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

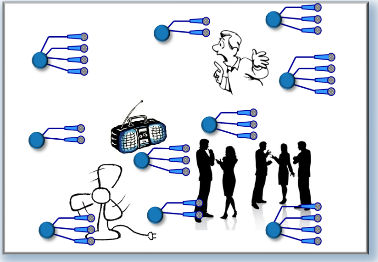
EUSIPCO 2013, Marrakech, Morocco

Outline

Ad hoc microphone arrays

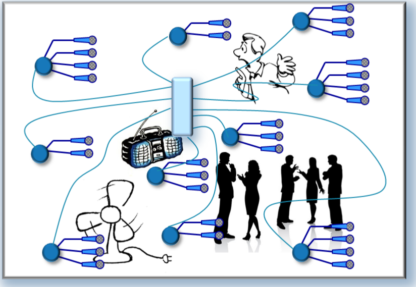
Ad hoc microphone arrays

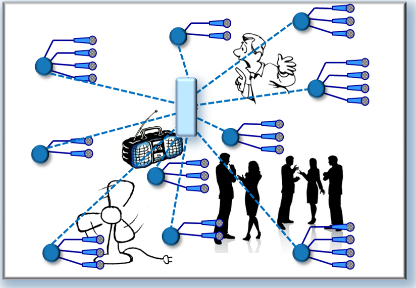
- No tedious calibration
- Improved spatial resolution and sound field sampling.
- High probability to find microphones close to a relevant sound source.
- Possibility to put (arrays of) microphones at strategic places



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)

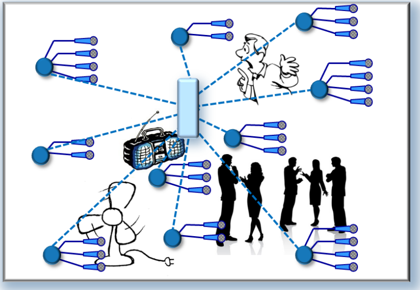
Possible applications:

- 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

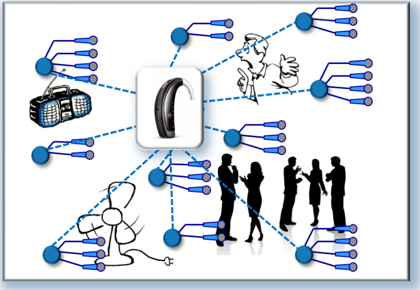
- Wireless link delay (e.g., in case of real-time constraints)
- Different sampling clocks (see also Part III)
- The 'data deluge' (see next slide)



WASNs and the data deluge

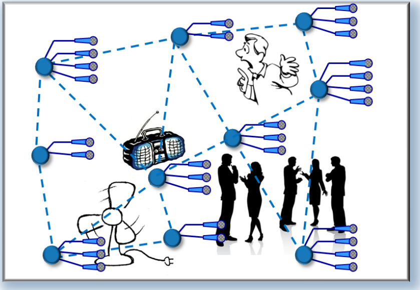
The 'data deluge' [Baraniuk, 2011]

- 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)



Distributed signal processing in WASNs

- 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:

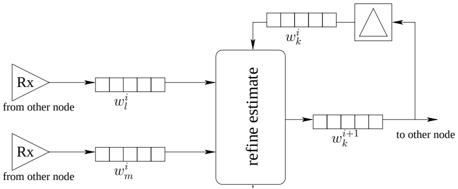
- Mainly driven by the concept of wireless sensor networks
- Theory and methods often build upon results from other fields, e.g.,
  - Parallel and distributed computing for multi-core processors
  - Modelling and control of multi-agent systems
  - Game theory
  - Graph theory
- Two fundamentally different 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, ...)

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

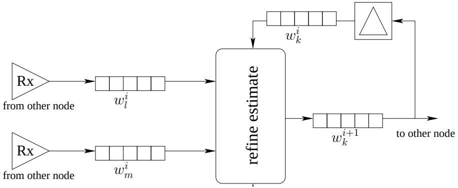
Note: target parameter vector is fixed over iterations, or varies only slowly



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.

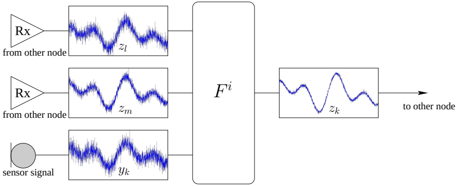
- ☺ DPE techniques usually have no network topology constraints
- ☹ Large communication cost: re-estimate and re-transmit same L samples many times (freeze time index until convergence)
- ☹ Communication cost depends on convergence speed (and hence also on network size)
- ☹ Not time-recursive: full reset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



See, e.g., [Zeng and Hendriks, 2012, Heusdens et al., 2012]

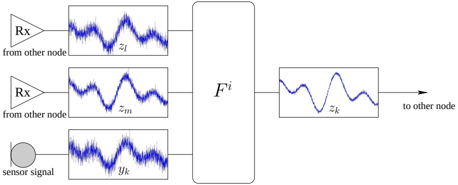
Distributed signal estimation (DSE)

- Avoid iterations over the signal sample estimates themselves
  $\Rightarrow$ 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?

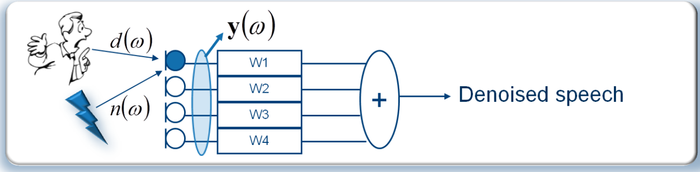
- No iterative refinement of sample estimates:
  ☺ Each block of samples is transmitted only once
  ☺ Fixed per-node communication cost, independent of convergence speed/network size
- ☹ Price to pay: specific order in data flow generally requires topology constraints (star, tree, fully-connected,...)



See, e.g., [Doclo et al., 2009, Bertrand and Moonen, 2009, Markovich-Golan et al., 2010, Markovich-Golan et al., 2013, Lawin-Ore and Doclo, 2011, Himawan et al., 2011, Hioka and Kleijn, 2011, Szurley et al., 2013]

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

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



$$\min_{\mathbf{w}} E\{| d_{ref} - \mathbf{w}^H \mathbf{y} |^2\}$$

$$\mathbf{w}(\omega) = \mathbf{R}_{yy}(\omega)^{-1} \mathbf{R}_{dd}(\omega)\, \mathbf{e}_{ref}$$

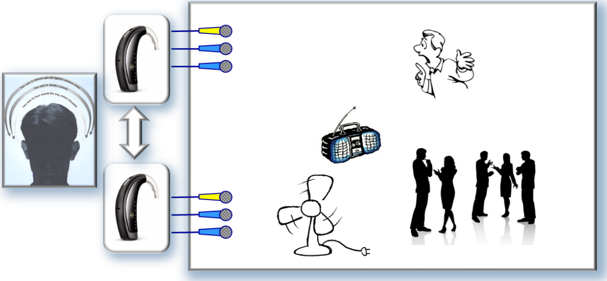
$$\mathbf{R}_{yy}(\omega) = E\{\mathbf{y}(\omega)\mathbf{y}(\omega)^H\}$$

Voice activity detection (VAD)

$$\mathbf{R}_{dd}(\omega) = \mathbf{R}_{yy}(\omega) - \mathbf{R}_{nn}(\omega)$$

Preliminary case study: binaural hearing aids

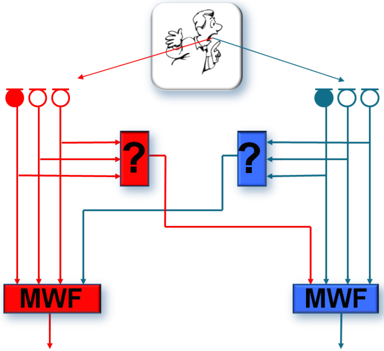
- Two hearing aids (HAs) with wireless link (=2-node WASN)
- Goal: compute MWF including extra signal(s) from other HA
- Each HA uses a local microphone as reference to preserve binaural cues of target speaker



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?

