ACTA UNIVERSITATIS AGRICULTURAE ET SILVICULTURAE MENDELIANAE BRUNENSIS

Volume LXI 233 Number 7, 2013
http://dx.doi.org/10.11118/actaun201361072105

ANALYSIS OF SOUND DATA STREAMED OVER THE NETWORK

Jiří Fejfar, Jiří Šťastný, Martin Pokorný, Jiří Balej, Petr Zach

Received: April 11, 2013

Abstract

FEJFAR JIŘÍ ŠŤASTNÝ JIŘÍ, POKORNÝ MARTIN, BALEJ JIŘÍ, ZACH PETR: *Analysis of sound data streamed over the network*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 2013, LXI, No. 7, pp. 2105–2110

In this paper we inspect a difference between original sound recording and signal captured after streaming this original recording over a network loaded with a heavy traffic. There are several kinds of failures occurring in the captured recording caused by network congestion. We try to find a method how to evaluate correctness of streamed audio. Usually there are metrics based on a human perception of a signal such as "signal is clear, without audible failures", "signal is having some failures but it is understandable", or "signal is inarticulate". These approaches need to be statistically evaluated on a broad set of respondents, which is time and resource consuming. We try to propose some metrics based on signal properties allowing us to compare the original and captured recording. We use algorithm called Dynamic Time Warping (Müller, 2007) commonly used for time series comparison in this paper. Some other time series exploration approaches can be found in (Fejfar, 2011) and (Fejfar, 2012).
The data was acquired in our network laboratory simulating network traffic by downloading files, streaming audio and video simultaneously. Our former experiment inspected Quality of Service (QoS) and its impact on failures of received audio data stream. This experiment is focused on the comparison of sound recordings rather than network mechanism.
We focus, in this paper, on a real time audio stream such as a telephone call, where it is not possible to stream audio in advance to a "pool". Instead it is necessary to achieve as small delay as possible (between speaker voice recording and listener voice replay). We are using RTP protocol for streaming audio.

Dynamic Time Warping, sound signal processing

There are two scenarios when transmitting audio data through the network: streaming audio from a server to the user without a need for a real-time response but with a need for a good quality. Another case is an audio streaming with a need for the real-time response.

The example of the first scenario can be given by a user listening to a favourite Internet broadcast. It is not important for the user listen to sound events (music / words) at the same time as they are recorded (in case of live broadcasting). It is possible to hear such a broadcast with a 10s delay or more. The radio time sound signals might be inaccurate, but they are not used nowadays in the Internet radio broadcasting, as there are different methods for clock synchronization. In this case we can use a long buffer when transmitting audio, which overcomes network bandwidth bottlenecks. When we are transmitting audio recorded earlier, we can send it to the user in advance, so it is not predisposed to suffer from network traffic dynamic changes.

In the second case, there is a completely different scenario, when we transmit voice of users talking to each other. Delay about 150 ms (sometimes 250 ms) will be disturbing in that case and when the delay is longer the communication turns to be inarticulate. That's the reason why we deploy different QoS mechanisms to give the voice traffic precedence over the data and video traffic.

We are comparing original recordings (broadcasting from computer A) with its counterparts (recorded on computer B) transmitted over the network utilizing QoS or without QoS being deployed. The network is in both of the cases congested with traffic generated by voice (VLC), video (RTSP) and data streams (wget and web server) (Zach, 2012).

