Combining Multiple Sound Sources Localization Hybrid Algorithm and Fuzzy Rule Based Classification for Real-time Speaker Tracking Application

Christian Ibala, Sergei Astapov, Frédéric Bettens, Fernando Escobar,
Xing Chang, Carlos Valderrama, and Andri Riid

***Abstract*—This work present a novel approach to track a specific speaker among multiple using the Minimum Variance Distortionless Response (MVDR) beamforming and fuzzy logic ruled based classification for speaker recognition. The Sound sources localization is performed with an improve delay and sum beamforming (DSB) computation methodology. Our proposed hybrid algorithm computes first the Generalized Cross Correlation (GCC) to create a reduced search spectrum for the DSB algorithm. This methodology reduces by more than 70 % the DSB localization computation burden. Moreover for high frequencies Sound sources beamforming, the DSB will be preferred to the MVDR for logic and power consumption reduction.**

***Index Terms*—DSB, GCC,Localization, Tracking, MVDR, Fuzzy Logic, Classification, speaker recognition, FPGA.**

I. Introduction

RAPID advancement in adaptive beamforming applications such as (sonar and radar) algorithms has greatly increased the computation and communication demands on beamforming arrays, particularly for applications that require autonomous and real-time computations. Parallel processing for adaptive beamformers can significantly reduce execution time, power consumption, cost and increase scalability and dependability [1]. Parallelism is well defined by Amdahl [2] and multiple papers defined power consumption control [3], [4], [5]. In this work the sound sources are captured with miniature electro-mechanical system microphones (MEMS microphones) which are configured as a linear acoustic array. After demodulation the microphones signals are transferred to the voice activity detector (VAD). An important problem in speech processing applications is the determination of active speech periods within a given audio signal. Speech can be characterized by a discontinuous signal since information is carried only when someone is talking [6]. Moreover speech activities normally occupy 60% of the time of a regular conversation. The VAD enables reallocating resources during the periods of speech absence [7] or disable the resources for power saving. Whenever the VAD detects the presence of speech in the audio signal it then triggers the localization of the sound sources.

However sequential implementation of beamforming algorithms with multiple microphones presents a significant computational challenge in real-time processing. Our contribution expands on the approach presented in [8] to multiple Sound sources:

Prior to the delay and sum beamforming (DSB) computation, all the Field Of View (FOV) is scanned using the Generalized Cross Correlation (GCC) methodology based on energy to find the direction of arrival (DOA) of the sound sources. Using the angles of arrivals ϕ1, ϕ2 of the sound sources provided by the GCC, the DSB search area is then restrained to those directions therefore its throughput improved.

Figure 1 shows a FOV with two sound sources located at (68 and 108) degrees of the microphone array center. The two bright cones represent the new DSB search spectrum after GCC computation. The DSB is then computed to locate the exact Sound sources position at (0.6 and 1) meters respectively. Our hybrid approach reduces the DSB throughput by reducing the FOV search spectrum. A FOV is the region in space where sound sources are susceptible to be found and the resolution is the smallest distinguishable region.

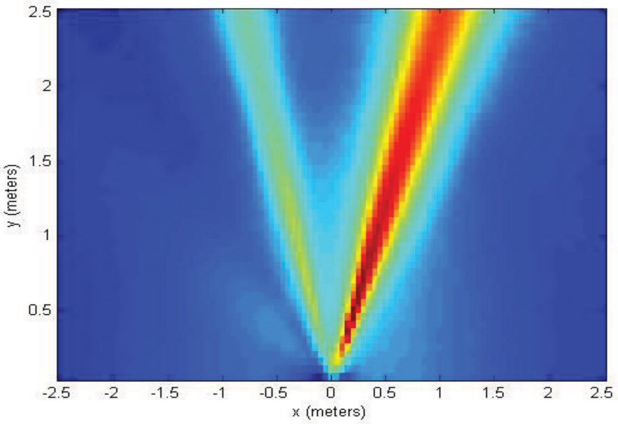


Figure 1. Two sound sourcesDOA, obtained with the GCC algorithm based on Energy.

Although DSB provides an accurate localization of sound sources it does not achieve a maximization of the Signal-to-Noise Ratio (SNR) especially in low frequency [9]. Therefore the Minimum Variance Distortion less Response (MVDR) is

Manuscript received March 4, 2013.

C. Ibala is with Department of Electronics and Computer Engineering, University of Limerick, Ireland (e-mail: sibala@acm.org)

S. Astapov and A. Riid are with Tallinn University of Technology in Estonia (e-mail: Sergei.astapov@dcc.ttu.ee)

F. Bettens, F. Escobar and C. Valderrama are with Faculty of Engineering, University of Mons, Belgium

used for low frequencies signals and DSB otherwise. This approach reduces logic resources while improving directivity. The directivity result is then used to identify the tracked speaker with a fuzzy rule based classification method.

To create a speaker voice model this work uses several types of spectral features [10], e.g. Mel-frequency Cepstral Coefficients (MFCC), that is extracted from temporal signal frames corresponding to specific person speech. The modeling of feature distribution in the feature space is somewhat similar to the one applied in Gaussian Mixture Models (GMM) that are frequently used in speaker identification systems. However, the application of the rule based approach instead of Maximum Likelihood (ML) for the classification procedure proves to be more robust [11]. The computationally expensive voice modeling process is performed during system off-line tuning. The classification procedure itself is very fast [12] and is able to be performed in real-time on embedded hardware.

The remaining paper is organized as follows: Section II describes the system. Section III presents Sound sources localization and beamforming algorithms used in this work. Section IV presents the proposed algorithms block diagram and their computation burden. Section V presents our contribution. Section VI presents the DSB and MVDR beamforming. Section VII explains the fuzzy logic ruled based classification applied to speaker recognition and tracking. Section VIII evaluate the results, discusses their limitation and proposes possible improvement. Section IX concludes the paper and advices on further work.

II. System Description

A. System Configuration

The localization system is composed of 8 equidistant microphones operating in a 3m by 3m FOV with a 10cm by 10cm resolution. For illustration purpose Figure 2 shows a miniature FOV composed of 4 microphones with two actives Sound sources (blue and red).

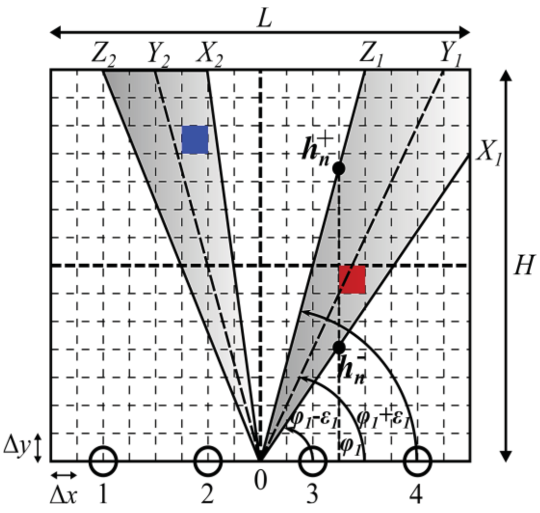


Figure 2. Two dimensions 16x16 small square FOV with 4 microphones and two speakers.

Referring to the FOV size and resolution given above the number of small square (NOSS) in the FOV is computed by equation (1). L.H= 9 $\Delta x \cdot \Delta y$ = 0.01. NOSS = 900.

$$NoSS = \frac{FOV}{Resolution} = \frac{L \cdot H}{\Delta x \cdot \Delta y} \tag{1}$$

Equation (2) models the far field approximation used in this work [13].

$$|r| > \frac{2(Nd)^2}{\lambda} \tag{2}$$

Where N represents the number of microphones, d the distance between microphones fixed to 4 cm, λ is the wavelength and r is the radial distance from the sound source to the microphone aperture.

B. System Constraints and Signals Model

In real-time applications execution speed is an important concept, the algorithm that requires the least logic resources and achieve the highest throughput for a given task is preferred. An algorithmic and architectural approach is proposed to respect real-time constraint. A frame of 512 or 1024 samples should be processed respectively at 11.6 or 23.16 ms for a sampling frequency of 44.1 KHz.

First a brief signals model and notation before describing localization algorithms [14] is presented. $(\cdot)^H$ denotes Hermitian transpose. $(\cdot)^*$ denotes the complex conjugate. Let $\{S_i(t)\}_{i=1}^{L}$ be the temporal waveforms of the sources, where L is the number of sources. The assumption is made that the sources are independent and stationary over several adjacent Ns samples intervals which is mathematically translated as: $E[S_i(n).S_j^*(n-l)] = 0$ The signal at the ith microphone is modeled as in equation (3).

$$X_i(t) = a_i s_i(t - \tau_i) + n_i(t) \tag{3}$$

Where a_i is the distance attenuation coefficients, t is the time index, τ_i is the time delays of arrival (TDOA) at the microphone and $n_i(t)$ is the noise sensed at the ith microphone.

III. Sound Source Localization and Beamfoming Algorithms

One of the most important functionalities of microphone arrays is to extract the speech of interest from its observation corrupted by noise, reverberation, and competing sound sources. This is done by aiming the beam towards the desired sound source [15]. The purpose of any beamforming algorithm is to determine the DOA of one or more signals [16]. Multiple localization algorithms are explored in this work, the MVDR the GCC and DSB are combined to create a robust tracking algorithm for a real-time application.

There are two major groups of microphone-array processing algorithms: time-invariant and adaptive [17]. The first group is fast and simple to get a real-time implementation. The second group acoustic adaptive algorithms are able to automatically adapt their response to different weightings or time-delays. However, they require more CPU power and are complex to implement. In this work both approaches are used.