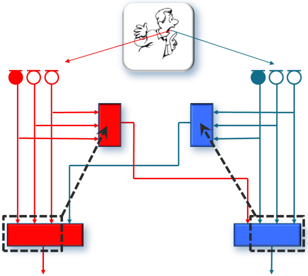


Preliminary case study: binaural hearing aids

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]

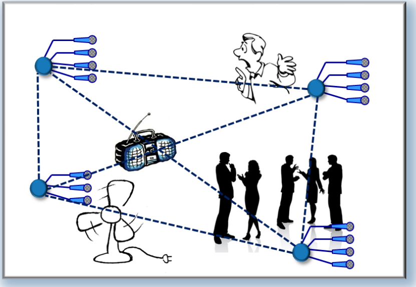


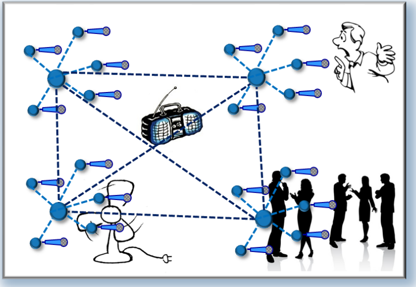
DANSE in fully-connected WASNs

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

  $\Rightarrow$ 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 $\mathbf{y}_k(\omega, t)$ (represented in short-time Fourier transform (STFT) domain)
- Will often omit (ω, t) in the sequel for conciseness, keep in mind that all operations are performed in STFT domain.
- Additive noise:

$$\mathbf{y}_k = \mathbf{d}_k + \mathbf{n}_k$$

  $\mathbf{n}_k$ is noise and $\mathbf{d}_k$ is the desired speech signal.
- Stacked vector $\mathbf{y} = [\mathbf{y}_1^T \ \ldots \ \mathbf{y}_N^T]^T$ defines M-channel signal with $M = \sum_{k \in \mathcal{J}} M_k$.
- Similar for $\mathbf{d}$ and $\mathbf{n}$, i.e., $\mathbf{y} = \mathbf{d} + \mathbf{n}$.
- y_{km} denotes the m-th microphone of node k, and $\mathbf{e}_{km} = [0 \ \ldots \ 0 \ 1 \ 0 \ \ldots \ 0]^T$ is a selection vector such that $y_{km} = \mathbf{e}_{km}^T \mathbf{y}$.

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

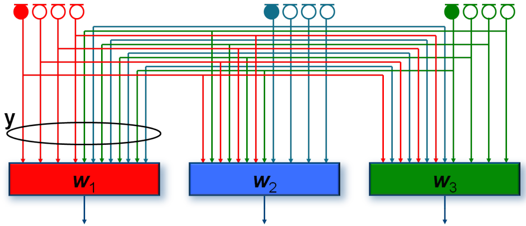
$$\hat{d}_{k1} = \hat{\mathbf{w}}_k^H \mathbf{y}$$

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

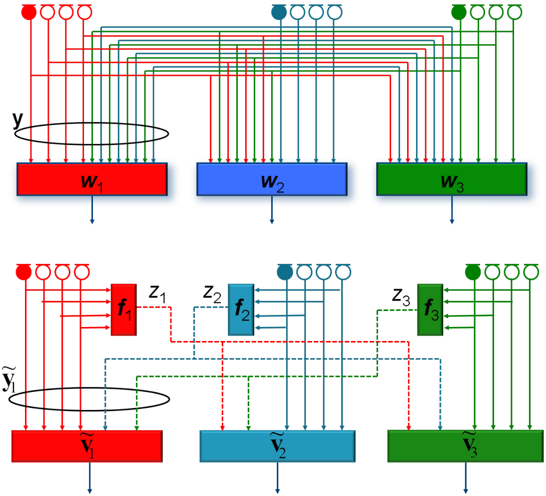
$$\hat{\mathbf{w}}_k = \arg\min_{\mathbf{w}_k} E\{|d_{k1} - \mathbf{w}_k^H \mathbf{y}|^2\} = \mathbf{R}_{yy}^{-1}\mathbf{R}_{dd}\mathbf{e}_{k1}$$

where $\mathbf{R}_{yy} = E\{\mathbf{y}\mathbf{y}^H\}$ and $\mathbf{R}_{dd} = E\{\mathbf{d}\mathbf{d}^H\} = \mathbf{R}_{yy} - \mathbf{R}_{nn}$ (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:

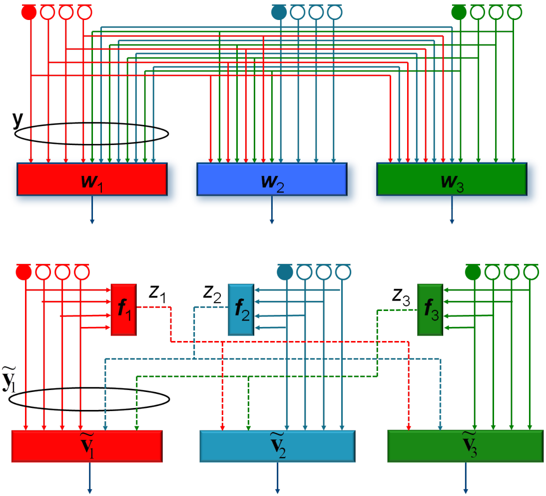
$$z_k^i = \mathbf{f}_k^{i\,H} \mathbf{y}_k$$

  where $\mathbf{f}_k^i$ is an M_k-dimensional fusion vector and i is an iteration index.
- Data compression: M_k-channel signal $\mathbf{y}_k \rightarrow$ single-channel signal z_k^i
- Between iteration i and $i+1$, node k collects samples of

$$\widetilde{\mathbf{y}}_k^i = \begin{bmatrix} \mathbf{y}_k \\ \mathbf{z}_{-k}^i \end{bmatrix} = \widetilde{\mathbf{d}}_k^i + \widetilde{\mathbf{n}}_k^i$$

  with $\mathbf{z}_{-k}^i = [z_1^i \ \ldots \ z_{k-1}^i \ z_{k+1}^i \ \ldots \ z_N^i]^T$.

DANSE signal exchange



DANSE per-node MWFs

- Node k will compute local MWF $\widetilde{\mathbf{v}}_k^i$ that minimizes

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

- This yields

$$\widetilde{\mathbf{v}}_k^i = \left(\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k}\right)^{-1} \mathbf{R}^i_{\tilde{d}_k\tilde{d}_k} \mathbf{e}_1$$

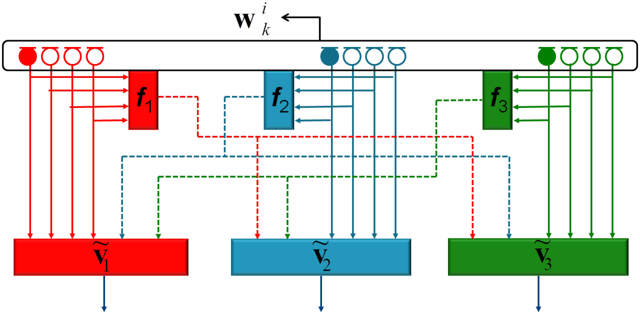
  where $\mathbf{e}_1 = [1\ 0\ \ldots\ 0]$, $\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k} = E\{\widetilde{\mathbf{y}}_k^i \widetilde{\mathbf{y}}_k^{i\,H}\}$, $\mathbf{R}^i_{\tilde{d}_k\tilde{d}_k} = E\{\widetilde{\mathbf{d}}_k^i \widetilde{\mathbf{d}}_k^{i\,H}\}$.
- With VAD: $\mathbf{R}^i_{\tilde{d}_k\tilde{d}_k} = \mathbf{R}^i_{\tilde{y}_k\tilde{y}_k} - \mathbf{R}^i_{\tilde{n}_k\tilde{n}_k}$ (PS: nodes can share VAD info)
- Between iterations i and $i+1$, estimated speech signal at node k:

$$\overline{d}_{k1}^i = \widetilde{\mathbf{v}}_k^{i\,H}\, \widetilde{\mathbf{y}}_k^i$$

Equivalent network-wide filter?

$\Rightarrow$ how does equivalent network-wide filter $\mathbf{w}_k^i$ look like?

$$\overline{d}_{k1}^i = \tilde{\mathbf{v}}_k^{i\,H}\,\tilde{\mathbf{y}}_k^i = \mathbf{w}_k^{i\,H}\,\mathbf{y} \Rightarrow \mathbf{w}_k^i?$$

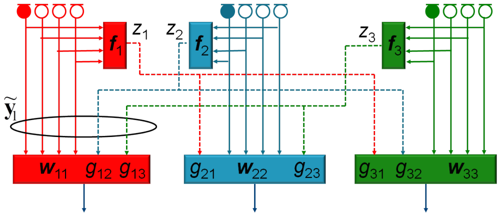


Equivalent network-wide filter?

