

Microphone Array project in MSR: approach and results

Ivan Tashev

Microsoft Research

June 2004



Agenda

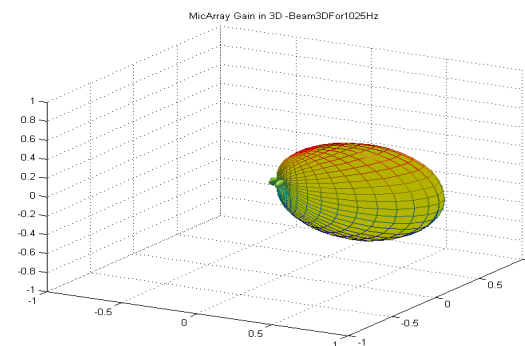
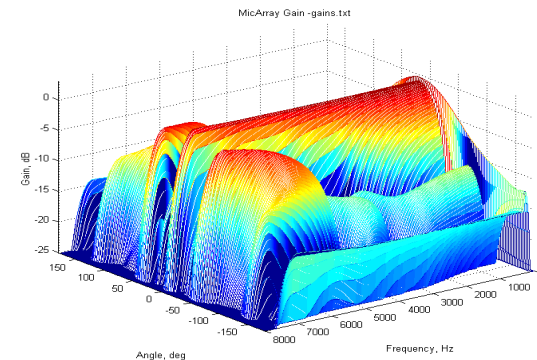
- Microphone Array project
- Beamformer design algorithm
- Implementation and hardware designs
- Demo

Motivation

- PCs today have pretty bad “ears”; audio captured or recorded from PCs sounds terrible (especially with laptops) – unless a good headset is used.
- Sound will play more and more important role in human-computer interaction, especially in devices without keyboard (tablets, handhelds)
- Increases using computers in collaboration and communication
- Users don't like headsets or other tethered microphones, especially in a video call.
- Existing wireless solutions do not provide enough good sound quality, you have to wear them

Microphone array project: goals

- Far goal: sound capturing quality for untethered user the same as with close-up microphone
- Near goal: Create technology for OS support and devices so cheap to become commodity on the market
- Beamforming is ability to make the microphone array to listen to given location, suppressing the signals coming from other locations



Target scenarios

- **Real-time communications**
 - Providing good sound capturing for Windows Messenger, MSN Messenger, other applications built on top of the RTC stack
 - New applications for VoIP and enhanced telephony
- **Collaboration and groupware**
 - High quality sound from meeting rooms for recording and broadcasting purposes (OneNote)
 - Voice messaging
- **Speech recognition**
 - Voice commands for Tablet PCs and handhelds
 - Voice control and dictation for PCs and laptops

Problems

- “Wear nothing” approach requires using separate microphones: connected or integrated
- These microphones deliver poor sound capturing quality:
 - Too much ambient and electronic noises
 - Reverberation and reflections – poor user experience and bad speech recognition results
- Noise suppression and de-reverberation are difficult with a single microphone channel

The solution

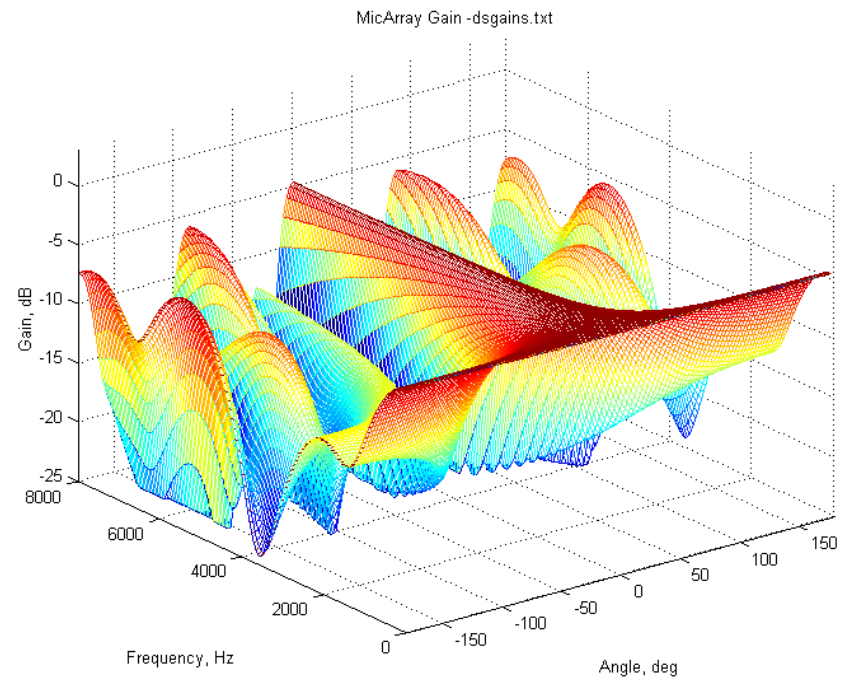
- Using microphone arrays for capturing the sound
 - A set of close positioned microphones
 - Synchronous capturing of the signals
- Microphone Array acts as an acoustic antenna
 - This is called spatial filtering or beamforming
 - Listens only to the direction of the speaker
 - Reduces the noises from other directions
 - Reduces the reverberation

Beamforming: known approaches

- Fixed beam formation
 - Delay and sum – most intuitive, irregular beam shape
 - Parametric solutions: very complex
 - Fast real-time execution
- Adaptive beamformers
 - Generalized side lobe canceller
 - Vary with the target criteria (MVDR, etc.)
 - Slow adaptation, CPU time intensive

Beamforming: known approaches

- Fixed beam formation
 - Delay and sum – most intuitive, irregular beam shape
 - Parametric solutions:
 - Fast real-time execution
- Adaptive beamforming
 - Generalized side lobe control
 - Vary with the target characteristics
 - Slow adaptation, CPI



Beamforming: known approaches

- Fixed beam formation
 - Delay and sum – most intuitive, irregular beam shape
 - Parametric solutions: very complex
 - Fast real-time execution
- Adaptive beamformers
 - Generalized side lobe canceller
 - Vary with the target criteria (MVDR, etc.)
 - Slow adaptation, CPU time intensive

Beamformer: canonical form

- Canonical form of the beamformer:

$$Y(f) = \sum_{i=0}^{M-1} W(f, i) X_i(f)$$

M – number of microphones

$X_i(f)$ – spectrum of i -th channel

$W(f, i)$ – weight coefficients matrix

$Y(f)$ – output signal

- For each weight matrix we have corresponding shape of the beam $B(\varphi, \theta, f)$ - the array gain as function of direction
- The goal is to find weight matrix to satisfy certain criteria

Beamformer: Array parameters

- Noise = ambient + non-correlated + correlated (jammers and reverberation)

- Ambient noise gain

$$20 \log \int_0^{\frac{f_s}{2}} \int_0^{2\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} N(f) B(\varphi, \theta, f) d\theta d\varphi df$$

- Non-correlated noise:

$$20 \log \left[\int_0^{\frac{f_s}{2}} \sqrt{\sum_{i=0}^{M-1} W(f, i)^2} df \right]$$

- Correlated (from given direction):

$$20 \log \frac{\int_0^{\frac{f_s}{2}} S(f) B(\varphi_s, \theta_s, f) df}{\int_0^{\frac{f_s}{2}} J(f) B(\varphi_j, \theta_j, f) df}$$

