Robust signal processing with Hidden Nonlinearities for Microphone Arrays

Greg Kochanski

&

Mohan Sondhi

(Fourier transform and filter work with David Thomson).

Outline

- What’s the idea?
- What is a robust estimator?
- How do you build a microphone array?
- How well does a 5 element array perform?
- How far can it be pushed?

What’s the idea?

- Using a robust estimator to combine signals in a microphone array will provide better directivity.
  - It rejects noise close to any microphone (>10dB). [auto: squeaks, rattles, open windows]
  - It rejects distant noise when the focus is close to a microphone (2-3 dB). [auto: road noise]
  - It behaves better on dirty data. [wireless links: dropouts]
  - It never behaves more than about 0.5dB worse than linear.

- Distortion and nonlinearities are controllable
  - <-30dB for source on focus in a 300ms reverberant room.
  - <-20dB for source off focus (relative to focus).

What’s the idea?

$$\sum_i c_i v_i \rightarrow \begin{matrix} \textit{Weighted} \\ \textit{Robust} \\ \textit{Estimator} \end{matrix} \left(\alpha_i v_i, \textit{weight} = w_i\right)$$

We replace sums and averages by robust estimators.

C translates into alpha and the weight (c, alpha, weight are independent of v). We use the extra freedom gained by introducing a weight to make the overall system behave as much as possible like the original sum (I.e., minimize distortion at the focus of a microphone array, or minimize distortion in a filter passband, or whatever seems desirable).

- Microphone arrays
- Fourier transforms
- Filters (especially FIR)

What’s a robust estimator?

For any non-Gaussian probability distribution, there is some robust estimator that is better than *any* linear estimator.

Better means “has lower variance” which is equivalent to saying “has a smaller response to a signal off the focus.”

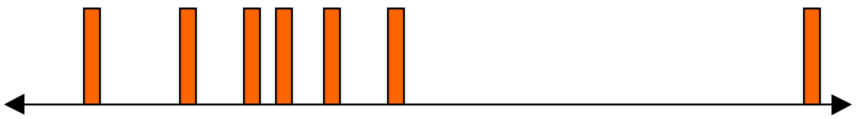
- Non-Robust:
  - Average
  - Sum
  - Optimal for Gaussian probability distributions
- Robust:
  - Median
  - Trimmed Mean
  - Anything that is insensitive to the tails of the probability distribution or to large motions of a few data.
  - Optimal for non-Gaussian distributions.

$$\{v_i\} \rightarrow \bar{v}$$

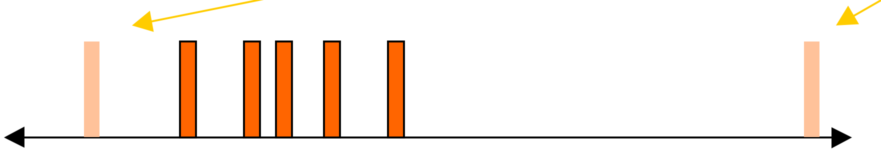
- Unless you know a *lot* about your data, then a robust estimator is safer.
  - Most real-world data is imperfect

What’s a robust estimator?

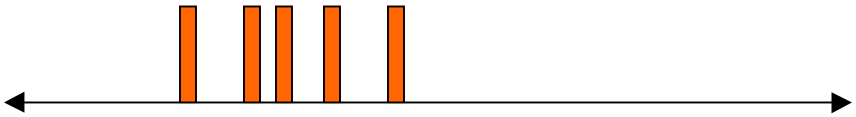
What’s a trimmed mean?



These points will be ignored, so they can move arbitrarily far and will have no effect on the final result.



Remove N points from each end.