METHODS AND RESOURCES

Data

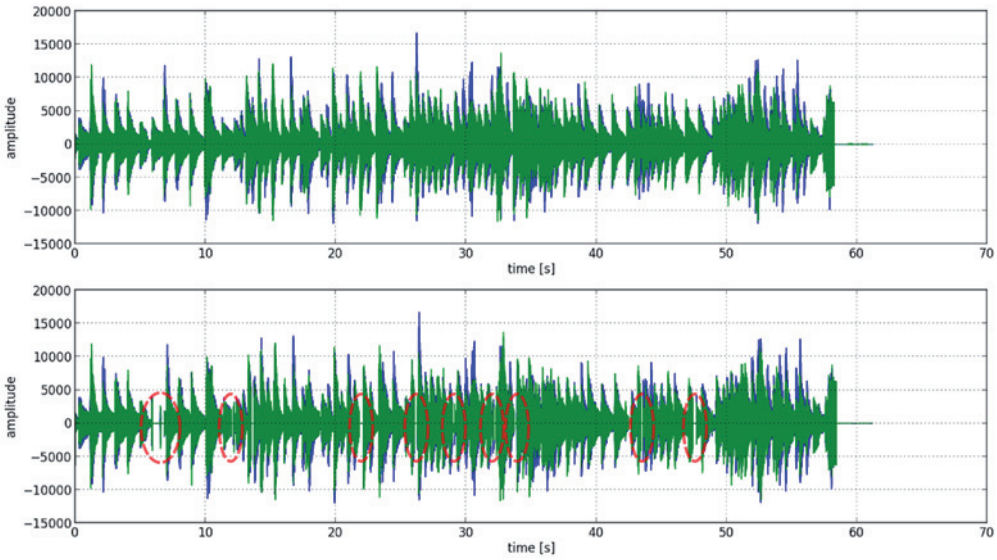
In the first part of the experiment, we had to select a signal which will be sent over the network. Although this experiment is concerned primarily with the voice traffic, we used a recording containing music. The reason is that the voice streams consist of silent parts, where it is difficult to detect errors. We made a 1 minute long clip of a permanent (without silent parts) music performance from this recording. The first and the last second of the recording was substituted with a reference signal, formed with a simple sinusoid signal with a constant defined amplitude and frequency, which enables proper alignment and comparison of original and captured recordings. We aligned starting reference signals in the captured and original recordings, so there are no delays caused by inaccurate synchronization of the broadcasting and recording start time. There will be only delays caused by network behaviour during transmission in the experiment. We can see a waveform (simply plotting amplitude of the signal on the time axis) of the original and captured recording in Fig. 1.

Errors in the captured recording are marked with red circles in the lower part of the figure. The recording is two-channel stereo, one channel is plotted with green colour and the other one with blue colour. In Fig. 2 we can see a detail between seconds 4 and 10. There is one of the bigger errors caused by the network congestion.

Experiment settings

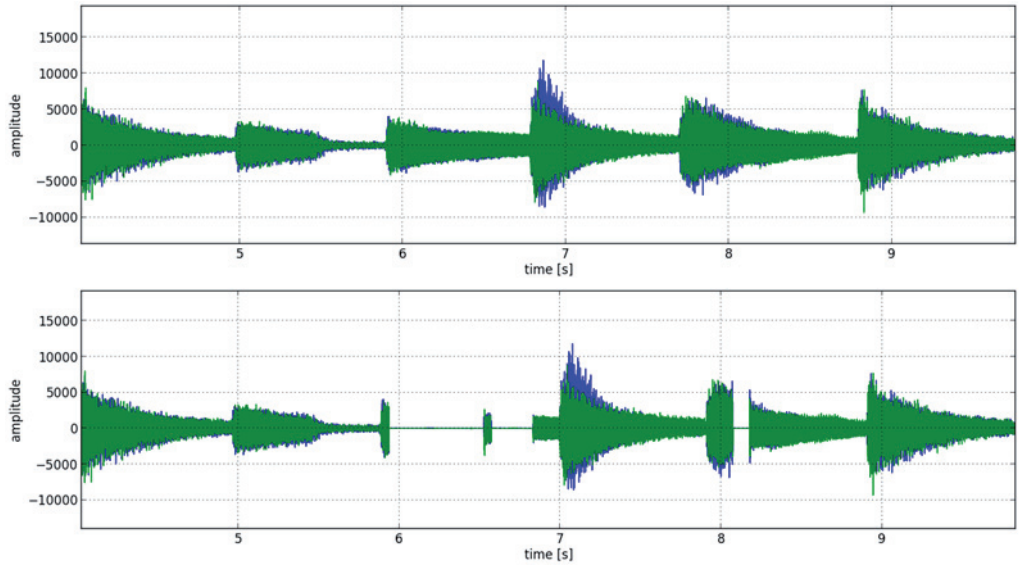
Broadcasting of the recording through the network was performed with VLC[1] player on the both of the communicating hosts, the server and client side, using RTP. The buffer of the VLC player (on the client side) was set to a small value (150 ms). The sound on the client side was sent to the sound card ensuring there will be no other buffers engaged while saving sound to the client's hard drive. The sound card output was monitored by the audacity[2] and the captured sound was saved to a .wav file.