1) GCC-PHAT

The Generalized Cross Correlation (GCC) algorithm returns an angle ф which is the sound source DOA. To compute ф, the GCC uses the estimation of the temporal shift between two microphones that lead to the maximum cross-correlation function between them as in equation (4):

$$R(k) = IFFT(\frac{FFT(x_i(t)).FFT(x_j^*(t))}{\left|FFT(x_i(t)).FFT(x_j^*(t))\right|^{\beta}}) \qquad (4)$$

where β is a coefficient factor in the interval of] 0, 1[, $x_i(t)$ and $x_j(t)$ are the signals at the microphone (i,j), t and k are time index. For a single sound source, the DOA can be estimated by finding the index of the maximum coefficient of R(k) which is modeled as in equation (5).

$$\Delta_{ij} = \arg_k \max R(k) \qquad (5)$$

The Sound source DOA is modeled as in the equation (6)

$$\phi_{ij} = \arccos(\frac{c.\Delta_{ij}}{F_s.d}) \qquad (6)$$

C is the Sound propagation speed and Fs is the sampling frequency. However when they are multiple sound sources, the estimation of the DOAs is very difficult due to the cross-correlation among different sound sources. For example let $s_1(t)$ and $s_2(t)$ be the signals that come from two sound sources. Then the signals received at the microphone $x_i(t)$ and $x_j(t)$ are written as in equation (7) at the microphone i.

$$X_i(t) = a_i s_i(t-\tau_i) + b_i s_j(t-\varsigma_i) \qquad (7)$$

As in equation (8) at the microphone j

$$X_j(t) = a_j s_i(t-\tau_j) + b_j s_j(t-\varsigma_j) \qquad (8)$$

where t is the time index, τ and ς are the time delay of arrival, a,b are the distance attenuation coefficients. Equation (4) numerator is then modeled as in equation (9) [18].

$$\begin{aligned} &FFT(x_i(t)).FFT(x_j^*(t)) = \\ &a_i a_j X(w)^2 e^{-jw(\tau_i-\tau_j)} + b_i b_j Y(w)^2 e^{-jw(\varsigma_i-\varsigma_j)} + \\ &X(w).Y(w).(a_i a_j e^{-jw(\tau_i-\varsigma_j)} + a_j a_i e^{-jw(\tau_i-\varsigma_j)}) \end{aligned} \qquad (9)$$

The GCC method can accurately estimate the DOAs when two signals are uncorrelated. However, when two signals are correlated as in the real environments, the GCC method fails to estimates the correct DOA [18]. Therefore an energy based computation approach will be proposed using the GCC results.

2) DSB - SRP (Time Domain Approach)

The DSB is a beamforming algorithm that can be used in conjunction with a FOV to compute a Steered Response Power (SRP). The point of the FOV with the highest SRP is the sound source location. The SRP is computed as in equation (10).

$$SRP_i = \sum_{t=1}^{Ns}(\sum_{n=1}^{N} w_{in} x_n(t-\tau_{in}))^2 - \sum_{n=1}^{N} w_{in}^2 x_n^2(t-\tau_{in})] \qquad (10)$$

The SRP is computed for every point *i* of the NOSS that vary as follow: $(0 < i < NOSS)$. w_{in} is the weight of the point *i* relative to microphone *n* computed as in equation(11).

$$w_{in} = \frac{1/d_{in}}{\sum_{n=1}^{N} 1/d_{in}} \qquad (11)$$

d_{in} is the distance of the i^{th} small square (SS) to the n^{th} microphone. It is computed as in equation (12):

$$d_{in} = \sqrt{(x_i - x_n)^2 + (y_i - y_n)^2} \qquad (12)$$

The pairs (x_i, y_i) and (x_n, y_n) are respectively the coordinates of point *i* and the microphone *n*.

3) MVDR

The main approach to find the DOA of the sound source using MVDR is to steer at every direction of the FOV and compute equation (13). The DOA of the audio signal is found when equation (13) reaches its maximum [19][20].

$$P(\phi) = \frac{1}{k} \sum_{i=0}^{Ns-1} \left|Y(f)\right|^2 = \frac{1}{d^H .R_{xx}^{-1}.d} \qquad (13)$$

Equation (14) is the signal R_{xx} coherent matrix.

$$R_{xx} = \begin{pmatrix} 1 & \gamma_{x0x1} & \cdots & \gamma_{x0xN-1} \\ \gamma_{x1x0} & 1 & \cdots & \gamma_{x1xN-1} \\ \cdots & \cdots & \cdots & \cdots \\ \gamma_{xN-1x0} & \gamma_{xN-1x1} & \cdots & 1 \end{pmatrix} \qquad (14)$$

γ_{x1x0} is the normalized correlation between the microphone (0) and (1) and defined as in equation (15):

$$\gamma_{x0x1}(f) = \frac{\gamma_{x0x1}(f)}{\sqrt{(\gamma_{x0x0}(f).\gamma_{x1x1}(f))}} \qquad (15)$$

d represents the propagation vector of the desired speech signal for a linear sensor array and is defined in equation (16).

$$d = [\alpha_1 e^{-\delta(d_1-d_0)}, \ldots, 1, \ldots, \alpha_N e^{-\delta(d_n-d_0)}]^T \qquad (16)$$

In the far-field approximation the coefficients $(\alpha_1..\alpha_N)$ are approximated to 1. δ is computed as in equation (17).

$$\delta = \frac{2\pi f \cos\phi}{c} \tag{17}$$

W_{MVDR} is the MVDR weight modeled as in equation (18).

$$W_{MVDR}^H = \frac{d^H.\Gamma_{vv}^{-1}}{d^H.\Gamma_{vv}^{-1}.d} \tag{18}$$

When replacing γ_{x0x1} by γ_{v0v1} equation (14) becomes the noise coherent matrix Γ_{vv} .The microphone array will receive noise signals that are mainly correlated at low frequencies and have approximately the same energy. The complex coherence function for such a noise field can be approximated as in equation (19):

$$\gamma_{v_i v_j}(f) = \frac{\sin(2\pi f d_{ij}/c)}{2\pi f d_{ij}/c} \tag{19}$$

where d_{ij} is the distance between the sensors (i,j) and f is the frequency [19]. Equation (19), in practice will have the tendency to amplify low frequency noise. To work around this issue literature propose to introduce the uncorrelated noise variance (σ_n^2) of the sensors in the computation of the coherence function modeled as in equation (20).

$$\gamma_{v_i v_j}(f) = \frac{\sin(2\pi f d_{ij}/c)}{2\pi f d_{ij}/c(1+\sigma_n^2/p_{nn}(f))} \tag{20}$$

IV. Block Diagramand Algorithm Computational Comparison

The DSB, GCC and MVDR are respectively presented in Figure 3, 4 and 6. They all share the same demodulator (Δ-Σ), framing, Voice Activity Detector (VAD) and the FFT. Figure 3 shows a proposed block diagram to implement and specially analyze both the GCC and DSB algorithms in terms of their computational complexity. The branch where the output is an “incidence angle”, represents the GCC and the one with “source localization” output represents the DSB.

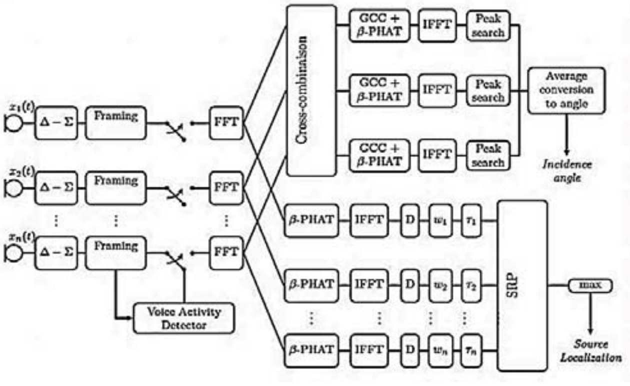


Figure 3. GCC (upper branch) and DSB-Donohue approach (lower branch)functional block diagrams.

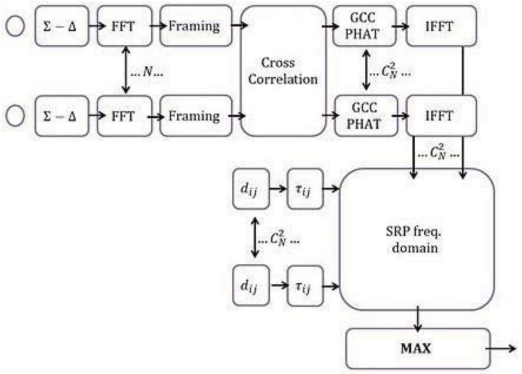


Figure 4. DOA using GCC computation of the *Steered Power Response*.

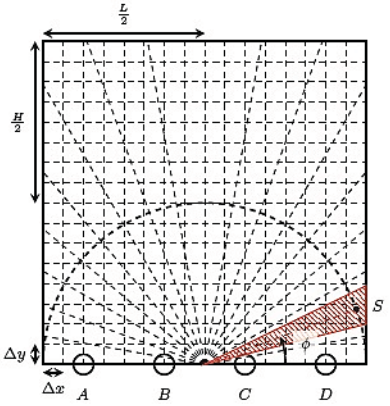


Figure 5. GCC based energy computationin every direction of the FOV.

