Source Localisation and Blind Source Separation

(UDRC Summer School)

Dr James R. Hopgood

James.Hopgood@ed.ac.uk

Room 2.05

Alexander Graham Bell Building

The King's Buildings

Institute for Digital Communications

School of Engineering

College of Science and Engineering

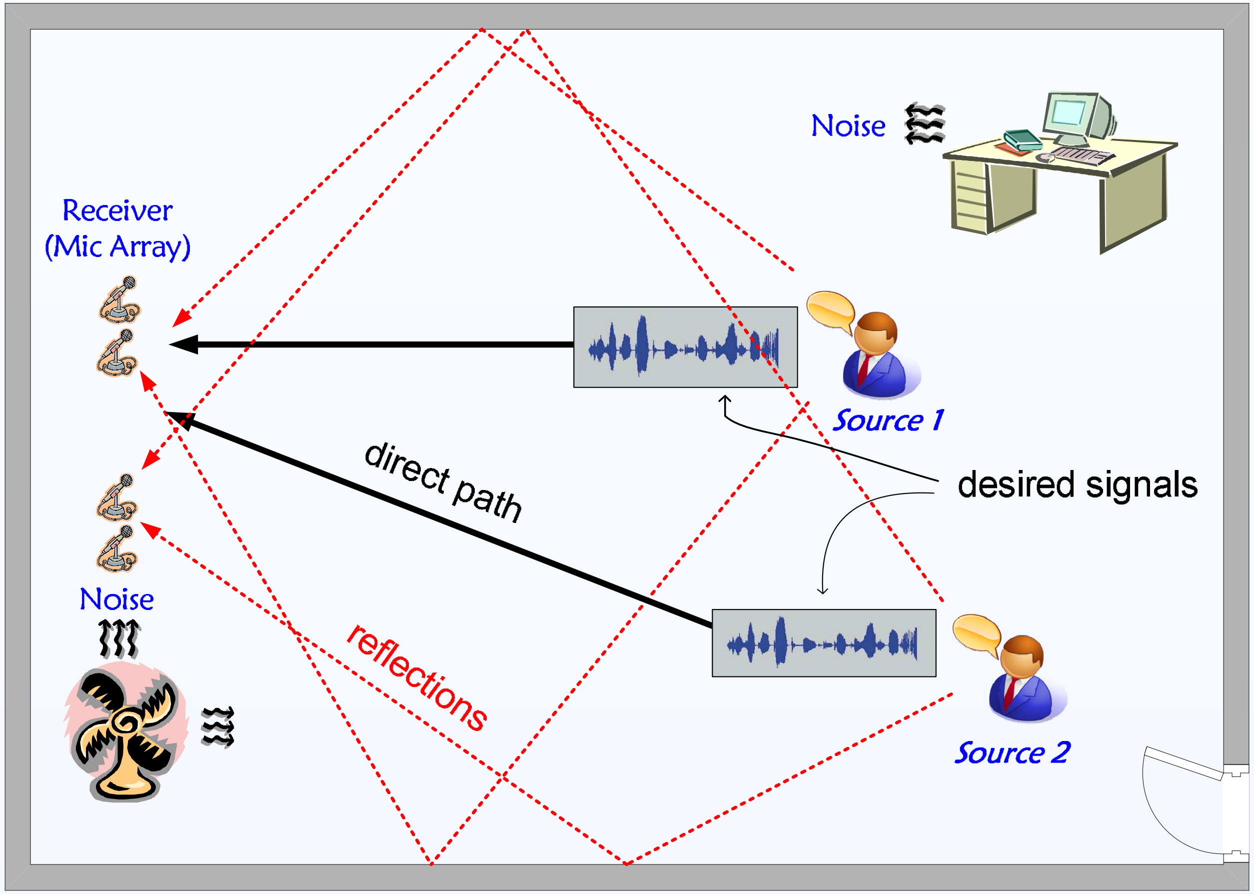
University of Edinburgh

Blank Page

This slide is intentionally left blank.

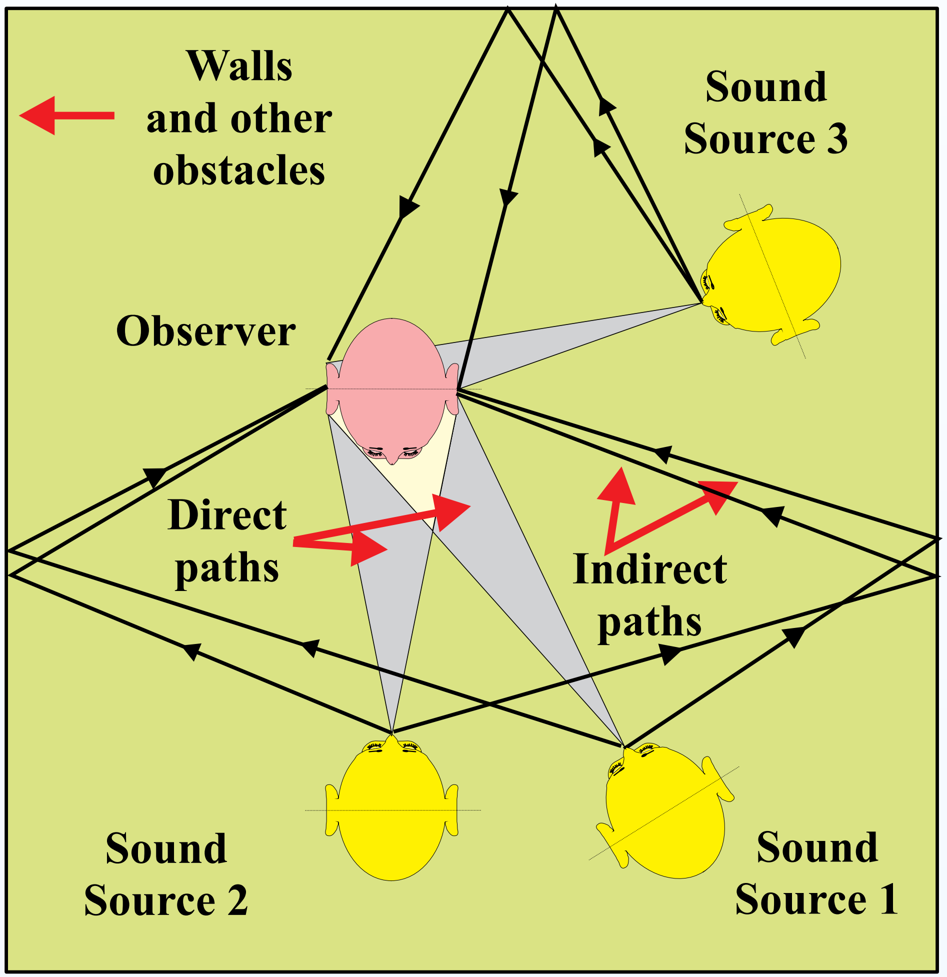
Handout 1
Source Localisation

Introduction



Source localisation and blind source separation (BSS).

Introduction



Humans turn their head in the direction of interest in order to reduce inteference from other directions; *joint detection, localisation, and enhancement*.

Introduction

- This research tutorial is intended to cover a wide range of aspects which link acoustic source localisation (ASL) and blind source separation (BSS).
- This tutorial is being continually updated, and feedback is welcomed. The documents published on the USB stick may differ to the slides presented on the day.
- The latest version of this document can be found online and downloaded at:

  http://www.see.ed.ac.uk/~jhopgoo1/Research/UDRC

- Thanks to Xionghu Zhong and Ashley Hughes for borrowing some of their diagrams from their dissertations.

Structure of the Tutorial

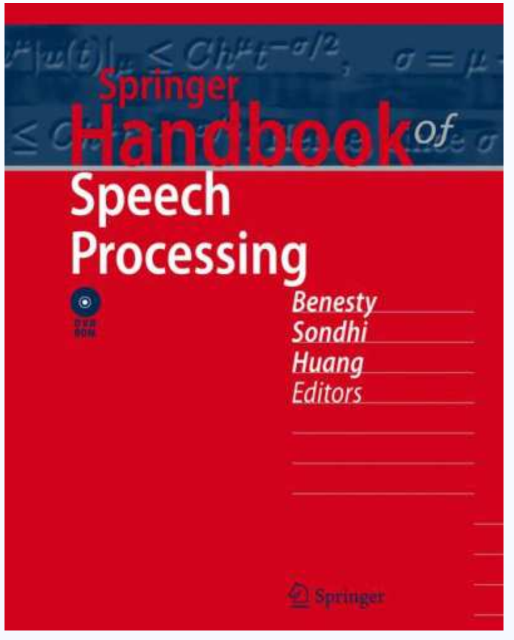
- Recommended Texts
- Conceptual link between ASL and BSS.
- Geometry of source localisation.
- Spherical and hyperboloidal localisation.
- Estimating TDOAs.
- Steered beamformer response function.
- Multiple target localisation using BSS.
- Conclusions.

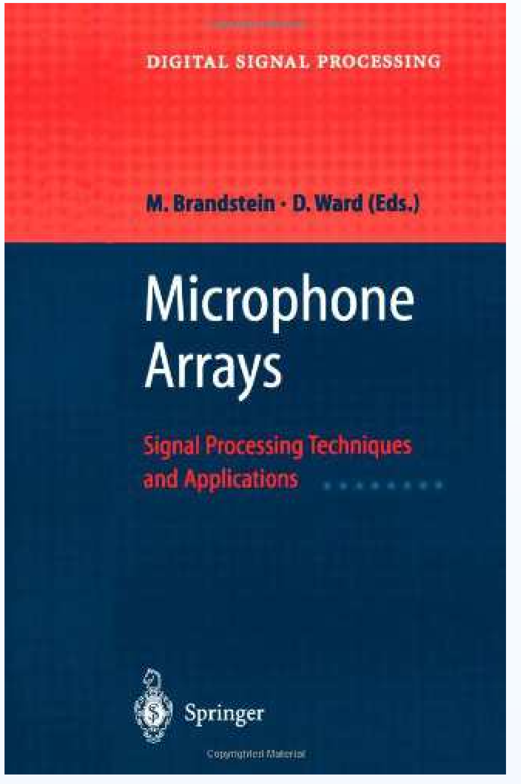
Recommended Texts

Recommended book chapters and the references therein.

- Huang Y., J. Benesty, and J. Chen, “Time Delay Estimation and Source Localization,” in *Springer Handbook of Speech Processing* by J. Benesty, M. M. Sondhi, and Y. Huang, pp. 1043–1063, , Springer, 2008.

Recommended Texts

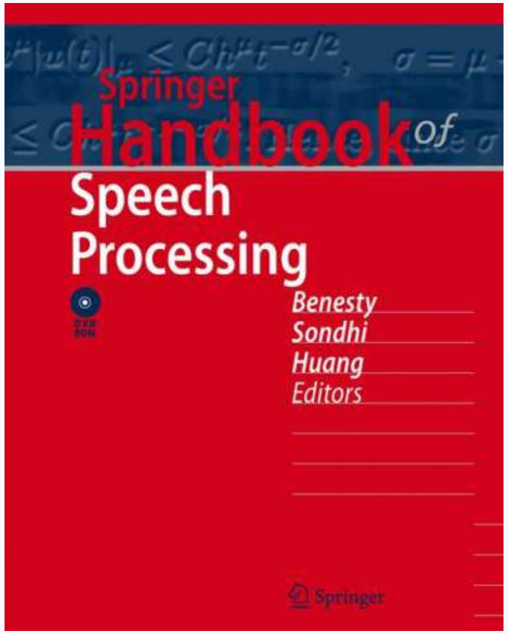


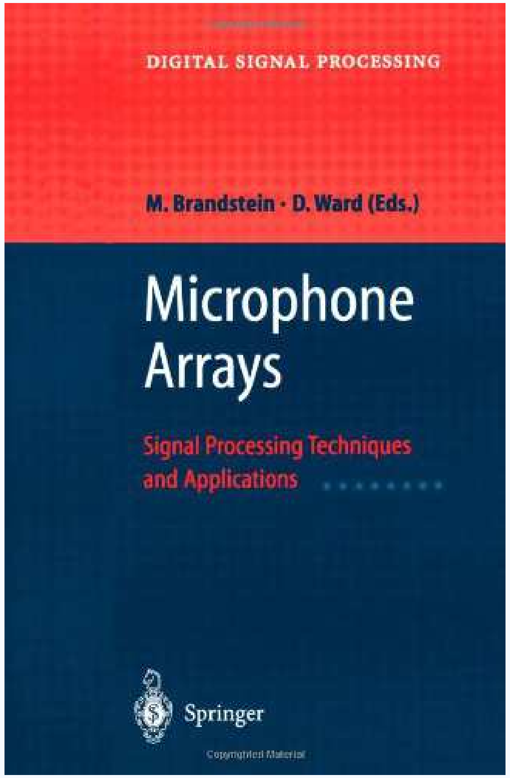


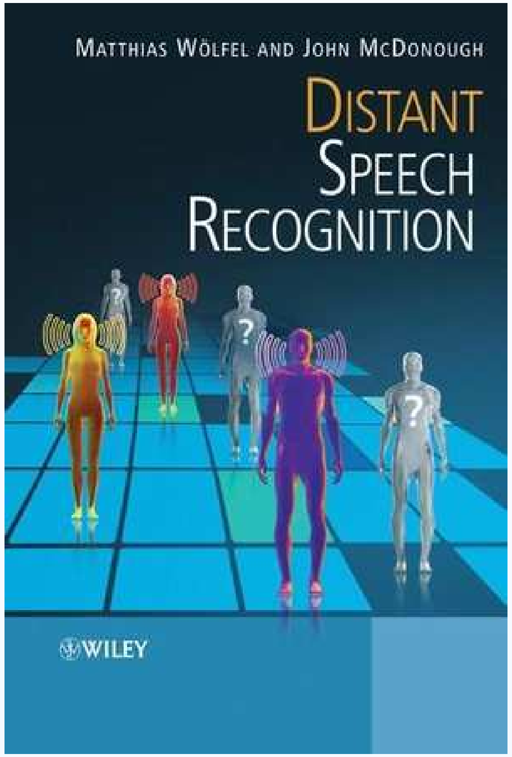
Recommended book chapters and the references therein.

- Chapter 8: DiBiase J. H., H. F. Silverman, and M. S. Brandstein, "Robust Localization in Reverberant Rooms," in *Microphone Arrays* by M. Brandstein and D. Ward, pp. 157–180, , Springer Berlin Heidelberg, 2001.

Recommended Texts







Recommended book chapters and the references therein.

