts for the partial observation sequence $O_1, O_2, \dots, O_t$ and state S_i, at time step t, while $\beta_t(i)$ accounts for the remainder of the observation sequence $O_{t+1}, O_{t+2}, \dots, O_T$, given state S_i at t. The normalization factor $\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)$ makes $\gamma_t(i)$ a probability measure so that

$$\sum_{i=1}^{N} \gamma_t(i) = 1 \tag{9}$$

Using $\gamma_t(i)$, we can solve for the individually most likely state q_t, at time t, as

$$q_t = arg \max_{1\le i\le N} [\gamma_t(i)],\ 1 \le t \le T. \tag{10}$$

Analysis:

a) Consider the Figure 4. explaining $\gamma_t(i)$ around the time step t.

The formula counterpart to $\alpha_t(i)\beta_t(i)$ in this picture is

$$([\textstyle\sum_{k=1}^{N} \alpha_{t-1}(k) \cdot a_{ki}]b_i(O_t)) \cdot \beta_t(i). \tag{11}$$

Having S_i fixed at time step t, and the sequence $O_{t+1}, O_{t+2}, \dots, O_T$ defined, maximization of $\beta_t(i)$ becomes an independent task from the left part of the formula (11). $b_i(O_t)$ is also fixed value, by the conditions given at time step t. So the part of formula to be maximized is the part included in square brackets. Rewrite $\alpha_{t-1}(k)$ in the form $\alpha'_{t-1}(k) \cdot b_k(O_{t-1})$ (which is valid by the definition of $\alpha_{t-1}(k)$) and then consider the transformation of the base formula into the:

$$([\textstyle\sum_{k=1}^{N} \alpha'_{t-1}(k) \cdot b_k(O_{t-1})\ a_{ki}]b_i(O_t)) \cdot \beta_t(i). \tag{12}$$

Formulas of $\alpha'_{t-1}(k)$ in (12) depend on parameters $O_1, O_2, \dots, O_{t-1}$ and the requirement that $q_{t-1} = S_k$. Our idea is to understand, and outline, the tendency here in recursive maximization of terms of type $b_k(O_{t-1})a_{ki}$. These is exactly the product of two probabilities – state probability and observation probability out of some current state S_k. The real formula requires finding a set of similar optimized pairs. In more detail, in one fragment, by O_t it is to take such state q_k for step $t-1$ that provides greater probability $b_k(O_{t-1})$ for O_{t-1} and then it is to choose the next to the q_k state with the highest probability a_{ki}. The formal description requires that the product of these two probabilities is maximal. If states and observations are correlated this can have a meaning. While so, then this is a hypothesis, functional dependency, and it is mandatory to formulate this at the beginning. Without it - states and observations - having no connection to each other - can enter into the same game.

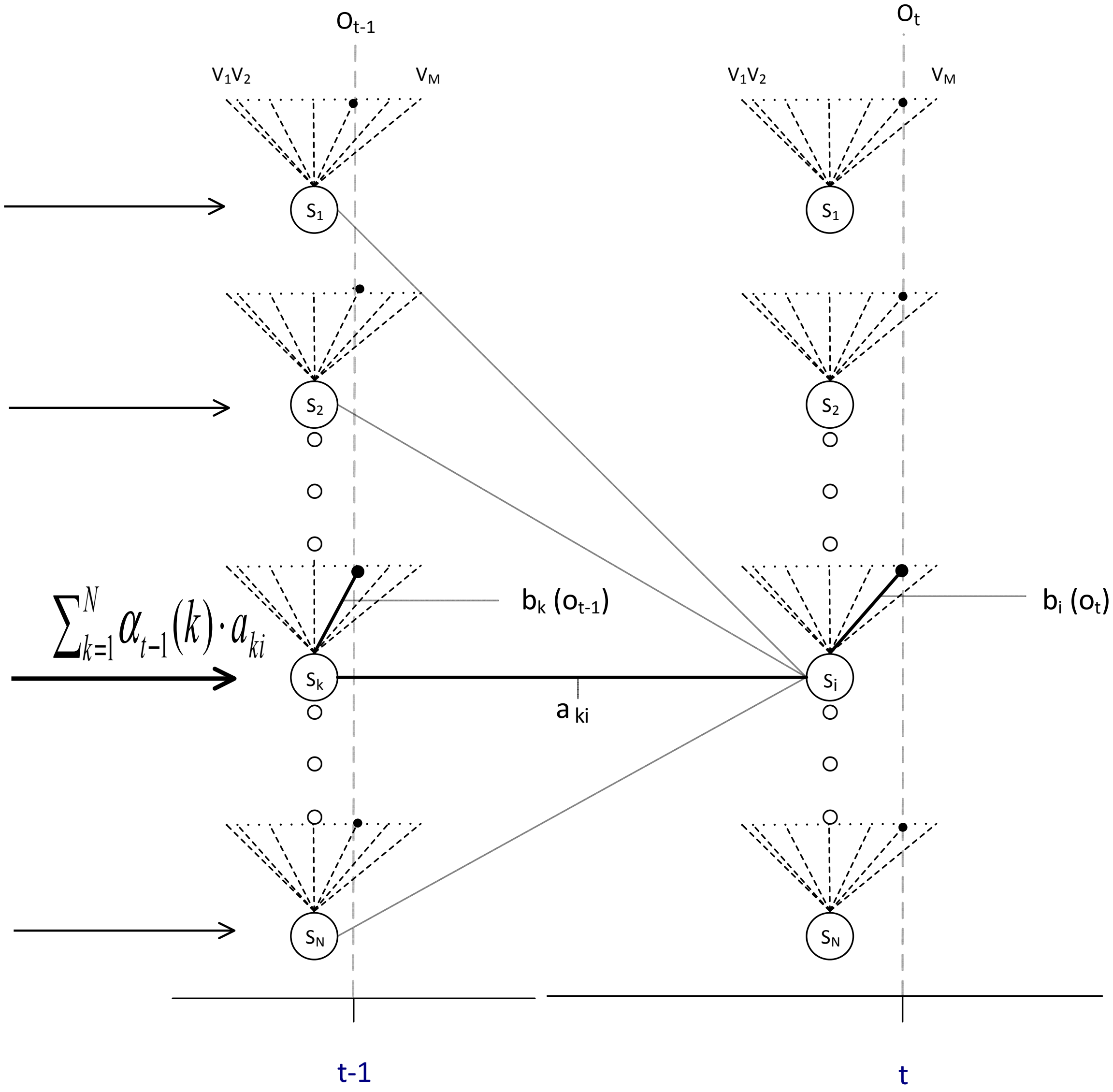


Figure 4. An Illustration for Forward and Backward Algorithm

b) Another caution to the considered algorithm is raised in [Rabiner, 1989]. Independent from one to the other construction of the optimal states at time steps $t, 1 \leq t \leq T$ can draw to a situation that the resulting state sequence can be not even valid. The simplest explanation of this is in use of the zero state transition probabilities.

In this section we analyzed the issues of applicability of HMM in ASR, in part of determination of best correlatedness between the state and observation sequences. Questions rise as it is mentioned in a), Figure 4 and in b). The formal technique for finding the single best state sequence is based on dynamic programming methods, and is called the Viterbi algorithm. The Viterbi algorithm combats the notion b) but not the a). The basic construction is again the pair $(b_k(O_{t-1}), a_{ki})$ and it is a question if consideration of 2 nearby states can be effectively manages the long state sequences. As an alternative technique, in last section of the work we will describe the use of String Data Mining technique in ASR.

Automatic Speech Recognition

This section brings the ASR internal information necessary to understand the use of HMM and SDM. Speech recognition is a type of the supervised recognition scenario. The final target is to create the alphanumeric information counterpart to the uttered speech but the way to this consists of different local steps such as setting the learning data and training, time and spectral domain analysis of signals, model composition that links the signal parameters to the linguistic elements like phonemes and words, and then can be some orthography and grammatical checks and empowerments. ASR uses both deterministic and statistical technique in model compositions. In statistical speech recognition it is wide spread to consider the Hidden Markov Models. Markov Models are really statistical but whole ASR framework in this case is quiet deterministic. And it is correct to mention another approach, statistical, when large speech corpuses are created, analyzed and characterized, and when the input signal and its parts are compared statistically with the fragments of these learning data. Anyway, our interest is to study a real example of ASR system with the use of this traditional technology, to analyze it, trying to understand the alternatives and empowerments. The base of our study in this part is the well made book [Becchetti & Ricotti, 1999]. We use it as a prototype environment to understand the whole pros and cons in area, determining novel research targets and developing technologies and solutions to them. Probably one of the starting points is that the maximal correct recognition percentage mentioned here is 75%. Usually it is 60-65%. Why the result of such hard work is so low? Is this because of incorrect or misuse of the selected models and is there a room for empowerment?

What we learn from [Rabiner, 1989]?

First, in “RES” project (Recognition Experimental System), that implemented in the book [Becchetti & Ricotti, 1999], there is information about the hardly made training data available for future experimentation such as TIMIT, and ATIS.

They are the most important speech databases used to build acoustic models of American English. TIMIT is an acronym composed by TI (Texas Instrument) and MIT, the two research centers involved in the project at NIST, National Institute of Standards and Technology, sponsored by DARPA-ISTO.

TIMIT contains a total of 6300 sentences that 630 American speakers have spoken 10 sentences. The database has been made up of the recorded waveform of the sentence, with a sampling frequency of 16 kHz, together with a time-aligned phonetic transcription of the sentence. The speakers in TIMIT have been subdivided into training and test sets using some criteria such as: Almost 20 to 30% of the database is considered for the test set, and the remaining 70 to 80% for training.

Every phrase is associated with:

.txt	the orthographic transcription of the phrase (spelling)
.wav	the wave file of the sound
.phn	the correspondence between the phonemes and the samples (the number of all Phonemes (included many times in different contexts in the mentioned speech corpora) is around 50 to 60).
.wrd	the correspondence between the words and the samples

Furthermore, there is the pair of letters (SX, SI and SA) before the name of files:

SX	are phonetically compact phrases in order to obtain a good coverage of every pair of phones
SI	phonetically varied phrases, for different allophonic contexts
SA	for dialectal pronunciation

The other mentioned database, ATIS, stands for Air Travel Information System, distributed from 1989 by ARPA-SLS project. It contains 10 722 utterances by 36 speakers.

Every phrase is associated with the following file types:

.cat	category of the phrase
.nli	phrase text with point describing what the speaker had in mind
.ptx	text in prompting form (question, exclamation,…)
.snr	SNOR (Standard Normal Orthographic Representation) transcription of the phrase (abbreviations and numbers explicitly expanded)
.sql	additional information
.sro	detailed description of the major acoustic events
.lsn	SNOR lexical transcription derived from the .sro
.log	scenario of the session
.wav	the waveform of the phrase in NIST_1A format (sampling rate, LSB or MSB byte order, min max amplitude, type of microphone, etc…)
.win	references for the interpretation

Furthermore, there are several labels (S, C, X and R):

.cat	category of the phrase
‘s’	close-speaking (Sennheiser mic)
‘c’	table microphone (Crown-mic)
‘x’	lack of direct microphone, ‘s’ spontaneous speech
‘r’	read phrases

In general, the project (RES) has the following general specifications:

It works on recorded speech files and it basically includes:

- The batch modules for acoustic model initialization and training;
- Grammar models training;
- Phoneme/word recognition;
- Performance evaluation.

It performs in speaker independent phonetic recognition:

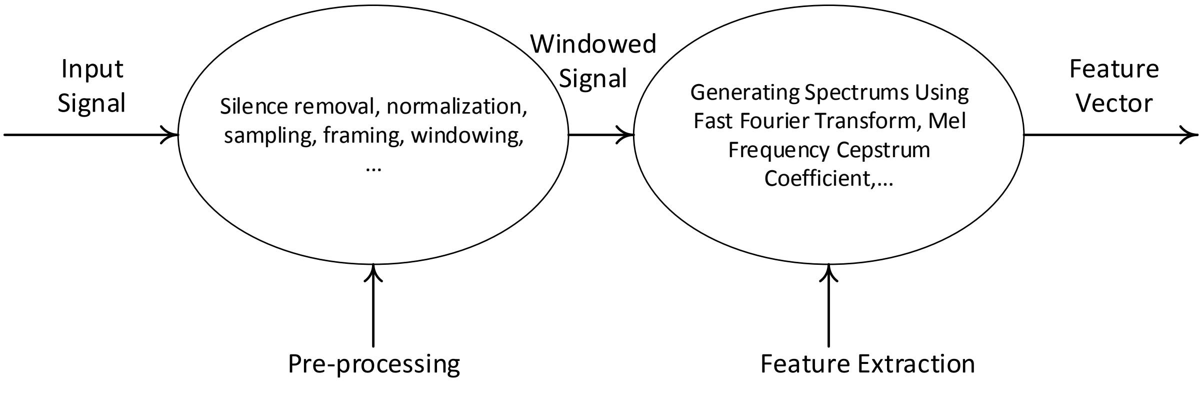
- With 69.2% of percent correct using all TIMIT test data using context independent phonetic models.

It yields 87.83% of percent correct in speaker independent word recognition on ATIS using context independent phonetic models not optimally tuned on this database.

Second are the input signal and its analysis. The base where speech signals and their fragments are compared is selected 16kHz. Then the recommended window length and the overlap is 512 and 384. The number of time spectral characteristics produced equals 39 including 13 base characteristics with their first and second order differences.

At this point it is seen that the base element for recognition is a fixed 39 length numerical vector. The input signal is coded/presented as the “overlapping” sequence of the numerical vectors. And surely the similar interpretation is possible to apply on training set signals, those already provided with the notations and transcriptions.

All the above information is acquired during the standard signal processing and feature extraction procedures:


Input Signal
Silence removal, normalization, sampling, framing, windowing, …
Windowed Signal
Generating Spectrums Using Fast Fourier Transform, Mel Frequency Cepstrum Coefficient,…
Feature Vector
Pre-processing
Feature Extraction

Speech is analyzed over short analysis window.

For each short analysis window, a spectrum is obtained using FFT (Fast Fourier Transform).

Spectrum is passed through Mel-Filters to obtain Mel-Spectrum.

Cepstral analysis is performed on Mel-Spectrum to obtain Mel-Frequency Cepstral Coefficients (MFCC).

Thus speech is represented as a sequence of Cepstral vectors.

It is these Cepstral vectors which are given to pattern classifiers for speech recognition purpose.

The following block diagram shows algorithmic internals of the time spectral domain analysis of input signals by [Becchetti & Ricotti, 1999]:

Speech Recognbition Software System Initialisation Project Block-Diagram

```
DbaseVoc dbase;
dbase.Configure(config_file, TRUE); gets dbase configuration options \to soundlab.cpp\
\includes reading phoneme positions \in phn\ at labelcl.cpp
NTimit39LabelClass::Open_Sym(const String & file_name);\

      conf.Open_File(config_file); \to resconf.cpp Where it opens\
                    file_ini.open(file_name,ios::in|ios::_Nocreate); \"res.ini" & "res.opt"\
      conf.Get_String_Opt("DBaseOptions", "ListOfSoundFNames", db_file); \resconf.cpp\
      Initialize(config_file,"DBaseOptions",read_transcription); \soundlab.cpp\ size 2048
      Inizialize_File_List(); \to soundlab.cpp "file_list" then vets_cardinality\
                                   soundfil.cpp reading 'headers" of sound files, 44

ini_options.Set_Options(config_file, dbase.Get_Num_Of_Symbols()); \iniopt.cpp symbol models\

symbol_models_initialization.Configure(ini_options, config_file, dbase.Get_Num_Of_Symbols());
                     \to initiali.cpp\ … feature.Configure(config_fname); \to feature.cpp\ …

symbol_models_initialization.Symb_Model_Calculation(ini_options.InitializedModelsFName,
ini_options.models_file_input, dbase, ini_options.full_covariance, ini_options.load_one_mixture,
ini_options.unif_sect, ini_options.model_type); \to initiali.cpp\ … where:

                          Write_Header_Of_File_Model (out_fname, dbase.Snd_Type(), dbase.Label_Type(),
      dbase.Db_File_List_Name(), dbase.Window_Lenght(), dbase.Window_Overlap(), stat_dim, full_cov);
                     tspecbas.cpp opens phonemes_1.spt, outputes the header and closes the file
      states_info.Initialize(num_sections_per_symbol[i], num_mix_per_symbol[i],stat_dim, full_cov);
                                    \initiali.cpp initialize (*this)[0-2][0] with zeros\
     Calculate_One_Mixture_Codebook(act_phon, num_frames, dbase, unif_sect); \at initiali.cpp\ in
                                                                                          what:

  not_end_of_dbase=dbase.Get_Filtered_Sequential_VetSmp_And_Its_Predecessors(vetsmp,act_phon,
  is_new_phone, pred_list); \at initiali.cpp Ln163 which goes to soundlab.cpp where:\

    while(not_end_of_dbase AND NOT
    SoundLabelledFile::Get_Filtered_Sequential_VetSmp_And_Its_Predecessors(vet,sym,
    is_new_fone, pred_list)) \to in the same soundlab.cpp, where:\
              act_smp=snd_file->Get_Actual_Position(); \to soundfil.cpp\ position=44 then
                           SetSymbol(sym, act_smp, new_smp); to labelcl.cpp, new_smp_pos
                                                          Set_Absolute_Position(new_smp);
                                                Backshift=numpred*(len_win_sample-overlap);
                     Set_bsolute_Positin(new_smp OR new_smp-backshift); \by soundfil.cpp and
                                                                       operation seekg();\
            Get_Sequential_Vet(vet); \to soundlab.cpp where \Read at soundfil.cpp\ and
               \Set_Relative_Position(-overlap) at soundfil.cpp by use of seekg()\;

  feature.Get_Previous_Info(pred_list, dbase.Smp_Rate()); \perform sequential
  transformations over prev_vetsmp_list to the memory of all the modules\
  feature.Extract(vet_features, vetsmp, dbase.Smp_Rate()); \retrieve configuration from
  configuration file apply all the required transformations\
  states_info[k][cluster].Do_Averages(); \utilize mean & cov accumulators to explicitly
  calculate.. for covariance, square of population, external product mean vector,
  main diagonal\
             States_info.Compute_Whole_Codebook_Clusters_Distortions(); \eigenvectors, eigenvalues\
                                                               states_info.Compute_Cluster_Weights();
  states_info.Store_Codebook(out_fname, act_phon, load_one_mixture, model_type); \model computation
                                                                                      and output\
```

This section described in short the internals of [Becchetti & Ricotti, 1999]. The basic initial information is the speech databases TIMIT and ATIS. All speech records are digitized and analyzed with help of DSP algorithms over the proper sliding windows. Multidimensional numerical vectors and their sequences are the observations, given as input to the HMM tools.

Modeling and Learning with Substring Data Mining

This section aims at introducing the elements of the Data Mining technique [Han & Kamber, 2006] in the form applicable to the ASR design. We mention the ordinary [Li et al, 2006], structural [Srivatsan & Sastry, 2006], sequential [Nizar & Ezeife, 2010] and finally the String [Ji & Bailey, 2007] Data Mining. We consider String Data Mining as an alternative to the HMM for ASR related applications.

Association Rule Mining. Consider a finite set $A = \{a_1, \dots, a_m\}, m \geq 1$. The elements of A we will also call items, due to example applications used in this area (supermarket transactions and market basket analysis). Further, let $\mathcal{D}: 2^A \rightarrow \mathbb{N}_0$ defines a multi-set over the power-set of A in a way that for each $X \in 2^A$ the number $\mathcal{D}(X)$ indicates how many times the subset (item-set) X occurs in multiset $\mathcal{D}$. We will refer to $\mathcal{D}$ as *the database* and to its elements as *transactions*. It is realistic to suppose the $\mathcal{D}$ to be finite in each time frame, although as a database it is a dynamically evolving relational table. We will use also the n-cube notation, where $\mathcal{D}$ induces natural item-set coding and their weights/labels to the n-cube vertices. In some cases we will consider and analyze only none zero occurrences/weights.

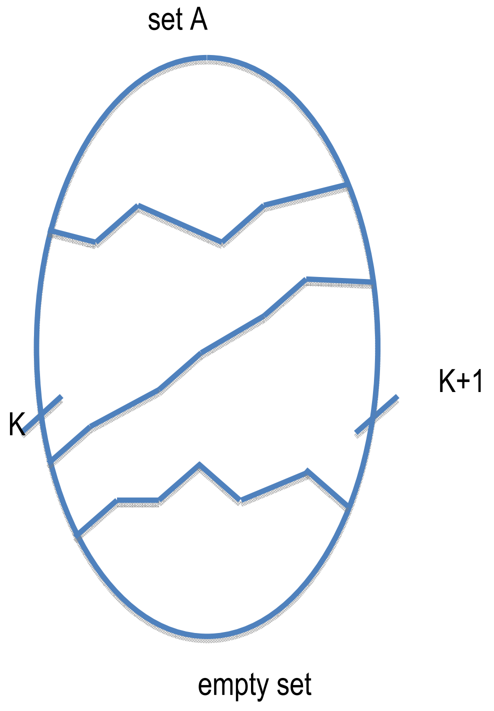


Figure 5. Hasse type diagram of power set of set A with isoareas by values of support function $s(X)$

For $X \subseteq A$ we define $s(X) \coloneqq \sum_{Y \supseteq X} \mathcal{D}(Y)$ to be the *support* of X in $\mathcal{D}$. We say that a subset $X \subseteq A$ is t*-frequent* in $\mathcal{D}$, $t \geq 0$, if $s(X) \geq t$, that is when there are at least t transactions in $\mathcal{D}$ containing X.

Monotonity is the important property of function $s(X)$,

$$(X \subseteq Y) \Rightarrow s(X) \geq s(Y).$$

Izo-areas of $s(X)$ can be presented by the sequences of embedded monotone Boolean functions. But such sequences can't behave arbitrarily. For example, if to consider a simple 2 row database then these rows correspond to 2 vertices of n-cube: v_1 and v_2. Consider subcubes $I(v_1, 1)$ and $I(v_2, 1)$ formed by these vectors and by the all 1 vector (set A). All points of these subcubes 1-support one of the rows, but the points in intersection support both rows. So it is not possible to design a database that gives homogeneous support value over the area covered by $I(v_1, 1)$ and $I(v_2, 1)$. So the mentioned "sequences of embedded monotone Boolean functions" are a specific inclusion-exclusion type object to be described and studied in deep.

Frequent subsets are the basis of the large research area of data mining. More precisely the association rule mining uses frequent subsets generating rules of some given confidence. The basic approach of association rule mining was formed inside the approach known as the APRIORI algorithm [Agrawal & Srikant, 1994], [Agrawal & Srikant, 1999]. An alternative approach is formulated in [Aslanyan & Sahakyan, 2009], [Aslanyan, 1976] using the well known technique of chain split [Hasel, 1966] and chain computations [Tonoyan, 1976], [Aslanyan & Khachatryan, 2008] of n-dimensional unite cube.

Today the real application problems that use the data mining technique deal with more complex information structures that the simple set of elements we considered so far. One typical problem is the trajectory analysis. Having the vertex set representing some real geographical points and considering movements of objects among these vertices we see that the valid trajectories are to be connected entities. Considering this in a graph we face the problem of mining connected sub-graphs [Rosen, 2010]. Another problem can consider only Sperner type subsets of a set trying to mine the frequent sets of this type. Interesting problems appear with sequences and its sub-sequences. The dominating part of research here supposes consideration og the so called embedded subsequences in manner of the known combinatorial LIS and LCS problems. Besides these mentioned structural diversity an attractive task is the algorithmic complexity issue. Bring an example. Consider the so called closed accessible set systems of the transactions database.

A subset $X \subseteq A$ is called *closed* (maximal subset) if for each $Y \subseteq A$, $Y \supset X$, it holds $s(Y) < s(X)$. For $X \subseteq A$ we define $\rho(X) \coloneqq \bigcap\{Y \subseteq A : Y \supseteq X \; and \; \mathcal{D}(Y) > 0\}$ to be the *closure* of X in $\mathcal{D}$. It can be checked that $X \subseteq \rho(X)$, $X \subseteq Y \Rightarrow \rho(X) \subseteq \rho(Y)$ and $\rho\big(\rho(X)\big) = \rho(X)$. It also can be checked that the closed subsets are the closures.

Obviously $\mathcal{D}: 2^A \to \mathbb{N}_0$ defines a monotonic decreasing integer function when X increases by set-inclusion, over the Boolean cube $(2^A, \subseteq)$, and it can be checked that the closed subsets are the upper k-points of $\mathcal{D}$, for $k \geq 0$. Also observe that the frequent subsets are all the subsets of the closed frequent subsets, and thereby the closed frequent subsets determine the frequent subsets. This technique is minimizing the mining constructions and computations.

Among the diverse problems of structural, sequential, time domain data mining there is a specific domain, known as string data mining (SDM) which is the one directly related to the speech recognition domain. There are plenty of algorithms dealing with sequential data mining. Each data structure with specific particularities of the applied problem inserts additional specifics in constructing the efficient mining algorithms. In speech recognition there appear vector sequences. Vectors are of fixed length and they have numerical coordinates (time/spectral domain coefficients). Particular coordinates and the vectors in whole are comparable – having a similarity/dissimilarity measure. One can imagine to cluster the vectors but the target is not the individual vector but their sequences, the sequence of utterances, moreover the important subsequences are of different length. We may imagine the Hasse diagram that interprets this situation. It is very simple.

The upper vertex (level 0) represents the entire sequence of length m, itself. In level 1, below the level 0, there are only 2 subsequences of length $m-1$. In k-th level there are k subsequences of length $m-k$. Graphically this can be represented as a triangular halve of a rectangular grid construction. Having the voce signal one have to create the vector subsequence counterpart, and scan all subsequences to accumulate their appearances on the grid vertices. Given a voice signal database we determine thresholds for each vector component (quantization, where rounding of coordinates by threshold values is applied) and accumulate subsequence repetitions into the grid points.

Our next postulation is about the validity of monotonity property on frequent subsequence area. This is easy to check and is very useful as an algorithmic construction of SDM. In global terms such algorithms identify all subsequences in groups by their initial vectors, then in each group the maximal frequent subsequence is sought. In real speech domain these frequent subsequences must obey additional requirements (properties). They will not involve one the other and they will not intersect in time axis. These are not absolute propositions but they are recommended and are almost mandatory.

In conclusion we bring the descriptions related to the technique we designed concerning the enhancements of the experimental speech recognizers.

String Mining. Let Σ denote a finite alphabet. A sequence S is an ordered list composed by letters from Σ. Denote the i-th item of a sequence S as $S[i]$. We will consider two types of structures:

subsequences and substrings. A sequence $S_1 = a_1a_2...a_m$ is called a subsequence of sequence $S_2 = a_1a_2...a_n$ if $m \leq n$ and there exist integers $1 \leq j_1 < j_2 <...< j_m \leq n$ such that $a_i = b_{j_i}$ for $1 \leq i \leq m$. We denote this relation as $S_1 \subseteq S_2$. In a similar way, if there exist an index $j, 1 \leq j \leq n - m + 1$ so that $a_i = b_{j+i-1}$ for $1 \leq i \leq m$, then sequence S_1 is called a substring of S_2 denoting this relation as $S_1 \sqsubseteq S_2$. We also express this relation as $S_1 = S_2[j, j + m - 1]$. In both cases there can be large number of subsequence and/or substring insertions within one general string. Subsequence and substring data mining algorithms in an initial stage discover frequent subsequence and substring insertions correspondingly. Sequence mining is known with its application in market basket type data analysis. String mining, as it is easy to understand, can be an important tool of speech signal analysis.

In this application the alphabet Σ, as it was described in previous section, consists of signal window characteristic-observation vector quantization. Let us insert, in addition, a concept of similarities of letters of Σ. We do not justify it but denote it as $\mathrm{d}(\mathrm{a}_{\mathrm{i},}\mathrm{a}_{\mathrm{j}})$, $\mathrm{a}_{\mathrm{i},}\mathrm{a}_{\mathrm{j}} \in \Sigma$. Given a threshold ε we may identify $\mathrm{a}_{\mathrm{i},}\mathrm{a}_{\mathrm{j}}$ when $\mathrm{d}(\mathrm{a}_{\mathrm{i},}\mathrm{a}_{\mathrm{j}}) \leq \varepsilon$. In this way we come to the approximate subsequence/substring mining concept.

Given a sequence database SDB and a minimum support threshold α, a string Q is frequent substring pattern of SDB if it holds

$$\frac{|\mathrm{S} \in \mathrm{SDB}|\mathrm{Q} \sqsubseteq \mathrm{S}|}{|\mathrm{SDB}|} \geq \alpha.$$

In this case it is used that SDB lists strings of our interest – words, phoneme code-words, etc. The same formula is applied in a case when SDB presents a long speech signals recording or a set of them. The variety is expressed by the use of the term “episode”. Frequent episodes are like a clustering model while the substring mining tends to the supervised learning. Technically substring mining is based on suffix tree models, well developed, and importantly linear in time and space complexities. And indeed still there are particular problems to understand and manage with this type of algorithms. Consider a couple of scenarios.

Let we are given a speech corpus scanned and computed the window based observations. Frequent substring mining in this case that gives a limited number of episodes is an equivalent to a cluster analysis procedure. Cluster analysis can not be applied directly due to different length of target subsequences. And the episodes derived can be used as the initial basis in manual notification of phonemes.

As the second scenario consider the case when a phoneme database is given and when it is to be enlarged by the use of speech corpus analysis. Then how the frequent episodes and the existing base can be combined? This seems like a supervised substring mining procedure. Here one possible approach may apply the well known parametric optimization of voting estimation algorithms by Yu. Zhuravlev.