- The total noise gain is the combination of the first two

Weights calculation

- Weights calculation as optimization process
- Minimization criterion: the total noise gain
- Multidimensional optimization
 - Slow, especially in real time (adaptive beamformers)
 - Can't follow the changes
- Multimodal $2M$ dimensional hypersurface – local minima
- In all cases the starting point is critical

Weights calculation (2)

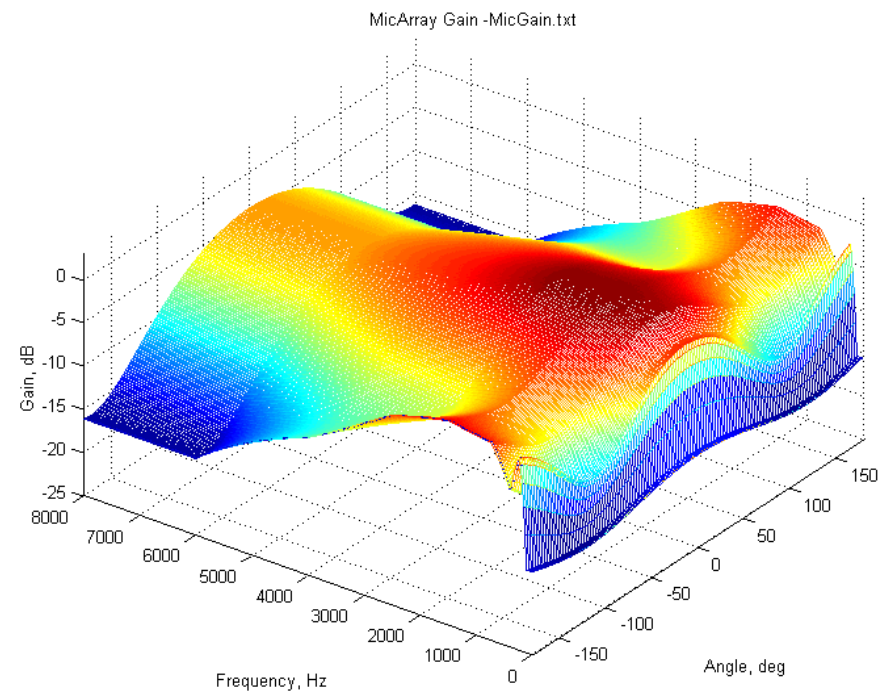
- Our approach:
 - Deterministic beam formation
 - Use as much prior info as possible
 - Do your homework: calculate the weights in advance
 - Calculate set of beams to cover the work volume
 - Fast real-time engine: switches the beams on the fly

Beamformer: Prior Info

- Prerequisites:
 - Microphone array geometry – microphones coordinates and orientation
 - Directivity response of the microphones $U_m(f, \mathbf{c})$
 - Hardware noise model $N_j(f)$
 - Ambient noise model $N_A(f)$

Beamformer: Prior Info

- Prerequisites:
 - Microphone array geometry – microphones coordinates and orientation
 - Directivity response of
 - Hardware noise model
 - Ambient noise model /

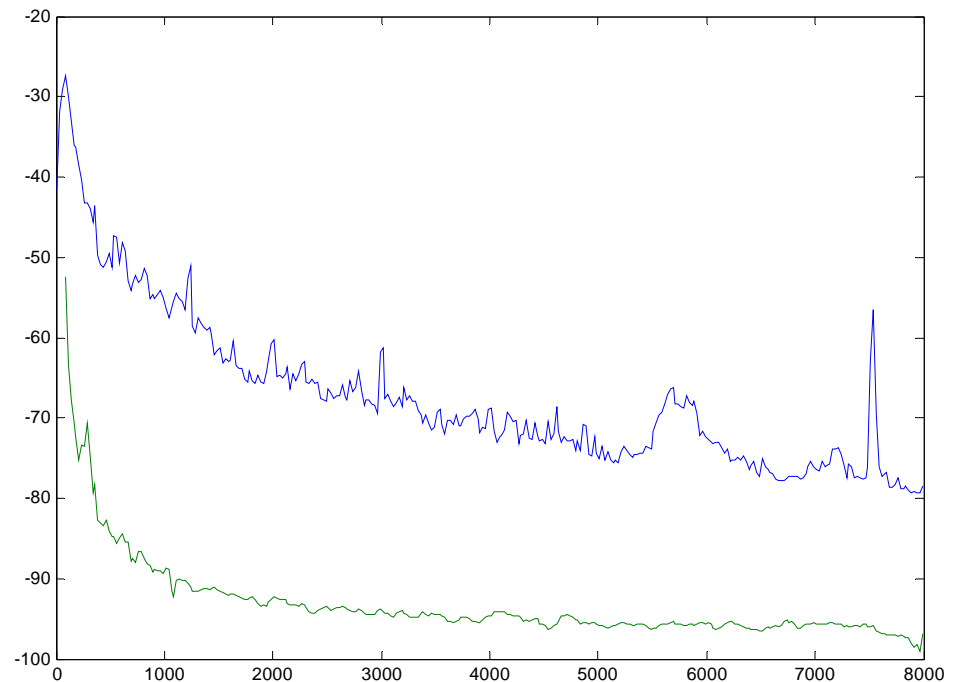


Beamformer: Prior Info

- Prerequisites:
 - Microphone array geometry – microphones coordinates and orientation
 - Directivity response of the microphones $U_m(f, \mathbf{c})$
 - Hardware noise model $N_j(f)$
 - Ambient noise model $N_A(f)$

Beamformer: Prior Info

- Prerequisites:
 - Microphone array geometry – microphones coordinates and orientation
 - Directivity response
 - Hardware noise model
 - Ambient noise model



Beamformer: Prior Info

- Prerequisites:
 - Microphone array geometry – microphones coordinates and orientation
 - Directivity response of the microphones $U_m(f, \mathbf{c})$
 - Hardware noise model $N_j(f)$
 - Ambient noise model $N_A(f)$

Pattern synthesis

- Design in the beamspace
 - Define the target beam shape:
- $$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
 - Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

Pattern synthesis

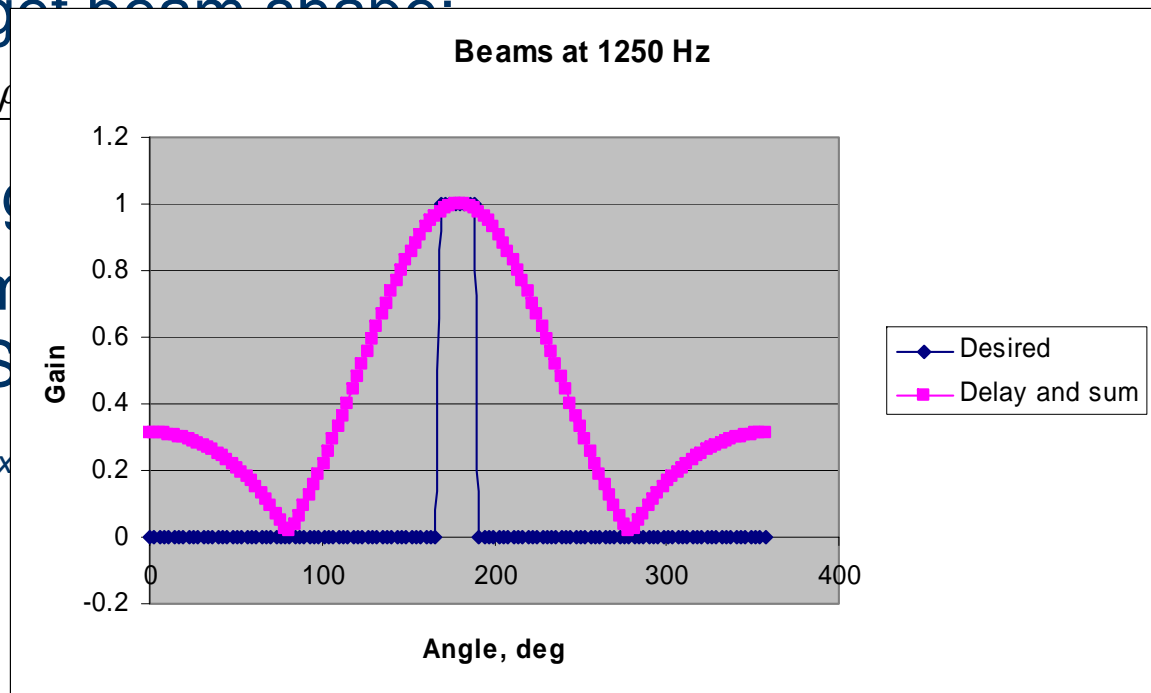
- Design in the beamspace
- Define the target beam shape:

$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho)}{\dots}\right)$$

- Define the weights
- Combine the main lobe with the side lobes
- weighted MMS

$$T_{1 \times L} = V_{1 \times L} D_{M \times M}$$

- Do the design



Pattern synthesis

- Design in the beamspace
 - Define the target beam shape:
- $$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
 - Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

Pattern synthesis

- Design in the beamspace

- Define the target

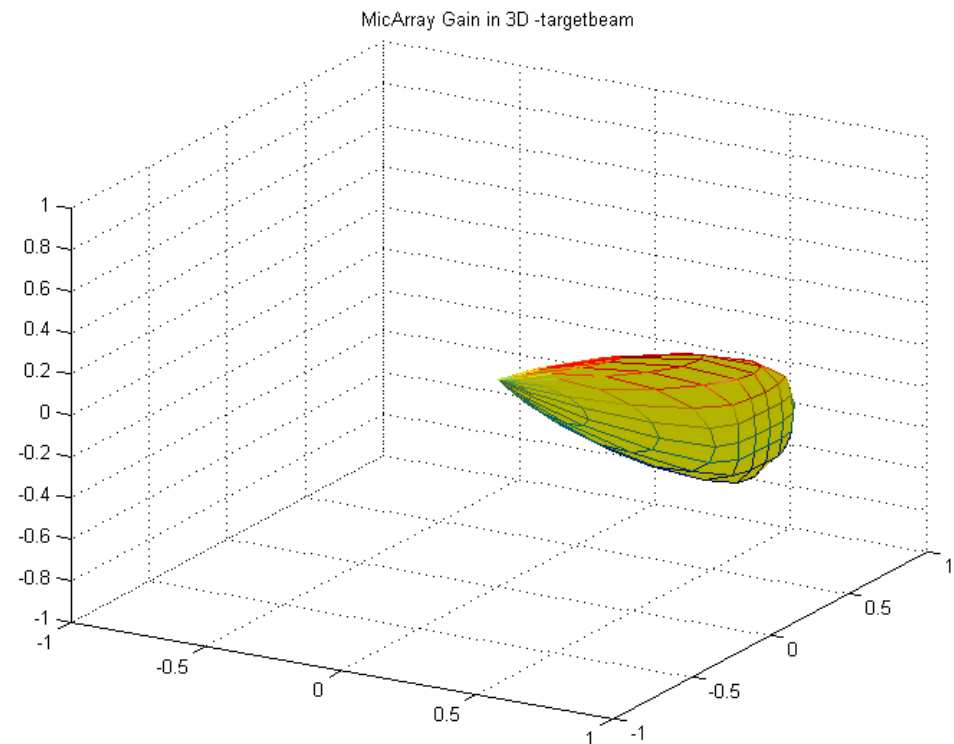
$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right)$$

- Define the weigh

- Combine the mic using weighted \mathbf{M}

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times M}$$

- Do the design in



Pattern synthesis

- Design in the beamspace
- Define the target beam shape:
$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
- Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

Pattern synthesis

- Design in the beamspace

- Define the target

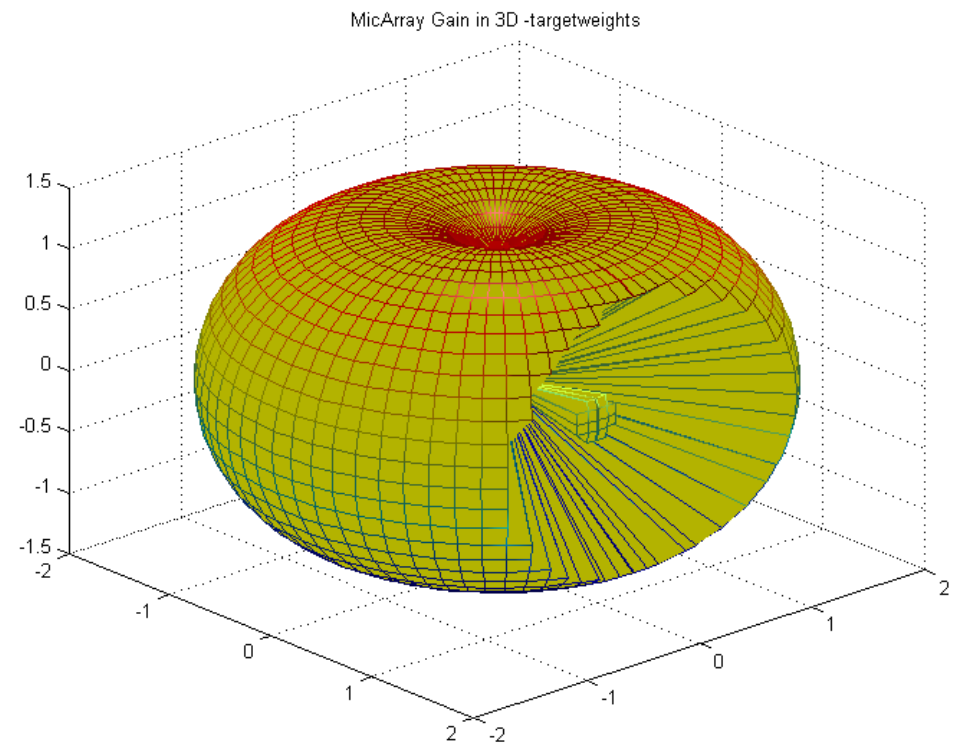
$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right)$$

- Define the weigh

- Combine the mic using weighted \mathbf{M}

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times M}$$

- Do the design in



Pattern synthesis

- Design in the beamspace
 - Define the target beam shape:
- $$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
 - Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