Figure 3 GCC computation approach is only valid when one sound source is present in the FOV, for multiple sound source tracking the GCC computation using Figure 4 block diagram based on energy is computed for every angle of the FOV see Figure 5. Figure 5 approach is based on space discretization (SD) and modeled as in equation (21). The points such as S which return the highest energy represents the Sound sources (DOA).

$$E(S) = \sum_{r=1}^{N-1} \sum_{s=r+1}^{N} \sum_{k=0}^{N_S-1} W_{rs}(f).X_r(f).X_s{}^*(f).\exp(-j2\pi f(\tau_r - \tau_s)) \tag{21}$$

Where XrXs is the cross-correlation between microphones signals (r,s), W_{rs} is the denominator of equation (4) and $\tau_r - \tau_s$ is defined as in equation (22) [18].

$$(\tau_r - \tau_s) = \frac{d_{rs}}{c}.fs \tag{22}$$

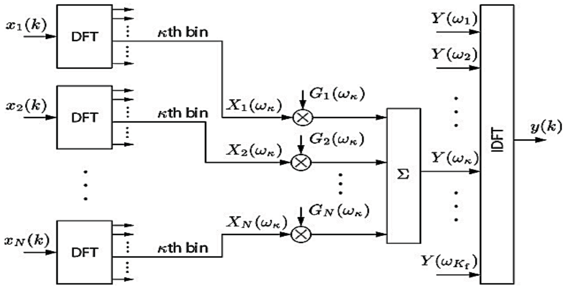


Figure 6. Structure in Frequency-domain broadband beam former by Narrow band decomposition [15].

A. DSB Computation Load and Problematic

The DSB algorithm combines accurate sound source localization with the flexibility of having pre-computed coefficients. Those coefficients (weight, delay etc…) could then be stored in an FPGA BRAM or in an external memory. The numbers in Table I are explained in [8]. Table I shows the number of DSB operation depending of the NOSS.

TABLE I
DSB COMPUTATION BURDEN NS = 512 N = 8

Localization	NOSS = 900	NOSS = 256
BLKREAD	11 074 500	3 150 080
MULT	15 207 300	4 325 632
ADD	11 059 200	3 145 728
DIV	7 372 800	2 097 152
SQRT	3 686 400	1 048 576

The DSB computation burden is mainly linked to the NOSS defined in equation (1), to the number of microphones (N) and the frame length (Ns). Other parameters have a slight impact such as the number of sound sources (L) and the algorithm used. The DSB throughput is computed using equation (23).

$$NCCF = Mult + Add + Blk_{read} + u.Div + v.Sqrt) \qquad (23)$$

where u and v are respectively the number of clock cycles necessary to compute a division and square root. In the Table II u and v are considered to be equal to 1 for combinatorial IP core or more for sequential. DSB is used for localization as it is considered to be one of the most robust [14] algorithms. However its computation is tedious and long. Table II shows that for a system of 8 microphones with a NOSS = 900 even with a clock speed of 600 MHz it is impossible to achieve real-time localization.

TABLE II
DSB THROUGHPUT WITH NOSS = 900 AND N = 8 REAL-TIME = 11.6 MS

DSB (Clock Speed)	200 MHz	400 MHz	600 MHz
Throughput	242 ms	121 ms	80.5 ms

B. GCC-PHAT Computational Load

The computational load of the GCC-PHAT based on spaced discretization (SD) is linked to the angular space cover by the FOV and to the number of cross-correlation between microphones modeled by equation (24).

$$C_N^P = \frac{N!}{P!\,(N-P)!} \qquad (24)$$

P equals 2 and N is the number of microphones in the array. For N equal to 4, 8, 16, 32 or 64, the cross-correlation C_N^P, will respectively be 6, 28, 120, 496, 2016. The CPU power increases drastically with the number of microphones and the angular region cover. Table III shows the number of sequential operations necessary to compute the GCC using the SD approach with ε varying from 0 to 180.

TABLE III
GCC COMPUTATION BURDEN BASED ON SPACE DISCRETIZATION N = 8

Localization	GCC + β =1 + Space Scan
BLKREAD	19 404
MULT	86 016
ADD	54 168
DIV	28 672
SQRT	14 336

The GCC based on SD computation load in Table III is smaller compare to the DSB in Table I and its throughput is computed by using Table III and equation (23). Table IV shows that less than 10 % of the time required for real-time processing is necessary to find the sound source DOA. This result will inspire our hybrid algorithm.

TABLE IV
GCC DISCRETIZATION APPROACH SEQUENTIAL THROUGHPUT

GCC (Clock Speed)	200 MHz	400 MHz	600 MHz
Throughput	1.1 ms	0.51 ms	0.38 ms

C. MVDR Computational Load

To minimize sound source localization time, MVDR is not considered due to its complexity as shown by equation (13) computation steps below.

1) Compute equation (14) correlation matrix whose parameters are defined in equation (15), (16) and (17).
2) Check if the correlation matrix of equation (14)is invertible using equation (25).

$$\det(\Gamma_{xx}) \neq 0 \qquad (25)$$

3) Compute equation (14) denominator and its inverse.
4) For all direction of ϕ repeat the points (2) and (3).

The three steps described above are time and hardware consuming. Thus the MVDR is only used for beamforming while the GCC and DSB are used for real-time localization.

V. CONTRIBUTION

For any clock speed, the DSB throughputs are far superior to the 11.6 ms real-time constraint as shown by Table II. Based on this challenge our contribution will be presented. To accelerate sound sources localization, this work proposes at the algorithmic level a hybrid algorithm that reduces the DSB NOSS and at the architectural level increasing buffer size provides more computation time.

A. Algorithmic Contribution

To detect the DOA of the sound sources, the entire FOV region is scanned and the highest energies are selected (see Figure 6). Then the search region is restricted to the cone delimited by the angles ϕ ± ε of each sound DOA (see Figure 2). ε is the localization error which can be limited to one or two degree for the primary source and a little bit more for the secondary source as the FOV region scanning is done with one degree step. This approach restricts the DSB search spectrum which can be mathematically defined using the left and right upper corner of Figure 2 denoted respectively δ_1 and δ_2 modeled as in equation (26) and (27),

$$\delta_1 = \arctan(\frac{H}{L/2}) \qquad (26)$$

The first search region is delimited by {0, X1, Y1, Z1} which represents ($\phi - \varepsilon < \delta_1$ and $\phi + \varepsilon > \delta_1$) or the region:

$$\{0,0\};\left\{\left(\frac{L}{2}\right),\left(\frac{L}{2}\cdot\tan(\phi-\varepsilon)\right)\right\};\left\{\frac{L}{2},H\right\};\{H\cdot\text{cotan}(\phi+\varepsilon)\,,H\}$$

$$\delta_2 = 180 - \arctan(\frac{H}{L/2}) \qquad (27)$$

The second search region is delimited by {0, X1, Y2, Z2} which represents ($\phi - \varepsilon < \delta_2$ and $\phi + \varepsilon > \delta_2$) or the region:

$$\{0,0\};\{-H\cdot\text{cotan}(\phi-\varepsilon)\,,H\};\left\{\left(\frac{-L}{2}\right);H\right\}\left\{\left(\frac{-L}{2}\right),\left(\frac{-L}{2}\right)\cdot\tan(\phi+\varepsilon)\right\}$$

These search regions must be redefined for each process frames. In each region the NOSS is computed using equation (28) and (29).Equation (28) is the higher line of the cone.

$$h_n^+ = n\Delta x \tan(\phi + \varepsilon) \qquad (28)$$

Equation (29) is the lower line of the cone

$$h_n^- = n\Delta x \tan(\phi - \varepsilon) \qquad (29)$$

With an estimation error of ε = 2° for the primary source and 10° for the secondary source, the NOSS of both cone in Figure 2 is 90. From a NOSS = 900 to 90 the algorithmic approach reduce the DSB search spectrum of 90%. After the algorithmic contribution, Table II is re-computed with the new NOSS, the localization throughput is within the real-time constraint for the 600 MHz clock see Table V. The architectural approach will then be applied on the algorithmic result to achieve real-time with a slower clock and reduce power consumption.

TABLE V
DSB LOCALIZATION THROUGHPUT WITH NOSS = 90 AFTER HYBRYD ALGORITHMIC COMPUTATION + TABLE IV RESULT

DSB (Clock Speed)	200 MHz	400 MHz	600 MHz
Throughput Serial	25.3 ms	12.5 ms	8.5 ms

B. Architectural Contribution

The flexibility of the hardware structure proposes to implement our hybrid algorithm which can be altered to respect the real-time constraint. Figure 4 block diagram has two parts: The acquisition modules composed of {Microphone, sigma delta filter and VAD} and the computation modules composed of {FFT, IFFT, β-PHAT, delay and Sum and SRP}.

1) Acquisition Modules

For the {Microphone, Sigma Delta, framing} modules few can be done to improve their flexibility; however the VAD and buffer storage can be duplicated. As stated above speech activities normally occupy 60% of the time of a regular conversation. Therefore in a 1024 or 2048 buffer use to collect data half of them only are usable see Figure 7.