- Chapter 10 of Wolfel M. and J. McDonough, *Distant Speech Recognition*, Wiley, 2009.

IDENTIFIERS – *Hardback*, ISBN13: 978-0-470-51704-8

Recommended Texts

Some recent PhD thesis on the topic include:

- Zhong X., *“Bayesian framework for multiple acoustic source tracking,”* Ph.D. thesis, University of Edinburgh, 2010.
- Pertila P., *“Acoustic Source Localization in a Room Environment and at Moderate Distances,”* Ph.D. thesis, Tampere University of Technology, 2009.
- Fallon M., *“Acoustic Source Tracking using Sequential Monte Carlo,”* Ph.D. thesis, University of Cambridge, 2008.

Why Source Localisation?

A number of blind source separation (BSS) techniques rely on knowledge of the desired source position:

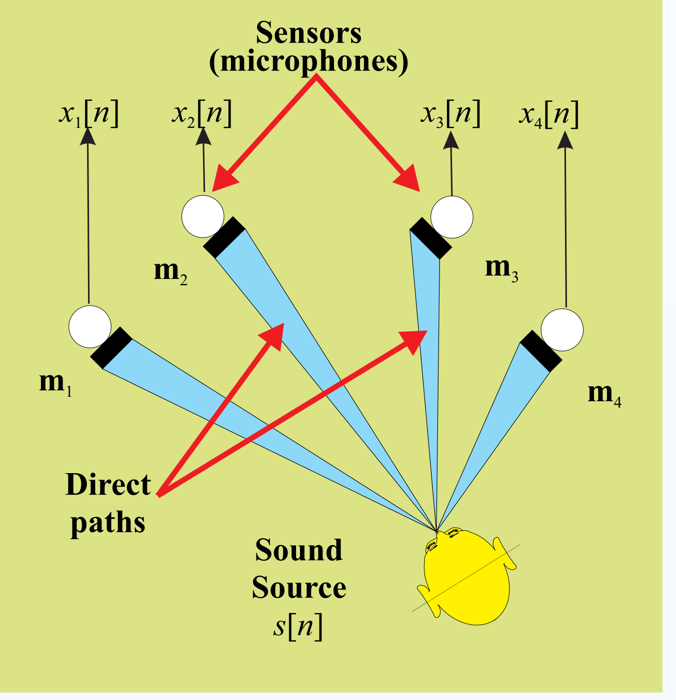
1. Look-direction in beamforming techniques.

2. Camera steering for audio-visual BSS (including Robot Audition).

3. Parametric modelling of the mixing matrix.

Equally, a number of multi-target acoustic source localisation (ASL) techniques rely on BSS.

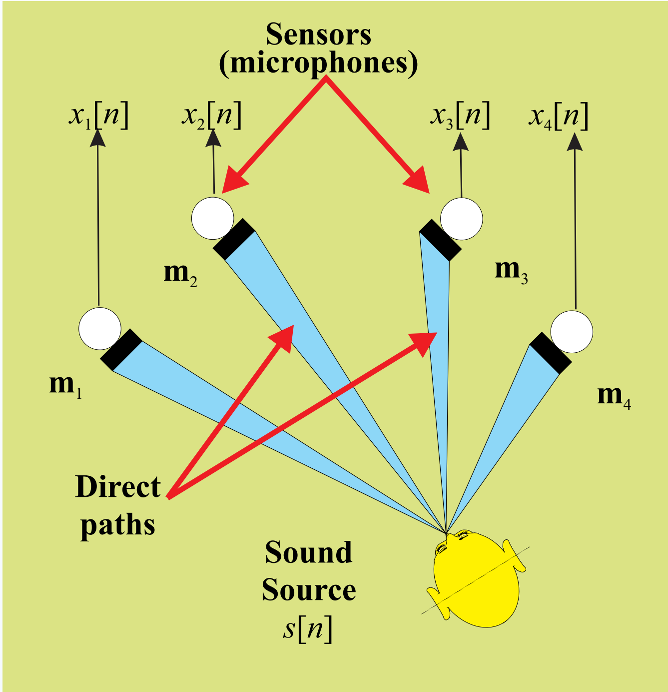
ASL Methodology



Ideal free-field model.

- Most ASL techniques rely on the fact that an impinging wavefront reaches one sensor before it reaches another.

ASL Methodology



Ideal free-field model.

- Most ASL techniques rely on the fact that an impinging wavefront reaches one sensor before it reaches another.
- Most ASL algorithms are designed assuming there is no reverberation present, the *free-field assumption*.

ASL Methodology

An uniform linear array (ULA) of microphones.

- Typically, this acoustic sensor is a microphone; will primarily consider *omni-directional pressure sensors*, and rely on the TDOA between the signals at different microphones.

ASL Methodology

An ULA of microphones.

- Typically, this acoustic sensor is a microphone; will primarily consider *omni-directional pressure sensors*, and rely on the TDOA between the signals at different microphones.
- Other measurement types include:
  - range difference measurements;
  - interaural level difference;
  - joint TDOA and vision techniques.

ASL Methodology

- Another sensor modality might include acoustic vector sensors (AVSs) which measure both air pressure and air velocity. Useful for applications such as sniper localisation.

An acoustic vector sensor.

Source Localization Strategies

Existing source localisation methods can loosely be divided into three generic strategies:

1. those based on maximising the steered response power (SRP) of a beamformer;

   - location estimate derivded directly from a filtered, weighted, and sum version of the signal data.

Source Localization Strategies

Existing source localisation methods can loosely be divided into three generic strategies:

1. those based on maximising the SRP of a beamformer;
   - location estimate derivded directly from a filtered, weighted, and sum version of the signal data.
2. techniques adopting high-resolution spectral estimation concepts (see Stephan Weiss's talk);
   - any localisation scheme relying upon an application of the signal correlation matrix.

Source Localization Strategies

Existing source localisation methods can loosely be divided into three generic strategies:

1. those based on maximising the SRP of a beamformer;
   - location estimate derivded directly from a filtered, weighted, and sum version of the signal data.
2. techniques adopting high-resolution spectral estimation concepts (see Stephan Weiss's talk);
   - any localisation scheme relying upon an application of the signal correlation matrix.
3. approaches employing TDOA information.
   - source locations calculated from a set of TDOA estimates measured across various combinations of microphones.

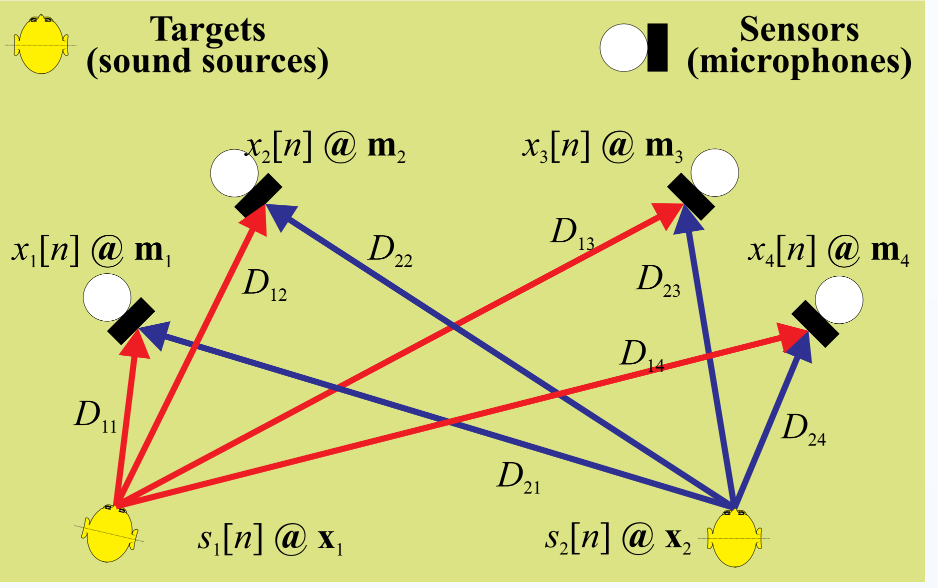
Source Localization Strategies

Spectral-estimation approaches See Stephan Weiss's talk :-)

TDOA-based estimators Computationally cheap, but suffers in the presence of noise and reverberation.

SBF approaches Computationally intensive, superior performance to TDOA-based methods. However, possible to dramatically reduce computational load.

Geometric Layout

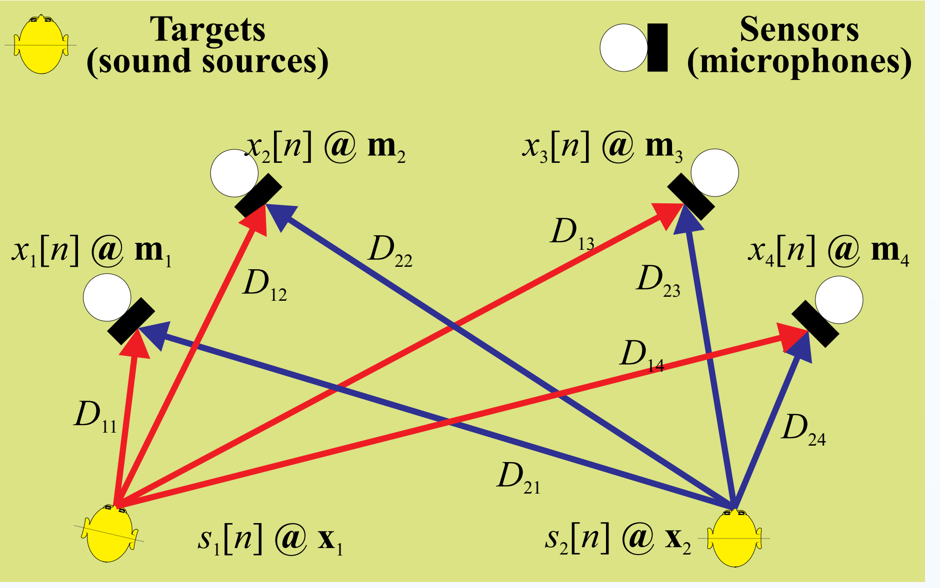


Geometry assuming a free-field model.

Suppose there is a:

- sensor array consisting of N microphones located at positions $\mathbf{m}_i \in \mathbb{R}^3$, for $i \in \{0, \dots, N-1\}$,
- M talkers (or targets) at positions $\mathbf{x}_k \in \mathbb{R}^3$, for $k \in \{0, \dots, M-1\}$.

Geometric Layout



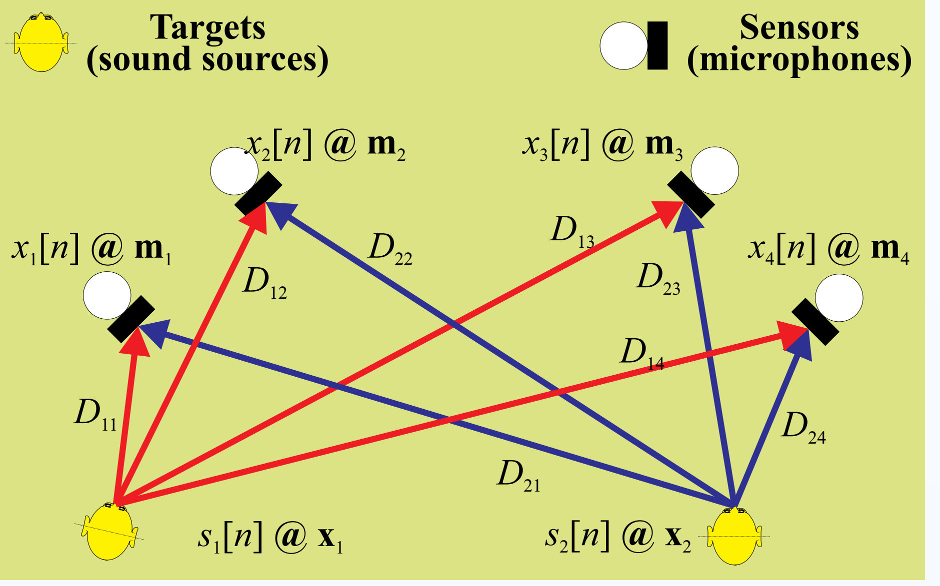
Geometry assuming a free-field model.

The TDOA between the microphones at position $\mathbf{m}_i$ and $\mathbf{m}_j$ due to a source at $\mathbf{x}_k$ can be expressed as:

$$T(\mathbf{m}_i, \mathbf{m}_j, \mathbf{x}_k) \triangleq T_{ij}(\mathbf{x}_k) = \frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}$$

where c is the speed of sound, which is approximately 344 m/s.

Geometric Layout



Geometry assuming a free-field model.

The distance from the target at $\mathbf{x}_k$ to the sensor located at $\mathbf{m}_i$ will be defined by D_{ik}, and is called the range.

$$T_{ij}(\mathbf{x}_k) = \frac{1}{c}\left(D_{ik} - D_{jk}\right)$$

Ideal Free-field Model

- In an anechoic free-field acoustic environment, the signal from source k, denoted by $s_k(t)$, propagates to the i-th sensor at time t according to the expression:

$$x_{ik}(t) = \alpha_{ik}\, s_k(t - \tau_{ik}) + b_{ik}(t)$$

where $b_{ik}(t)$ denotes additive noise. Note that, in the frequency domain, this expression is given by:

$$X_{ik}\,(\omega) = \alpha_{ik}\, S_k\,(\omega)\; e^{-j\omega\,\tau_{ik}} + B_{ik}\,(\omega)$$

- The additive noise source is assumed to be uncorrelated with the source signal, as well as the noise signals at the other microphones.

Ideal Free-field Model

- In an anechoic free-field acoustic environment, the signal from source k, denoted by $s_k(t)$, propagates to the i-th sensor at time t according to the expression:

$$x_{ik}(t) = \alpha_{ik}\, s_k(t - \tau_{ik}) + b_{ik}(t)$$

where $b_{ik}(t)$ denotes additive noise. Note that, in the frequency domain, this expression is given by:

$$X_{ik}(\omega) = \alpha_{ik}\, S_k(\omega)\, e^{-j\omega\,\tau_{ik}} + B_{ik}(\omega)$$

- The additive noise source is assumed to be uncorrelated with the source signal, as well as the noise signals at the other microphones.

