HAND DETECTION APPLICATION BASED ON QRD RLS LATTICE ALGORITHM AND ITS IMPLEMENTATION ON XILINX ZYNQ ULTRASCALE+

R. Likhonina[*], *E. Uglickich*[†]

Abstract: The present paper describes hand detection application implemented on Xilinx Zynq Ultrascale+ device, comprising multi-core processor ARM Cortex A53 and FPGA programmable logic. It uses ultrasound data and is based on adaptive QRD RLS lattice algorithm extended with hypothesis testing. The algorithm chooses between two use-cases: (1) "there is a hand in front of the device" vs (2) "there is no hand in front of the device". For these purposes a new structure of the identification models was designed. The model presenting use-case (1) is a regression model, which has the order sufficient to cover all incoming data. The model responsible for use-case (2) is a regression model, which has a smaller order than the model (1) and a certain time delay, covering the maximal distance where the hand can possibly appear. The offered concept was successfully verified using real ultrasound data in MATLAB optimized for parallel processing and implemented in parallel on four cores of ARM Cortex A53 processor. It was proved that computational time of the algorithm is sufficient for applications requiring real-time processing.

Key words: *hand detection, ultrasound, QRD RLS lattice algorithm, parallel implementation, hypothesis testing*

Received: December 14, 2021
Revised and accepted: April 30, 2022

DOI: 10.14311/NNW.2022.32.005

1. Introduction

The present paper focuses on QRD recursive least squares (RLS) lattice algorithm used for solving hand detection problem based on ultrasound technology. The paper describes a new approach to hand detection as well as algorithm implementation on Xilinx Zynq Ultrascale+ device.

[*]Raissa Likhonina – Corresponding author; Czech Academy of Sciences, Institute of Information Theory and Automation, Signal Processing Department, Pod Vodárenskou věží 4, CZ-180 00, Praha 8, Czech Republic; Czech Technical University in Prague, Faculty of Transportation Sciences, Konviktská 20, CZ-110 00 Praha 1, Czech Republic, E-mail: likhonina@utia.cas.cz

[†]Evženie Uglickich; Czech Academy of Sciences, Institute of Information Theory and Automation, Signal Processing Department, Pod Vodárenskou věží 4, CZ-180 00, Praha 8, Czech Republic, E-mail: uglickich@utia.cas.cz

Usually, such types of algorithms are demanding on computer resources and can suffer from slow performance given a large amount of data. The work is motivated by the need to present a robust solution for hand detection problem, which can function reliably on small platforms and process a large amount of data in real time. Such kind of applications can be used in automobile industry, in smart home/buildings areas and in other fields, where touchless applications can bring benefits in terms of more safety, better hygiene and improved comfort.

In general, the algorithms from RLS family are very popular and widely used in many areas. Particularly, the QRD RLS lattice is favoured for its structure, which allows relatively easy implementation and parallelization on hardware platforms. To name a few areas of its implementation, the following should be mentioned: system identification, speech analysis, noise and echo cancellation and etc. In order to implement the algorithm on a small platform with small memory footprint, it is necessary to solve the problem of its high computational complexity and numerical stability [24]. Detailed researches in this area are provided in [4, 5, 9, 24, 30, 31]. There are also works describing the ways to increase throughput of the algorithm and giving precision analysis [2, 7, 8, 19, 23, 26].

The algorithm under consideration is based on Bayesian theory for adaptive system identification in real time. In this respect [15, 18, 27] greatly contributed in probability approach to identification of stochastic systems and in using forgetting factor (whether exponential [15] or directional [18]) during identification process.

Kadlec J. developed the algorithm of probability identification for a model of the vocal tract, where the algorithm had lattice structure and could be used for parallel implementation [13]. The researcher also tried to find a method of continuous identification of the probability of a regression model in [14].

As it was mentioned above, the algorithm is very often used for noise/echo cancellation applications. The most popular applications are those in the area of telecommunication and mobile speech recognition application [3, 17, 25, 29, 33].

A very interesting work in the field of using the algorithm for noise cancellation is [21], which describes noise reduction technique based on the QRD RLS algorithm and the ways of its implementation on FPGA-based platform. The paper also introduces simulation in MATLAB and on FPGA and discusses obtained results [21]. However, in this case there is no parallel architecture being used.

As far as ultrasound technology is concerned, only one more or less relevant article was found [1]. In the work [1] the authors try to use a new method based on RLS adaptive filtering to eliminate the effect of the blurring of tissue reflectivity, which deteriorates the biomedical ultrasound image quality. The experiments proved that due to the RLS algorithms, it is possible to improve contrast and resolution of the image and the algorithm itself can be considered reliable. The authors also managed to reduce the dimensionality, which led to the computational complexity decrease [1].

Underling the novelty of the present research, the following points come into consideration:

1. It is clear that the field of investigation in regards of RLS algorithms in general and in regards of QRD RLS lattice algorithm in particular is well studied and profoundly described in many works, and that the existing algorithms function very efficiently on large computers; however, there is still a problem

to implement them on small area chips. The microprocessors have usually a small memory footprint. Processing the large amount of data, which is often the case in acoustic signal processing, can cause slow performance. The present research describes the way of using the QRD RLS lattice algorithm on Xilinx Zynq Ultrascale+ device.

2. Moreover, this particular work is performed within the project focused on ultrasound technology. It can be noticed that there is a gap in the research area. There exist several works dealing with ultrasonic diagnostics and improvement of diagnostics results with the help of the RLS algorithm [1, 3]. Still no work was found, which would describe how to use the RLS algorithms for hand detection applications based on ultrasound.

3. Furthermore, the QRD RLS lattice algorithm in the present work is based on noise cancellation technique. In the research field, there is a large amount of scientific investigations focusing on this issue, e.g. [3, 17, 21, 25, 29, 33]. However, they mainly concentrate on telecommunication and mobile speech recognition applications, which are out of scope of the present paper, where the algorithm is supposed to pre-process incoming data in a way to remove undesired ultrasound responses from the target signal, subject to use for hand detection.

4. Last, but not least, the present paper describes the way the algorithm can be extended with hypothesis testing to identify the probability of identification model best suited for a particular situation depending on the incoming data (hand presence/absence) [13,14,27]. Such kind of the algorithms was already proposed in [13, 14] and successfully tested for RLS lattice in application for speech coding. However, hand detection applications based on ultrasound technology have their specific features. In this context the hypothesis testing is supposed to be applied in a different way. As far as the signals can come to microphones at different angles (not necessarily perpendicular) and with different delay, it seems to be more appropriate and important to identify the structure of a regression model and to choose a particular identification model, which corresponds better to a real-time situation, rather than to estimate only the order. Such kind of a solution in the field under consideration was not found in literature.

The present paper consists of four sections. The first section describes the state of the art, motivation of the work and its novelty.

