Introduction to distributed speech enhancement algorithms for ad hoc microphone arrays and wireless acoustic sensor networks

Part II: DANSE-based distributed speech enhancement in WASNs

Sharon Gannot\(^1\) and Alexander Bertrand\(^2\)

\(^1\)Faculty of Engineering, Bar-Ilan University, Israel

\(^2\)KU Leuven, E.E. Department ESAT-STADIUS, Belgium

EUSIPCO 2013, Marrakech, Morocco
Outline

1. Introduction and motivation
2. The DANSE algorithm in fully-connected WASNs
3. DANSE in WASNs with a tree topology (T-DANSE)
4. LCMV-based DANSE (LC-DANSE)
5. Bibliography
Ad hoc microphone arrays

- No tedious calibration
- Improved spatial resolution and sound field sampling.
- High probability to find microphones close to a relevant sound source.
- Possibility to put (arrays of) microphones at strategic places
Wireless acoustic sensor networks (WASNs)

- Wired ad hoc arrays:
  - Tedious deployment
  - Unaesthetic
  - Not flexible (e.g., adding/removing/repositioning microphones)
  - Not suitable for wearable or mobile applications (e.g., hearing aids)
- Aim for *wireless* ad hoc microphone arrays.
- A.k.a. *wireless acoustic sensor network (WASN)*
  (due to similarities with wireless sensor networks)
Wireless acoustic sensor networks (WASNs)

Possible applications:

- Cooperative hearing devices (e.g., binaural hearing aids)
- Hearing devices supported by external microphones or other audio devices
- Domotics, smart homes and ambient intelligence
- Surveillance
- ...
Wireless acoustic sensor networks (WASNs)

**Challenges**

- Wireless link delay (e.g., in case of real-time constraints)
- Different sampling clocks (see also Part III)
- The ‘data deluge’ (see next slide)
WASNs and the data deluge

The ‘data deluge’ [Baraniuk, 2011]

- WASNs generate a **massive amount of data**:
  - Requires a large **communication bandwidth**
  - Sensor nodes consume a large amount of **transmission energy**
  - Requires high **computing power** at the receiver end (fusion center)

- =big problem, in particular when battery-powered
  (even in small-scale WASNs such as binaural hearing aids)
Tackle the data deluge by physically shifting the signal processing to the microphone nodes themselves

Goals:
- Minimize data exchange
- Distribute computational burden over all nodes
- Let nodes cooperate in signal processing task(s)

Algorithm design = challenging (e.g., no access to full correlation matrix)
Distributed signal processing

The field of distributed signal processing:

- Mainly driven by the concept of wireless sensor networks
- Theory and methods often build upon results from other fields, e.g.,
  - Parallel and distributed computing for multi-core processors
  - Modelling and control of multi-agent systems
  - Game theory
  - Graph theory
- Two fundamentally different approaches:
  1. Distributed parameter estimation techniques (DPE)
     (e.g., diffusion [Sayed et al., 2013], consensus [Olfati-Saber et al., 2007], gossip [Shah, 2009], ...)
  2. Distributed signal estimation techniques (DSE)
     (e.g., DANSE-family, distributed/cooperative beamforming, distributed/remote source coding, ...)

S. Gannot (BIU) and A. Bertrand (KUL)
Distributed parameter estimation (DPE)

General script:

1. Extract initial parameter vector estimate from sensor observations
2. Repeat until convergence (or other stop criterion):
   - Share intermediate estimate with neighbors
   - Refine intermediate estimate using estimates from neighbors

Note: target parameter vector is fixed over iterations, or varies only slowly
DPE for speech enhancement in WASNs

Collect $L$ microphone signal samples at each node and iterate on $L$-dimensional vector until the estimate converges. Then collect $L$ new samples, etc.

- 😊 DPE techniques usually have no network topology constraints
- 😞 Large communication cost: re-estimate and re-transmit same $L$ samples many times (freeze time index until convergence)
- 😞 Communication cost depends on convergence speed (and hence also on network size)
- 😞 Not time-recursive: full reset between blocks

See, e.g., [Zeng and Hendriks, 2012, Heusdens et al., 2012]
Distributed signal estimation (DSE)

- Avoid iterations over the signal sample estimates themselves
  ⇒ In-network data flow and iterative process are uncoupled
- Instead: time-recursive iterative refinement of in-network fusion rules
- Assumption: spatial coherence of sensor signals is fixed over iterations (or varies slowly)
DSE for speech enhancement in WASNs?

- No iterative refinement of sample estimates:
  - 😊 Each block of samples is transmitted only once
  - 😊 Fixed per-node communication cost, independent of convergence speed/network size

- 😊 Price to pay: specific order in data flow generally requires topology constraints (star, tree, fully-connected,...)