- The TDOA between the i-th and j-th microphone is given by:

$$\tau_{ijk} = \tau_{ik} - \tau_{jk} = T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right)$$

TDOA and Hyperboloids

It is important to be aware of the geometrical properties that arise from the TDOA relationship

$$T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right) = \frac{\left|\mathbf{x}_k - \mathbf{m}_i\right| - \left|\mathbf{x}_k - \mathbf{m}_j\right|}{c}$$

TDOA and Hyperboloids

It is important to be aware of the geometrical properties that arise from the TDOA relationship

$$T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right) = \frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}$$

- This defines one half of a hyperboloid of two sheets, centered on the midpoint of the microphones, $\mathbf{v}_{ij} = \frac{\mathbf{m}_i + \mathbf{m}_j}{2}$.

$$\left(\mathbf{x}_k - \mathbf{v}_{ij}\right)^T \mathbf{V}_{ij} \left(\mathbf{x}_k - \mathbf{v}_{ij}\right) = 1$$

TDOA and Hyperboloids

It is important to be aware of the geometrical properties that arise from the TDOA relationship

$$T\left(\mathbf{m}_i, \mathbf{m}_j, \mathbf{x}_k\right) = \frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}$$

- This defines one half of a hyperboloid of two sheets, centered on the midpoint of the microphones, $\mathbf{v}_{ij} = \frac{\mathbf{m}_i + \mathbf{m}_j}{2}$.

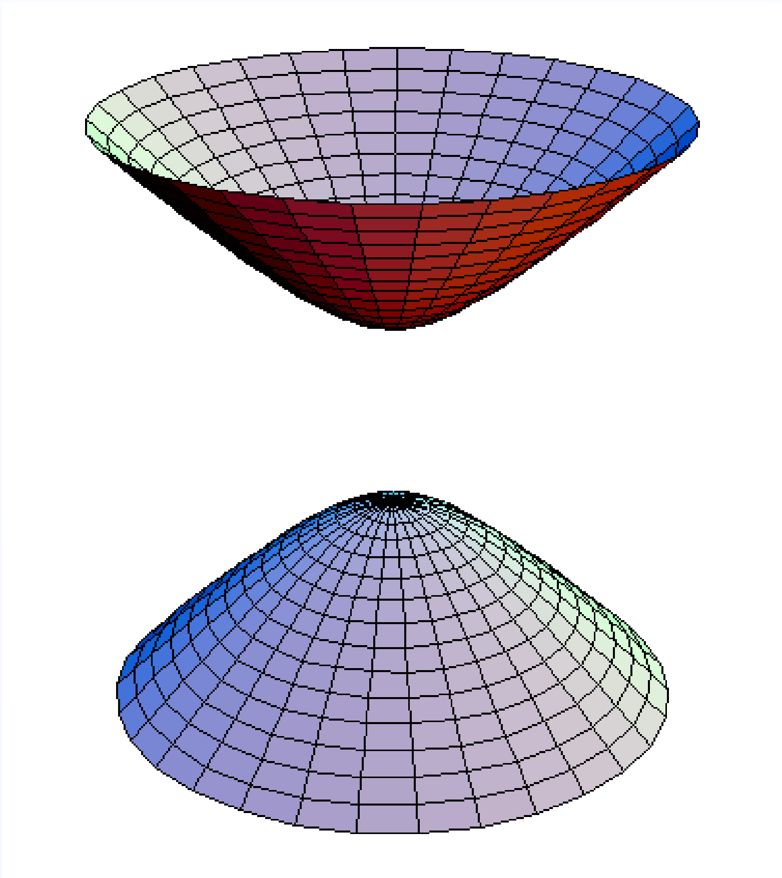
$$\left(\mathbf{x}_k - \mathbf{v}_{ij}\right)^T \mathbf{V}_{ij} \left(\mathbf{x}_k - \mathbf{v}_{ij}\right) = 1$$

- For source with a large source-range to microphone-separation ratio, the hyperboloid may be well-approximated by a cone with a constant direction angle relative to the axis of symmetry.

$$\phi_{ij} = \cos^{-1}\left(\frac{c\,T\left(\mathbf{m}_i, \mathbf{m}_j, \mathbf{x}_k\right)}{|\mathbf{m}_i - \mathbf{m}_j|}\right)$$

TDOA and Hyperboloids

$$T(\mathbf{m}_i, \mathbf{m}_j, \mathbf{x}_k) = \frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}$$

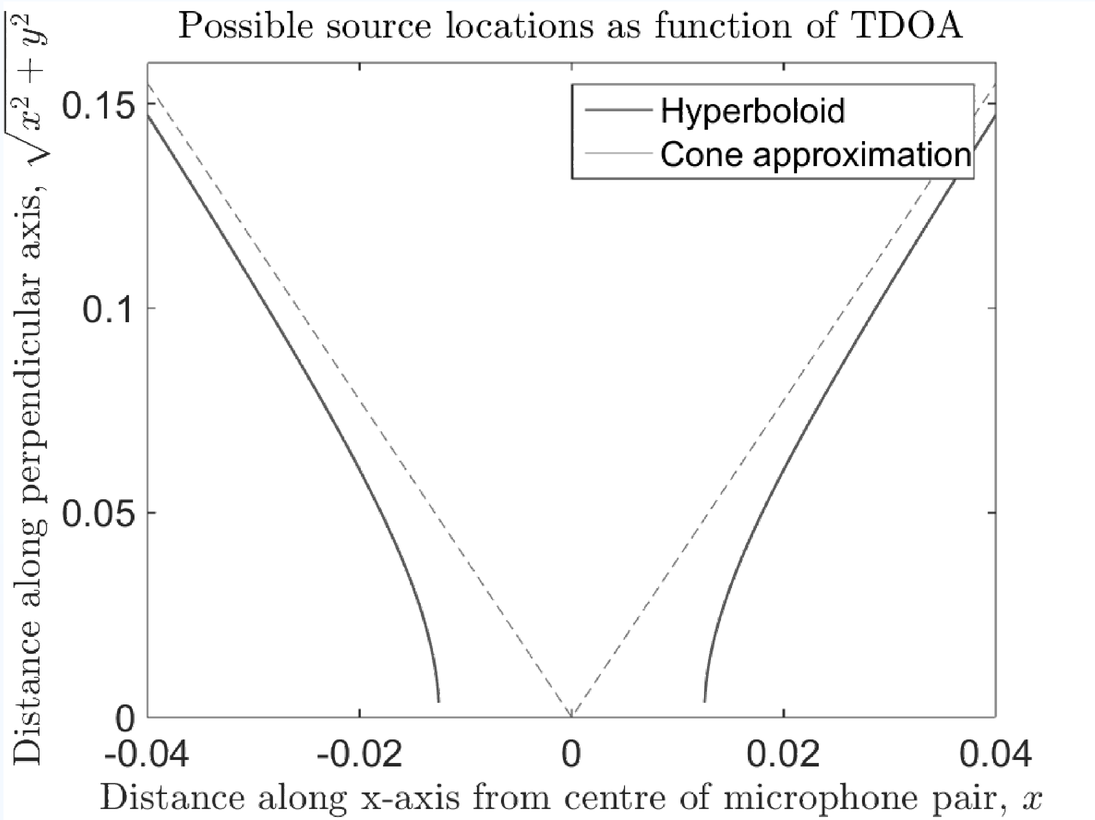


Hyperboloid of two sheets

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = -1$$

TDOA and Hyperboloids

$$T\left(\mathbf{m}_i, \mathbf{m}_j, \mathbf{x}_k\right) = \frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}$$



Hyperboloid, for a microphone separation of $d = 0.1$, and a time-delay of $\tau_{ij} = \frac{d}{4c}$.

Indirect TDOA-based Methods

This is typically a two-step procedure in which:

- Typically, TDOAs are extracted using the GCC function, or an adaptive eigenvalue decomposition (AED) algorithm.

Indirect TDOA-based Methods

This is typically a two-step procedure in which:

- Typically, TDOAs are extracted using the GCC function, or an AED algorithm.
- A hypothesised spatial position of the target can be used to predict the expected TDOAs (or corresponding range) at the microphone.

Indirect TDOA-based Methods

This is typically a two-step procedure in which:

- Typically, TDOAs are extracted using the GCC function, or an AED algorithm.
- A hypothesised spatial position of the target can be used to predict the expected TDOAs (or corresponding range) at the microphone.
- The error between the measured and hypothesised TDOAs is then minimised.

Indirect TDOA-based Methods

This is typically a two-step procedure in which:

- Typically, TDOAs are extracted using the GCC function, or an AED algorithm.
- A hypothesised spatial position of the target can be used to predict the expected TDOAs (or corresponding range) at the microphone.
- The error between the measured and hypothesised TDOAs is then minimised.
- Accurate and robust TDOA estimation is the key to the effectiveness of this class of ASL methods.

Indirect TDOA-based Methods

This is typically a two-step procedure in which:

- Typically, TDOAs are extracted using the GCC function, or an AED algorithm.
- A hypothesised spatial position of the target can be used to predict the expected TDOAs (or corresponding range) at the microphone.
- The error between the measured and hypothesised TDOAs is then minimised.
- Accurate and robust TDOA estimation is the key to the effectiveness of this class of ASL methods.
- An alternative way of viewing these solutions is to consider what **spatial positions** of the target could lead to the estimated TDOA.

Spherical Least Squares Error Function

- Suppose the first microphone is located at the origin of the coordinate system, such that $\mathbf{m_0} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$.

- The range from target k to sensor i can be expressed as :

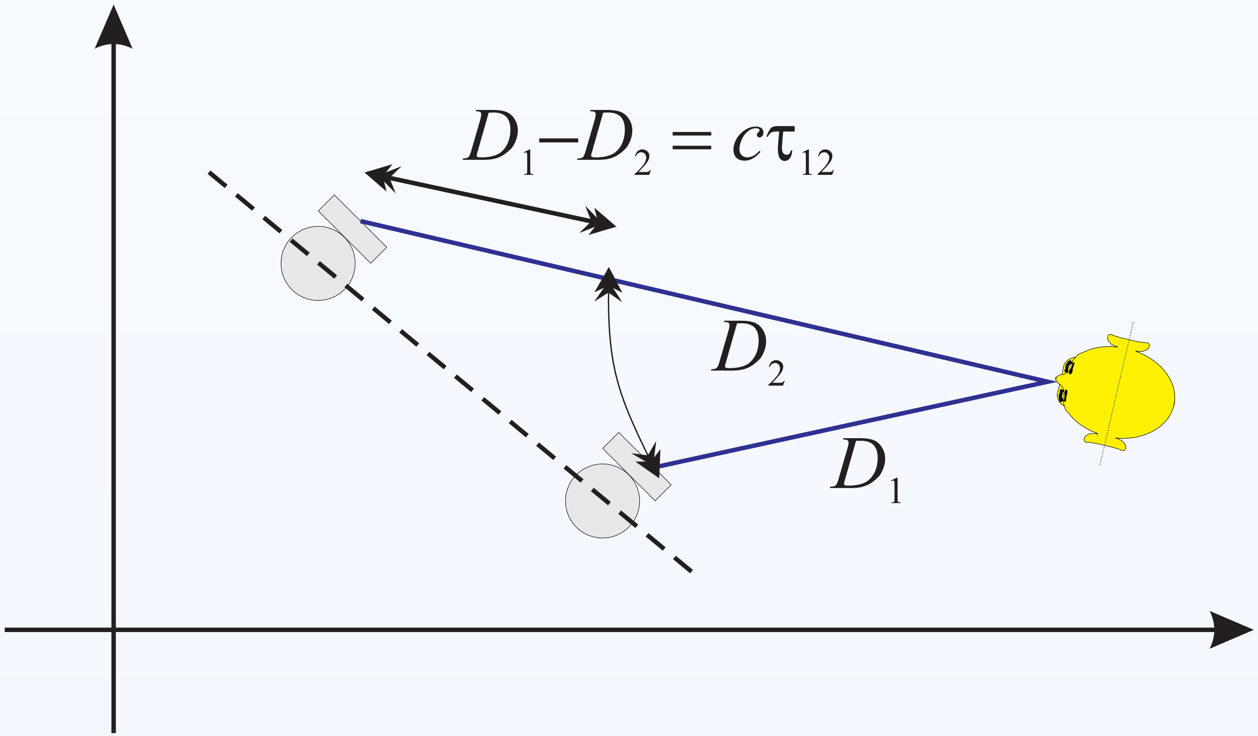
$$\begin{aligned} D_{ik} &= D_{0k} + D_{ik} - D_{0k} \\ &= R_s + c\,T_{i0}\left(\mathbf{x}_k\right) \end{aligned}$$

where $R_{sk} = |\mathbf{x}_k|$ is the range to the first microphone which is at the origin.

Spherical Least Squares Error Function

- In practice, the observations are the TDOAs and, given R_{sk}, these ranges can be considered the **measurement ranges**.

  Of course, knowing R_{sk} is half the solution, but it is just one unknown at this stage.



Range and TDOA relationship.

Spherical Least Squares Error Function

- The source-sensor geometry states that the target lies on a sphere centered on the corresponding sensor. Hence,

$$
\begin{aligned}
D_{ik}^2 &= |\mathbf{x}_k - \mathbf{m}_i|^2 \\
&= \mathbf{x}_k^T \mathbf{x}_k - 2\mathbf{m}_i^T \mathbf{x}_k + \mathbf{m}_i^T \mathbf{m}_i \\
&= R_s^2 - 2\mathbf{m}_i^T \mathbf{x}_k + R_i^2
\end{aligned}
$$

$R_i = |\mathbf{m}_i|$ is the distance of the i-th microphone to the origin.