The second chapter comprises problem formulation and the ways of solving the problem, including algorithm description, methodology, implementation in MATLAB [22] and parallel implementation on Xilinx Zynq Ultrascale+ device.

The third chapter discusses the results of the work, summarizes the main contributions, potential applications and limitations of the approach.

The last chapter is Conclusion, where the main points of the paper are reminded.

2. Hand detection using QRD RLS lattice algorithm

2.1 Problem formulation

The basic device for hand detection applications based on ultrasound technology consists of a network of ultrasound transducers with integrated pre-processing unit. The ultrasound impulses transmitted by the system reflect from a hand of the user and return back to the system. On the basis of responses and their characteristics, the device is supposed to detect the presence, position and distance of the hand. This seemingly easy principle of hand detection in theory turns to be challenging in practice. The problem is the reflections from the objects in the environment other than the hand. These undesired responses can present a great challenge for a detection process and, therefore, should be removed from the target signal.

It follows out that for pre-processing of incoming data in a way of reducing acoustic noise, it is necessary to develop the corresponding algorithm based on noise cancellation technique. The algorithm has to be numerically robust and fast, able to process incoming data in real-time.

The present paper describes the QRD RLS algorithm based on noise cancellation technique and extended with hypothesis testing. It helps to identify the probability of identification model, which is more suitable for the given situation [13, 14, 27]. Application of recursive Bayesian identification to the defined problem enables to compare the structures of regression models and to make decision between two use cases ("there is a hand in front of the device" vs "there is no hand in front of the device") at a certain computation step. It results in more precise hand presence detection and in determining the exact moment when there is a response from the hand. Furthermore, based on the results of identification and due to the specific form of the input signal, the distance between the hand and the device can be measured.

The algorithm is computational demanding in respect to memory and time of computation. However, in industrial applications, e.g. for hand detection in automobile applications, smart building applications, applications for wearables, using large computers is not considered. Therefore, the algorithm was optimized for small platforms, which can be already used in above mentioned areas, and was implemented for illustration purposes on Xilinx Zynq Ultrascale+ device. For these purposes the Trenz Electronic TE0808 SoC and the Trenz Electronic TEBF0808 carrier board are used. The main characteristics and parameters of the device are described in a subsection devoted to parallel implementation of the algorithm on Xilinx Zynq Ultrascale+ device.

2.2 Algorithm description and methodology

This section is devoted to the preliminaries and the description of a proposed approach to the hand detection problem.

2.2.1 Mathematical basis of the algorithm

In the present approach noise cancellation technique based on the QRD RLS lattice algorithm is used. The basic principle of the approach is grounded on the assumption that there are two types of signals: the desired signal and the reference ultrasound source signal. The desired signal is constituted with the temporary present short distance reflection signal and the reflection signal mixed with the environment. Due to the static relation of the environment reflection signal and the ultrasound source signal, the temporary present short distance reflection signal can be reconstructed as a prediction/filtration error of the adaptive QRD RLS lattice algorithm. Using prediction/filtration errors, hand detection can be performed. However, to make the detection more accurate and reliable and to enable to calculate hand distance from the device, the QRD RLS lattice algorithm is supplemented with hypothesis testing.

To detect a hand, two regression models are used. One model describes the situation, when there is a hand in front of the device, whereas the other one – when there is no hand in front of the device.

According to [14, 27], a regression model can be presented by the following equation:

$$y_t = \theta_t^{\mathrm{T}} \cdot Z(N) + e_t, \tag{1}$$

where y_t is output in time t, θ_t is a vector of unknown regression parameters, T is transposition, $Z(N)$ is a data vector consisting of the delayed output values and input data u_t, N is a number of data, e_t is a white normal noise with unknown variance.

As it was previously mentioned, calculating prediction/filtration errors, it is already possible to detect the hand presence, as far as it is the hand, which causes a short-time disturbance. However, the exact moment when the hand appeared and disappeared is not always detectable due to the sensitivity of the model to the setting of the exponential forgetting factor. Therefore, in the present work it is offered to use two models with different structures to make a detection process more precise. The choice of the appropriate model given incoming data is performed using hypothesis testing.

Generally, in Bayesian RLS regression model approaches including works [14,15, 18,27], several assumptions for performing hypothesis testing in regards to the least square computation are proposed. First of all, it was assumed that a stochastic system can be described by the regression model (1) in the form of conditional probability density function:

$$p(y_t|D(t-1); u_t, \theta_t, \omega_t) = \epsilon \cdot \omega_t^{1/2} \cdot exp\{-\omega_t/2 \cdot [y_t - \theta_t^{\mathrm{T}} \cdot Z(N)]^2\}, \tag{2}$$

where $D(t-1)$ is all previously observed data till time $t-1$, ϵ is a normalizing constant, ω_t is an unknown degree of accuracy [14, 16].

The second important assumption is that the prior conditional probability density of the parameters θ_t, ω_t for time $t = T, T+1, \ldots$ has the form of Gaussian-Wishart distribution [14, 16].

These equations (1) and (2) serve as a basis, which is applied then for the QRD RLS lattice algorithm. The detailed derivation is provided in [13–16, 27]. Here only the final equation for hypothesis testing after all approximations is presented. Note that the upper line "$\overline{\ }$" above the letter under the characteristics in the equation (3) is used to show that the corresponding characteristics is after updating with y_t [14].

$$p(y_t|D(t-1);H_n) = \pi^{-1/2} \cdot \frac{\Lambda^{(\frac{\upsilon-n}{2}+1)}}{\overline{\Lambda}^{(\frac{\overline{\upsilon}-n}{2}+1)}} \cdot \frac{|V(n)|^{1/2}}{|\overline{V}(n)|^{1/2}} \cdot \frac{\Gamma(\frac{\overline{\upsilon}-n}{2}+1)}{\Gamma(\frac{\upsilon-n}{2}+1)}, \tag{3}$$

where

- H_n is a hypothesis, where n is the order of the model.
- Λ is a scalar, which enables to express the conditional mean value by distribution of unknown parameters. It is a residue after completion to the full squares and it represents a point estimate of the parameters.
- υ is the number of data samples accumulated in $V(n)$.
- $V(n)$ is a square symmetric positive definite matrix of size $n \times n$. It is an information matrix, which comprises mean value, variance and covariance, and it represents statistics, which are necessary for parameter estimation of the model. See, e.g. [16].
- $\Gamma(\cdot)$ is gamma function.

More detailed information can be found in [13–16, 24, 27].
In [13] further approximations are made to avoid numerical problems.

2.2.2 Model description and the principle of algorithm functioning

The general description of the way the algorithm functions is presented below.

As it was already mentioned above, in the present research the structures of identification models and their orders are designed and chosen to describe two use cases: there is a hand in front of the device and there is no hand in front of the device. The algorithm estimates the probability of two hypotheses and chooses the identification model, which suits best for the incoming data. The basic concept is presented in Fig. 1.