We can see the experiment layout in the Fig. 3. We obtained feature vectors both of the original and captured recordings by dividing these recordings into time windows (Černocký, 2009) with a constant length of (440 samples = 0.01 s) as we can see in a left part of Fig. 3. As the network failure results in a silent section of the recording we used standard deviation of the signal in a time window as an indicator of the sound volume. We also tried to use mfcc (Černocký, 2009) resulting in multidimensional feature vectors, which were turned into the single dimension using Self Organizing Map (Kohonen, 2001), (Škorpil, 2008), which belongs to hexagonal diagram in a lower middle part of Fig. 3. Self Organizing Maps was used also in analysis of data in (Balogh, 2010). Single dimensional feature vectors (upper middle

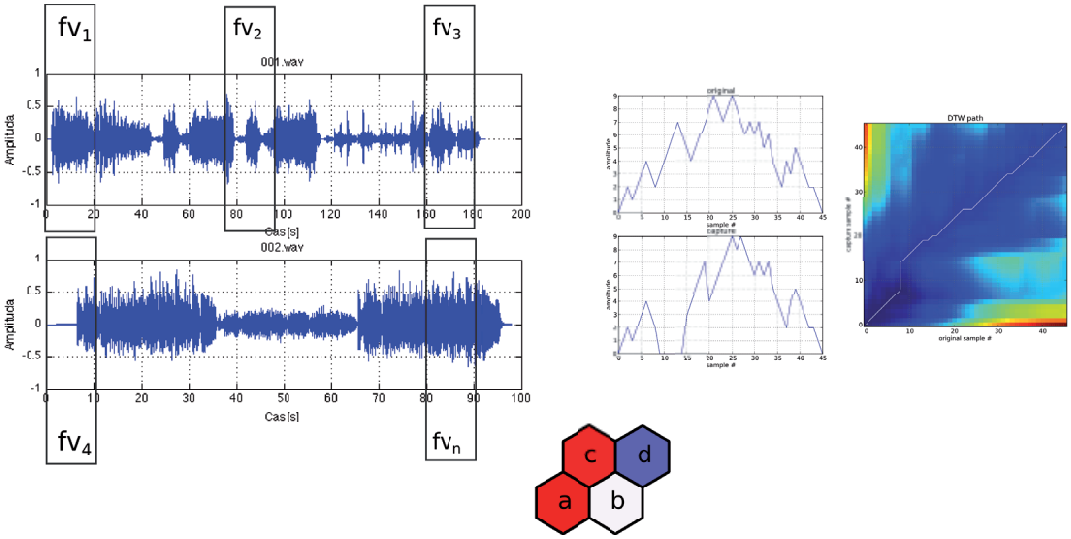


1: *Original and captured recording*

1 http://www.videolan.org/vlc/
2 http://audacity.sourceforge.net/



2: *Original and captured recording, detail 4–10 s*



3: *Experiment layout*

part of Fig. 3) were compared with Dynamic Time Warping algorithm (right part of Fig. 3).

Dynamic Time Warping

The Dynamic Time Warping algorithm (Müller, 2007) is an algorithm designed for the time series comparison. The input consists of two single dimensional vectors (Albanese, 2012) to be compared. The similarity of vectors is represented in the graphical mean by the similarity matrix comparing each sample in one succession with all samples in other one. Moreover, we can find the "path" through the most similar points in a similarity matrix. This "path" forms a vector of best alignment between two series.

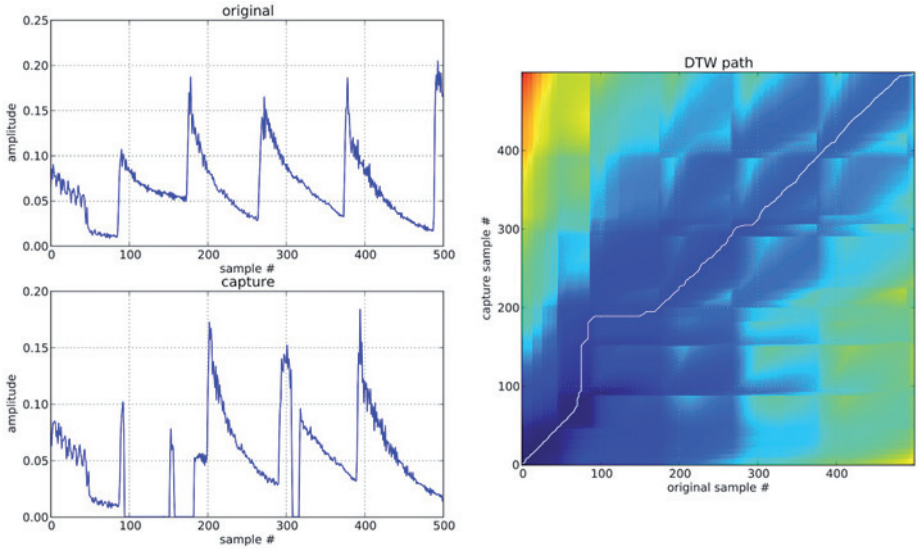
The total cost of the "path" going through the most similar points of two aligned time series can be calculated as:

$$c_p(X,Y)=\sum_{I=1}^{L} c(x_{n_I}, y_{m_I}),$$

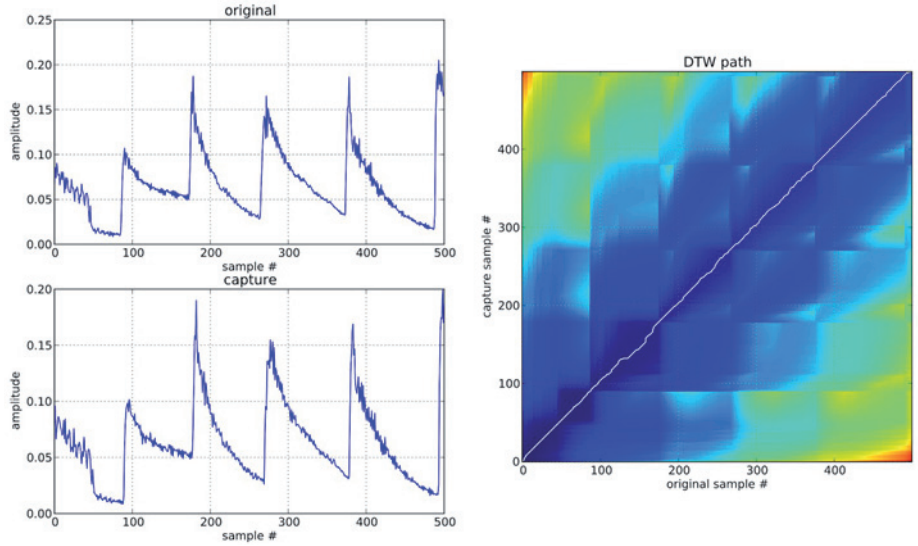
where c_p denotes total cost of warping path between vectors x and y and $c()$ denotes local cost measure. This total cost of warping path can give us a numerical measure of time series similarity (Müller, 2007).

RESULTS

We can compare detail (5 to 10 s) of captured recordings on Fig. 4 and Fig. 5. There are significant failures between sample (standard deviation of the time frame) number 100 and 200 indicated by the 0 value of the amplitude in bottom left part of the figure and also indicated by the misalignment of the path in the right "DTW path" part of the Fig. 4. In addition we can see a more detailed view of the



4: *Detail of captured recording (5–10 s) without QoS*



5: *Detail of captured recording (5–10 s) with QoS*

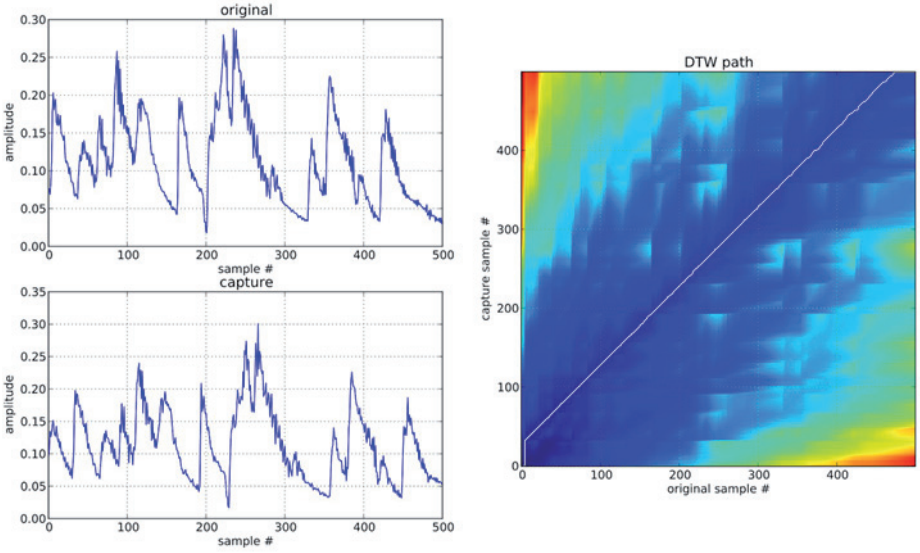
failure: if all packets transmitted during the failure are simply thrown away and sound continues, as nothing happened, the rest of the captured recording will be aligned in time to the original. In our case, the rest of recording is also shifted, so the path do not follow the (bottom left – top right) diagonal of the matrix. We can observe the horizontal path in the top right corner of the "DTW path". This means, that there is a delay between original and captured recording. The total cost of this path is 6.19.