Spherical Least Squares Error Function

- The source-sensor geometry states that the target lies on a sphere centered on the corresponding sensor. Hence,

$$\begin{aligned} D_{ik}^2 &= \left|\mathbf{x}_k - \mathbf{m}_i\right|^2 \\ &= \mathbf{x}_k^T\,\mathbf{x}_k - 2\mathbf{m}_i^T\,\mathbf{x}_k + \mathbf{m}_i^T\,\mathbf{m}_i \\ &= R_s^2 - 2\mathbf{m}_i^T\,\mathbf{x}_k + R_i^2 \end{aligned}$$

- Define the **spherical error function** as:

$$\epsilon_{ik} \triangleq \frac{1}{2}\left(\hat{D}_{ik}^2 - D_{ik}^2\right)$$

Spherical Least Squares Error Function

- The source-sensor geometry states that the target lies on a sphere centered on the corresponding sensor. Hence,

$$
\begin{aligned}
D_{ik}^2 &= \left|\mathbf{x}_k - \mathbf{m}_i\right|^2 \\
&= \mathbf{x}_k^T \mathbf{x}_k - 2\mathbf{m}_i^T \mathbf{x}_k + \mathbf{m}_i^T \mathbf{m}_i \\
&= R_s^2 - 2\mathbf{m}_i^T \mathbf{x}_k + R_i^2
\end{aligned}
$$

- Define the **spherical error function** as:

$$
\begin{aligned}
\epsilon_{ik} &\triangleq \frac{1}{2}\left(\hat{D}_{ik}^2 - D_{ik}^2\right) \\
&= \frac{1}{2}\left\{\left(R_s + c\,\hat{T}_{i0}\right)^2 - \left(R_s^2 - 2\mathbf{m}_i^T \mathbf{x}_k + R_i^2\right)\right\}
\end{aligned}
$$

Spherical Least Squares Error Function

- The source-sensor geometry states that the target lies on a sphere centered on the corresponding sensor. Hence,

$$
\begin{aligned}
D_{ik}^2 &= |\mathbf{x}_k - \mathbf{m}_i|^2 \\
&= \mathbf{x}_k^T \mathbf{x}_k - 2\mathbf{m}_i^T \mathbf{x}_k + \mathbf{m}_i^T \mathbf{m}_i \\
&= R_s^2 - 2\mathbf{m}_i^T \mathbf{x}_k + R_i^2
\end{aligned}
$$

- Define the **spherical error function** as:

$$
\begin{aligned}
\epsilon_{ik} &\triangleq \frac{1}{2}\left(\hat{D}_{ik}^2 - D_{ik}^2\right) \\
&= \frac{1}{2}\left\{\left(R_s + c\,\hat{T}_{i0}\right)^2 - \left(R_s^2 - 2\mathbf{m}_i^T \mathbf{x}_k + R_i^2\right)\right\} \\
&= \mathbf{m}_i^T \mathbf{x}_\mathbf{k} + c\,R_s\,\hat{T}_{i0} + \frac{1}{2}\left(c^2 \hat{T}_{i0}^2 - R_i^2\right)
\end{aligned}
$$

Spherical Least Squares Error Function

- Concatenating the error functions for each microphone gives the expression:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\,\mathbf{x}_k - \underbrace{(\mathbf{b}_k - R_{sk}\mathbf{d}_k)}_{\mathbf{v}_k}$$

$$\equiv \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{d}_k \end{bmatrix}}_{\mathbf{S}_k} \underbrace{\begin{bmatrix} \mathbf{x}_k \\ R_{sk} \end{bmatrix}}_{\boldsymbol{\theta}_k} - \mathbf{b}_k$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{m}_0^T \\ \vdots \\ \mathbf{m}_{N-1}^T \end{bmatrix}, \quad \mathbf{d} = c \begin{bmatrix} \hat{T}_{00} \\ \vdots \\ \hat{T}_{(N-1)0} \end{bmatrix}, \quad \mathbf{b}_k = \frac{1}{2} \begin{bmatrix} c^2 \hat{T}_{00}^2 - R_0^2 \\ \vdots \\ c^2 \hat{T}_{(N-1)0}^2 - R_{N-1}^2 \end{bmatrix}$$

Spherical Least Squares Error Function

- The least-squares estimate (LSE) can then be obtained by using $J = \boldsymbol{\epsilon}_i^T \boldsymbol{\epsilon}_i$:

$$J(\mathbf{x}_k) = \left(\mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)\right)^T \left(\mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)\right)$$

$$J\left(\mathbf{x}_k,\, \boldsymbol{\theta}_k\right) = \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)^T \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)$$

Spherical Least Squares Error Function

- The LSE can then be obtained by using $J = \boldsymbol{\epsilon}_i^T \boldsymbol{\epsilon}_i$:

$$J(\mathbf{x}_k) = (\mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k))^T (\mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k))$$

$$J\left(\mathbf{x}_k,\, \boldsymbol{\theta}_k\right) = \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)^T \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)$$

- Note that as $R_{sk} = |\mathbf{x}_k|$, these parameters aren't independent. Therefore, the problem can either be formulated as:
  - a nonlinear least-squares problem in $\mathbf{x}_k$;
  - a linear minimisation subject to quadratic constraints:

$$\hat{\boldsymbol{\theta}}_k = \arg\min_{\boldsymbol{\theta}_k} \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)^T \left(\mathbf{S}_k\boldsymbol{\theta}_k - \mathbf{b}_k\right)$$

  subject to the constraint

$$\boldsymbol{\theta}_k\,\Delta\,\boldsymbol{\theta}_k = 0 \qquad \text{where} \qquad \Delta = \text{diag}\left[1,\,1,\,1,\,-1\right]$$

Spherical Least Squares Error Function

Two-step Spherical LSE Approaches

To avoid solving either a nonlinear or a constrained least-squares problem, it is possible to solve the problem in two steps, namely:

1. solving a LLS problem in $\mathbf{x}_k$ *assuming* the range to the target, R_{sk}, is known;

2. and then solving for R_{sk} given an estimate of $\mathbf{x}_k$ in terms of (i. t. o.) R_{sk}.

Two-step Spherical LSE Approaches

To avoid solving either a nonlinear or a constrained least-squares problem, it is possible to solve the problem in two steps, namely:

1. solving a LLS problem in $\mathbf{x}_k$ *assuming* the range to the target, R_{sk}, is known;
2. and then solving for R_{sk} given an estimate of $\mathbf{x}_k$ i. t. o. R_{sk}.

- Assuming an estimate of R_{sk} this can be solved as

$$\hat{\mathbf{x}}_k = \mathbf{A}^\dagger \mathbf{v}_k = \mathbf{A}^\dagger \left(\mathbf{b}_k - \hat{R}_{sk} \mathbf{d}_k \right) \quad \text{where} \quad \mathbf{A}^\dagger = \left[\mathbf{A}^T \mathbf{A} \right]^{-1} \mathbf{A}^T$$

Note that $\mathbf{A}^\dagger$ is the pseudo-inverse of $\mathbf{A}$.

Spherical Intersection Estimator

This method uses the physical constraint that the range R_{sk} is the Euclidean distance to the target.

- Writing $\hat{R}_{sk}^2 = \hat{\mathbf{x}}_k^T \, \hat{\mathbf{x}}_k$, it follows that:

$$\hat{R}_{sk}^2 = \left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)^T \mathbf{A}^{\dagger\,T}\mathbf{A}^{\dagger} \left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)$$

Spherical Intersection Estimator

This method uses the physical constraint that the range R_{sk} is the Euclidean distance to the target.

- Writing $\hat{R}_{sk}^2 = \hat{\mathbf{x}}_k^T \, \hat{\mathbf{x}}_k$, it follows that:

$$\hat{R}_{sk}^2 = \left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)^T \mathbf{A}^{\dagger\,T}\mathbf{A}^{\dagger}\left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)$$

which can be written as the quadratic:

$$a\,\hat{R}_{sk}^2 + b\,\hat{R}_{sk} + c = 0$$

Spherical Intersection Estimator

This method uses the physical constraint that the range R_{sk} is the Euclidean distance to the target.

- Writing $\hat{R}_{sk}^2 = \hat{\mathbf{x}}_k^T \, \hat{\mathbf{x}}_k$, it follows that:

$$\hat{R}_{sk}^2 = \left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)^T \mathbf{A}^{\dagger\,T}\mathbf{A}^{\dagger}\left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right)$$

  which can be written as the quadratic:

$$a\,\hat{R}_{sk}^2 + b\,\hat{R}_{sk} + c = 0$$

- The unique, real, positive root is taken as the spherical intersection (SX) estimator of the source range. Hence, the estimator will fail when:

  1. there is no real, positive root, or:

  2. if there are two positive real roots.

Spherical Interpolation Estimator

The spherical interpolation (SI) estimator again uses the spherical least squares error (LSE) function, but this time the range R_{sk} is estimated in the least-squares sense.

Consider again the **spherical error function**:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Spherical Interpolation Estimator

The SI estimator again uses the spherical LSE function, but this time the range R_{sk} is estimated in the least-squares sense.

Consider again the **spherical error function**:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Substituting the LSE gives:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T\left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right) - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Spherical Interpolation Estimator

The SI estimator again uses the spherical LSE function, but this time the range R_{sk} is estimated in the least-squares sense.

Consider again the **spherical error function**:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Substituting the LSE gives:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T\left(\mathbf{b}_k - \hat{R}_{sk}\mathbf{d}_k\right) - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Defining the projection matrix as $\mathbf{P}_\mathbf{A} = \mathbf{I}_N - \mathbf{A}\left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T$,

$$\boldsymbol{\epsilon}_{ik} = R_{sk}\,\mathbf{P}_\mathbf{A}\mathbf{d}_k - \mathbf{P}_\mathbf{A}\mathbf{b}_k$$

Spherical Interpolation Estimator

The SI estimator again uses the spherical LSE function, but this time the range R_{sk} is estimated in the least-squares sense.

Consider again the **spherical error function**:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Defining the projection matrix as $\mathbf{P_A} = \mathbf{I}_N - \mathbf{A}\left[\mathbf{A}^T\mathbf{A}\right]^{-1}\mathbf{A}^T$,

$$\boldsymbol{\epsilon}_{ik} = R_{sk}\,\mathbf{P_A}\mathbf{d}_k - \mathbf{P_A}\mathbf{b}_k$$

Minimising the LSE using the normal equations gives:

$$R_{sk} = \frac{\mathbf{d}_k^T\mathbf{P_A}\mathbf{b}_k}{\mathbf{d}_k^T\,\mathbf{P_A}\mathbf{d}_k}$$

Spherical Interpolation Estimator

The SI estimator again uses the spherical LSE function, but this time the range R_{sk} is estimated in the least-squares sense.

Consider again the **spherical error function**:

$$\boldsymbol{\epsilon}_{ik} = \mathbf{A}\mathbf{x}_k - (\mathbf{b}_k - R_{sk}\,\mathbf{d}_k)$$

Substituting back into the LSE for the target position gives the final estimator:

$$\hat{\mathbf{x}}_k = \mathbf{A}^\dagger \left(\mathbf{I}_N - \mathbf{d}_k \frac{\mathbf{d}_k^T \mathbf{P}_{\mathbf{A}}}{\mathbf{d}_k^T \mathbf{P}_{\mathbf{A}} \mathbf{d}_k} \right) \mathbf{b}_k$$

This approach is said to perform better, but is computationally slightly more complex than the SX estimator.

Other Approaches

There are several other approaches to minimising the spherical LSE function .

- In particular, the **linear-correction** LSE solves the constrained minimization problem using Lagrange multipliers in a two stage process.

- For further information, see: Huang Y., J. Benesty, and J. Chen, “Time Delay Estimation and Source Localization,” in *Springer Handbook of Speech Processing* by J. Benesty, M. M. Sondhi, and Y. Huang, pp. 1043–1063, , Springer, 2008.

Hyperbolic Least Squares Error Function

- If a TDOA is estimated between two microphones i and j, then the error between this and modelled TDOA is:

$$\epsilon_{ij}(\mathbf{x}_k) = \tau_{ijk} - T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right)$$

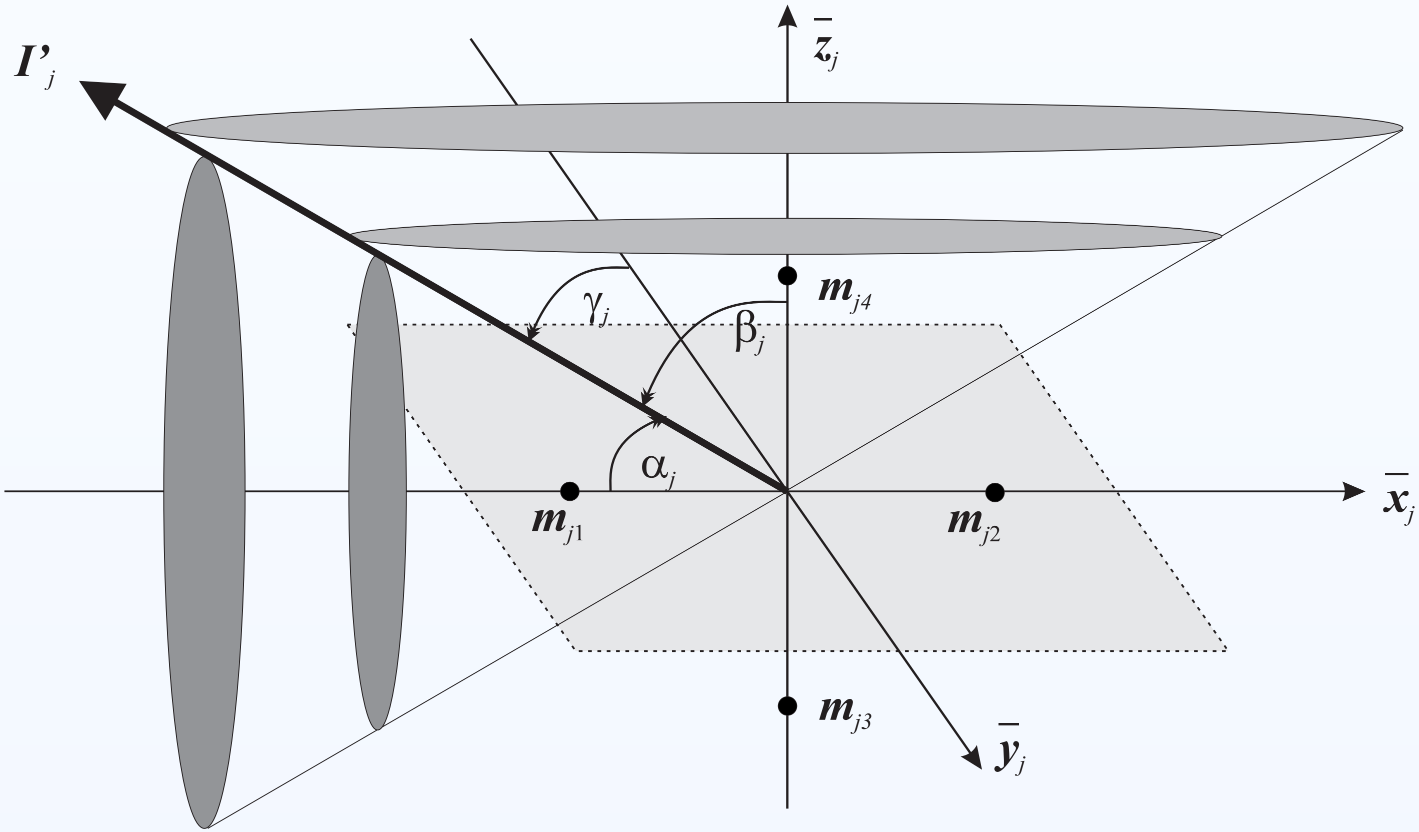
- The total error as a function of target position

$$J(\mathbf{x}_k) = \sum_{i=1}^{N} \sum_{j \neq i=1}^{N} \left(\tau_{ijk} - T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right)\right)^2$$

- Unfortunately, since $T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right)$ is a nonlinear function of $\mathbf{x}_k$, the minimum LSE does not possess a closed-form solution.