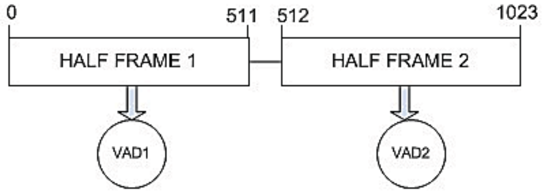


Figure 7. Internal structures of buffer storing data before VAD computation

Figure 7 presents the hardware structure of the VAD. Only 512 samples will be processed out of 1024 samples collected. Each half frame VAD is computed as in equation (30). The half frame with the highest VAD is processed if superior to the VAD threshold.

$$VAD_{tresh} = \mu + cst \cdot \sigma \qquad (30)$$

where μ and σ^2 are the mean and variance modeled asin equation (31), (32) and cst is a constant with a value≥ 3.

$$\mu = \frac{\sum_{i=0}^{Ns} x_i}{Ns} \qquad (31)$$

$x_i$is the value of sample i

$$\sigma^2 = \frac{\sum_{i=0}^{Ns} x_i^2}{Ns} - \mu^2 \qquad (32)$$

The first architectural contribution was to increase the acquisition buffer size to relax the real-time constraint. The time to complete the computation is then increased to 23.21ms for a 512 samples frame. Thus the systems clock can be reduced to 400 MHz and respect real-time constraint as shown in Table V.

2) Computation Modules

The second architectural contribution is to duplicate the Delay and Sum modules composed of {D, W, τ} and SRP in Figure 4 to localize the sound sources in parallel. Table VI shows that combining an hybrid algorithm to a flexible hardware improves drastically the throughput of the system. A margin 10.65ms is gained compared to the real-time constraint in the worst case scenario 18.96 ms in the best.

TABLE VI
DSB THROUGHPUT WITH NOSS = 90 WITH FLEXIBLE HARDWARE

DSB (Clock Speed)	200 MHz	400 MHz	600 MHz
Throughput Parallel	12.65 ms	6.25 ms	4.25 ms

Other modules such as: FFT, IFFT or β-phat have a limited impact on the throughput. The FFT computation is modeled as in equation (33).

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft}dt \qquad (33)$$

where x(t) is the input signal, X(f) the spectrum and f the frequency. For computation speed, a Fast Fourier Transform (FFT) which is a fast DFT algorithm that reduces the computing burden from N^2 to $\mathrm{N}\cdot\log_2\mathrm{N}$ is used. Since FFT processors using radix-4 architecture have fewer multiplications than processors using radix-2 [23], they are preferred in order to reduce the memory access rate and arithmetic workload, hence, power consumption. After FFT the β-PHAT is computed as in equation (34) see Figure 4.

$$W(f)=\frac{X(f)}{|X(f)|^{\beta}} \qquad (34)$$

The modified spectrum W(f) is then used as an input to the IFFT using equation (35) before computing the SRP defined in equation (10).

$$w(n)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}W(f)e^{2\pi ft}\mathrm{df} \qquad (35)$$

IFFT can be implemented re-using FFT modules by inverting the imaginary and real part as shown in Figure (8). This methodology increase modules re-usability. More on increasing FFT computation speed is found in [24].

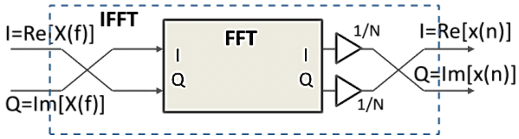


Figure 8. Computing the IFFT with FFT processing element.

VI. Beamforming Using DSB or MVDR

The signal frequency band needs to be determined to select which of the DSB or MVDR beamforming to use.

A. *Sound Frequency Band Determination*

To detect the signal frequency equation (36) or (37) is computed. The frequency band[0.3..1.5] is considered low frequencyand [1.5..to higher] KHz high Frequency.

$$X_{BE}(i)=\frac{\sum_{l\in S_i}|X(l)|^2}{\sum_{k=1}^{K}|X(k)|^2} \qquad (36)$$

S_i denotes the low frequency band and the denominator is the signal entire spectrum. Although equation (36) is a good approach the division's computation in hardware is costly. Therefore equation (37) is preferred. where p is the number of low frequency bin and TH is fixed to 0.5.

$$X_{SR}=\arg\max_{p}(\sum_{l=1}^{p}|X(l)|^2 \le TH.\sum_{k=1}^{K}|X(k)|^2) \qquad (37)$$

B. *DSB Beamforming*

The DSB beamforming is known to require only limited logic resources for its implementation (see Figure 9) [15].

On the other hand DSB beamforming do not uniformly attenuate the noise and interference signals coming from direction different from the beamformer's look direction as it was developed for narrow band. One way to circumvent this problem is to perform narrowband decomposition and design narrowband beamformers independently at each frequency, as shown in Figure 7 [15]. Table VII shows the computation load of the DSB.

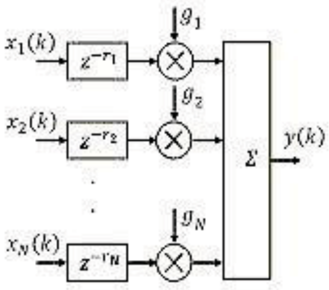


Figure 9. Structure of the Delay and Sum Beamformer[15].

TABLE VII
DSB BEAMFORMING THROUGHPUT FOR N = 8 AND NS = 512

OPERATION	DSB Beamforming	Burden
MULT	N.Ns	4 096
ADD	Ns(N-1)	3 584
DIV	0	64
BLK-READ	N(1+Ns)	4 104

C. MVDR Beamforming

MVDR beamforming is computed as shown in Figure 7, for low frequency signals, as it performs better than the DSB. However the MVDR computation load compared to the DSB is colossal as shown by Table VII compared to Table VIII. The DSB and MVDR beamforming computations are not dependent on the NOSS but mainly on the frame size and number of microphones.

TABLE VIII
MVDR BEAMFORMING SERIAL THROUGHPUT FOR N = 8 AND NS = 512

OPERATION	MVDR BEAM	Computation
MULT	(2N+1)N.Ns	69 632
ADD	Ns(N-1)(1+2N)	60 928
DIV	Ns*N	4 096
BLK-READ	N.Ns(N+4)	49 152

MVDR is often coupled to wiener filter. Using wiener post filter was suggested by Bitzer and McCowan [19, 24, 25] to improve the global performance of the beamformer. Simmer et al express the optimal broadband Minimum Mean Square Error (MMSE) filter solution as a classical Minimum Variance Distortionless Response (MVDR) beamformer followed by a single-channel Wiener which is modeled as in equation (38).

$$W_{MMSE}^{H}=\frac{d^{H}\Phi_{vv}^{-1}}{d^{H}\Phi_{vv}^{-1}d}\cdot\left(\frac{\phi_{ss}}{\phi_{ss}+\phi_{nn}}\right) \qquad (38)$$

where W_{MMSE}^{H} is the optimal filter coefficient vector, ϕ_{ss} and ϕ_{nn} are respectively the (single-channel) target signal and noise (after the MVDR noise filtering) auto-power spectrum vectors, and Φ_{vv} is the (multichannel) noise cross-spectral density matrix. The bracketed item in the equation (38) is the single-channel Wiener filter part and the remaining item is the well known MVDR beamformer [26]. The bracketed expression of (38) can be seen as a Wiener transfer function modeled as in equation (39).

$$H = \left(\frac{\phi_{ss}}{\phi_{ss} + \phi_{nn}} \right) \quad (39)$$

Solving equation (39) required expressing all the different parameters. Few assumptions regarding our noise field working environments need to be described. The target signal and the noise are uncorrelated, the noise power spectrum is the same on all the sensors and the noise is uncorrelated between sensors. Under a stationary environment noise, with the noise spectral density power (φ_{vv}), we can express the noise spectral density ϕ_{nn} as in equation (40)

$$\phi_{nn} = \left(\frac{\varphi_{vv}}{d^H . \Gamma_{vv}^{-1} . d} \right) \quad (40)$$

Using all the assumptions above ϕ_{ss} could be approximated as in equation (41) below:

$$\hat{\phi}_{ss}^{ij} = \max\left(\frac{\Re\left|\phi_{x_i x_j}\right| - \frac{1}{2} R(\gamma_{v_i v_j})(\phi_{x_i x_i} + \phi_{x_j x_j})}{1 - \Re(\gamma_{v_i v_j})}, 0 \right) \quad (41)$$

$\hat{\phi}_{ss}^{ij}$ is an estimation of ϕ_{ss} using the microphone i and j. $\Re\left| \; \right|$ mean positive real value, the imaginary part is considered to be zero. For N microphones there will be C_N^2 ways to estimate ϕ_{ss} . Taking the average improve system robustness at the cost of computations load and resource utilization. That approach is modeled in equation (42)

$$\hat{\phi}_{ss} = \frac{2}{N(N-1)} \sum_{i=0}^{N-2} \sum_{j=j+1}^{N-1} \hat{\phi}_{ss}^{ij} \quad (42)$$

The same approach is used to estimate the noise spectral density. It is modeled as in equation (43) and the average value is computed using (44) with the same combination C_N^2 .

$$\hat{\phi}_{vv}^{ij} = \max\left(\frac{\frac{1}{2} R(\phi_{x_i x_i} + \phi_{x_j x_j}) - R\left|\phi_{x_i x_j}\right|}{1 - R(\gamma_{v_i v_j})}, 0 \right) \quad (43)$$

Therefore equation (44) is the average of equation (43).

$$\hat{\phi}_{vv} = \frac{2}{N(N-1)} \sum_{i=0}^{N-2} \sum_{j=j+1}^{N-1} \hat{\phi}_{vv}^{ij} \quad (44)$$

ϕ_{nn} is then modeled as in equation (45)

$$\phi_{nn} = \begin{cases} \phi_{vv'} & f \le f_1 \\ \phi_{nn'} & f \ge f_1 \end{cases} \quad (45)$$

Adding Wiener in front the MVDR will not increase the system throughput as the necessary parameters are computed prior to the MVDR during the GCC as shown in equation (4) and (21).