Introduction and motivation

The DANSE algorithm in fully-connected WASNs

DANSE in WASNs with a tree topology (T-DANSE)

LCMV-based DANSE (LC-DANSE)

Bibliography
Multi-channel Wiener filtering [Doclo and Moonen, 2002]

- Goal: estimate speech component at reference microphone
- Optimal filter-and-sum operation based on input statistics

\[
\begin{align*}
\min_w & \ E \left\{ |d_{\text{ref}} - w^H y|^2 \right\} \\
& \\
& w(\omega) = R_{yy}(\omega)^{-1} R_{dd}(\omega) e_{\text{ref}} \\
& R_{yy}(\omega) = E\{y(\omega)y(\omega)^H\} \\
& \text{Voice activity detection (VAD)} \\
& R_{dd}(\omega) = R_{yy}(\omega) - R_{nn}(\omega)
\end{align*}
\]
Preliminary case study: binaural hearing aids

- Two hearing aids (HAs) with wireless link (=2-node WASN)
- Goal: compute MWF including extra signal(s) from other HA
- Each HA uses a local microphone as reference to preserve binaural cues of target speaker
Preliminary case study: binaural hearing aids

**Problem statement** [Doclo et al., 2009, Srinivasan and Den Brinker, 2009]

- Wireless link only allows exchange of 1 signal (in duplex)
- Which signal should be transmitted?
Preliminary case study: binaural hearing aids

Result from [Doclo et al., 2009]

- Copy part of the local MWF coefficients and use it as fusion rule to generate transmit signal (=optimal for single target speaker)
- Iterative computation (details omitted, see later)
- Will extend this result to more general WASN scenarios in this tutorial

PS: similar result exists for binaural MVDR BF [Markovich-Golan et al., 2010]
DANSE in fully-connected WASNs

Assumptions:

- Multiple mics per node (array or hierarchical architecture)
- Network is **fully connected** (=easiest case, will be extended to multi-hop topologies later)
- Each node is a data sink, and requires a **node-specific** estimate of the target source(s) to preserve spatial cues

⇒ Distributed adaptive node-specific signal estimation (DANSE)
Notation

- WASN with $N$ nodes $\{1, \ldots, N\} = \mathcal{J}$
- Node $k \in \mathcal{J}$ collects an $M_k$-channel microphone signal $y_k(\omega, t)$ (represented in short-time Fourier transform (STFT) domain)
- Will often omit $(\omega, t)$ in the sequel for conciseness, keep in mind that all operations are performed in STFT domain.
- Additive noise:
  \[ y_k = d_k + n_k \]
  $n_k$ is noise and $d_k$ is the desired speech signal.
- Stacked vector $y = [y_1^T \ldots y_N^T]^T$ defines $M$-channel signal with $M = \sum_{k \in \mathcal{J}} M_k$.
- Similar for $d$ and $n$, i.e., $y = d + n$.
- $y_{km}$ denotes the $m$-th microphone of node $k$, and $e_{km} = [0 \ldots 0\ 1\ 0 \ldots 0]^T$ is a selection vector such that $y_{km} = e_{km}^T y$. 
Centralized per-node MWFs

- At each node: choose 1st mic as reference microphone (w.l.o.g.)
- Assume all nodes have access to all signals: node $k \in \mathcal{J}$ computes

$$\hat{d}_{k1} = \hat{w}_k^H y$$

with $H$ denoting conjugate transpose and $\hat{w}_k$ is node $k$’s MWF

$$\hat{w}_k = \arg \min_{w_k} E\{|d_{k1} - w_k^H y|^2\} = R_{yy}^{-1} R_{dd} e_{k1}$$

where $R_{yy} = E\{yy^H\}$ and $R_{dd} = E\{dd^H\} = R_{yy} - R_{nn}$ (VAD)

PS: will only focus on MWF, but can easily be extended to SDW-MWF.
DANSE signal exchange
DANSE signal exchange

- Node $k$ broadcasts the fused signal $z_k$ to the other nodes:

$$z_i^k = f_i^k H_k y_k$$

where $f_i^k$ is an $M_k$-dimensional fusion vector and $i$ is an iteration index.

- **Data compression**: $M_k$-channel signal $y_k \rightarrow$ single-channel signal $z_i^k$

- Between iteration $i$ and $i+1$, node $k$ collects samples of

$$\tilde{y}_k^i = \begin{bmatrix} y_k^i \\ z_i^{i-k} \end{bmatrix} = \tilde{d}_k^i + \tilde{n}_k^i$$

with $z_i^{i-k} = [z_1^i \ldots z_{k-1}^i z_{k+1}^i \ldots z_N^i]^T$. 
DANSE signal exchange
DANSE per-node MWFs

- Node $k$ will compute local MWF $\tilde{v}^i_k$ that minimizes

$$\min_{\tilde{v}_k} E \{ |d_{k1} - \tilde{v}_k^H \tilde{y}_k|^2 \}.$$ 

- This yields

$$\tilde{v}^i_k = \left( R^i_{\tilde{y}_k \tilde{y}_k} \right)^{-1} R^i_{\tilde{d}_k \tilde{d}_k} e_1$$

where $e_1 = [1 \ 0 \ \ldots \ 0]$, $R^i_{\tilde{y}_k \tilde{y}_k} = E \{ \tilde{y}_k \tilde{y}_k^H \}$, $R^i_{\tilde{d}_k \tilde{d}_k} = E \{ \tilde{d}_k \tilde{d}_k^H \}$.

- With VAD: $R^i_{\tilde{d}_k \tilde{d}_k} = R^i_{\tilde{y}_k \tilde{y}_k} - R^i_{\tilde{n}_k \tilde{n}_k}$ (PS: nodes can share VAD info)

- Between iterations $i$ and $i + 1$, estimated speech signal at node $k$:

$$\tilde{d}^i_{k1} = \tilde{v}_k^i \tilde{y}_k^i$$
Equivalent network-wide filter?

⇒ how does equivalent network-wide filter $w^i_k$ look like?

$$d^i_{k1} = \tilde{v}_k^i H \tilde{y}_k = w^i_k H y \Rightarrow w^i_k$$
Equivalent network-wide filter?