And of course the main case is the phoneme search and partitions by phonemes that is based on suffix tree constructions and their extensions. Just mention two extensions of these types [Fischer et al, 2005] inserts minfreq and maxfreq in constraint-based frequent string mining computing all strings that are frequent in one database and infrequent in another. The technique used is the suffix tree and the longest common prefix (LCP) array constructions. [Tsuboi, 2003] proposed a divide-and-conquer type algorithm that decomposes the mining task into a set of smaller tasks by using a ternary partitioning technique. It brings to memory minimization and to the computation reduces.

Next group of technique we bring is about classification, recognition. An evident strategy is the use of search by phonemes (observation vector sequences) that can be effectively implemented by the suffix tree based algorithms. The necessary extension may address the use of distances and similarities, the idea of scaling over the time domain, and the none-intersection (separation) requirement of the phonemes. In addition, [Chan et al, 2003] introduces the emerging substring concept that aims at mining data classes, substrings, which occurs more frequently in one particular class rather than in other classes. In this model, above the support threshold, a growth rate threshold is determined. And again the technique is the prefix suffix trees and their transformations.

String Recognition. We use the special case of general data mining, in our case the association rule mining technique that tries to generate if-then type rules with a property of having satisfactory support and confidence. The rule has left hand attributes with their constraints and a similar set of right hand attributes with constraints. Previously, the same construction appeared in relational databases in part of functional dependency generation. Even that period, in several applications, appeared an interest in generating rules and dependencies with only one right hand attribute [Armstrong et al, 1998]. Now, [Liu et al, 1998] use this scheme in associated rule mining with one right side attribute rules. This attribute corresponds to the class label. Parallel frequency determination for the part of observation attributes and then for the complete attribute set – the class label included, forms a proper base for class associated rule mining. This can correspond to the phoneme classification rules by the use of window based observation vectors and their analytics.

Conclusion

Having that the recognition rate is quite low in HMM based ASR systems it is to try to find out the real bottleneck of the problem. HMM analysis shows that even being well defined and an attractive technique for applications it raises question with their implementation in ASR system design. Even the Artificial Neural Network (ANN) can be a good alternative. Due to ANN is not well interpretable the SDM technique might be a better choice. For experimentation on this it is necessary to set up an open source research HS environment with necessary learning databases, with DSP and interfaces. An open source prototype is selected. The additional models and components to be inserted into these environment were sought and determined as SDM (and ANN potentially).

Bibliography

[Agrawal & Srikant, 1994] R. Agrawal, R. Srikant, Fast algorithms for mining association rules in large databases. In Proc. 20th Int. Conf. on Very Large Data Bases, 1994, pp. 487–499.

[Agrawal & Srikant, 1999] R. Agrawal, R. Srikant, Mining sequential patterns. In Proc. 11th Int. Conf. on Data Engineering, 1995, Washington, DC, IEEE Comput. Soc.

[Armstrong et al, 1998] T. Armstrong, K. Marriott, P. Schachte, H. Sondergaard, Two classes of Boolean functions for dependency analysis, Selected Papers of the First International Static Analysis Symposium, Science of Computer Programming, vol. 32, issue 1, 1998, pp. 3 - 45.

[Aslanyan, 1976] L. Aslanyan. Isoperimetry problem and related extremal problems of discrete spaces, Problemy Kibernetiki, 36, 1976, pp. 85-126.

[Aslanyan & Khachatryan, 2008] L. Aslanyan and R. Khachatryan. Association rule mining inforced by the chain decomposition of an n-cube, Mathematical Problems of Computer Science, XXX, 2008, ISSN 0131-4645.

[Aslanyan & Sahakyan, 2009] L. Aslanyan, H. Sahakyan, Chain Split and Computations in Practical Rule Mining, International Book Series, Information Science and Computing, Book 8, Classification, forecasting, Data Mining, Sofia, Bulgaria, 2009, pp. 132 - 135.

[Becchetti & Ricotti, 1999] C. Becchetti, K. P. Ricotti, Speech Recognition: Theory and C++ Implementation, Wiley, the University of Michigan, 1999, 428p.

[Chan et al, 2003] Sarah Chan, Ben Kao, Chi Lap Yip, Michael Tang, Mining Emerging Substrings, Eighth International Conference on Database Systems for Advanced Applications, 26-28 March 2003, Proceedings, pp. 119-126.

[Fischer et al, 2005] J. Fischer, V. Heun, S. Kramer, Fast Frequent String Mining Using Suffix Arrays, Proceedings of the 5th IEEE International Conference on Data Mining, IEEE Computer Society, 2005.

[Han & Kamber, 2006] J. Han and M. Kamber. Data Mining: Concepts and Techniques, 2nd ed. The Morgan Kaufmann Series in Data Management Systems, ISBN 1-55860-901-6, 2006, 743p.

[Hasel, 1966] G. Hansel. Sur le nombre des functions booleennes monotones de n variables, C.R. Acad. Sci. Paris, 262, serie A (1966), 1088.

[Ji & Bailey, 2007] X. Ji, J. Bailey, An efficient technique for mining approximately frequent substring patterns, Seventh IEEE International Conference on Data Mining – Workshops, 2007, pp. 325-330.

[Li et al, 2006] H. F. Li, S. Y. Lee, and M. K. Shan, “DSM-PLW: single-pass mining of path traversal patterns over streaming web click-sequences”, Proc. of Computer Networks on Web Dynamics, pp. 1474–1487, 2006.

[Liu et al, 1998] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. In KDD’98, New York, NY, Aug. 1998, pp. 80 - 86.

[Nizar & Ezeife, 2010] M. Nizar R., and C. I. Ezeife, A taxonomy of sequential pattern mining algorithms, ACM Computing Surveys (CSUR) 43.1, 2010: 3.

[Rabiner, 1989] L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, 1989, pp. 257-286

[Rosen, 2010] Kenneth H. Rosen editor, Handbook of Discrete and Combinatorial Mathematics, 2010, 1248 p.

[Srivatsan & Sastry, 2006] L. Srivatsan, and P. Shanti Sastry, A survey of temporal data mining, Sadhana 31.2, 2006, pp. 173-198.

[Tonoyan, 1976] G. P. Tonoyan. Chain decomposition of n dimensional unit cube and reconstruction of monotone Boolean functions, JVM&F, v. 19, No. 6, 1976, pp. 1532-1542.

[Tsuboi, 2003] Yuta Tsuboi, Mining Frequent Substring Patterns with Ternary Partitioning, IBM Research Report, Tokyo Research Laboratory, Sept. 2003, 8p.

Authors' Information

Levon Aslanyan *– Principal Researcher, Discrete Mathematics Department, Institute for Informatics and Automation Problems, NAS RA; e-mail: lasl@sci.am*

Major Fields of Scientific Research: Discrete optimization, Pattern recognition.

Minoosh Heidari *– PhD student, Discrete Mathematics Department, Institute for Informatics and Automation Problems, NAS RA; e-mail: meinoosh@gmail.com*

Major Fields of Scientific Research: HMM, NLP, Software engineering.

ON ATTACK GRAPH MODEL OF NETWORK SECURITY

Hasmik Sahakyan, Daryoush Alipour

***Abstract**: All types of network systems are subject to computer attacks. The overall security of a network cannot be determined by simply considering the vulnerable points in the network; it is essential to realize how vulnerabilities can be combined in the same host or in a set of connected hosts to initiate an attack. Attack graph is a tool for modeling compositions of vulnerabilities and thus representing possible multi-stage multi-host attacks in networks. Attack graphs can be used for measuring network security; supporting security solutions by identifying vulnerabilities that should be removed such that no attack can be realized targeting given critical resources, and thus hardening the network. We consider a general model of attack graphs and a scheme of attack graph generating algorithm; and investigate graph-theoretical problems related to particular tasks of network hardening.*

***Keywords**: Attack graph model; Network security*

***ACM Classification Keywords**: C.2 Computer-communication networks; G2.2 Graph Theory; G2.3 Applications*

1. Introduction

All types of network systems are subject to computer attacks. The overall security of a network cannot be determined by simply considering the vulnerable points in the network. To evaluate the network security, it is essential to understand how vulnerabilities can be combined in the same host or in a series of connected hosts to initiate attacks. Attack graph is a tool for modelling compositions of vulnerabilities and thus enumerating multi-stage multi-host attacks in networks (see e.g. [Aslanyan et al, 2013; Barik et al, 2014; Noel et al, 2010; Sheyner et al, 2002; Zhang et al, 2009]). Generally, attack graphs are large and complex, and automatic and efficient generation of attack graphs is an important issue. It is also important to analyze attack graphs for measuring network security, supporting security solutions by identifying vulnerabilities that should be removed such that none of the attack paths leading to a given critical resource can be realized, and thus by hardening the network. In this paper we address graph-theoretical problems related to particular tasks in the network security.

The paper is organized as follows. A brief overview of network security is given in Section 2 below. Section 3 introduces network vulnerability model and the role of attack graphs. A simple algorithm of generating attack graph is described for an attack graph model. Section 4 is devoted to graph-theoretical problems and algorithms related to particular network hardening tasks.

2. Network Security

A *computer network* is a group of computer systems and other computing hardware devices that are linked together through communication channels enabling communication, data exchange and resource sharing between users (Figure 1).

Network security refers to protection of resources. The resources to be protected include:

- All types of information resources (user-generated data, programs, computer services and processes);
- Communication infrastructure (communications devices, transmission paths, communication data);
- Computer system (hardware software operability).

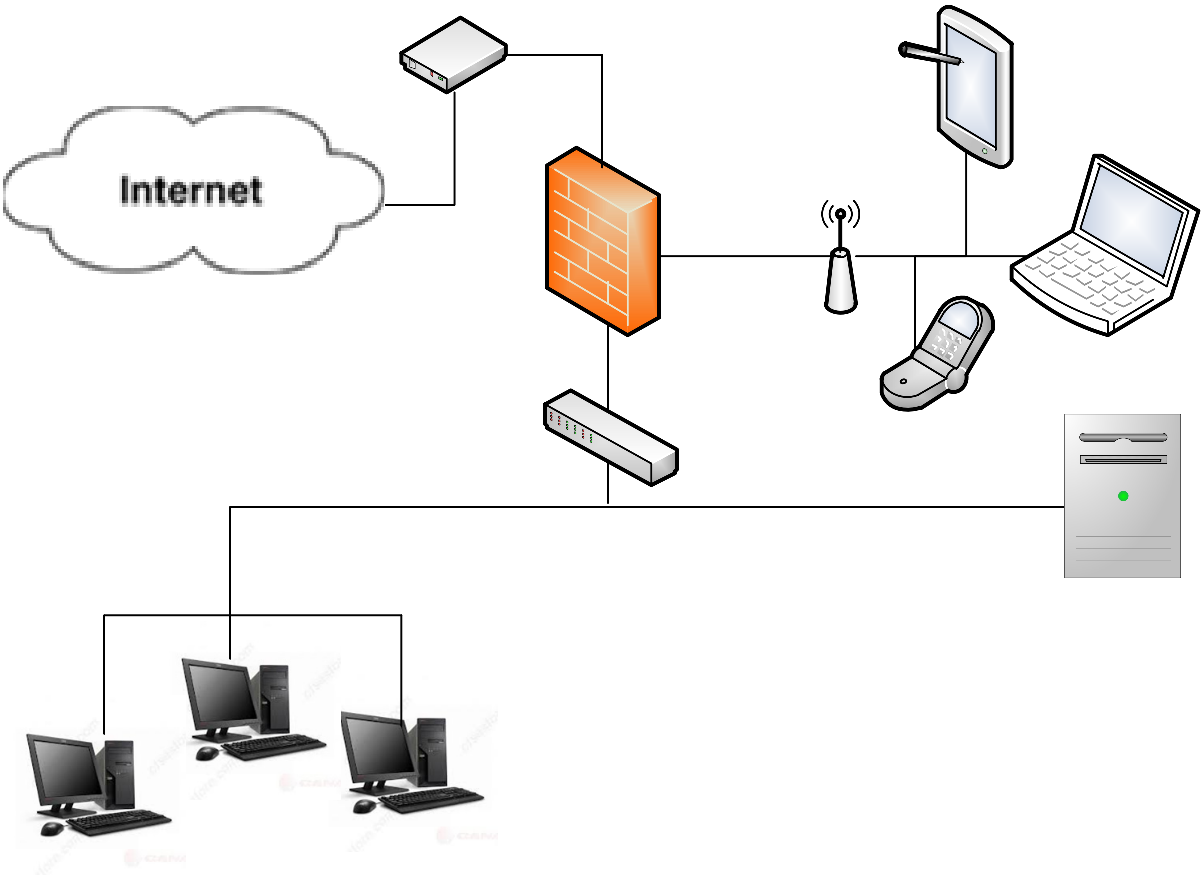


Figure 1. Computer network in an example

In general, three main components of *network security* are of basic interest:

- Confidentiality: the information must be protected from disclosure to unauthorized parties;
- Integrity: the information must be protected from being modified by unauthorized parties;
- Availability: the information must be available when it is needed and access right is in place, - must be protected against unauthorized deletion/modification of data or causing a denial of service of data.

The base elements of network attacks are threats and vulnerabilities.

Threats are actions, events, or circumstances that have the potential to compromise network security, for example, to obtain unauthorized access to data.

Vulnerability is a flaw or weakness in a network that can be exploited by one or more threats to gain access to a system or network.

For example, the vulnerability can allow an attacker:

- To execute commands as another user;
- To access data that has access restrictions;
- To conduct a denial of service action, etc.

Vulnerabilities can be exploited by human or another (technical) system to initiate an *attack*. *Network attack* is an intrusion on the network infrastructure. First it can collect and analyze information in order to exploit the existing vulnerabilities. In such cases the purpose is only to get some information from the system, - these are *passive attacks*. *Active attacks* occur when resources or data are modified, disabled or destroyed.

Examples of network attacks are:

DoS (Denial-of-Service) – Send more requests to the computer than it can handle.

Unauthorized access – Access some resource that the computer should not provide the attacker. For example, a host might be a web server and should provide anyone with requested web pages. However the host should not provide command shell access to a person who should not get it.

There are some standards to classify vulnerabilities (see e.g. [CVE], [NVD], [OSVD]). Common *Vulnerabilities and Exposures* (CVE®) is a dictionary of common names for publicly known information security vulnerabilities. The CVE vulnerabilities have three parts. Their styles are in the following format: CVE-year-vulnerability number. For example:

Name: *CVE-1999-0012*

Description: Some web servers under Microsoft Windows allow remote attackers to bypass access restrictions for files with long file names.

Common Vulnerability Scoring System (CVSS) is a standard designed to convey vulnerability severity and help determine urgency and priority of response ([CVSS]). CVSS has been adopted by a number of vulnerability database providers. CVSS consists of 3 groups: Base, Temporal and Environmental. Each group produces a numeric score ranging from 0 to 10, and a Vector, a compressed textual representation that reflects the values used to derive the score. The Base group represents the intrinsic qualities of vulnerability. The Temporal group reflects the characteristics of vulnerability that change over time. The Environmental group represents the characteristics of vulnerability that are unique to any user's environment. CVSS base score consists of *exploitability metrics* and *impact metrics*. In the exploitability metrics there are three metrics for *Access Vector* (AV), *Attack Complexity* (AC) and *Authentication* (Au). The exploitability metrics measure characteristics of the vulnerability that affect the difficulty of exploitation of the vulnerability. The impact metric contains the following components: Confidentiality Impact (C), Integrity Impact (I), and Availability Impact (A). The impact metric measures how vulnerability, if exploited, will directly affect an IT asset, where the impacts are independently defined as the degree of loss of confidentiality, integrity, and availability. For example, vulnerability could cause a partial loss of integrity and availability, but no loss of confidentiality.

Exploitability Metrics

Access Vector (AV)		
Local (AV:L)	Adjacent Network (AV:A)	Network (AV:N)
Access Complexity (AC)		
High (AC:H)	Medium (AC:M)	Low (AC:L)
Authentication (Au)		
Multiple (Au:M)	Single (Au:S)	None (Au:N)

Impact Metrics

Confidentiality Impact (C)		
None (C:N)	Partial (C:P)	Complete (C:C)
Integrity Impact (I)		
None (I:N)	Partial (I:P)	Complete (I:C)
Availability Impact (A)		
None (A:N)	Partial (A:P)	Complete (A:C)

Scoring equations and algorithms for the base, temporal and environmental metric groups are also described.

Consider the following example. Let assume that in one computer of the network “Microsoft Office 2010” is installed.

Related to “Microsoft Office 2010” CVE includes the following vulnerability:

Name: CVE-2015-1649

Description: Use-after-free vulnerability in Microsoft Word 2007 SP3, Office 2010 SP2, Word 2010 SP2, Word Viewer, Office Compatibility Pack SP3, Word Automation Services on SharePoint Server 2010 SP2, and Office Web Apps Server 2010 SP2 allows remote attackers to execute arbitrary code via a crafted Office document, aka "Microsoft Office Component Use After Free Vulnerability."

Consider the corresponding CVSS vector for CVE-2015-1649.

Exploitability Metrics

Access Vector (AV)		
Local (AV:L)	Adjacent Network (AV:A)	[✓ Network (AV:N)]
Access Complexity (AC)		
High (AC:H)	[✓ Medium (AC:M)]	Low (AC:L)
Authentication (Au)		
Multiple (Au:M)	Single (Au:S)	[✓ None (Au:N)]

Impact Metrics

Confidentiality Impact (C)		
None (C:N)	Partial (C:P)	[✓ Complete (C:C)]
Integrity Impact (I)		
None (I:N)	Partial (I:P)	[✓ Complete (I:C)]
Availability Impact (A)		
None (A:N)	Partial (A:P)	[✓ Complete (A:C)]

In the part of Exploitability Metrics

1. Access Vector is "Network", since CVE-2015-1649 can be exploited remotely;
2. Access Complexity is not "High" because this vulnerability is not exploitable at the attacker's whim, and it is not low because some additional access or specialized circumstances need to exist for the exploit to be successful;
3. Authentication is "None" because the attacker does not need to authenticate to any additional system.

In the part of Impact Metrics

> If an administrative user were to run the virus scan, causing the buffer overflow, then a full system compromise would be possible. Since the most harmful case must be considered, each of the three Impact metrics is set to "Complete" because of the possibility of a complete system compromise.

Thus, CVSS base score for CVE-2015-1649 is: (AV: N/AC:M/Au:N/C:C/I:C/A:C).

There are known software /vulnerability scanners/ designed to scan computers and networks for vulnerabilities and then report about the identified vulnerabilities. Network-based vulnerability scanners, such as Port Scanners (Nmap, Nessus), Web application security scanner, Network vulnerability scanner (BoomScan) - are installed on a computer that scans a number of other hosts on the network. Host-based scanners, such as Database Security Scanner, - are installed in the host.

However, to achieve the attack goals attackers may need to use not only separate vulnerabilities but also combinations of vulnerabilities, i.e. they can attack a vulnerable computer and then use it for further attack goal. Thus, the overall security of the network cannot be determined by simply counting the vulnerabilities, and an important task in network security is to analyze which vulnerabilities are acceptable risks; how particular vulnerabilities or exploits can be combined and exploited in complex attacks; and to support security solution.

One approach for modeling how particular vulnerabilities can be combined for an attack that is our interest in this article - is the model of attack graphs.

3. Network Vulnerability Model and Attack Graphs

Generally, to efficiently evaluate security of a network system, it is necessary to develop a network vulnerability model that illustrates the security risk properties of the system.

For composing *network vulnerability model* it is necessary to know network configuration (hosts, operating systems, application programs, network services, etc.); network connectivity, including the connectivity-limiting effects of devices such as firewalls and router access control lists. Then it should be identified vulnerabilities and interdependency between them.

Thus the model will have the following components:

- Hosts;
- Services in every host;
- Vulnerabilities of every service;
- Connectivity between services/vulnerabilities;
- Possible attacks.

Let $H = \{h_1, \cdots, h_n\}$ denote the set of *hosts* in a network that can potentially be targeted by an attacker.

Let $V_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,m_i}\}$ denote the set of vulnerable services running at host $h_\mathrm{i} \in H$, $i = 1,2, \cdots, n$; we suppose that vulnerabilities descriptions are known (for example, from CVE, CVSS).

Let $C_{i,j} = \{c_{i,j,1}, c_{i,j,2}, \cdots, c_{i,j,m_j}\}$ denote the set of connectivity relations from the host h_i to the vulnerable services running at host h_j, $i = 1,2, \cdots, n, j = 1,2, \cdots, n, i \neq j$

$$c_{i,j,k} = \begin{cases} 1, if\ there\ is\ a\ connection\ from\ host\ h_i\ to\ v_{j,k} \\ \qquad\qquad 0, \qquad otherwise \end{cases}$$

Let A denote the set of possible attacks.

Generally, an attack $a \in A$ can be initiated if some certain network conditions exist (for example, a service running in the destination host can be accessed from a source host) and/or an attacker has certain privilege on certain hosts (for example, attacker has *user* privilege on source host). These are *attack preconditions*. Successful execution of an attack may create new attacker privilege or new network conditions. These are *postconditions* of the attack. Let a_{pre} and a_{post} denote the sets of preconditions and postconditions of the attack a, respectively.

Thus, each attack a can be given by the following elements:

- Source host h_{src} from where a is launched;
- Target host h_{dest};
- Target vulnerability v that exist at h_{dest};
- Set a_{pre} of attack preconditions that enable to attack the vulnerability v from the host $h_i \in H$;
- Set a_{post} of attack postconditions on host h_{dest} obtained after successfully attacking target vulnerability v.

To achieve the attack goals attackers may need to use not only separate vulnerabilities but also combinations of vulnerabilities. However, the vulnerability scanners cannot directly identify the complex attack routes on the network. The attack postconditions of an attack $a \in A$ initiated from the source host h_{src} to the destination host h_{dest}, can be the attack precondition for another attack $a' \in A$ from the host h_{dest}. By knowing the characteristics of vulnerabilities, the preconditions required exploiting them, and the postcondition of exploiting them, it becomes possible to chain possible simple/atomic attacks $a \in A$ together into a sequence of attacks that achieve a certain goal. This is the information that attack graphs represent.

Thus, *attack graph* is a tool for enumerating multi-stage multi-host attacks in networks. Without this tool it is very difficult to manually discover how an attacker can combine vulnerabilities in the same host or in connected hosts to compromise critical resources. The task becomes more difficult as the number of vulnerabilities as well as the size of network increases. Attack graphs can be used for measuring network security, supporting security solutions by identifying vulnerabilities that should be removed such that none of the attack paths leading to a given critical resource can be realized, and thus by hardening the network. The low cost of removing the vulnerabilities is also important.

There are different types of attack graphs, and different algorithms for generating them. We will address a general model of attack graphs based on exploit dependency attack graph model.

Formally, attack graph can be represented as a directed bipartite graph $G = (V_1 \cup V_2, E).$ Nodes in V_1 correspond to either *attacker privilege* or *network conditions*. Nodes in V_2 correspond to *Attacks/Exploits.* Directed edges from V_1 to V_2 are preconditions of attacks/exploits. Directed edges from V_2 to V_1 are postconditions of executing exploits/attacks.

Consider nodes in V_1. An attacker can have certain privilege on certain hosts. For example, the attacker can have *user* privilege on host "h". Then, there will be a corresponding node in V_1: *user("h")*. Certain services running at the destination host can be accessed from a source host. There will be corresponding nodes in V_1, for example, if *ftp* service is running, then the node *ftp("h1","h2")* will refer that *ftp* service running at "*h2"* is accessible from "*h1"*.

To execute attack attackers may need multiple network conditions (preconditions of the attack). Successful execution of an exploit may create new attacker privilege or new network conditions (postconditions of the attack). Nodes in V_2 can be of form: *Exploit("h1","h2")* or *Exploit("h1")* - meaning that having some privilege on "*h1"*, an attacker can perform the exploit *Exploit at "h2"*; or the *Exploit* can be performed locally at "*h1".*

Consider a simple example: the network consisting of *host1*, *host2*, *host3*. Let in *host1* and *host2* "*Office Web Apps Server*"[1] has been installed. It is mentioned in CVE database that this software has a vulnerability CVE-2015-1649, which allows remote attackers to execute arbitrary code via a crafted Office document, aka "Microsoft Office Component Use after Free Vulnerability." Therefore in our assumed network, attacker using this vulnerability can gain *user* privilege on the host running the service "*Office Web Apps Server*". According to sources of CVE, this kind of vulnerability is called Use-after-free, that means that when a user execute a weak software then an error occurs, and the pointer is immediately freed ([CWE]). However, this pointer is later incorrectly used in the other function.

Attacker uses a *buffer overflow* or *use-after-free* kinds of *memory corruption* errors to overwrite control-data and control flow of the program finally.

[1] Office Web Apps Server is a new Office server product that delivers browser-based versions of Word, PowerPoint, Excel, and OneNote. A single Office Web Apps Server farm can support users who access Office files through SharePoint 2013, Lync Server 2013, Exchange Server 2013, shared folders, and websites.[https://technet.microsoft.com/en-us/library/jj219437.aspx]

Assume that "*Office Web Apps Server*" service at host2 can be accessed from both host3 and host1, and "*Office Web Apps Server*" service at host1 can be accessed from host2 only. The attacker has initially *user* privilege on host3. Figure 2 demonstrates the network.

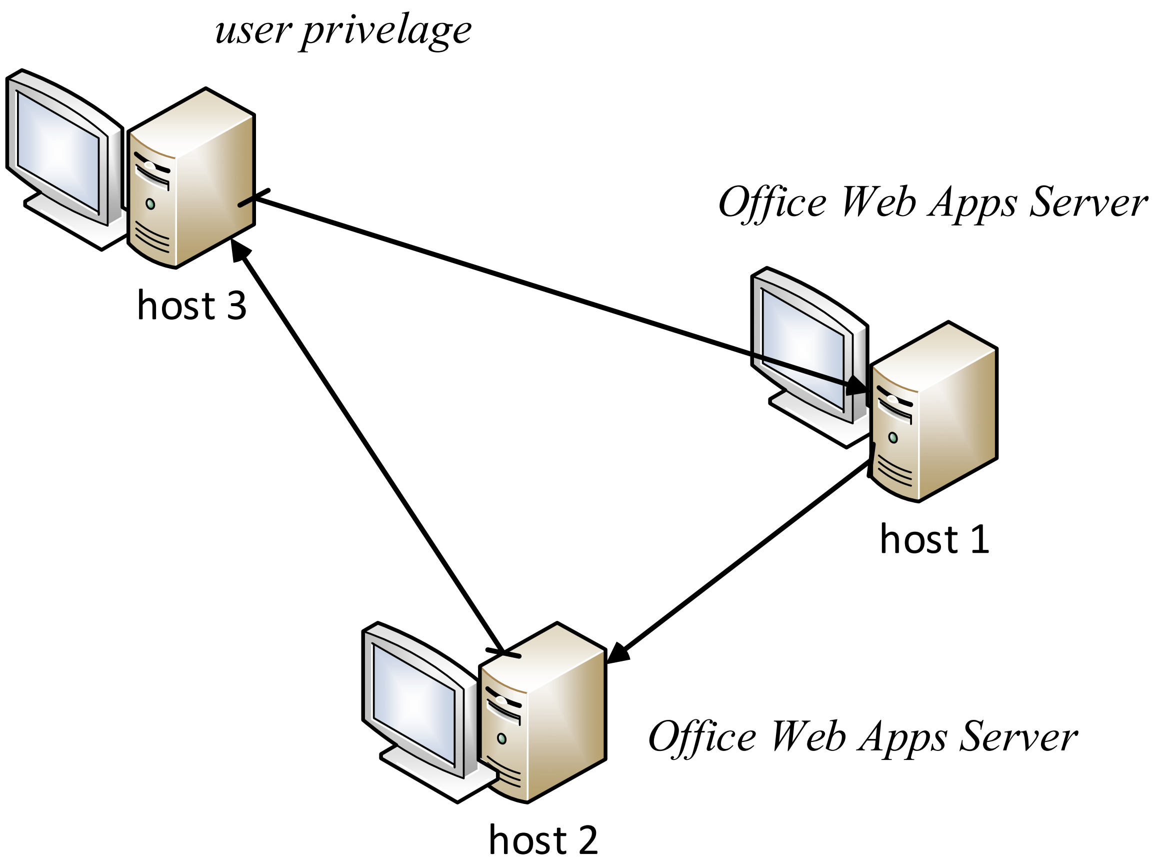


Figure 2. An example network to construct the Attack Graph

Now we construct the corresponding attack graph. Initially, the *precondition* nodes (nodes in V_1) are the following: *user(3)* (attacker privilege) and *OfficeWebAppsServer(3,1)*, *OfficeWebAppsServer(3,2)*, *OfficeWebAppsServer(1,2)* (network conditions).

user(3) and *OfficeWebAppsServer(3,1)* make it possible the exploit: *use-after-free (3,1).* Similarly, *user(3)* and *OfficeWebAppsServer(3,2)* make it possible the exploit: *use-after-free(3,2).* Thus, the graph should have *use-after-free(3,1)* and *use-after-free(3,2) exploit/attack* nodes (nodes in V_2).

This part of attack graph is shown in Figure 3. Oval-nodes correspond to nodes in V_1; and the rectangle-nodes correspond to nodes in V_2.