VII. Fuzzy Logic Algorithm for Sound Source Tracking

After the sound sources have been localized and separated (i.e. the spectral pattern of each one is purged from the patterns of the rest via beamforming), the signals incoming from localized sound sources are put through the classification process. Our speaker identification application consists of two distinct stages. These are: feature extraction, where the spectrum of the speaker sound signal is transformed into a shorter set of features, that reflect the spectral pattern in a compact manner; and classification, during which every set of features is assigned a class label, representing the person from the knowledge base.

A. Feature Extraction

Feature extraction is performed in order to obtain a compact representation of the signal temporal or spectral pattern. This work will focus on pattern analysis in the frequency domain to save computation time. There is a great variety of spectral features that may be used for pattern extraction [27]. In our work we focus on a well known technique, which is called Mel-Frequency Cepstral Coefficients (MFCC). It is proven to perform well for human voice feature extraction and is applied in many audio signal processing applications [28]. The MFCC is executed in several stages, which are presented in the flow chart of Figure 10. The temporal (preprocessed) signal frame is first passed through the FFT to obtain its complex spectrum.

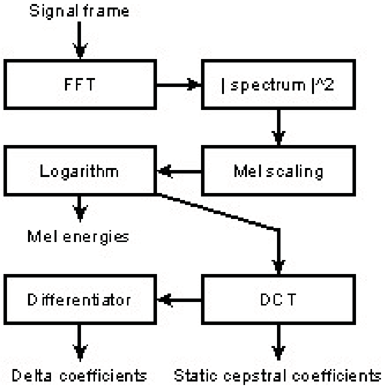


Figure 10. Flow chart of MFCC computation.

The absolute value of the spectrum is then squared for the real power spectrum defined as in equation (46).

$$P_i(k) = \frac{1}{N_s} \left| X_i(k) \right|^2 \quad (46)$$

The power spectrum is then transformed into the mel-scale, defined and modeled as in equation (47).

$$f_{mel} = 2595 \cdot \log_{10}\left(1+\frac{f}{700}\right) \tag{47}$$

f is the frequency in Hz and f_{mel} is the frequency warped to the mel-scale. This scale models the human auditory system which interprets the lower portion of frequencies better than the higher one and thus the distribution of higher frequencies is less in the mel-scale. Consider the mel-curve of Figure 11. (upper) is almost linear up to 1 kHz and logarithmic thereafter. The power spectrum is usually warped to mel-frequencies by applying a filter bank of triangular overlapping windows (Figure 12. lower) modeled as in equation (48).

$$H(k,m)=\begin{cases} 0 & f(k)<f(m-1) \\ \dfrac{f(k)-f(m-1)}{f(m)-f(m-1)} & f(m-1)\le f(k)<f(m) \\ \dfrac{f(m+1)-f(k)}{f(m+1)-f(m)} & f(m)\le f(k)<f(m+1) \\ 0 & f(k)\ge f(m+1) \end{cases} \tag{48}$$

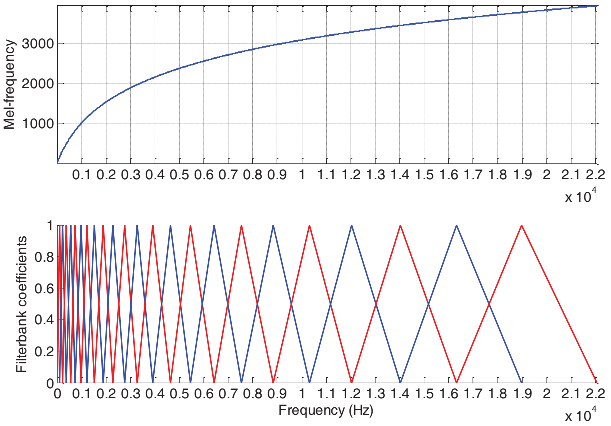


Figure 11. Mel-scaled frequencies (upper), mel filter bank (lower).

The number of filter banks (in Figure 11 lower) specify the Mel-energies vector length which vary from (20...40) and are modeled as in equation (49).

$$E(m)=\sum_{k=1}^{(Ns/2)+1} H(k,m).P_i(k) \tag{49}$$

The cepstral coefficients are acquired by applying the Discrete Cosine Transform (DCT) to the mel energies using equation (50).

$$. \quad C(l)=\sum_{m=1}^{M}\log_{10}.E(m).\cos\left[l(m-1/2)\frac{\pi}{M}\right] \tag{50}$$

For l = 1, 2, . . . ,M, where c(l) is the lth MFCC, M is the required number of MFCC parameter, E_k is the power spectrum coefficient[29-30]. The cepstrum holds information on the spectral harmonics, e.g. in the case of human voice emphasizes the voice pitch. One additional step may be performed on the cepstral coefficients, a differentiator over several successive cepstral frames can be computed to get the delta coefficients, which account for dynamic information of coefficient variation. It is modeled as in equation (51) [29].

$$\Delta C(m)=\frac{\sum_{i=1}^{J} i(c[m+i]-c[m-i])}{2.\sum_{i=1}^{J} i^2} \tag{51}$$

where 2j+1 is the size of the regression window and c[m] is the m^{th} MFCC coefficient [31]. The Mel energies, cepstral coefficients and deltas may be used as separate sets of features or concatenated into solid feature vector for future analysis.

B. *Fuzzy Classification*

The principal task of the classification algorithm is to determine the likelihood of an incoming sample of speech belonging to any of the predefined classes of speakers as recorded in the knowledge base of the classifier. For speaker identification, each speaker's voice is recorded prior to online identification, the speech portions of the recorded signal are put through feature extraction, concatenated into the dataset and complemented with class labels, that in the future will represent the specific speakers. The main features extracted using MFCC approaches are: Mel-Energies, Static Cepstral Coefficients and delta coefficients (see Figure 10). In this work, only the Mel-Energies and Static Cepstral Coefficients are considered to constitute the reference model which is defined as in equation (52). The entire rule base is optimized offline therefore it does not impact the application throughput.

$$CL(T)=\begin{cases} Y_T & Z_T \\ Mel_{11}..Mel_{1V} & CC_{11}..CC_{1V} \\ Mel_{21}..Mel_{2V} & CC_{21}..CC_{2V} \\ . & . \\ Mel_{n1}..Mel_{nV} & CC_{n1}..CC_{nV} \end{cases} \tag{52}$$

where Y_T and Z_T are the object containing the MEL and Static Cepstral Coefficients of length V of n-th frames. The fuzzy rule based classifier approach used in this work is highly comprehensive for manual data model analysis, unlike the black box structure of an Artificial Neural Networks (ANN) mapping and also computationally lightweight [32]. The MEL and Static Cepstral Coefficients are concatenated as in equation (53) prior to speaker recognition computation.

$$G=\{g_1,g_2...g_F\}=\{Mel_{11}...Mel_{1V},CC_{11}...CC_{1V}\} \tag{53}$$

For better understanding let us consider a classification of a feature vectors of length 2 (i.e. F = 2) which means that equation (53) has one Mel and one MFCC coefficient. Let assume that there are two speakers or classes in the database m (T = 2). These two classes are determined by the following rules:

Rule 1: If (g1) is A_{11} and (g2) is A_{21} then y belong to class CL (1) and

Rule 2: If (g1) is A_{12} and (g2) is A_{22} then y belong to class CL (2).

where A_{ir} is the linguistic term of the i^{th} input (i.e. feature vector element) associated with the r^{th} rule and CL(r) $c_r \in (1,...,T)$ is the class label assigned to the r^{th} rule ($i = 1,...,F$). Each linguistic term A_{ir} is numerically represented by a membership function μ_{ir} (MF), such as a typical triangle-shaped MF determined by three parameters a_{ir} , b_{ir} , c_{ir} (right base, peak and left base of the triangle, respectively) that are determined using the speaker database model. Equation (54) checks to which triangle the incoming database belong.

$$\mu_{ir}(g_i) = \begin{cases} \dfrac{g_i - a_{ir}}{b_{ir} - a_{ir}} & a_{ir} < g_i < b_{ir} \\ \dfrac{c_{ir} - g_i}{c_{ir} - b_{ir}} & b_{ir} < g_i < c_{ir} \\ 0 & (g_i \le a_{ir}) \vee (c_{ir} \le g_i) \end{cases} \tag{54}$$

Let consider two incoming feature vectors, one represented by a star in Figure 12, which belongs to class 2 and the second, represented by a pentagon, which does not belong to any class. For the star, the MF values for the terms A_{11}, A_{21}, A_{12} and A_{22} are: $\mu_{11} = 0$, $\mu_{21} = 0$, $\mu_{12} = 0.3$ and $\mu_{22} = 0.8$ respectively. The class label is assigned in a winner-takes-it-all manner, where the final label is specified by the rule with the highest activation degree τ_r as in equation (55).

$$y = c_r, \arg\max_{1<r<R}(\tau_r) \tag{55}$$

where τ_r is defined as in equation (56)

$$\tau_r = \bigcap_{i=1}^{F} \mu_{ir}(g_i) \tag{56}$$

where $\bigcap_i^F$ is the conjunction operator corresponding to the linguistic operator AND (in our case a product operator)[33]. Thus the activation degree for rule 1 is $\tau_1 = \mu_{11} \cdot \mu_{21} = 0$ and the activation degree for rule 2 - $\tau_1 = \mu_{12} \cdot \mu_{22} = 0.24$. The class label for the star feature vector is then $y = \arg\max\{0, 0.24\} = 2$. Similarly for the pentagon feature vector the MF values for the terms A_{11}, A_{21}, A_{12} and A_{22} are $\mu_{11} = 0$, $\mu_{21} = 0$, $\mu_{12} = 0$ and $\mu_{22} = 0.4$. The activation degree for rule 1 is $\tau_1 = \mu_{11} \cdot \mu_{21} = 0$ and the activation degree for rule 2 - $\tau_1 = \mu_{12} \cdot \mu_{22} = 0$. The class label is thus 0, which means that the vector does not belong to any class (see below Figure 12). The parameters { a_{ir} , b_{ir} , c_{ir} } are defined as follow $a_{ir} = \min\limits_{k \in s}(g_i(k))$ and $C_{ir} = \max\limits_{k \in s}(g_i(k))$ and b_{ir} is Average of g_i and s is the corresponding subset.