Local MWF $\leftrightarrow$ network-wide filter

$$\mathbf{w}_1^i = \begin{bmatrix} \mathbf{w}_{11}^i \\ g_{12}^i \mathbf{f}_2^i \\ g_{13}^i \mathbf{f}_3^i \end{bmatrix}, \quad \mathbf{w}_2^i = \begin{bmatrix} g_{21}^i \mathbf{f}_1^i \\ \mathbf{w}_{22}^i \\ g_{23}^i \mathbf{f}_3^i \end{bmatrix}, \quad \mathbf{w}_3^i = \begin{bmatrix} g_{31}^i \mathbf{f}_1^i \\ g_{32}^i \mathbf{f}_2^i \\ \mathbf{w}_{33}^i \end{bmatrix}$$

g_{kq}^i is the coefficient that node k applies to the z_q^i signal from node q.



$$\mathbf{w}_k^i = \begin{bmatrix} \mathbf{w}_{k1}^i \\ \mathbf{w}_{k2}^i \\ \vdots \\ \mathbf{w}_{kk}^i \\ \vdots \\ \mathbf{w}_{kN}^i \end{bmatrix} = \begin{bmatrix} g_{k1}^i \mathbf{f}_1^i \\ g_{k2}^i \mathbf{f}_2^i \\ \vdots \\ \mathbf{w}_{kk}^i \\ \vdots \\ g_{kN}^i \mathbf{f}_N^i \end{bmatrix}$$

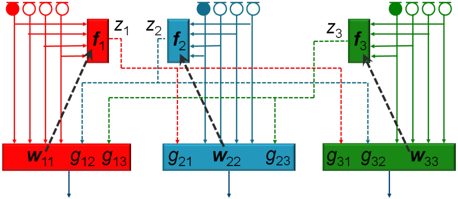
DANSE parametrization

Choice of $\mathbf{f}_k^i$'s

DANSE sets $\mathbf{f}_k^i = \mathbf{w}_{kk}^i$, i.e., $\mathbf{w}_{kk}^i$ serves both as compressor and estimator

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

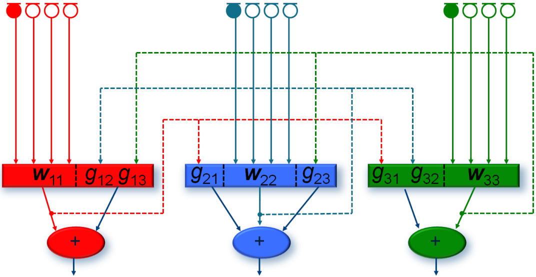
PS: chicken-and-egg problem: need samples of z_k signals to compute local MWFs, but need MWFs to compute samples of z_k's



DANSE parametrization

Example of DANSE parametrization (3-node case)

$$\mathbf{w}_1^i = \begin{bmatrix} \mathbf{w}_{11}^i \\ g_{12}^i \mathbf{w}_{22}^i \\ g_{13}^i \mathbf{w}_{33}^i \end{bmatrix}, \quad \mathbf{w}_2^i = \begin{bmatrix} g_{21}^i \mathbf{w}_{11}^i \\ \mathbf{w}_{22}^i \\ g_{23}^i \mathbf{w}_{33}^i \end{bmatrix}, \quad \mathbf{w}_3^i = \begin{bmatrix} g_{31}^i \mathbf{w}_{11}^i \\ g_{32}^i \mathbf{w}_{22}^i \\ \mathbf{w}_{33}^i \end{bmatrix}$$



PS: similar to $\mathbf{z}_{-k}^i$, introduce notation

$$\mathbf{g}_{k,-k}^i = [g_{k1}^i \ \cdots \ g_{k,k-1}^i \ g_{k,k+1}^i \ \cdots \ g_{kN}^i]^T.$$

Algorithm description (for fixed frequency index ω)

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

1. Initialize: $i \leftarrow 0$, $u \leftarrow 1$
   Initialize $\mathbf{w}_{kk}^0$ and $\mathbf{g}_{k,-k}^0$ 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 $\mathbf{y}_k(\omega, iB+n)$, $n = 0 \ldots B-1$.
   - Compress these M_k-dimensional observations to

$$z_k^i(\omega, iB+n) = \mathbf{w}_{kk}^{i\,H}\mathbf{y}_k(\omega, iB+n), \;\; n = 0 \ldots B-1\,.$$

   - Broadcast B samples of z_k^i to other nodes.
   - Collect B samples of $\mathbf{z}_{-k}^i$ from other nodes.
   - Compute new estimator parameters $\mathbf{w}_{kk}^{i+1}$ and $\mathbf{g}_{k,-k}^{i+1}$ (see next slide).
   - Compute B samples of speech estimate (for $n = 0 \ldots B-1$)

$$\overline{d}_{k1}^i(\omega, iB+n) = \mathbf{w}_{kk}^{i+1\,H}\mathbf{y}_k(\omega, iB+n) + \mathbf{g}_{k,-k}^{i+1\,H}\mathbf{z}_{-k}^i(\omega, iB+n)\,.$$

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 $\mathbf{w}_{kk}^{i+1}$ and $\mathbf{g}_{k,-k}^{i+1}$

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

$$\begin{bmatrix} \mathbf{w}_{kk}^{i+1} \\ \mathbf{g}_{k,-k}^{i+1} \end{bmatrix} = \begin{cases} \left(\mathbf{R}^i_{\tilde{y}_k\tilde{y}_k}\right)^{-1} \mathbf{R}^i_{\tilde{d}_k\tilde{d}_k}\mathbf{e}_1 & \text{if } k = u \\ \begin{bmatrix} \mathbf{w}_{kk}^{i} \\ \mathbf{g}_{k,-k}^{i} \end{bmatrix} & \text{if } k \neq u \end{cases}$$

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 ω)

Convergence and optimality of DANSE?

Convergence

Does DANSE converge to an equilibrium?
$\Rightarrow$ Does $\lim_{i\to\infty} \mathbf{w}_k^i$ 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?
$\Rightarrow$ Is $\lim_{i\to\infty} \mathbf{w}_k^i = \hat{\mathbf{w}}_k,\ \forall\, k \in \mathcal{J}$?

1st result

First question: are $\hat{\mathbf{w}}_k$, $\forall\, k \in \mathcal{J}$, 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 $\hat{\mathbf{w}}_k$, $\forall\, k \in \mathcal{J}$.

Proof outline:

- Single desired speech source:

$$\forall\, k \in \mathcal{J} :\ \mathbf{d}_k(\omega, t) = \mathbf{a}_k(\omega) s(\omega, t)$$

  where $s(\omega, t)$ contains desired speech source and steering vector $\mathbf{a}_k(\omega)$ contains M_k transfer functions from source to M_k microphones.
- Let $\mathbf{a} = [\mathbf{a}_1^T\ \ldots\ \mathbf{a}_N^T]^T$, then $\mathbf{d}(\omega, t) = \mathbf{a}(\omega) s(\omega, t)$.

Proof (continued)

- Centralized MWF at node k:

$$\begin{aligned}\hat{\mathbf{w}}_k &= \mathbf{R}_{yy}^{-1}\mathbf{R}_{dd}\mathbf{e}_{k1} \\ &= \mathbf{R}_{yy}^{-1}\mathbf{a}E\{|s|^2\}\mathbf{a}^H\mathbf{e}_{k1} \\ &= \mathbf{R}_{yy}^{-1}\mathbf{a} \cdot a_{k1}^* E\{|s|^2\}\end{aligned}$$

- It follows that $\forall\, k, q \in \mathcal{J}$:

$$\hat{\mathbf{w}}_k = \alpha_{kq}\hat{\mathbf{w}}_q$$

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

- In DANSE: set $g_{kq}^i = \alpha_{kq}$ and $\mathbf{w}_{kk}^i = \hat{\mathbf{w}}_{kk}$, $\forall\, k, q \in \mathcal{J}$

$$\forall\, k \in \mathcal{J} : \ \mathbf{w}_k^i = \begin{bmatrix} g_{k1}^i\mathbf{w}_{11}^i \\ \vdots \\ g_{kN}^i\mathbf{w}_{NN}^i \end{bmatrix} = \begin{bmatrix} \alpha_{k1}\hat{\mathbf{w}}_{11} \\ \vdots \\ \alpha_{kN}\hat{\mathbf{w}}_{NN} \end{bmatrix} = \hat{\mathbf{w}}_k \ .$$

2nd result

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

In case of a single desired speech source, and if $a_{k1} \neq 0$, $\forall\, k \in \mathcal{J}$, then $\lim_{i\to\infty} \mathbf{w}_k^i = \hat{\mathbf{w}}_k$, $\forall\, k \in \mathcal{J}$.

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
  $\Rightarrow$ Small *per-node* processing power
- Inherent dimensionality reduction
  $\Rightarrow$ Many small problems vs. single large problem
  $\Rightarrow$ Often smaller *overall* processing power (due to $O(M^2)$ or $O(M^3)$ complexity)

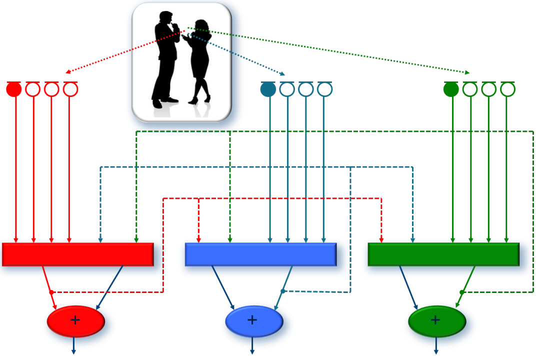
Disadvantages of DANSE

- 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_{k1} is a mixture of Q desired speech sources?