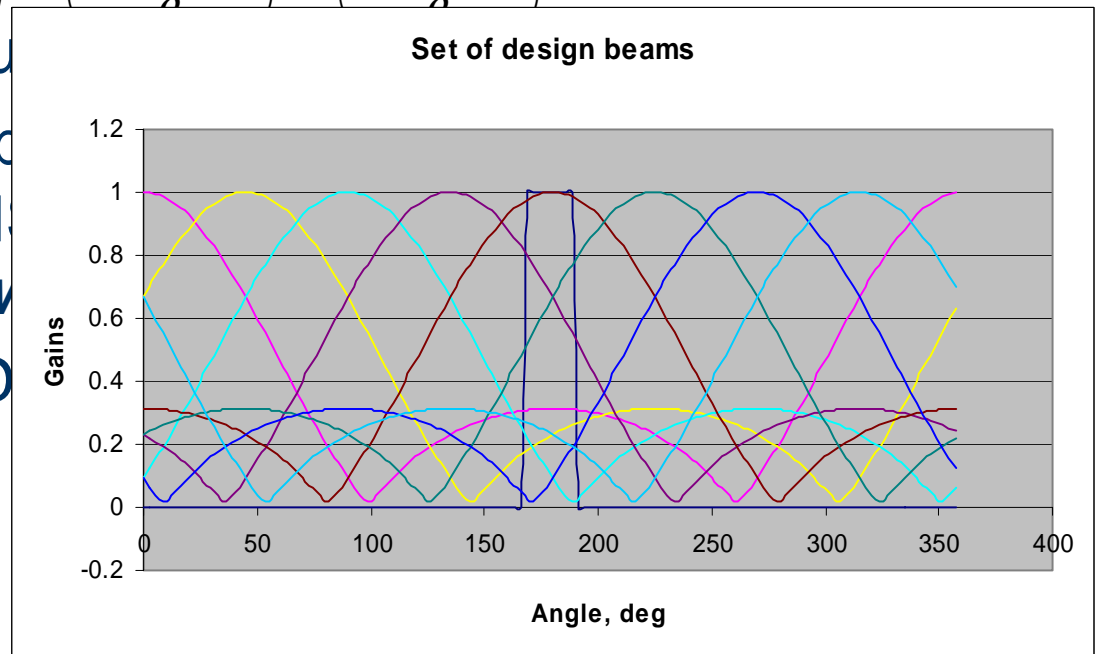
Pattern synthesis

- Design in the beamspace
- Define the target beam shape:

$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$

- Define the weight function
- Combine the microarray elements using weighted MMSE
- Do the design in 3D

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{V}_{M \times L}$$



Pattern synthesis

- Design in the beamspace
- Define the target beam shape:
$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
- Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

Pattern synthesis

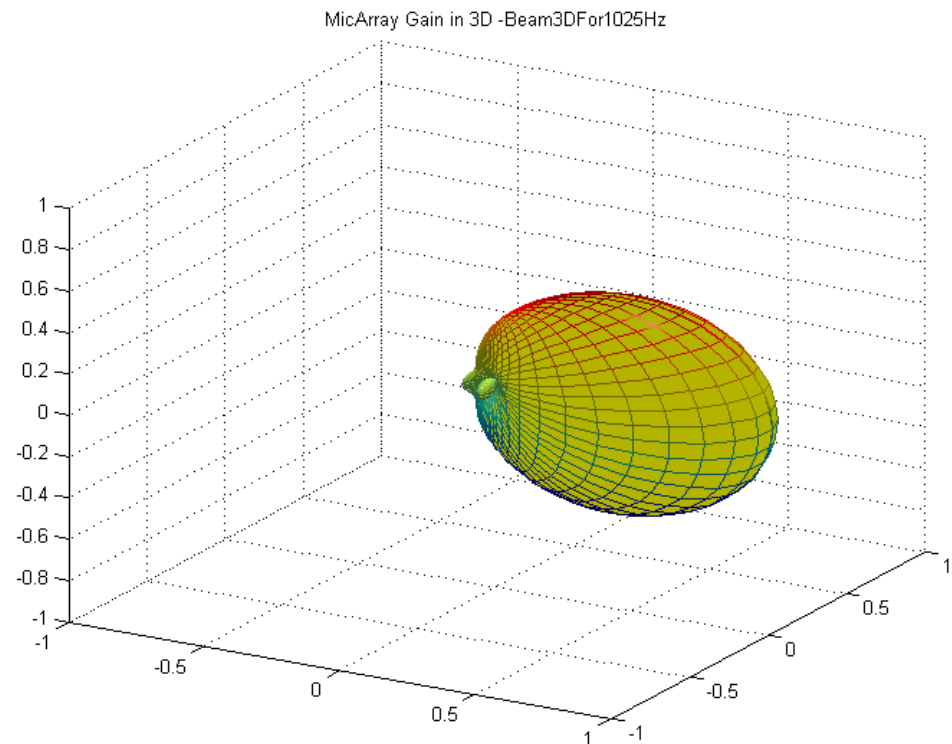
- Design in the beamspace
- Define the target

$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right)$$

- Define the weigh
- Combine the mic using weighted \mathbf{M}

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times M}$$

- Do the design in



Pattern synthesis

- Design in the beamspace
 - Define the target beam shape:
- $$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
 - Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

Pattern synthesis

- Design in the beamspace

- Define the target

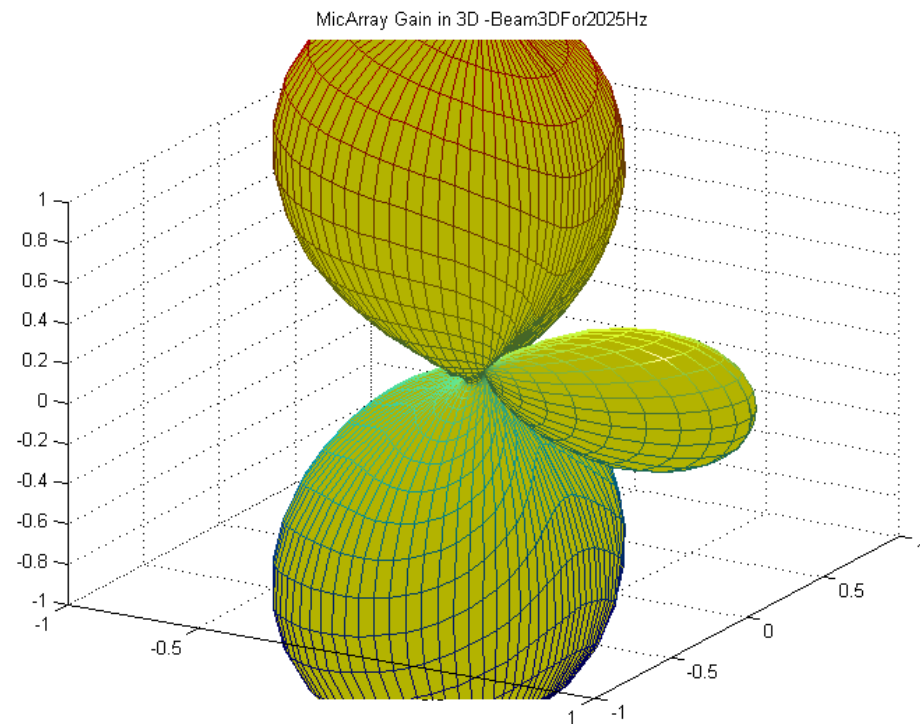
$$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right)$$