Linear Intersection Method

The linear intersection (LI) algorithm works by utilising a *sensor quadruple* with a common midpoint, which allows a bearing line to be deduced from the intersection of two cones.

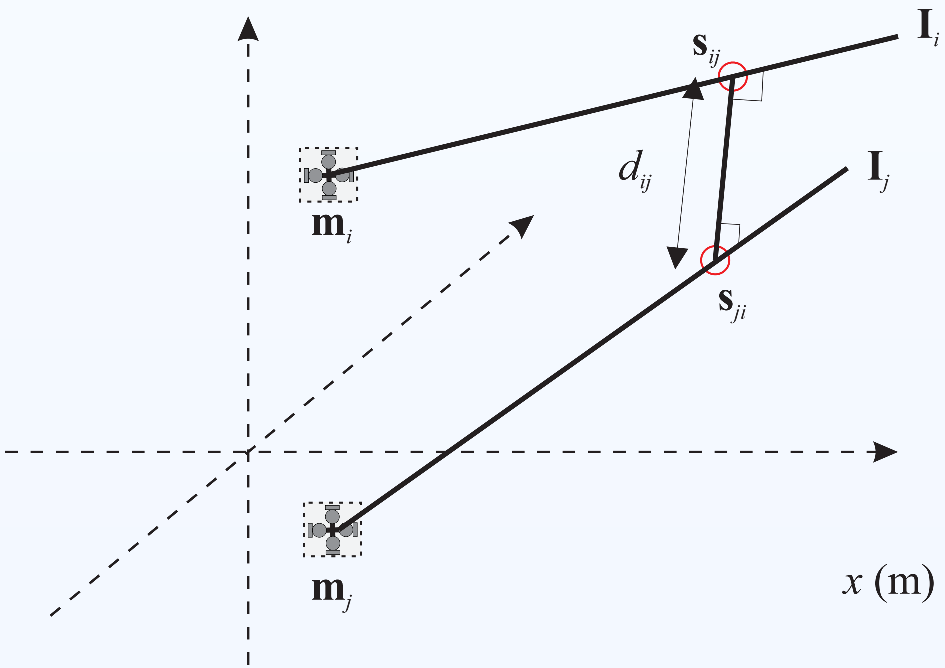


Quadruple sensor arrangement and local Cartesian coordinate system.

Linear Intersection Method

- Given the bearing lines, it is possible to calculate the points $\mathbf{s}_{ij}$ and $\mathbf{s}_{ji}$ on two bearing lines which give the closest intersection. This is basic gemoentry.

- The trick is to note that given these points $\mathbf{s}_{ij}$ and $\mathbf{s}_{ji}$, the theoretical TDOA, $T(\mathbf{m}_{1i}, \mathbf{m}_{2i}, \mathbf{s}_{ij})$, can be compared with the observed TDOA.



Calculating the points of closest intersection.

TDOA estimation methods

Two key methods for TDOA estimation are using the GCC function and the AED algorithm.

GCC algorithm most popular approach assuming an ideal free-field movel

- computationally efficient, and hence short decision delays;
- perform fairly well in moderately noisy and reverberant environments.

TDOA estimation methods

Two key methods for TDOA estimation are using the GCC function and the AED algorithm.

GCC algorithm most popular approach assuming an ideal free-field movel

- computationally efficient, and hence short decision delays;
- perform fairly well in moderately noisy and reverberant environments.

However, GCC-based methods

- fail when room reverberation is high;
- focus of current research is on combating the effect of room reverberation.

TDOA estimation methods

Two key methods for TDOA estimation are using the GCC function and the AED algorithm.

AED Algorithm Approaches the TDOA estimation approach from a different point of view from the *traditional* GCC method.

- adopts a reverberant rather than free-field model;
- computationally more expensive than GCC;
- can fail when there are common-zeros in the room impulse response (RIR).

GCC TDOA estimation

The GCC algorithm proposed by *Knapp and Carter* is the most widely used approach to TDOA estimation.

- The TDOA estimate between two microphones i and j

$$\hat{\tau_{ij}} = \arg\max_{\ell} r_{x_i\,x_j}[\ell]$$

- The cross-correlation function is given by

$$\begin{aligned} r_{x_i\,x_j}[\ell] &= \mathcal{F}^{-1}\left(\Phi\left(e^{j\omega T_s}\right) P_{x_1 x_2}\left(e^{j\omega T_s}\right)\right) \\ &= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} \Phi\left(e^{j\omega T_s}\right) P_{x_1 x_2}\left(e^{j\omega T_s}\right) \, e^{j\ell\omega T}\, d\omega \end{aligned}$$

where the cross-power spectral density (CPSD) is given by

$$P_{x_1 x_2}\left(e^{j\omega T_s}\right) = \mathbb{E}\left[X_1\left(e^{j\omega T_s}\right) X_2\left(e^{j\omega T_s}\right)\right]$$

CPSD for Free-Field Model

For the free-field model , it follows that for $i \neq j$:

$$
\begin{aligned}
P_{x_i x_j}(\omega) &= \mathbb{E}\left[X_j(\omega)\, X_j(\omega)\right] \\
&= \mathbb{E}\left[\left(\alpha_{ik}\, S_k(\omega)\; e^{-j\omega\,\tau_{ik}} + B_{ik}(\omega)\right)\left(\alpha_{jk}\, S_k(\omega)\; e^{-j\omega\,\tau_{kk}} + B_{jk}(\omega)\right)\right] \\
&= \alpha_{ik}\alpha_{jk} e^{-j\omega\, T(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k)}\mathbb{E}\left[\left|S_k(\omega)\right|^2\right]
\end{aligned}
$$

where $\mathbb{E}\left[B_{ik}(\omega)\, B_{jk}(\omega)\right] = 0$ and $\mathbb{E}\left[B_{ik}(\omega)\, S_k(\omega)\right] = 0$.

CPSD for Free-Field Model

For the free-field model , it follows that for $i \neq j$:

$$
\begin{aligned}
P_{x_i x_j}(\omega) &= \mathbb{E}\left[X_j(\omega)\, X_j(\omega)\right] \\
&= \mathbb{E}\left[\left(\alpha_{ik}\, S_k(\omega)\, e^{-j\omega\,\tau_{ik}} + B_{ik}(\omega)\right)\left(\alpha_{jk}\, S_k(\omega)\, e^{-j\omega\,\tau_{kk}} + B_{jk}(\omega)\right)\right] \\
&= \alpha_{ik}\alpha_{jk} e^{-j\omega\, T(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k)} \mathbb{E}\left[\left|S_k(\omega)\right|^2\right]
\end{aligned}
$$

where $\mathbb{E}\left[B_{ik}(\omega)\, B_{jk}(\omega)\right] = 0$ and $\mathbb{E}\left[B_{ik}(\omega)\, S_k(\omega)\right] = 0$.

- In particular, note that it follows:

$$
\angle P_{x_i x_j}(\omega) = -j\omega\, T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right)
$$

In otherwords, all the TDOA information is conveyed in the phrase rather than the amplitude of the CPSD. This therefore suggests that the weighting function can be chosen to remove the amplitude information.

GCC Processors

Processor Name	Frequency Function
Cross Correlation	1
PHAT	$\dfrac{1}{\left\|P_{x_1x_2}\left(e^{j\omega T_s}\right)\right\|}$
Roth Impulse Response	$\dfrac{1}{P_{x_1x_1}\left(e^{j\omega T_s}\right)}$ or $\dfrac{1}{P_{x_2x_2}\left(e^{j\omega T_s}\right)}$
SCOT	$\dfrac{1}{\sqrt{P_{x_1x_1}\left(e^{j\omega T_s}\right) P_{x_2x_2}\left(e^{j\omega T_s}\right)}}$
Eckart	$\dfrac{P_{s_1s_1}\left(e^{j\omega T_s}\right)}{P_{n_1n_1}\left(e^{j\omega T_s}\right) P_{n_2n_2}\left(e^{j\omega T_s}\right)}$
Hannon-Thomson or ML	$\dfrac{\left\|\gamma_{x_1x_2}\left(e^{j\omega T_s}\right)\right\|^2}{\left\|P_{x_1x_2}\left(e^{j\omega T_s}\right)\right\| \left(1 - \left\|\gamma_{x_1x_2}\left(e^{j\omega T_s}\right)\right\|^2\right)}$

where $\gamma_{x_1x_2}\left(e^{j\omega T_s}\right)$ is the normalised CPSD or **coherence function**

GCC Processors

The phase transform (PHAT)-GCC approach can be written as:

$$
\begin{aligned}
r_{x_i\,x_j}[\ell] &= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} \Phi\left(e^{j\omega T_s}\right) P_{x_1 x_2}\left(e^{j\omega T_s}\right)\, e^{j\ell\omega T}\, d\omega \\
&= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} \frac{1}{\left|P_{x_1 x_2}\left(e^{j\omega T_s}\right)\right|} \left|P_{x_1 x_2}\left(e^{j\omega T_s}\right)\right| e^{j\angle P_{x_1 x_2}\left(e^{j\omega T_s}\right)}\, e^{j\ell\omega T}\, d\omega \\
&= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} e^{j\left(\ell\omega T + \angle P_{x_1 x_2}\left(e^{j\omega T_s}\right)\right)}\, d\omega \\
&= \delta\left(\ell\, T_s + \angle P_{x_1 x_2}\left(e^{j\omega T_s}\right)\right) \\
&= \delta(\ell\, T_s - T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right))
\end{aligned}
$$

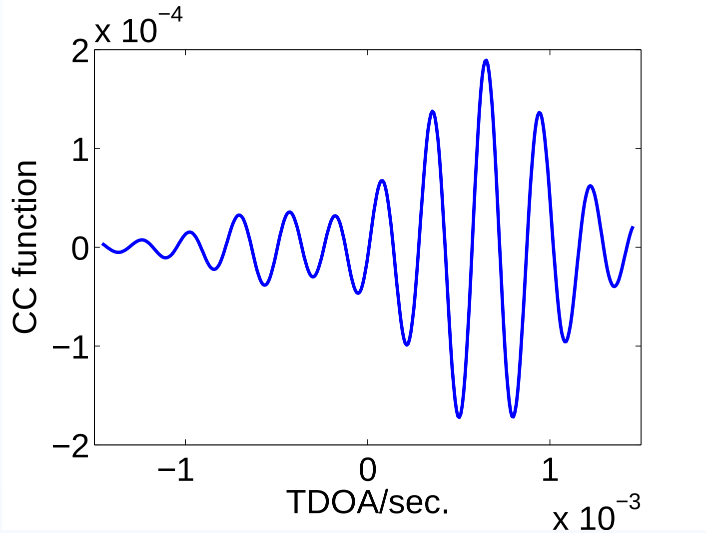
GCC Processors

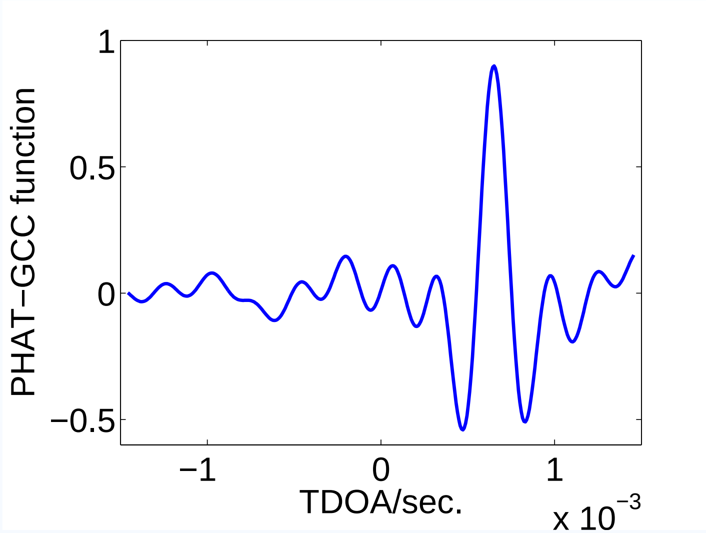
The PHAT-GCC approach can be written as:

$$
\begin{aligned}
r_{x_i\,x_j}[\ell] &= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} \Phi\left(e^{j\omega T_s}\right) P_{x_1x_2}\left(e^{j\omega T_s}\right) e^{j\ell\omega T}\, d\omega \\
&= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} \frac{1}{\left|P_{x_1x_2}\left(e^{j\omega T_s}\right)\right|} \left|P_{x_1x_2}\left(e^{j\omega T_s}\right)\right| e^{j\angle P_{x_1x_2}\left(e^{j\omega T_s}\right)} e^{j\ell\omega T}\, d\omega \\
&= \int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} e^{j\left(\ell\omega T + \angle P_{x_1x_2}\left(e^{j\omega T_s}\right)\right)}\, d\omega \\
&= \delta\left(\ell\, T_s + \angle P_{x_1x_2}\left(e^{j\omega T_s}\right)\right) \\
&= \delta(\ell\, T_s - T\left(\mathbf{m}_i,\, \mathbf{m}_j,\, \mathbf{x}_k\right))
\end{aligned}
$$

- In the absence of reverberation, the GCC-PHAT (GCC-PHAT) algorithm gives an impulse at a lag given by the TDOA divided by the sampling period.