$\Rightarrow$ $\hat{\mathbf{w}}_k = \alpha_{kq}\hat{\mathbf{w}}_q$ does not hold anymore (see next slide)
$\Rightarrow$ $\hat{\mathbf{w}}_k$ not in solution space of DANSE ☹



Multiple target speakers

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

$$\begin{aligned}\hat{\mathbf{w}}_k &= \mathbf{R}_{yy}^{-1}\mathbf{R}_{dd}\mathbf{e}_{k1} \\ &= \mathbf{R}_{yy}^{-1}\,[\mathbf{a}_1\ \mathbf{a}_2]\begin{bmatrix} E\{|s_1|^2\} & 0 \\ 0 & E\{|s_2|^2\}\end{bmatrix}\begin{bmatrix}\mathbf{a}_1^H \\ \mathbf{a}_2^H\end{bmatrix}\mathbf{e}_{k1} \\ &= \mathbf{R}_{yy}^{-1}\,[\mathbf{a}_1\ \mathbf{a}_2]\cdot\mathbf{b}_k\end{aligned}$$

- It follows that $\forall\, k \in \mathcal{J}$:

$$\hat{\mathbf{w}}_k = \mathbf{W}\cdot\mathbf{b}_k$$

with $\mathbf{W} = \mathbf{R}_{yy}^{-1}\,[\mathbf{a}_1\ \ldots\ \mathbf{a}_Q]$ an unknown $M \times Q$ matrix.

Conclusion

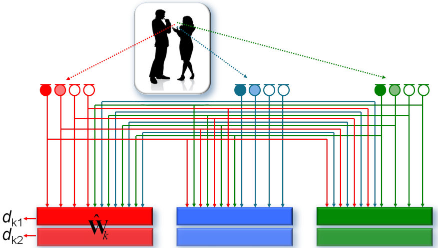
All MWF's $\hat{\mathbf{w}}_k, \forall\, k \in \mathcal{J}$, span a Q-dimensional subspace!
$\Rightarrow$ Need to capture this subspace with DANSE

Generalization: DANSE$_Q$

- Choose $Q-1$ auxiliary reference microphones at each node
- Q-channel desired signal, e.g., $\mathbf{d}_{k,\text{ref}} = [d_{k1} \ \ldots \ d_{kQ}]^T$ (w.l.o.g.)
- Compute Q different MWF's ($M \times Q$ matrix):

$$\hat{\mathbf{W}}_k = \mathbf{R}_{yy}^{-1}\mathbf{R}_{dd}\left[\mathbf{e}_{k1} \ \ldots \ \mathbf{e}_{kQ}\right]$$

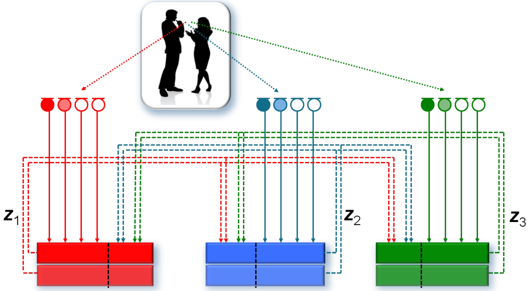
- From previous slide: $\forall\, k, q \in \mathcal{J},\ \exists\, \mathbf{A}_{kq} \in \mathbb{C}^{Q\times Q}:\ \hat{\mathbf{W}}_k = \hat{\mathbf{W}}_q \mathbf{A}_{kq}$.
- If $\mathbf{d}_{k,\text{ref}} = \mathbf{A}_{k,\text{ref}} \cdot \mathbf{s}$, with $\mathbf{A}_{k,\text{ref}} \in \mathbb{C}^{Q\times Q}$ containing the Q-speakers to Q ref.-mic acoustic transfer functions, then $\mathbf{A}_{kq} = \mathbf{A}_{q,\text{ref}}^{-H} \cdot \mathbf{A}_{k,\text{ref}}^{H}$



Generalization: DANSE$_Q$

Q-channel signal broadcasts

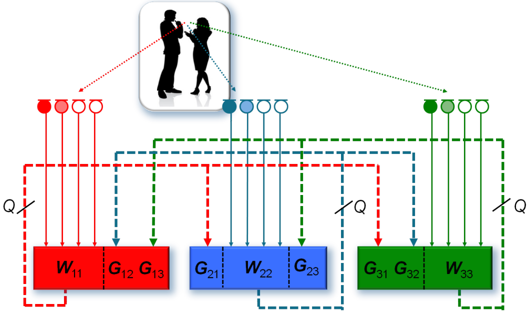
Replace single-channel $z_k^i = \mathbf{w}_{kk}^{i\,H}\mathbf{y}_k$ with a Q-channel signal $\mathbf{z}_k^i = \mathbf{W}_{kk}^{i\,H}\mathbf{y}_k$.
$\Rightarrow$ Communication cost increases linearly with # target speakers



DANSE$_Q$ parametrization

Example of DANSE$_Q$ parametrization (3-node case)

$$\mathbf{W}_1^i = \begin{bmatrix} \mathbf{W}_{11}^i \\ \mathbf{W}_{22}^i\mathbf{G}_{12}^i \\ \mathbf{W}_{33}^i\mathbf{G}_{13}^i \end{bmatrix}, \quad \mathbf{W}_2^i = \begin{bmatrix} \mathbf{W}_{11}^i\mathbf{G}_{21}^i \\ \mathbf{W}_{22}^i \\ \mathbf{W}_{33}^i\mathbf{G}_{23}^i \end{bmatrix}, \quad \mathbf{W}_3^i = \begin{bmatrix} \mathbf{W}_{11}^i\mathbf{G}_{31}^i \\ \mathbf{W}_{22}^i\mathbf{G}_{32}^i \\ \mathbf{W}_{33}^i \end{bmatrix}$$



DANSE$_Q$ parametrization

DANSE$_Q$ parametrization

$$\mathbf{W}_k^i = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{k1}^i \\ \vdots \\ \mathbf{W}_{NN}^i \mathbf{G}_{kN}^i \end{bmatrix} \quad \text{where } \mathbf{G}_{kq}^i \in \mathbb{C}^{Q \times Q},\ \mathbf{G}_{kk}^i = \mathbf{I}_Q$$

Since $\forall\, k, q \in \mathcal{J},\ \exists\, \mathbf{A}_{kq} \in \mathbb{C}^{Q \times Q} :\ \hat{\mathbf{W}}_k = \hat{\mathbf{W}}_q \mathbf{A}_{kq}$, the optimal MWF's are in the DANSE solution space (set $\mathbf{W}_{kk}^i = \hat{\mathbf{W}}_{kk}$ and $\mathbf{G}_{kq}^i = \mathbf{A}_{kq}$).

Algorithm description

DANSE$_Q$ algorithm: computation of $\mathbf{W}_{kk}^{i+1}$ and $\mathbf{G}_{k,-k}^{i+1}$

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

$$\begin{bmatrix} \mathbf{W}_{kk}^{i+1} \\ \mathbf{G}_{k,-k}^{i+1} \end{bmatrix} = \begin{cases} \left(\mathbf{R}_{\tilde{y}_k\tilde{y}_k}^i\right)^{-1} \mathbf{R}_{\tilde{d}_k\tilde{d}_k}^i [\mathbf{e}_1 \ \ldots \ \mathbf{e}_Q] & \text{if } k = u \\ \begin{bmatrix} \mathbf{W}_{kk}^i \\ \mathbf{G}_{k,-k}^i \end{bmatrix} & \text{if } k \neq u \end{cases}$$

where $\widetilde{\mathbf{y}}_k^i$ and $\widetilde{\mathbf{d}}_k^i$ are defined as earlier (but with Q-channel $\mathbf{z}_k^i$ signals).

Convergence and optimality of DANSE$_Q$

Theorem (Convergence and optimality of DANSE$_Q$)

In case of Q desired speech sources, and if $\mathbf{A}_{k,ref}$ 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 $Q <$ 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 $\mathbf{W}_{kk}^i$ and $\mathbf{G}_{kq}^i$, $\forall\, k, q \in \mathcal{J}$, such that none of the nodes wants to change them.

- Convergence to equilibrium is not proven, but is generally observed in simulations.
- Equilibrium = 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
  $\Rightarrow$ 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])

$$\mathbf{W}_{kk}^{i+1} = (1-\alpha)\mathbf{W}_{kk}^{i} + \alpha \mathbf{W}_{kk}^{\textit{unrelaxed update}}$$

with $0 < \alpha \leq 1$.