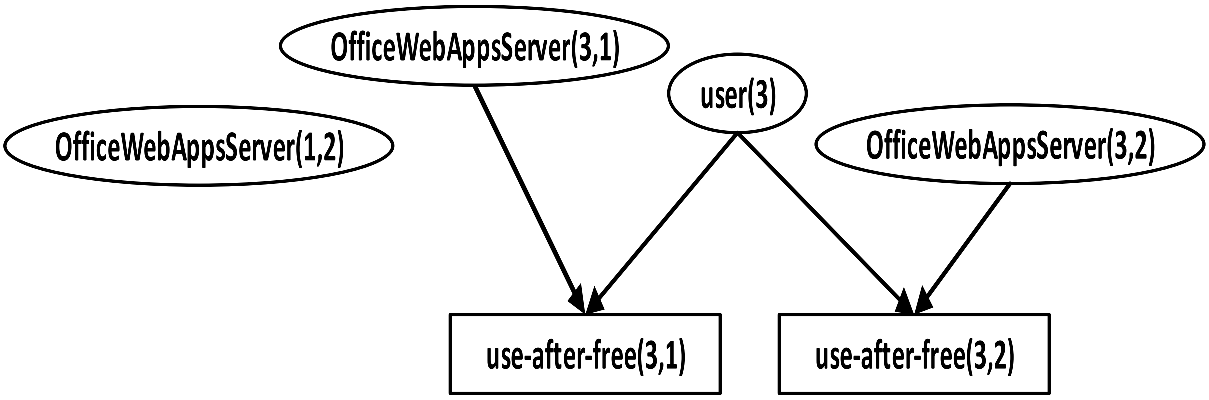


Figure 3. Part of Attack Graph

Exploiting the *use-after-free(3,1),* an attacker obtains *user* privilege on host1 *and* exploiting the use-after-free*(3,2)* an attacker obtains *user* privilege on host2 (*postcondition nodes user(1) and user(2))* (Figure 4).

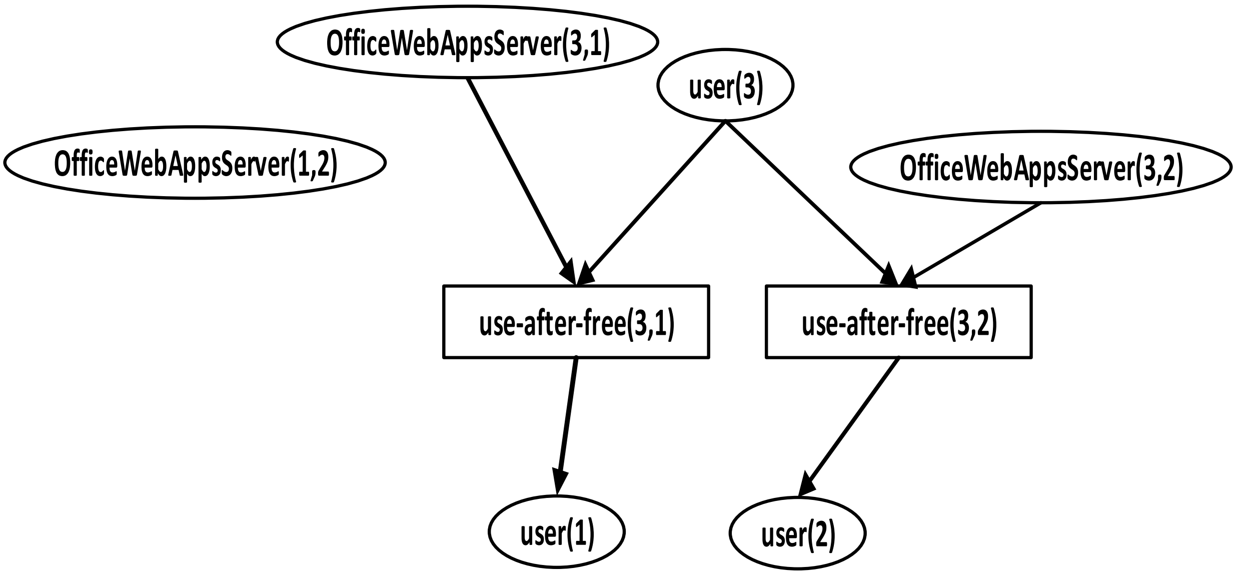


Figure 4. Part of Attack Graph in the next step.

user(1) and OfficeWebAppsServer(1,2) make it possible the exploit use-after-free(1,2), which in its turn give the attacker user privilege on host2. Figure 5 demonstrates the whole graph.

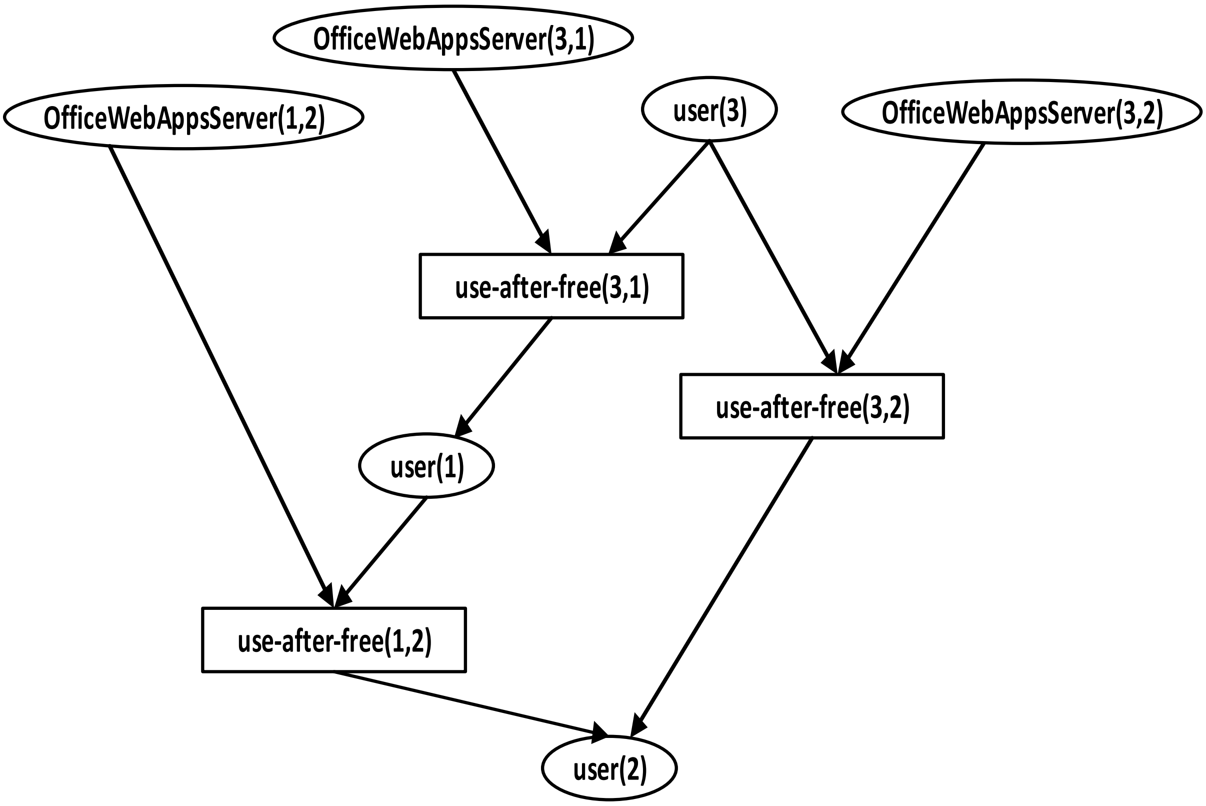


Figure 5. An Attack Graph example

Indeed the real practical networks are quite large and an automated Attach Graph generation is an objective. It is to be composed by the collected descriptions of the network and it is supposed that the data collection is also automated. Let us formulate our Attack Graph generation algorithm.

Algorithm A_G

Input of the algorithm A_G.

Let $H = \{h_1, \cdots, h_n\}$ denote the set of *hosts* in a network that can potentially be targeted by an attacker. Let $V_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,m_i}\}$ denote the set of vulnerable services running at host $h_\mathrm{i} \in H$, $i = 1,2, \cdots, n$. We suppose that vulnerabilities descriptions are known (for example, from CVE, CVSS). Let $C_{i,j} = \{c_{i,j,1}, c_{i,j,2}, \cdots, c_{i,j,m_j}\}$ denote the set of connectivity relations from the host h_i to the vulnerable services running at host h_j, $i = 1,2, \cdots, n$, $j = 1,2, \cdots, n$, $i \neq j$.

Input of the algorithm A_G

H, V_i for $i = 1,2,\cdots,n$, and $C_{i,j}$ for $i = 1,2,\cdots,n$, $j = 1,2,\cdots,n$

Output of the algorithm A_G is the corresponding attack graph.

```
begin
source_H=dest_H=H;
      while (source_H is not empty)
      {
      source:=take_host_from(source_H);
      if user(source) then
      {
      source_H=current(source_H,source)
      add_node_vuln_type_with_label("user(source)");
      dest_H=current(dest_H,source);
      while (dest_H is not empty)
      {
       dest= take_host_from(dest_H);
       dest_H=current(dest_H,dest);
       while (V_dest is not empty)
       {
        vuln=take_vuln_from(V_dest);
        V_dest=current(V_dest,vuln);
        if (connect(source,dest,vuln)) then
        {
         add_node_vuln_type_with_label(vuln);
         Precond = add(Precond)
         if (find_attack(Precond)) then
            {attack(attack_precond, attack_postcond); add_node_attack_type_with_label(attack_name);
            add_Connection(attack_precond, attack_name); Postcond=attack_postcond;
            add_node_vuln_type_with_label(Postcond); add_Connection(attack_name,attack_Postcond);
            Precond = add(Postcond }
            }
         }
        }
       }
      }
end
```

Let us explain the terms and definitions used:

take_host_from(H) – returns the current host from the set H;

user(h) – returns 1 if attacker has “user” privilege on host “h”, and 0 – otherwise;

current(H,h) – returns the set H\{h};

add_node_vuln_type_with_label(l) – adds a node of type “vulnerability” with the label “l”;

add_node_attack_type_with_label(l) – adds a node of type “attack” with the label “l”;

take_vuln_from(V) – returns the current vulnerable service name from the set V;

connect(h1,h2,v) – returns 1 if the vulnerability “v” at the host “h2” can be accessed from the host “h1”;

P=add(p) – adds into P all vuln_type node labels, created during the current cycle;

find_attack(P) – returns 1 if there is a known attack, which has preconditions that are in P;

attack(name,P1,P2) – returns the attack name, the set of attack preconditions and the set attack postconditions;

add_Connection(F,T) - adds connection from all elements of the set F to the all elements of the set T.

4. Graph-theoretical tasks

In this section we consider graph-theoretical local actions related to the tasks of network hardening.

Let us start with definitions.

Definition: A *directed graph* is a graph, where the edges have a direction associated with them. In formal terms, a directed graph is a pair $D = (V, A)$, where V is the set of vertices, and A is a set of ordered pairs of vertices, called arcs, or directed edges.

An ark $a = (v_1, v_2)$, is considered to be directed from v_1 to v_2. v_2 is called the *head* and v_1 is called the *tail* of the arc.

For a vertex v, the number of head endpoints adjacent to v is called the *in-degree* of the vertex and the number of tail endpoints adjacent to v is its *out-degree*. The in-degree of v is denoted as $deg^-(v)$, and the out-degree as $deg^+(v)$.

Definition: A directed graph $D = (V, A)$ is called a directed bipartite graph if there exists a partition of the vertex set: $V = V_1 \cup V_2$ such that $D[V_1]$ and $D[V_2]$, the two induced directed subgraphs of D, contain no arcs of A (Undefined terms can be found in [Jorgen et al, 2007]).

Tasks related to the vertex degrees in attack graph.

Let $D = (V_1 \cup V_2, A)$ is a directed bipartite graph corresponding to attack graph model, described in the previous section. V_1 consists of *attacker privilege* or *network condition* type nodes, and V_2 consists of *Attack* type nodes.

If a vertex $v_1 \in V_1$ has large number of tail endpoints, then network conditions corresponding to v_1 may allow large number of possible attacks.

If a vertex $v_2 \in V_2$ has small number of head endpoints, then the corresponding attack/exploit has small number of preconditions, and thus, is easy to execute. If v_2 has large number of tail endpoints, then the execution of the corresponding to v_2 attack will create large number of conditions for further attacks. Removal of the corresponding services will reduce the number of possible attacks. Thus, the vertex degree is a managing parameter to be computed.

Thus we formulate:

Task1. Find all vertices in V_1, which have out-degree greater than the given threshold.

Find all vertices in V_2 which have in-degrees less than the given threshold.

These are easy tasks, and simple algorithms can be used (complexity is $|V_1| \times |V_2|$).

Similar task is to find subsets of V_1, such that the summary out-degrees is greater than the given threshold.

Tasks related to the covering problems.

One of the main tasks in network hardening is to identify minimal number of vulnerable services that should be removed such that no attack will be possible.

If we pay no attention to the fact that vertices of V_1 are not homogeneous (meaning that the corresponding vulnerabilities can be created step by step, as results of attack exploits), then we deal with D as undirected graph, and thus consider the following problem.

Find minimal number of nodes in V_1 which “cover” all nodes in V_2.

Now consider an equivalent form of the task – formulated as the set cover problem.

Set cover

Given a finite set S, a collection C of subsets of S, and a positive integer $K \leq |C|$. Does there exist a cover $C' \subseteq C$ of S such that $|C'| \leq K$, i.e. does there exist a subset $C' \subseteq C$ such that $|C'| \leq K$ and every element of S is in at least one subset of C'. This is the *decision version* of the set cover problem.

In the set covering *optimization version*, the task is to find minimal cover. The decision version of the set covering is NP-complete, and the optimization version is NP-hard ([Garey,Johnson, 1979]).

If we associate with each vertex $v \in V_1$ the subset of vertices of V_2 connected to v, then in this manner, V_1 can be considered as collection of subsets of V_2.

Initially, V_1 is a cover for V_2, since each attack node in V_2 is connected with some vulnerability in V_1. Thus, we formulate the task as optimization version of set covering:

Task2. Given a finite set V_2, a collection V_1 of subsets of V_2. Find a minimal cover of V_1.

Many algorithms have been developed for solving the set cover problem. The exact algorithms are mostly based on branch-and-bound and branch-and-cut (e.g. [Balas et al, 1996]). Since exact methods require substantial computational effort to solve large-scale instances of the problem, heuristic algorithms are often used to find a good or near-optimal solution in a reasonable time [Sahakyan, 2014]. Greedy algorithms may be the most natural heuristic approach for quickly solving large combinatorial problems ([Vazirani, 2001], [Bendorz, 2008]).

Greedy Approximation Algorithm for Set cover.

Consider the greedy approximation algorithm for *Task2* :

Algorithm G

```
Input: Undirected bipartite graph $D$ with parts $V_1$ and $V_2$.
Output: cover $C$.
begin
$C = \emptyset$;
while ($V_2$ is not empty)
{ take $v \in V_1$ which is connected to maximum number of vertices of $V_2$;
  remove those vertices from $V_2$;
  add $v$ into $C$
}
end
```

The algorithm G finds cover whose size is at most O($ln|V_2|$) times the size of minimal set cover.

A similar task is when neutralizing of certain attacks is the interest, which requires that certain vertices are to be covered in V_2.

Task3. Find minimal number of nodes in V_1 that “cover” given vertices $v_1, \cdots, v_k$ in V_2.

Observe that we have simplified the task when consider all vertices of V_1, whilst only those vertices, which are starting points of attack paths may be satisfactory.

Algorithm G' below is a modified version of G.

Algorithm G'.

Input: Directed bipartite graph D with parts V_1 and V_2.

Output: cover C.

```
begin
$C = \emptyset$;
while ($V_2$ is not empty)
{ take $v \in V_1$ such that $deg^-(v) = 0$ and $deg^+(v)$ is maximal ($V'$ denotes the set of heads for arcs starting in $v$);
 for each $v' \in V'$
        {
          if ($deg^+(v') \neq 0$)
          Removing_procedure($v'$);
        }
        remove $V'$ from $V_2$;
        add $v$ into $C$;
}
end
```

Removing_procedure(v') - removes all arc heads having v' as head or tail; and continue this process while the current node has out-degree greater than 0.

Conclusion

Attack graphs as a useful model and tool in the areas of network security can be identifying vulnerabilities that should be removed. We have proposed a general model of representing and generating attack graphs. A simple algorithm of generating attack graph has been described for the attack graph model. Some graph-theoretical problems and algorithms are investigated related to particular network hardening tasks. For future work we plan to improve the algorithm and to create an implementation using real-world network data in realistic situations.

Bibliography

[Aslanyan et al, 2013] L. Aslanyan, D.Alipour and M.Heidari, Comparative Analysis of Attack Graphs. Mathematical Problems of Computer Science, 40, 2013, pp. 85-95.

[Balas et al, 1996] E. Balas, M. Carrera, A dynamic subgradient-based branch-and-bound procedure for set covering. Operations Research 44, 1996, pp. 875–890.

[Barik et al, 2014] M.S. Barik, Ch. Mazumdar, A Graph Data Model for Attack Graph Generation and Analysis, Recent Trends in Computer Networks and Distributed Systems Security, Communications in Computer and Information Science Volume 420, 2014, pp 239-250.

[Bendorz, 2008], Greedy algorithms, edited by W. Bednorz, Publisher: InTech, 2008, 586 pages.

[CVE] Common Vulnerabilities and Exposures (CVE®), the standard for Information security Vulnerability Names, [Online]. Available: http://cve.mitre.org.

[CVSS] Common Vulnerability Scoring System (CVSS-SIG), [Online], Available: http://www.first.org/cvss

[CWE] Common Weakness Enumeration, [Online], Available: http://cwe.mitre.org/data/definitions/416.html.

[Jorgen et al, 2007] Jørgen Bang-Jensen, Gregory Gutin, Digraphs: Theory, Algorithms and Applications, Springer-Verlag, 2002, 754 pages.

[Noel et al, 2010] S. Noel, L. Wang, A. Singhal and S. Jajodia, Measuring security risk of networks using attack graphs, International Journal of Next-Generation Computing, vol. 1, no. 1, 2010, pp. 135-147.

[NVD] National Vulnerability Database, [Online], Available: https://nvd.nist.gov/.

[OSVD] Open Sourced Vulnerability Database, [Online], Available: http://osvdb.org/.

[Sahakyan, 2014] H. Sahakyan, Constrained object-characterization tables and algorithms, International Journal "Information Content and Processing", Volume 1, Number 2, 2014, pp.136-144.

[Shey et al, 2002] O. Sheyner, J. Haines, S. Jha, R. Lippmann, J. M. Wing, Automated generation and analysis of attack graphs, Proceedings of the IEEE Symposium on Security and Privacy, 2002, pp. 254–265

[Vazirani, 2001], V. Vazirani,. Approximation Algorithms, Springer, 2001.

[Zhang et al, 2009] Zhang Lufeng, Tang Hong,Cui YiMing, Zhang JianBo, Network Security Evaluation through Attack Graph Generation, World Academy of Science, Engineering and Technology, 54, 2009.

Authors' Information

Hasmik Sahakyan *– Scientific Secretary, Institute for Informatics and Automation Problems, NAS RA, P. Sevak St. 1, Yerevan 14, Armenia, e-mail: hasmik@ipia.sci.am*

Daryoush Alipour *– PhD student, Institute for Informatics and Automation Problems, NAS RA, P. Sevak St. 1, Yerevan 14, Armenia, e-mail: computernano@gmail.com*

ARTIFICIAL ANALYSIS OF MOLECULAR MARKER LOCI LINKED TO TREE RESISTANCE RESPONSE BY AN ARTIFICIAL NEURAL NETWORK

Jorge Fernández, Angel Castellanos, Juan Castellanos

***Abstract**: Citrus tristeza virus (CTV) is one of the most important pathogen affecting citrus and no CTV resistant scion varieties are available. Since Chandler pummelo was found to be CTV resistant, this variety is being used as a donor of CTV resistance given that incorporation of resistance genes into commercial varieties will definitely offer an ultimate solution to CTV problem. To facilitate this breeding program through marker assisted selection (MAS), the analysis of the percentage of influence of molecular marker loci linked to CTV resistance response was done using an artificial neural network (ANN) model. Three main molecular marker loci associated with the Chandler pummelo resistant response to CTV inoculation were detected, allowing the MAS and decreasing the economic cost of breeding programs. Two of those molecular marker loci located at the same genomic region were the Poncirus trifoliata dominant gene responsible for its CTV resistance is located, supporting the theory of a common disease resistance gene cluster between those species that may supply a resource for P. trifoliata and citrus resistance to different pathogens including CTV.*

***Keywords**: Citrus tristeza virus, tree resistance to virus, marker loci, neural network*

Introduction

Citrus trees are economically the most important fruit crop, with an annual production exceeding 149 million tons in 2012 [FAOSTAT, 2015]. The main cultivated species are sweet oranges (*Citrus sinensis* (L.) Osb.), mandarins (mainly *C. clementina* Hort. Ex Tan. and *C. unshiu* (Mak.) Marc.), grapefruits (*C. paradise* Macf.), pummelos (*C. grandis* (L.) Osb.) and lemons (*C. limon* L. Burm. f.). Cultivars of all these species are always vegetatively propagated by bud-grafting onto a seedling rootstock in order to obtain a more uniform and early yielding tree with tolerance to *Phytophthora*, parasitic nematodes and some viruses, and well adapted to the local edaphoclimatic conditions. Sour orange (*C. aurantium* L.) was the most common rootstock before 1930, while rough lemon (*C. jambhiri* Lush.) and trifoliate orange (*Poncirus trifoliate* (L.) Raf.) were used in areas where sour orange performed poorly, such as Australia and South Africa, because of tristeza disease (caused by Citrus tristeza virus, CTV). Therefore, two kinds of citrus breeding populations, groups of species, and target traits, are managed for rootstock and scion improvement.

CTV is a member of the genus Closterovirus, family Closteroviridae, and one of the most important pathogens affecting citrus. It has a genome of 19.2 kb, which is the largest among RNA plant viruses. CTV probably originated in Asia, which is also the centre of origin of Citrus, and has been disseminated to many countries by movement of infected plant material. Subsequent natural spread by aphid vectors has created major epidemics. CTV dispersal to other regions and its interaction with new scion varieties and rootstock combinations resulted in three distinct syndromes named tristeza, stem pitting and seedling yellows. The first, inciting decline of varieties propagated on sour orange, has forced the rebuilding of many citrus industries using tristeza-tolerant rootstocks. The second, inducing stunting, stem pitting and low bearing of some varieties, causes economic losses in an increasing number of countries. The third is usually observed by biological indexing, but rarely in the field. It was estimated that almost 100 million citrus trees had been killed by CTV [Moreno et al, 2008]. In the absence of exclusion of infection, there are no satisfactory management strategies against severe CTV-induced diseases [Bar-Joseph et al, 1989].

One of the most effective general means of managing plant diseases has been through the use of resistant varieties, but most citrus species are hosts of CTV. Citrus genetic resources are rich but underutilized in breeding because their complex reproductive biology and the scarceness of inheritance studies on agronomic traits. Up to now, the citrus and related genotypes where CTV resistance has been found are: trifoliate orange [Yoshida et al, 1983], the Meiwa kumquat (*Fortunella crassifolia*) [Mestre et al, 1997b] and the pummelo “Chandler” (*C. grandis*) [Garnsey et al, 1996]. All cultivars of *P. trifoliate* tested have been found resistant to most CTV isolates, making it the specie of choice as donor of CTV resistance in all breeding programs for citrus rootstocks. Nevertheless, breeding a marketable CTV-resistant scion cultivar by crossing trifoliate orange with citrus has not been possible because undesirable traits of trifoliate orange remain after several generations of backcrossing with citrus. Then, breeding strategies based on transgenic technology directed to introduce the dominant gene responsible for the CTV resistance of *P. trifoliata* in to scion varieties, and the obtaining of pathogen-derived resistance by plant transformation are presented as interesting open lines of research. Otherwise, the resistance reported in some pummel cultivar such as Chandler opens a possible way to breed CTV-resistant citrus scion cultivars by sexual hybridization, but the genetic control of such resistance has hardly been studied [Fang & Rose, 1999].

Citrus breeding takes a long time due to the long juvenility period of these species, and is very expensive because of the long time needed and the huge cultivation costs for maintaining and evaluating large segregating progenies. To overcome such limitations, marker assisted selection (MAS) within the progenies is a valuable tool. A first step towards obtaining those tools has been the genetic dissection and mapping of the resistance gene(s). Previous studies have reported quantitative trait loci (QTL) controlling CTV resistance in Chandler pummelo cultivars [Asins et al, 2012].

Computational tools such as artificial neural networks (ANNs) represent a new approach hardly employed in genetic studies. The attractiveness of ANNs comes from their remarkable information processing characteristics pertinent mainly to non linearity, high parallelism, fault and noise tolerance, and learning and generalization capabilities [Basheer & Hajmeer, 2000]. Neural networks [Anderson & Rosenfeld, 1988] are non-linear systems whose structure is based on principles observed in biological neuronal systems [Hanson & Burr, 1990]. Neural networks can predict any continuous relationship between inputs and the target. Then, a neural network could be seen as a system that can be able to answer a query or give an output as answer to a specific input. Similar to linear or non-linear regression, ANNs develop a gain term that allows prediction of target variables for a given set of input variables. Physical–chemical relationships between input variables and target variables mayor may not built in to the association of target and input variables. The in/out combination, i.e. the transfer function of the network is not programmed, but obtained through a training process on empiric datasets. In practice the network learns the function that links input together with output by processing correct input/output couples. Actually, for each given input, within the learning process, the network gives a certain output that is not exactly the desired output, so the training algorithm modifies some parameters of the network in the desired direction. Hence, every time an example is input, the algorithm adjusts its network parameters to the optimal values for the given solution: in this way the algorithm tries to reach the best solution for all the examples. These parameters we are speaking about are essentially the weights or linking factors between each neuron that forms our network.

Calibrating a neural network means to determinate the parameters of the connections (synapses) through the training process. Once calibrated there is needed to test the network efficiency with known datasets, which has not been used in the learning process. There is a great number of Neural Networks [Anderson, 1995] which are substantially distinguished by: type of use, learning model (supervised/non-supervised), earning algorithm, architecture, etc. Multilayer perceptrons (MLPs) are layered feed forward networks typically trained with static back propagation. These networks have found their way in to countless applications requiring static pattern classification. Their main advantage is that they are easy to use, and that they can approximate any input-output map. In principle, back propagation provides a way to train networks with any number of hidden units arranged in any number of layers. In fact, the network does not have to be organized in layers, any pattern of connectivity that permits a partial ordering of the nodes from input to output is allowed. In other words, there must be a way to order the units such that all connections go from earlier (closer to the input) to later ones (closer to the output). This is equivalent to stating that their connection pattern must not contain any cycles. Networks that respect this constraint are called feed forward networks; their connection pattern forms a directed acyclic graphordag.

The objective of our study was to appraise the percentage of influence of eleven molecular marker loci linked to Chandler pummelo cultivar CTV resistance response using an ANN.

Materials and methods

Plant materials

The segregating population of 201 *C. grandis* x *C. clementina* full-sib hybrids derived from a cross between Chandler (Ch) and Fortune (F) commercial varieties, was employed. Fortune is a hybrid mandarin derived from the cross between *C. clementina* Hort. ex Tan. and *C. tangerine* Hort. ex Tan. Chandler is a hybrid pummelo derived from a cross between two accessions of *C. grandis* (L) Osbeck. Genetic linkage maps of these species were previously built up using this segregating population [Bernet et al, 2010]. Quantitative trait locus (QTL) analysis of accumulation and distribution of CTV was also previously carried out with this segregating population [Asins et al, 2012].

To evaluate these 201 hybrids for CTV resistance, they were propagated on sweet orange rootstocks, given that CTV replicates and accumulates abundantly in sweet orange. Every plant was grown in a separate container, in the same greenhouse (25 ± 10ºC). Each propagation was inoculated at the rootstock by grafting two patches of infected sweet orange with CTV isolate T-346, a common Spanish isolate, kept at the bank of CTV isolates at Instituto Valenciano de Investigaciones Agrarias (IVIA) [Ballester-Olmos et al, 1993].

Evaluation of CTV accumulation and distribution was done as described in [Asins et al, 2012]. CTV was monitored and its titer evaluated in each tree by two recommended serological ELISA methods [EPPO, 2004] using specific monoclonal antibodies 3CA5 and 3DFI together, as described in [Cambra et al, 1993]. A plant was declared resistant when CTV was detected at the original inoculum but not at any branch, and its DAS-ELISA (Double Antibody Sandwich ELISA) values through years were similar to those of un-inoculated plants (negative control). Those hybrids where the virus was detected by both serological methods were considered susceptible.

Molecular markers

Eleven of the molecular markers loci analyzed in the segregating population were selected, because of their linkage with the resistance response to CTV inoculation. Simple Sequence Repeats (SSRs), Sequence Characterized Amplified Regions (SCARs), Inter-Retrotransposon Amplified Polimorphism (IRAPs) and one resistance gene analogue were included. Four of those markers were common between both parents and behaved as codominant. Given that they allowed the unambiguous classification of hybrids into four possible genotypes, were considered independently for the neural networks analysis obtaining a total of 15 molecular marker loci.