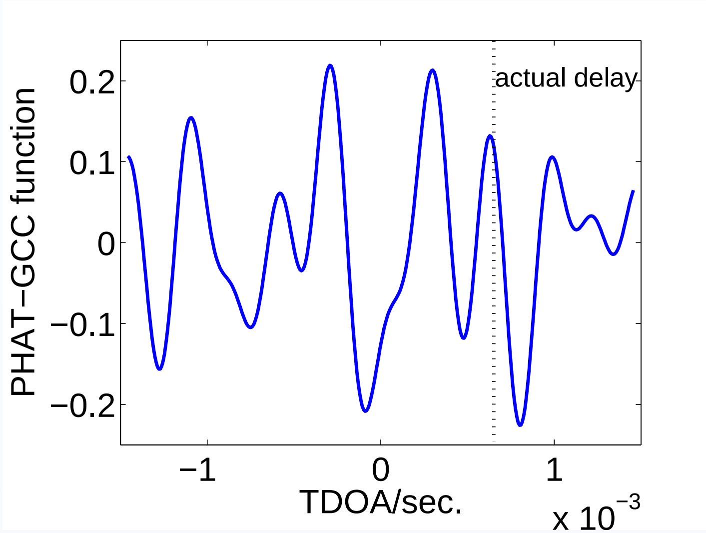
GCC Processors

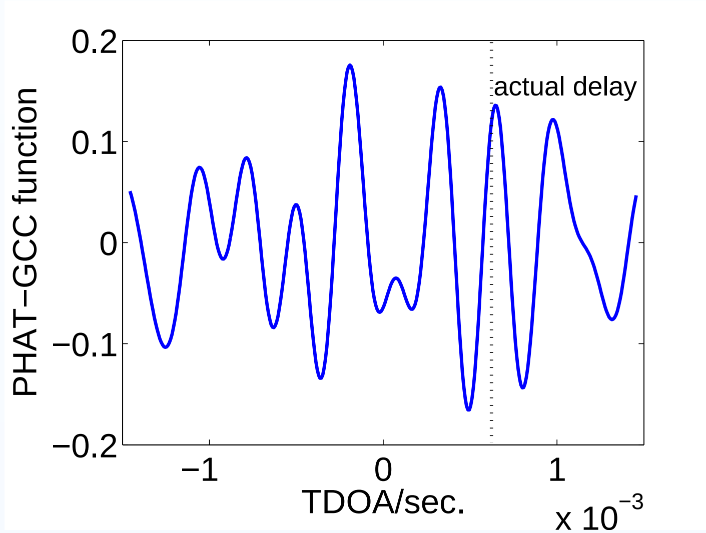




Normal cross-correlation and GCC-PHAT functions for a frame of speech.

GCC Processors





The effect of reverberation and noise on the GCC-PHAT can lead to poor TDOA estimates.

Adaptive Eigenvalue Decomposition

The AED algorithm actually amounts to a **blind channel identification** problem, which then seeks to identify the channel coefficients corresponding to the direct path elements.

Adaptive Eigenvalue Decomposition

The AED algorithm actually amounts to a **blind channel identification** problem, which then seeks to identify the channel coefficients corresponding to the direct path elements.

- Suppose that the acoustic impulse response (AIR) between source k and i is given by $h_{ik}[n]$ such that

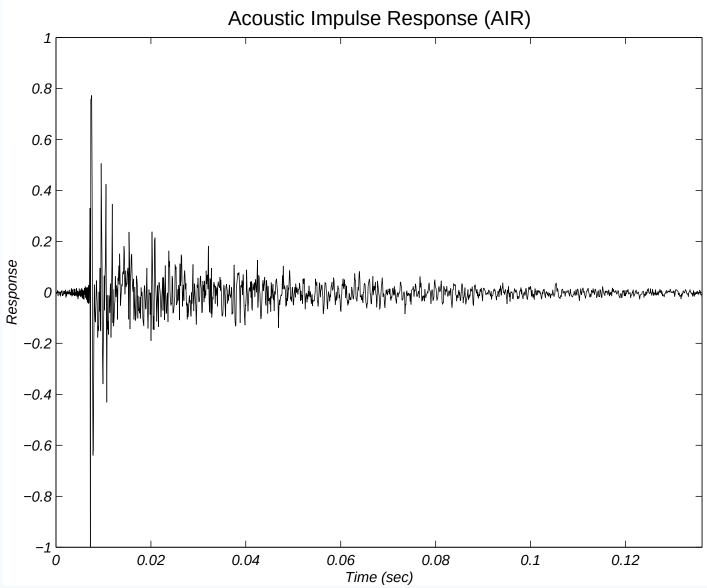
  $$x_{ik}[n] = \sum_{m=-\infty}^{\infty} h_{ik}[n-m]\, s_k[m] + b_{ik}[n]$$

  then the TDOA between microphones i and j is:

  $$\tau_{ijk} = \left\{ \arg\max_{\ell} |h_{ik}[\ell]| \right\} - \left\{ \arg\max_{\ell} |h_{jk}[\ell]| \right\}$$

  This assumes a minimum-phase system, but can easily be made robust to a non-minimum-phase system.

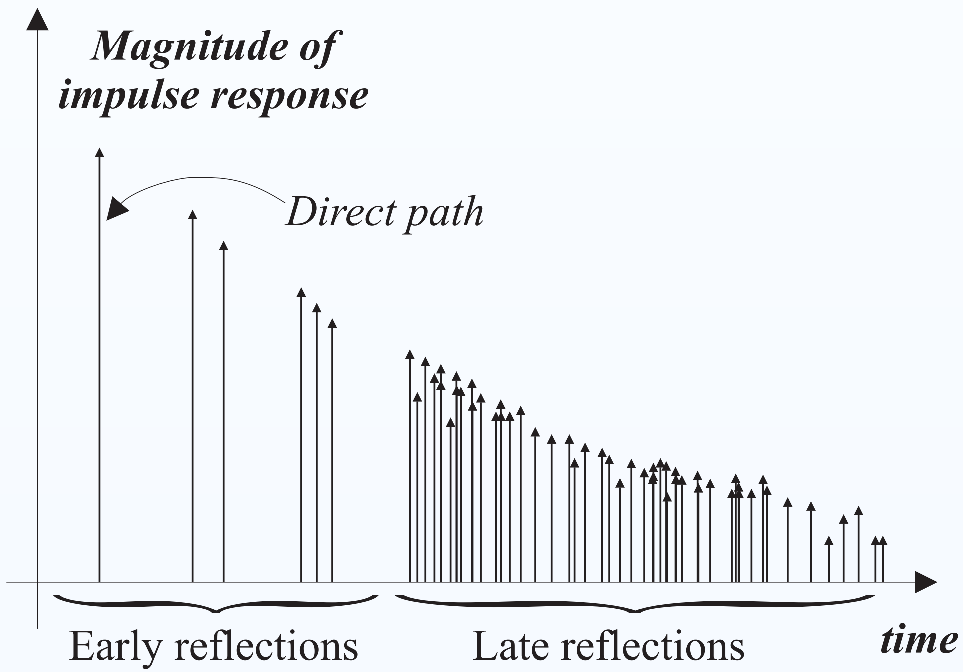
Adaptive Eigenvalue Decomposition



A typical room acoustic impulse response.

- Reverberation plays a major role in ASL and BSS.
- Consider reverberation as the sum total of all sound reflections arriving at a certain point in a room after room has been excited by impulse.

Adaptive Eigenvalue Decomposition

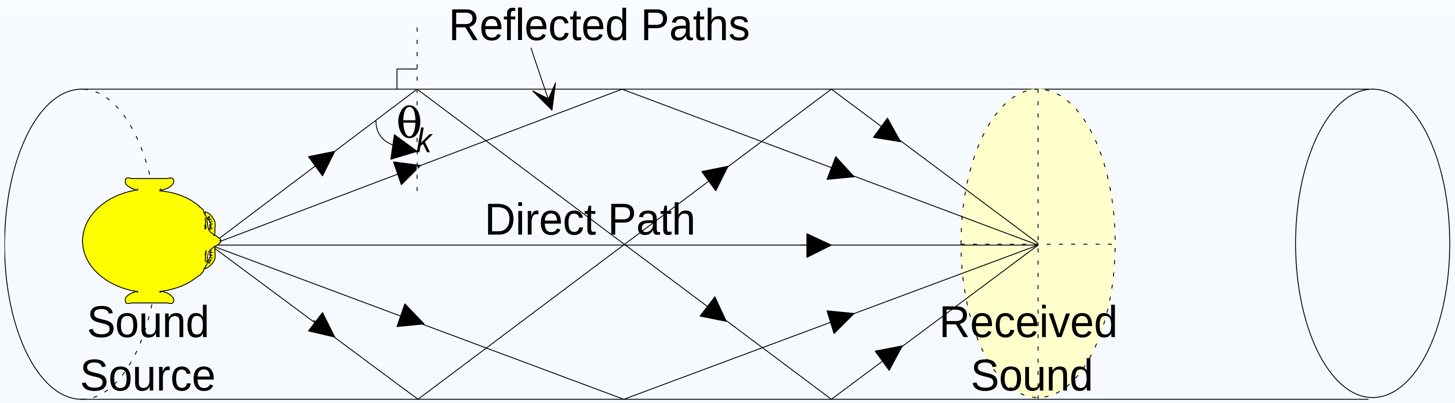


Early and late reflections in an AIR.

Trivia: Perceive early reflections to reinforce direct sound, and can help with speech intelligibility. It can be easier to hold a conversation in a closed room than outdoors

Adaptive Eigenvalue Decomposition

- Room transfer functions are often nonminimum-phase since there is more energy in the reverberant component of the RIR than in the component corresponding to direct path.



Demonstrating nonminimum-phase properties

- Therefore AED will need to consider multiple peaks in the estimated AIR.

Direct Localisation Methods

- Direct localisation methods have the advantage that the relationship between the measurement and thestate is linear.

- However, extracting the position measurement requires a multi-dimensional search over the state space and is usually computationally expensive.

Steered Response Power Function

The steered beamformer (SBF) or SRP function is a measure of correlation across *all pairs* of microphone signals for a set of relative delays that arise from a hypothesised source location.

The frequency domain **delay-and-sum beamformer** steered to a spatial position $\hat{\mathbf{x}}_k$ such that $\hat{\tau}_{pk} = |\hat{\mathbf{x}} - \mathbf{m}_p|$:

$$S(\hat{\mathbf{x}}) = \int_{\Omega} \left| \sum_{p=1}^{N} W_p\left(e^{j\omega T_s}\right) X_p\left(e^{j\omega T_s}\right) e^{j\omega\,\hat{\tau}_{pk}} \right|^2 d\omega$$

Steered Response Power Function

The SBF or SRP function is a measure of correlation across *all pairs* of microphone signals for a set of relative delays that arise from a hypothesised source location.

The frequency domain **delay-and-sum beamformer** steered to a spatial position $\hat{\mathbf{x}}_k$ such that $\hat{\tau}_{pk} = |\hat{\mathbf{x}} - \mathbf{m}_p|$:

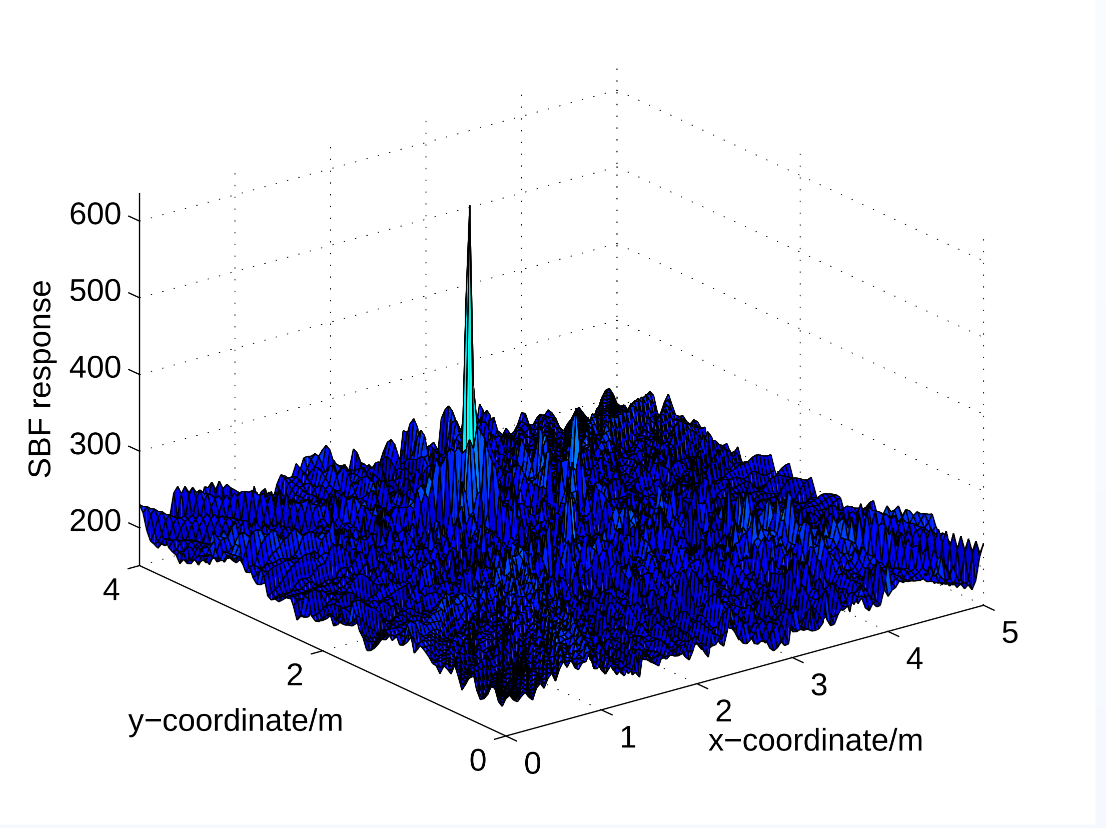
$$S(\hat{\mathbf{x}}) = \int_\Omega \left| \sum_{p=1}^{N} W_p\left(e^{j\omega T_s}\right) X_p\left(e^{j\omega T_s}\right) e^{j\omega\,\hat{\tau}_{pk}} \right|^2 d\omega$$

Taking expectations, $\Phi_{pq}\left(e^{j\omega T_s}\right) = W_p\left(e^{j\omega T_s}\right) W_q^*\left(e^{j\omega T_s}\right)$

$$\mathbb{E}\left[S(\hat{\mathbf{x}})\right] = \sum_{p=1}^{N}\sum_{q=1}^{N} \int_\Omega \Phi_{pq}\left(e^{j\omega T_s}\right) P_{x_p x_q}\left(e^{j\omega T_s}\right) e^{j\omega\hat{\tau}_{pqk}}\, d\omega$$

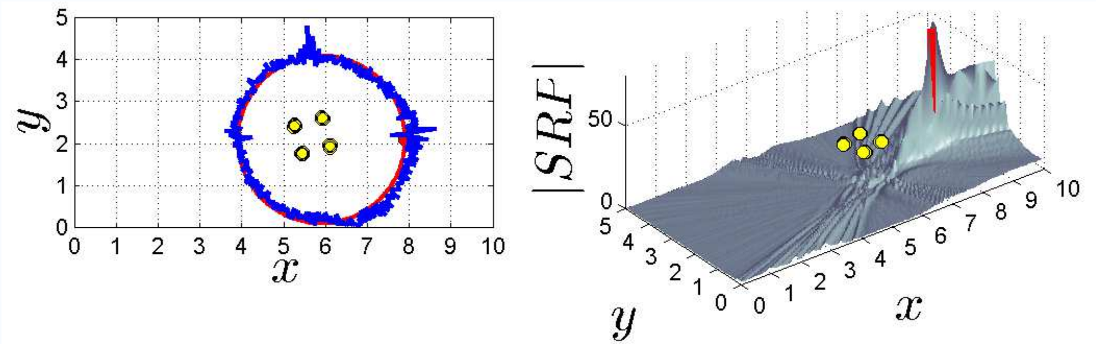
$$= \sum_{p=1}^{N}\sum_{q=1}^{N} r_{x_i\,x_j}[\hat{\tau}_{pqk}] \equiv \sum_{p=1}^{N}\sum_{q=1}^{N} r_{x_i\,x_j}\left[\frac{|\mathbf{x}_k - \mathbf{m}_i| - |\mathbf{x}_k - \mathbf{m}_j|}{c}\right]$$

Steered Response Power Function



SBF response from a frame of speech signal. The integration frequency range is 300 to 3500 Hz. The true source position is at $[2.0, 2.5]m$. The grid density is set to 40 mm.