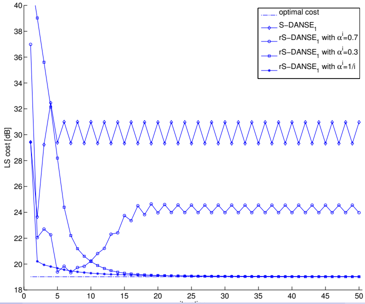
Relaxed simultaneous DANSE (rS-DANSE)

rS-DANSE$_Q$ algorithm: computation of $\mathbf{W}_{kk}^{i+1}$ and $\mathbf{G}_{k,-k}^{i+1}$

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

$$\begin{bmatrix} \mathbf{W}_{kk}^{\text{new}} \\ \mathbf{G}_{k,-k}^{i+1} \end{bmatrix} = \left(\mathbf{R}_{\tilde{y}_k\tilde{y}_k}^{i}\right)^{-1} \mathbf{R}_{\tilde{d}_k\tilde{d}_k}^{i} [\mathbf{e}_1 \ \ldots \ \mathbf{e}_Q]$$

$$\mathbf{W}_{kk}^{i+1} = (1-\alpha)\mathbf{W}_{kk}^{i} + \alpha \mathbf{W}_{kk}^{new}$$



Robustified DANSE (R-DANSE)

- Sometimes ill-conditioned nodes:

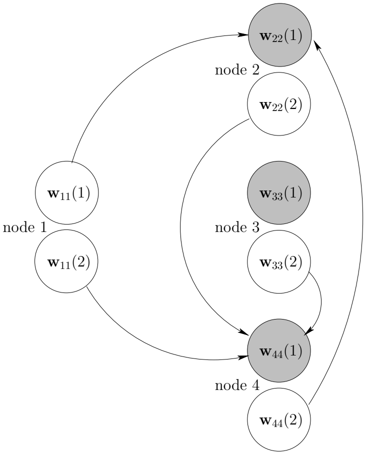
$$a_{k1} \approx 0 \text{ or } \mathbf{A}_{k,\text{ref}} \approx \text{rank deficient}$$

- 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



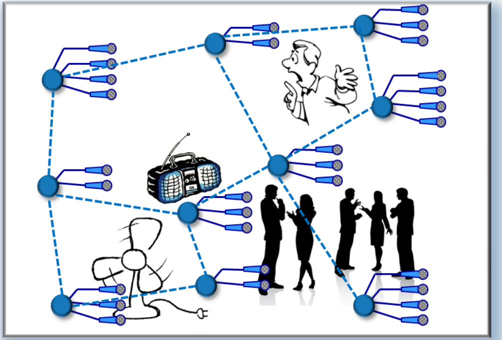
Dependency graph:

- Each column $\mathbf{w}_{kk}^i(m)$ of $\mathbf{W}_{kk}^i$, $\forall\, k \in \mathcal{J}$, $\forall\, m \in \{1, \ldots, Q\}$ is a vertex.
- Note: each $\mathbf{w}_{kk}^i(m)$ corresponds to a particular reference mic
- Draw edge $\mathbf{w}_{kk}^i(m) \rightarrow \mathbf{w}_{qq}^i(n)$ if update of $\mathbf{w}_{kk}^i(m)$ is based on the reference signal $z_q^i(n)$ instead of a local microphone.

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

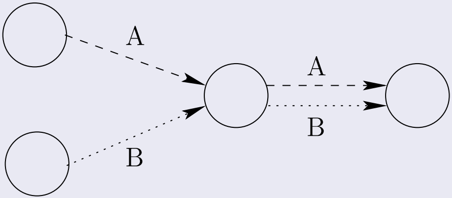
Multi-hop WASNs

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



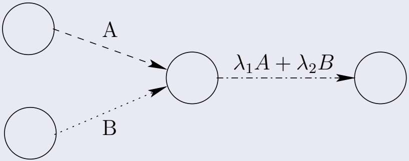
Passing on information

The relay case



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

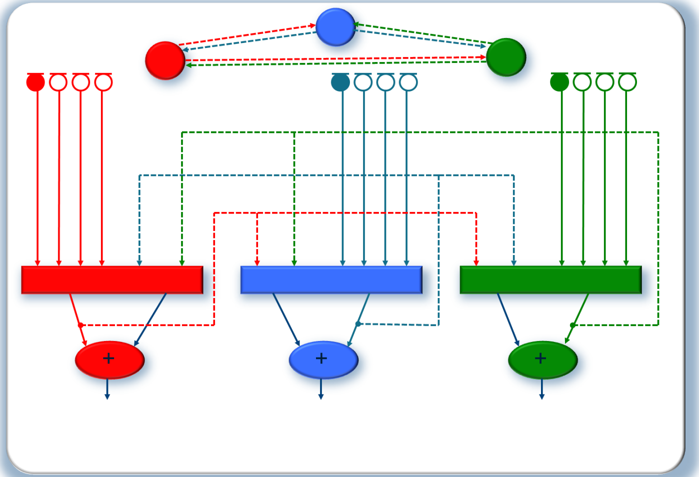
Filter-and-sum combination of inputs



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

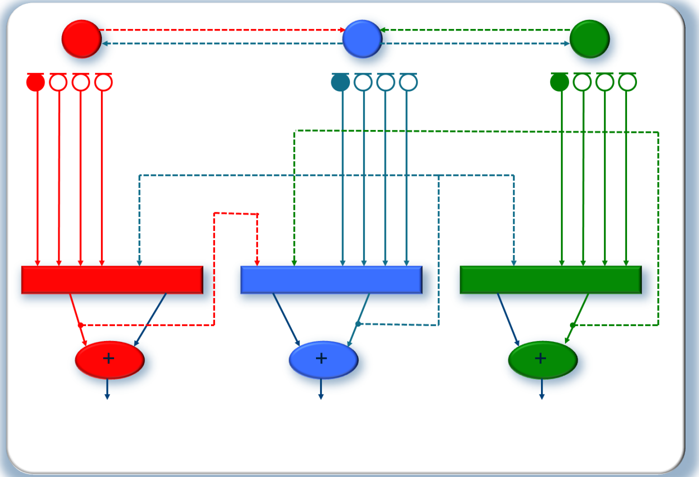
First attempt

Fully-connected DANSE:



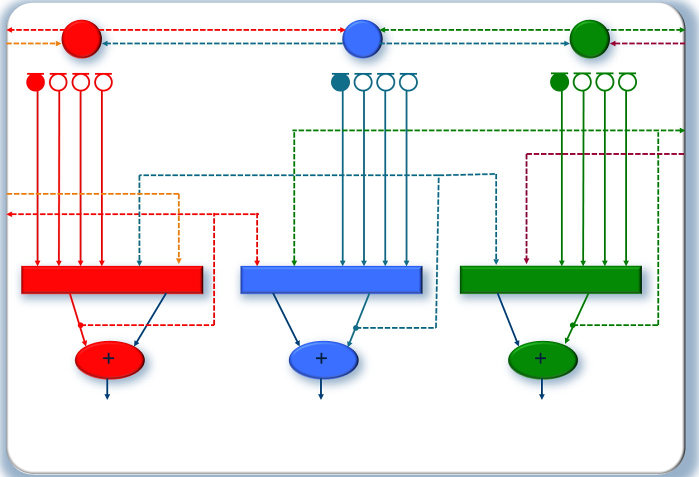
First attempt

Disconnect red and green node...



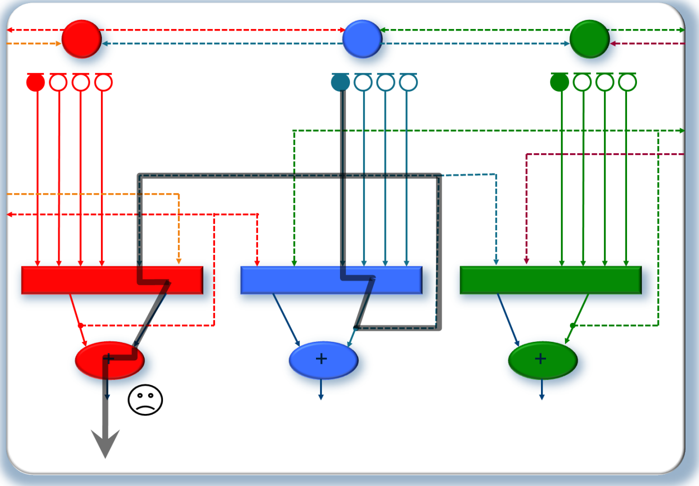
First attempt

... and add new neighbors instead:



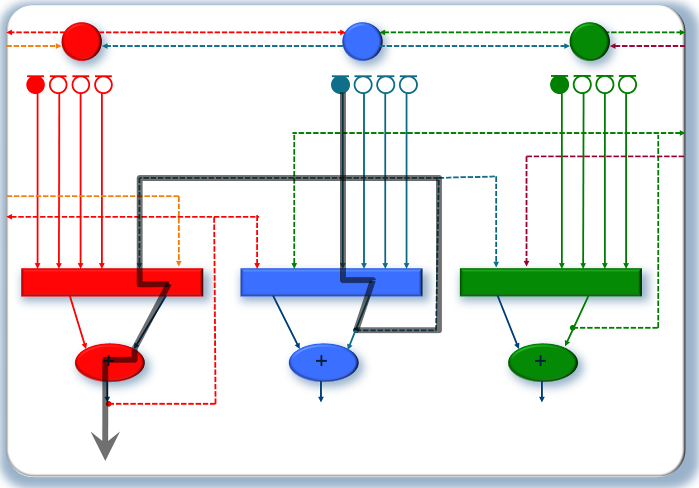
First attempt

Blue node's data is blocked and does not travel beyond red node:



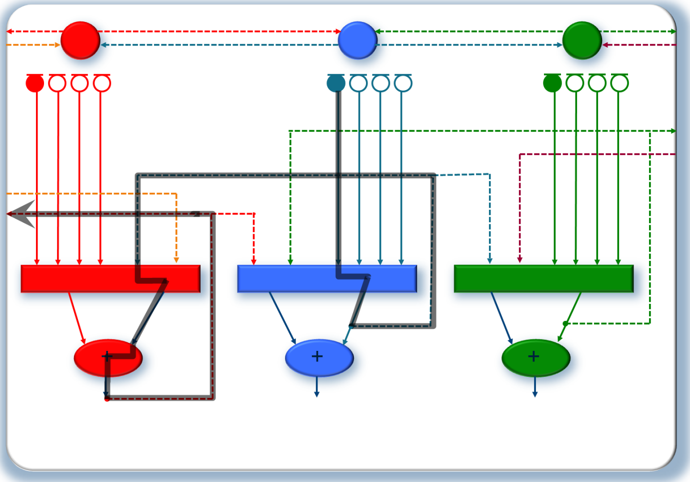
First attempt

Change definition of transmitted signal $\mathbf{z}_k^i$ ('wild guess'):



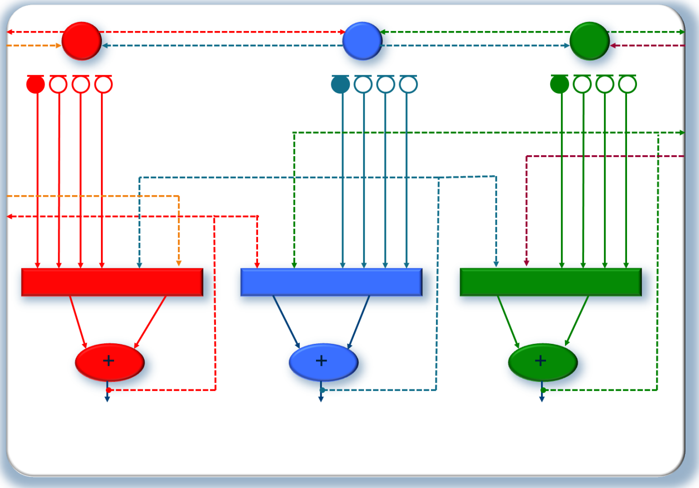
First attempt

Data from blue node travels beyond single-hop region:

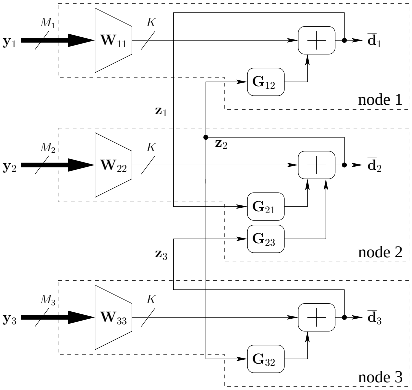


First attempt

Apply similar idea in all nodes:



First attempt



Will this 'wild guess' work???

- $\mathcal{N}_k$= neighbours of k (k excl.)
- Implicit definition of $\mathbf{z}_k^i$:
  $\mathbf{z}_k^i = \mathbf{W}_{kk}^{i\,H}\mathbf{y}_k + \sum_{q\in\mathcal{N}_k} \mathbf{G}_{kq}^{i\,H}\mathbf{z}_q^i$

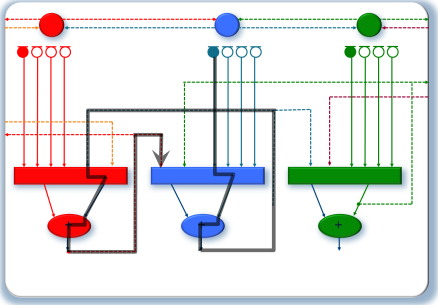
Problem 1: acausality in data flow

Deadlock: nodes wait for each other's **z**-signals

First attempt

Problem 2: feedback

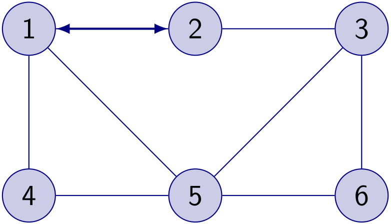
- Feedback path considerably changes algorithm dynamics
- Centralized MWF's are not in solution space (provable)



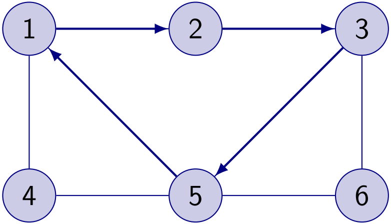
How to get rid of this feedback and causality problem?

2 types of feedback

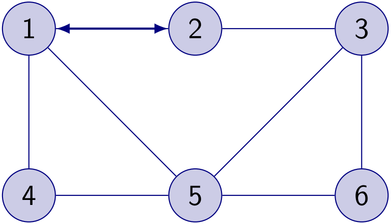
Direct feedback



Indirect feedback



Eliminating direct feedback



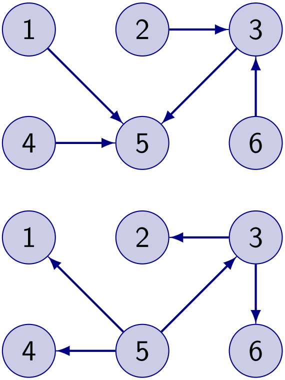
Direct feedback:

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

$$\mathbf{z}_{kq}^{i} = \mathbf{W}_{kk}^{i\,H}\mathbf{y}_k + \sum_{l \in \mathcal{N}_k \backslash \{q\}} \mathbf{G}_{kl}^{i\,H}\mathbf{z}_{lk}^{i}$$

- 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

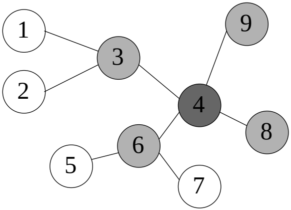


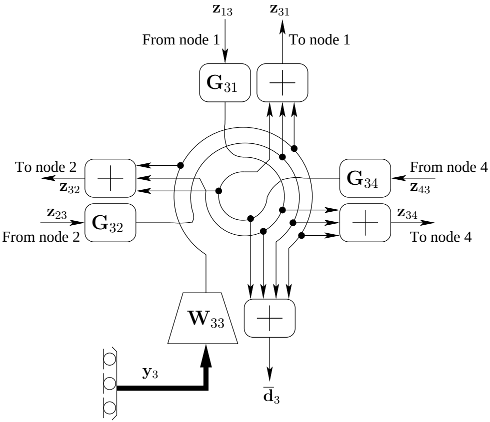
Indirect feedback:

- Prune to tree topology
- In combination with TFC: all feedback eliminated
- Definition of $\mathbf{z}_{kq}^{i}$'s can be resolved:
  - Start at leaf nodes ($|\mathcal{N}_k| = 1$)
  - Leaf node k: $\mathbf{z}_{kq}^{i} = \mathbf{W}_{kk}^{i\,H}\mathbf{y}_k$, 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 leafs

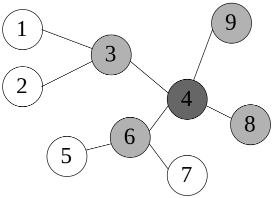
Data-driven signal exchange

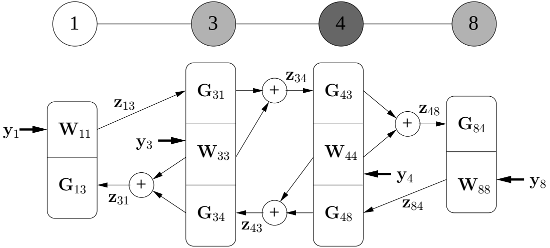
- Data-driven paradigm: each block 'fires' if all of its inputs are available ⇒ no global coordination needed to organize data flow
- Fusion and diffusion flow emerge automatically





Parametrization: example





$$\mathbf{W}_1^i = \begin{bmatrix} \mathbf{W}_{11}^i \\ * \\ \mathbf{W}_{33}^i \mathbf{G}_{13}^i \\ \mathbf{W}_{44}^i \mathbf{G}_{34}^i \mathbf{G}_{13}^i \\ * \\ * \\ * \\ \mathbf{W}_{88}^i \mathbf{G}_{48}^i \mathbf{G}_{34}^i \mathbf{G}_{13}^i \end{bmatrix}$$

$$\mathbf{W}_4^i = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{31}^i \mathbf{G}_{43}^i \\ * \\ \mathbf{W}_{33}^i \mathbf{G}_{43}^i \\ \mathbf{W}_{44}^i \\ * \\ * \\ * \\ \mathbf{W}_{88}^i \mathbf{G}_{48}^i \end{bmatrix}$$

General parametrization of Tree-DANSE (T-DANSE)