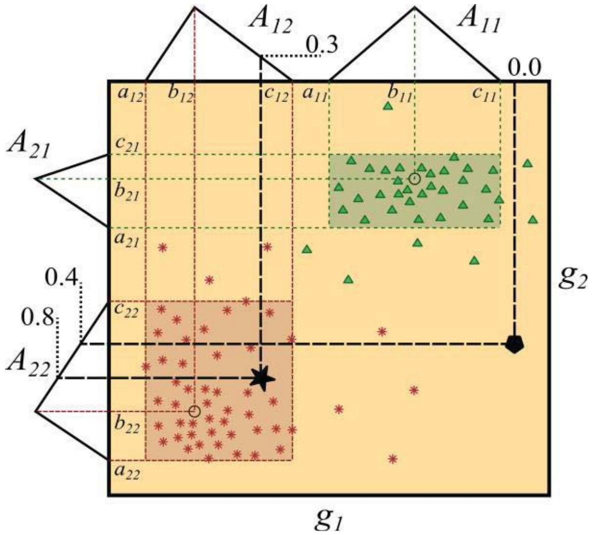


Figure 12. Two clusters in a 2D space modeled by triangular MFs

The classifier based on triangle MFs cannot operate on samples that fall beyond the rule borders of specified by the MF base parameters. This can be fixed if desired by replacing the triangular MFs with their nearly equivalent Gaussian curves defined as in equation (57).

$$\mu_{ir}(g_i) = \begin{cases} \exp\left\{ -\dfrac{(g_{ir} - b_{ir})^2}{2.(0.4247.(b_{ir} - a_{ir}))^2} \right\}, & g_i < b_{ir} \\ \exp\left\{ -\dfrac{(g_{ir} - b_{ir})^2}{2.(0.4247.(c_{ir} - b_{ir}))^2} \right\}, & g_i \ge b_{ir} \end{cases} \tag{57}$$

Table IX shows that Figure 10 and Figure 11 computation time and their impact on the overall throughput.

TABLE IX
FIGURE 11 COMPUTATION LOAD FLOW CHART OF THE MFCC AT 200 MHZ

OPERATION	Computation	Throughput
Spectrum	(Ns*5)/2	0.5ms/per frame
Mel-Scaling	NML*[(3Ns/2) -1]	
Logarithm	2*NML	
DCT	4*NML*NDCT+NDCT+NML	
Differentiator	J*[2*NDCT+3]	
Similarity	8*NML*NCL	

where NML and NDCT are the number of Mel and DCT coefficients, NCL is the number of people in the database and J is half of the number of frame use for feature extraction. The decision branch does not require a series of operations but rather some comparisons.

VIII. RESULTS AND DISCUSSION

All Tables from I to IX presented above are computed in the most pessimistic scenario under the assumption that the Hardware that will be used to implement this work has only one adder, multiplier, divisor and one square root primitive. Therefore the computations are made sequentially. However hardware such as FPGAs (Field Programmable Gate Array) have a huge computation power and all the results presented could be reduced by 20 to 30 %. The results presented in this section are modeled in Matlab using the above mathematical expressions. Table X is the overall throughput from the speaker localization to its recognition.

TABLE X
SPEAKER RECOGNITION THROUGHPUT USING DSB FOR LOCALIZATION, MVDR FOR BEAMFORMING AND MFCC FOR VOICE EXTRACTION FEATURE FOR N = 8 NS = 512 FOR A REAL-TIME SPEED CONSTRAINS OF 23.11 MS USING DIFFERENT CLOCK SPEED

Throughput	MVDR Beamforming	DSB Beamforming
200 MHz	14.15 ms	13.26 ms
400 MHz	7.07 ms	6.63 ms
600 MHz	4.75 ms	4.55 ms

This work has shown that it is possible to combine a hybrid algorithm to a flexible hardware architecture to successfully locate a particular speaker and track him among others using voice recognition technique in real-time.

Figure 13 shows the localization of two sources at different distance of the microphone. It shows that the closer the sources are from the microphones the more difficult it is to find their DOA see data1 to data 5. The localization accuracy becomes reliable beyond 0.5m see data 6 to data 7. Another limitation of this work is the angular distance necessary between both sources to avoid any masking of one source by the other. An angular distance of (20-30) degree is necessary meaning that the number of sources in the FOV should be limited to 4.

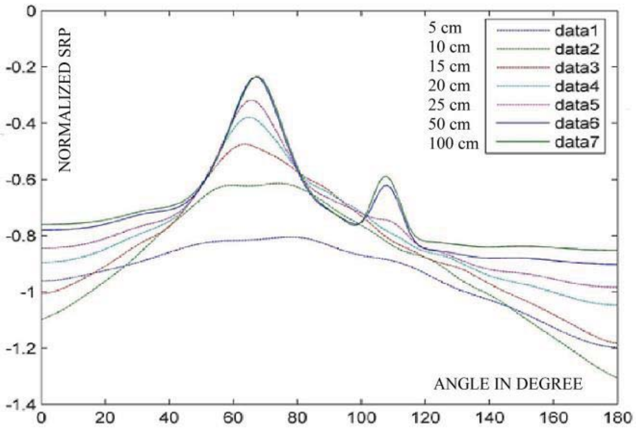


Figure 13. Detection of two I sources in the FOV with N = 8 and β = 0.8.

Figure 13 was drawn using equation (21) and Figure 5 block diagram and β = 0.8 which in literature represents the best value to overcome reverberation [34]. The power 0.8 in equation (4) is very challenging to compute in hardware. The best approximate value in terms of computation is β = 0.75 = 3/4. But a logarithmic computation approach can also be considered. For instance if $Y = \sqrt[p]{X^q}$, Y can be computed as modeled in equation (58).

$$Y = b^{\frac{q}{p}\log_b X} \tag{58}$$

The localization of two speakers simultaneously creates a dominant speaker, called primary source, which masks totally or partially the secondary speaker. Figure 14 shows that the DOA localization errors of both speakers depend on the angular distance between them and their position in the FOV.

Figure 14 shows the algorithm estimation error of the secondary and primary source location respectively in green and in black. The test is run over 306 frames with the primary source exact position varying from 175° to 100° degree with a 5° degree steps and the secondary source varying from 5° to 80° with the same step. Results show that the highest error for the secondary source is 15 degrees while it is less than 5 degree for the primary source.

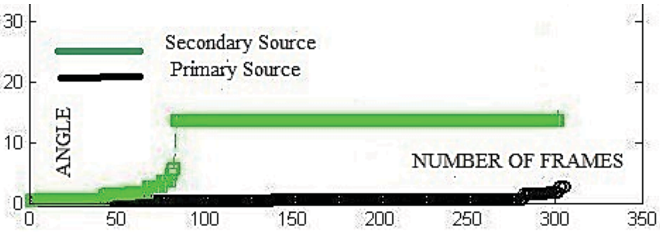


Figure 14. Estimation Error of the DOA with two I Source.

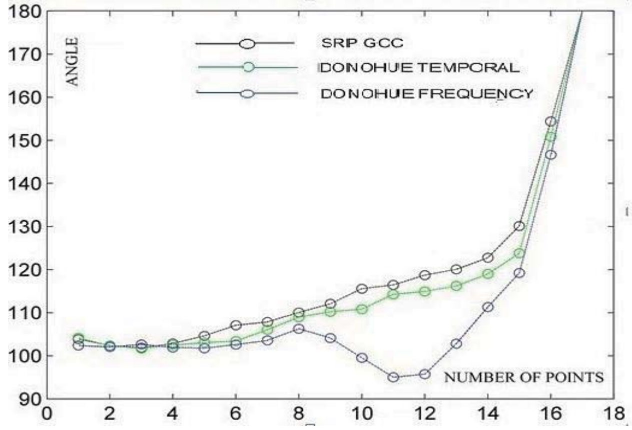


Figure 15. Secondary Error comparisons between SRP GCC (Figure 4) and DONOHUE (Figure 5).

Figure 15 results represent the localization estimation error of the secondary speaker taking into account the interferences between speakers using three different algorithms. If due to the interferences the secondary speaker is totally masked by the primary speaker the error assigned to the frame is 180° as it cannot be detected. The algorithms used are: the GCC SRP, Donohue in the temporal and frequency domain. Donohue algorithm in the Frequency domain outperforms the two other algorithms, but the errors due to the interferences are still very high. Figure 16 presents a sub array microphone structure to resolve the interference issue.

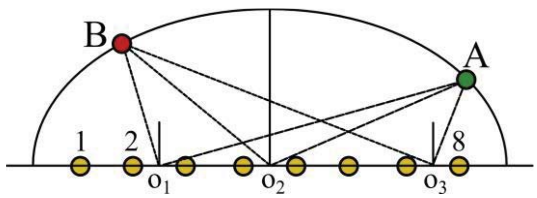


Figure 16. Interference Reduction approach between speakers.