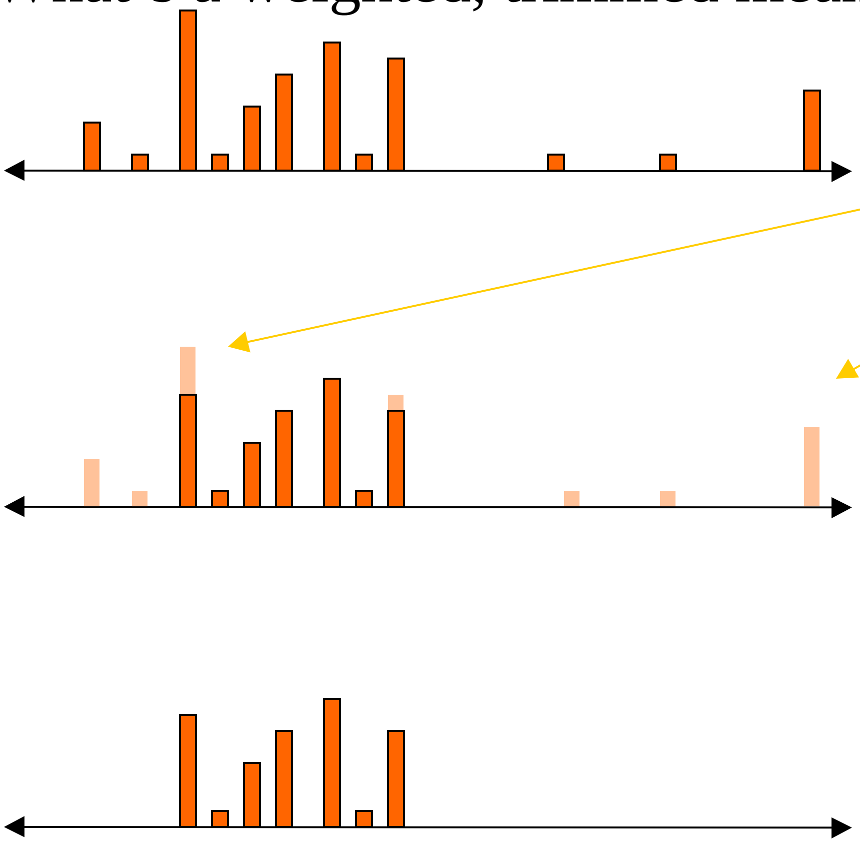
Average what is left over.

$$\{v_i\} \rightarrow \bar{v}$$

This is like the olympic voting scheme: drop the high and low judges, and average the rest. Of course, you can drop more than one if you have enough data.

What’s a robust estimator?

What’s a weighted, trimmed mean?



These points will be ignored, so they can move arbitrarily far and will have no effect on the final result.

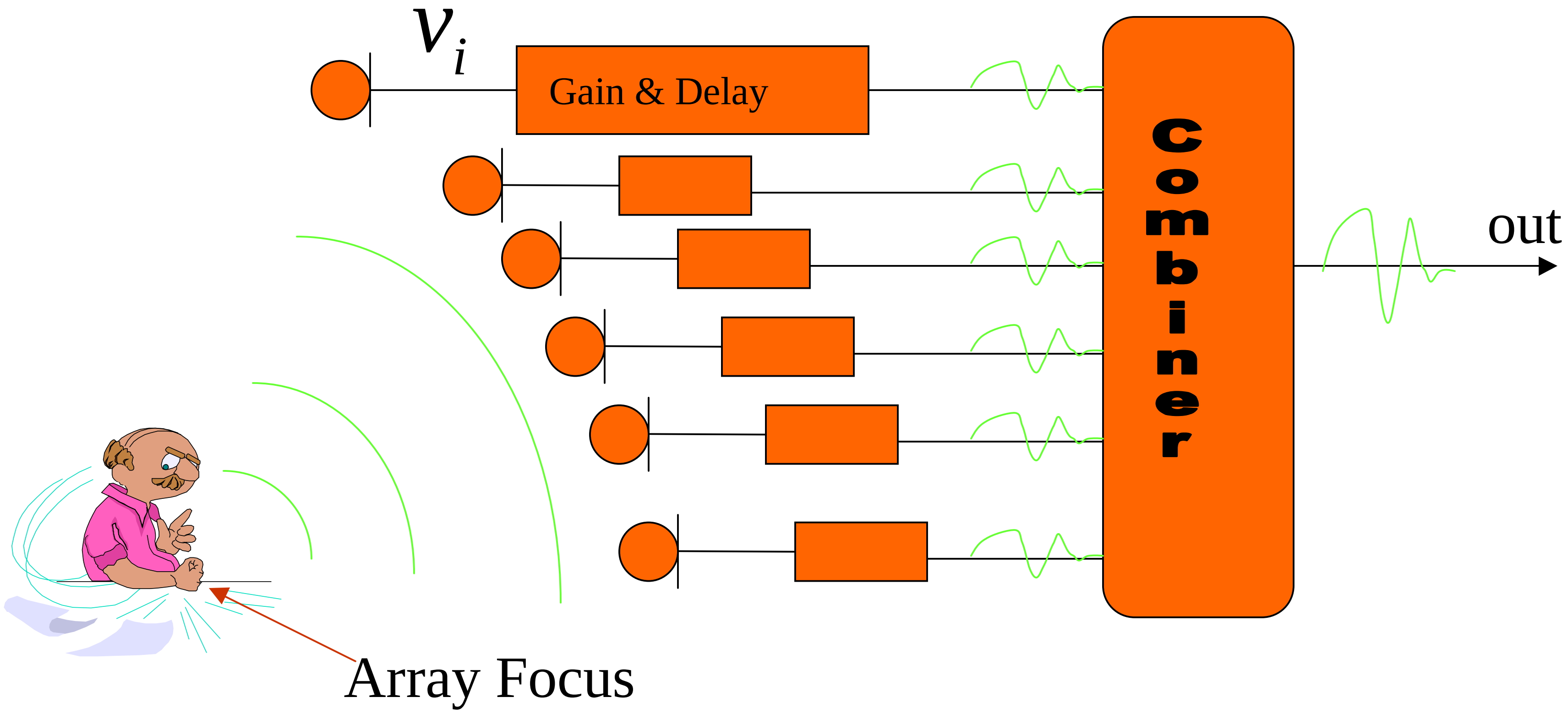
Remove some fraction of the weight from each end.