General parametrization of T-DANSE

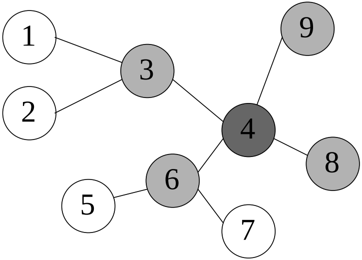
$$\mathbf{W}_k^i = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{k \leftarrow 1}^i \\ \vdots \\ \mathbf{W}_{NN}^i \mathbf{G}_{k \leftarrow N}^i \end{bmatrix}$$

- $\mathbf{G}_{p_1 \leftarrow p_t}^i = \mathbf{G}_{p_{t-1} p_t}^i \mathbf{G}_{p_{t-2} p_{t-1}}^i \ldots \mathbf{G}_{p_2 p_3}^i \mathbf{G}_{p_1 p_2}^i$ 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: $\mathbf{G}_{k \leftarrow k}^i = \mathbf{I}_Q$

Compare with fully-connected DANSE:

$$\mathbf{W}_k^i = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{k1}^i \\ \vdots \\ \mathbf{W}_{NN}^i \mathbf{G}_{kN}^i \end{bmatrix}$$

Parametrization: example



Complete parametrization of network-wide filter $\mathbf{W}_4^i$:

$$\mathbf{W}_4^i = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{4\leftarrow 1}^i \\ \mathbf{W}_{22}^i \mathbf{G}_{4\leftarrow 2}^i \\ \mathbf{W}_{33}^i \mathbf{G}_{4\leftarrow 3}^i \\ \mathbf{W}_{44}^i \\ \mathbf{W}_{55}^i \mathbf{G}_{4\leftarrow 5}^i \\ \mathbf{W}_{66}^i \mathbf{G}_{4\leftarrow 6}^i \\ \mathbf{W}_{77}^i \mathbf{G}_{4\leftarrow 7}^i \\ \mathbf{W}_{88}^i \mathbf{G}_{4\leftarrow 8}^i \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11}^i \mathbf{G}_{31}^i \mathbf{G}_{43}^i \\ \mathbf{W}_{22}^i \mathbf{G}_{31}^i \mathbf{G}_{43}^i \\ \mathbf{W}_{33}^i \mathbf{G}_{43}^i \\ \mathbf{W}_{44}^i \\ \mathbf{W}_{55}^i \mathbf{G}_{65}^i \mathbf{G}_{46}^i \\ \mathbf{W}_{66}^i \mathbf{G}_{46}^i \\ \mathbf{W}_{77}^i \mathbf{G}_{67}^i \mathbf{G}_{46}^i \\ \mathbf{W}_{88}^i \mathbf{G}_{48}^i \end{bmatrix}$$

Centralized MWF in T-DANSE solution space?

Theorem

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

Proof:

- Reminder: $\forall\, k, q \in \mathcal{J}$: $\hat{\mathbf{W}}_k = \hat{\mathbf{W}}_q \mathbf{A}_{kq}$, where

$$\mathbf{A}_{kq} = \mathbf{A}_{q,\text{ref}}^{-H} \cdot \mathbf{A}_{k,\text{ref}}^{H}$$

- Therefore: $\forall\, k, q, n \in \mathcal{J}$: $\mathbf{A}_{nq}\mathbf{A}_{kn} = \mathbf{A}_{kq}$
- Set $\mathbf{G}_{mn}^{i} = \mathbf{A}_{mn}$, then

$$\begin{aligned}\mathbf{G}_{k\leftarrow q}^{i} &= \mathbf{A}_{p_{t-1}q} \cdot \mathbf{A}_{p_{t-2}p_{t-1}} \ldots \mathbf{A}_{p_2 p_3} \cdot \mathbf{A}_{k p_2} \\ &= \mathbf{A}_{kq}\end{aligned}$$

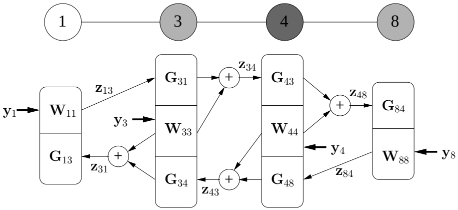
  where $P_{k\leftarrow q} = (q, p_{t-1}, p_{t-2}, \ldots, p_3, p_2, k)$
- Hence, set $\mathbf{W}_{kk}^{i} = \hat{\mathbf{W}}_{kk}$ and $\mathbf{G}_{mn}^{i} = \mathbf{A}_{mn}$, then $\mathbf{W}_{k}^{i} = \hat{\mathbf{W}}_{k}$, Q.E.D.

T-DANSE updating procedure

- Let $\mathbf{z}^i_{\rightarrow k} = [\mathbf{z}^{i\,T}_{n_1} \; \ldots \; \mathbf{z}^{i\,T}_{n_{N_k}}]^T$.
- Node k sets internal fusion rules

$$\mathbf{W}^i_{kk} \text{ and } \mathbf{G}^i_{k,-k} = \left[\mathbf{G}^{i\,T}_{n_1} \; \ldots \; \mathbf{G}^{i\,T}_{n_{N_k}}\right]^T$$

with $n_j \in \mathcal{N}_k$ and $N_k = |\mathcal{N}_k|$.



T-DANSE updating procedure

T-DANSE$_Q$ algorithm: computation of $\mathbf{W}_{kk}^{i+1}$ and $\mathbf{G}_{k,-k}^{i+1}$

- If $k \neq u$, then $\mathbf{W}_{kk}^{i+1} = \mathbf{W}_{kk}^{i}$ and $\mathbf{G}_{k,-k}^{i+1} = \mathbf{G}_{k,-k}^{i}$
- If $k = u$:

$$\begin{bmatrix} \mathbf{W}_{kk}^{i+1} \\ \mathbf{G}_{k,-k}^{i+1} \end{bmatrix} = \underset{\mathbf{W}_{kk}, \mathbf{G}_{k,-k}}{\arg\min} \, E\left\{ \left\| \mathbf{d}_k - \left[\mathbf{W}_{kk}^H \;\; \mathbf{G}_{k,-k}^H \right] \begin{bmatrix} \mathbf{y}_k \\ \mathbf{z}_{\rightarrow k}^i \end{bmatrix} \right\|^2 \right\}$$

$$= \left(\mathbf{R}_{\tilde{y}_k \tilde{y}_k}^i \right)^{-1} \mathbf{R}_{\tilde{d}_k \tilde{d}_k}^i [\mathbf{e}_1 \; \dots \; \mathbf{e}_Q]$$

where $\widetilde{\mathbf{y}}_k^i = [\mathbf{y}_k^T \; \mathbf{z}_{\rightarrow k}^{i\,T}]^T$, and similarly for $\widetilde{\mathbf{d}}_k^i$.

- Identical to fully-connected DANSE updates (but less input signals per node)
- Note: sequential updates (only one node updates in each iteration)

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 $\mathbf{A}_{k,ref}$ is full rank, $\forall\, k \in \mathcal{J}$, and if the node-per-node updating order of T-DANSE is defined by a path through the network that visits all nodes, then $\lim_{i\to\infty} \mathbf{W}_k^i = \hat{\mathbf{W}}_k$, $\forall\, k \in \mathcal{J}$.

- 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)

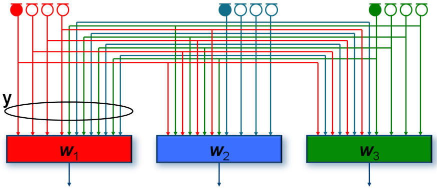
LCMV beamforming revisited

Centralized node-specific LCMV BF at node k:

$$\hat{\mathbf{w}}_k = \arg\min_{\mathbf{w}_k} \left(\mathbf{w}_k^H \mathbf{R}_{yy} \mathbf{w}_k, \text{ s.t. } \mathbf{A}^H \mathbf{w}_k = \mathbf{f}_k \right)$$

$$= \mathbf{R}_{yy}^{-1} \mathbf{A} \left(\mathbf{A}^H \mathbf{R}_{yy}^{-1} \mathbf{A} \right)^{-1} \mathbf{f}_k$$

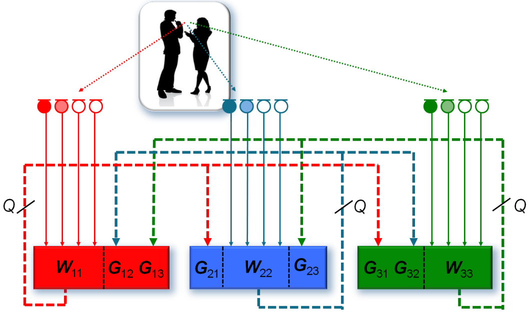
- $\mathbf{A}$ $M \times Q$ steering matrix from Q 'relevant' sources to M microphones
- $\mathbf{f}_k$ node-specific response for each of the Q sources
- Relevant sources may also contain interferers!



