Time Difference Estimation with Sub-sample Interpolation

Simayijiang Zhayida and Kalle Åström
Centre for Mathematical Sciences, Lund University, Sweden
E-mail: {zhayida, kalle}@maths.lth.se

Abstract Estimates of time difference of arrival (TDoA) can be used in audio (and radio) for localization, positioning and navigation. Such estimates are typically obtained from one or several microphone (or antenna) pairs. In this paper we demonstrate that better TDoA estimates can be obtained by using sub-sample correlation. The main idea is to first obtain an initial estimate of the TDoA estimate and then refining these using a local optimization over sub-sample shifts. For a sub-sample shift one calculates the interpolated signal and evaluates the correlation criterion and the derivatives of the correlation with respect to sub-sample shifts. These are then used in an efficient implementation of determining optimal TDoA estimates using a local optimization of the correlation criterion. The method has been implemented and tested on real data, where it is demonstrated that using the improved TDoA estimates in a system for simultaneously estimating microphone and sound source positions gives better precision and lower residuals in the results.

Keywords: time difference estimation, sub-sample interpolation

1. Introduction

Sound localization has been a topic of interest in a wide range of applications for centuries, it has been widely used in positioning and navigation system. Although such problems have been studied extensively in the literature in the form of localization of e.g. a sound source using a calibrated detector array, see e.g. [1]-[4], the problem of self-calibration of a sensor array is still an open problem. Typically, localization estimates are formed by exploiting time of arrival (ToA), time difference of arrival (TDoA) estimates, and by matching signal attenuation over an array of sensors, often using cross-correlation or canonical correlation analysis (CCA) techniques, allowing the source positions to be determined using tri- and multilateration (see, e.g., [5], [6]).

TDoA based localization has been adopted for a variety of applications, range from navigation system to sonar devices. The TDoA measurement and sensor position data are used to generate hyperbolic curves which are then intersected in some optimal sense to estimate a source location. A number of variations on this principle have been developed, see [7]-[9].

In [10], an automatic system for microphone self-localization using ambient sound is presented. The system is based on first finding several time-difference matching vectors for the recordings. These are then used as input to robust geometric algorithms based on minimal solvers and RANSAC to provide initial estimates of unknown parameters, i.e. the offsets and the 3D positions of the sound sources and the receivers. These estimates are then improved by non-linear optimization to obtain the maximum likelihood estimate of the parameters. The development of automatic system is dependent on several components. In this paper, we focus on time-difference estimation as a refinement step for the system in [10].

The problem considered here can be stated as follows: Given a time index (by sample) in a reference channel, the problem is to find the position of the corresponding time index in the second channel with sub-sample accuracy.

Fig. 1 In the experiments, microphones connected to an audio interface, then it connected to a laptop as depicted in the figure: There is one moving sound source in the figure, whose trajectory is depicted as a red line. For the system, we assume that the microphones can be in unknown general 3D positions and that the microphones are synchronized. We do not put any constraints on the motion of sound source.
Correlation is usually done on sample level, sub-sample estimation is the process of estimating the value of a geometric quantity to better than sample accuracy, even though the data was originally sampled on an integer sample quantized space, see [11].

2. Overview of the System

This paper is a continuation of [10]. Here we will give an only brief introduction to the system. The input to the system consists of sound recordings with M channels \((x_1, \ldots, x_M)\). Typically sampling rates are 96000 Hz, which means that sample points are \(h = 1/96000\) seconds apart. The microphones are at unknown positions \((m_1, \ldots, m_M)\), where \(3 \leq R_i \leq m_i\) for \(i = 1, \ldots, M\). See Fig. 1. We assume that among the sounds there are one or several, possibly moving sound sources. Due to distance differences, there is a time difference of arrival for the same pulse in different receivers. The receivers do not need to know the absolute time at which the pulse was transmitted, only the time difference is needed. At several time instances along the sound channels there are one or several matchings between channels. Each such match corresponds to a set of time instants \(t_i\) of arrival times to the microphones. Each such time vector \((t_1, \ldots, t_M)\) correspond to a sound made at instant \(t_0\) at 3D position \(s \in \mathbb{R}^3\) fulfilling

\[
c(t_i - t_0) = \|m_i - s\|
\]

where \(c\) is the speed of sound, assumed to be known and constant, and \(\|\|\) is \(l_2\)-norm. Without loss of generality, we will in the sequel assume that all time differences are measured against channel 1.