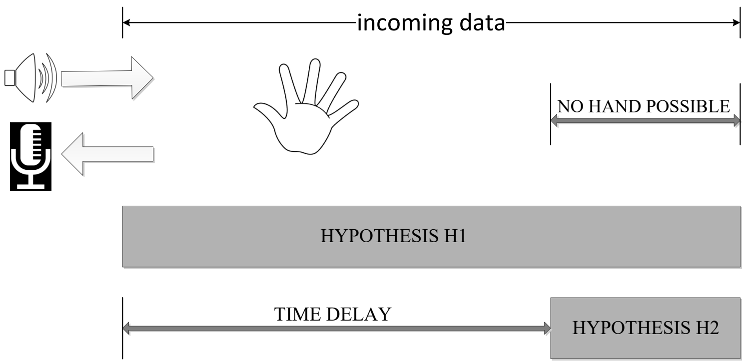

Fig. 1 *Hypothesis testing models.*

As for the structure of the models, the following assumption is made. Both identification models use the same incoming data from an ultrasound device. The first model, which presents hypothesis H_1, has a higher order, which is chosen on the knowledge of the nature of the incoming data. It should cover all incoming data from the sensors; thus, the model learns using all available data during specific period of time and performs echo cancellation as precise as it is possible. The first model learns on the situation, when there is no hand, so it creates a certain relation between its finite impulse response (FIR) parameters and input/output data. When there is no hand, the identification model performs filtration process very accurate. However, when the hand appears, the relation between parameters of the model and input/output data is strongly affected. The changed situation makes the identification model learn again; therefore, it becomes inappropriate for the given situation with the hand presence in front of the device.

The second model, which presents hypothesis H_2, has a smaller order and a certain time delay. It works only on data where there is no hand presence possible. The second identification model does not have all available data; thus, it cannot compute filtration errors in a right way, so that there is a constant problem with estimation process for the second model. When the hand appears in front of the device, nothing changes for the second model, because due to time delay, the hand is out of the area where the model has its FIR parameters. Thus, the relation between its FIR parameters and input/output are the same, though there will be a higher dispersion on the output due to more noise related to the reflections from the hand. Based on this fact, the second identification model will have a higher probability when the hand appears in front of the device, because its parameters do not change. The choice of the order and time delay for the second model depends on the assumption that the hand can appear only in a certain distance from the device.

To summarize all points mentioned above, from Fig. 1 it is clear that the hand can appear only in the area of time delay. In this way, two use-cases are possible:

- if there is no hand in front of the device, the identification model with a higher order has a higher probability; thus, hypothesis H_1 says "there is no hand in front of the device".
- Otherwise, the identification model with a smaller order and time delay describes the given situation better and more accurately, and, therefore, has a higher probability. Thus, hypothesis H_2 says "there is a hand in front of the device".

In this way, on the basis of the outputs of the recursive model probability estimation, it is possible to identify the hand presence. The block diagram of identification process is presented in Fig. 2.

The block diagram comprises three main blocks, among which the block representing real ultrasound data from a microphone and two identification blocks representing two regression models of different orders, one of which with a predefined time delay. The block diagram has the following notations: u is input, y is output, n_1, n_2 are orders of the models, $\widehat{e}_1$, $\widehat{e}_2$ are filtration errors, which are strongly dependable on the short-time disturbance caused by the hand appearance. Therefore, already on their basis it is possible to identify hand presence.

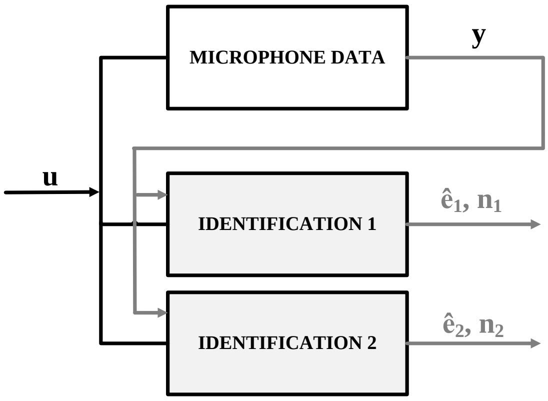

Fig. 2 *Block diagram.*

Due to the fact that using hypothesis testing one can detect the exact moment of hand appearance with a certain precision and due to the fact that input signal is in the form of pulses (chirps), knowing the number of samples between the input (a chirp) and the response from the hand, it is possible to calculate the hand distance from the device according to the well-known equation:

$$s = \frac{v \cdot t}{2}, \tag{4}$$

where s is the distance [m], v is the speed of ultrasound in the air, which is 343 m/s, t is time [s].

In the Eq. (4) the time is divided by 2, because it should be taken into consideration that the signal goes from and back to the device.

Time can be calculated according to the following equation:

$$t = \frac{N}{f_s}, \tag{5}$$

where N is the number of samples between the input and a hand response, f_s is a sampling frequency [kHz].

2.3 Verification of algorithm functionality

The experiments with the algorithm described above were firstly performed and verified in MATLAB [22]. The goal of the experiments is to show that the algorithm functions in a correct way and detects the hand presence in front of the device as well as calculates distance between the hand and the device. Then, the algorithm from MATLAB and its outputs served as a golden model for implementation on Xilinx Zynq Ultrascale+ device.

Ultrasound data for the experiments described below were provided from the device designed within the bound of SILENSE project by UTIA. The device consists of three basic components: TE0720 FPGA SoM module, TE0706 carrier board and UTIA evBoard v1.7. The board is equipped with 32 digital microphones and the ultrasound speaker. The ultrasound speaker sends chirps 600 times on the 40 kHz frequency. Between sending the preceding signal and the following signal there is a certain waiting period or a delay. During this waiting period nothing is sent (see Fig. 3). Each signal has 880 PCM samples, which are obtained by sampling with sampling frequency of 192 kHz.

For the purposes of the experiments and to reduce the computation complexity and a high demand on memory of the HW platform due to its limited computational resources, data only from one microphone are used.

The raw output signal is presented in Fig. 3. It has 11 520 000 samples, lasts for 60 s and after each pulse there is a certain time delay, which constitutes 18 320 samples or 0.095 s.