We’ll remove data, starting from each end, until we’ve removed bars that total 1 inch high, in this example.
So, we remove the same total weight from each end, but we end up removing 3 (and a bit) bars from the right end, and 2 (and a bit) bars from the left end.

Do a weighted average of what is left over.

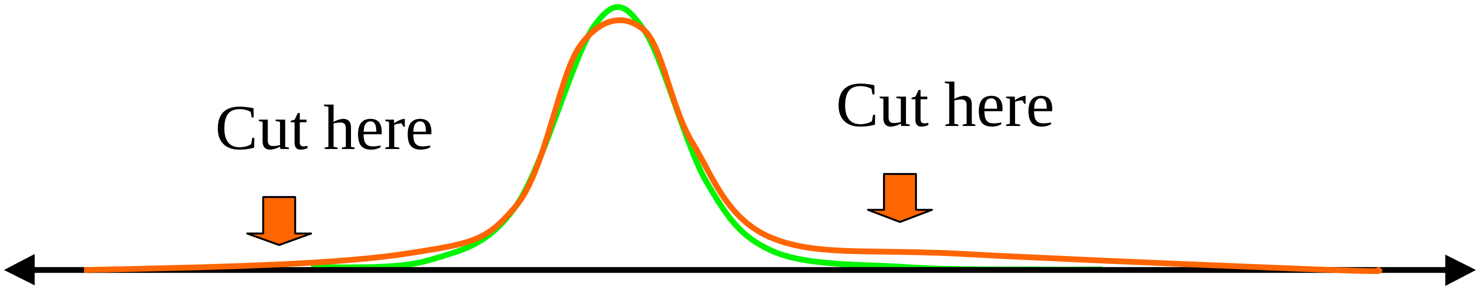
$$\{(v, weight)_i\} \rightarrow \bar{v}$$

Building Microphone Arrays



Often, $out(t) = \sum_i g_i v_i(t - \tau_i)$, a weighted average of the delayed input signals.