On the other hand, when deploying QoS mechanisms much better results were achieved: the failure (there is the same scenario of network traffic) disappeared, as we can observe on the Fig. 5. In addition, the path follows the diagonal of the similarity matrix much closer, which indicates a smaller delay. The total cost of this path is 1.19.

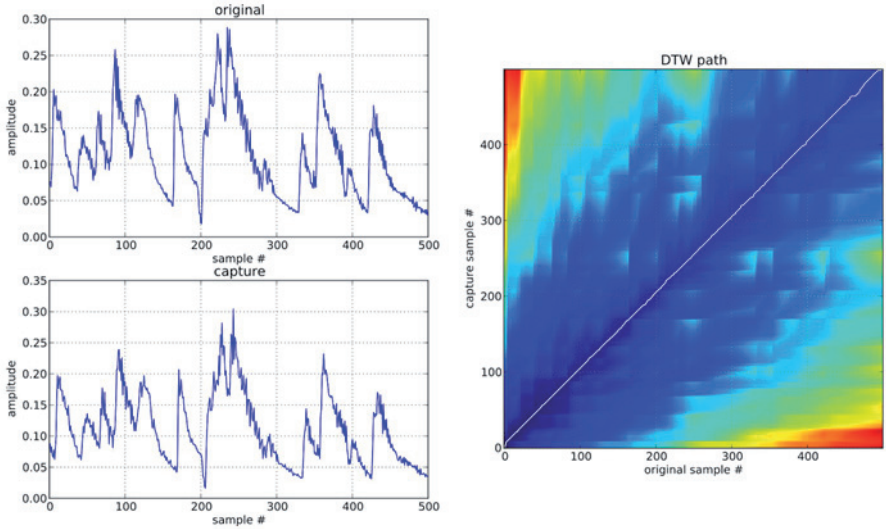
We can see another part, covering the interval from 50 s to 55 s, of the captured recordings in Fig. 6 and Fig. 7, which is situated in the final part of the capture (compared with former case covering part in the beginning of the capture). The network traffic in that area is more "smooth", there are no significant dynamic changes caused by the beginning of the video stream transmission.

This resulted in a straighter path even in the case when we are not using QoS. But we can observe that this path is not going through the diagonal but is shifted up by approximately 30 samples indicating a delay of 0.3 s, which is unacceptable in the case of telephony. The total cost of this path is 3.18.

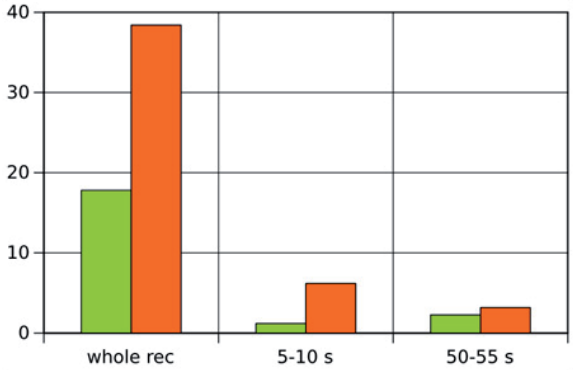
Using the QoS mechanisms this delay almost disappears. Furthermore, we can observe how the small delay (see values of amplitude on Fig. 7 around the sample 200) is still reduced (path going closer to



6: *Detail of captured recording (50–55 s) without QoS*



7: *Detail of captured recording (50–55 s) with QoS*



8: *Total cost with (green) and without (orange) QoS*

the diagonal) finishing almost in the top right corner (zero delay) of Fig. 7. The total cost of this path is 2.26.

DISCUSSION

In the situations with the QoS activated, the results are always better, as depicted on Fig. 8. The advantage of QoS lies in two ways concerning our experiment: the errors caused by strong fluctuations in network traffic (start of a video stream) almost disappeared, and the delay between the original recording and captured recording is much smaller and it is reduced further. Mechanism reducing the delay is beyond the scope of this article and can be analysed in our future work.