PS: Will assume in sequel that **A** is known. For unknown **A**, refer to [Markovich et al., 2009] or [Bertrand and Moonen, 2012a]

Linearly-constrained DANSE (LC-DANSE)

- DANSE ↔ (SDW-)MWF
- LC-DANSE ↔ LCMV
- Similar idea, similar block scheme



Note: Q is # constraints

Linearly-constrained DANSE (LC-DANSE)

- $\hat{\mathbf{w}}_k = \mathbf{R}_{yy}^{-1}\mathbf{A}\left(\mathbf{A}^H\mathbf{R}_{yy}^{-1}\mathbf{A}\right)^{-1}\mathbf{f}_k$
  $\Rightarrow$ joint Q-dim subspace: $\hat{\mathbf{w}}_k = \mathbf{W}\cdot\mathbf{f}_k,\ \forall\, k \in \mathcal{J}$.
- Add $Q-1$ auxiliary LCMV-problems:

$$\hat{\mathbf{W}}_k = \underset{\mathbf{W}_k}{\arg\min}\ \left(\mathrm{Tr}\left(\mathbf{W}_k^H\mathbf{R}_{yy}\mathbf{W}_k\right),\ \text{s.t. } \mathbf{A}^H\mathbf{W}_k = \mathbf{F}_k\right)$$
$$= \mathbf{R}_{yy}^{-1}\mathbf{A}\left(\mathbf{A}^H\mathbf{R}_{yy}^{-1}\mathbf{A}\right)^{-1}\mathbf{F}_k$$

  with $\mathbf{F}_k$ a $Q\times Q$ matrix of full rank, with $\mathbf{f}_k$ in first column.
- $\forall\, k, q \in \mathcal{J}$: $\hat{\mathbf{W}}_k = \hat{\mathbf{W}}_q\mathbf{A}_{kq}$ with

$$\mathbf{A}_{kq} = \mathbf{F}_q^{-1}\mathbf{F}_k$$

Conclusion: Centralized LCMV solutions are in (LC-)DANSE solution space! (set $\mathbf{W}_{kk}^i = \hat{\mathbf{W}}_{kk}$ and $\mathbf{G}_{kq}^i = \mathbf{A}_{kq}$)

Linearly-constrained DANSE (LC-DANSE)

Match constraints with compressed signals:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} \leftrightarrow \mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_N \end{bmatrix}$$

$$\widetilde{\mathbf{y}}_k^i = \begin{bmatrix} \mathbf{y}_k \\ \mathbf{z}_{-k}^i \end{bmatrix} \leftrightarrow \widetilde{\mathbf{A}}_k^i = \begin{bmatrix} \mathbf{A}_k \\ \mathbf{C}_{-k}^i \end{bmatrix}$$

$$\mathbf{z}_{-k}^i = \begin{bmatrix} \mathbf{z}_1^i \\ \vdots \\ \mathbf{z}_{k-1}^i \\ \mathbf{z}_{k+1}^i \\ \vdots \\ \mathbf{z}_N^i \end{bmatrix} \leftrightarrow \mathbf{C}_{-k}^i = \begin{bmatrix} \mathbf{C}_1^i \\ \vdots \\ \mathbf{C}_{k-1}^i \\ \mathbf{C}_{k+1}^i \\ \vdots \\ \mathbf{C}_N^i \end{bmatrix}$$

$$\mathbf{z}_k^i = \mathbf{W}_{kk}^{i\,H} \mathbf{y}_k \leftrightarrow \mathbf{C}_k^i = \mathbf{W}_{kk}^{i\,H} \mathbf{A}_k$$

LC-DANSE Algorithm description

LC-DANSE$_Q$ algorithm: computation of $\mathbf{W}_{kk}^{i+1}$ and $\mathbf{G}_{k,-k}^{i+1}$

Update at node k:

$$\begin{bmatrix} \mathbf{W}_{kk}^{i+1} \\ \mathbf{G}_{k,-k}^{i+1} \end{bmatrix} = \begin{cases} \left(\mathbf{R}_{\tilde{y}_k\tilde{y}_k}^i\right)^{-1} \widetilde{\mathbf{A}}_k^i \left(\widetilde{\mathbf{A}}_k^{i\,H} \left(\mathbf{R}_{\tilde{y}_k\tilde{y}_k}^i\right)^{-1} \widetilde{\mathbf{A}}_k^i\right)^{-1} \mathbf{F}_k & \text{if } k = u \\ \begin{bmatrix} \mathbf{W}_{kk}^i \\ \mathbf{G}_{k,-k}^i \end{bmatrix} & \text{if } k \neq u \end{cases}$$

Note: computation of $\widetilde{\mathbf{A}}_k^i$ requires exchange of $\mathbf{W}_{kk}^i$'s. However, filter coefficients are typically frozen for some time (2-3s), hence negligible compared to data rate of $\mathbf{z}_k^i$'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 $\mathbf{f}_k$ 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)

References and Further Reading I

Baraniuk, R. (2011).
More is less: Signal processing and the data deluge.
Science, 331:717–719.

Bertrand, A. and Moonen, M. (2009).
Robust distributed noise reduction in hearing aids with external acoustic sensor nodes.
EURASIP Journal on Advances in Signal Processing, 2009.

Bertrand, A. and Moonen, M. (2010a).
Distributed adaptive node-specific signal estimation in fully connected sensor networks – part I: sequential node updating.
IEEE Transactions on Signal Processing, 58:5277–5291.

Bertrand, A. and Moonen, M. (2010b).
Distributed adaptive node-specific signal estimation in fully connected sensor networks – part II: simultaneous & asynchronous node updating.
IEEE Transactions on Signal Processing, 58:5292–5306.

Bertrand, A. and Moonen, M. (2011).
Distributed adaptive estimation of node-specific signals in wireless sensor networks with a tree topology.
IEEE Trans. Signal Processing, 59(5):2196–2210.

Bertrand, A. and Moonen, M. (2012a).
Distributed node-specific LCMV beamforming in wireless sensor networks.
IEEE Transactions on Signal Processing, 60:233–246.

Bertrand, A. and Moonen, M. (2012b).
Distributed signal estimation in sensor networks where nodes have different interests.
Signal Processing, 92(7):1679 –1690.

References and Further Reading II

Bertrand, A. and Moonen, M. (2013).
Distributed LCMV beamforming in a wireless sensor network with single-channel per-node signal transmission.
IEEE Transactions on Signal Processing, 61:3447–3459.

Doclo, S. and Moonen, M. (2002).
GSVD-based optimal filtering for single and multimicrophone speech enhancement.
IEEE Transactions on Signal Processing, 50(9):2230 – 2244.

Doclo, S., van den Bogaert, T., Moonen, M., and Wouters, J. (2009).
Reduced-bandwidth and distributed MWF-based noise reduction algorithms for binaural hearing aids.
IEEE Trans. Audio, Speech and Language Processing, 17:38–51.

Heusdens, R., Zhang, G., Hendriks, R. C., Zeng, Y., and Kleijn, W. B. (2012).
Distributed MVDR beamforming for (wireless) microphone networks using message passing.
In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*.

Himawan, I., McCowan, I., and Sridharan, S. (2011).
Clustered blind beamforming from ad-hoc microphone arrays.
IEEE Trans. Audio, Speech, and Language Processing, 19(4):661 –676.

Hioka, Y. and Kleijn, W. B. (2011).
Distributed blind source separation with an application to audio signals.
In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 233 –236.

Lawin-Ore, T. C. and Doclo, S. (2011).
Analysis of rate constraints for MWF-based noise reduction in acoustic sensor networks.
In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages pp. 269–272, Prague, Czech Republic.

References and Further Reading III

Markovich, S., Gannot, S., and Cohen, I. (2009).
Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals.
IEEE Trans. Audio, Speech, and Language Processing, 17(6):1071–1086.

Markovich-Golan, S., Gannot, S., and Cohen, I. (2010).
A reduced bandwidth binaural MVDR beamformer.
In *Proc. of the International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Tel-Aviv, Israel.

Markovich-Golan, S., Gannot, S., and Cohen, I. (2013).
Distributed multiple constraints generalized sidelobe canceler for fully connected wireless acoustic sensor networks.
IEEE Transactions on Audio, Speech, and Language Processing, 21(2):343–356.

Olfati-Saber, R., Fax, J., and Murray, R. (2007).
Consensus and cooperation in networked multi-agent systems.
Proceedings of the IEEE, 95(1):215–233.

Sayed, A., Tu, S.-Y., Chen, J., Zhao, X., and Towfic, Z. (2013).
Diffusion strategies for adaptation and learning over networks: an examination of distributed strategies and network behavior.
IEEE Signal Processing Magazine, 30(3):155–171.

Shah, D. (2009).
Gossip algorithms.
Foundations and Trends in Networking, 3(1):1–125.

Srinivasan, S. and Den Brinker, A. C. (Article ID 257197, 14 pages, 2009.).
Rate-constrained beamforming in binaural hearing aids.
EURASIP Journal on Advances in Signal Processing, 2009.

References and Further Reading IV

Szurley, J., Bertrand, A., and Moonen, M. (2013).
Improved tracking performance for distributed node-specific signal enhancement in wireless acoustic sensor networks.
In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal processing (ICASSP)*, pages 336–340, Vancouver, Canada.

Zeng, Y. and Hendriks, R. (2012).
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.