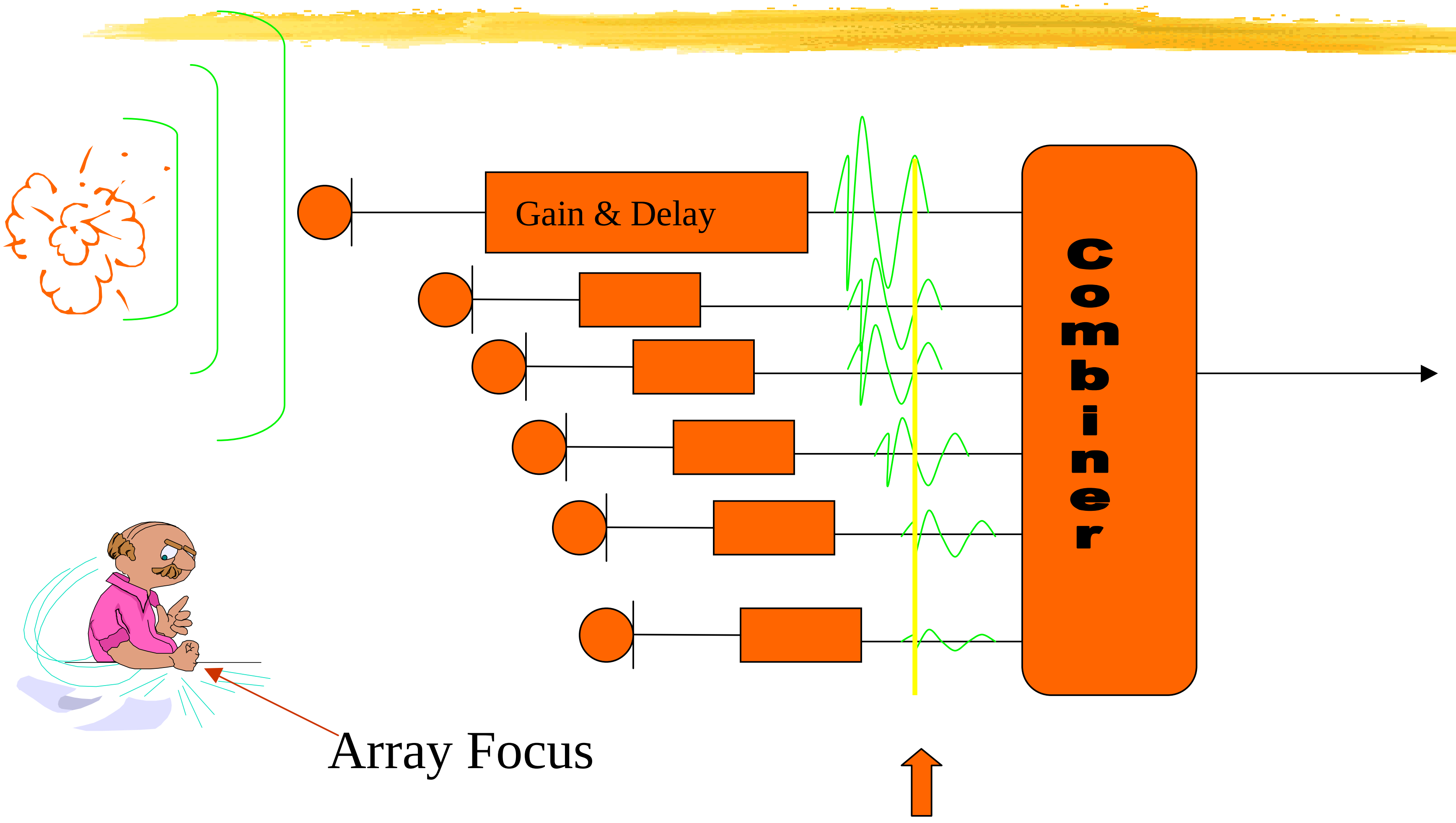
Building microphone arrays -- Why a robust estimator?



Any time a probability distribution is non-Gaussian, there is some robust estimator that has a lower variance than the optimal linear estimator.

So what?

Building Microphone Arrays



Gain & Delay

Combiner

Array Focus

Consider this sample

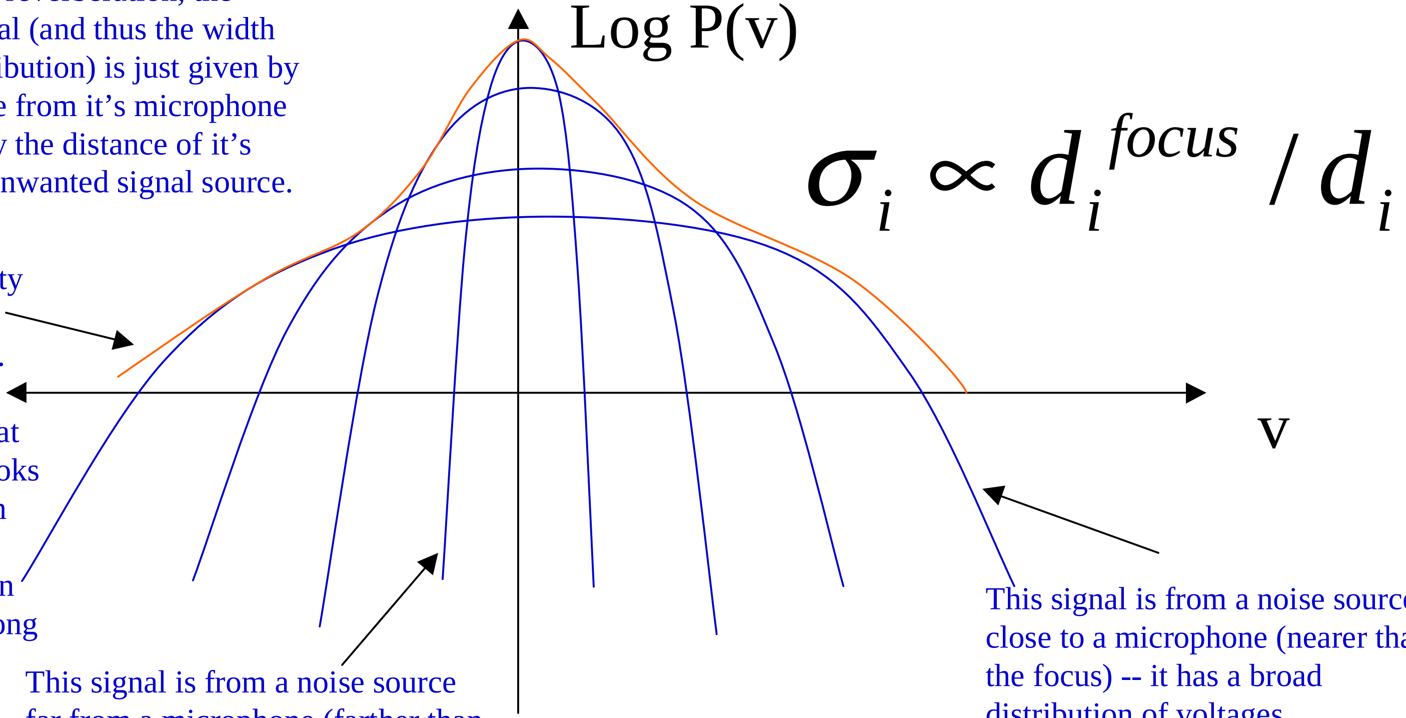
Now, consider what happens for a source away from the focus.