We also introduce \(o = (t_1 - t_0) = \|m_1 - s\|\) as the offset. This can be interpreted as the distance from the sound to microphone 1. Let \(j\) be used as an index for different sounds. It is possible to estimate the unknown parameters \((m_i, s_j, o_j)\) using a number of measurements \(u_{ij}\). \(u_{ij} = \|m_i - s_j\| - o_j\) (2)

Here \(u_{ij}\) are the time-difference of arrival measurement between microphone \(i\) and microphone 1 for a sound at position \(s_j\) multiplied by the velocity of sound. These measurements are placed in so called time difference vectors \(u_j = (u_{1j}, \ldots, u_{Mj})\).

The recordings from M channels, are first processed to find the time difference vectors. The estimated time difference vectors typically contain noise, missing data, and possible outliers. We then follow a stratified approach, where we first estimate offsets using a robust method, [12]. Another possibility would be to use [13]. This is followed by a robust method for finding the 3D positions of all the microphones and the 3D positions of all sound sources, [10]. Next is a refinement of matching matrix by finding more inliers from raw matches and also a refinement of correlation pattern using sub-sample methods. A block diagram of the procedures can be seen on Fig. 2.

The system divided into six steps as below. Steps (A) - (C) are described in [10], Steps (D) - (F) are appear as refinement to system in [10]. In this paper, we will mainly focus on Step (E). However, for the completeness, the main functions of each step are summarized here:

(A) Time difference estimation using feature detection and matching strategy for calculating correlation pattern. This produce TDOA data, possibly with missing data and with outliers.

(B) Offset estimation from raw matches by using robust techniques to calculate initial estimates on the offsets parameters, followed by non-linear optimization based on a rank criterion.

(C) Estimate microphone, sound positions and refine offset. The initial estimates of the sound source and microphone positions are calculated with the robust method followed by non-linear Maximum Likelihood estimation of all parameters.

(D) Find more inliers from matching matrix which is calculated at Step (A).
Refinement of time differences estimation using the sub-sample algorithm and find more inliers from matching scores.

Using refined offset and matching vectors to estimate microphone and sound positions similar as Step (C).

3. Find More Inliers from Time-Difference Estimation

In order to find more inliers among the remaining columns in \( \mathbf{u} \), first we calculate sound positions \( \mathbf{s}_j \) and offsets \( o_j \) for each \( j \) individually, by trilateration using time differences vector \( \mathbf{u}_j \) and microphone positions \( \mathbf{m}_1, \ldots, \mathbf{m}_M \). For each column in the inlier set of \( \mathbf{u} \), trilaterate one point using RANSAC followed by non-linear least squares optimization of \( \mathbf{s}_j \) and \( o_j \), according to

\[
\min_{s_j, o_j} \sum_i (u_{ij} - \|m_i - s_j\| + o_j)^2
\]

This corresponds to Step 1 of Algorithm 1.

Next we calculate residuals

\[
r_{ij} = u_{ij} - \|m_i - s_j\| + o_j \quad \text{(Step 2)}
\]

and then update the inlier set \( I \). The new inlier set \( I \) consists of index pairs \((i, j)\) for which the absolute value of the residual is less than a threshold \( T \), (Step 3). A typical value for the threshold is \( T = 0.05 \) meters. This gives, i.e.

\[
I = \{(i, j) | r_{ij} < T \}
\]

Finally we optimize simultaneously over all microphone positions, all sound positions and all offsets according to

\[
\min_{m_i, s_j, o_j} \sum_{(i, j) \in I} (u_{ij} - \|m_i - s_j\| + o_j)^2
\]

This corresponds to Step 4 of Algorithm 1.

Algorithm 1

Step D — Find more inliers among the remaining columns of matching matrix \( \mathbf{u}^A \)

Set parameters: \( \mathbf{m} \) for microphone positions, \( \mathbf{s} \) for sound positions, \( \mathbf{o} \) for offsets, and \( \mathbf{u} \) for time difference matches. The superscript indicates that the variable is output of which step, for e.g. \( \mathbf{u}^A \) is time difference match we get from Step A as an output.

Input: \( \mathbf{u}^A, \mathbf{m}^C \)

Output: \( \mathbf{u}^D, \mathbf{m}^D, \mathbf{s}^D, \mathbf{o}^D \).

1. Trilaterate sound positions using \( \mathbf{u}^A \) and \( \mathbf{m}^C \), then update offsets.
2. Calculate residuals.
3. Find all matching points which has residual < \( T \) meters, and update matches as \( \mathbf{u}^D \).
4. Use \( \mathbf{u}^D \) to do bundle adjustment for update \( \mathbf{m}, \mathbf{s} \) and \( \mathbf{o} \).