Artificial neural networks analysis

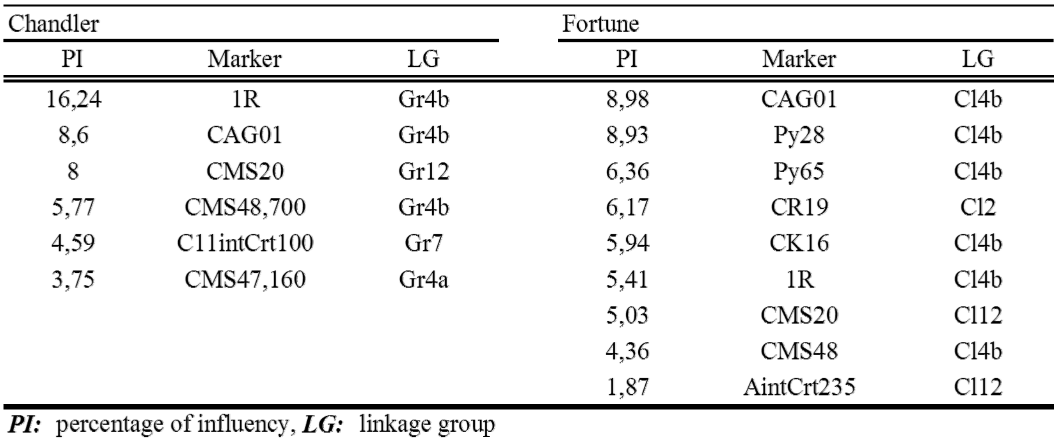
We use neural networks models with analysis of sensibility because the process of finding relevant data components is based on the concept of sensitivity analysis applied to trained neural networks. This model predicts more accurately the relationship existing between variables. The suitable way to find the individual effects of forecasting variables (15 molecular marker loci) over the variable to forecast (plant response to CTV inoculation), and the way to find a set of forecasting variables (additional marker loci) to include in the new model generated. We have studied different analysis for detecting relationships between those 15 molecular marker loci analyzed in the segregating population and the response to CTV inoculation. In order to study the relationships between those different variables, neural networks models MLP (multilayer perceptron) with a two hidden layer with 4 axons and a Tanh transfer function were used.

Results

Chandler and 13 of its hybrids (6.47%) were found to be resistant to CTV isolate T-346, since it was not detected after the inoculation during the whole experiment.

After training the network, a study of sensitivity was made obtaining the percentages of influence of those molecular markers loci considered to CTV inoculation response (Table 1).

Table 1. Percentages of influence of the molecular markers loci considered to CTV inoculation response

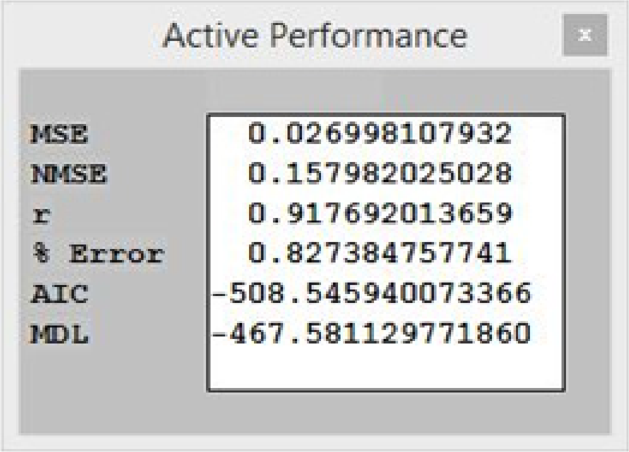


Chandler			Fortune		
PI	Marker	LG	PI	Marker	LG
16,24	1R	Gr4b	8,98	CAG01	Cl4b
8,6	CAG01	Gr4b	8,93	Py28	Cl4b
8	CMS20	Gr12	6,36	Py65	Cl4b
5,77	CMS48,700	Gr4b	6,17	CR19	Cl2
4,59	C11intCrt100	Gr7	5,94	CK16	Cl4b
3,75	CMS47,160	Gr4a	5,41	1R	Cl4b
			5,03	CMS20	Cl12
			4,36	CMS48	Cl4b
			1,87	AintCrt235	Cl12

PI: percentage of influency, ***LG:*** linkage group

The General performance probe displays the Mean Squared Error (MSE), the Normalized Mean Squared Error (NMSE), the Correlation Coefficient (r), and the Percent Error. See Figures 1 and 2 below:

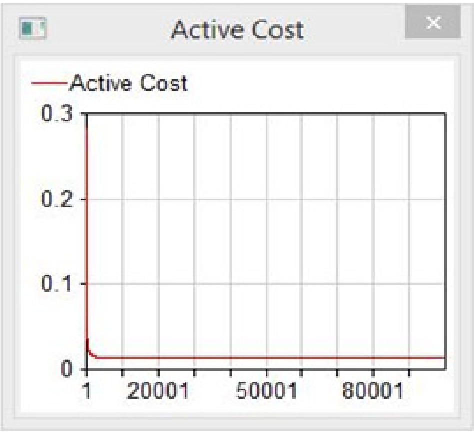
Figure 1. Table of mistake obtained.



Active Performance

```
MSE        0.026998107932
NMSE       0.157982025028
r          0.917692013659
% Error    0.827384757741
AIC     -508.545940073366
MDL     -467.581129771860
```

Figure 2. Graph of mistake.



ANN analysis indicates that the main molecular markers loci linked to CTV inoculation response are 1R and CAG01. Both markers loci are located in linkage group 4 of *C. grandis* and *C. clementina* maps. CMS20 marker locus, located on linkage group 12, is also important. This methodology has allowed the detection of two main genomic regions, one of them associated with candidate gene 1R [Bernet et al, 2004], involved in the plant response to CTV inoculation.

Discussion

Pummelo is commercially cultivated in many countries, such as China, Thailand, Japan, Mexico and Israel, because pummelo fruits and juice have abundant antioxidant compounds, including vitamin C, carotenoids, flavonoids and limonoids [Mokbel & Hashinaga, 2006]. Pummelo is also considered a valuable germoplasm for citrus breeding. Since Chandler pummelo was found to be CTV resistant, this variety is being used as a donor of CTV resistance for scion improvement given that incorporation of resistance gene(s) into commercial varieties will definitely offer an ultimate solution to CTV problem. To facilitate this breeding program through MAS, the genetic analysis of CTV resistance was pursued. The ANNs analysis reveal the detection of two main genomic regions, one of them associated with candidate gene 1R [Bernet et al, 2004], involved in the plant response to CTV inoculation. The region with the higher percent of influence located at linkage group 4, in the same genomic region where the major CTV resistance QTL from *P. trifoliata* and *C. aurantium* were previously mapped [Asins et al, 2004]. Our results disagree with those reported by [Fang & Rose, 1999], as they indicate that Chandler CTV resistance was controlled by a single dominant gene not allelic with *Ctv* (P. trifoliata gene). Sequence analyses of the genomic region were the *P. trifoliate* dominant gene responsible for its CTV resistance is located reveal the presence of a disease resistance-gene cluster including at least five functional R genes [Yang et al, 2003]. On the other hand, comparative analysis of that genomic region sequence of Poncirus, C. grandis, C. Clementine and C. sinensis detected the presence of a diverse group of retrotransposable elements (REs) suggesting that their activity has led to the considerable variation in localization and resistance-gene copy number between those species [Rawat et al, 2015]. Additionally, hypothetical chromosomal rearrangements affecting this genomic region, as those reported in the evolution of this group of species [Raghuvanshi, 1962] can also explain those variations observed. The clustering of disease resistance genes is a common occurrence in plant genomes [Michelmore & Meyers, 1998]. This disease resistance gene cluster may supply a resource for *P. trifoliata* and citrus resistance to different pathogens including CTV. Therefore, further genetic and physical mapping of that genomic region would provide important information on the evolution and function of disease-resistance genes in *Poncirus* and *Citrus*.

This is the first time CTV resistance has been genetically analyzed by an ANN system, a form of machine learning from the field of artificial intelligence utilized in many areas of bioinformatics, biotechnology and medicine [Basheer & Hajmeer, 2000]. Previous studies have reported QTL controlling CTV resistance. However, before embarking upon a QTL analysis, two conditions need to be fulfilled to ensure the data are suitable [Gupta, 2002]:

1. The molecular marker whose association with the trait of interest is being examined can’t exhibit any segregation distortion since it may lead to biased estimate of marker-trait association.
2. The phenotypic data on the quantitative trait should show a normal distribution among the segregating population, and in case normality is not present, the data need to be transformed on a scale that will achieve normality of distribution.

On the other hand, ANNs have many advantages in their ability to derive meaning from large complex datasets. First, they do not rely on data to be normally distributed, an assumption of classical parametric analysis methods. They are able to process highly dimensional datasets, data containing complex (non-linear) relationships and interactions that are often too difficult or complex to interpret by conventional linear methods. Another advantage is that they are fault tolerant they have the ability of handling noisy or fuzzy information, whilst also being able to endure data which is incomplete or contains missing values. Nevertheless, before embarking upon ANN analysis care needs to be exercised as the addition of a given variable into a forecasting model does not implies that this variable will have an important effect over the response of the model. That is, if a researcher identifies a set of forecasting variables, he must check if they really affect the response. A frequent problem is that some of the forecasting variables are correlated. If the correlation is small, then consequences will be less important. However, if there is a high correlation between two or more forecasting variables, then the model results will be ambiguous but not for obtain a bad prediction, the problem is the high correlation between variables (high lineal association) decrease in a drastic way the individual effect over the response for each correlation variable and sometimes is difficult to detect and is not possible measure the real effect for each variable over the output.

Conclusion

Here we report the two main genomic regions involved in the Chandler Pummelo CTV resistant response detected by an ANN analysis. Those results allow the molecular marker assisted selection in citrus breeding programs based on the sexual hybridization of Chandler Pummelo with commercial varieties, in order to obtain CTV resistant scion cultivars.

Bibliography

[Anderson & Rosenfeld, 1988] Anderson JA, Rosenfeld E. Neurocomputing: Fundations of research. The MIT Press. 1988.

[Anderson, 1995] Anderson JA. An introduction to neural networks. The MIT Press. 1995.

[Asins et al, 2004] Asins MJ, Bernet GP, Ruiz C, Cambra M, Guerri J, Carbonell EA. QTL analysis of Citrus Tristeza Virus-citradia interaction. Theor Appl Genet. 2004, 108:603-611.

[Asins et al, 2012] Asins MJ, Fernández-Ribacoba J, Bernet GP, Gadea J, Cambra M, Gorris MT, Carbonell EA. The position of the major QTL for Citrus tristeza virus resistance is conserved among Citrus grandis, C. aurantium and Poncirus trifoliata. Mol Breeding 2012, 29:575–587.

[Ballester-Olmos et al, 1993] Ballester-Olmos JF, Pina JA, Carbonell EA, Moreno P, Hermoso de Mendoza A, Cambra M, Navarro L. Biological diversity of citrus tristeza virus (CTV) isolates in Spain. Plant Pathology 1993, 42:219-229.

[Bar-Joseph et al,1989] Bar-Joseph M, Marcus R, Lee RF. The continuous challenge of Citrus Tristeza Virus control. Annu Rev Phytopatol 1989, 27:291-316.

[Basheer & Hajmeer, 2000] Basheer IA, Hajmeer M. Artificial neural networks: fundamentals, computing, design, and application. J Microbiol Methods 2000, 43:3–31.

[Bernet et al, 2004] Bernet GP, Bretó MP, Asins MJ. Expressed sequence enrichment for candidate gene analysis of Citrus Tristeza Virus resistance. Theor Appl Genet 2004, 108:592-602.

[Bernet et al, 2010] Bernet GP, Fernández-Ribacoba J, Carbonell EA, Asins MJ. Comparative genome-wide segregation analysis and map construction using a reciprocal cross design to facilitate citrus germplasm utilization. Molecular Breeding 2010, 25:659-673.

[Cambra et al, 1993] Cambra M, Camarasa E, Gorris MT, Garnsey SM, Gumpf DJ, Tsai MC. Epitope diversity of citrus tristeza virus isolates in Spain. Proc 12th Conf IOCV 1993, Riverside, California, USA. pp 33-38.

[EPPO, 2004] Standards PM 7/31 diagnostic protocol for citrus tristeza closterovirus. OEPP/EPPO Bull 2004, 34:155–157

[Fang & Rose, 1999] Fang DQ, Roose ML. A novel gene conferring Citrus Tristeza Virus Resistance in Citrus maxima (Burm.) Merril. HortScience 1999, 34:334-335.

[FAOSTAT, 2015] FAOSTAT http://faostat.fao.org/site/567/DesktopDefault.aspx Cited 4 May 2015.

[Garnsey et al, 1996] Garnsey SM, Su HJ, Tsai MC. Differential susceptibility of Pummelo and Swingle citrumelo to isolates of citrus tristeza virus. Thirteenth IOCV Conference 1996, 138–146.

[Gupta, 2002] Gupta PK. Molecular marker and QTL analysis in crop plants. Current Science 2002, 83:113-114.

[Hanson & Burr, 1990]. Hanson SJ, Burr DJ. What connectionist models learn: Learning and representation in connectionist networks. Behavioral and Brain sciences 1990, 13:471-518.

[Mestre et al, 1997b] Mestre PF, Asins MJ, Pina JA, Navarro L. Efficient search for new resistant genotypes to the citrus tristeza closterovirus in the orange subfamily Aurantioideae. Theor Appl Genet 1997b, 95:1282-1288.

[Michelmore & Meyers, 1998] Michelmore RW, Meyers BC. Clusters of resistance genes in plants evolve by divergent selection and a birth-and-death process. Genome Res 1998, 8:1113–1130.

[Mokbel & Hashinaga, 2006] Mokbel MS, Hashinaga F. Evaluation of the antioxidant activity of extracts from buntan (Citrus grandis Osbeck) fruit tissues. Food Chemistry 2006, 94:529-534.

[Moreno et al, 2008] Moreno P, Ambros S, Biach-Marti MR, Guerri J, Peña L. Plant diseases that changed the world – Citrus tristeza virus: a pathogen that changed the course of the citrus industry. Molecular Plant Pathology 2008, 9(2):251-268.

[Raghuvanshi, 1962] Raghuvanshi SS. Cytogenetical studies in the genus Citrus IV. Evolution in genus Citrus. Cytologia 1962, 27:172–188

[Rawat et al, 2015] Rawat N, Deng Z , Gmitter FG. Genomic structure and evolution of the Citrus Tristeza Virus (CTV) resistance locus in Poncirus and Citrus. XXIII International plant and animal genome conference 2015, P0927.

[Yang et al, 2003] Yang ZN, Ye XR, Molina J, Roose ML, Mirkov TE. Sequence Analysis of a 282-Kilobase Region Surrounding the Citrus Tristeza Virus Resistance Gene (Ctv) Locus in Poncirus trifoliata L. Raf. Plant Physiol 2003, 131:482-492.

[Yoshida et al, 1983] Yoshida T, ShichijoT, Ueno I, Kihara T, Yamada Y, Hirai M, Yamada S, Leki H, Kuramoto T. Survey for resistance of citrus cultivars and hybrid seedlings to citrus tristeza virus (CTV). Bull Fruit Tree Res Stn B 1983, 10:51-68.

Authors' Information

Jorge Fernández *– Ph.D. student at faculty of Biological Sciences, Universidad Complutense de Madrid, Ciudad Universitaria, 28040, Madrid, Spain; e-mail: jribacob@gmail.com*

Angel Castellanos *– Applied Mathematics Department. Universidad Politécnica de Madrid, Madrid; Spain; e-mail: angel.castellanos@upm.es*

Major Fields of Scientific Research: Artificial Intelligence, applied mathematics

Juan Castellanos *– Head of Natural Computing Group, Universidad Politécnica de Madrid, Campus de Montegancedo s.n., 28660 Boadilla del Monte, Madrid, Spain; e-mail: jcastellanos@fi.upm.es*

Major Fields of Scientific Research: Natural computing, formal language and automata theory.

ANT COLONY OPTIMIZATION FOR TIME DEPENDENT SHORTEST PATH PROBLEM IN DIRECTED MULTIGRAPH

Leonid Hulianytsky, Anna Pavlenko

***Abstract**: The paper concerns the approach for searching shortest path between specified nodes in a given graph that represents scheme of possible flights and takes into account time-dependent price. The path may be constructed according to request constraints: time limits, cost, mandatory transit or prohibited items. To find the lowest path cost we developed the ant colony optimization (ACO) based algorithm. Natural parallelism and iterativity of original ACO processing scheme gives the possibility to get and update the best current solution at any moment taking into account flight data changes. The approach of single-generation ACO, that allows optimizing the use of resources and reducing the processing time is suggested and investigated. The paper presents a formal model of the problem, and describes the basic ACO scheme and properties of suggested approach. For assessment practical effectiveness of single-generation algorithm, the experiments are made. The comparison between offered and classical ACO schemes in time and accuracy is given.*

***Keywords**: ant colony optimization, time dependent shortest path problem*

***ACM Classification Keywords**: I.2.8 Problem Solving, Control Methods, and Search*

Introduction

Many of real life situations of communication or transportation networks can be well modeled into multigraphs (graphs in which multiple edges between nodes might exist) because of their ability operating multiple edges connecting a pair of nodes. Due to the increasing interest in the dynamic management of transportation systems, there are needs to find shortest paths over a large graph (e.g., a road network), where the weights associated with edges dynamically change over time [Ding, 2008].

Finding shortest path in graphs has been playing an important role in various fields of human activity for over 40 years. Typically, results must be found within a very short time period. In real-time searching systems new routes must be identified within a reasonable time after a customer requests [Fu et al, 2006]. General time-dependent shortest path problem is not new. Some of the first studies were published in 1958 in which Cook and Halsey [Cook & Halsey, 1969] proposed algorithm based on dynamic programming with discretizing time. Alternative ways to solve the problem for different problem variations where investigated by Dreyfus [Dreyfus, 1969], Dijkstra, Halpern, Orden and Rom and others.

The complexity of the problem and its wide application in many fields of human activity stimulates researching different approaches and methods. Much attention is paid to approximate bio-inspired search techniques [Pintea, 2014]. These include ant colony optimization proposed by Dorigo [Dorigo & Stützle, 2004; Dorigo & Stützle, 2010], which is successfully applied to combinatorial optimization problems.

Traditional optimal shortest path techniques often cannot be applied because they are too computationally intensive to be feasible for real-time operations. Numerous heuristic search strategies have been designed for enhancing computational efficiency of shortest path search. Algorithm ant colony optimization (ACO) has been successfully used to solve combinatorial optimization problems, including the traveling salesman problem, routing, sequential ordering, assignment problem, classification, etc. (particularly on dynamic graphs).

The following problem is described below: given an airlines flights scheme between specified set of cities (airports) with appropriate conditions and restrictions. The research concentrates on solving problem of finding the cheapest path for travelling via planes from source city to target through specified points. Worth to mention that flight's price varies over time that makes the problem time-dependent. For simplicity in our approach time space is discretized in a suitable way. Since there may be several flights between the same airports, network is represented via multigraph.

Problem Formulation

Given a directed multigraph $G=(V,A)$ where multiple edges or arcs might exist between pair of vertices and represent flight connections between airports offered by airlines, where $V=\{v_1,...,v_n\}$ is a set of n vertices; A is a set of arcs. Let (v_i,v_j) be a set of arcs from node v_i to v_j, $v_i,v_j \in A$, $N_{ij}=\|(v_i,v_j)\|$ is a number of such arcs, a_{ij}^k is a specific arc $a_{ij}^k \in (v_i,v_j), k\in \{1,...,N_{ij}\}$. It is possible if $(v_i,v_j)=\varnothing$ for some nodes and destinations.

According to the problem, it is required to find *optimal path* from source (starting) node $s\in V$ to destination node $d\in V$ when starting time t_0 (departure time from the source) can be selected in a user given starting-time interval $T=[t_{0_{min}},t_{0_{max}}]\subseteq T$ (it is supposed that at least one such path exists).

Consider a path $x(s,d,t_0)$ from point s to point d starting in time t_0 to be an arcs sequence $(a_{i_1i_2},a_{i_2i_3},a_{i_3i_4},\ldots,a_{i_{w-1}i_w})$ if

1. $i_1=s$, $i_w=d$;
2. $a_{ij}\in(v_i,v_j), v_i,v_j\in V, i,j\in\{i_1,i_2,\ldots,i_w\}$.

If transit across $a_{kl} \in A$, that belongs to path and corresponds to flight from point k to point l, is starts from k in time t_{k-1}, than arrival time to point l is $t_k = t_{k-1} + \lambda(a_{kl})$, where $\lambda(a_{kl})$ is flight's duration. Travel time is the difference between arrival time and starting time (1). Flight's durations is considered to be fixed.

$$t(x) = \sum_{k=1}^{w-1} \left[t(a_{i_k,i_{k+1}}) + g(a_{i_k}) \right] - t_0 \tag{1}$$

where $t(x)$ – full path duration;

$t(a_{i_k,i_{k+1}})$ – transition time across $a_{i_k,i_{k+1}}$ arc;

$g(a_{i_k})$ – time of waiting in starting node v_{i_k} of arc a_{i_k}, $g(a_{i_w}) = 0$;

$c(a_{i_k,i_{k+1}},t)$ is nonnegative transit-time function which represents generalized cost of travelling across arc $a_{i_k,i_{k+1}}$.

The cost of path x is defined as $c(x,t) = \sum_{k=1}^{w-1} c(a_{i_k,i_{k+1}},t)$, where w – number of arcs in route x.

The goal is to find optimal path $x^*(s,d,t_0)$ in terms of price (or a set of allowed routes with account to additional constraints).

There is also a set of additional constraints (2)-(6): given a set of mandatory vertices $V_{mandatory}$ (2) included in optimal path $x^*(s,d,t_0)$; a set of prohibited vertices $V_{prohibited}$ excluded from $x^*(s,d,t_0)$ (3); c_{max} – maximum allowed cost of x from s to d (4); maximum number of transition nodes n_{max} (5); satisfy the constraint of route time length (6). Worth to mention that it is not obligatory to change nodes at each iteration (every day).

$$V_{mandatory} \subseteq x^*, \tag{2}$$

$$\forall v \in V_{prohibited} : v \notin x^*, \tag{3}$$

$$c(x^*,t) \le c_{max}, \tag{4}$$

$$\left| x^* \right| \le n_{max}, \tag{5}$$

$$t_{min} \le t(x^*) \le t_{max}, \tag{6}$$

where t_{min}, t_{max} – minimum and maximum time period;

Time-Dependent Shortest-Path (TDSP) problem is to minimize travel cost (7) among allowed paths:

$$c(x^{*}, \mathrm{t}) = \min\{\mathrm{c(x, t)}\}, \forall x(s,d,t_{0}) . \tag{7}$$

ACO Approach Description

Ant Colony Optimization [Dorigo & Stützle, 2004] is inspired by the idea of solving optimization problems using low-level communication behavior of cooperative ants that seek a path between their colony and a source of food.

Same as in real life, ants start randomly wander upon finding food and then return to their anthill. During the walk they leave pheromone trails which make their path more attractive for other ants since their chance to succeed in finding food increases. The following ants will choose trails taking in account the amount of pheromone deposited on the ground and visible path length (distances to neighboring nodes). However pheromone trails are evaporating all the time thereby reducing its attractiveness strength. Obviously short and popular paths have higher density than longer ones. Modelling evaporation process in ACO helps to avoid convergence to local optimal solution. Figure 1 represents general scheme of ACO.

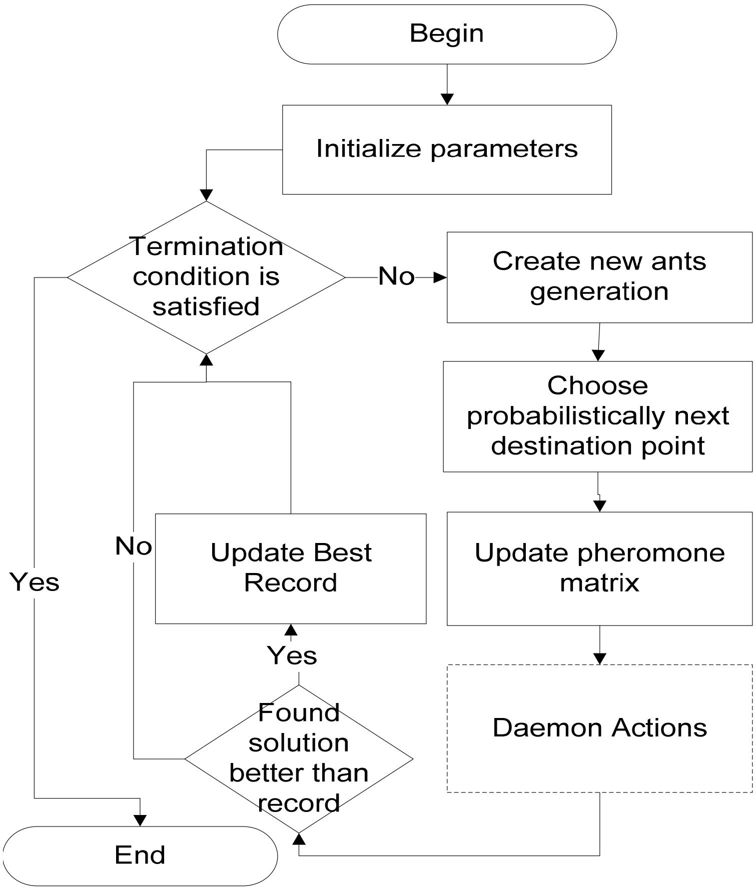

Begin
Initialize parameters
Termination condition is satisfied
No
Create new ants generation
Choose probabilistically next destination point
Update pheromone matrix
Daemon Actions
Found solution better than record
Yes
Update Best Record
No
Yes
End

Figure 1. General Ant Colony Optimization Scheme

Artificial ants in ACO represent stochastic procedures that move on the graph and constructs solutions.

The algorithm iteratively generates ants, which stochastically choose transit points and build route. Starting point for each ant is determined via problem constraints. On the first step algorithm creates and initializes matrices of distances between airports and pheromones, initial "best" route cost and other algorithm's variables. The next step creates a generation of ants that simultaneously go from the starting point. Each ant with a probability (8) defines its next point:

$$p_{ij}^{u} = \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum\limits_{allowed\ j}\tau_{ij}^{\alpha}\eta_{ij}^{\beta}} \tag{8}$$

where τ_{ij}^{α} – the amount of pheromone, deposited on the way from i to j, $0 \le \alpha$ – a parameter that controls impact of the deposited pheromone; η_{ij}^{β} – attractiveness of transition ij, based on a priori knowledge of the distance, $\eta_{ij} = 1/d_{ij}$, β – parameter that controls impact of η_{ij}. After all ants of current generation complete building routes, pheromone update occurs via formula (9):

$$\tau_{ij} := (1-\rho)\tau_{ij} + \sum_{u}\Delta\tau_{ij}^{u} \tag{9}$$

where ρ – pheromone evaporation coefficient, $\Delta\tau_{ij}^{u}$ – the amount of deposited pheromone by ant u, which is calculated via (10):

$$\Delta\tau_{ij}^{u} = \begin{cases} c_{predefined} / c(x_u), & if\ (ij) \in x_u\ , \\ 0, & otherwise \end{cases} \tag{10}$$

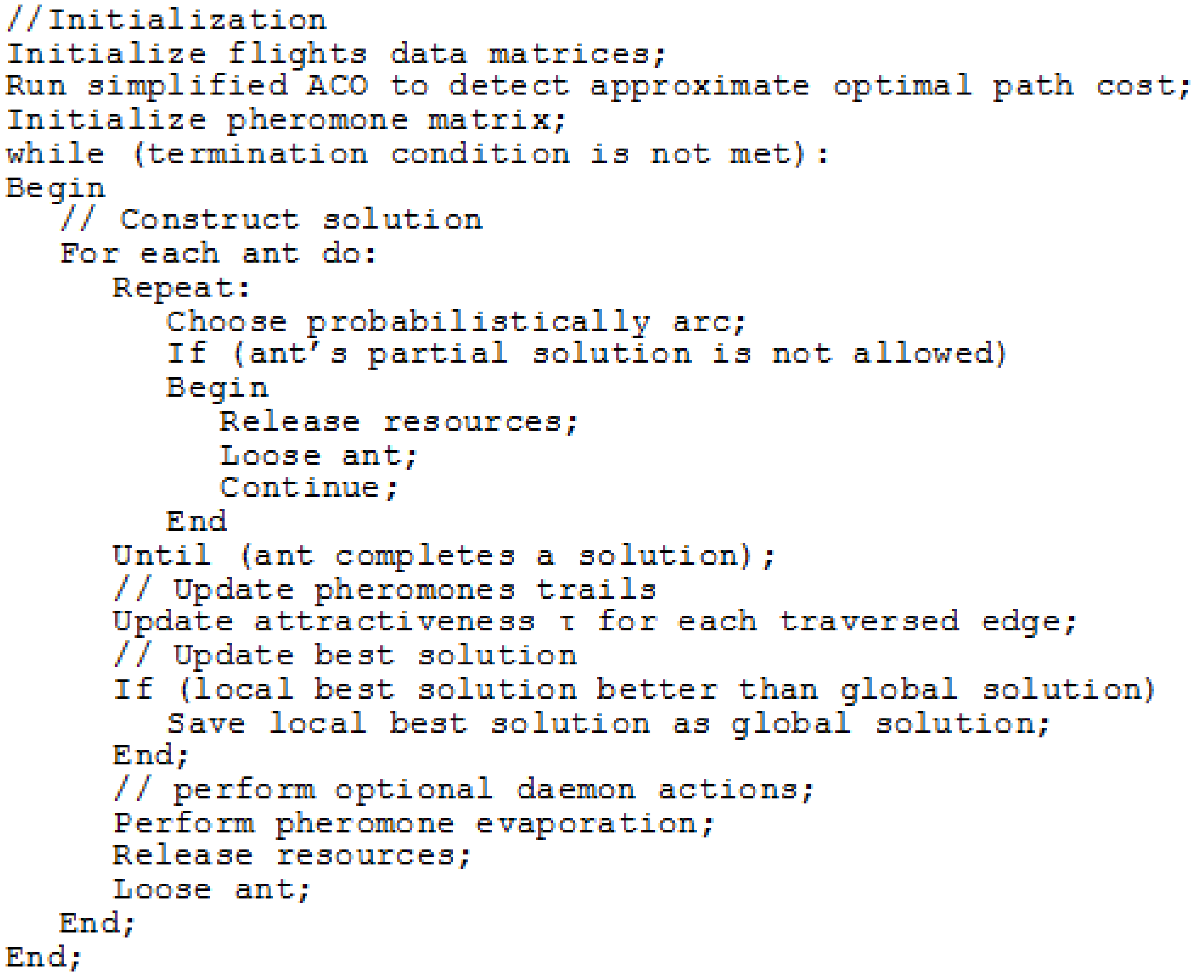
where $c_{predefined}$ – coefficient, which usually corresponds to the order of optimal route; $c(x_u)$ – ant's found route cost. If ants' generation produces better solution, than current record, it has to be updated. Different types of problems might have own heuristics – a priori knowledge that should be used to improve constructing solutions. Daemon actions are a kind of custom optional procedures that can be applied as a final step of the iteration according to specificity of the problem. After constructing full paths ants release all allocated resources and disappear.