We’ve adjusted gains and time delays for the focus, so all the signals will (in general) be of different amplitudes.

Building microphone arrays

- The instantaneous distribution of voltages at the input to the combiner is non-Gaussian, *even if the signals are each white and Gaussian.*

In a toy model without reverberation, the amplitude of each signal (and thus the width of it's probability distribution) is just given by the ratio of the distance from it's microphone to the focus, divided by the distance of it's microphone from the unwanted signal source.



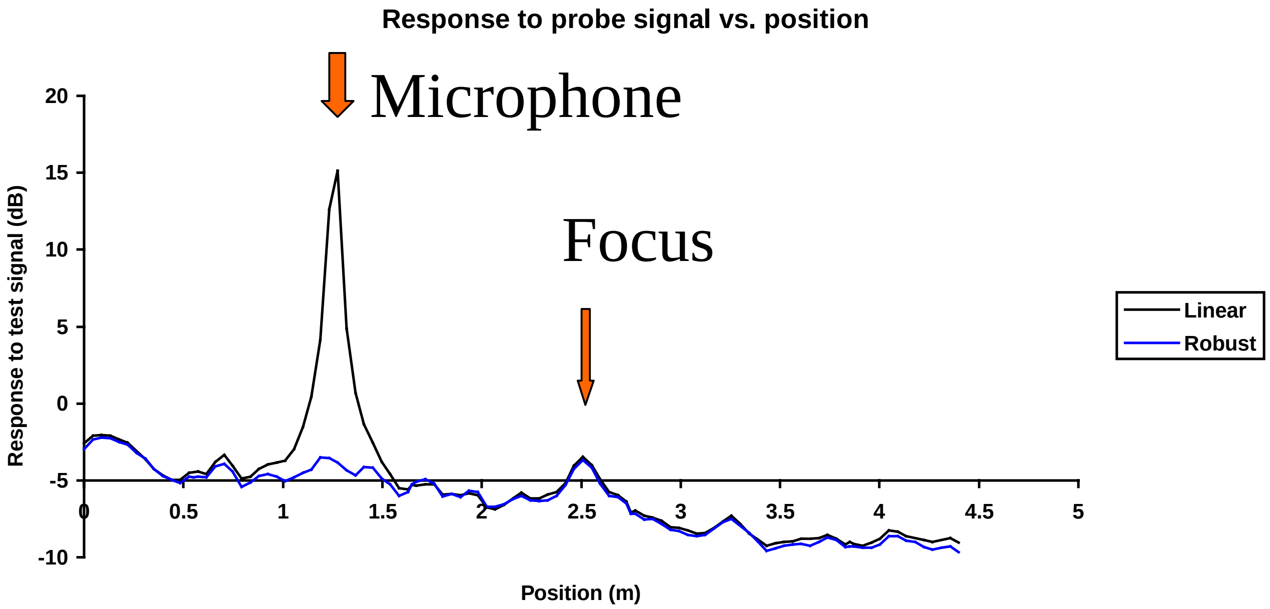
$$\sigma_i \propto d_i^{focus} / d_i$$

Red marks the probability distribution of voltages going into the combiner. The combiner is an instantaneous operator: at any given moment, it looks at one voltage from each microphone. The distribution is a Gaussian mixture, and can have long tails.

This signal is from a noise source far from a microphone (farther than the focus) -- it has a narrow distribution of voltages.

This signal is from a noise source close to a microphone (nearer than the focus) -- it has a broad distribution of voltages.

How well does it work?