This experiment also verified that it is possible to use the Dynamic Time Warping for audio errors inspection. We can utilize its graphical output as well as normalized minimum-distance warp path between sequences c_p (X,Y) as a numerical measure of audio quality.

We used Python with NumPy, SciPy and matplotlib libraries for numeric calculations and

plotting in this experiment, which turned to be a good open-source tool for sound signal analysis. We used DTW implementation from the mlpy[3] package, which is a handy tool for machine learning tasks or augmented reality problems. Our source code can be found in the soundpylab repository[4].

SUMMARY

In this paper we show the possibility of utilizing Dynamic Time Warping algorithm (its total warping distance measure) for numerical evaluating of audio quality streamed over the network. Numerical evaluation has most benefits in fast and less expense realisation compared with statistical evaluation of human respondents who are involved when it is necessary to evaluate quality of sound transmitted through the network. These respondents usually evaluate sound quality of voice as "clear", "understandable" or "inarticulate". It has its benefits in lack of need for original recording (voice), which was streamed into the network. By contrast, when we can use the original recording (when using IP telephony, it is usually easy to record original voice) we can compare original and received recordings by means of Dynamic Time Warping. Our results have shown, that recordings transmitted using QoS had better quality in all examined cases. But in cases of rapidly increase of traffic (start of several huge downloads in the network) there are failures in voice stream even when is QoS utilized. We have found several different types of failures and have shown, that we can inspect some aspects of these failures with DTW path visualisation (e.g. delay initiated due to failure of stream decreased in time – the path forthcoming to lower-left upper-right diagonal). Further investigation of types of failures and conditions, when they appear should be interesting topic for consequent research.

REFERENCES

ALBANESE, D., 2012: *mlpy Documentation*. available from http://sourceforge.net/projects/mlpy/files/mlpy%203.5.0/mlpy.pdf/download.

BALOGH, Z., MUNK, M., CÁPAY, M., TURČÁNI, M., 2010: Usage Analysis in e-Learning System for Healthcare. *The 4th International Conference on Application of Information and Communication Technologies AICT2010*. Tashkent: IEEE, p. 131–136. ISBN 978-1-4244-6904-8.

ČERNOCKÝ, J., HUBEIKA, V., 2009: *Speech Recognition using DTW and HMM*. FIT BUT Brno. Available from http://www.fit.vutbr.cz/~ihubeika/ZRE/lab/05_dtw_hmm.pdf.

FEJFAR, J., ŠŤASTNÝ, J., 2011: Time Series Clustering in Large Data Sets. *Acta Univ. Agric. et Silvic. Mendel. Brun.*, 59, 2: 75–80, ISSN 1211- 8516.

FEJFAR, J., ŠŤASTNÝ, J., CEPL, M., 2012: Time Series Classification Using K-nearest Neighbours, Multilayer Perceptron and Learning Vector Quantization algorithms. *Acta Univ. Agric. et Silvic. Mendel. Brun.*, 60, 2: 69–72. ISSN 1211-8516.

KOHONEN, T., SCHROEDER, M. R., HUANG, T. S. (Ed.), 2001: *Self-Organizing Maps*. Secaucus, NJ, USA: Springer-Verlag New York, ISBN 3-540-67921-9.

MÜLLER, M., 2007: *Information Retrieval for Music and Motion*. Berlin: Springer Berlin Heidelberg, 318 p. ISBN 978-3-540-74047-6.

ZACH, P., 2012*: The design of network traffic generator core*. Diploma thesis. 90 p. Brno: MENDELU.

ŠKORPIL, V., ŠŤASTNÝ, J., 2008: Comparison of Learning Algorithms. In: *24th Biennial Symposium on Communications*. Kingston, Canada: Queens University Kingston, p. 231–234. ISBN 978-1-4244-1945-6.

3 http://mlpy.sourceforge.net/
4 https://code.google.com/p/soundpylab/

Address

Ing. Jiří Fejfar, Ph.D., prof. RNDr. Ing. Jiří Šťastný, CSc., Ing. Martin Pokorný, Ph.D., Ing. Jiří Balej, Ing. Petr Zach, Department of Informatic, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: jiri.fejfar@mendelu.cz, jiri.stastny@mendelu.cz, martin.pokorny@mendelu.cz, jiri.balej@mendelu.cz, petr.zach@mendelu.cz