If the termination condition is not satisfied, it creates a new generation of ants and new iteration of the algorithm starts. Termination conditions might be selected according to algorithm processing time, iterations number, number of iterations without updating best records etc.

ACO for Solving Time-Dependent Shortest Path in Multigraphs with Additional Constraints

Current research describes developed ACO based algorithm for finding optimal path in dynamic multigraphs with additional constraints. An important criterion of solving described problem is algorithm's processing time. In the research it is suggested to operate a given number of ants, not generations. Classical scheme uses the same total number of ants through all generations like single-generation approach.

Figure 2 represents general scheme of suggested single-generation approach. During initialization $c_{predefined}$ value is defined by additional run of simple ACO without additional constraints (3) - (7). If $c_{predefined}$ could not be found for specified number of iterations (no acceptable routes where found), the algorithm stops, informs about inability to calculate path and moves to the next request.



```
//Initialization
Initialize flights data matrices;
Run simplified ACO to detect approximate optimal path cost;
Initialize pheromone matrix;
while (termination condition is not met):
Begin
   // Construct solution
   For each ant do:
      Repeat:
         Choose probabilistically arc;
         If (ant's partial solution is not allowed)
         Begin
            Release resources;
            Loose ant;
            Continue;
         End
      Until (ant completes a solution);
      // Update pheromones trails
      Update attractiveness τ for each traversed edge;
      // Update best solution
      If (local best solution better than global solution)
         Save local best solution as global solution;
      End;
      // perform optional daemon actions;
      Perform pheromone evaporation;
      Release resources;
      Loose ant;
   End;
End;
```

Figure 2. Single Generation Ant Colony Optimization Scheme

At the first step all arcs that contain $V_{mandatory}$ (or target node) are predefined with value τ_{max} , others – default τ_0. While choosing transit points ants operate by arc, not nodes as in classical algorithm. It effectively deals with the problem of multiple arcs between some nodes. Once an ant completes building route or its partial route does not satisfy the conditions (2) - (6), the ant disappears and releases resources. Therefore, the algorithm does not hold any unused resources like classical ACO scheme does while synchronizing generation of ants. Worth to mention that ants collaborate with each other through pheromones trails all the time, whereas traditionally ants get information only from preceding ants' generations. Number of ants belongs to algorithm's parameters. Performance comparisons of classical and suggested approaches are presented below.

Pheromone update is performed after each successful ant's completed walk by formula (11) - (13) for arcs that belong to constructed route:

$$\tau_{ij} := \tau_{ij} + \Delta\tau_{ij}^{u}, \tag{11}$$

$$\tau_{min} \le \tau_{ij} \le \tau_{max}, \tag{12}$$

$$(ij) \in x_u, \tag{13}$$

where τ_{min} , τ_{max} – parameters of the algorithm. Pheromone evaporation is performed as daemon actions after each b -th created ant for the whole pheromone matrix (14). This is done to reduce update operations under database.

$$\tau_{ij} := (1-\rho)\tau_{ij}, \ \forall (ij) \in A \tag{14}$$

Computational Results

A number of tests have been conducted under realistic conditions, using the data collected by developed parsing client and APIs from global travel search site SkyScanner and Google (QPX Express). Obtained data includes 15497 flights for one week. According to the experiment's conditions, it is necessary to find the optimal path between Boryspil airport and 113 airports in Europe. For descriptive reasons the following charts contain info about 50 random target cities only. Figure 3 shows comparison of single-generation approach and classical ACO in time (in seconds). Classical scheme is on average 31% slower than suggested scheme. Offered algorithm was unable to find acceptable solutions for 25 target cities from 113, classical scheme – for 16 cities.

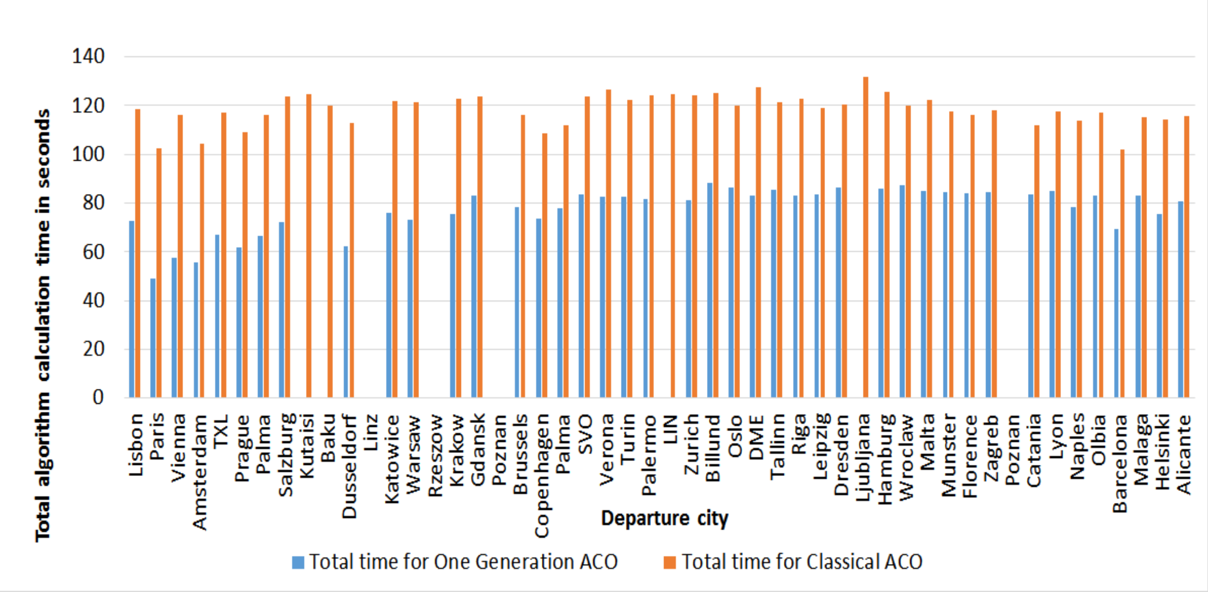


Figure 3. Comparison of Single-Generation Approach and Classical ACO by Algorithm's Processing Time

Figure 4 demonstrates experiment's results for random 50 target cities and illustrates comparison of single-generation approach and classical ACO by best solution in terms of route price (in EUR). Suggested scheme provides 16% better results than classical.

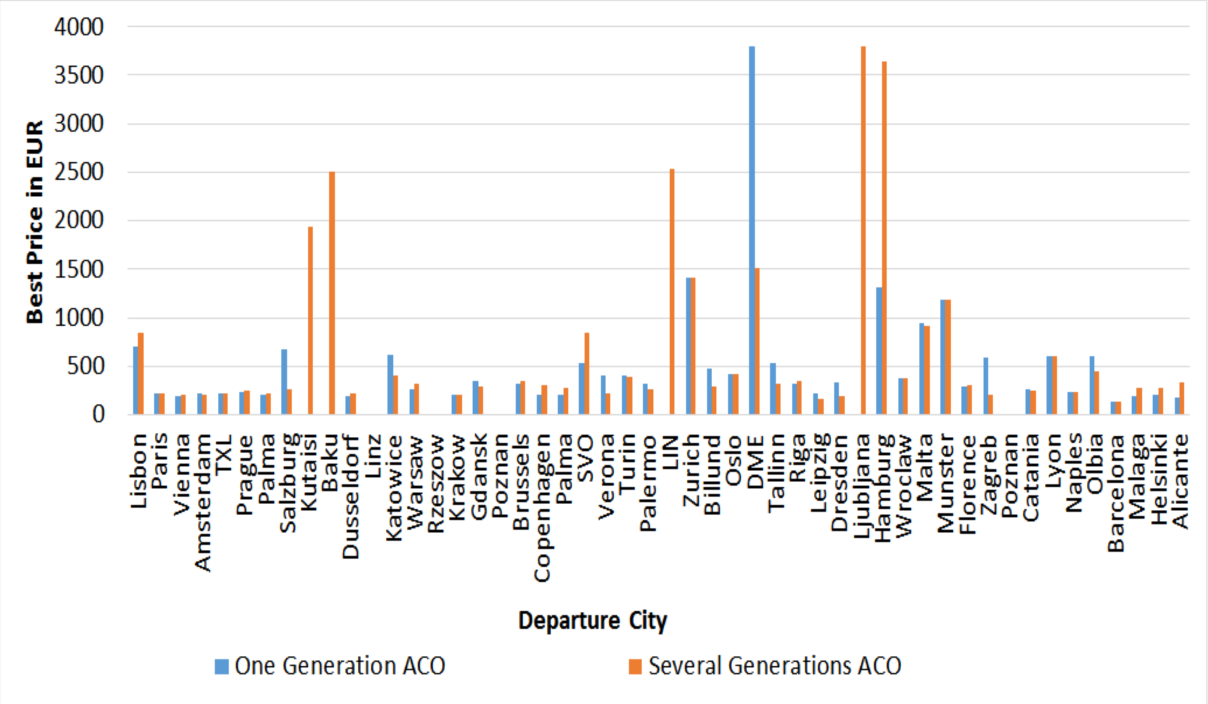


Figure 4. Comparison of Single-Generation Approach and Classical ACO by Best Solution in EUR

Table1 contains algorithm parameters, which were used in the experiments.

Table 1. Algorithm parameters values

Name	Value
Number of runs for each suggested approach	100
Total number of ants produced by algorithm	500
Impact rate of the deposited pheromone α	0.5
Impact rate of $\eta_{ij} - \beta$	0.5
Maximum number of transition nodes n_{max}	10
Generation size for classical ACO	20
Ant number after which starts evaporation in single-generation approach b	20
Default initial pheromone amount τ_0	0.1
Minimum pheromone amount τ_{min} per arc	0.1
Maximum pheromone amount τ_{max} per arc	0.7
Pheromone evaporation coefficient ρ	0.1
Maximum number of iterations to detect $c_{predefined}$	3000
Source airport	KBP

Conclusion

We propose the ACO based algorithm for solving the time dependent shortest path problem for large multigraph.

The description and formalization of time-dependent shortest problem with additional constraints are presented in the paper. For the need to minimize algorithm's processing time, the ACO classical scheme has been modified: the algorithm works with single generation and choosing next point procedure that operates with arches. The approach solves the problem with additional constraints: maximum number of transit cities, maximum price value, prohibited and mandatory cities. The algorithm allows to expand flexibly model by means of additional conditions and constantly to control the feasibility of solutions. In addition, it is proposed to carry out additional updates pheromone matrix along the arcs that contain mandatory and final city.

The paper represents experiments, that using the real data sets, and compares the suggested approach and the classical ant colony optimization algorithm. Our preliminary results show that the proposed algorithm provides good quality results and significantly less processing time algorithm. However, the algorithm often does not find optimal (feasible) solutions than the classical scheme.

Additional research needs optimization algorithm parameters, and the comparison with other search algorithms. Updating pheromone scheme could be improved for work with several user requests reusing existing information.

Bibliography

[Cook & Halsey, 1969] K. L. Cook, E. Halsey, “The shortest route through a network with time-dependent intermodal transit”, In J. Math. Anal. Appl., 1966, pp. 493-498.

[Ding, 2008] Ding. “Finding Time-Dependent Shortest Paths over Large”, In: Proceedings of the 11th international conference on Extending database technology: Advances in database technology. Nantes, France, 2008, pp. 205-216.

[Dorigo & Stützle, 2004] M. Dorigo M., T. Stützle, “Ant Colony Optimization, A Bradford Book”, MIT Press.Cambridge, Massachusetts, London, 2004, 305 p.

[Dorigo & Stützle, 2010] M. Dorigo, T. Stützle, “Ant colony optimization: overview and recent advances”, In: Handbook of metaheuristics. Springer US, 2010, pp. 227-263.

[Dreyfus, 1969] S.E. Dreyfus, “An appraisal of some shortest-path algorithms”, In Operations Research, 1969, pp. 395-412.

[Fu et al, 2006] L. Fu, D. Sun, L.R. Rilett, “Heuristic shortest path algorithms for transportation applications: State of the art”, Computers & Operations Research, Vol. 33, No. 11, 2006, pp. 3324-3343.

[Pintea, 2014] Pintea C.M., “Advances in Bio-inspired Computing for Combinatorial Optimization Problems”, Berlin, Heidelberg: Springer-Verlag, 2014, 188 p.

Authors' Information

Hulianytsky Leonid *– Dr.Sc. (Technology), Head of department of Glushkov Institute of Cybernetics of NAS of Ukraine, Professor of NTUU “KPI” (Kyiv); e-mail: leonhul.icyb@gmail.com*

Major fields of scientific research: combinatorial optimization; decision making; mathematical modeling and applications; forecasting.

Pavlenko Anna *– PhD student at Glushkov Institute of Cybernetics of NAS of Ukraine (Kyiv) Glushkov Ave., 40, Kyiv, 03680, Ukraine; e-mail: dmitrieva.anya@gmail.com*

Major fields of scientific research: artificial intelligence; combinatorial optimization; mathematical economics, forecasting

EXPERIMENTATION OF “ADVANCED INFTHEO” MODULE FOR “R” ON THE EXAMPLE OF BIOMETRIC GENERATED SECRET KEY SHARING SYSTEM

Mariam Haroutunian, Narek Pahlevanyan

Abstract: *Many information-theoretical results in practice are difficult to compute, because of the large volume of distributions. To perform computations of complex formulas of Information Theory authors have developed new module (Advanced Inftheo) for R language. For high performance the main functions of module use parallel algorithms and tools of C++ to dynamically and optimally allocate memory. In this paper we demonstrate some results of computations that are done by Advanced Inftheo module. As an example we compute and represent the lower and upper bounds of E-achievable secret key rate of the biometric generated secret key sharing system obtained in [Haroutunian-Pahlevanyan, 2014]. E-achievable secret key rate is the generalization of the secret key rate, studied by [Ignatenko-Willems, 2012].*

Keywords: *E-achievable secret key rate, R language, parallel computations*

ACM Classification Keywords: *H.0 Information Systems - Conference proceedings*

Introduction

The usage of biometric secrecy systems in modern society is growing very quickly. Biometric secrecy systems are based on the person’s physiological or behavioral characteristics. Physiological characteristics include fingerprints, hand geometry, facial, voice, iris, retinal features etc. [Chen-Vinck, 2011]. Behavioral characteristics include the dynamics of signatures, keystrokes etc. Biometric secrecy systems capture and process person’s unique characteristics, and then authenticate that person’s identity based on comparison of the record of captured characteristics with a biometric sample presented by the person to be authenticated. Biometric characteristics cannot be lost or forgotten, they are difficult to copy, share and distribute. Therefore, the advantage of biometric secrecy systems against the traditional password based security systems is evident.

Biometric secrecy systems can be used in various applications, such as in authentication, identification, examinations, payment processing, secure travel documents, visas et al. In many applications, such as for example, examinations the person is required to be present at the time and point of authentication. Moreover, there are access scenarios, which require participation of multiple previously registered users for a successful authentication or to get an access grant for a certain entity. For instance there are cryptographic constructions known as secret sharing schemes, where a secret key is split into shares and distributed amongst users in such a way that it can be reconstructed only when the necessary number of the secret key holders comes together. The revealed secret can then be used for encryption or authentication. One of such applications could be sharing of a bank account by family members.

As mentioned in [Ignatenko-Willems, 2012], biometric secrecy systems are grouped around two classes: cancelable biometrics and fuzzy encryption. In fuzzy encryption class systems a secret key is generated/chosen during an enrollment procedure, in which the biometric data are observed for the first time. The secret key is to be reconstructed after these biometric data are observed again, during an attempt to get access. Reliable biometric secret key sharing systems extract helper data from the biometric information at the time of enrollment. These helper data contributes to reliable reconstruction of the secret key.

However, the usage of biometric secrecy systems has its own disadvantages. Since biometric data are gathered from individuals under environmental conditions and the channels are exposed to noise the biometric secrecy system may accept an impostor or reject an authorized individual. It's not possible to build ideal biometric secrecy system, it can be information-theoretical secure up to a certain level. From information-theoretical point of view biometric secrecy systems were studied by O'Sullivan and Schmid [O'Sullivan - Schmid, 2002], Willems et al [Willems et al, 2003; Haroutunian-Pahlevanyan, 2014]. Willems [Willems et al, 2003] investigated the fundamental properties of biometric identification system. It has been shown that it is impossible to reliably identify more persons than capacity which is an inherent characteristic of any identification system. By analogy with notion of E-capacity or rate-reliability function introduced by E. Haroutunian [Haroutunian, 2007; Haroutunian et al, 2008] in [Haroutunian-Pahlevanyan, 2014] we introduce the new concept of E-achievable secret key rate for biometric generated secret key sharing system. The authors derived the upper and the lower bounds for E-achievable secret key rate of biometric generated secret key sharing system. E-achievable secret key rate expresses the dependance of the main characteristics of the system.

The construction of biometric secrecy systems that are both reliable and secure is strongly connected with investigation of rate-reliability function. In practice the investigation of rate-reliability function is complex and computational results are complicated to obtain, mostly because of the large volume of distributions. There are many statistical software packages that can help in computations, the popular ones are SAS, STATA, SPSS, Matlab, Mathematica. Moreover, statistical libraries are available in most of programming languages, for instance Pandas in Python, Alglib in C++ and C#. In [Pahlevanyan, 2014] the author made a comparative analysis of the R language with other statistical packages and demonstrated several significant advantages of R. Furthermore, for data analysis, large companies such as Google, Facebook, and Twitter use R.

R is a language for statistical computing, data manipulation, data mining and graphics. Robert Gentleman and Ross Ihaka started development of R in 1993, but it became popular last years, particularly for data scientists, as it contains a number of built-in functions for organizing data, running calculations on the information and creating elegant graphical representations of data sets. R provides a lot of different techniques for statistical linear and nonlinear modeling, time-series analysis, classification, clustering as well as graphical packages for creating high quality, and sophisticated, customized plots with very simple syntax. The capabilities of R can be extended through user-created modules. Modules are libraries developed in C++ that include specific functions for usage in certain applications. A core set of packages included with the installation of R, with more than 5,800 additional packages and 120,000 functions are free available for download [Venables et al, 2014]. R already had an extension for calculating various measures of Information Theory, but there was a need in creation of new module for estimation and computation of more complex formulas mentioned above.

To perform computations of complex formulas of Information Theory authors have developed new module for R, called Advanced Inftheo. Module Advanced Inftheo was developed in C++, because in R there is no multi-threading support, and there are restrictions of memory management. For high performance the main functions of module use parallel algorithms and tools of C++ to dynamically and optimally allocate memory. Moreover, in parallel algorithms there are often problems associated with the usage of same system resources by parallel running threads, to overcome such problems the module uses thread interaction techniques known as semaphores and critical sections. The module provides functionality for computation of the lower and upper bounds of E-achievable secret key rate, as well as functionality for computation of mutual information, conditional mutual information, Kullback - Leibler (divergence) distance and other quantities of Information Theory. It has an option to connect with cluster (using the library MPI) and execute all computational functions on cluster.

In this paper we demonstrate some results of computations that are done by Advanced Inftheo module. As an example we compute the lower and upper bounds of E-achievable secret key rate of the biometric generated secret key sharing system for various distributions. We give graphical representations of the computations to simplify the solutions in building of applications.

Biometric generated secret key sharing model

Let's define some conventions that are applied within this paper. Capital letters are used for random variables (RV) X, Y taking values in the finite alphabets $\mathcal{X}, \mathcal{Y}$ correspondingly. The cardinality of the alphabet $\mathcal{X}$ is denoted by $|\mathcal{X}|$. The notation $|a|^+$ is used for $\max(a, 0)$. Biometric generated secret key sharing model is represented in Figure 1.

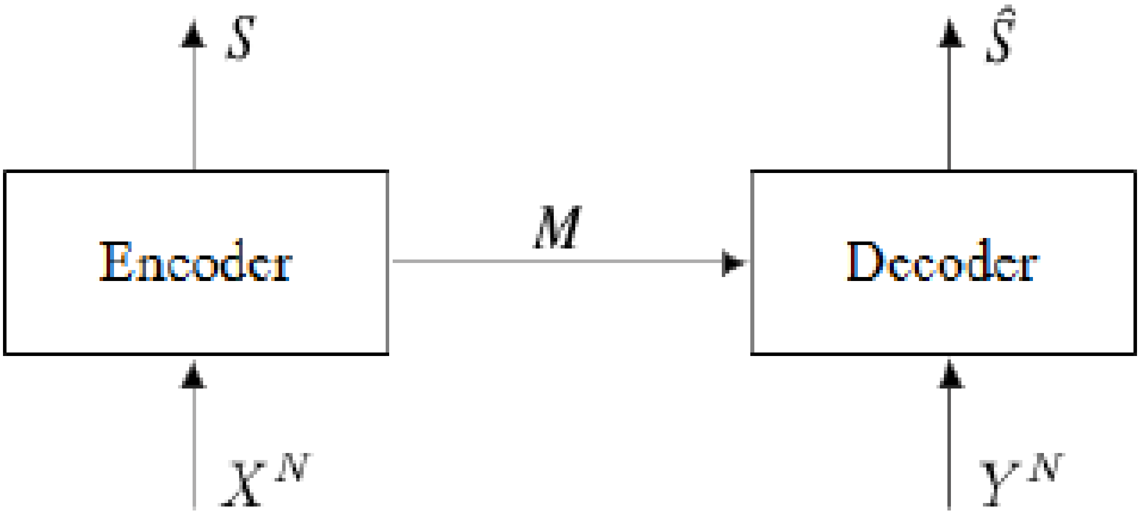


Figure 1. Biometric generated secret key sharing model

The model is based on biometric source with distribution $\{Q(x,y), x \in \mathcal{X}, y \in \mathcal{Y}\}$. This source produces
$\mathbf{x} \equiv x^N = (x_1, x_2, \dots, x_N)$ of N symbols from the finite alphabet $\mathcal{X}$ and a second sequence $\mathbf{y} \equiv y^N = (y_1, y_2, \dots, y_N)$ of N symbols from the finite alphabet $\mathcal{Y}$. The first sequence is called the enrollment sequence, and the second sequence the authentication sequence. Moreover, the second sequence Y^N is a noisy version of the first sequence X^N. Let us denote

$$Q(x,y) = Q_1(x)Q_2(y|x), x \in \mathcal{X}, y \in \mathcal{Y}.$$

We assume that

$$Q^N(\mathrm{x}, \mathrm{y}) = \prod_{n=1}^{N} Q(x_n, y_n).$$

Then consider an encoder that explores enrolment sequence X^N. From this sequence in biometric generated secret key sharing model the encoder generates a secret $S \in \{1,2,\dots,|S|\}$ and then a public helper data $\in \{1,2,\dots,|M|\}$. That means that

$$f(X^N) = (S, M),$$

where by $f(\cdot)$ we denote the encoder function. The helper data is sent to decoder. The decoder explores the authentication sequence Y^N and produces an estimate $\hat{S}$ of the secret S using the received helper data M, hence

$$g(Y^N, M) = \hat{S},$$

where by g$(\cdot,\cdot)$ we denote the decoder function. The channel between encoder and decoder is expected to be public. We assume that attacker has an access to that channel, so he can see all the public information but cannot modify it. The information outflow is described in terms of mutual information, and the size of the secret key in terms of entropy. Fingerprints and irises can be modeled as such biometric sources.

The important quantitative measures of a biometric secrecy system are reliability E, error probability, secret key rate, size of secret key and the information that the helper data leak on the biometric observation. That leak of biometric information is called privacy leakage. The privacy leakage should be small, to avoid the biometric data of an individual to become compromised. Moreover, the secret key length should be large to minimize the probability that the secret key is guessed. The goal of encoder and decoder is to produce a secret key as large as possible, that satisfies to condition $Pr\{S \neq \hat{S}\} \approx 0$, this means that probability that the estimated secret $\hat{S}$ is not equal to generated secret S is close to zero.

The definition for achievable secret key rate can be found in [Ignatenko-Willems, 2012]. We investigate the exponentially high reliability criterion in biometric generated secret key sharing systems. The new performance concept introduced in [Haroutunian-Pahlevanyan, 2014] for biometric generated secret key sharing system, takes into account a stronger requirement on authentication fault events with extremely small probability. In terms of practical applications an exponential decrease in error probability (namely, authentication fault events) is more desirable. Here is the definition of E-achievable secret key rate [Haroutunian-Pahlevanyan, 2014].

Definition. A secret key rate $R(E)$, for $R(E) \geq 0$, is called E-achievable if for all $\delta > 0, E > 0$ and N large enough, there exists a code such that

$$Pr\{S \neq \hat{S}\} \leq 2^{-N(E-\delta)},$$

$$\frac{1}{N}H(S) + \delta \geq \frac{1}{N}\log_2|S| \geq R(E) - \delta,$$

$$\frac{1}{N}I(S \wedge M) \leq \delta.$$

We shall use the following PD in the formulation of result:

$$Q_1 = \{Q_1(x), x \in \mathcal{X}\}, Q_2 = \{Q_2(y|x), y \in \mathcal{Y}, x \in \mathcal{X}\},$$

$$P_1 = \{P_1(x), x \in \mathcal{X}\}, P_2 = \{P_2(y|x), y \in \mathcal{Y}, x \in \mathcal{X}\},$$

$$Q = \{Q(x, y), x \in \mathcal{X}, y \in \mathcal{Y}\},$$

$$P = \{P(x, y), x \in \mathcal{X}, y \in \mathcal{Y}\}.$$

We refer to [Haroutunian, 2007; Haroutunian et al, 2008; Csiszar, 1998] and [Cover-Thomas, 2006] for notions of divergence $D(P||Q)$, mutual information $I_P(X \ \wedge Y)$, information-theoretic quantities.

The main result found in [Haroutunian-Pahlevanyan, 2014] is the theorem.

Theorem. For biometric generated secret key sharing model the largest E-achievable secret key rate $R(E)$ is lower bounded by

$$R_r(E) = \min_{P:D(P||Q)\le E} |I_P(X \wedge Y) + \ D(P||Q) - E|^+$$

And upper bounded by:

$$R_{sp}(E) = \min_{P:D(P||Q)\le E} I_P(X \wedge Y).$$

The proof of theorem can be found in [Haroutunian-Pahlevanyan, 2014].

Corollary. When $E \to 0$, the limits of lower and upper bounds coincide and equal the largest achievable secret key rate defined in [Ignatenko-Willems, 2012]

$$\lim_{E\to 0} R_r(E) = \lim_{E\to 0} R_{sp}(E) = I_Q(X \wedge Y).$$

Graphical representations of computations

We have performed computation of above formulas in the theorem using Advanced Inftheo module. In this section we present some graphical representations of results. Let's consider binary symmetric channel and denote d as a parameter. Let $Q(y|x)$ conditional distribution matrix be defined as

$$Q(1|1) = Q(0|0) = \ 1 - d,$$
$$Q(1|0) = Q(0|1) = d.$$

From the Figures 2, 3 and 4 of lower and upper bounds of E-achievable secret key rate we obtain the dependence of achievable secret key rate from reliability E for various d.

The computations have been done on machine with medium parameters (Intel Core 2 Duo 2 x 2.00GHz, with 2.5GB RAM). In worst case when only single thread is used inside Advanced Inftheo module to perform calculations the computation time, for instance when $d = 0.2$ and E changes from 0 to 1.2 with step 0.001, would be around 8.36 seconds. The above computation for same instance (when $d = 0.2, E = \overrightarrow{0; 1.2}$ with step 0.001) took 3.25 seconds on multithreaded environment of Advanced Inftheo module. As we can see, we have around 61% time gain. If we increase the precision of computations the time gain will be significant. It's worth mentioning that threads inside Advanced Inftheo are being allocated based on pc's technical capabilities and on range and step of E. A lot more time gain can be achieved if computations would be done on cluster and then forwarded to the user. The considered dependence of achievable secret key rate from reliability E will help to design practical biometric generated secret sharing systems.