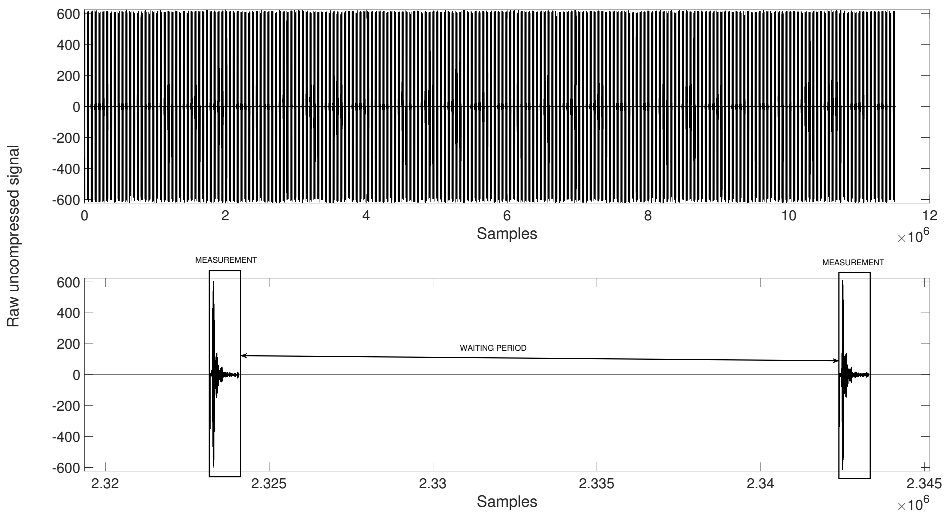

Fig. 3 *Raw uncompressed output signal from the ultrasound device.*

On the upper graph of Fig. 3 the raw uncompressed signal is presented. It is impossible to differentiate by a human eye where there is a hand appeared. The bottom graph of Fig. 3 shows the enlarged pulses and the waiting period between them.

For the practical and computational purposes the raw signal is compressed, i.e. the delays between the preceding and the following signals are removed. Moreover, the raw output signal is modified in a way of replacing the self-listened inputs (crosstalks) by zeros. It is possible to make it without affecting the validity of the experiments, because the period of their occurrence is known. The final output signal used in the experiments is shown in Fig. 4.

On the graph of Fig. 4 there are some higher and smaller peaks. The higher peaks represent the responses from the hand. The smaller peaks come back from other objects in the environment. They should be removed. Fig. 4 shows that during the measurement there were six hand appearances on different distances from the device. The development of the third hand presence is different from the others. It can be explained by the fact that in the case of the third hand appearance,

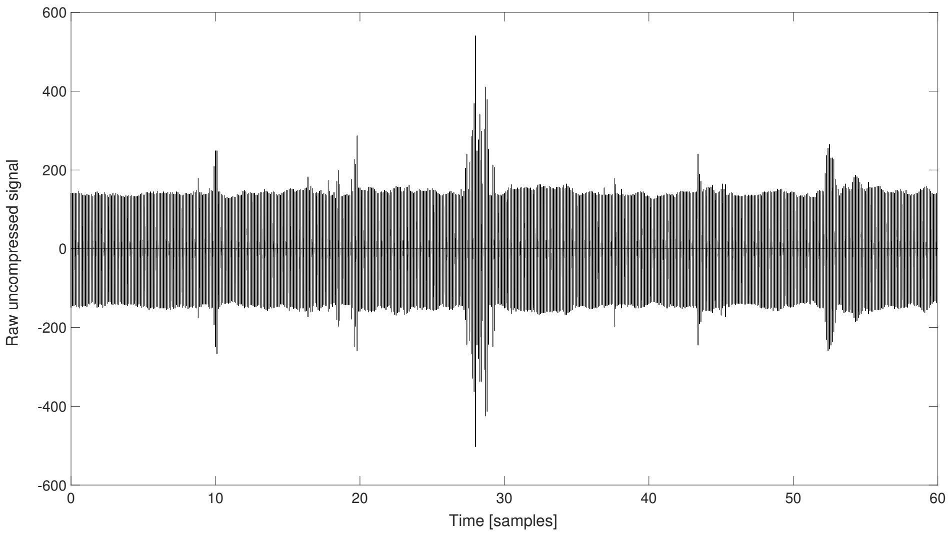

Fig. 4 *Output signal.*

the hand was moving forwards and backwards from the device. The algorithm has to, firstly, to detect all six hand appearances, and, secondly, to identify the distance of the hand from the device in each particular case.

The input signal is provided by the ultrasound speaker. It is in the form of chirps represented by a sinusoid wave. Its period is 5 samples, sampling frequency is 192 kHz and there is an 880 sample space between the chirps. To measure the input signal directly is not possible. Therefore, its reconstruction from the raw output signal is performed. Due to the fact that the microphone and the speaker are situated close to each other, the reconstructed input signal is very similar to the real one (see Fig. 5).

The settings of the identification models are the following. The order of the first model is set to $n_1 = 768$. This order is enough to cover the available data for the one pulse and at the same time it is good for pipelining into 3 or 6 processes for parallel implementation on Xilinx Zynq Ultrascale+ device.

The order of the second model is set to $n_2 = 256$ and there is a time delay of 512 samples. In this way it is assumed that the second identification system makes estimation on the data, where there is no hand appearance possible, i.e. on the distance in the range from 0 cm to 46 cm. Besides, this order is also easily divided into 1 or 2 processes.

After fulfilling a set of experiments, the optimal value of the exponential forgetting (EF) factor was found, which is 0.99997.

Fig. 6 shows computed filtration error (upper black curves) and hypothesis development during time (grey and light grey curves in the bottom of the graph). In the beginning of the estimation process, the algorithm needs some time to estimate parameters in a right way. It is the learning stage of the algorithm. Therefore, there is uncertainty, which identification model to choose. However, after the learning stage, the estimation process converges to correct values and the algorithm