- Define the weight

- Combine the mic using weighted I

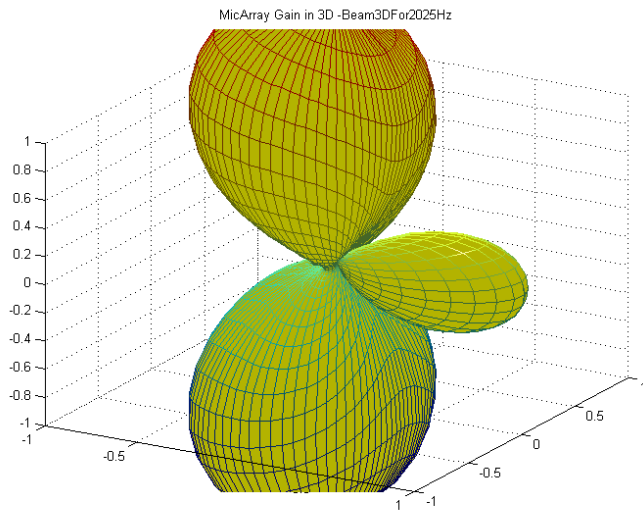
$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_M$$

- Do the design in



Pattern synthesis

- Design in the beamspace



at beam shape:

$$\cos\left(\frac{\pi}{2}\right)$$

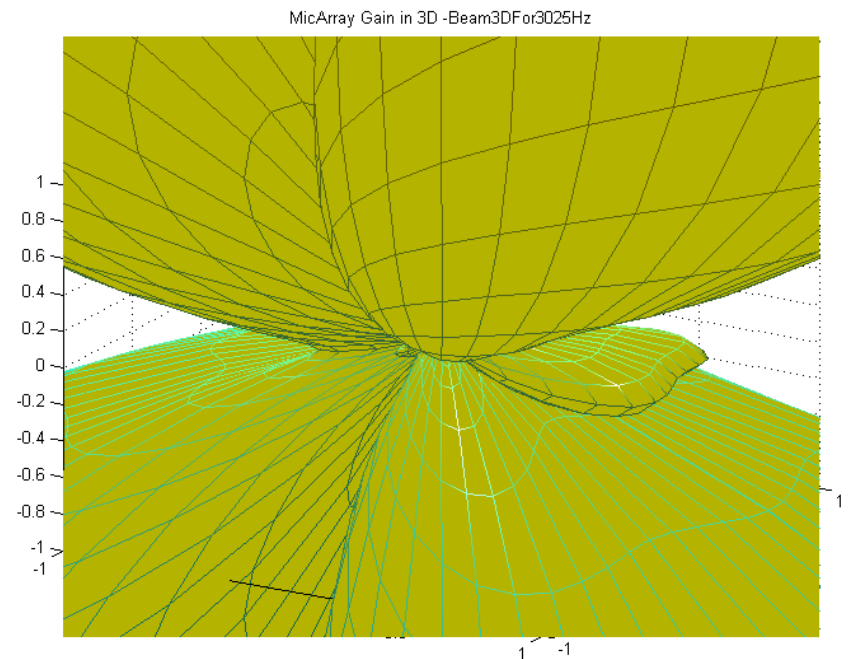
ht fur

icrop

MMS

$$\mathbf{I}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{W}_{M \times L} \mathbf{W}_{1 \times L}$$

- Do the design in 3D



Pattern synthesis

- Design in the beamspace
 - Define the target beam shape:
- $$T(\rho, \varphi, \theta, \delta) = \cos\left(\frac{\pi(\rho_T - \rho)}{k\delta}\right) \cos\left(\frac{\pi(\varphi_T - \varphi)}{\delta}\right) \cos\left(\frac{\pi(\theta_T - \theta)}{\delta}\right)$$
- Define the weight function
 - Combine the microphone directivity patterns using weighted MMSE

$$\mathbf{T}_{1 \times L} = \mathbf{V}_{1 \times L} \mathbf{D}_{M \times L} \mathbf{M}_{M \times L} \mathbf{W}_{1 \times M}$$

- Do the design in 3D

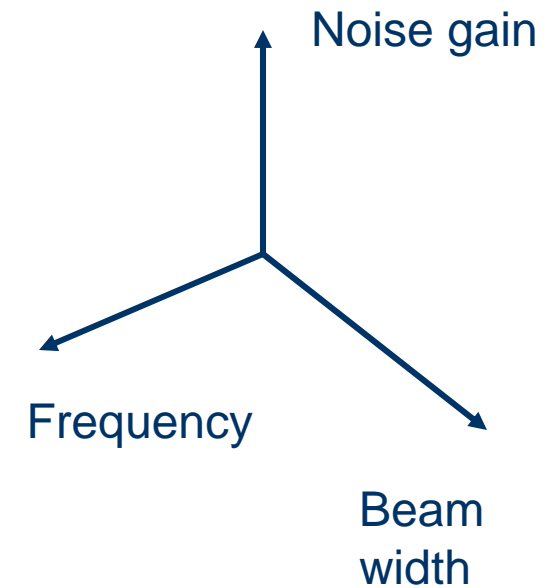
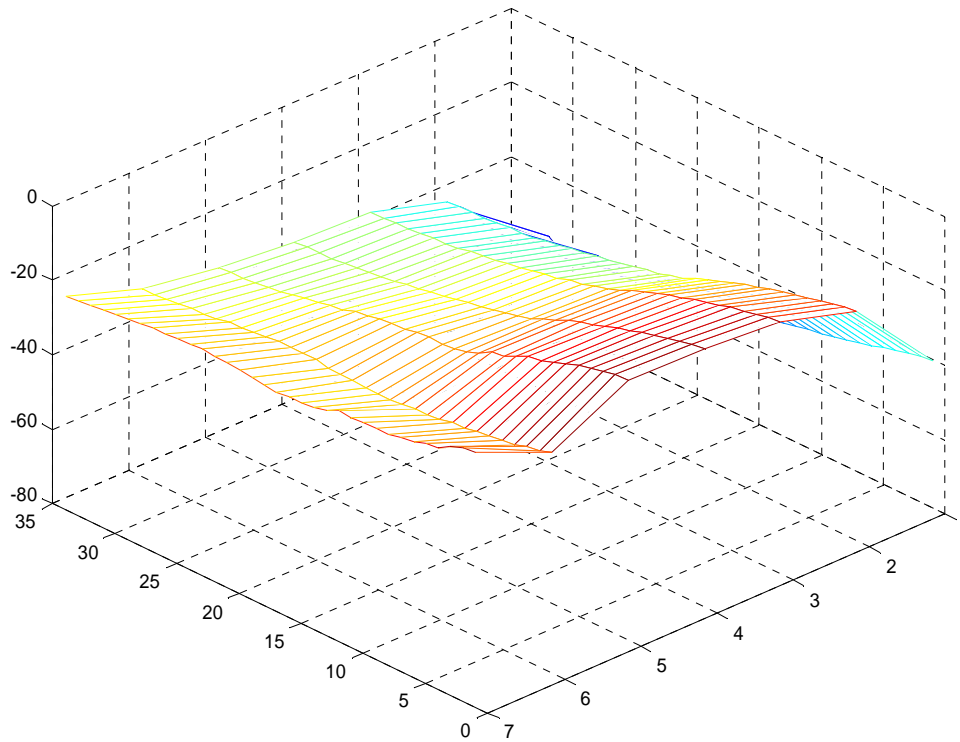
Dimensions reduction

- Dimensions reduction: from 2M to 1
- Two controversial processes:
 - Narrow beam: better ambient noise reduction
 - Wide beam: better internal noise reduction
- One dimensional search: beam width
- Cover the whole frequency band
- Calculate set of beams