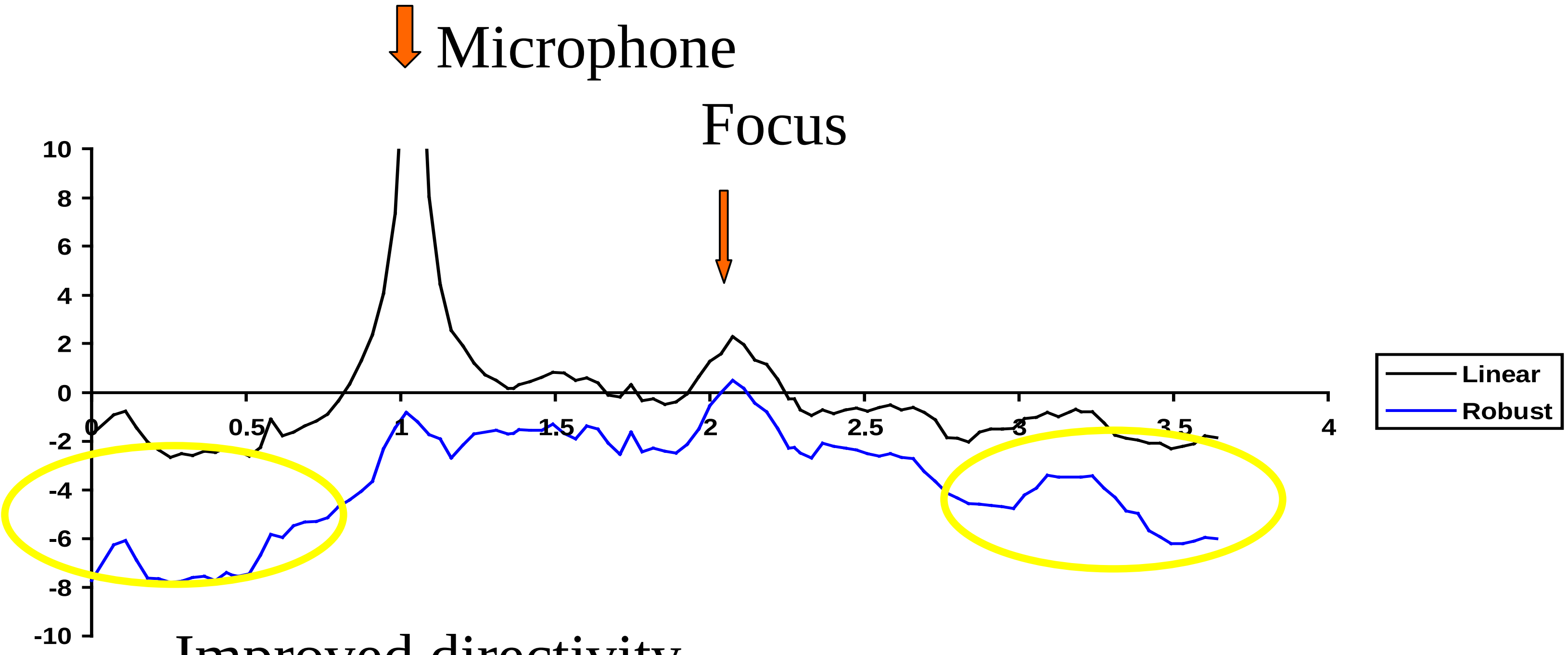
Microphone

Focus

5 element array, 400 ms reverberation time

You can see that the robust estimator dramatically reduces the unwanted signal when the source passes close to one of the microphones in the array.

How well does it work?