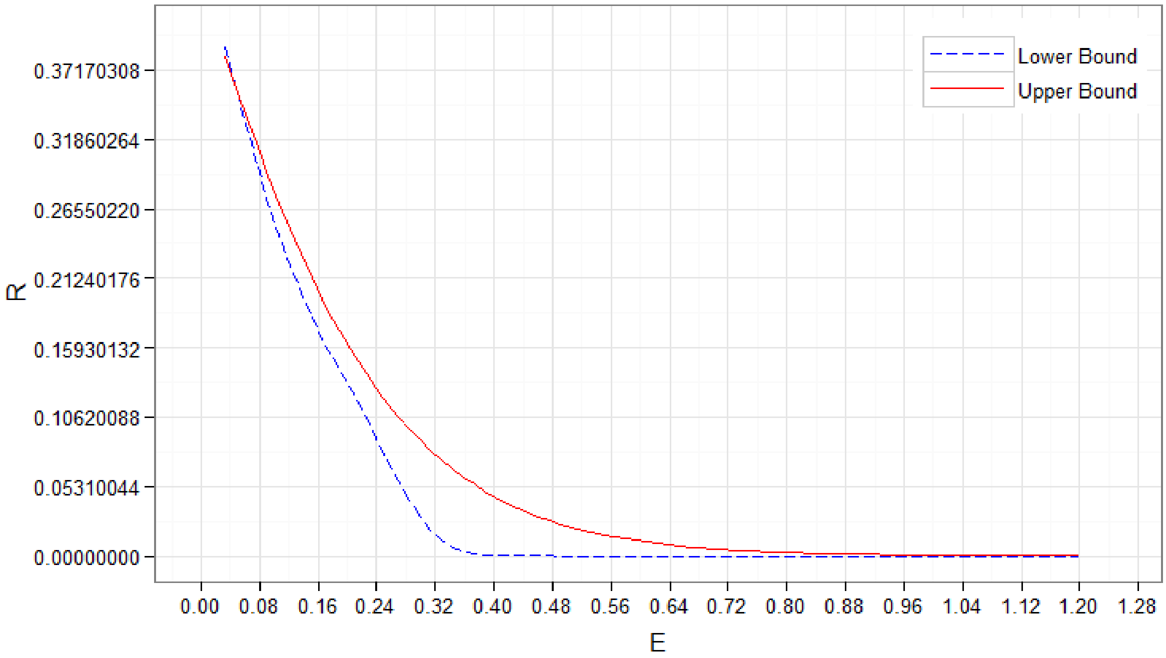


Figure 2. Bounds of of E-achievable secret key rate, when $d = 0.1$

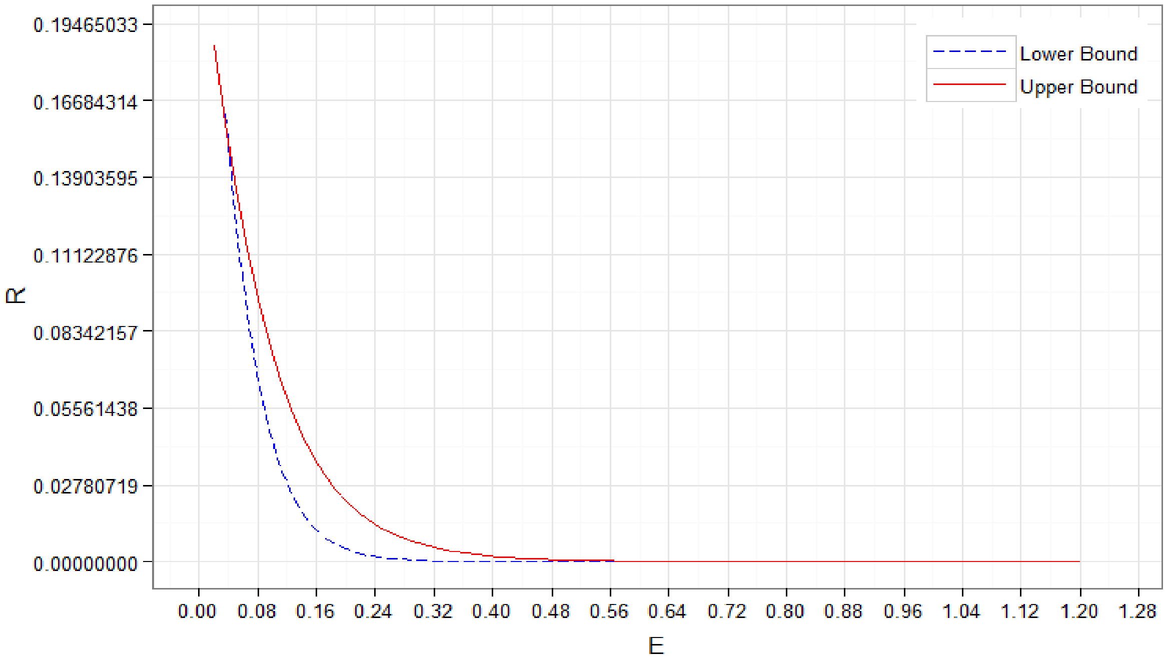


Figure 3. Bounds of of E-achievable secret key rate, when $d = 0.2$

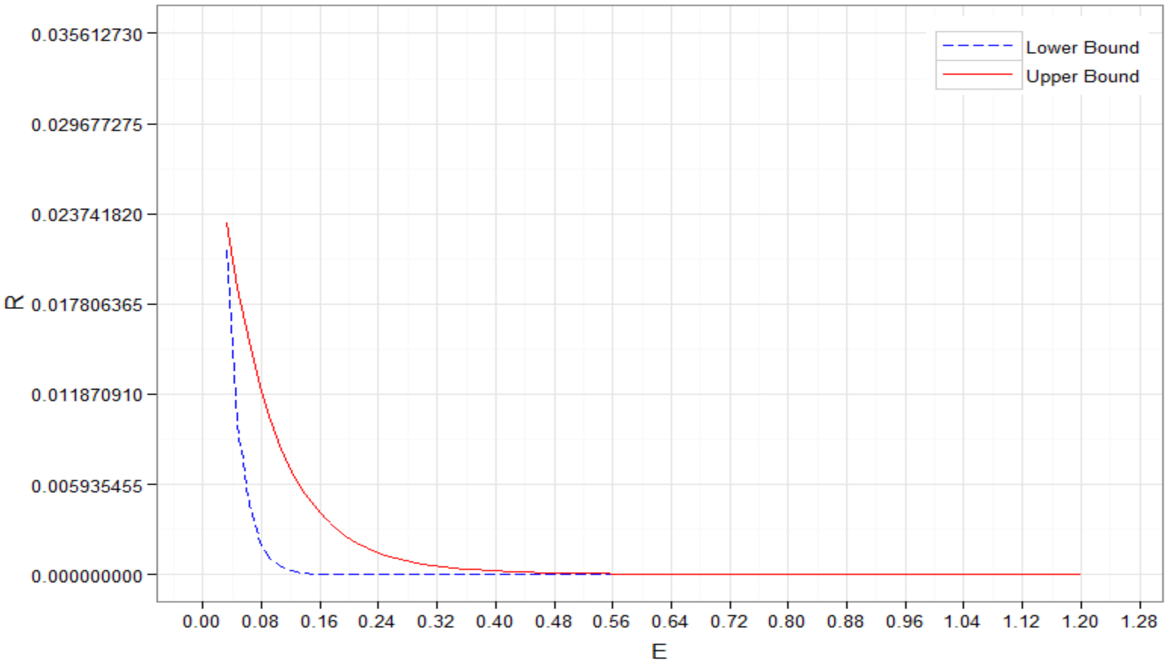


Figure 4. Bounds of of E-achievable secret key rate, when $d = 0.3$

Acknowledgements

The paper is published with partial support by the project ITHEA XXI of the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua).

Bibliography

[Chen-Vinck, 2011] Y. Chen and A. H. Vinck, "From password to biometrics: How far can we go," in 7th Asia-Europe Workshop on Concepts in Information theory, Boppard, Germany, pp. 1–8, 2011.

[Cover-Thomas, 2006] T. M. Cover and J. A. Thomas, Elements of Information Theory 2nd Edition. New York, NY, USA: Wiley-Interscience, 2006.

[Csiszar, 1998] I. Csiszar, "The method of types," IEEE Transactions on Information Theory, vol. 44, no. 6, pp. 2505–2523, 1998.

[Haroutunian et al, 2008] E. A. Haroutunian, M. E. Haroutunian, and A. N. Harutyunyan, "Reliability criteria in information theory and in statistical hypothesis testing," Foundations and Trends in Communications and Information Theory, vol. 4, no. 23, pp. 97–263, 2008.

[Haroutunian, 2007] E. Haroutunian, "On bounds for E - capacity of dmc," IEEE Transactions on Information Theory, vol. 53, no. 11, pp. 4210–4220, 2007.

[Haroutunian-Pahlevanyan, 2014] M.E. Haroutunian, N. S. Pahlevanyan "Information theoretical analysis of biometric secret key sharing model," Transactions of IIAP of NAS of RA, Mathematical Problems of Computer Science, vol.42, pp. 17-27, 2014.

[Ignatenko-Willems, 2012] T. Ignatenko and F. M. Willems, "Biometric security from an information-theoretical perspective," Foundations and Trends in Communications and Information Theory,vol. 7, no. 2-3, pp. 135-316, 2012.

[OSullivan-Schmid, 2002] J. A. OSullivan and N. A. Schmid, "Large deviations performance analysis for biometrics recognition." Proc. 40th Annual Allerton Conf. on Communication, Control, and Computing, pp. 1–10, Oct. 2002.

[Pahlevanyan, 2014] N. S. Pahlevanyan "Comparison of R language with other statistical tools," Armenian Mathematical Union Annual Session, p. 20, 2014.

[Venables et al, 2014] W. N. Venables, D. M. Smith and the R Core Team. "An Introduction to R", version 3.1.1, pp. 51-77, 2014.

[Willems et al, 2003] F. Willems, T. Kalker, J. Goseling, and J.-P. Linnartz, "On the capacity of a biometrical identification system," in Information Theory, 2003. Proceedings. IEEE International Symposium on Information Theory, Yokohama, Japan, 2003, p. 82.

Authors' Information

Mariam Haroutunian *– Professor, Doctor of Physical and Mathematical Sciences, Leading Researcher and Head of department for Information Theory and Cognitive Models at Institute for Informatics and Automation Problems, National Academy of Sciences, Armenia;*
e-mail: armar@ipia.sci.am

Major Fields of Scientific Research: Information theory, Probability theory and Mathematical Statistics, Information-theoretic aspects of information security.

Narek Pahlevanyan *– Studying for PhD; Gyumri State Pedagogical Institute, Armenia;*
e-mail: narek@ravcap.com

Major Fields of Scientific Research: Information theory, cryptography, cloud computing, machine learning, biometric secrecy systems.

NOVEL APPROACH TO CONTENT-BASED VIDEO INDEXING AND RETRIEVAL BY USING A MEASURE OF STRUCTURAL SIMILARITY OF FRAMES

David Asatryan, Manuk Zakaryan

***Abstract**: Extracting a small number of key-frames that can abstract the content of video is very important for efficient browsing and retrieval in video databases. Most research on video content involves automatically detecting the boundaries between camera shots. After shot boundary detection there is a need for shots indexing. In this paper, we present the gradient field based algorithm of shot detection and new method of key-frames determination. We provide a novel algorithm aimed to find a compact set of key-frames that can represent a video segment for a given degree of fidelity. The advantages of proposed approach are high performance of detection of the key-frames and linear speed of retrieval from the databases. Also we provide a new concept of the shot segment analysis and interpretation, using graphical and numerical methods.*

***Keywords**: shot detection, similarity measure, indexing, key-frame, content-based retrieval.*

***ACM Classification Keywords**: Image Processing and Computer Vision*

Introduction

Multimedia information indexing and retrieval are required to describe, store and organize multimedia information and to assist people in finding multimedia resources conveniently and quickly [Weiming, 2011, Lew, 2006]. Content-based video indexing and retrieval have a wide range of applications such as quick browsing of video folders, analysis of visual electronic commerce, remote instruction, digital museums, news event analysis, intelligent management of web videos etc.

The framework includes the following: 1) structure analysis to detect shot boundaries, extract key-frames, and segment scenes; 2) feature extraction from segmented video units (shots or scenes): these features include static features in key-frames, object features, motion features; 3) query: the video database is searched for the desired videos using the index and the video similarity measures; 4) video browsing and feedback.

A shot is a consecutive sequence of frames captured by a camera action that takes place between start and stop operations, which mark the shot boundaries. Methods for shot boundary detection usually first extract visual features from each frame, then measure similarities between frames using the extracted features, and, finally, detect shot boundaries between frames that are dissimilar. In the following, we discuss the main three steps in shot boundary detection: feature extraction, similarity measurement, and detection. The features used for shot boundary detection include color histogram or block color histogram, edge change ratio, motion vectors, together with more novel features such as scale invariant feature transform, corner points etc.

After the shot boundaries are identified, most of the existing works for video abstraction generally go through the following two steps: first, select the key-frames in each shot, and then cluster the similar shots based on the key-frames to construct the hierarchical or transition representation of video. Key-frames are a small set of images that can represent the visual content of a video. They can be used to compute the similarity between two video sequences, as well as to browse the video based on its content.

Following to [Truong, 2007], current approaches to extract key-frames are classified into six categories: sequential comparison-based, global comparison-based, reference frame-based, clustering based, curve simplification-based, and object/event-based.

1) **Sequential Comparison of Frames**: In these algorithms, frames subsequent to a previously extracted key-frame are sequentially compared with the key-frame until a frame which is very different from the key-frame is obtained. This frame is selected as the next key-frame. For instance, Zhang et al. [Zhang, 1997] used the color histogram difference between the current frame and the previous key-frame to extract key-frames. The merits of the sequential comparison-based algorithms include their simplicity, intuitiveness, low computational complexity.

The limitations of these algorithms include the following: a) key-frames represent local properties of the shot rather than the global properties, b) the irregular distribution and uncontrolled number of key-frames make these algorithms unsuitable for applications that need an even distribution or a fixed number of key-frames, c) a redundancies can occur when there are contents appearing repeatedly in the same shot.

2) **Global Comparison of Frames**: The algorithms based on global differences between frames in a shot distribute key-frames by minimizing a predefined objective function that depends on the application. In general, the objective function has one of the following four forms [Truong, 2007].

a) Even temporal variance: These algorithms select key-frames in a shot such that the shot segments, each of which is represented by a key-frame, have equal temporal variance;

b) Maximum coverage: These algorithms extract key-frames by maximizing their representation coverage, which is the number of frames that the key-frames can represent;

c) Minimum correlation: These algorithms extract key-frames to minimize the sum of correlations between key-frames (especially successive key-frames), making key-frames as uncorrelated with each other as possible;

d) Minimum reconstruction error: These algorithms extract key-frames to minimize the sum of the differences between each frame and its corresponding predicted frame reconstructed from the set of key-frames using interpolation.

3) **Reference Frame**: These algorithms generate a reference frame and then extract key-frames by comparing the frames in the shot with the reference frame. For instance, Ferman and Tekalp [Ferman, 2003] construct an alpha-trimmed average histogram describing the color distribution of the frames in a shot. Then, the distance between the histogram of each frame in the shot and the alpha-trimmed average histogram is calculated. Key-frames are located using the distribution of the distance curve. The merit of the reference frame-based algorithms is that they are easy to understand and implement. The limitation of these algorithms is that they depend on the reference frame.

4) **Clustering**: These algorithms cluster frames and then choose frames closest to the cluster centers as the key-frames. Girgensohn and Boreczky [Girgensohn, 2000] select key-frames using the complete link method of hierarchical agglomerative clustering in the color feature space. The merits of the clustering-based algorithms are that they can use generic clustering algorithms, and the global characteristics of a video can be reflected in extracted key-frames. The limitations are: first, they are dependent on the clustering results, but successful acquisition of semantic meaningful clusters is very difficult, and second, the sequential nature of the video cannot be naturally utilized.

5) **Curve Simplification**: These algorithms represent each frame in a shot as a point in the feature space. The points are linked in the sequential order to form a trajectory curve and then searched to find a set of points which best represent the shape of the curve. Calic and Izquierdo [Calic, 2002] generate the frame difference metrics by analyzing statistics of the macro block features extracted from the MPEG compressed stream. The merit of the curve simplification-based algorithms is that the sequential information is kept during the key-frame extraction. Their limitation is that optimization of the best representation of the curve has a high computational complexity.

6) **Objects/Events**: These algorithms [Kang, 2005] jointly consider key-frame extraction and object/event detection in order to ensure that the extracted key-frames contain information about objects or events. Calic and Thomas [Calic, 2004] use the positions of regions obtained using frame segmentation to extract key-frames where objects merge. The merit of the object/event-based

algorithms is that the extracted key-frames are semantically important, reflecting objects or the motion patterns of objects. The limitation of these algorithms is that object/event detection strongly relies on heuristic rules specified according to the application.

Once video indices are obtained, content-based video retrieval can be performed. On receiving a query, a similarity measure method is used, based on the indices, to search for the candidate videos in accordance with the query. The retrieval results are optimized by relevance feedback, etc. In the following, we review query types, similarity matching, and relevance feedback.

Similarity Measure and Algorithm for Shot Detection

Shot detection algorithm is usually based on consecutive determination of similarity of neighboring frames and detecting abrupt dissimilarities between them. When the level of similarity measure exceeds some predefined threshold t_c, then corresponding frame is considered as a cut frame. The quality of a decision rule depends on used similarity measure.

The measure which is applied in this paper is based on the structural properties of an image [Asatryan, 2009]. The mentioned measure is described below. We consider a model of image structure based on the set of edges which are determined by the gradient field of the image.

Let $\|G_H(m,n)\|$ and $\|G_V(m,n)\|$ (m = 0, 1, ..., M-1, n = 0,1,...,N-1) at a point (m, n) of an image be the horizontal and vertical gradients, determined by one of known gradient methods, and the matrix of gradient magnitude $\|\Delta(m,n)\|$, where

$$\Delta(m,n) = \sqrt{G_H^2(m,n) + G_V^2(m,n)} \tag{1}$$

We suppose that the gradient magnitude (1) is a random variable of two-parameter Weibull distribution density with parameters c > 0 and b > 0. As a measure of structural similarity of two images with probability distribution functions of gradient magnitude $f_1(x;b_1,c_1)$ and $f_2(x;b_2,c_2)$ accordingly, we accept

$$W^2 = \frac{\min(b_1,b_2)\min(c_1,c_2)}{\max(b_1,b_2)\max(c_1,c_2)},\ 0 < W^2 \le 1, \tag{2}$$

where the parameters $b_j, c_j, j = 1,2$ are statistically estimated by corresponding samples of gradient magnitudes of comparing images. This approach of images similarity assessment was successfully applied to different problems of image processing, see for example [Asatryan, 2009, Asatryan, 2010].

The algorithm for shot detection was described in our previous papers [Asatryan, 2014]. It was shown the advantages of algorithm against other algorithms based on mean-square deviation between images.

Key-frame Determination Algorithm

According to analysis of key-frame extraction algorithms, which have been described above, we can consider that the main difficulty of existing algorithms is the huge amount of comparisons of the content of frames inside each shot and between adjacent shots as well.

Our proposed algorithm of key-frame extraction have a big advantage in comparison with others, considering the fact that the content of each frame is characterized by only two parameters, which have already been determined during the shot detection procedure. The count of calculation needed to determine the similarity measure between any two shots is done using formula (2).

Key-frame detection is based on determining the parameters of the hypothetical frame, which are calculated as arithmetic mean of corresponding parameters b and c in the current shot. As there may not exists a frame with such parameters in the considered shot, we consider as a key-frame the frame which parameters b and c are the most near to parameter of hypothetical frame.

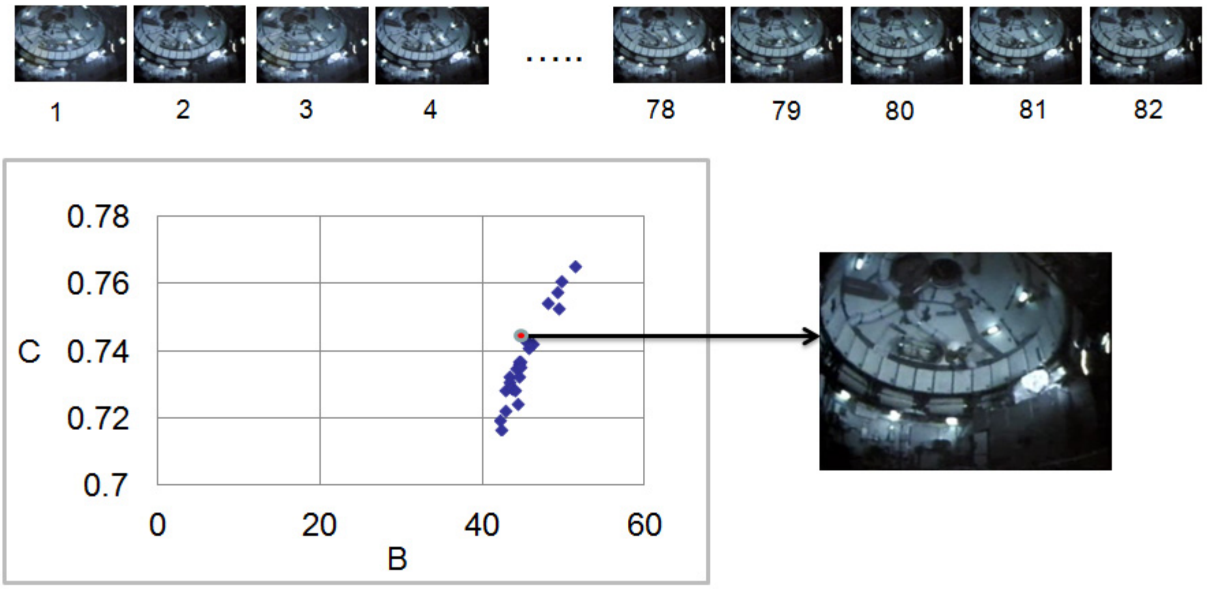


Figure 1. Illustration of key-frame

Figure 1 shows the process and the result of key-frame extraction from the given shot. The points in the graph corresponds to parameters b and c for each frame of the given shot above (the vertical axis shows the parameter c, the horizontal one - the parameter b).

It is also important to mention one more advantage of current approach. The thing is that the allocation of the points placed in the graph contains additional information about the content modifications of the frames inside the given shot. Therefore arises the problem of formal analysis of the parameters distribution in the (b, c) plot and fully interpretation of transitions inside the shot. Experimental results of key-frame extraction and corresponding formal analysis are given in the next section.

Results of Experiments

Described method of shot boundary detection was tested for various video sequences and some results have been given in our previous articles [Asatryan, 2014]. Here we graphically illustrate the results, which we got for key-frame extraction for exact video sequence, and also the analysis of the shot behavior using statistical regression technique for dependency between specified parameters.

The considered test video has 5 shots, which we determined by method described in section 2. Here in Figure 2 and Figure 3 the dependency graphs between b and c parameters as example for first and second shots are illustrated.

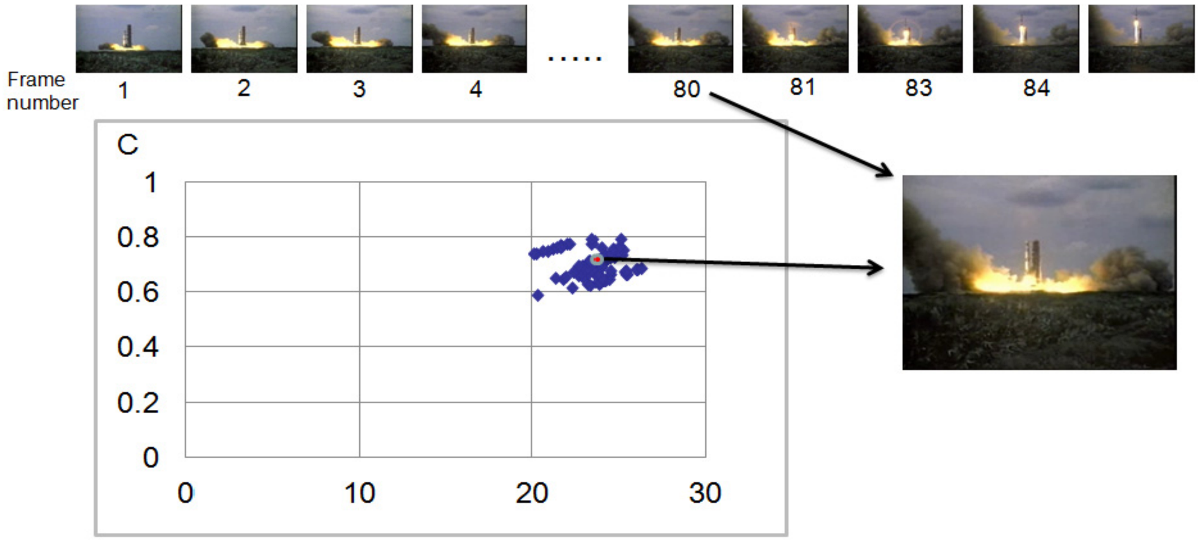


Figure 2. Distribution of points (b, c) for 1st shot

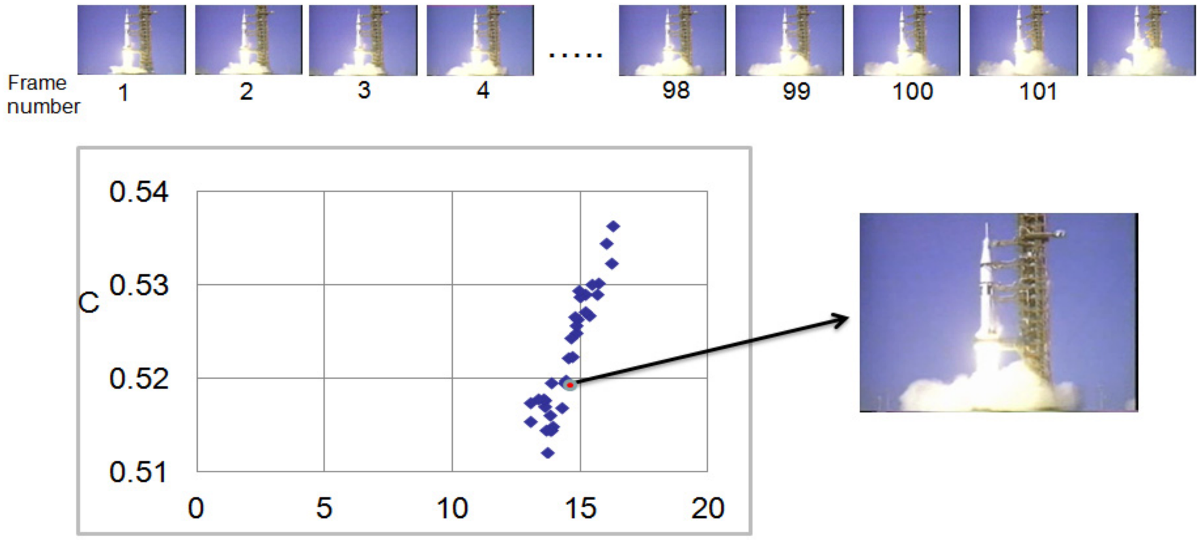


Figure 3. Distribution of points (b, c) for 2nd shot

In Table 1 are given the averaged values of parameters b and c for all frames in each shot, and also the maximum value of W^2 which are calculated during comparison of averaged values of parameters with parameters of all frames of corresponding shot. In the last column of Table 1 there are mentioned the frame numbers which are chose as key-frames.

Table 1. Key-frames which has been selected as maximum similar to frames of each shot

Shot number	$\bar{b}$	$\bar{c}$	$W^2_{\max}$	Number of chosen frame
1	23.44	0.7044	0.997	42
2	14.53	0.5223	0.999	101
3	28.25	0.5405	0.938	163
4	45.20	0.7435	0.991	219
5	118.77	0.9011	0.997	234

In the Figures 2 and 3 it is shown the distribution of points (b,c) for 1st and 2nd shots correspondingly. The visual analysis of the point's allocation in the (b, c) plot shows that the most convenient mathematical model of investigation for this kind of problem is the regression analysis. For simplicity we consider a linear regression analysis for the stochastic dependency between parameters b and c, wherein the absence of significant value of regression indicates the slight changeability of the frames content inside the shot, and on the contrary, the existence of significant regression may evidence about quit rapid changeability of frames.

In Figures 4 - 7 the experimental results of regression analysis for corresponding four shots is illustrated. For more visibility we also bring the determined linear regression graphs, and also the correlation coefficient R and F-ratio are given.

We would like to mention that the given results of formal analysis are considered as auxiliary characteristics, and therefore they should be accompanied with informative analysis of the video sequence itself.