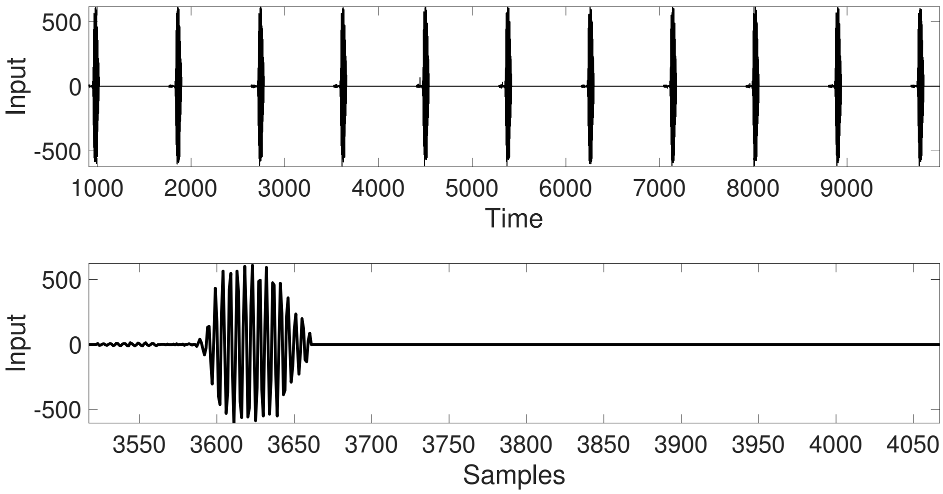

Fig. 5 *Input signal.*

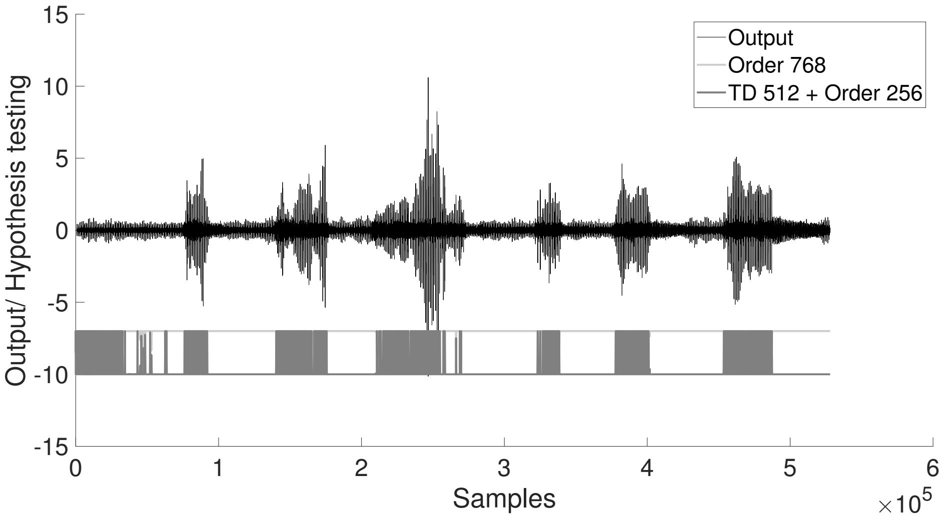

Fig. 6 *Hand detection.*

accurately recognizes the hand appearance by switching between two hypotheses. In case when there is a hand, the system switches into the identification model with a smaller order and with the time delay (grey coloured). Contrarily, if there is no hand, the identification model with a higher order (light grey coloured) has a higher priority.

The filtration error computed by the identification model trained on all available data, i.e. the identification model with a higher order, is accurately estimated and can be used along with the hypotheses to detect the hand presence (see Fig. 7).

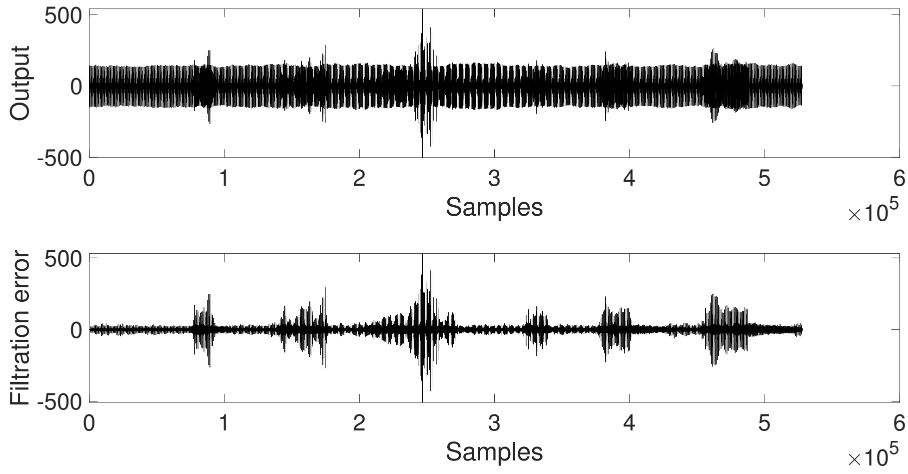

Fig. 7 *Hand detection (filtration errors).*

The upper graph of Fig. 7 represents the output signal y, while the bottom graph shows the development of the filtration error. It is clear from Fig. 7 that unwanted responses are removed from the target signal and it can be used for the further data processing. Moreover, hypotheses help to define the precise moment the hand appears and disappears and in this way they allow to say from where the further signal analysis should be started.

Fig. 8 illustrates compressed signal and hypothesis development during the computation process along with the calculated distance of the hand to the device. From

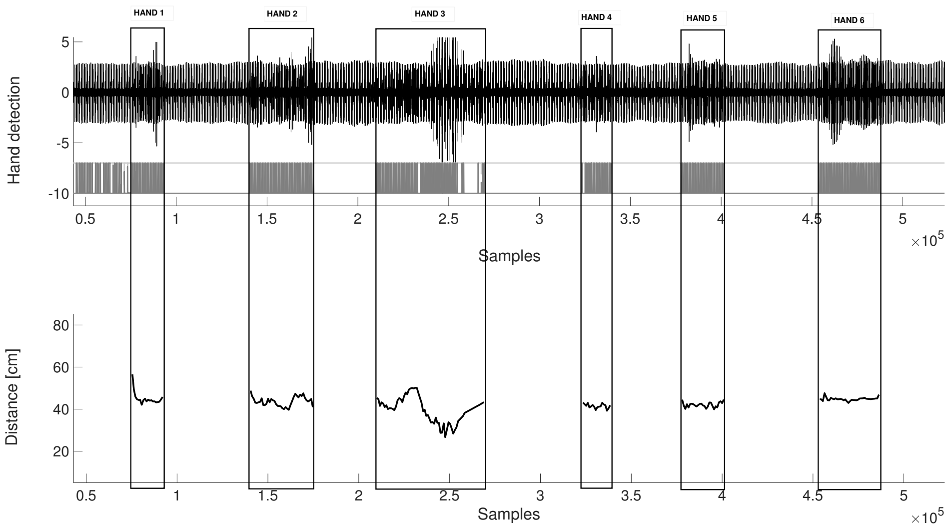

Fig. 8 *Hand distances (compressed signal).*

the graph in Fig. 8 it is obvious that during the measurement five hand occurrences were more or less in the same position and their distance was approximately the same, i.e. approximately 40–45 cm. There are some minor changes in the position of Hand 2, which constitutes only several cm. However, in the case of Hand 3 the changes in position are more visible. They can be explained by the fact that during the measurement the operator was moving his hand towards and backwards from the device; therefore, its distance varies from 27 cm to 50 cm. Though the assumption about the maximal possible distance of the hand from the device, that is 46 cm, is not valid for Hand 3 as far as it was moving in the range from 27cm to 50 cm, still the algorithm proves to be robust and the hypothesis testing functioned reliable in this case too.

In the case of uncompressed output signal, it is also possible to obtain the information about the duration, during which the hand was present in front of the device. More detailed information about time as well as about precise hand distance to the device is presented in Tab. I.

Hand	Distance [cm]	Time [s]
Hand 1	44	1.9
Hand 2	40–45	3.9
Hand 3	27–50	6.7
Hand 4	41	1.7
Hand 5	40	2.6
Hand 6	45	3.7

Tab. I *Hand distances.*

2.4 Parallel implementation on Xilinx Zynq Ultrascale+ device

To use the algorithm in applications, where small platforms are used, the algorithm should be optimized and accelerated. The further experiments show implementation of the algorithm on a small platform with a small memory footprint. For the purposes of implementation of the QRD RLS lattice algorithm, the Trenz Electronic TE0808 SoC and the Trenz Electronic TEBF0808 carrier board are used.