Microphone

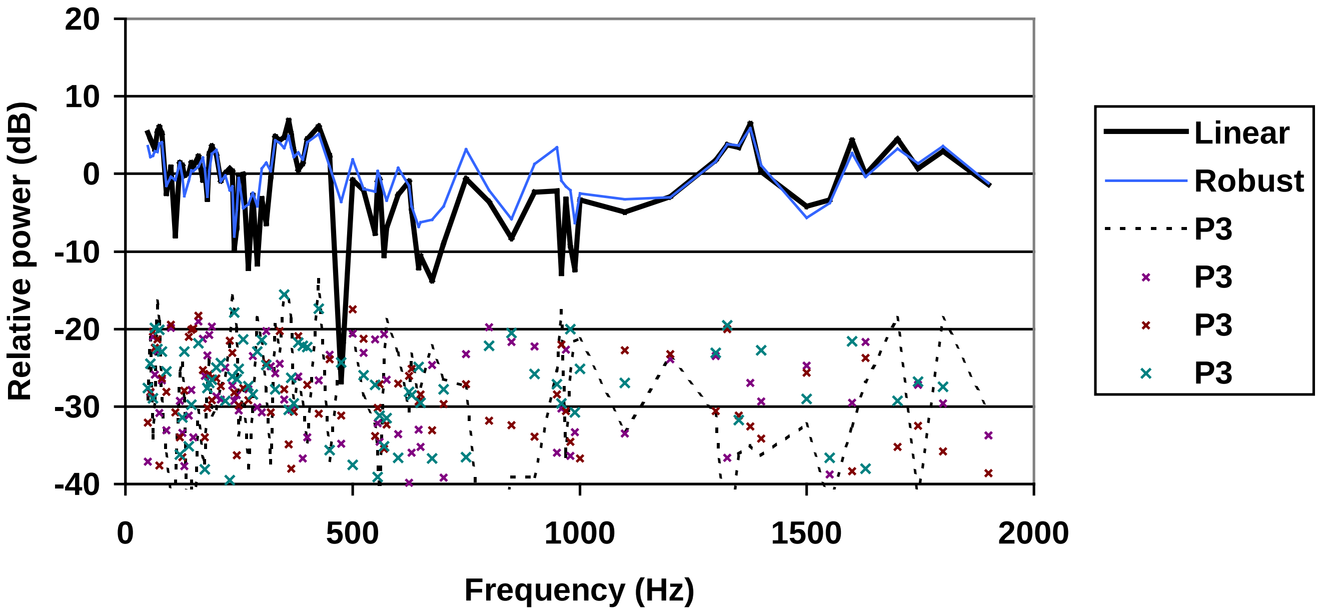
Focus

Improved directivity

Here, we've moved the focus close to one microphone, and you can see an extra 2-3dB suppression of the unwanted signal over large areas of the room (circled in yellow). This is the case where d_i^{focus} / d_i is much smaller than one.

How well does it work?

Nonlinearity in a reverberant room



5 - element array, 400ms room reverberation time.

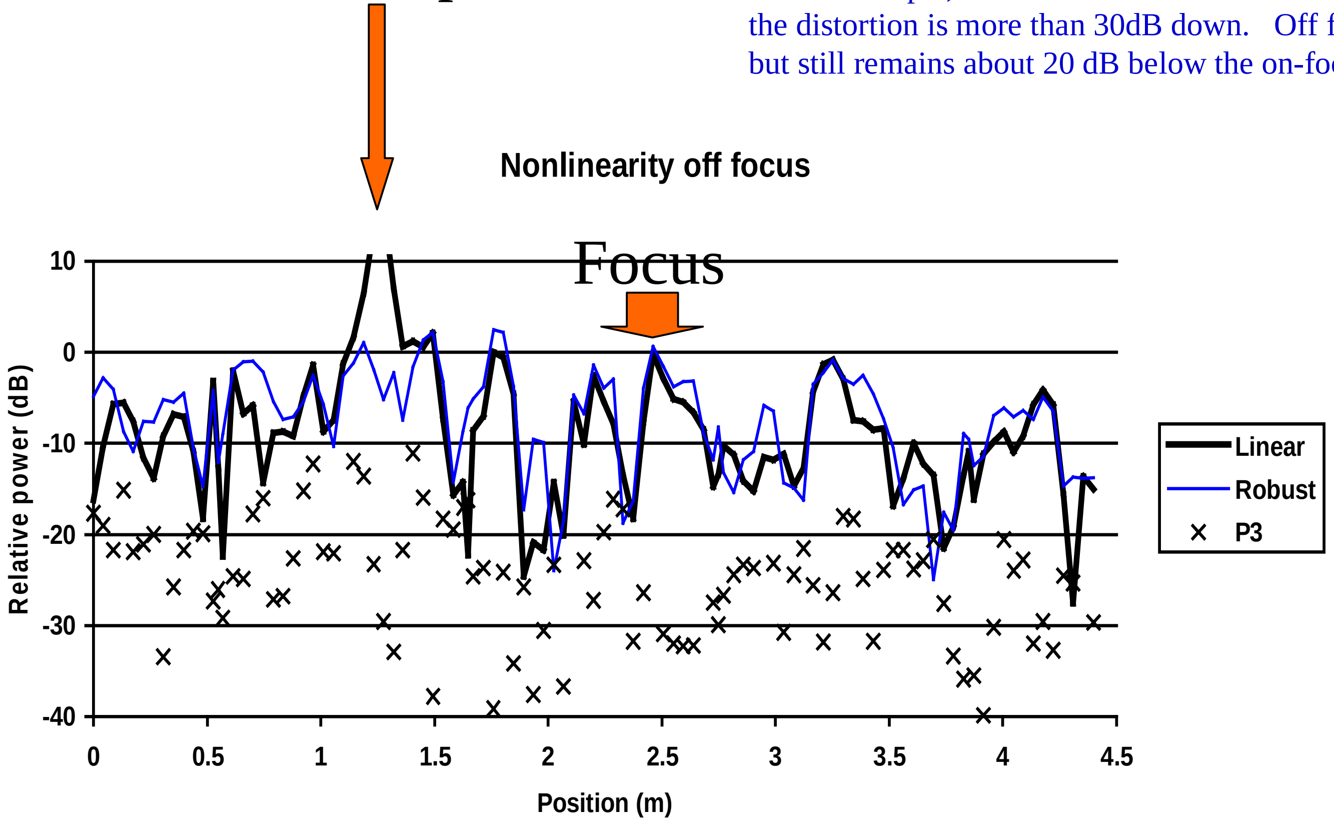
Sound source is on focus.