4. Sub-sample Interpolation

The sample displacement is an integer value due to the discretization of signal. Discrete signals are obtained from the continuous signal during sound acquisition. This process can be modeled by low pass convolution followed by ideal sampling plus added noise. If the continuous signal existed, one could compare two signals at sub-sample shift. One alternative is to reconstruct the continuous signal from the discrete signal. When one reduces a continuous function to a discrete sequence and interpolates back to a continuous function, the fidelity of the result depends on the sample rate of the original samples. Nevertheless, these assumptions will help us to model and analyses estimating the continuous signal from the discrete signal, see Fig. 3. Smoothing is used to reduce the effects of noise. To avoid aliasing effects, we assume that the function \( F_2 = F \ast G \) is band-limited, i.e. the Fourier transform is zero outside a bounded interval, and thus all energy in the high frequencies is canceled before discretization. By the Nyquist-Shannon sampling theorem, [14], we have no aliasing effects and it is possible to reconstruct the signal by ideal interpolation.

Therefore, the shift exhibit a range of integer values, which limits the accuracy of time difference estimation. To overcome this problem, we utilize subpixel interpolation for recalculate time difference estimation.

The basic idea is to align the position of matching windows with sub-sample accuracy in two blocks. In practice, there are many applications that require high-accuracy matching of small window blocks. Such applications include 3D measurement by stereo vision and super-resolution image reconstruction, where corresponding points in two images must be determined with high accuracy using block matching.
In these applications, the accuracy of block matching (i.e., the accuracy of local displacement estimation) determines the total system performance directly [15].

The sub-sample interpolation algorithm consists of two stages: a stage for sample-level correspondence estimation and a stage for sub-sample-level correspondence estimation. In sample-level estimation, we detect the corresponding point with sample level accuracy, so that the estimation error becomes less than 1 sample, in the sub-sample estimation stage, we recursively improve the sub-sample accuracy of matching by adjusting the location of the window function with sub-sample accuracy. We call this technique sub-sample window alignment.

An example of sub-sample shift estimation from eight channels is shown in Fig. 4 in which the sub-sample shift of each frame with respect to a reference frame is illustrated.

For our time varying signals, in the first step one channel translated in whole sample units and matched to parts of a second channel so that the sum of squared differences is minimized. Next, our aim is to optimize the shifts so that all curves of microphones align, this implies that we have better time difference estimation. Sub-sample shift requires interpolation techniques, and its accuracy is limited by the accuracy of interpolation.

First determine the location of the peak to the nearest sample, search along the direction of maximum gradient for the peak and the location of these peaks will be refined through sub-sample interpolation. Sub-sample interpolation is based on “fitting” a continuous interpolation function to the discrete gradient values. We consider 1D Gaussian filter (i.e., convolution with a Gaussian kernel) for sub-sample displacement.

Let us denote \( \mathbf{u}_j \in \mathbb{R}^M \) as a matching vector at time \( t_j \).

Note that the matching vector \( \mathbf{u}_j \) denotes distance difference and that it is closely related to time-differences by dividing with the speed of sound \( c \). The absolute arrival times for channel \( i \) out of the \( M \) channels can be obtained as \( t_{ij} = t_j + \frac{1}{c} \mathbf{u}_j(i) \), where \( \mathbf{u}_j(i) \) denotes element \( i \) in the matching vector \( \mathbf{u}_j \).

Let us concentrate on one matching vector \( j \) and collect the \( M \) arrival times in a vector

\[
\left( \begin{array}{c}
    t_{1j} \\
    t_{2j} \\
    \vdots \\
    t_{Mj}
\end{array} \right)
\]

Such absolute arrival times can be evaluated by studying the signal around the times.

The matching vector is an element of \( \mathbb{R}^M \), whose first element can be assumed to be zero. We parametrize this vector as
\[ u_j = \begin{pmatrix} 0 \\ \theta \end{pmatrix} \]

with \( \theta \in R^{M^{-1}} \). To simplify the notation in subsequent steps we start the index numbering of the elements of the vector \( \theta \) with 2, i.e.,

\[ \theta = \begin{pmatrix} \theta_2 \\ \vdots \\ \theta_M \end{pmatrix} \]

Thus the absolute arrival times are a linear function of the parameter vector \( \theta \) according to

\[ \begin{pmatrix} t_{ij} \\ t_{2j} \\ \vdots \\ t_{Mj} \end{pmatrix} = \begin{pmatrix} t_j \\ t_j + \frac{1}{c} \theta_2 \\ \vdots \\ t_j + \frac{1}{c} \theta_M \end{pmatrix} \]  \( (6) \)

The raw sound data consists of M sampled sound recordings \( x_1, \ldots, x_M \). These are only defined at sampled positions, i.e. we have values of \( x_i \) at integer positions \( k \) corresponding to time \( hk \), where again \( h \) is the distance in time between two sample points, typically \( h = \sqrt[8]{96000} \).