Local MWF ↔ network-wide filter

\[
\begin{align*}
\mathbf{w}_1^i &= \begin{bmatrix} w_{11}^i \\ g_{12}^i f_2^i \\ g_{13}^i f_3^i \end{bmatrix}, & \mathbf{w}_2^i &= \begin{bmatrix} g_{21}^i f_1^i \\ \mathbf{w}_2^i \\ g_{23}^i f_3^i \end{bmatrix}, & \mathbf{w}_3^i &= \begin{bmatrix} g_{31}^i f_1^i \\ g_{32}^i f_2^i \\ \mathbf{w}_3^i \end{bmatrix} \\
\end{align*}
\]

\(g_{kq}^i\) is the coefficient that node \(k\) applies to the \(z_q^i\) signal from node \(q\).
DANSE parametrization

**Choice of \( f_k^i \)'s**

DANSE sets \( f_k^i = w_{kk}^i \), i.e., \( w_{kk}^i \) serves both as compressor and estimator

\[
\mathbf{w}_k^i = \begin{bmatrix} g_{k1}^i w_{11}^i \\ \vdots \\ g_{kN}^i w_{NN}^i \end{bmatrix} \quad (g_{kk}^i = 1, \text{by definition})
\]

PS: chicken-and-egg problem: need samples of \( z_k \) signals to compute local MWFs, but need MWFs to compute samples of \( z_k \)'s
The DANSE algorithm in fully-connected WASNs

DANSE parametrization

Example of DANSE parametrization (3-node case)

\[ w_1^i = \begin{bmatrix} w_{11}^i \\ g_{12}^i w_{22}^i \\ g_{13}^i w_{33}^i \end{bmatrix}, \quad w_2^i = \begin{bmatrix} g_{21}^i w_{11}^i \\ w_{22}^i \\ g_{23}^i w_{33}^i \end{bmatrix}, \quad w_3^i = \begin{bmatrix} g_{31}^i w_{11}^i \\ g_{32}^i w_{22}^i \\ w_{33}^i \end{bmatrix} \]

PS: similar to \( z_{-k}^i \), introduce notation

\[ g_{k,-k}^i = [g_{k1}^i \ldots g_{k,k-1}^i g_{k,k+1}^i \ldots g_{kN}^i]^T. \]
Algorithm description (for fixed frequency index $\omega$)

**DANSE$_1$ algorithm** [Bertrand and Moonen, 2010a]

1. Initialize: $i \leftarrow 0$, $u \leftarrow 1$
   Initialize $w_{kk}^0$ and $g_{k,-k}^0$ with random vectors, $\forall k \in \mathcal{J}$

2. Each node $k \in \mathcal{J}$ performs the following operation cycle:
   - Collect $B$ new sensor observations $y_k(\omega, iB + n)$, $n = 0 \ldots B - 1$.
   - Compress these $M_k$-dimensional observations to
     $$z_k^i(\omega, iB + n) = w_{kk}^i y_k(\omega, iB + n), \quad n = 0 \ldots B - 1.$$  
   - Broadcast $B$ samples of $z_k^i$ to other nodes.
   - Collect $B$ samples of $z_{-k}^i$ from other nodes.
   - Compute new estimator parameters $w_{kk}^{i+1}$ and $g_{k,-k}^{i+1}$ (see next slide).
   - Compute $B$ samples of speech estimate (for $n = 0 \ldots B - 1$)
     $$\bar{d}_{k1}^i(\omega, iB + n) = w_{kk}^{i+1} y_k(\omega, iB + n) + g_{k,-k}^{i+1} z_{-k}^i(\omega, iB + n).$$

3. Set $i \leftarrow i + 1$, $u \leftarrow (u \mod N) + 1$, and return to step 2
Algorithm description (continued)

**DANSE$_1$ algorithm: computation of $w^{i+1}_{kk}$ and $g^{i+1}_{k,-k}$**

- Node $u$ re-estimates $R_{\tilde{y}_u\tilde{y}_u}^i$ and $R_{\tilde{d}_u\tilde{d}_u}^i$, based on the collected samples in $z_{-u}^i(\omega, iB + n)$ and $y_u(\omega, iB + n)$, $n = 0 \ldots B - 1$.
- $\forall k \in \mathcal{J}$, update:

\[
\begin{bmatrix}
w^{i+1}_{kk} \\
g^{i+1}_{k,-k}
\end{bmatrix} = \begin{cases} 
\left( R_{\tilde{y}_k\tilde{y}_k}^i \right)^{-1} R_{\tilde{d}_k\tilde{d}_k}^i e_1 & \text{if } k = u \\
\begin{bmatrix}
w_i^i \\
g_i^i_{k,-k}
\end{bmatrix} & \text{if } k \neq u
\end{cases}
\]

**Note:**
- **Sequential round-robin** updating
- $B$ should be large (filters are typically frozen for 1-3 sec)
- Several DANSE algorithms in parallel (one for each frequency bin $\omega$)
Convergence and optimality of DANSE?

**Convergence**

Does DANSE converge to an equilibrium?

⇒ Does \( \lim_{i \to \infty} w^i_k \) exist, \( \forall k \in \mathcal{J} \)?

**Optimality**

If DANSE converges to an equilibrium setting, does it have the same estimation performance as the centralized MWF?

⇒ Is \( \lim_{i \to \infty} w^i_k = \hat{w}_k, \forall k \in \mathcal{J} \)?
1st result

First question: are $\hat{w}_k$, $\forall k \in \mathcal{J}$, in the solution space of DANSE?