The Trenz Electronic TE0808 is an industrial MPSoC module, which comprises Zynq UltraScale+ ZU9EG, four core ARM processor of frequency 1GHz, programmable logic max. 200 MHz, 64-bit DDR4 (8 GB maximum), dual SPI boot Flash in parallel (512 MB maximum), user I/Os, B2B connectors. It is of size 52×76 mm and it requires 3.3 V power supply [11].

The Trenz Electronic TEBF0808 carrier board is a baseboard, which is used for the module described above. It comprises on-board components, which serve for testing and evaluating modules compatible with this board. The board can be fitted into a PC enclosure [12].

The prototype of the device consists of Trenz Electronic platform, a computer, a display where the results of computation can be viewed, a ventilator to cool the ARM processor and other accessories. The board is connected via ethernet cable to a mini computer UMAX U-Box N41 with Intel Celeron Quad Core N4100 (Gemini Lake) Quad-Core 1.1 GHz (max. 2.4 GHz), Intel UHD Graphics 600, 4 GB DDR4 RAM, 64 GB eMMC. The prototype device allows performing computation of the algorithm, viewing the results on the display and making necessary changes in the algorithm if required. Besides, it is portable and, thus, it enables a certain level of flexibility and convenience while working with it.

Because the ARM device has 4 core processor, it is reasonable to modify the algorithm in a way of using computational resources of all four cores; thus, accelerating the algorithm performance.

The input parameters are copied to local memory via .h files prepared in MATLAB in advance. Once they are in the local memory, one core computes the QRD RLS lattice algorithm with order $n_2 = 256$ and time delay $TD = 512$ for hypothesis H_2. Other three cores compute the QRD RLS lattice algorithm with order $n_1 = 768$ for hypothesis H_1. However, the second algorithm is split into three threads $T_2 = T_3 = T_4 = 256$ and each part is processed separately by a separate core. The results of computation were verified with the reference model provided by MATLAB.

Knowing the sampling frequency, which is 192 kHz, and the number of data samples, which is 11520000, it is possible to calculate time, during which the algorithm has to perform outputs to be able to process data in real time.

$$T_{real} = \frac{N}{f_s} = \frac{11520000}{192000} = 60s. \tag{6}$$

Fig. 9 and Fig. 10 compare computational time and MFLOP/s for one core, two core and four core versions of the algorithm on the ARM device. The example is given for single precision (SP) arithmetic and for computing with data blocks equal to 528, i.e. there are outputs ready for hypothesis testing each 60 ms, which corresponds to 1000 results in 60 ms.

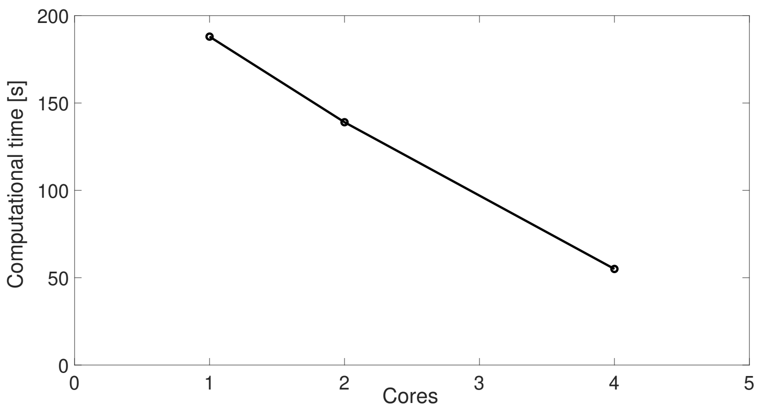

Fig. 9 *Computational time for a different number of cores.*

Tab. II shows computational time for four cores given data divided into smaller blocks, where ns is a division factor. The results are presented both for single precision (SP) and double precision (DP) arithmetic.

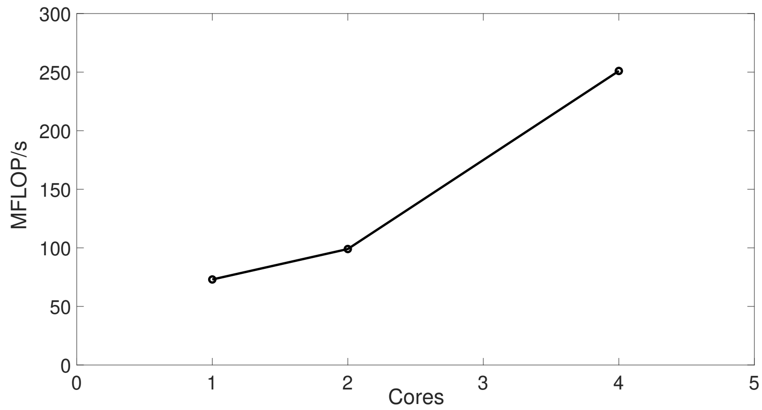

Fig. 10 *MFLOP/s for a different number of cores.*

ns	Time [s] for DP	Time [s] for SP
10	66.57	62.74
20	62.18	62.70
50	59.40	56.40
100	58.57	55.77
200	58.22	55.23
1000	***57.51***	***54.85***
2000	57.66	55.25
4000	57.83	55.33
8000	57.89	55.13
16000	58.82	55.56

Tab. II *Computation time for different division factors.*

The shortest time for DP and SP is shown in bold italic in Tab. II. It means that the optimal division factor is $ns = 1000$.

To make the comparison clearer, Fig. 11 represents an example of the computational time for four core version of the algorithm for SP arithmetic.

The graph shows that the computational time is decreasing by making smaller data blocks purposed for computation and reaches its best value at $ns = 1000$, i.e. a block contains 528 samples at each step of computation process. However, after this value it begins increasing again as far as communication also increases. Thus, a division factor $ns = 1000$ is considered to be optimal for this case of data processing.

It should be also noted that the algorithm provides the whole information about identification process including the probabilities of each model for 528 samples every 60 ms, which helps to reconstruct what the hand did during this period of time.

Thus, from the experiments it is obvious that four core version of the algorithm reliably gives outputs within 60s and fulfils the requirement for real-data processing.