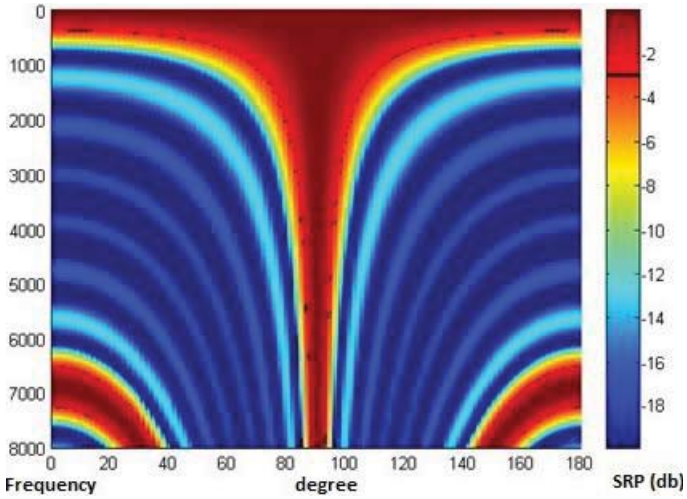


Figure 17. DSB Directivity.

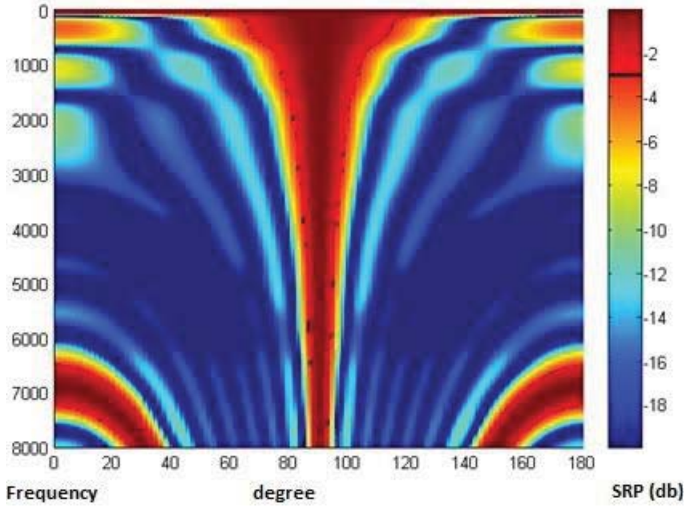


Figure 18. MVDR Directivity.

MVDR has a better directivity than DSB under 1 KHz as shown in Figure 17 compared to Figure 18. Moreover MVDR can be coupled with Wiener filter as explained above for better results. However, above 1 KHz the DSB and MVDR algorithms have similar results; it is then preferable to use DSB for its smaller computation load compared to MVDR (see Table VII and VIII).

An accurate localization combine to very directivity beamforming algorithm drastically improve speaker recognition results. Table XI is computed on the secondary speaker (see Figure (14)). The percentage of speakers recognized using the DOA angle over 306 frames is 100% for 90 frames and little least than 90% on 216 frames as shown in litterature [9]. Using the DSB after the GCC for localization improves drastically the percentage of speaker recognition due to the reduced localization error.

TABLE XI
PERCENTAGE OF SPEAKER RECOGNIZE USING BEAMFORMING WITH DOA ANGLE PER NUMBER OF FRAMES

NB-FRAMES	90	216
Percentage	100%	90%

In this work, the angular distance between both speakers need to be superior to 20% to reduce the effect of interferences between them.

A. *Localization Limit and Interference Reduction*

To reduce the 20° degree minimum angular distance between speakers a more directive algorithm such as Multiple Signal Classification (MUSIC) could be used. The DOA of the speakers is modeled as in equation (59).

$$P_{MUSIC}(\phi) = \frac{d^{H}.d}{d^{H}(\phi).Q.Q^{H}.d(\phi)} \qquad (59)$$

Where d is presented in equation (16) and Q represents the matrix of the Eigen vectors. Equation (59) is computed for every bin in the frequency domain therefore the DOA on the wide frequency band is the average between all the bins and modeled as in equation (60).

$$P_{WBMUSIC}(\phi) = \frac{1}{Ns}\sum_{k=1}^{Ns} P_{MUSIC}(\phi) \qquad (60)$$

Q is computed as in equation (61). R is the covariance matrix which is closely similar to equation (15) and D is a diagonal matrix composed of Eigen values.

$$R = QDQ^{-1} \qquad (61)$$

Result of Figure 19 shows sharper peaks compare to Figure 13 which mean less interference between speakers.

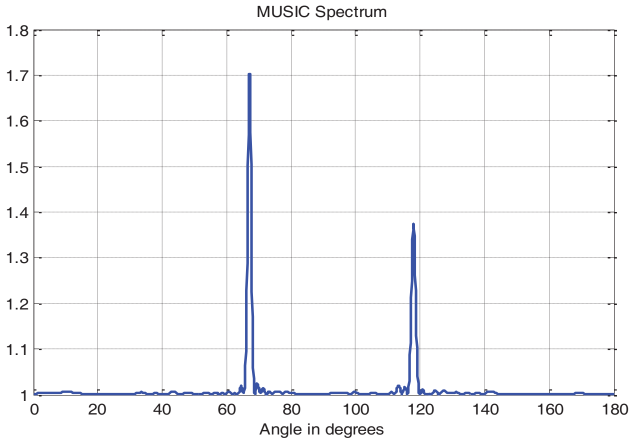


Figure 19. Speakers DOA detection using MUSIC algorithm

The sequential computation load of the MUSIC algorithm is presented in Table XII

TABLE XII
MUSIC SERIAL THROUGHPUT FOR N = 8 AND NS = 512

Operation	MUSIC Localization	Burden
BLKREAD	NN[Ns.cpl+2] + N[N+1]	1 043 667
MULT	NN[(Ns-1)(CPL-1)]+(N-1)(NN-1)	
ADD	N*N	
DIV	N*N*(2Ns+3N+4)	

Beside the higher throughput of the MUSIC algorithm compare to GCC (see Table XIII and Table IV), the need to know the number of speakers in the FOV before the computation of MUSIC algorithm presents its main drawback. This issue could be resolved by using equation (62) which represents the minimum description length (MDL) [36]. The number of speakers is the point at which equation (62) reach its minimum.

TABLE XIII
MUSIC SERIAL LOCALIZATION THROUGHPUT FOR 180 LOCATIONS

DSB (Clock Speed)	200 MHz	400 MHz	600 MHz
Throughput Parallel	5.3 ms	2.65 ms	1.8 ms

$$MDL(d) = L(d) + \frac{1}{2}.d.(2N-d).\log N_s \qquad (62)$$

L(d) is defined as in equation (62) with d being the number of speakers.

$$L(d) = -N_s(N-d).\log\left\{ \frac{\left| \prod_{n=d+1}^{N} \lambda_n \right|^{1/N-d}}{\frac{1}{N-d} \sum_{n=d+1}^{N} \lambda_n} \right\} \qquad (63)$$

Another approach to determine the number of speakers from a multi speaker speech signal can be based on the computation of linear prediction residual (LPE) and Hilbert envelope (HE) as defined in [37].

IX. Conclusion and Further Work

Multiple steps from the source acquisition to the tracking of the speaker using voice recognition were necessary and divided as follow: sources localization, beamforming, features extraction and classification. As each block mentioned above needs the output of the previous, parallelizing their computation is impossible. This work then proposed to reduce each block throughput individually using the approach stated above to achieve speaker tracking using voice recognition in real-time.

In further work, interferences between speakers can be addressed to allow more than 4 speakers to be located and tracked. In the most optimistic scenario, this work can be coupled with video localization to allow the tracking to be done by voice and face recognition.

Acknowledgement

This work was funded by the Wallonia Region DG06 Belgium under grant 917005-CTEUC 2009 Eureka ITEA DIYSE and the Innovative Manufacturing Engineering Systems Competence Centre IMECC, co-financed by European Union Regional Development Fund (project EU30006).

References

[1] Priyabrata Sinha, Alan D. George and Keonwook Kim, “Parallel Algorithms for Robust Broadband MVDR Beamforming,” http://www.hcs.ufl.edu/pubs/JCA2001.pdf

[2] C. Cave, R. Wasser, “Estimating Parallel Processing Speed Multiplier”, http://www.visisoft.us/PDF_Files/EstimatingSpeedMultipliers.pdf, pp. 216-228, 31 March 2007.

[3] Fahad Qureshi, Syed Asad Alam and Oscar Gustafsson, “4K-Point FFT Algorithms based on optimized twiddle factor multiplication for FPGAs” The Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia), Sanghai, Sept 22-24 2010. http://www.ep.liu.se/PubList/Default.aspx?userid=fahqu64.

[4] Sanjay Thatte, John Blaine, “How to Manage Power Consumption in Advanced FPGAs,” Xcell Journal Xilinx Fall 2002.

[5] Hichem Belhadj, Vishal Aggrawal, Ajay Pradhan and AmalZerrouki, “Power-Aware FPGA Design” 2009.

[6] Kirill Sakhnov, Ekaterina Verteletskaya, and Boris Simak, “Approach for Energy-Based Voice Detector with Adaptive Scaling Factor”. IAENG International Journal of Computer Science, 36:4, IJCS_36_4_16

[7] H. Othman and T. Aboulnasr, “A Semi-Continuous State-Transition Probability Based Voice Activity Detector”, Hindawi Publishing Corporation EURASIP Journal on Audio, Speech, and Music Processing Volume 2007, Article ID 43218, 7 pages doi:10.1155/2007/43218.