**Theorem**

*In case of a single desired speech source, and if all nodes in $\mathcal{J}$ can 'hear' this source, then the solution space defined by the parametrization of DANSE contains the optimal (centralized) MWFs $\hat{w}_k$, $\forall k \in \mathcal{J}$.*

Proof outline:

- **Single desired speech source:**

  $$\forall k \in \mathcal{J} : d_k(\omega, t) = a_k(\omega)s(\omega, t)$$

  where $s(\omega, t)$ contains desired speech source and steering vector $a_k(\omega)$ contains $M_k$ transfer functions from source to $M_k$ microphones.

- Let $a = [a_1^T \ldots a_N^T]^T$, then $d(\omega, t) = a(\omega)s(\omega, t)$. 
Proof (continued)

- Centralized MWF at node $k$:
\[
\hat{w}_k = R_{yy}^{-1} R_{dd} e_{k1} \\
= R_{yy}^{-1} a E\{|s|^2\} a^H e_{k1} \\
= R_{yy}^{-1} a \cdot a^*_{k1} E\{|s|^2\}
\]

- It follows that $\forall k, q \in J$:
\[
\hat{w}_k = \alpha_{kq} \hat{w}_q
\]

with $\alpha_{kq} = \frac{a^*_{k1}}{a^*_{q1}}$.

- In DANSE: set $g_{kq}^i = \alpha_{kq}$ and $w_{kk}^i = \hat{w}_{kk}$, $\forall k, q \in J$
\[
\forall k \in J : \ w_k^i = \begin{bmatrix} g_{k1}^i w_{11}^i \\ \vdots \\ g_{kN}^i w_{NN}^i \end{bmatrix} = \begin{bmatrix} \alpha_{k1} \hat{w}_{11} \\ \vdots \\ \alpha_{kN} \hat{w}_{NN} \end{bmatrix} = \hat{w}_k.
\]
2nd result

**Theorem (Convergence and optimality of DANSE [Bertrand and Moonen, 2010a])**

*In case of a single desired speech source, and if $a_{k_1} \neq 0$, $\forall k \in \mathcal{J}$, then*

$$
\lim_{i \to \infty} \mathbf{w}_i^k = \mathbf{\hat{w}}_k, \quad \forall k \in \mathcal{J}.
$$

*In other words: each node obtains the speech estimate of its corresponding centralized MWF, as if it had access to all the microphone signals.*

(proof omitted)
DANSE vs. Centralized MWF

Advantages of DANSE

- Reduced communication bandwidth and reduced transmission energy
- All nodes contribute/cooperate in the processing
  ⇒ Small *per-node* processing power
- Inherent dimensionality reduction
  ⇒ Many small problems vs. single large problem
  ⇒ Often smaller overall processing power (due to $O(M^2)$ or $O(M^3)$ complexity)

Disadvantages of DANSE

- Reduced tracking performance due to iterative nature (per-node tracking can be improved [Szurley et al., 2013])
- Ripple of errors to other nodes (will be addressed later)
Multiple target speakers

What if desired signal $d_{k_1}$ is a mixture of $Q$ desired speech sources?

$\Rightarrow \hat{w}_k = \alpha_{kq}\hat{w}_q$ does not hold anymore (see next slide)

$\Rightarrow \hat{w}_k$ not in solution space of DANSE 😞
Multiple target speakers

- Centralized MWF at node $k$ (for $Q = 2$):

$$\hat{w}_k = R_{yy}^{-1} R_{dd} e_k$$

$$= R_{yy}^{-1} [a_1 \ a_2] \left[ \begin{array}{cc} E\{|s_1|^2\} & 0 \\ 0 & E\{|s_2|^2\} \end{array} \right] \left[ \begin{array}{c} a_1^H \\ a_2^H \end{array} \right] e_k$$

$$= R_{yy}^{-1} [a_1 \ a_2] \cdot b_k$$

- It follows that $\forall \ k \in J$:

$$\hat{w}_k = W \cdot b_k$$

with $W = R_{yy}^{-1} [a_1 \ldots a_Q]$ an unknown $M \times Q$ matrix.

Conclusion

All MWF’s $\hat{w}_k, \forall \ k \in J$, span a $Q$-dimensional subspace!

$\Rightarrow$ Need to capture this subspace with DANSE
Generalization: \( \text{DANSE}_Q \)

- Choose \( Q - 1 \) auxiliary reference microphones at each node
- \( Q \)-channel desired signal, e.g., \( d_{k,\text{ref}} = [d_{k1} \ldots d_{kQ}]^T \) (w.l.o.g.)
- Compute \( Q \) different MWF's (\( M \times Q \) matrix):
  \[
  \hat{W}_k = R_{yy}^{-1} R_{dd} [e_{k1} \ldots e_{kQ}]
  \]
- From previous slide: \( \forall k, q \in J, \exists A_{kq} \in \mathbb{C}^{Q \times Q}: \hat{W}_k = \hat{W}_q A_{kq} \).
- If \( d_{k,\text{ref}} = A_{k,\text{ref}} \cdot s \), with \( A_{k,\text{ref}} \in \mathbb{C}^{Q \times Q} \) containing the \( Q \)-speakers to \( Q \) ref.-mic acoustic transfer functions, then \( A_{kq} = A_{q,\text{ref}}^{-H} \cdot A_{k,\text{ref}}^H \)
Generalization: DANSE$_Q$

**Q-channel signal broadcasts**

Replace single-channel $z^i_k = w^i_{kk} H y_k$ with a $Q$-channel signal $z'^i_k = W^i_{kk} H y_k$.