Now we consider the ideal interpolation of these signals as functions defined at every time instant. Furthermore consider slight Gaussian smoothing of these interpolated signals. Introduce the normalized \( \sin c \) function

\[ \sin c(x) = \frac{\sin \pi x}{\pi x} \]

and the one-dimensional Gaussian function of width \( \sigma \),

\[ G_{1D,\sigma}(x) = \frac{1}{\sqrt{2\pi} \sigma} e^{-\frac{(x-\bar{\mu})^2}{2\sigma^2}} \]

The result are M functions, which according to [11] can be calculated using

\[ x_i(t) = \sum_k x_i(k) g(\frac{t-k}{h}) \]  \( (7) \)

and where the interpolation function used is \( g = \sin c \star G_{1D,\sigma} \).

This makes it possible to define the function at every non-integer point. Also, as shown in [11], it is possible to analyze the effect of noise in \( x \) on the interpolated function using random processes. Now form a data matrix

\[ A = \begin{pmatrix} x_1(t_{ij} - 1200h) & \ldots & x_1(t_{ij} + 1200h) \\ x_2(t_{2j} - 1200h) & \ldots & x_2(t_{2j} + 1200h) \\ \vdots & \cdots & \vdots \\ x_M(t_{Mj} - 1200h) & \ldots & x_M(t_{Mj} + 1200h) \end{pmatrix} \]

of size \( M \times 2401 \) by sampling each of the M signals at 1200 points before and after the mid points \( s_i \) for channel \( i \). These M sampled signals should ideally be similar in shape, although they could be slightly different in scale. We regard the matrix to be a function of \( \theta \) through Eq. (6). Part of such a matrix is illustrated in Fig. 4(a). Notice that it is straightforward to calculate the derivative of \( A \) with respect to \( \theta \) through Eq. (6).

Step 1: Since some of the values are missing from the matching vector \( u \), we need to calculate reprojected matching vector by

\[ u_j = \begin{pmatrix} 0 \\ \theta_2 \\ \vdots \\ \theta_M \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{m_i - s_j}{o_j} \\ \vdots \\ \frac{m_M - s_j}{o_j} \end{pmatrix} \]

This gives an initial estimate of \( \theta \), here \( o_j = \| m_i - s_j \| \).

Step 2: Calculate the matrix \( A(\theta) \) which containing the sampled cut-outs of the signals centered at \( s \) as a function of \( \theta \) and also the derivatives of \( A \) with respect to \( \theta \), i.e. \( \frac{\partial A}{\partial \theta} \).

Notice that the first row of \( A \) is fixed and that for row \( i \) with \( i \geq 2 \), the row vector \( a_i(\theta) \) only depends on one parameter, i.e. \( \theta_i \).

By doing this for all \( M \) channels, we obtain \( M \) sub-sample shifted channels \( a_i(\theta_i) \) as a function of \( \theta_i \) through the center points \( t_q = t_j + \frac{1}{c} \theta_i \) respectively. This is illustrated in Fig. 4 for 8 channels. Notice that in this example, the signals are indeed of similar shape, since they are aligned within a sample point or two correctly. We have also calculated \( \frac{\partial A}{\partial \theta} a_i(\theta) \).

Step 3: Calculate sub-sample step size. After Step 2 we get new sampling data \( a_i(\theta_i) \) and \( \frac{\partial A}{\partial \theta} a_i(\theta_i) \). Using a singular value decomposition of the \( M \times 2401 \) matrix \( A \) we find
the best rank 1 approximation \( \tilde{A} \) of \( A \). Row \( i \) of \( \tilde{A} \), i.e. \( \tilde{a}_i \), can be interpreted as the mean sound signal scaled to channel \( i \). The idea is to optimize the shifts \( \theta \) so as to minimize the difference between \( A \) and \( \tilde{A} \).

The sub-sample step size that makes \( a_i(\theta + \delta\theta) \) as similar to \( \tilde{a}_i \) in a non-linear least squares sense, can be obtained using a Gauss-Newton step according to

\[
\delta\theta_i = -\left( \frac{\partial}{\partial \theta_i} a_i(\theta) \right)^T \frac{\partial}{\partial \theta_i} (a_i(\theta) - \tilde{a}_i(s_i)) \frac{\partial}{\partial \theta_i} a_i(\theta) \right]^{-1} \frac{\partial}{\partial \theta_i} (a_i(\theta) - \tilde{a}_i(s_i)).
\]

Step 4: Update the parameters \( \theta_i \) according to \( \tilde{\theta}_i = \theta_i + \delta\theta_i \).