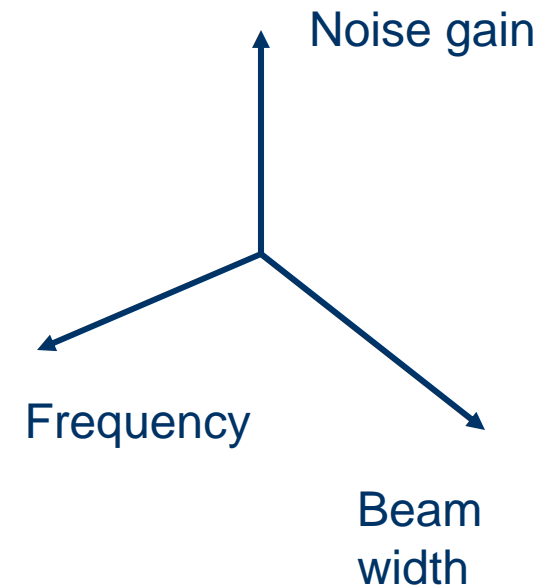
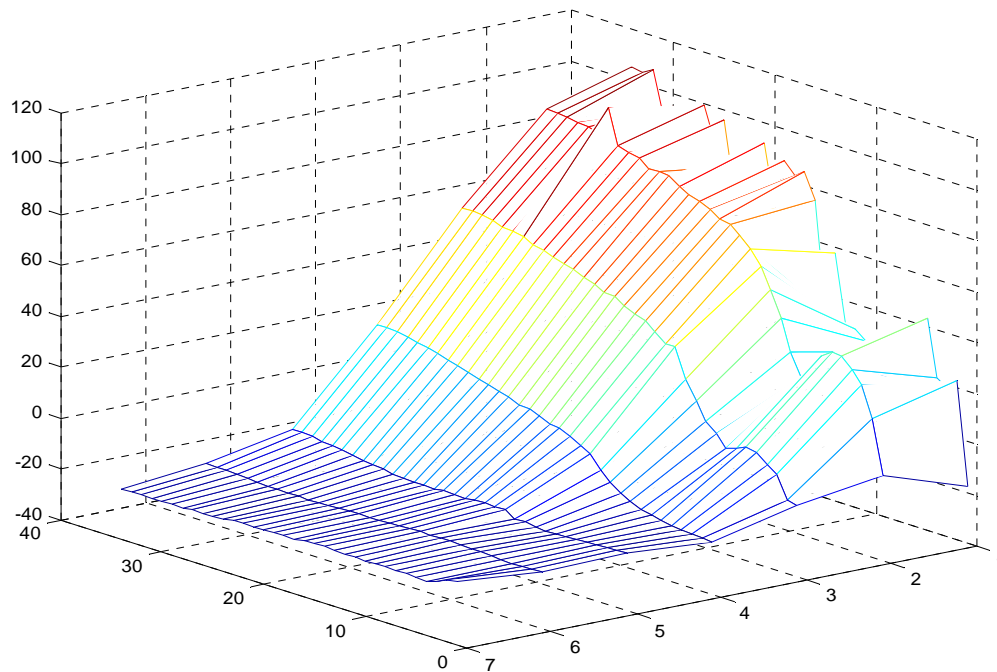
On next charts:

- Z-axis: noise gain in dB
- X-axis: frequency, logarithmic, 1-100Hz, 2-200 Hz, 3-400Hz, ...7-6400Hz
- Y-axis: beam width, linear, 0 – 180⁰, every 5⁰, 33-15⁰.

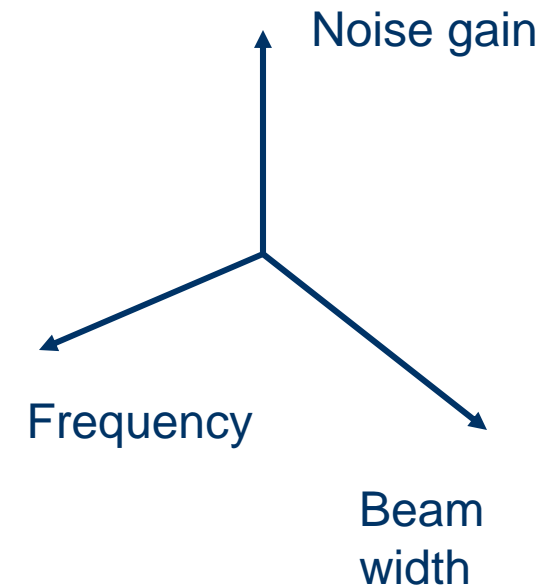
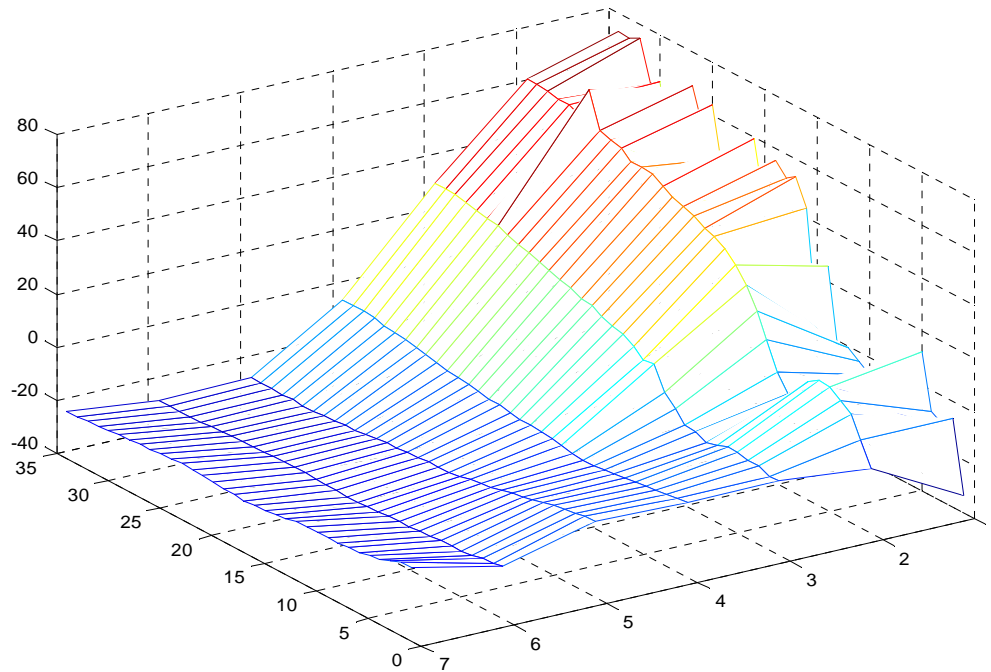
Ambient noise gain