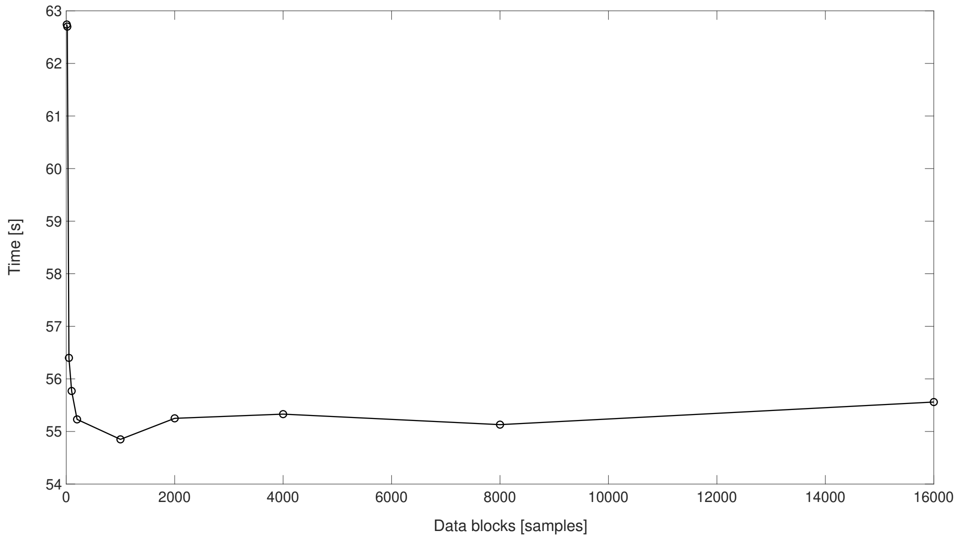

Fig. 11 *Computational time for four core version of the SP algorithm.*

3. Discussion

To summarize the points mentioned in previous sections, the main goals of the present research were:

- to create a new approach to hand detection based on ultrasound signals;
- to implement the proposed approach on a small platform with limited computational resources and to ensure that the algorithm can function reliable and provide computation in real time.

To achieve the first goal, the QRD RLS lattice algorithm was extended with hypothesis testing. A newly designed structure of regression models was proposed. On the basis of computed probabilities, it compares the structure of the given regression models and enables to eliminate the undesired responses from the objects in the environment. Application of recursive Bayesian identification allows making decision between two use-cases ("there is a hand in front of the device" vs "there is no hand in front of the device") at a certain computation step. It results in determining the exact moment when there is a response from the hand and due to the specific form of the input signal to measure the distance between the hand and the device. Thus, in this way the predetermined goals of the specified application can be achieved and the certain benefits are provided: a) noise cancellation, b) distance computation between the hand and the device.

The important contribution of the research is also the algorithm implementation on a small hardware platform, which opens the way for many potential applications in the areas, where hand detection is a need and using large computers is not possible. The examples of such areas are automotive industry, smart buildings, wearables, etc. Parallel implementation of the QRD RLS lattice algorithm was made on four cores of ARM Cortex A53 processor of Xilinx Zynq Ultrascale+ device. It succeeds to compute the algorithm using real ultrasound data sufficiently fast and can be used for real-time applications. Besides, the algorithm provides the complete information about identification process including probabilities for every 528 samples each 60 ms, i.e. the reconstruction of the hand behaviour is possible for the purposes of simple gesture identification based on the calculated distance between the hand and the device.

Still there are several limitations of the offered approach. Firstly, to set regression models correctly, the assumption about the maximal possible distance between the hand and the device (i.e. how far from the device the hand can still appear) should be made. Thus, the algorithm should be set for each individual application. However, generally this information is available and the settings can be made within trustworthy thresholds.

Secondly, the computation time of the algorithm on the described device is almost 60 s. It means that for less powerful platforms (less than 1.1 GHz) the algorithm could perform slower and could need further acceleration.

Thirdly, for the moment the algorithm works only with data from one microphone, while using the data from all microphones will increase its complexity and the demand on computational resources. However, it was proved during the experiments that this approach still provides reliable results, can detect the hand presence and calculate the distance between the hand and the device sufficiently accurate even using data only from one microphone.

4. Conclusion

The present work is devoted to the QRD RLS lattice algorithm applied for the problem of hand detection and to implementation of the algorithm on the HW platform from Trenz Electronic. The platform comprises multi-core processor ARM Cortex A53 and FPGA programmable logic.

The algorithm is supposed to solve a hand detection problem using noise cancellation technique. For these purposes, from the family of RLS algorithms the QRD RLS lattice algorithm was chosen. The choice was conditioned by the inner structure of the algorithm, which allowed pipelining and parallel processing. The algorithm was incorporated with hypothesis testing to make hand detection even more precise. The special structure of regression models was proposed. Thus, two identification models corresponding to two different use-cases ("there is a hand in front of the device" and "there is no hand in front of the device") were designed. It was assumed that a regression model with a higher order describes situation when there is no hand in front of the device, while a regression model with a smaller order and time delay has higher probability when there is a hand in front of the device. The choice of the orders was conditioned, firstly, to cover all incoming data; secondly, by further steps of the research, i.e. pipelining and parallel processing. As an additional value of the developed algorithm was the possibility to compute the distance between the hand and the device.

The algorithm was implemented on four cores of the ARM device. The whole computational time constituted 58 s for DP version of the algorithm and 55.24 s for SP version of the algorithm. As far as real data measurement was 60 s, the algorithm implementation on the HW platform was considered to be fast enough to process data in real time and delivers results every 60 ms.

To conclude, the proposed method for a hand detection based on noise cancellation technique and using the QRD RLS lattice algorithm functions reliable and accurately and fulfils its goals.

The further step is to implement the algorithm on FPGA programmable logic part of the ARM device. It can possibly contribute into the algorithm acceleration.

Acknowledgement

This work has been supported from project StorAIge, project number ECSEL 101007321.

References

[1] ALESSANDRINI M., MARCHI DE L., SPECIALE N. Recursive Least Squares adaptive filters for ultrasonic signal deconvolution. *Conference: International Symposium on Circuits and Systems (ISCAS 2008)*. Seattle, Washington, USA: May, 2008, doi: 10.1109/ISCAS.2008.4542073.

[2] APOLINARIO J., DINIZ P.S.R. Fast QRD-RLS algorithms. In: *QRD-RLS adaptive filtering*. Springer: December, 2009, pp. 1–27, doi: 10.1007/978-0-387-09734-3_4.

[3] BENZIANE M., BOUAMAR M., MAKDIR M. Simple and efficient double-talk-detector for acoustic echo cancellation. *Traitement du Signal*, 37(4), pp. 585-592, doi: https://doi.org/10.18280/ts.370406.

[4] BOTTOMLEY G.E., ALEXANDER S.T. A novel approach for stabilizing recursive least squares filters. *IEEE Transactions on Signal Processing*, 1991, 39(8), pp. 1770–1779, doi: https://doi.org/10.1109/78.91147.

[5] CIOFFI J.M., KAILATH T. Fast recursive-least-squares transversal filters for adaptive filtering. *IEEE Transactions on Acoustics Speech and Signal Processing*, 1984, 32(2), pp. 304–337, doi: 10.1109/TASSP.1984.1164334.

[6] DINIZ P.S.R. Adaptive filtering: algorithms and practical implementation. Third edition, Springer: Boston, MA, 2008. doi: 10.1007/978-0-387-68606-6.

[7] DINIZ P.S.R. Adaptive Lattice-based RLS algorithms. *In: Adaptive Filtering*. Fourth edition, Springer, Boston, MA, 2013. doi: 10.1007/978-1-4614-4106-9_7.