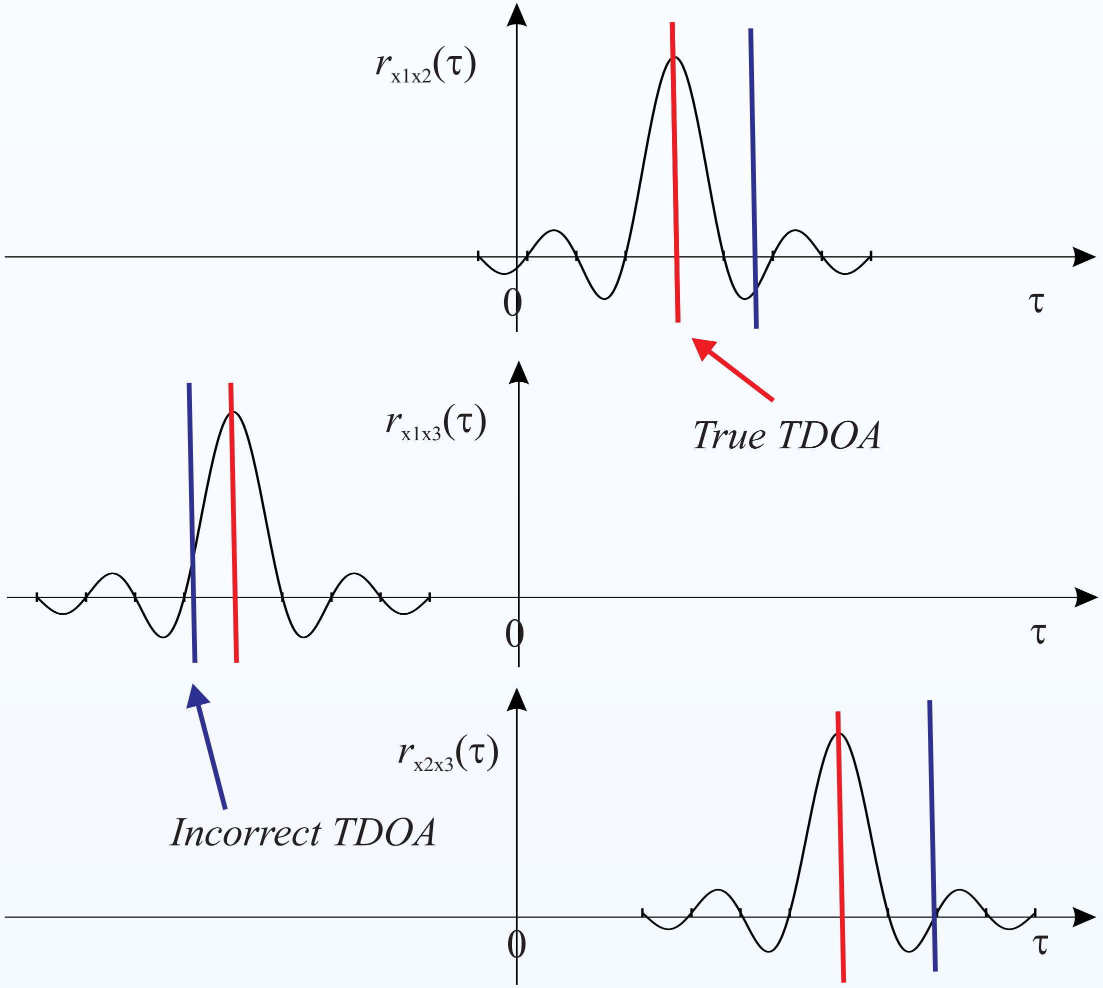
Steered Response Power Function



An example video showing the SBF changing as the source location moves.

- Show video!

Conceptual Intepretation



GCC-PHAT for different microphone pairs.

$$T\left(\mathbf{m}_i, \mathbf{m}_j, \hat{\mathbf{x}}_k\right) = \frac{\left|\hat{\mathbf{x}}_k - \mathbf{m}_i\right| - \left|\hat{\mathbf{x}}_k - \mathbf{m}_j\right|}{c}$$

Handout 2
Blind Source Separation

DUET Algorithm

The degenerate unmixing estimation technique (DUET) algorithm is an approach to BSS that ties in neatly to ASL. Under certain assumptions and circumstances, it is possible to separate more than two sources using only two microphones.

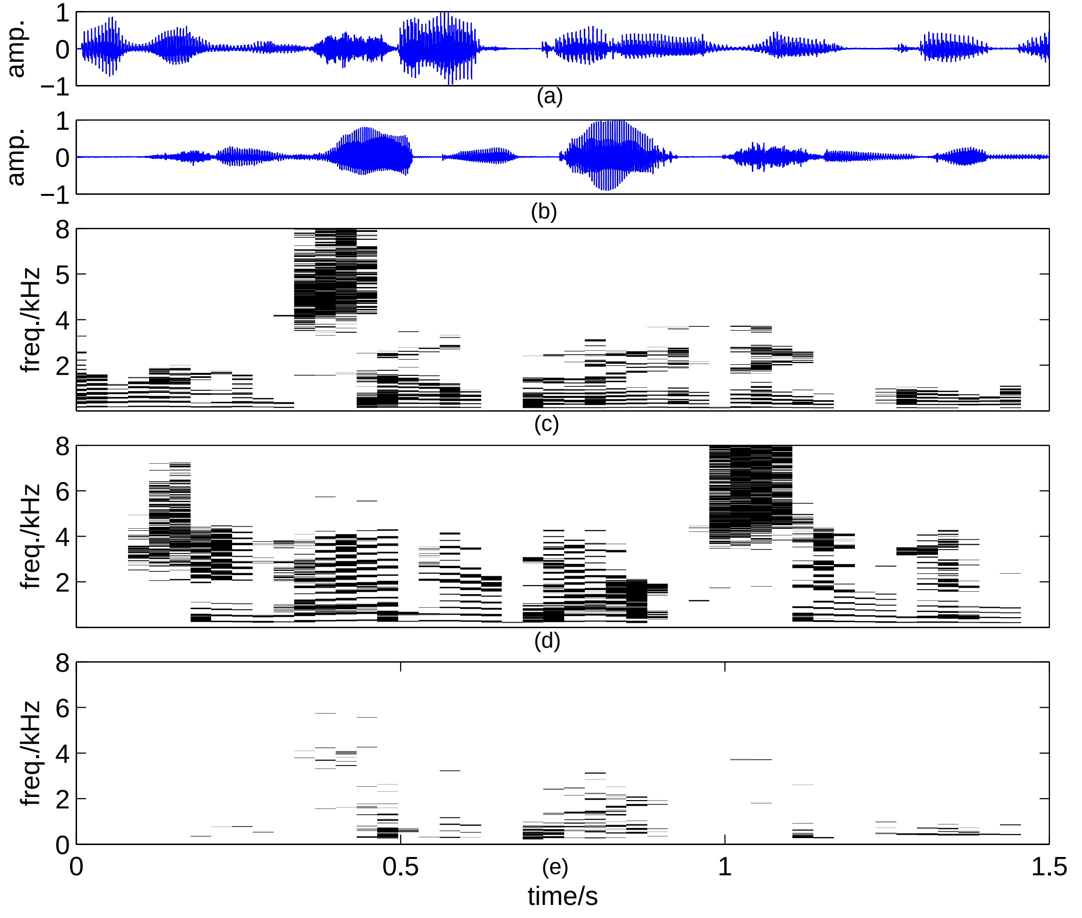
DUET Algorithm

The DUET algorithm is an approach to BSS that ties in neatly to ASL. Under certain assumptions and circumstances, it is possible to separate more than two sources using only two microphones.

- DUET is based on the assumption that for a set of signals $x_k[t]$, their time-frequency representations (TFRs) are predominately non-overlapping. This condition is referred to as W-disjoint orthogonality (WDO):

$$S_p\left(\omega,\, t\right)\, S_q\left(\omega,\, t\right) = 0\, \forall p \neq q,\, \forall t,\, \omega$$

DUET Algorithm



W-disjoint orthogonality of two speech signals. Original speech signal (a) $s_1[t]$ and (b) $s_2[t]$; corresponding STFTs (c) $|S_1(\omega, t)|$ and (d) $|S_2(\omega, t)|$; (e) product $|S_1(\omega, t) S_2(\omega, t)|$.

DUET Algorithm

Consider taking a particular time-frequency (TF)-bin, (ω, t), where source p is known to be active. The two received signals in *that TF-bin* can be written as:

$$X_{ip}(\omega, t) = \alpha_{ip} \, e^{-j\omega \, \tau_{ip}} \, S_p(\omega, t) + B_i(\omega, t)$$

$$X_{jp}(\omega, t) = \alpha_{jp} \, e^{-j\omega \, \tau_{jp}} \, S_p(\omega, t) + B_j(\omega, t)$$

Source Localisation

DUET Algorithm

Consider taking a particular TF-bin, (ω, t), where source p is known to be active. The two received signals in *that TF-bin* can be written as:

$$X_{ip}(\omega, t) = \alpha_{ip} e^{-j\omega \tau_{ip}} S_p(\omega, t) + B_i(\omega, t)$$

$$X_{jp}(\omega, t) = \alpha_{jp} e^{-j\omega \tau_{jp}} S_p(\omega, t) + B_j(\omega, t)$$

Taking the ratio and ignoring the noise terms gives:

$$H_{ikp}(\omega, t) \triangleq \frac{X_{ip}(\omega, t)}{X_{jp}(\omega, t)} = \frac{\alpha_{ip}}{\alpha_{jp}} e^{-j\omega\tau_{ijp}}$$

DUET Algorithm

Consider taking a particular TF-bin, (ω, t), where source p is known to be active. The two received signals in *that TF-bin* can be written as:

$$X_{ip}(\omega, t) = \alpha_{ip}\, e^{-j\omega\,\tau_{ip}}\, S_p(\omega, t) + B_i(\omega, t)$$

$$X_{jp}(\omega, t) = \alpha_{jp}\, e^{-j\omega\,\tau_{jp}}\, S_p(\omega, t) + B_j(\omega, t)$$

Taking the ratio and ignoring the noise terms gives:

$$H_{ikp}(\omega, t) \triangleq \frac{X_{ip}(\omega, t)}{X_{jp}(\omega, t)} = \frac{\alpha_{ip}}{\alpha_{jp}}\, e^{-j\omega\tau_{ijp}}$$

Hence,

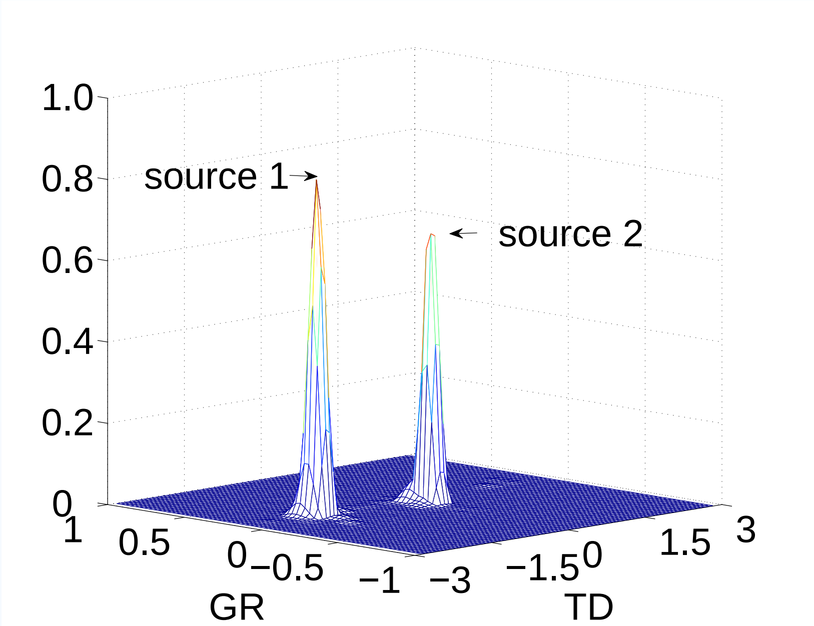
$$\tau_{ijp} = -\frac{1}{\omega} \arg H_{ikp}(\omega, t), \qquad \text{and} \qquad \frac{\alpha_{ip}}{\alpha_{jp}} = |H_{ikp}(\omega, t)|$$

Source Localisation

Blind Source Separation

- DUET Algorithm
- Effect of Reverberation and Noise
- Estimating multiple targets
- Further Topics

DUET Algorithm



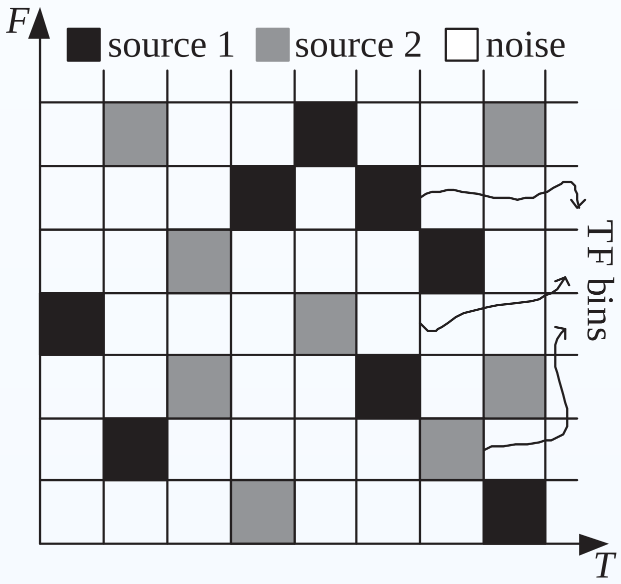


Illustration of the underlying idea in DUET.