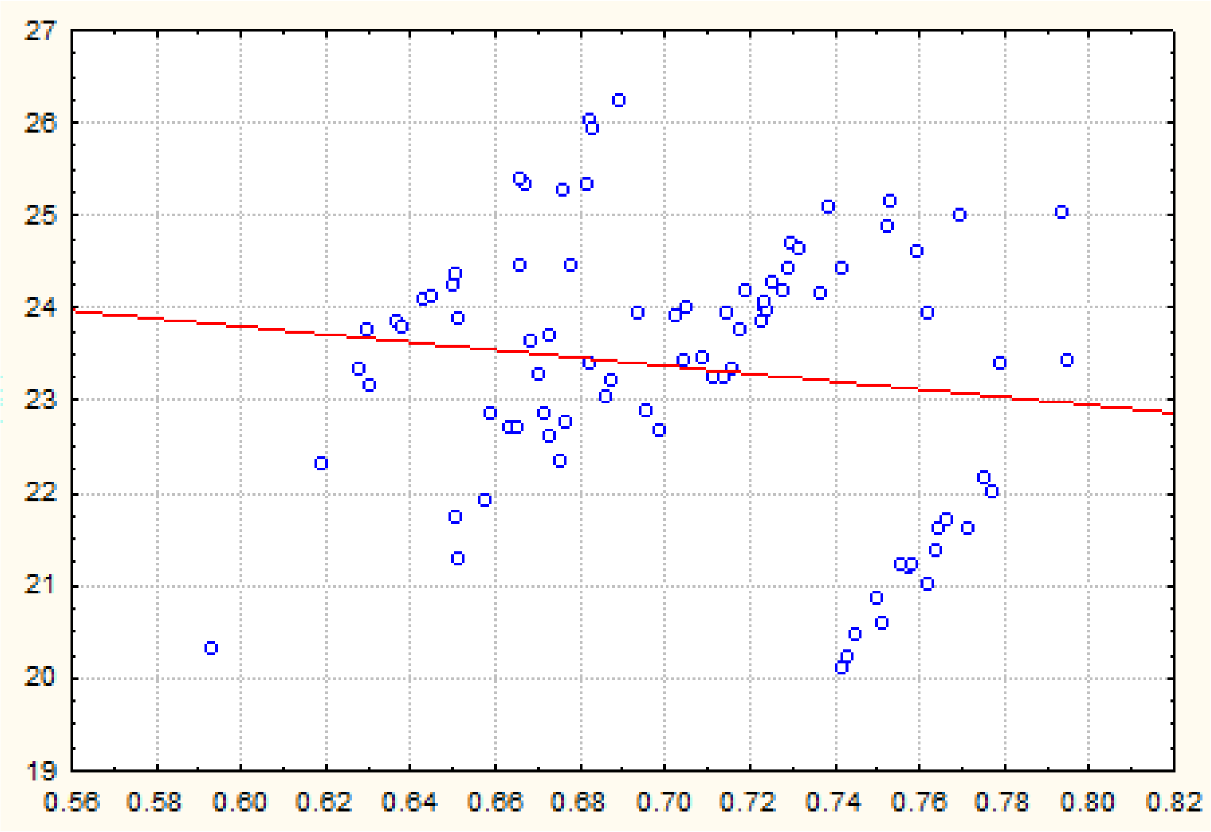


Figure 4. Regression analysis for 1st shot (R=0.138, F=1.61)

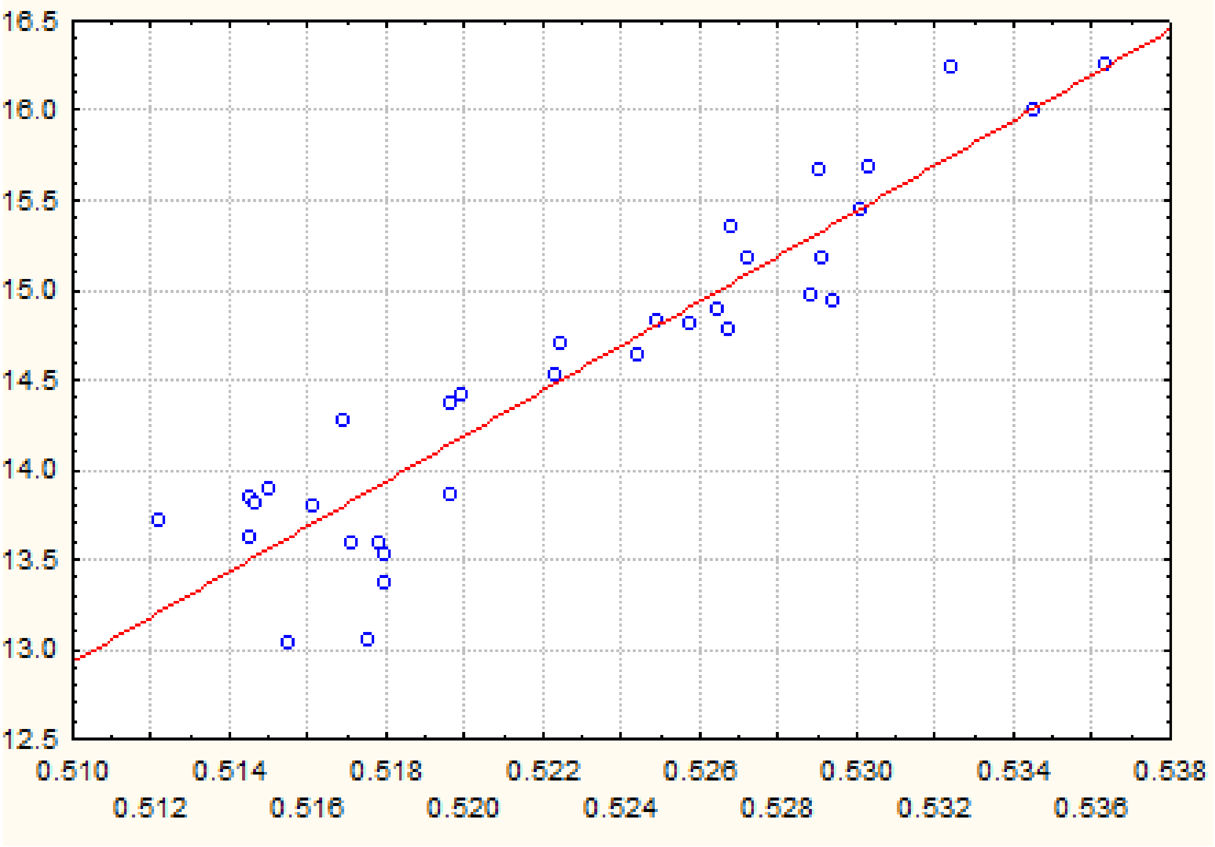


Figure 5. Regression analysis for 2nd shot (R=0.929, F=200.5)

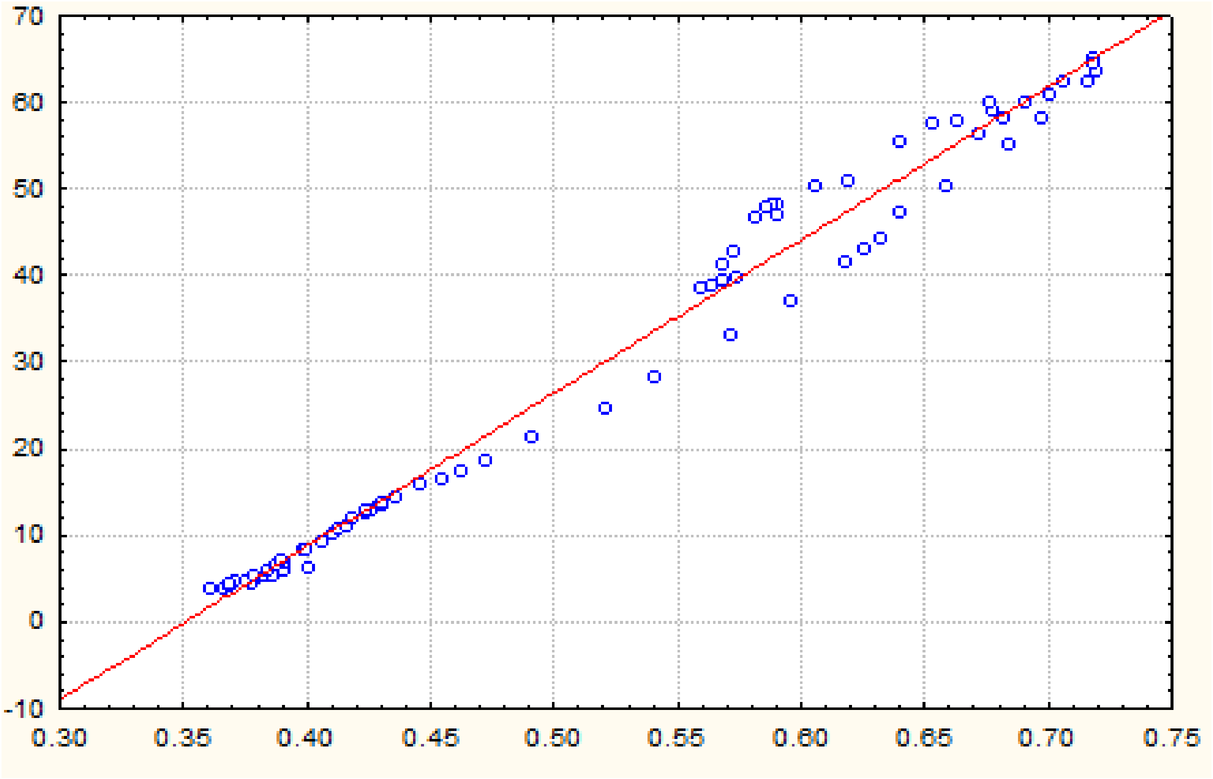


Figure 6. Regression analysis for 3rd shot (F=4559, R=0.991)

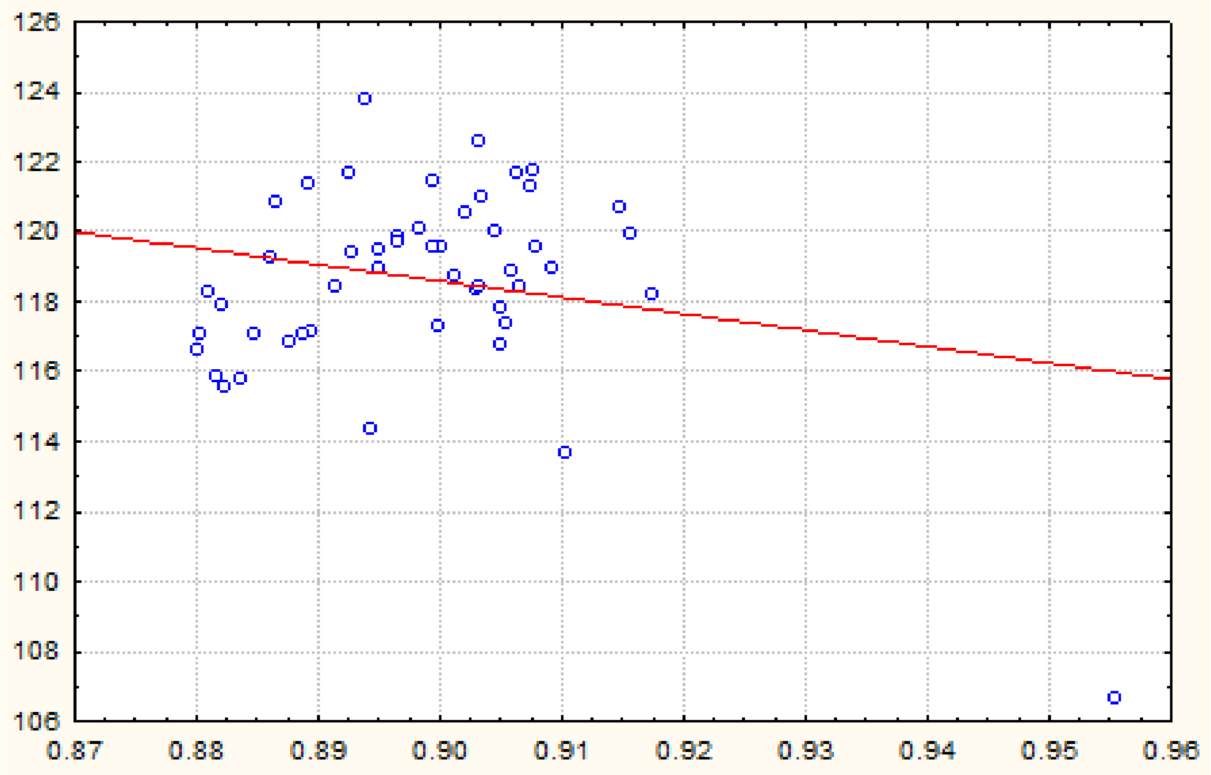


Figure 7. Regression analysis for 4rd shot (R=0.224, F=2.53)

Conclusion

In this paper, we have proposed a novel approach to content-based video indexing and retrieval by using a measure of structural similarity of frames. The first step of the proposed procedure is a shot detection; the second one is the key-frames determination for each shot. The key-frames are determined by using only the information obtained in the first step and its corresponding statistical analysis. The main advantages of proposed approach are the computational low time and keeping only two parameters for each shot which are very important for video retrieval tasks.

As a side effect during this investigation, we provide a new concept of shot segment behavior by analyzing the dependence between specified parameters.

Bibliography

[Asatryan, 2009] D. Asatryan, K. Egiazarian. “Quality Assessment Measure Based on ImageStructural Properties”. Proc. Of International Workshop on Local and Non-Local Approximation in Image Processing. Finland, Helsinki, pp. 70-73, 2009.

[Asatryan, 2010] David Asatryan, Karen Egiazarian, Vardan Kurkchiyan. Orientation Estimation with Applications to Image Analysis and Registration. International Journal “Information Theories and Applications”, Vol. 17, Number 4, pp. 303-311, 2010.

[Asatryan, 2014] D.G. Asatryan, M.K. Zakaryan. Improved Algorithm for Video Shot Detection. International Journal "Information Content and Processing", vol. 1, pp. 66-72, Number 1, 2014.

[Asatryan, 2014] D.G. Asatryan, M.K. Zakaryan. Method for Video Shot Detection and Separation. International Journal "Information Models and Analyses" Volume 3, pp. 247-251, Number 3, 2014.

[Calic, 2002] J. Calic and E. Izquierdo, “Efficient key-frame extraction and video analysis,” in Proc. Int. Conf. Inf. Technol.: Coding Comput., Apr. 2002, pp. 28–33.

[Calic, 2004] J. Calic and B. Thomas, “Spatial analysis in key-frame extraction using video segmentation,” in Proc. Workshop Image Anal. Multimedia Interactive Services, Lisbon, Portugal, Apr. 2004.

[Ferman, 2003] A.M. Ferman and A.M. Tekalp. “Two-stage hierarchical video summary extraction to match low-level user browsing preferences,” IEEE Trans. Multimedia, vol. 5, no. 2, pp. 244–256, Jun. 2003.

[Girgensohn, 2000] A. Girgensohn and J. Boreczky, “Time-constrained key-frame selection technique,” Multimedia Tools Appl., vol. 11, no. 3, pp. 347–358, 2000.

[Kang, 2005] H.W. Kang and X.S. Hua. “To learn representativeness of video frames,” in Proc. ACM Int. Conf.Multimedia, Singapore, 2005, pp. 423-426.

[Lew, 2006] M.S. Lew, N. Sebe, C. Djeraba, and R. Jain, “Content-based multimedia information retrieval: State of the art and challenges,” ACM Trans. Multimedia Comput., Commun. Appl., vol. 2, no. 1, pp. 1–19, Feb. 2006.

[Truong, 2007] B.T. Truong and S. Venkatesh, “Video abstraction: A systematic reviewand classification,” ACM Trans. Multimedia Comput., Commun. Appl.,vol. 3, no. 1, art. 3, pp. 1–37, Feb. 2007.

[Weiming, 2011] Weiming Hu, Senior Member, IEEE, Nianhua Xie, Li Li, Xianglin Zeng, and Stephen Maybank, “A Survey on Visual Content-Based Video. Indexing and Retrieval”. IEEE transactions on systems, man, and cybernetics-Part c: Applications and reviews, vol. 41, no. 6, November 2011, pp 797-813.

[Zhang, 1997] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar, “An integrated system for content-based video retrieval and browsing,” Pattern Recognit., vol. 30, no. 4, pp. 643–658, 1997

Authors' Information

David Asatryan *– Professor, Head of group of the Institute for Informatics and Automation Problems of NAS Armenia, 1, P.Sevaki Str., 0014, Yerevan, Armenia; e-mail: dasat@ipia.sci.am*

Major Fields of Scientific Research: Digital signal and image processing.

Manuk Zakaryan *– Ph.D. student at Russian – Armenian (Slavonic) University, Software Developer at EGS Armenia; e-mail: zakaryanmanuk@yahoo.com*

Major Fields of Scientific Research: Digital signal and image processing, Software developing.

SURVEY OF SOFTWARE FOR THE TEST QUALITY ANALYSIS

Varazdat Avetisyan

***Abstract**: A test method of checking and evaluating the knowledge is one of the most reliable and promising ways to increase educational process efficiency. Anyway, the test method can be efficient only if the test system is provided with a qualified test and test items. The developed test theory has complex mathematical – statistical apparatus which makes its usage impracticable. There are a lot of software packages to contribute to the pilot testing results' analysis and quality features' evaluation of a test. In Armenia such software is missing, which will give advices and suggestions on quality features' improvement. To develop a new system of quality analysis of the tests in Armenian language researches in the field of similar systems have been carried out. In this survey the peculiarities, advantages and disadvantages of the most applicable modern software are discovered.*

***Keywords**: Classical Test Theory (CTT), Item Response Theory (IRT), item difficulty, latent parameters*

***ACM Classification Keywords**: G.3 Statistical software.*

Introduction

A test method of checking and evaluating the knowledge is one of the most reliable and promising ways to increase educational process efficiency.

Testing method has a number of advantages over the other ways of knowledge assessment: objectivity and independence, provision of the same conditions for all examinees, possibility to assess many students' abilities and analyze the results as well as to test the knowledge on the respective course material [Avanesov, 1989].

Testing method efficiency depends on not only the application of objective and reliable technology but also the quality of applied test items [Chelishkova, 2002; Avanesov, 1989]. Based on this fact, the problem of providing the testing system with reliable and valid tests becomes very important and modern.

The qualitative analysis of tests is based on the test system of correlated features. This system expresses the quality of test items and the entity of algorithms and formulas which calculate and evaluate certain features. Test theories are concerned with these issues. Nowadays two theories of tests are known: Classical Test Theory (CTT) and mathematical-Item Response Theory (IRT) [Kim, 2007].

The founder of CTT is considered to be British famous psychologist Charles Edward Spearman (1863-1945). R. Cattell and D. Wechsler were his students. A. Anastasi, J. P. Guilford, P. Vernon, C. Burt, A. Jensen are considered to be his followers. Louis Guttman (1916-1987) has his great contribution in the development process of CTT: The classical theory of comprensive tests was first presented in H. Gulliksen's (1950.) work: The classical theory of tests is presented in L. Crocker J. Aligna's book (1986) [Crocker, 1986] in a modern way. In Russia one of the first introducers of this theory is V. Avanesov (1989) [Avanesov, 1989]. In the work by M. Chelishkova (2002) [Chelishkova, 2002] information about statistical methods of a test's quality assessment is presented.

By means of CTT application it is possible to calculate the reliability and criterion-related validity of the test, to evaluate the correspondence between test items and examinees' individual score, connection between test reliability and length, correlation between test items and so on [Kim, 2007].

IRT is foreseen to evaluate the examinees's latent parameteres as well as test items' parameteres [Avanesov, 2007]. In this theory the mathematical models are widely applied. IRT one parameter model is suggested by G. Rasch [Rasch, 1980]. The improved variants of IRT one parameter model are considered to be two and three parameter models suggested by Birnbaum [Birnbaum, 1968]. D. Andrich [Andrich, 2000] and B. Wright [Wright, 1979] have greatly contributed to IRT theory development.

IRT primarily focuses on the item-level information in contrast to the CTT's primary focus on test-level information. IRT main advantage is that items' difficulty coefficients' assessment does not depend on the selection of a certain group of examinees taking the test. Besides, the parameters of the examinee and test items are assessed through the same scale and the measurements of implemented test scores are turned into line measurement. As a result the qualitative data are analyzed by means of quantitative methods. It is possible to decide the test item information function through IRT.

In IRTthe measurements are implemented based on the following models [Wim,1997]:

Unidimensional Dichotomous Models:

- Normal Ogive Model;
- One-Parameter Logistic Model (Rasch Model);
- Two-Parameter Logistic Model;
- Three-Parameter Logistic Model;
- Nonparametric Model.

Unidimensional Polytomous Models:

- Partial Credit Model;
- Generalized Partial Credit Model;
- Rating Scale Model;

- Graded Response Model;
- Nominal Response Model (Nominal Categories Models).

Multidimensional Dichotomous Model;

Compensatory Three-Parameter Logistic Model.

IRT models are widely applied not only in the field of education but also psychology, medicine, sociology. As a result, computer programs of making analysis through the theory of IRT are widely known.

Thus, for statistical analysis of tests it is necessary to apply some systems, software packages which will make some test results' analysis and qualitative features' assessment based on one or two test theories.

To develop a quality examination system of tests in the Armenian language and for the Armenian market some research has been done in the field of similar systems. The research aim is to discover the peculiarities and advantages of the similar systems. The research of multifunctional and widely-applied tests' qualitative analysis' systems is presented.

Analysis of the most applicable modern software

A number of computer programs for simulating IRT data have been developed since the early 1970s. However, most of them were developed in the DOS environment (e.g. Bigsteps [Bigsteps, 1998], Facets [Facets, 1999], GENIRV [GENIRV, 1989], RESCEN [RESCEN, 1992]). As a result, these programs are limited today because of inherent problems in DOS: (1) slow performance speed (16-bit), (2) limited usable system resources, (3) incompatibility with recent 32-bit Windows-based OSs, and (4) not a user-friendly interface. Nowadays windows based IRT programs with user-friendly interface are widely used.

CITAS: CITAS [CITAS, 2015] (Classical Item and Test Analysis Spreadsheet) is a straightforward Excel Workbook that provides basic analysis of testing results based on classical test theory.The results are such indicators as mean and SD of scores, reliability, SEM, item P values, item point-biserials, and distractor analysis. By means of CITAS it is possible to analyze the results of the test which consists of not more than 50 examinees and 50 dichotomous items. It is also available in OpenOffice.org format and is an ideal tool for pedagogists to organize the test analysis. CITAS is for free and available at www.assess.com website.

Iteman 4: Iteman 4 [ITEMAN 4, 2015] is Windows based software which enables to receive detailed analysis report of tests and test items based on classical test theory. The aim of the report is to use received indicators to assess the qualitative characteristics of the tests.

Testing results and the data of test items are uploaded as .txt or .csv format files and the result of the analysis is received in RTF (word) format as a separate file, which has the possibility of editing.

Main Features are:

- Results' analysis is made in Word format;
- Analysis is made in the form of graphs;
- Ensures analysis in the levels of tests and test items variants;
- Implements calculation of the various coefficients of reliability.

A number of organizations operating in the field of testing use this program to provide their customers with testing results' analysis report. The program is not free and is provided based on several licenses. Its minimum price is $ 495. It is available at www.assess.com website. In the analyzed results, there is no limitation to examinees' number. Maximum test items number is 10.000. It's free of charge and its demonstration variant is available as well. This variant can be used to make an analysis of the tests with not more than 50 examinees and 50 test items.

Xcalibre 4: Xcalibre4 [Xcalibre 4, 2015] is a Windows based software of making test results' analysis based on IRT theory. While analysing the tests through Xcalibre 4, 4 dichotomous and 5 polytomous IRT models are used. Detailed and summarized analysis is given. This kind of analysis includes graphics and tables. This program is used very easily, has a point-and click interface, there is no need to work with programming codes. Testing results and test items' data are uploaded as .txt or .csv format files. Analysis report is received in rich text file (RTF) format, which means that there is no need to develop a report based on analysis results. Graphics include the item response function (IRF), the item information function (IIF), the test information function (TIF), and conditional standard error of measurement (CSEM), and numerous frequency distributions and so on.

Supported IRT Models are:

- 3-parameter dichotomous model (3PL);
- 2-parameter dichotomous model(2PL);
- 1-parameter dichotomous model (1PL);
- Rasch dichotomous model, scaled to items rather than people;
- Rasch rating scale model (RRSM, or RSM);
- Rasch partial credit model (RPCM, or PCM);
- Generalized rating scale model (GRSM);
- Generalized partial credit model (GPCM);
- Samejima's Graded response model (SGRM, or GRM).

One of the feautures peculiarities is that some data of the analysis are received in CSV format as well. This format is widely used in the program working with electronic tables. Its enables to analyze 1500 test items. Xcalibre 4 program is developed and is periodically updated by the organization of **Assessment Systems Corporation**. The program is not free and there are a number of licensed variants. Its minimum price is $ 495. It is available at www.assess.com website and its demonstration variant is available as well. This variant can be used to make an analysis of the tests with not more than 50 examinees and 50 test tasks.

Winsteps: Winsteps is a Windows-based software [Winstep, 2014] by means of which, based on IRT theory, the analysis is made due to Rasch model (Rasch Analysis) [Rasch, 2015].

It is developed by Benjamin Wright and John Michael Linacre [Linacre, 2004] at the University of Chicago in the 1980s. It is applied to analyze teaching tests, public surveys and rating scales. Item's analysis includes dichotomous, multiple-choice (MCQ), Rating Scales (RSM), Partial Credit (PCM) scales each of which has up to 255 categories. The analysis includes the data tables and many charts. The analysis is presented according to different categories for both items and examinees. There are a number of comprehensive guides both in printed and electronic versions. Some of them are available for downloading. The program correlates with Excel, R, SAS, SPSS, STATA, Txt files, which enables to upload the test results in the form of the above mentioned files. The program provides the receiving of about 48 kinds of tables, files and charts. The module of chart presentation has its separate functionality. By means of the chart the different features of the tests, test items, examinees are received. It is always developed by the group of Winsteps. The last version is Winsteps 3.81.0 which is issued in February, 2014. By means of the program the analysis of 1000000 examinees and 30.000 items can be made. The program requires some fee, the price is 149$. There is also the demonstration variant of the program MiniStep [Ministep, 2015], by means of which it is possible to make the analysis of not more than 75 examinees and 25 items. The program is presented at http://www.winsteps.com official website, where a number of descriptions, guides publications on Rasch model's measurements [Winstep, 2014].

Facets: Like Winsteps, Facets [Facets, 2014] was also developed by Mike Linacre. It is foreseen for applications of more complex (Unidimensional) Rasch measurements. It is a powerful and flexible program and includes all the models and possibilities available in Winsteps and provides a many-facet Rasch model [MFRM, 2015] which is not found in Winsteps. Many-facet Rasch model is applied when it is necessary to analyze the results of the experiments where heterogenic different tests and tasks are applied. Facets program enables to present the results of different items flexibly enough. By means of Facets, in the result of analysis, it is possible to receive the results' files with a respective scale. These files are inserted in Winsteps very easily. Facets also requires some fee, the price is 149$. It enables to analyze the data of about 1.000.000 examinees. The free demonstration version is called Minifac [Minifac, 2015] through which it is possible to analyze the responses of up to 2000 examinees. In Facets, during the calculations of much more complex models much time is needed. It is advisable to apply at least 1GB operative memory. The user guides and Minfac program are available at www.winsteps.com.

jMetrik: jMetrik is a free and open source computer program for psychometric analysis [jMetrik, 2015]. jMetrik is a pure Java application. It runs on Windows, Max OSX, and Linux operating systems. It features a user-friendly interface, integrated database, and a variety of statistical procedures and charts. The test results are inserted separately in .txt, .csv format files and their editing in the form of a table becomes possible. Based on this table the database is created. There is no limitation to the number of examinees and test items. It depends on the computer memory. The test is analyzed based on CTT and IRT theories. It provides a number of models like Rasch, 2PL, 3PL, PCM, RSM, GRM, GPCM as well as their combinations. A number of coefficients of test reliability, correlation, standard deviation of measurements are taken into account. The program provides charts concerning the tests, test items and examinees as well as the editing process of these charts. Analysis results are received in the program environment and can be downloaded in txt. and the charts in JPG, PNG formats. The program, sample files of test results, the guide are available at http://www.itemanalysis.com/ website. The book of Applied Measurement with jMetrik [Meyer, 2014] is available on a payable basis.

RUMM 2030: The RUMM 2030 [RUMM, 2015] is a Windows application which enables to make an analysis based on Rasch model. It is developed by David Andrich [Andrich, 2012], the author of RASCH - Andrich rating model as well as staff members of the laboratory in Australia, Perth. It is applied in case of the tests with both dichotomous and polytomous scales. It provides the receiving of necessary data and charts according to Rasch model. Particularly, this program is applicable in the educational process while studying the theory of measurements with Rasch model. There is a possibility to easily copy and insert the data taken from different electronic tables' formats /Excel or SPSS/. The disadvantage is the license price. It is 700$ for one academic year. It is necessary to participate in the courses for studying.

ACER ConQuest: ACER ConQuest [ACER, 2012] program combines the models of item response and latent regression. It is developed in Australian Council for Educational Research (ACER), by Margaret L. Wu**,** Ray J. Adams**,** M. R. Wilson. The program provides analysis and assessment by means of the following models:

- Rasch's Simple Logistic Model;
- Rating Scale Model;
- Partial Credit Model;
- Ordered Partition Model;
- Linear Logistic Test Model;
- Multifaceted Models;
- Generalized Unidimensional Models;

- Multidimensional Item Response Models;
- Latent Regression Models.

One of the important peculiarities is that it can assess by means of not only one-diamensional but also multi-diamensional IRT model [Briggs, 2003]. Input data may be uploaded immediately in SPSS format. The results may be received in Excel and SPSS formats. ConQuest has both graphical interface and consol interface. It provides the receiving of many colorful, information charts and maps. But because of the fact that here different measurement models and charts are applied, and there is no limitation to data amount, cost is much. ConQuest is run in Windows environment and costs $699. At http://www.acer.edu.au/conquest/ official website a short description of the program is available. The demonstration version is available for download.

IRTPRO: IRTPRO [IRTPRO, 2015] is a Windows application, which was developed by SSI (Scientific Software International). It is foreseen to make an IRT analysis. It provides an analysis through the following models:

- Parameter logistic (1PL);
- Two-parameter logistic (2PL);
- Three-parameter logistic (3PL);
- Graded;
- Generalized Partial Credit;
- Nominal;

In IRTPRO the data are inserted in the formats of .csv, .fixed, .txt, .xls. The inserted data are saved in the form of IRTPRO file (.ssig). In the environment of the program it is possible to receive different colorful charts. It has inclusive user guide, where IRT theory is described. The program requires some fee; some variants of license are available. The minimum price is $495. It is possible to download the demonstration variant of the program for 15 days duration. In this case the following limitations are found:

- The number of test items – 25;
- The number of examinees – 1000;
- The number of measurements- 3.