Non-correlated noise gain



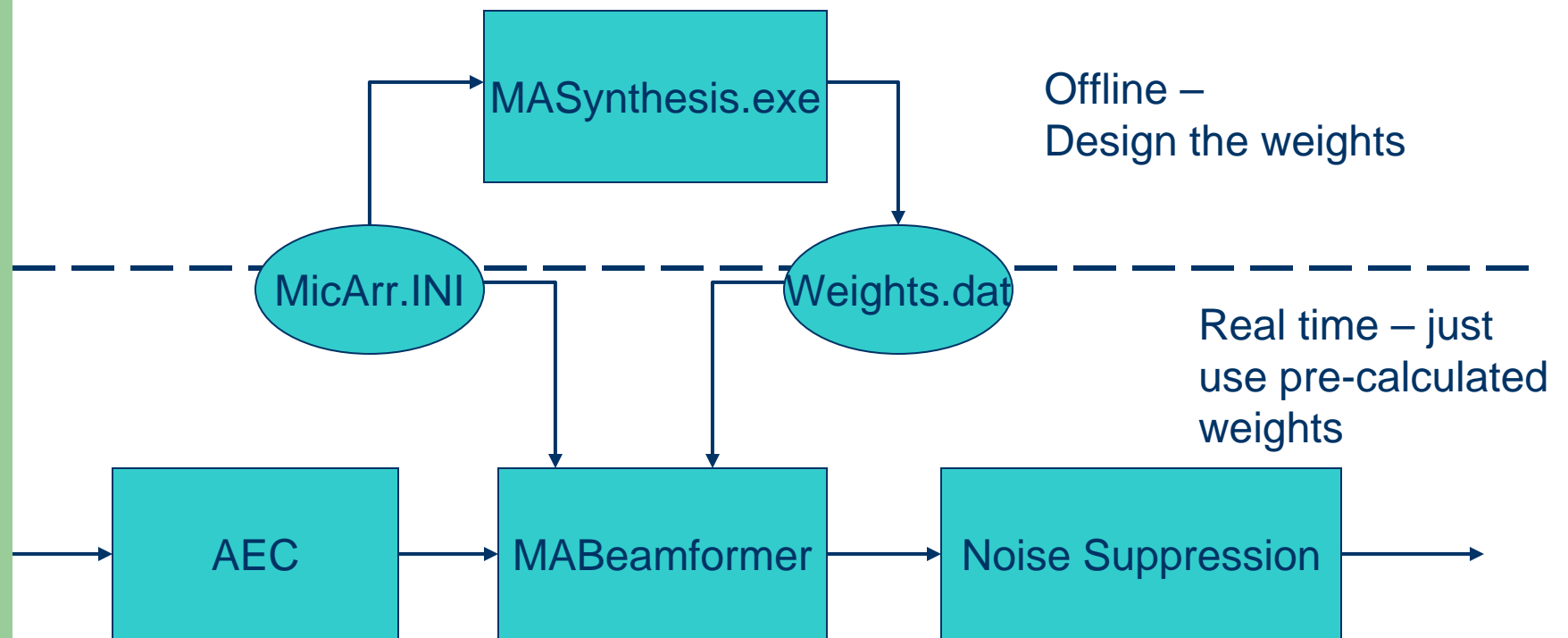
Total noise gain



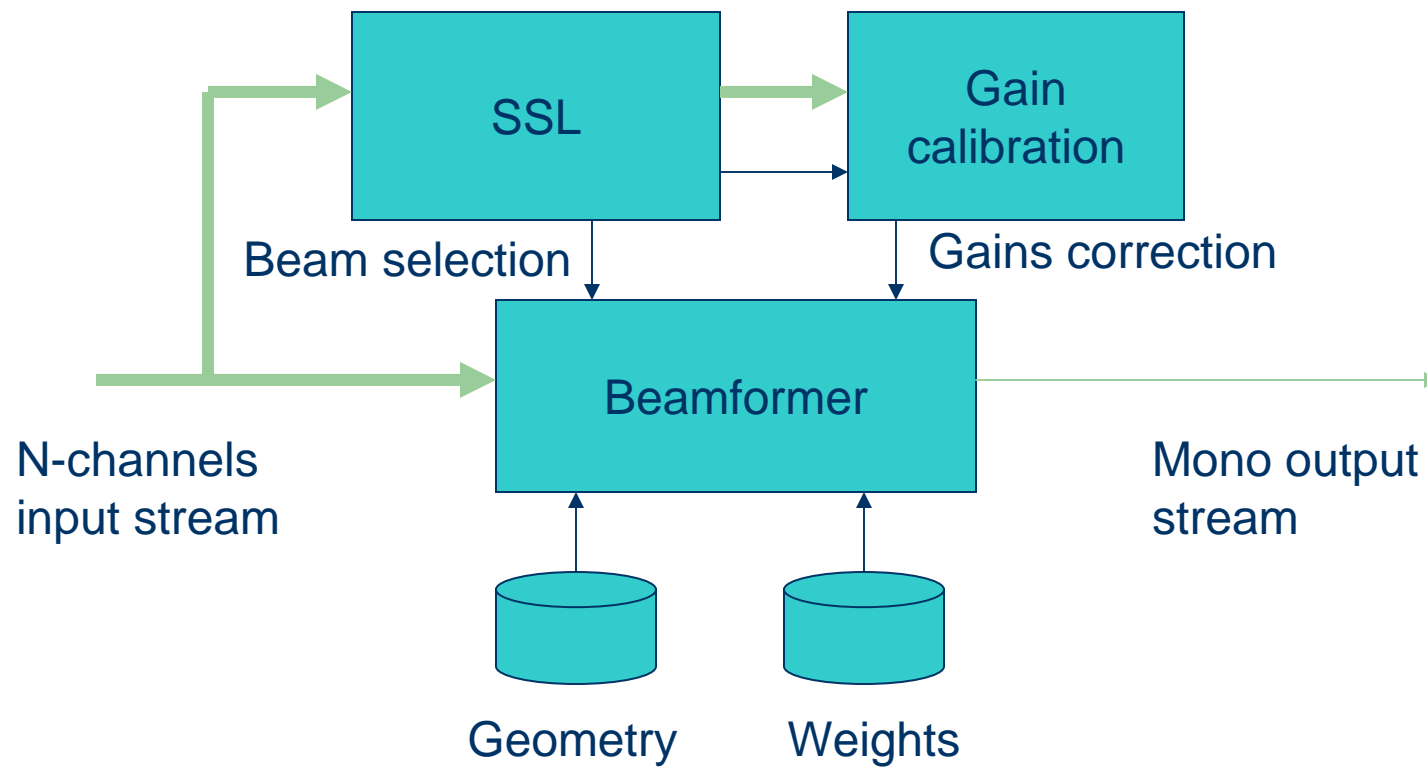
Dimensions reduction

- Dimensions reduction: from 2M to 1
- Two controversial processes:
 - Narrow beam: better ambient noise reduction
 - Wide beam: better internal noise reduction
- One dimensional search: beam width
- Cover the whole frequency band
- Calculate set of beams

Implementation: overall

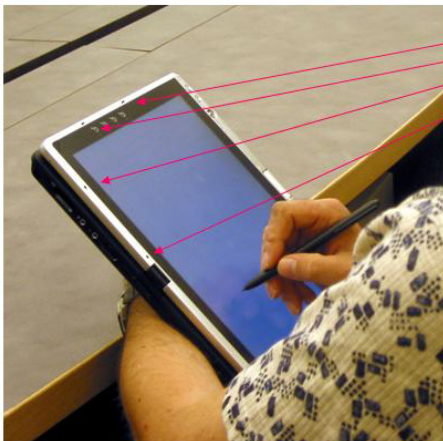


Implementation: Real-time engine



Hardware designs

- USB MicArray Prototypes
 - 4-mic desktop
 - 8-mic conference tabletop
 - Bus-powered (no power grid)
 - Compatible with USB audio (no device drivers to install)
- Integrated in laptops/monitors



Microphones of the L-shaped microphone array for Tablet PC



Results: noise suppression

- Microphone Array noise suppression
 - Provides itself 14-18 dB ambient noise suppression
 - Helps the noise suppressor to do better job
 - More at <http://micarray>
- One of the best technologies on the market

