Kernel-Interpolation-based Filtered-x Least Mean Square for Spatial Active Noise Control in Time Domain

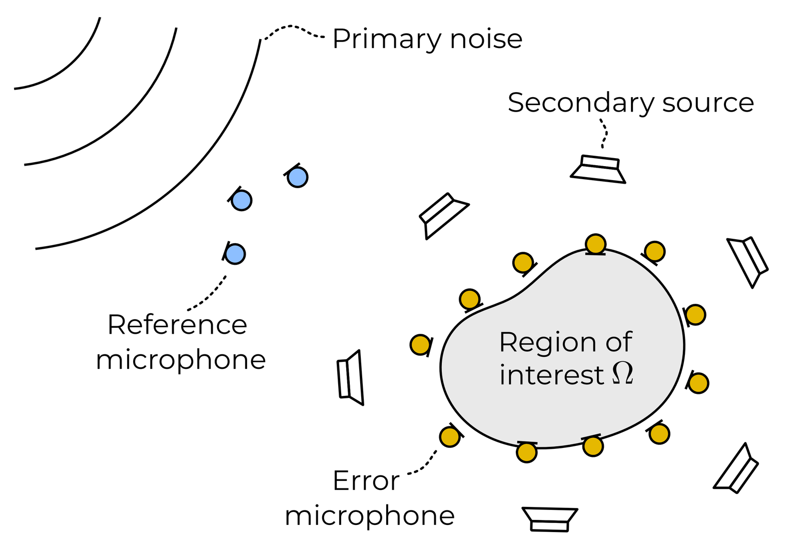
Jesper Brunnström
Shoichi Koyama
The University of Tokyo

Abstract

- A time-domain spatial active noise control algorithm for broadband noise using kernel interpolation of sound field is proposed.
- Proposed method outperform conventional multipoint pressure control with regards to regional power reduction.
- Small to no increase in computational cost compared to conventional FxLMS.
- A fast block implementation is proposed, significantly reducing computational cost at a minor performance cost.

Problem statement

Goal: Broadband noise reduction over a continuous region with flexible array placement and low computational cost.



Error signal

$$\boldsymbol{e}(n) = \underbrace{\boldsymbol{d}(n)}_{\text{Primary noise}} + \sum_{j=0}^{J-1} \overbrace{\boldsymbol{G}(j)}^{\text{Secondary path}} \underbrace{\boldsymbol{y}(n-j)}_{\text{Loudspeaker driving signal}}$$

Loudspeaker signal

$$\boldsymbol{y}(n) = \sum_{i=0}^{I-1} \overbrace{\boldsymbol{W}(i)}^{\text{Control filter}} \underbrace{\boldsymbol{x}(n-i)}_{\text{Reference signal}}$$

Cost function: Total sound power within region of interest

$$\mathcal{L} = \mathbb{E}\left[\int_{\Omega} \underbrace{u(n,\boldsymbol{r})}_{\text{Sound pressure}}{}^2 d\boldsymbol{r}\right]$$

Kernel Interpolation

- Estimate sound pressure function from error microphone measurements.
- Optimal interpolation filter is given in closed form in frequency domain.
- Obtain time domain filter through inverse Fourier transform.

$$u(n,\boldsymbol{r}) = \sum_{i=-\infty}^{\infty} \boldsymbol{z}^{\top}(i,\boldsymbol{r})\boldsymbol{e}(n-i)$$

$$\boldsymbol{z}(i,\boldsymbol{r}) = \mathcal{F}^{-1}\Big[\boldsymbol{z}(\omega,\boldsymbol{r})\Big](i) = \mathcal{F}^{-1}\Big[\big[(\boldsymbol{K}(\omega)+\lambda\boldsymbol{I})^{-1}\big]^{\top}\boldsymbol{\kappa}(\omega,\boldsymbol{r})\Big](i)$$

- Interpolation filter is determined by the kernel function.
- This kernel function imposes requirement that pressure function must satisfy Helmholtz equation.
- Only positions of microphones and region of interest is required to compute the filter.

$$\kappa(\omega,\boldsymbol{r},\boldsymbol{r}') = \overbrace{j_0}^{\text{Spherical Bessel function}}\Big(\frac{\omega}{\underbrace{c}_{\text{Speed of sound}}}\|\boldsymbol{r}-\boldsymbol{r}'\|\Big)$$

$$\boldsymbol{\kappa}(\omega,\boldsymbol{r}) = \big[\kappa(\omega,\boldsymbol{r},\boldsymbol{r}_1),\ldots,\kappa(\omega,\boldsymbol{r},\boldsymbol{r}_M)\big]$$

$$\boldsymbol{K}(\omega) = \begin{bmatrix} \kappa(\omega,\boldsymbol{r}_1,\boldsymbol{r}_1) & \ldots & \kappa(\omega,\boldsymbol{r}_M,\boldsymbol{r}_1) \\ \vdots & \ddots & \vdots \\ \kappa(\omega,\boldsymbol{r}_1,\boldsymbol{r}_M) & \ldots & \kappa(\omega,\boldsymbol{r}_M,\boldsymbol{r}_M) \end{bmatrix}$$