ConstructMap: ConstructMap program [ConstructMap, 2015] is developed in the centre of The Berkeley Evaluation and Assessment Research (BEAR). Two variants of the program are in use foreseen for Widows and MacOS operation systems. By means of the program it is possible to make an analysis due to both dichotomous assessment of Rasch model application and polytomous assessment of PartialCredit Model and Rating Scale Model’s application. Test results may be inserted in the forms of txt and Excel files. There are a number of ready examples in the program packages. It enables to receive different results of analysis, which include many IRT and CTT coefficients. It is possible to receive the features (TIF, ICC, etc.) of the test, test items and examinees in the form of charts. Analysis results may be saved in the form of .txt and the charts in the form of PNG or JPG files. The program, its description, user guide are available at the official website. ConstructMap is for free. Java environment needed for its installation.

Conclusion

The research shows that the main peculiarity of the qualitative examination of the tests are the list of supported IRT models, the number of examinees taking the tests and the number of items as well as tables, graphics of received data, formats of results' reports and so on.

Winsteps and Facets programs are distinguished due to the diversity and quantity, visual environment possibility to work with received graphics as well as existing multifaceted guides received in the result of analysis. The formats of received results are important as well. From this perspective Iteman 4, Xcalibre 4 programs are particularly distinguished. By means of these programs the test results are presented as full reports in rtf format. The report includes the tables with numbers as well as the graphics and information about the features. Such reports are much more available for the pedagogues. In case of other programs, the data are available in txt format and graphics- in png, jpg formats.

In the most part of the studied programs there is a limitation to the number of examinees and items of the test which is being analyzed. From this perspective jMetrik program is distinguished. In this program there is not such a limitation and based on the data in it a table is demonstrated in the visual environment. This table is kept in the database.

The diversity of supporting the formats of input and received files is an important feature as well. The files of the test results and items’ answers are input in txt sometimes in Excel or CSV formats. In a number of programs (Winsteps, Facets, RUMM 2030) there is a possibility to correlate with R, SAS, SPSS, STATA formats’ files.

During IRT analysis’s implementation the diversity of models supported by a given program is important. Almost in all IRT programs 1PL, 2PL, 3PL models are applied. Acer ConQuest program is distinguished as it enables to analyze through multivariate IRT. Xcalibre 4, WinSteps, ACER programs are known with their supported models’ diversity.

The studied programs are mainly based on Windows. The programs running in Linux, MacOS operating systems’ environments are very rare.

The most of programs are applied to analyze not only test results. This fact makes the programs’ functionality much more complex. Besides, the measured models become diverse.

The studied programs are mainly in English, require some fee, have different licensed packages. The demonstration variants of these programs are available as well. In the most part of such variants the limitations to the functional opportunities are found.

Some of the disadvantages of the modern widely-known programs may be emphasized:

- The programs making the analysis through IRT are multifunctional and are applied to assess different measurements. To make an analysis connected with testing process it is necessary to find, take out and sort the test models of the program, which is not an easy task at all;
- Available systems are mainly in English. Very rarely they can be in Russian as well;
- They have complex mathematical apparatus, which is used not only for making test analysis. For pedagogues it is very difficult to comprehend the different features of the apparatus;
- The test analysis results are mainly received in the form of different tables, which are kept in txt formats. The graphics, in their turn, are received in the form of separate files, in jpg or png formats. So, in order to receive a report in the form of one file it is necessary to make edits in different files and receive a new report, which is more applicable for the pedagogue;
- There is no detailed description of the quality features, which are being assessed. There are no methodological instructions on quality features' change;
- They mainly have multi-functionality and have appreciable values;

So, the issue of having such a system for the Armenian market comes forward. The new system requirements are to:

- Implement the test quality analysis based on CTT and IRT;
- Have the peculiarities which are typical to similar systems;
- Be in Armenian language;
- Have very simple and available interface convenient for pedagogues;
- Present results in the form of a report in one file;
- Give the detailed description of assessed quality features;
- Provide methodological instructions to change the value of this or that feature.

Acknowledgements

The paper is published with partial support by the project ITHEA XXI of the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua).

Bibliography

[ACER, 2012] ACER ConQuest 3.0.1 computer program-http://www.acer.edu.au/conquest

[Andrich, 2000] Andrich D., SheridanB., Lyne A. & LuoG. RUMM: A windows-based item analysis program employing Rasch unidimensional measurement models (Perth: Murdoch University), 2000.

[Andrich, 2012] Andrich D., Sheridan, B.S., & Luo, G. (2012). Rumm 2030: Rasch Unidimensional Measurement Models (software). RUMM Laboratory Perth, Western Australia.

[Avanesov, 1989] Avanesov V.S., The bases of the scientific organization of pedagogical control in the higher school (M., 1987)

[Avanesov, 2007] Avanesov V. S. Item Response Theory: The basic concepts and propositions. 2007. (Russian). (http://testolog.narod.ru/Theory67.html).

[Bigsteps, 1998] Bigsteps-DOS precursor to WINSTEPS. Final Version: 2.82, December 1998- Retrieved February 25, 2015, from http://www.winsteps.com/bigsteps.htm

[Birnbaum, 1968] Birnbaum A. Some Latent Trait Models and Their Use in Inferring an Examinee's Ability// F.M. Lord and M.R.Novick. Statistical Theories of Mental Test Scores. Readinf Mass.: Addison-Wesly, 1968. -Ch.17-20. -P.397-479.

[Briggs, 2003] Briggs D. C., & Wilson M. R. (2003). An Introduction to Multidimensional Measurement using Rasch Models. 4(1), 87-100.

[Chelishkova, 2002] Chelishkova M.B., Theory and practice of pedagogical tests constructing, 2002, Moscow: Logos

[CITAS, 2015] CITAS - free item analysis with classical test theory, Assessment Systems Corporation- http://www.assess.com/xcart/product.php?productid=407&cat=25&page=1

[ConstructMap, 2015] ConstructMap software- http://bearcenter.berkeley.edu/software/constructmap

[Crocker, 1986] Crocker Linda, Algina James. Introduction to Classical and Modern Test Theory. –New-York: Harcourt Brace Jovanovich, 1986.

[Facets, 1999] Facets: DOS precursor to the current Windows-based Facets. Final Version: 3.22, October 1999. Retrieved February 25, 2015, from http://www.winsteps.com/facdos.htm

[Facets, 2014] John M. Linacre. A User's Guide to FACETS Rasch-Model Computer Programs http://www.winsteps.com/facetman/index.htm

[GENIRV, 1989] Baker F. B. (1989). GENIRV: A program to generate item response vectors (Unpublished manuscript). Madison, WI: University of Wisconsin, Laboratory of Experimental Design.

[IRTPRO, 2015] IRTPRO 2.1 for Windows by Li Cai, David Thissen & Stephen du Toit- http://www.ssicentral.com/irt/

[Iteman 4, 2015] Iteman 4 - Test and item analysis software with classical test theory, Assessment Systems Corporation- http://www.assess.com/xcart/product.php?productid=417&cat=25&page=1

[jMetrik, 2015] Metrik-computer program for psychometric analysis. Retrieved February, 2015, from http://www.jmetrik.com/index.php.

[Kim, 2007] Kim V. S., Testing of educational achievements. Ussuriysk: USPI Publishing (2007).

[Linacre, 2004] From Microscale to Winsteps: 20 years of Rasch Software development, Linacre J.M. … Rasch Measurement Transactions, 2004, 17:4 p.958- http://www.rasch.org/rmt/rmt174g.htm

[Linacre, 2015] Winsteps and Facets Comparison. In Winsteps and Facets Rasch Software. Retrieved February, 2015, from http://www.winsteps.com/winfac.htm.

[Meyer, 2014] J. Patrick Meyer. Applied Measurement with Metrik. Routledge – 2014 http://www.routledge.com/books/details/9780415531979/

[MFRM, 2015] John Michael Linacre. Brief Explanation of the theory behind Many-Facets Rasch Measurement (MFRM). Help for Facets Rasch Measurement Software: http://www.winsteps.com/facetman/theory.htm

[Minifac, 2015] MINIFAC- Evaluation, Student and Demonstration (Demo) Version of FACETS (http://www.winsteps.com/minifac.htm)

[Ministep, 2015] Ministep-Evaluation, Student and Demonstration (Demo) Version of WINSTEPS (http://www.winsteps.com/ministep.htm)

[Rasch, 1980] Rasch G. Probabilistic Models for Some Intelligence and Attainment Tests.-Copenhagen, 1960, Danish Institute of Educational Research. (Expanded edition, Chicago, 1980, The University of Chicago Press).

[Rasch, 2015] Rasch Analysis - http://www.rasch-analysis.com/index.htm

[RESCEN, 1992] Muraki E. (1992). RESGEN: Item response generator [computer program]. Princeton, NJ: Educational Testing Service.

[RUMM, 2015] RUMM for analyzing assessment and attitude questionnaire data -http://www.rummlab.com/

[Wim, 1997] Wim J. van der Linden, Ronald K. Hambleton (1997). Handbook of modern item response theory. New York: Springer-Verlag.

[Winstep, 2014] John M. Linacre- A User's Guide to WINSTEPS® MINISTEP Rasch-Model Computer Programs-http://www.winsteps.com/winman/index.htm

[Wright, 1979] Wright B.D. & Stone M.H. Best Test Design. -Chicago, MESA PRESS, 1979. -222 p.

[Xcalibre 4, 2015] Xcalibre 4 - Software for IRT analysis and calibration, Assessment Systems Corporation-http://www.assess.com/xcart/product.php?productid=415&cat=22&page=1

Authors' Information

Varazdat Avetisyan*– Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia, PHD student, P.O. Box: P. Sevak 1, 0014 Yerevan, Armenia; e-mail: avetvarazdat@gmail.com*

Major Fields of Scientific Research: Test theory, e-learning, web programming, probability and statistics

PALM-VEIN AND FINGERPRINT BASED IMPROVED MULTIMODAL FUZZY VAULT SCHEME

Sergey Chidemyan

***Abstract**: Multi-biometric systems are kind of systems, where multiple templates from different biometric sources for the same user are stored. The authentication based on biometrics is a very good mechanism; however, such authentication technology needs large storage of biometric data, which should be protected. Fuzzy Vault is one of the most popular biometric encryption schemes, which aims to encode users' critical information in such a way that only the legitimate users are able to access it. In this paper, multimodal biometric template protection scheme is combined with biometrics, which results a high security. In particular, the approach of feature-level fusion for obtaining single multi-biometric template is described and construction of fuzzy vault scheme for the palm veins and fingerprints is presented. The proposed fuzzy vault scheme has been implemented and tested on the publicly available databases.*

***Keywords**: Palm Vein, Fingerprints, Fuzzy Vault, Template Protection, Multi-biometric systems*

***ACM Classification Keywords**: D.4.6 Security and Protection (K.6.5)*

Introduction

Often there are situations when we need to protect some critical information called key. People cannot remember cryptographically secure keys, so it is a good idea to use physiological features of a person (e.g. fingerprints, palm prints, palm veins, etc.) to provide an access to this kind of information. The authentication based on biometrics is a very good mechanism; however, such authentication technology needs large storage of biometric data, which appears to be the drawback, and also there is a risk of private data leakage and identity theft. It is a big issue, because biometric characteristics are inherent to a person and once lost, they would never be refreshed.

One of the most potentially harmful attacks on a multi-biometric system is against the biometric templates [Brindha & Natarajan, 2012].

In this work we consider multi-biometric systems, the kind of systems, where multiple templates from different biometric sources for the same user are stored. Since in multi-biometric systems multiple templates for the same user corresponding to the different biometric sources are stored template security is even more critical here.

Biometric template protection schemes that are combining cryptography with biometrics are considered to be a promising solution to issues above. Many famous biometric template protection schemes have been proposed such as fuzzy commitment scheme [Juels & Wattenberg, 1999], fuzzy vault scheme [Juels & Sudan, 2002] and fuzzy extractor [Dodis et al, 2008]. Among them the fuzzy vault scheme proposed by Juels and Sudan [Juels & Sudan, 2002] has become one of the most popular key-binding approaches, because it provides high security for biometric template protection. The scheme introduced by A. Juels and M. Wattenberg [Juels & Wattenberg, 1999] is not order invariant, which is the weakest point of the algorithm described in [Juels & Wattenberg, 1999], because the data extracted from the biometric template is not in the same order for the most types of biometrics. In contrast, the fuzzy vault scheme has a property of order invariance.

Multi-biometric fuzzy vault provides better identification and higher security compared to a unibiometric fuzzy vault. The only disadvantage here that the storage of multiple templates for the same user is required; however multi-biometric systems are more secure compared to their single biometric counterparts.

In this paper the approach of feature-level fusion for obtaining single multi-biometric template is described and construction of multi-biometric fuzzy vault scheme for the palm veins and fingerprints is presented.

Fuzzy Vault Scheme

As it was mentioned above the fuzzy vault scheme provides an effective and high security for biometric template protection [Juels & Sudan, 2002] and has a property of order invariance. So it suits the best for our purpose. Let us briefly introduce that scheme.

Let $\mathcal{F}$ be a finite field of size n and biometric template of the user can be written as follows: $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2, \dots , \mathrm{x}_\mathrm{s}),\ \mathrm{where}\ \forall \mathrm{i} = 1 \dots \mathrm{s}{:}\ \mathrm{x}_\mathrm{i} \in \mathcal{F}$. Let us denote the secret polynomial by *p(x)*. The degree of *p(x)* is k = s – t - 1, where t < s and coefficients of *p(x)*: $\mathrm{p}_\mathrm{j} \in \mathcal{F}$. Let $\mathrm{r} \in \{\mathrm{s}+1, \dots \mathrm{n}\}$

Locking algorithm

1. Having p(x) of degree k we evaluate it on the points of biometric. Let: $y_i = p(x_i)\ \forall i = 1 \dots s$.
2. Choose r – s distinct random points from $\mathcal{F}$ \ X so called chaff points: $x_{s+1}, \dots\ x_r$.
3. Choose $y_i \in \mathcal{F}$such that $\forall i = s+1 \dots r{:}\ y_i \neq p(x_i)$.
4. Construct vault: $V = \{(x_1, y_1)(x_2, y_2) \dots (x_r, y_r)\}$.

Unlocking algorithm

1. Let we have new biometric $X' = (x'_1, x'_2, \dots , x'_s),\ where\ \forall i = 1 \dots s{:}\ x'_i \in \mathcal{F}$

Having vault V, constructed by locking algorithm, the secret polynomial can be reconstructed if X′ has at least s – t common points with the original biometric X, using Lagrange interpolation.

Multi-biometric Fuzzy-Vault scheme construction

Let us briefly introduce the methods of feature extraction from palm-veins and fingerprints and construction of fuzzy vault scheme, based on these features.

✓ Extraction of biometric data from palm-veins

The vein pattern can be well represented by a number of critical points referred as minutiae points. The branching points and the ending points in the vein pattern skeleton image are the two types of critical points to be extracted. Ending points here are mainly ending points of vein skeleton curves that placed at the edge of region of interests (ROI) and resulted from the cropping of hand image while obtaining ROI. Although these ending points are not real ending points of vein on palm, they are taken because they contain geometrical information about the shape of the skeletons of the vein pattern. As for bifurcation points, they are the junction points of three curves. Figure 1 illustrates some of bifurcation and ending points on vein pattern's skeleton representation. Experiments on CASIA database [CASIA, 2015] show that we can extract on average 25 minutiae points from each vein pattern, including 10 bifurcation points and 15 ending points on average for each vein pattern. This quantity of minutiae points is quite enough for our purpose.

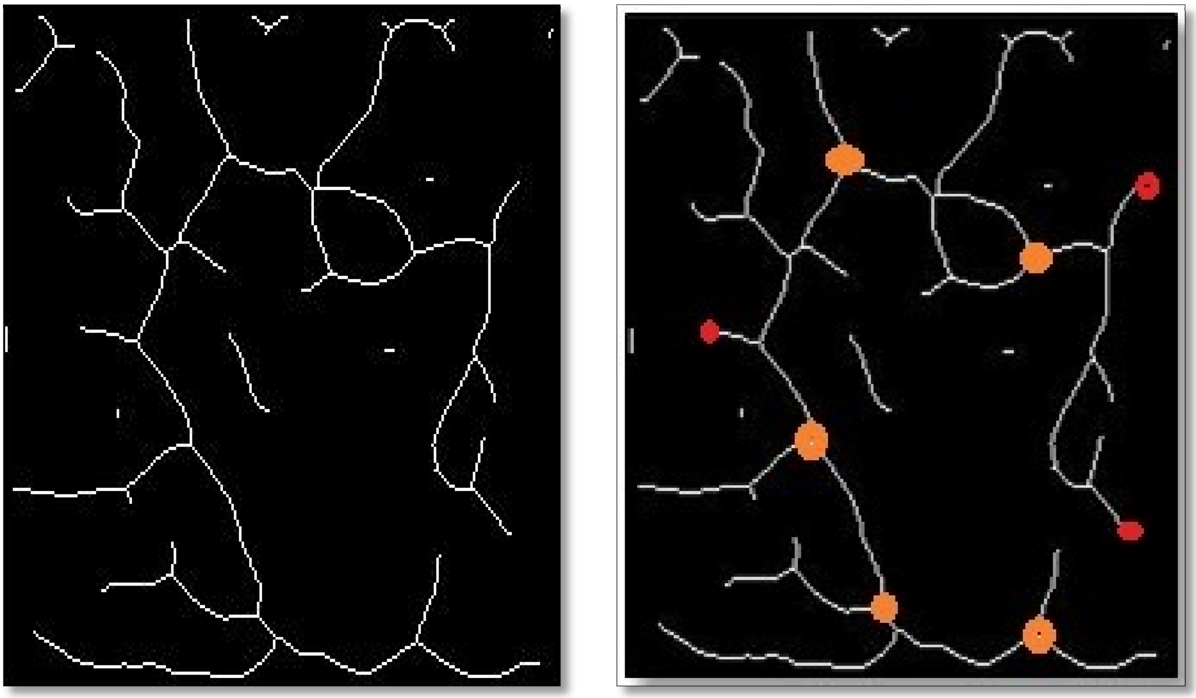


Figure 1. Some of bifurcation points and ending points are marked by red circle

- ✓ Extraction of biometric data from fingerprints

The most popular matching approach for fingerprint identification is usually based on lower-level features determined by singularities in finger ridge patterns called minutiae [Więcław, 2009]. In general, the two most prominent used features are ridge ending and ridge bifurcation (Figure 2). More complex fingerprint features can be expressed as a combination of these two basic features. Particularly, each detected minutiae m can be described by three parameters:

$$m = (x, y, \theta), \quad (1)$$

where (x,y) are coordinates of minutiae point, θ is minutiae direction typically obtained from local ridge orientation (Figure 3).

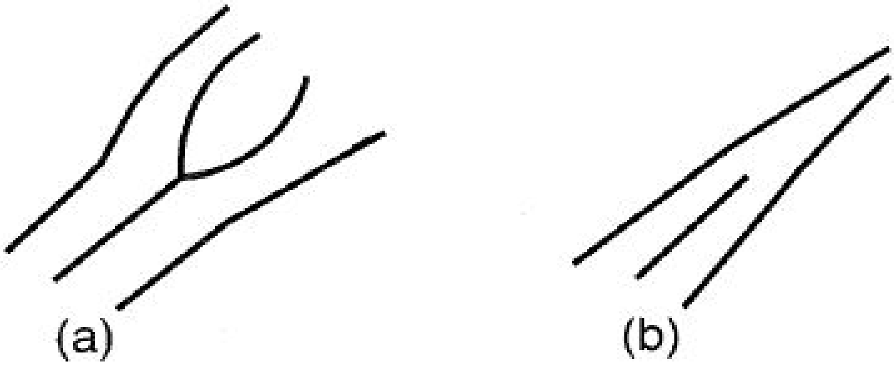


Figure 2. Features extracted from fingerprint: (a) ridge bifurcation; (b) ridge ending

The experiments show that we can extract on average 30 minutiae points from each imprint of fingerprint.

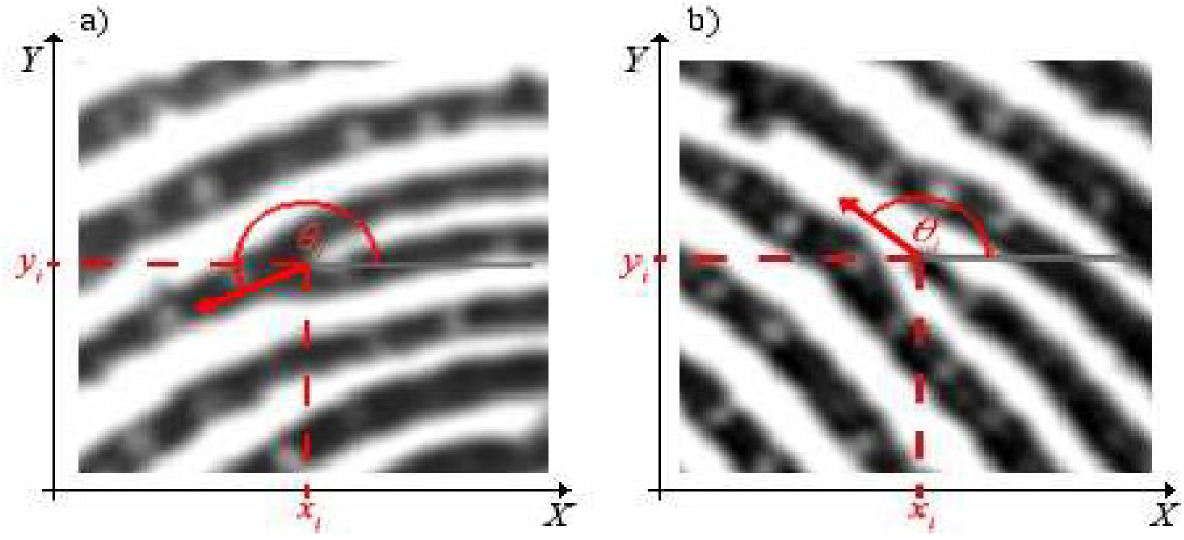


Figure3. θ component of minutiae feature for a) ridge bifurcation, b) ridge ending

- ✓ Implementation of fuzzy vault based on the extracted data

As it was mentioned above we consider multi-biometric systems, where multiple templates from different biometric sources for the same user are stored. We introduce the feature-level fusion for palm-veins and fingerprints to obtain a single multi-biometric template.

Let $\mathrm{X_F}$ and $\mathrm{X_{PV}}$ are sets of feature points extracted from fingerprints and palm-veins respectively. Here all elements $\mathrm{x_F^i} \in \mathrm{X_F}$ are in Galois fields GF (2^{16}). Note that 5 bits are taken from x coordinate, 5 bits from y coordinate and 6 bits from θ. The elements $\mathrm{x_{PV}^i} \in \mathrm{X_{PV}}$ are also in Galois fields GF (2^{16}). Here 8 bits are taken from coordinate x of palm-vein minutiae point and 8 bit from coordinate y. The union X, of the two sets $\mathrm{X_F}$ and $\mathrm{X_{PV}}$ is formed such that the Hamming distance between any two elements in the union is greater than or equal to 2 [Nandakumar, 2008].

Encoding

The biometric template X, discussed above, is constructed using 16 bits from minutiae points. In the current implementation, a randomly generated secret S is 224-bit random key, which is used for constructing the secret polynomial p(x).

For each degree of the polynomial n in range from 8 to 14 and the number of the minutiae points in X is s = 55, the chaff points were taken r-s = 550.

Decoding

Here, the user tries to unlock the vault V using the query minutiae. Assume we have s query minutiae $(\mathrm{X'})$ and $\mathrm{u'_1, u'_2, \dots ,\ u'_s}$ are the points to be used in polynomial reconstruction. These points are found by comparing $\mathrm{u_i}$, i = 1, 2… s. with the values of the vault V, namely $v_1, v_2, \dots, v_r$. If any u_i is equal to $v_1, v_2, \dots, v_r$, the corresponding vault point is added to the list of points to be used. Assume that this list has m points, where m ≤ r. Now, for decoding a n-degree polynomial, n + 1 unique projections are necessary. We have to find all possible $\mathrm{C_m^{n+1}}$ combinations of n + 1 points, among the list with size m. For each of these combinations, we construct the Lagrange interpolating polynomial.

If the query minutiae list $(\mathrm{X'})$ overlaps with template minutiae list (X) in at least (n+1) points, for some combinations, the correct secret will be decoded. This indicates the desired outcome when query and template multi-biometric data are from the same user.

The results of the experiments

There are six imprints of the same palm-vein and one of them was used to enroll the user. The one imprint of fingerprint is also used to enroll the same user.

Let us check the probability that attacker can decode the secret using all possible combinations of points in our vault. In case we have a secret of a size 224 bit, 55 genuine minutiae points and 550 chaff points, the probability that a random combination of points decodes the secret is:

$$C_{55}^{15} \Big/ C_{605}^{15} \approx 2^{-55}$$

This gives approximately 55 bits of security.

Below the results of the system performance tests on virtual database, which generated from the palm-veins and fingerprints databases, are attached (Figure 4).

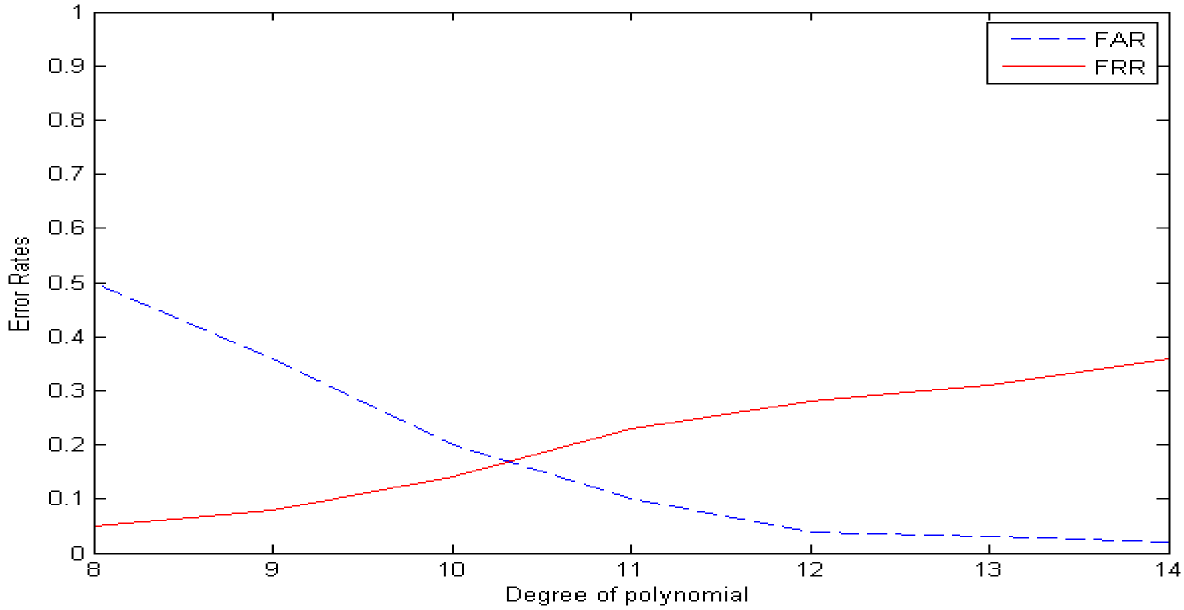


Figure 4. Error rate curves of palm-vein and fingerprint based multimodal fuzzy vault scheme

Conclusion

In this paper we have presented the results of actual implementation of the multimodal fuzzy vault using palm-vein and fingerprint biometric data. In particular, the fusion mechanism of palm-vein and fingerprint minutiae sets is proposed. Experiments show that there is 55 points on average after described feature-level fusion has been applied. This quantity of points is enough for 224-bit key generation and for the practical accuracy of the system (FAR < 0.01). In the last section proposed multi-biometric scheme is discussed in term of security bits and how it follows from experiments our scheme guarantees high security.

Bibliography

[Brindha & Natarajan, 2012] E. Brindha, A. M. Natarajan, “Multi-Modal Biometric Template Security: Fingerprint and Palmprint Based Fuzzy Vault”, Journal of Biometrics and Biostatistics, Vol.5, Issue 4, Aug 2012, pp. 1 - 6

[CASIA, 2015] CASIA MS Palmprint V1 Database, Available: http://biometrics.idealtest.org/dbDetailForUser.do?id=5.

[Dodis et al, 2008] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, “Fuzzy extractors: how to generate strong keys from biometrics and other noisy data,” SIAM Journal on Computing, vol. 38, no. 1, 2008, pp. 97 – 139

[Juels & Sudan, 2002] A. Juels and M. Sudan, “A fuzzy vault scheme,” in Proceedings of the IEEE International Symposium on Information Theory, July 2002, 408 p.

[Juels & Wattenberg, 1999] A. Juels and M. Wattenberg, “Fuzzy commitment scheme,” in Proceedings of the 1999 6th ACM Conference on Computer and Communications Security (ACM CCS ’99), November 1999, pp. 28 – 36

[Nandakumar, 2008] K. Nandakumar, “Multibiometric Systems: Fusion Strategies and Template Security”, PhD thesis, Department of Computer Science and Engineering, Michigan State University, January 2008

[Więcław, 2009] Ł. Więcław, “A Minutiae-Based Matching Algorithms In Fingerprint Recognition systems”, Journal Of Medical Informatics & Technologies, Vol. 13, 2009, pp. 65 - 72

Authors' Information

Sergey S. Chidemyan *– Russian – Armenian (Slavonic) University; e-mail: serchch@gmail.com*

TABLE OF CONTENTS