In these examples (4 different room configurations), the distortion (measured by the third harmonic power) is about 28 dB below the fundamental. The two top curves show the frequency response of the linear and robust systems.

How well does it work?

Microphone

In this example, we move the source across the room. On focus, the distortion is more than 30dB down. Off focus, it is higher, but still remains about 20 dB below the on-focus signal.

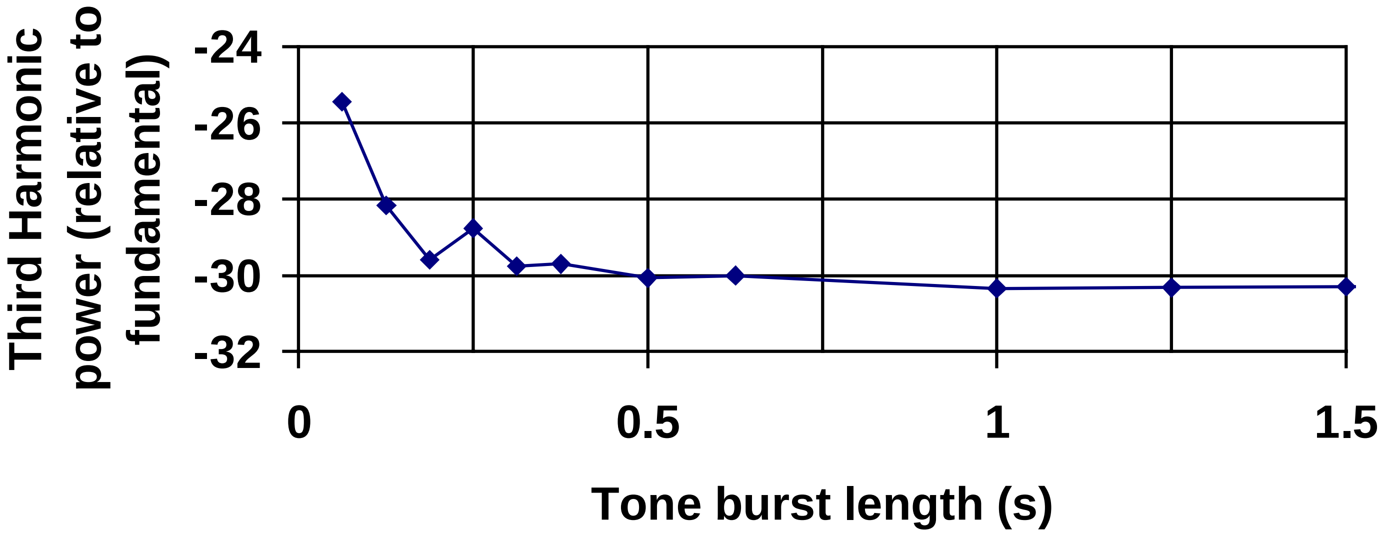


5-element array, 400ms reverberation time

How far can it be pushed?

- Distortion scales as 1/N
- There are operators that give less distortion than a trimmed mean
- I use a simple, dynamic adjustment of the gains for each channel.
  - Could add dynamic phase tracking
- I don’t use room reflections - they’re just noise
  - Could treat reflections as more microphones
- Need to test on real data
  - I assume only free-space propagation of sound and wall reflections.
- Use this idea for other algorithms.

Decrease in distortion via Dynamic Gain Adjustment



The whole trick to getting the distortion down is to minimize the variance of the signals presented to the combiner.

I used a simple dynamic gain adjustment to account for the difference between the free-space transfer function and the transfer function of the simulated reverberant room. It takes about 0.25s to adjust the gains. Since I start the gains from their free-space values, short tone bursts are processed with inaccurate gain settings, and show more distortion.