Kernel-interpolation-based FxLMS

- Express cost function using kernel interpolation filter and error signal.

$$\mathcal{L} = \mathbb{E}\left[\sum_{i,j=-\infty}^{\infty} \boldsymbol{e}^{\top}(n-i)\boldsymbol{\Gamma}(i,j)\boldsymbol{e}(n-j)\right]$$

$$\boldsymbol{\Gamma}(i,j) = \int_{\Omega} \boldsymbol{z}(i,\boldsymbol{r})\boldsymbol{z}^{\top}(j,\boldsymbol{r})\,\mathrm{d}\boldsymbol{r}$$

- Gradient is obtained with a simplified interpolation weighting filter.

$$\nabla\mathcal{L}(i) = \sum_{k=-\infty}^{\infty}\sum_{j=0}^{J-1} \boldsymbol{G}^{\top}(j)\boldsymbol{A}(k)\mathbb{E}\Big[\boldsymbol{e}(n)\boldsymbol{x}^{\top}(n-i-j-k)\Big]$$

$$\boldsymbol{A}(k) = \sum_{j=-\infty}^{\infty} \boldsymbol{\Gamma}(j,j+k) = \mathcal{F}^{-1}\left[\int_{\Omega} \boldsymbol{z}^{*}(\omega,\boldsymbol{r})\boldsymbol{z}^{\top}(\omega,\boldsymbol{r})d\boldsymbol{r}\right](k)$$

Simplify to obtain a practical algorithm

- Truncate and delay weighting filter to obtain casual linear phase filter.

$$\bar{\boldsymbol{A}}(i) = \begin{cases} \boldsymbol{A}(i-K) & i \in \{0,\ldots,2K\} \\ \boldsymbol{0} & \text{otherwise} \end{cases}$$

- Delay error signal by K samples to obtain correct cross-correlation.
- If secondary path is estimated offline, it can be combined ahead of time with the weighting filter to save computations.

$$\boldsymbol{H}(i) = \sum_{j=0}^{2K} \bar{\boldsymbol{A}}^{\top}(j)\boldsymbol{G}(i-j)$$

- Update filter with instantaneous value instead of expected value.

$$\boldsymbol{W}_{n+1}(i) = \boldsymbol{W}_n(i) - \mu \sum_{j=0}^{J+2K-1} \boldsymbol{H}^{\top}(j)\boldsymbol{e}(n-K)\boldsymbol{x}^{\top}(n-i-j)$$

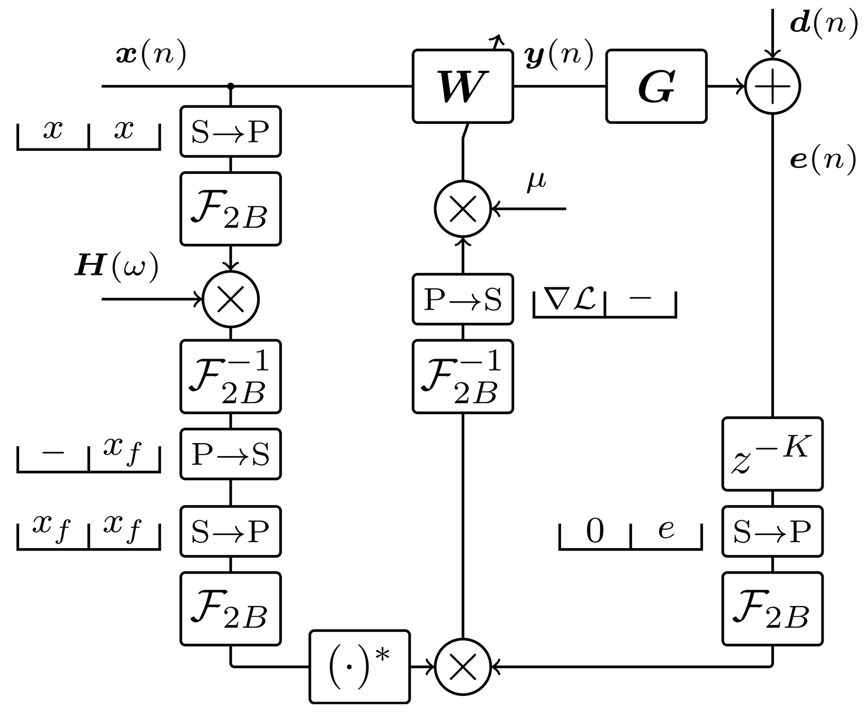
Fast block KI-FxLMS

Block-based implementation to reduce computational cost

- Update filter once per block.
- Compute convolutions and correlations with overlap-save and fast Fourier transform.
- Control filter is obtained in time domain to easily maintain causality.
- Loudspeaker signals are generated every sample.