$\Rightarrow$ Communication cost increases linearly with \# target speakers.
The DANSE algorithm in fully-connected WASNs

DANSE\textsubscript{Q} parametrization

Example of DANSE\textsubscript{Q} parametrization (3-node case)

\[
W_1^i = \begin{bmatrix} W_{11}^i & W_{12}^i \end{bmatrix}, \quad W_2^i = \begin{bmatrix} W_{21}^i & G_{21}^i \\ W_{22}^i & W_{22}^i \end{bmatrix}, \quad W_3^i = \begin{bmatrix} W_{31}^i & G_{31}^i \\ W_{32}^i & G_{32}^i \\ W_{33}^i & W_{33}^i \end{bmatrix}
\]
DANSE\textsubscript{Q} parametrization

\[ W_k^i = \begin{bmatrix} W_{11}^i G_{k1}^i \\ \vdots \\ W_{NN}^i G_{kN}^i \end{bmatrix} \quad \text{where } G_{kq}^i \in \mathbb{C}^{Q \times Q}, \quad G_{kk}^i = I_Q \]

Since \( \forall k, q \in \mathcal{J} \), \( \exists A_{kq} \in \mathbb{C}^{Q \times Q} : \hat{W}_k = \hat{W}_q A_{kq} \), the optimal MWF’s are in the DANSE solution space (set \( W_{kk}^i = \hat{W}_{kk} \) and \( G_{kq}^i = A_{kq} \)).
Algorithm description

**DANSE\textsubscript{Q} algorithm:** computation of $W_{kk}^{i+1}$ and $G_{k,-k}^{i+1}$

Let $G_{k,-k}^{i} = [G_{k1}^{T} \ldots G_{k,k-1}^{T} G_{k,k+1}^{T} \ldots G_{kN}^{T}]^{T}$. Update at node $k$:

$$
\begin{bmatrix}
W_{kk}^{i+1} \\
G_{k,-k}^{i+1}
\end{bmatrix} = \begin{cases}
\left( R_{\tilde{y}_k\tilde{y}_k}^i \right)^{-1} R_{\tilde{d}_k\tilde{d}_k}^i [e_1 \ldots e_Q] & \text{if } k = u \\
W_{kk}^{i} \\
G_{k,-k}^{i}
\end{cases}
$$

if $k \neq u$

where $\tilde{y}_k^i$ and $\tilde{d}_k^i$ are defined as earlier (but with $Q$-channel $z_k^i$ signals).
Convergence and optimality of DANSE$_Q$

**Theorem (Convergence and optimality of DANSE$_Q$)**

In case of $Q$ desired speech sources, and if $\mathbf{A}_{k,\text{ref}}$ is full rank, $\forall \ k \in \mathcal{J}$, then $\lim_{i \to \infty} \mathbf{W}_k^i = \hat{\mathbf{W}}_k$, $\forall \ k \in \mathcal{J}$.

(proof omitted)
Other scenarios

What if the centralized solution is not in DANSE$_Q$ solution space, e.g.,
- DANSE$_Q$ with $Q < \text{number of desired speakers}$?
- DANSE$_Q$ where nodes have ‘different interests’

**Theorem (Existence of equilibrium [Bertrand and Moonen, 2012b])**

*Under some technical conditions (details omitted), the DANSE$_Q$ algorithm always has an equilibrium point, i.e., a choice of the local parameters $W_i^{kk}$ and $G_i^{kq}$, $\forall k, q \in J$, such that none of the nodes wants to change them.*

- Convergence to equilibrium is not proven, but is generally observed in simulations.
- Equilibrium $\neq$ suboptimal due to selfish updates.
- Game-theoretic framework (selfish nodes) $\rightarrow$ Nash equilibria
Simultaneous node-updating

- In DANSE, the nodes update in a sequential round-robin fashion
  ⇒ Slow overall convergence, and slow per-node adaptation
- Can we also let all nodes update simultaneously?
- Sometimes convergence...
- ... but often no convergence 😞 (limit cycle behavior)
- Reason: ‘optimal’ local update immediately becomes suboptimal due to simultaneous changes in the filters at other nodes
- Solution: Relaxation (details omitted, see [Bertrand and Moonen, 2010b])

\[
W_{kk}^{i+1} = (1 - \alpha)W_{kk}^i + \alpha W_{kk}^{\text{unrelaxed update}}
\]

with \(0 < \alpha \leq 1\).
Relaxed simultaneous DANSE (rS-DANSE)

rS-DANSE_Q algorithm: computation of $W_{kk}^{i+1}$ and $G_{k,-k}^{i+1}$

Update at all nodes $k \in \mathcal{J}$ simultaneously:

$$\begin{bmatrix} W_{kk}^{new} \\ G_{k,-k}^{i+1} \end{bmatrix} = (R_{\tilde{y}_k\tilde{y}_k})^{-1} R_{\tilde{d}_k\tilde{d}_k}^i [e_1 \ldots e_Q]$$

$$W_{kk}^{i+1} = (1 - \alpha)W_{kk}^i + \alpha W_{kk}^{new}$$

![Graph showing the performance of various algorithms](graph.png)

Legend:
- Optimal cost
- S-DANSE
- rS-DANSE with $\alpha=0.7$
- rS-DANSE with $\alpha=0.3$
- rS-DANSE with $\alpha=1/i$
Robustified DANSE (R-DANSE)

- Sometimes ill-conditioned nodes:
  \[ a_{k1} \approx 0 \text{ or } A_{k,\text{ref}} \approx \text{rank deficient} \]
- E.g.: low-SNR node \( k \) can be useful as noise reference, but \( a_{k1} \approx 0 \).
- DANSE suffers from error ripple: erroneous update at one node has an impact on the performance at all other nodes.