Device	Noise	Signal	SNR
Omni-directional Microphone	-45.53	-40.64	4.89
Unidirectional Microphone	-44.51	-33.91	10.6
Close-Up Microphone	-64.46	-30.04	34.42
Andrea DA 400 2.0, 4 el. MA, \$135	-51.72	-26.19	25.53
Acoustic Magic, 8 element MA, \$250	-62.39	-32.6	28.79
MSR 4 elements + WinXP NS	-61.68	-33.86	27.82
MSR 4 elements + New NS	-64.41	-32.14	33.27

Results: speech recognition

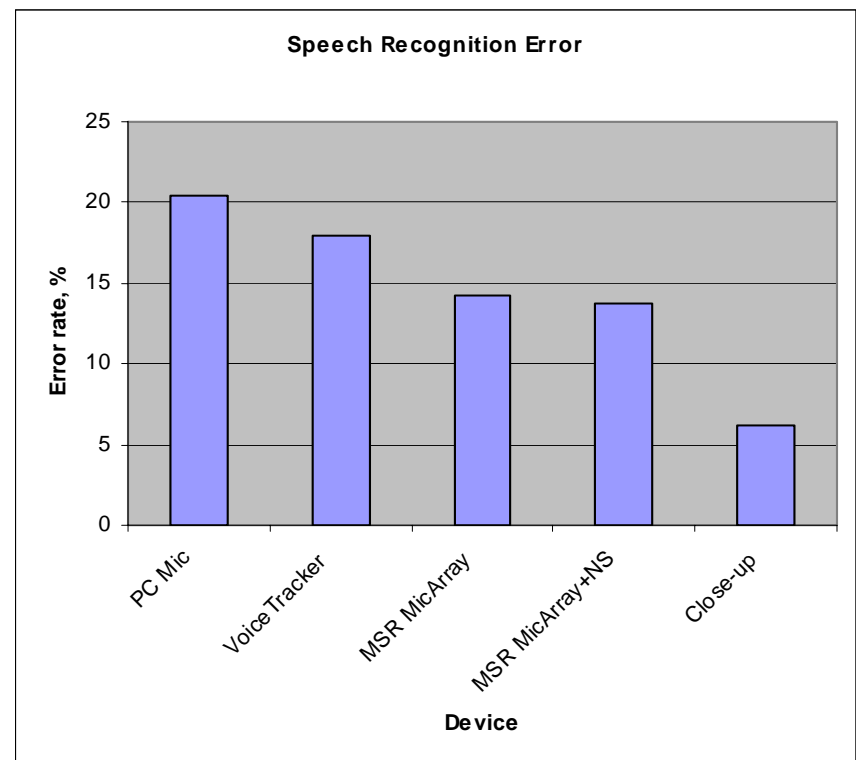
- Microphone Arrays for speech recognition

- Linear processing, speech recognition friendly
- Reduces ambient noises
- Partial de-reverberation

- Results

Device	Error rate, %	Time
PC Mic	20.391	3:25
VoiceTracker	17.9	3:17
MSR MicArray	14.22	4:03
MSR MicArray+NS	13.683	3:34
Close-up	6.171	2:35

4 element array, Yakima SAPI 5.2
374 utterances, 7 speakers
(4 male, 3 female), age 25-53



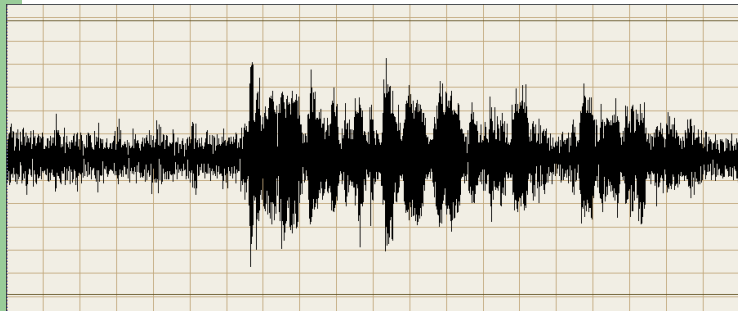
Results: conclusions

- Ambient noise suppression
 - The current technology provides good noise suppression under the quality requirements constraints
 - Telecommunication scenario has good quality sound
 - Meetings recording for listening purposes – OK.
- Speech recognition results
 - Need improvement
 - Reverberation as major reason
 - Important for recorded meetings search technology

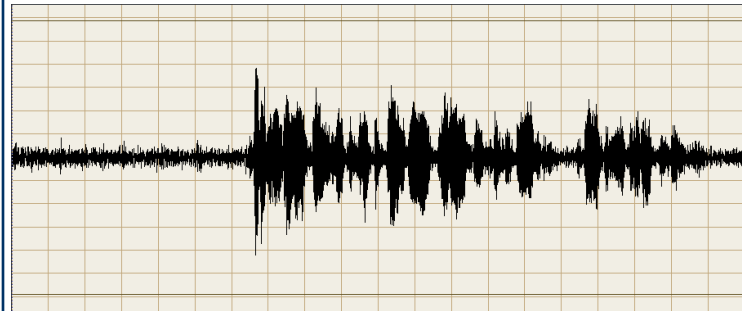
Microphone Array - Example

- Person speaking at 3 ft from microphones

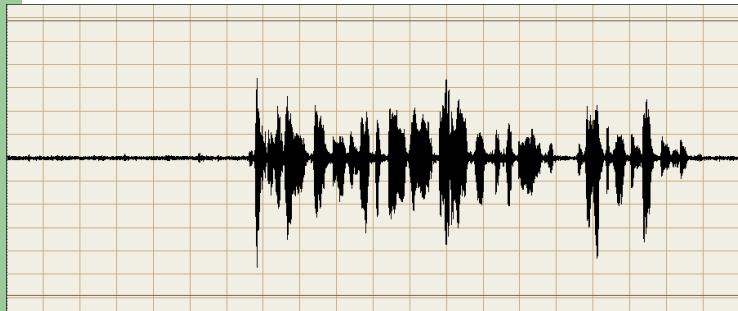
Typical \$10 PC microphone SNR=10.3 dB



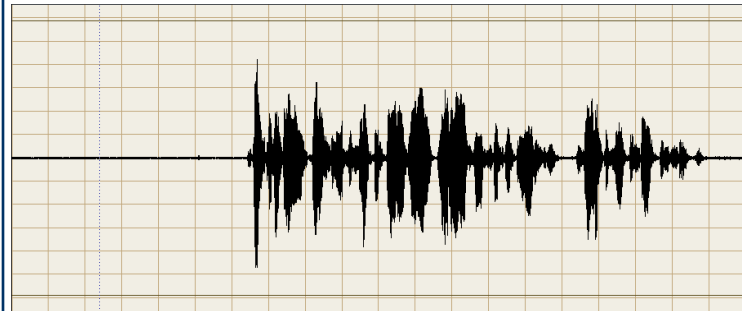
PC mic + WinXP noise reduction SNR=18.4 dB



Competitor (HW DSP) SNR=34.4dB



MSR USB desktop array SNR=42.5dB



Microphone array - demo

- First demo:
 - Records in parallel the output of the microphone array and a regular PC microphone.
 - After this merges both WAV files to one file ...
 - ... and plays it with CoolEdit.
- Second demo: ClearMessage application

Take outs

Most of our projects are optimization in one way or another:

- Try carefully to define the optimization criterion
- Reduce the number of dimensions as much as possible
- Choose the method, especially if there are too many papers and no definite answer

Finally

Questions?

Contact: ivantash@microsoft.com

See: <http://research.microsoft.com/users/ivantash/>