Step 5: Repeat from Step 2 to Step 4 for specific times. This is called the “sub-sample window alignment” technique, which gradually reduces the sub-sample error in the estimated displacement vector. In our experiments, the number of iterations is 10.

In the end, we obtained updated matching vector \( \tilde{\theta} \) for time \( t_j \), which can be used for calculate microphone and sound positions.

5. Experimental Validation

The data was obtained by 8 microphones (Shure SV100) which are connected to an audio interface (M-Audio Fast Track Ultra 8R), then it connected to a laptop. The 8 sound channels were sampled at 96000 Hz. We illustrate some of the steps of the automatic system with one of the experiment. In this experiment, the part of a song played by a mobile phone through a small speaker and sound source is moving slowly through the room. We assume \( c = 343m/s \) in room temperature.

In [10], the system is analyzing channel 1 vs channel \( i \) for \( \{i = 2, \ldots, M \} \) at 1000 positions along the track. At each such time instant, channel 1 is compared to channel \( i \) with shifts from -500 sample points to +500 sample points. If there are sufficient number of confident matching scores for most channels at a position, a matching vector \( \tilde{u} = (u_1, u_2, \ldots, u_M)^T \) is generated.

The matching algorithm in Step (A) produces 110 matching vectors. In Step (B), the RANSAC based algorithm finds an inlier set of 67 (out of the 110). These are then used to estimate the 3D positions of the sender and receiver. In proposed algorithm, after refinement, we get 110 inlier set (out of the 110), then the updated matching vectors used for recalculation of microphone and sound positions. The residuals,

![Graph](image_url)
Fig. 6 Histogram of residuals (in terms of sampling points) in different steps: The residuals between the matching vector $u_i$ and the fit $\|m_t - s\|^2 - o_j$: We get different number of residuals in these steps mentioned as Fig. 5, but for the fair comparison, we used 54 and 106 residuals for Figs. 6(a) and 6(b), respectively.

6. Conclusions

The system is based on first finding several time difference matching vectors for the recording. Then using these vectors as an input of robust geometric algorithms which based on minimal solvers and RANSAC to provide initial estimates of the unknown parameters, i.e. the offset and the 3D positions of the sound source and the receiver. Next, these estimates are improved by non-linear optimization to obtain the maximum likelihood estimate of the parameters. After that, as a refinement of the system, more inliers from time difference matchings are found, and then they are refined by sub-sample correlation. In the end, the microphone and sound positions recalculated based on updated matching vectors.

In this paper, in order to achieve more accurate microphone and sound positions, we employ sub-sample interpolation for refined time difference estimation, see [11]. The high-precision evaluation of the position requires interpolation calculations between the spatial samples to obtain a precision better than the geometrical pitch of the elementary samples. The cross-correlation between two sequences can be used to measure for how similar these two sequences are for different shifts. Correlation is usually done on sample level, sub-sample estimation is the process of estimating the value of a geometric quantity to better than sample accuracy, even though the data was originally sampled on an integer sample quantized space.
results of automatic microphone self-localization system presented in [10].

References


Simayijiang Zhayida received her M.Sc. degree in applied mathematics with branch of graph theory in 2009 from Xinjiang University, China. Since then she has been doing her Ph.D. studies in mathematics at the Centre for Mathematical Sciences, Lund University. Her main research interests are biomedical signal processing and sensor localization.

Kalle Åström received his B.Sc. in mathematics in 1990, M.Sc. degree in engineering physics in 1991 and Ph.D. degree in mathematics in 1996 from Lund University, Sweden. His thesis was awarded Best Nordic Ph.D. Thesis in pattern recognition and image analysis, 1995-1996, at the Scandinavian Conference in Image Analysis, in 1997. He was a post-doctoral research fellow, Associate Professor, and is now Professor at the Centre for Mathematical Sciences, Lund University. His teachings include undergraduate and graduate courses in mathematics, image processing and computer vision. His current research interests include stochastic analysis of low level vision, computer vision for autonomous systems, geometry and algebra of multiple views of points, curves and surfaces, structure from sound, radio and WiFi, semantic structure from motion, handwriting recognition and medical image analysis.

(Received May 21, 2016; revised August 28, 2016)