**Solution**

- At ill-conditioned node \( k \): choose \( z_q^i \) as reference signal, where node \( q \) is a high-SNR node.
- Note: ‘desired’ signal at node \( k \) changes with iteration index \( i \)!
Convergence and optimality of R-DANSE

Dependency graph:
- Each column $w_{ik}(m)$ of $W_{ik}$, $\forall k \in J$, $\forall m \in \{1, \ldots, Q\}$ is a vertex.
- Note: each $w_{ik}(m)$ corresponds to a particular reference mic
- Draw edge $w_{ik}(m) \rightarrow w_{ij}(n)$ if update of $w_{ik}(m)$ is based on the reference signal $z_{ik}(n)$ instead of a local microphone.

If dependency graph contains no loops: convergence and optimality of R-DANSE [Bertrand and Moonen, 2009].
1 Introduction and motivation

2 The DANSE algorithm in fully-connected WASNs

3 DANSE in WASNs with a tree topology (T-DANSE)

4 LCMV-based DANSE (LC-DANSE)

5 Bibliography
Multi-hop WASNs

- Fully-connected WASNs may require significant transmit power
- Low-power nodes may not be able to reach all other nodes
Passing on information

The relay case

- Make network virtually fully connected
- Complex routing problem
- Per-node communication cost grows with network size

Filter-and-sum combination of inputs

- No routing problems
- Per-node communication cost independent of network size
First attempt

Fully-connected DANSE:
First attempt

Disconnect red and green node...
First attempt

... and add new neighbors instead:
First attempt

Blue node’s data is blocked and does not travel beyond red node:
First attempt

Change definition of transmitted signal $z_k^i$ (‘wild guess’):
First attempt

Data from blue node travels beyond single-hop region:
First attempt

Apply similar idea in all nodes:
First attempt

Will this ‘wild guess’ work???

- $\mathcal{N}_k =$ neighbours of $k$ ($k$ excl.)
- Implicit definition of $z^i_k$:
  \[ z^i_k = W^i_{kk} y_k + \sum_{q \in \mathcal{N}_k} G^i_{kq} z^i_q \]

Problem 1: acausality in data flow

Deadlock: nodes wait for each other’s $z$-signals
First attempt

**Problem 2: feedback**

- Feedback path considerably *changes algorithm dynamics*
- Centralized MWF’s are not in solution space (provable)

How to get rid of this *feedback and causality problem*?
2 types of feedback

Direct feedback

Indirect feedback
Eliminating direct feedback

Direct feedback:

- **Transmitter feedback cancellation (TFC):** send different signal to each neighbour

\[ z_{kq}^i = W_{kk}^i y_k + \sum_{l \in N_k \setminus \{q\}} G_{kl}^i z_{lk}^i \]

- Better alternative: **Receiver feedback cancellation (RFC), i.e., single broadcast signal to all neighbors** (details omitted [Bertrand and Moonen, 2011])

- RFC vs. TFC: no influence on algorithm! (will assume TFC in sequel w.l.o.g.)
Eliminating indirect feedback

Indirect feedback:

- Prune to tree topology
- In combination with TFC: all feedback eliminated
- Definition of $z_{kq}^i$'s can be resolved:
  - Start at leaf nodes ($|N_k| = 1$)
  - Leaf node $k$: $z_{kq}^i = W_{kk}^i H y_k$, i.e., no dependency on other $z$-signals
  - Rest follows in natural order as dictated by the tree

Similarly, also causality problem in data flow (deadlock) is resolved:

1. Fusion flow from leaf nodes to root...
2. ... followed by diffusion flow from root to leaves
Data-driven signal exchange

- **Data-driven paradigm**: each block ‘fires’ if all of its inputs are available ⇒ no global coordination needed to organize data flow
- Fusion and diffusion flow emerge automatically
Parametrization: example

\[ W^i_1 = \begin{bmatrix} W_{11}^i \\ \ast \\ W_{33}^i G_{13}^i \\ W_{44}^i G_{34}^i G_{13}^i \\ \ast \\ \ast \\ \ast \\ W_{88}^i G_{48}^i G_{34}^i G_{13}^i \end{bmatrix} \]

\[ W^i_4 = \begin{bmatrix} W_{11}^i G_{31}^i G_{43}^i \\ \ast \\ W_{33}^i G_{43}^i \\ W_{44}^i \\ \ast \\ \ast \\ \ast \\ W_{88}^i G_{48}^i \end{bmatrix} \]
General parametrization of Tree-DANSE (T-DANSE)

General parametrization of T-DANSE

\[
W^i_k = \begin{bmatrix}
W_{11}^i G^i_{k \leftarrow 1} \\
\vdots \\
W_{NN}^i G^i_{k \leftarrow N}
\end{bmatrix}
\]

- \( G^i_{p_1 \leftarrow p_t} = G^i_{p_{t-1} p_t} G^i_{p_{t-2} p_{t-1}} \ldots G^i_{p_2 p_3} G^i_{p_1 p_2} \) with order defined by unique path \( P_{p_t \rightarrow p_1} = (p_t, p_{t-1}, \ldots, p_2, p_1) \) from \( p_t \) to \( p_1 \).