DUET Algorithm

This leads to the essentials of the DUET method which are:

1. Construct the TF representation of both mixtures.

2. Take the ratio of the two mixtures and extract local mixing parameter estimates.

DUET Algorithm

This leads to the essentials of the DUET method which are:

1. Construct the TF representation of both mixtures.

2. Take the ratio of the two mixtures and extract local mixing parameter estimates.

3. Combine the set of local mixing parameter estimates into N pairings corresponding to the true mixing parameter pairings.

4. Generate one binary mask for each determined mixing parameter pair corresponding to the TF-bins which yield that particular mixing parameter pair.

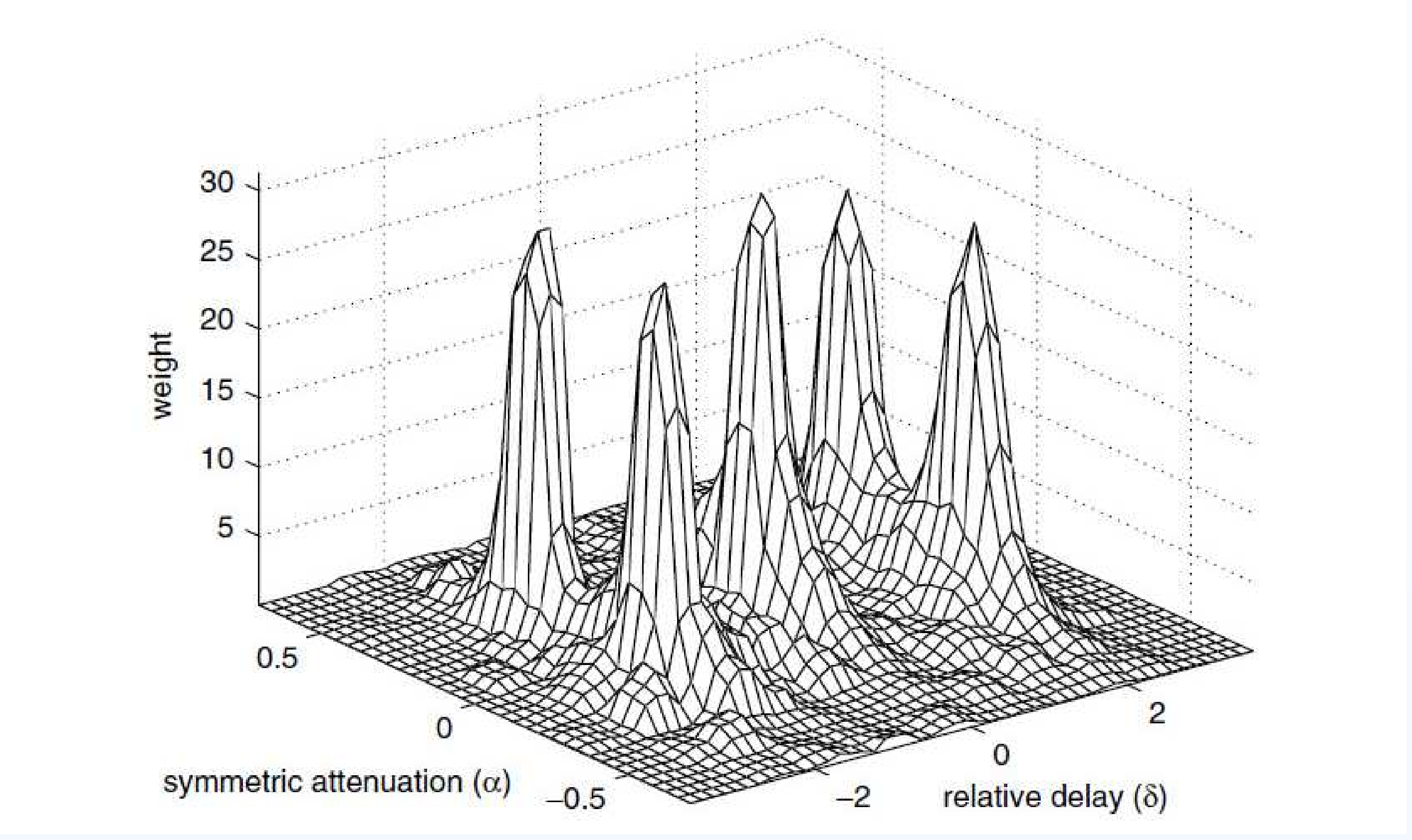
DUET Algorithm

This leads to the essentials of the DUET method which are:

1. Construct the TF representation of both mixtures.
2. Take the ratio of the two mixtures and extract local mixing parameter estimates.
3. Combine the set of local mixing parameter estimates into N pairings corresponding to the true mixing parameter pairings.
4. Generate one binary mask for each determined mixing parameter pair corresponding to the TF-bins which yield that particular mixing parameter pair.
5. Demix the sources by multiplying each mask with one of the mixtures.
6. Return each demixed TFR to the time domain.

DUET Algorithm

This leads to the essentials of the DUET method which are:



DUET for multiple sources.

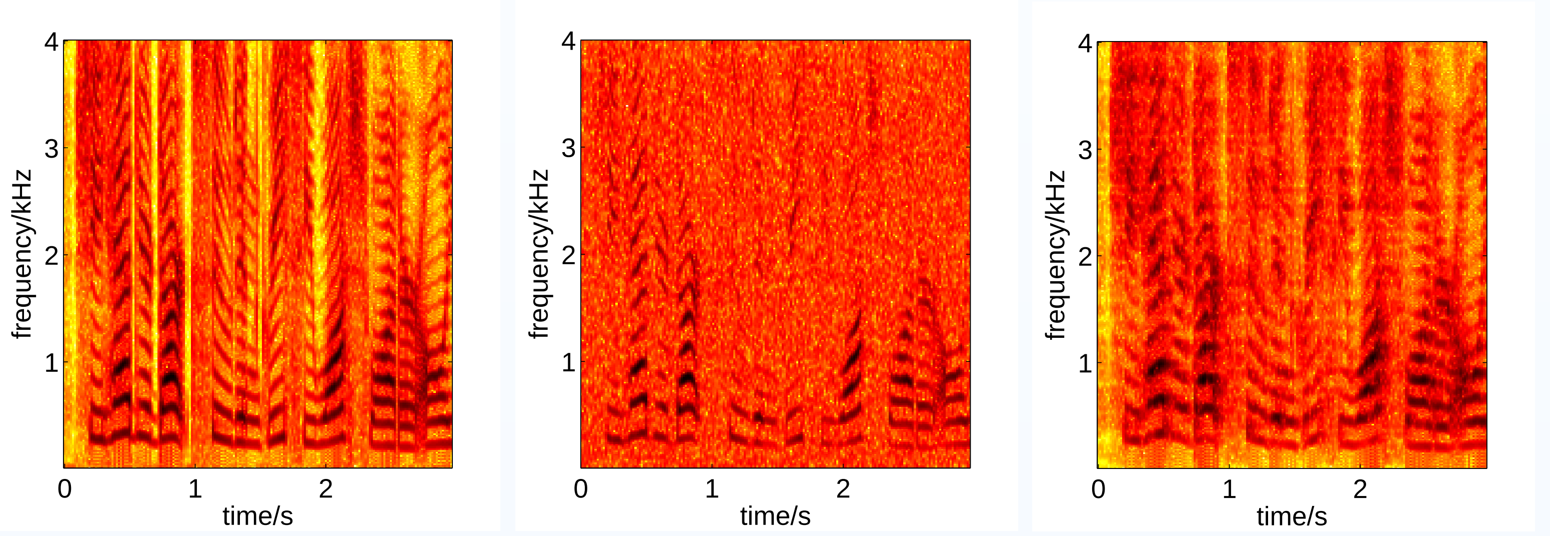
DUET Algorithm

Source Localisation

Blind Source Separation

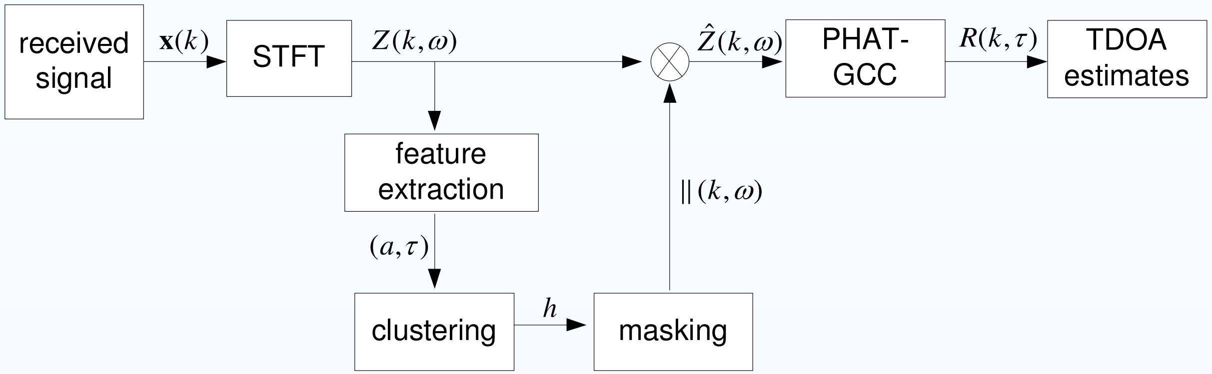
- DUET Algorithm
- Effect of Reverberation and Noise
- Estimating multiple targets
- Further Topics

Effect of Reverberation and Noise

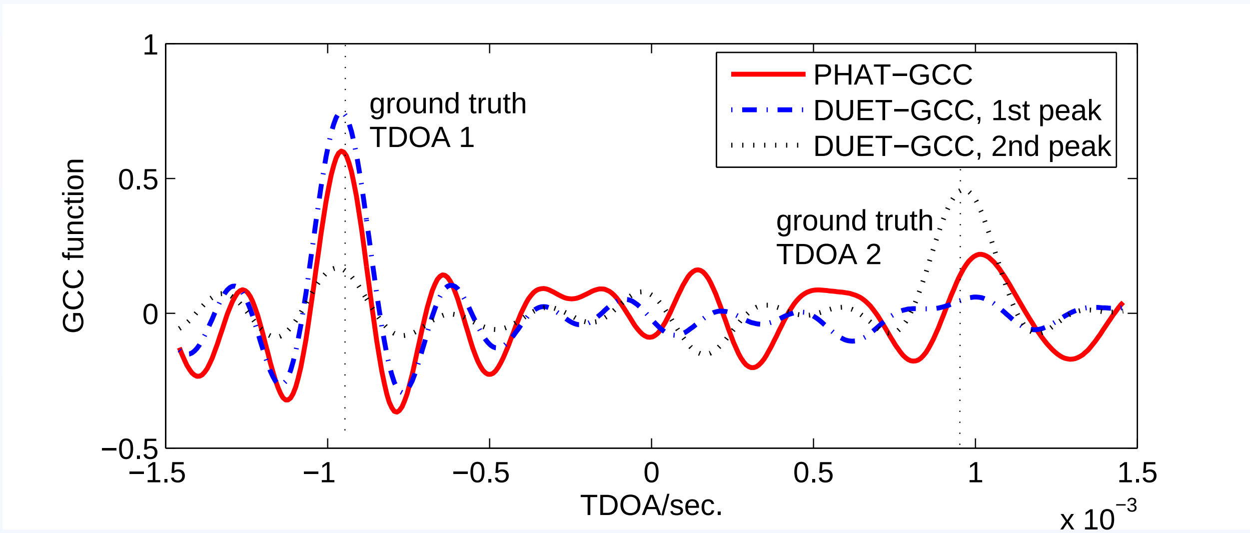


The TFR is very clear in the anechoic environment but smeared around by the reverberation and noise.

Estimating multiple targets



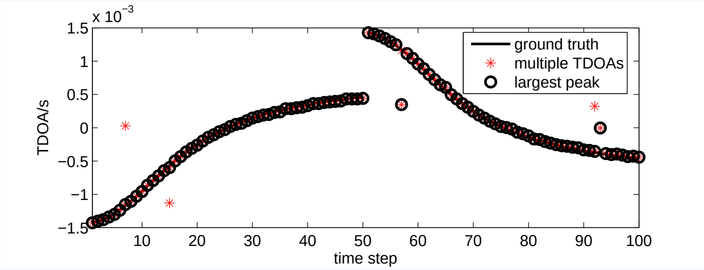
Flow diagram of the DUET-GCC approach. Basically, the speech mixtures are separated by using the DUET in the TF domain, and the PHAT-GCC is then employed for the spectrogram of each source to estimate the TDOAs.

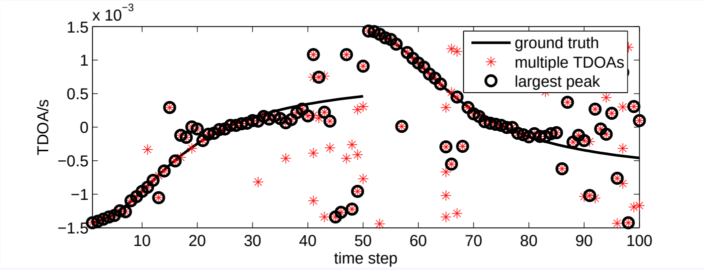


GCC function from DUET approach and traditional PHAT weighting. Two sources are located at (1.4, 1.2)m and (1.4,

Further Topics

- Reduction in complexity of calculating SRP. This includes stochastic region contraction (SRC) and hierarchical searches.
- Multiple-target tracking (see Daniel Clark's Notes)
- Simultaneous (self-)localisation and tracking; estimating sensor and target positions from a moving source.





Acoustic source tracking and localisation.

Source Localisation

Blind Source Separation

- DUET Algorithm
- Effect of Reverberation and Noise
- Estimating multiple targets
- Further Topics

Further Topics

- Joint ASL and BSS.
- Explicit signal and channel modelling! (None of the material so forth cares whether the signal is speech or music!)
- Application areas such as gunshot localisation; other sensor modalities; diarisation.