[8] DINIZ P.S.R., SIQUEIRA M.G. Finite and infinite-precision properties of QRD-RLS algorithms. *In: QRD-RLS Adaptive Filtering*. Springer, Boston, MA: December, 2009, doi: 10.1007/978-0-387-09734-3_9.

[9] FABRE P., GUEGUEN C. Improvement of the fast recursive least-squares algorithms via normalization: a comparative study. *In: IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1986, 34(2), pp. 296–308, doi: 10.1109/TASSP.1986.1164813.

[10] GUPTA V.K., CHANDRA M., SHARAN S.N. Noise minimization from speech signals using RLS algorithm with variable forgetting factor. *In: Research Journal of Applied Sciences, Engineering and Technology*. 2012, 4(17), pp. 3102–3107.

[11] HARTFIEL J. TE0808 TRM. *In: Trenz Electronic Wiki* (last modified on June 01, 2020). Available at https://wiki.trenz-electronic.de/display/PD/TE0808+TRM

[12] HARTFIEL J. TEBF0808 TRM. *In: Trenz Electronic Wiki* (last modified on March 12, 2019). Available at https://wiki.trenz-electronic.de/display/PD/TEBF0808+TRM

[13] KADLEC J. Průběžná pravděpodobnostní identifikace autoregresního modelu s neznámým řádem. *In: Analýza, syntéza a rozpoznávání řeči*. ČSVTS, Praha, 1985.

[14] KADLEC J. Pravděpodobnostní identifikace regresního modelu v pevné řádové čárce. Praha, 1986.

[15] KÁRNÝ M. Bayesian estimation of model order. *In: Problems of Control and Information Theory*, 1980, 9(1), pp. 33–46, MR0561805.

[16] KÁRNÝ M., BÖHM J., GUY T. V., JIRSA L., NAGY I., NEDOMA P., TESAŘ L. Optimized Bayesian Dynamic Advising. *Theory and Algorithms, Springer*, London, 2006

[17] KIM J.Y., LEE Y. I., KIM K.J., NAM S.W., KO C.C. Utilization of a correlation RLS algorithm for nonlinear echo cancellation. *In: WSEAS Transactions on Circuits and Systems*, 2006, 5(4), pp. 435–441. Available at http://scholarbank.nus.edu.sg/handle/10635/57763

[18] KULHAVÝ R. Směrové zapomínání a průběžná identifikace systémů s pomalu se měnícími parametry. Výzkumná zpráva č. 1170, ÚTIA ČSAV, 1983.

[19] LI TS., TIAN K., LI WX. Method for improving RLS algorithms. *In: Journal of Marine Science and Application*, 2007, 6(3), pp. 68–70, doi: 10.1007/s11804-007-5077-x.

[20] MARTINEK R., KAHANKOVA R., NEDOMA J., FAJKUS M., SKACEL M. Comparison of the LMS, NLMS, RLS, and QR-RLS algorithms for vehicle noise suppression. *In: ICCMS 2018, Proceedings of the 10th International Conference on Computer Modeling and Simulation*, pp. 23–27, January, 2018, doi: 10.1145/3177457.3177502.

[21] IGLESIAS M.E. Implementation of QRD-RLS algorithm on FPGA. Application to noise canceller system. *In: IEEE Latin America Transactions*, 2011, 9(4), pp. 458–462, doi: 10.1109/TLA.2011.5993728.

[22] MATLAB. (9.7.0.1261785). (R2019b), Natick, Massachusetts: The MathWorks Inc., 2019.

[23] MOHAMMAD M., AL-SHEBEILLI S. Finite-precision analysis: Fast QR-decomposition algorithm. *In: The 10th IEEE International Symposium on Signal Processing and Information Technology*, 2010, pp. 161–165, doi: 10.1109/ISSPIT.2010.5711765.

[24] MOONEN M. Introduction to adaptive signal processing. K. U. Leuven, Leuven, Belgium, 1999.

[25] MUNJAL A., AGGARWAL V., SINGH G. RLS algorithm for acoustic echo cancellation. *In: Proceedings of the 2nd National Conference on Challenges and Opportunities in Information Technology (COIT-2008) RIMT-IET*, Mandi Gobindgarh, March, 2008. Available at https://www.researchgate.net/publication/228901207_RLS_algorithm_for_acoustic_echo_cancellation

[26] NISHIKAWA K., KIYA H. Novel implementation technique of RLS algorithm for improving throughput of adaptive filters. *In: 10th European Signal Processing Conference*, pp. 1–4, 2000. Available at https://ieeexplore.ieee.org/document/7075545/metrics#metrics

[27] PETERKA V. Bayesian approach to system identification. *In: Trends and Progress in System Identification* (P. Eykhoff, ed.), IFAC Series for Graduates, Research Workers and Practicing Engineers, Pergamon Press, 1981.

[28] POHL Z., TICHY M., KADLEC J. Implementation of the Least-Squares Lattice with order and forgetting factor estimation for FPGA. *:* In: EURASIP Journal on Advances in Signal Processing, 2008, 394201, doi: 10.1155/2008/394201.

[29] SETHY PRABIRA. Noise cancellation in adaptive filtering through RLS algorithm using TMS320C6713DSK. *In: International Journal of Electronics and Communication Engineering and Technology (IJECET*, 2012, 3(1), pp. 154–159. Available at https://www.researchgate.net/publication/272175478_NOISE_CANCELLATION_IN_ADAPTIVE_FILTERING_THROUGH_RLS_ALGORITHM_USING_TMS320C6713DSK

[30] REGALIA P.A. Numerical stability properties of a QR-based fast least squares algorithm. *In: IEEE Transactions on Signal Processing*, 1993, 41(6), pp. 2096–2109, doi: 10.1109/78.218139.

[31] SLOCK D.T.M., KAILATH T. (1991) Numerically stable fast transversal filters for recursive least-squares adaptive filtering. *In: Golub G. H., Van Dooren P. (eds) Numerical Linear Algebra, Digital Signal Processing and Parallel Algorithms. NATO ASI Series (Series F: Computer and Systems Sciences)*, vol. 70, pp. 605–615, Springer, Berlin, Heidelberg, doi: 10.1007/978-3-642-75536-1_49.

[32] SUKHUMALCHAYAPHONG S., BENJANGKAPRASERT C. Variable forgetting factor RLS algorithm for adaptive echo cancellation. *In: 14th International Conference on Control, Automation and Systems (ICCAS 20014)*, 2014, pp. 971–974, doi: 10.1109/ICCAS.2014.6987926.

[33] XU J., ZHOU W., GUO Y. A simplified RLS algorithm and its application in acoustic echo cancellation. *In: 2nd International Conference on Information Engineering and Computer Science*, 2010, pp. 1–4, doi: 10.1109/ICIECS.2010.5678354.