- By definition: \( G^i_{k \leftarrow k} = I_Q \)

Compare with fully-connected DANSE:

\[
W^i_k = \begin{bmatrix}
W_{11}^i G^i_{k_1} \\
\vdots \\
W^i_{NN} G^i_{k_N}
\end{bmatrix}
\]
DANSE in WASNs with a tree topology (T-DANSE)

Parametrization: example

Complete parametrization of network-wide filter $W_i^4$:

$$W_i^4 = \begin{bmatrix}
W_{11}^i G_i^4 \leftarrow 1 \\
W_{22}^i G_i^4 \leftarrow 2 \\
W_{33}^i G_i^4 \leftarrow 3 \\
W_{44}^i \\
W_{55}^i G_i^4 \leftarrow 5 \\
W_{66}^i G_i^4 \leftarrow 6 \\
W_{77}^i G_i^4 \leftarrow 7 \\
W_{88}^i G_i^4 \leftarrow 8
\end{bmatrix} = \begin{bmatrix}
W_{11}^i G_i^3 G_i^{43} \\
W_{22} G_i^3 G_i^{43} \\
W_{33} G_i^{43} \\
W_{44}^i \\
W_{55} G_i^6 G_i^{46} \\
W_{66} G_i^{46} \\
W_{77} G_i^6 G_i^{46} \\
W_{88} G_i^{48}
\end{bmatrix}$$
Centralized MWF in T-DANSE solution space?

Theorem

In case of $Q$ desired speech sources, and if $A_{k,\text{ref}}$ is full rank, $\forall \ k \in \mathcal{J}$, then the solution space defined by the parametrization of T-DANSE contains the optimal MWFs $\hat{W}_k$, $\forall \ k \in \mathcal{J}$.

Proof:

- Reminder: $\forall \ k, q \in \mathcal{J}: \hat{W}_k = \hat{W}_q A_{kq}$, where
  $$A_{kq} = A_{q,\text{ref}}^{-H} \cdot A_{k,\text{ref}}^H$$
- Therefore: $\forall \ k, q, n \in \mathcal{J}: A_{nq} A_{kn} = A_{kq}$
- Set $G_{mn}^i = A_{mn}$, then
  $$G_{k\leftarrow q}^i = A_{p_{t-1}q} \cdot A_{p_{t-2}p_{t-1}} \cdots A_{p_2p_3} \cdot A_{kp_2}$$
  $$= A_{kq}$$

  where $P_{k\leftarrow q} = (q, p_{t-1}, p_{t-2}, \ldots, p_3, p_2, k)$
- Hence, set $W_{kk}^i = \hat{W}_{kk}$ and $G_{mn}^i = A_{mn}$, then $W_k^i = \hat{W}_k$, Q.E.D.
T-DANSE updating procedure

- Let \( z^i_k = [z^{iT}_{n_1} \ldots z^{iT}_{n_{N_k}}]^T \).
- Node \( k \) sets internal fusion rules

\[
W^i_{kk} \text{ and } G^i_{k,-k} = \begin{bmatrix} G^{iT}_{n_1} & \ldots & G^{iT}_{n_{N_k}} \end{bmatrix}^T
\]

with \( n_j \in \mathcal{N}_k \) and \( N_k = |\mathcal{N}_k| \).
T-DANSE updating procedure

T-DANSE\(_Q\) algorithm: computation of \(W_{kk}^{i+1}\) and \(G_{k,-k}^{i+1}\)

- If \(k \neq u\), then \(W_{kk}^{i+1} = W_{kk}^i\) and \(G_{k,-k}^{i+1} = G_{k,-k}^i\)
- If \(k = u\):

\[
\begin{bmatrix}
W_{kk}^{i+1} \\
G_{k,-k}^{i+1}
\end{bmatrix} = \arg \min_{W_{kk}, G_{k,-k}} E \left\{ \left\| d_k - \begin{bmatrix} W_{kk}^H & G_{k,-k}^H \end{bmatrix} \begin{bmatrix} y_k \\ z_i \to_k \end{bmatrix} \right\|^2 \right\}
\]

\[
= (R_{\tilde{y}_k \tilde{y}_k}^i)^{-1} R_{\tilde{d}_k \tilde{d}_k}^i [e_1 \ldots e_Q]
\]

where \(\tilde{y}_k^i = [y_k^T \ z_i \to_k]^T\), and similarly for \(\tilde{d}_k^i\).

- Identical to fully-connected DANSE updates (but less input signals per node)
- Note: sequential updates (only one node updates in each iteration)
DANSE in WASNs with a tree topology (T-DANSE)

Convergence and optimality of T-DANSE

Theorem (Convergence and optimality of T-DANSE [Bertrand and Moonen, 2011])

In case of Q desired speech sources, if $A_{k,\text{ref}}$ is full rank, $\forall k \in J$, and if the node-per-node updating order of T-DANSE is defined by a path through the network that visits all nodes, then $\lim_{i \to \infty} W^i_k = \hat{W}_k$, $\forall k \in J$.

- Note: updating order must follow a path through the network
- Random order updating also works in general, but no proof
- However: path-based updating converges faster (experimental observation)
1. Introduction and motivation

2. The DANSE algorithm in fully-connected WASNs

3. DANSE in WASNs with a tree topology (T-DANSE)

4. LCMV-based DANSE (LC-DANSE)

5. Bibliography
LCMV beamforming revisited

Centralized node-specific LCMV BF at node $k$:

$$\hat{w}_k = \arg \min_{w_k} \left( w_k^H R_{yy} w_k, \text{s.t.} \ A^H w_k = f_k \right)$$

$$= R_{yy}^{-1} A \left( A^H R_{yy}^{-1} A \right)^{-1} f_k$$

- $A$ $M \times Q$ steering matrix from $Q$ ‘relevant’ sources to $M$ microphones
- $f_k$ node-specific response for each of the $Q$ sources
- Relevant sources may also contain interferers!