[8] Christian Ibala, F. Escobar, X. Chang, C. Valderrama, “Hybrid Algorithm Computation Methodology to accelerate Sound source localization” International Journal of Microelectronic and Computer Science VOL 3, NO 3, 2012.

[9] E. Lleida, J. Fernandez, E. Masgrau, “Robust Continous Speech Recognition System Based on a Microphone Array” Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference. 12-15 May 1998, Page 241-244 Vol 1.

[10] Lupu, E., Feher, Z., Pop, P.G. “On the speaker verification using the TESPAR coding method”. International Symposium on Signals, Circuits and Systems, 2003, pp. 173 – 176.

[11] S. Astapov, and A. Riid, “A Hierarchical Algorithm for Moving Vehicle Identification Based on Acoustic Noise Analysis,” 19th International Conference “Mixed Design of Integrated Circuits and Systems” MIXDES 2012, Warsaw, Poland, pp. 467-472, 24-26 May 2012.

[12] Astapov, S. Preden, J.S. Suurjaak, E. “A method of real-time mobile vehicle identification by means of acoustic noise analysis implemented on an embedded device,” 13th Biennial Baltic Electronics Conference (BEC), pp.283-286, 3-5 Oct 2012.

[13] I.A. McCowan, ”Robust Speech Recognition using Microphone Arrays,” PhD Thesis, Queensland University of Technology, Australia, 2001.

[14] Satish Mohan, Michael E. Lockwood, Michael L. Kramer, Douglas L. Jones, “Localization of multiple acoustic sources with small arrays using a coherence test” J. Acoust. Soc. Am. Volume 123, Issue 4, pp. 2136-2147 (2008); (12 pages)

[15] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Jacek Dmochowski, “On Microphone Array Beamforming From a MIMO Acoustic Signal Processing Perspective,” Audio, Speech, and Language Processing, vol. 15, no. 3, pp. 1053 – 1065, March 2007.

[16] Master Thesis **“**David K. Campbell” **“**Adaptive Beamforming Using a Microphone Array for Hands-Free Telephony” http://my.fit.edu/~vkepuska/ece5525/MicrophoneArray/etd.pdf

[17] Ivan Tashev, I Capture and Processing, Wiley, Ed., 2009.

[18] Takanobu Nishiura, Takeshi Yamada, Satoshi Nakamura, Kiyohiro Shikano, “Localization of Multiple I Sources Based on a CSP Analysis with a Microphone Array” http://library.naist.jp/dspace/bitstream/10061/8030/1/ICASSP_2000_1053.pdf

[19] Lefkimmiatis, S., et P. Maragos. «A Generalized Estimation Approach for Linear and Nonlinear Microphone Array PostFilters. » *Speech Communication*, n° 49(2007): 657-666.

[20] S. N. Bhuiya, F. Islam, and M. A. Matin, “Analysis of Direction of Arrival Techniques Using Uniform Linear Array” International Journal of Computer Theory and Engineering, Vol. 4, No. 6, December 2012.

[21] Christophe Ris Polytech Mons Internal Document “Développement d’un logiciel de beamforming pour réseau de microphone linéaire.”

[22] Weidong Li and Lars Wanhammar, “Efficient Radix-4 and Radix-8 Butterfly Elements” www.es.isy.liu.se/publications/papers_and_reports

[23] The Scientist and Engineer’s Guide to Digital Signal Processing By Steven W. Smith, PhD. http://www.dspguide.com/

[24] Btzier, J., K,U. Simmer, and K.D. Kammeyer, “Mulit-Microphone Noise Reduction Techniques as Front-End Devices for Speech Recognition.” Speech Communication, n°34(2001) : P 3-12.

[25] McCowan, I.A and H. Bourlard, “Microphone Array Post-Filter Based on Noise Field Coherence.” IEEE transaction on Speech and Audio Processing 11, n°6 (November 2003): P 709-716BEAMFORMING,” http://www.hcs.ufl.edu/pubs/JCA2001.pdf

[26] Ning Cheng, Wen-Ju Liu, Peng Li, Bo Xu, “Microphone array speech enhancement based on a generalized post-filter and a novel perceptual filter.” Signal Processing, ICSP 26-29-09-2008 Proceedings; P370-3

[27] Peeters, G. “A large set of audio features for sound description (similarity and classification) in the CUIDADO project”. CUIDADO I.S.T. Project Report

[28] Vaseghi, S.V. Multimedia signal processing: Theory and applications in speech, music and communications. John Wiley & Sons Ltd., UK, 2007.

[29] P. Mahesha and D.S Vinod, “Vector Quantization and MFCC based classification of Dysfluencies in Stuttered Speech” Bonfring International Journal of Man Machine Interface, Vol. 2, No. 3, September 2012.

[30] Sigurdur Sigurdsson, Kaare Brandt Petersen and Tue Lehn-Schiøler, «Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music

[31] Jinjin Ye, B.S ”Speech Recognition Using Time Domain Features from Phase Space Reconstruction”. MASTER OF SCIENCE.” Marquette University Milwaukee, Wisconsin May 2004

[32] Jang, J.-S., Sun, C.-T., Mizutani, E. Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Prentice-Hall, Inc., 1997

[33] A Riid, and N. Saadallah, “Unsupervised learning of well drilling operations: Fuzzy rule-based approach,” IEEE 16th International Conference on Intelligent Engineering Systems, Lisbon, Portugal, pp. 375-380, 13-15 June 2012

[34] Hoang Do, Harvey Silverman, and Ying Yu, “A Real-Time SRP-PHAT Source Location Implementation Using Stochastic Region Contraction (SRC) on a Large-aperture Microphone Array,” in IEEE International Conference on Acoustics, Speech and Signal Processing, 2007, pp. I-121 – I-124

[35] Maurice F. Fallon and Simon . Godsill, “Acoustic Source Localization and Tracking of a Time-Varying Number of Speakers” IEEE Transaction on Audio, Speech, and Language Processing, Vol. 20. No. 4, May 2012

[36] Zhizhang Chen, Gopal Gokeda, Yiqiang Yu, “Introduction to Direction-of-Arrival Estimation” ARTECH HOUSE ISBN 13: 978-1-59693-089-6

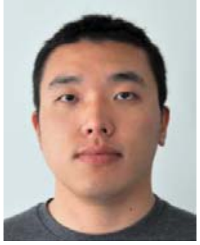
[37] R. Kumara Swamy, K. Sri Rama Murty, and B. Yegnanarayana, Senior Member, IEEE “Determining Number of Speakers From Multispeaker Speech Signals Using Excitation Source Information”, IEEE Signals Processing Letters, Vol. 14, No. 7, July 2007

Christian Serge Ibala received his MSc in 2000, he went to work for Cadence Scotland from 2000 to 2002 then from 2003 to 2009 worked for Xilinx Ireland. In 2008 start his PhD at the University of Limerick (Ireland) he is working toward finishing it in collaboration with the University of Mons (Belgium). His research interest includes reconfigurable architecture, Digital signal processing and systems digital design and validation.

Sergei Astapov received his M.Sc. degree in the field of Computer System Engineering at the Tallinn University of Technology in 2011. He continues his education as a PhD student at the Department of Computer Control at the Tallinn University of Technology and is a member of the Department’s Research Laboratory for Proactive Technologies. His research interests include object tracking using wideband signal analysis, classification tasks and distributed computing in embedded multi-agent systems. His recent research concerns object localization and identification in open environments and acoustic signal based diagnostics of industrial machinery.

Frédéric Bettens received his MSc degree in 1996 and his PhD degree in 2003, both at the Free University of Brussels (ULB, Belgium). He is now working as senior researcher at the University of Mons (UMONS, Belgium). His research interests include audio and speech signal processing.

Fernando Escobar was born in 1985. He received bachelor's and MSc and degrees in Electronic Engineering from Universidad de Los Andes, Bogotá, Colombia, in 2008 and 2011 respectively. He is currently a PhD student in the Department of Electronics and Microelectronics, University of Mons, Mons, Belgium. His research interests include high level modelling using Hardware Description Languages, SystemC and MATLAB, among others; his areas of expertise are computer architecture, embedded systems, networks on chip and digital design.



Xin Chang was born in 1988. He received MSc degree in Embedded Computing from University of Turku (Finland) in 2012. He is currently working as a research assistant in the Department of Electronic and Microelectronics, University of Mons. His research interests include on-chip interconnection, high-level synthesis, multiprocessor system-on-chip and signal processing.

Carlos Valderrama’s research interests are power processing, consumption and management. He is active in the area of embedded applications, wireless smart sensors for logistics, and signal processing for biomedical and telecommunication applications, among others. His main research activities are methodologies and tools for the design of multi-core architectures and SoC platforms for embedded applications. He is currently member of several scientific committees of international conferences (DAC, FPL, RAW, IDT, ReConfig and Iberchip among others). His research activity is supported by several publications and books chapters, and tutorials.

Andri Riid received his M.Sc. and Ph.D. degrees in System Engineering from Tallinn University of Technology in 1997 and 2002, respectively. He currently works as a Senior Research Scientist in the Laboratory for Proactive Technologies of the same university. His research interests include properties of fuzzy systems and development of algorithms for fuzzy control, modeling and classification. He has published over 40 papers in international peer-reviewed journals and conference proceedings.