PS: Will assume in sequel that $A$ is known. For unknown $A$, refer to [Markovich et al., 2009] or [Bertrand and Moonen, 2012a]
DANSE ↔ (SDW-)MWF
LC-DANSE ↔ LCMV
Similar idea, similar block scheme

\[ Q \] is \# constraints
Linearily-constrained DANSE (LC-DANSE)

- \( \hat{w}_k = R_{yy}^{-1} A (A^H R_{yy}^{-1} A)^{-1} f_k \)
  \( \Rightarrow \) joint \( Q \)-dim subspace: \( \hat{w}_k = W \cdot f_k, \forall k \in J. \)

- Add \( Q - 1 \) auxiliary LCMV-problems:
  \[
  \hat{W}_k = \arg \min_{W_k} \left( \text{Tr} \left( W_k^H R_{yy} W_k \right) \right), \text{ s.t. } A^H W_k = F_k
  \]
  \[
  = R_{yy}^{-1} A \left( A^H R_{yy}^{-1} A \right)^{-1} F_k
  \]
  with \( F_k \) a \( Q \times Q \) matrix of full rank, with \( f_k \) in first column.

- \( \forall k, q \in J : \hat{W}_k = \hat{W}_q A_{kq} \) with
  \[
  A_{kq} = F_q^{-1} F_k
  \]

Conclusion: Centralized LCMV solutions are in (LC-)DANSE solution space! (set \( W_{kk}^i = \hat{W}_{kk} \) and \( G_{kq}^i = A_{kq} \))
Match constraints with compressed signals:

\[ y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \leftrightarrow A = \begin{bmatrix} A_1 \\ \vdots \\ A_N \end{bmatrix} \]

\[ \tilde{y}_k^i = \begin{bmatrix} y_k^i \\ z_{k-1}^i \\ \vdots \\ z_N^i \end{bmatrix} \leftrightarrow \tilde{A}_k^i = \begin{bmatrix} A_k^i \\ C_{k-1}^i \\ \vdots \\ C_N^i \end{bmatrix} \]

\[ z_{-k}^i = \begin{bmatrix} z_1^i \\ \vdots \\ z_{k-1}^i \\ z_{k+1}^i \\ \vdots \\ z_N^i \end{bmatrix} \leftrightarrow C_{-k}^i = \begin{bmatrix} C_1^i \\ \vdots \\ C_{k-1}^i \\ C_{k+1}^i \\ \vdots \\ C_N^i \end{bmatrix} \]

\[ z_k^i = W_{kk}^i H y_k \leftrightarrow C_k^i = W_{kk}^i A_k \]
LC-DANSE Algorithm description

**LC-DANSE**

Note: computation of \( \tilde{A}_k \) requires exchange of \( W_{kk} \)'s. However, filter coefficients are typically frozen for some time (2-3s), hence negligible compared to data rate of \( z_k \)'s.
LC-DANSE: final remarks

- Provable convergence and optimality
- Further reading: [Bertrand and Moonen, 2012a]
- $Q$ constraints $\Rightarrow$ $Q$-channel broadcast signals
- If node-specific aspect is removed (same $f_k$ in all nodes): single-channel $z_k^i$’s are sufficient! [Bertrand and Moonen, 2013]
- Related GSC implementation: [Markovich-Golan et al., 2013] (covered in part III)
Bibliography

References and Further Reading I

More is less: Signal processing and the data deluge.

Robust distributed noise reduction in hearing aids with external acoustic sensor nodes.

Distributed adaptive node-specific signal estimation in fully connected sensor networks – part I: sequential node updating.

Distributed adaptive node-specific signal estimation in fully connected sensor networks – part II: simultaneous & asynchronous node updating.

Distributed adaptive estimation of node-specific signals in wireless sensor networks with a tree topology.

Distributed node-specific LCMV beamforming in wireless sensor networks.

Distributed signal estimation in sensor networks where nodes have different interests.
Distributed LCMV beamforming in a wireless sensor network with single-channel per-node signal transmission.

GSVD-based optimal filtering for single and multimicrophone speech enhancement.

Reduced-bandwidth and distributed MWF-based noise reduction algorithms for binaural hearing aids.

Distributed MVDR beamforming for (wireless) microphone networks using message passing.
In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*.

Clustered blind beamforming from ad-hoc microphone arrays.

Distributed blind source separation with an application to audio signals.

Analysis of rate constraints for MWF-based noise reduction in acoustic sensor networks.
References and Further Reading III

Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals. 

A reduced bandwidth binaural MVDR beamformer.
In *Proc. of the International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Tel-Aviv, Israel.

Distributed multiple constraints generalized sidelobe canceler for fully connected wireless acoustic sensor networks.

Consensus and cooperation in networked multi-agent systems.

Diffusion strategies for adaptation and learning over networks: an examination of distributed strategies and network behavior.

Gossip algorithms.

Srinivasan, S. and Den Brinker, A. C. (Article ID 257197, 14 pages, 2009.).
Rate-constrained beamforming in binaural hearing aids.
Improved tracking performance for distributed node-specific signal enhancement in wireless acoustic sensor networks.

Distributed delay and sum beamformer for speech enhancement in wireless sensor networks via randomized gossip.
In IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), pages 4037–4040.