More computationally efficient than KI-FxLMS when the control filter length is over 32

- Block length and control filter length is B.
- Fast Fourier transform (FFT) are performed for sequences of length 2B.
- Frequency domain filters are obtained as the FFT of B filter coefficients followed by B zeroes.
- Serial to parallel (S→P) operation concatenates 2 blocks worth of samples.
- Parallel to serial (P→S) operation discards one blocks worth of samples.



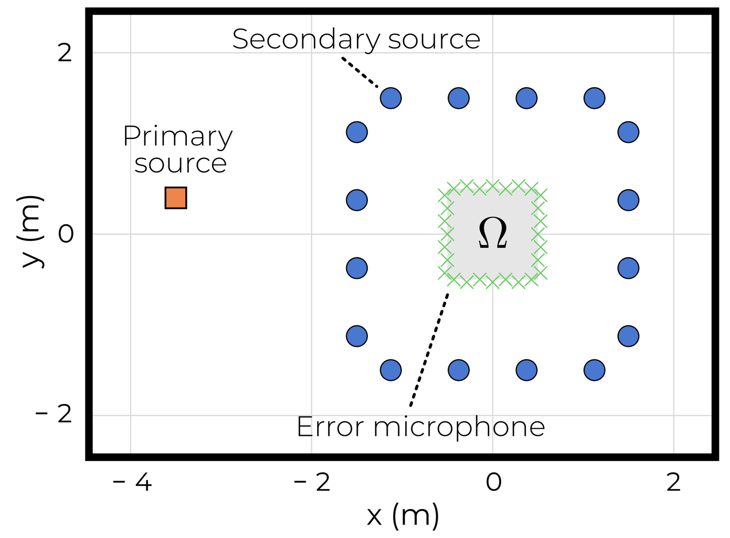
Evaluation

- Comparison between FxLMS and KI-FxLMS

Simulation in reverberant space

- 28 error microphones
- 16 loudspeakers
- Reference signal obtained directly

$\mathrm{RT}_{60} \approx 0.35\,\mathrm{s}$



Source types

- **Noise** - White noise bandlimited to 100-500 Hz
- **Song** - *High Horse* by artist *Secret Mountains*
- **Instrumental** - Instrumental version of **Song**

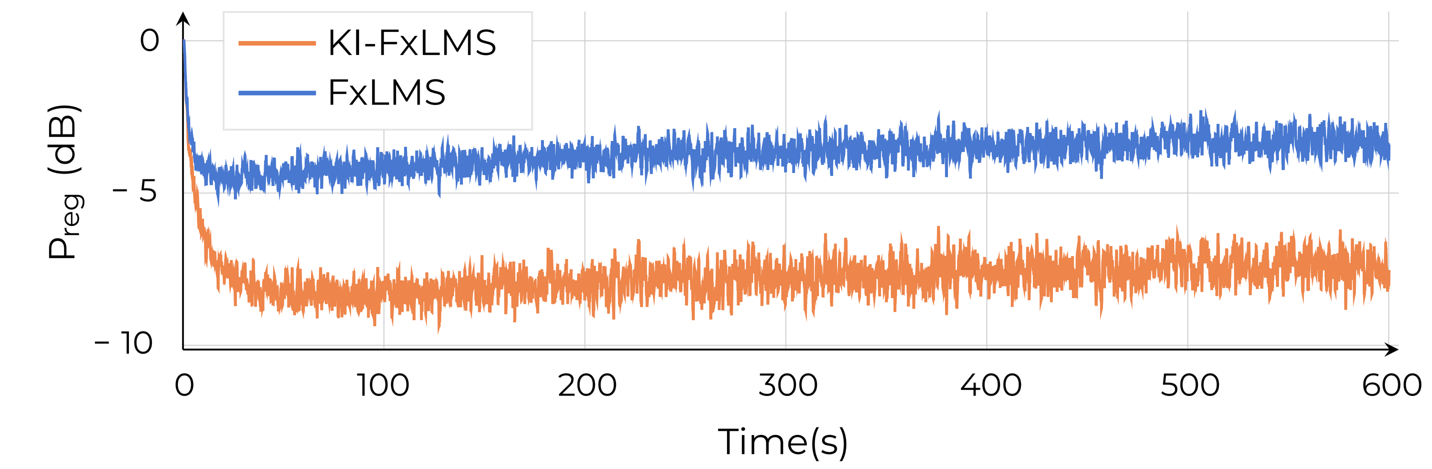
Regional power reduction

- Measured by discretizing the region of interest with 400 equally spaced points.

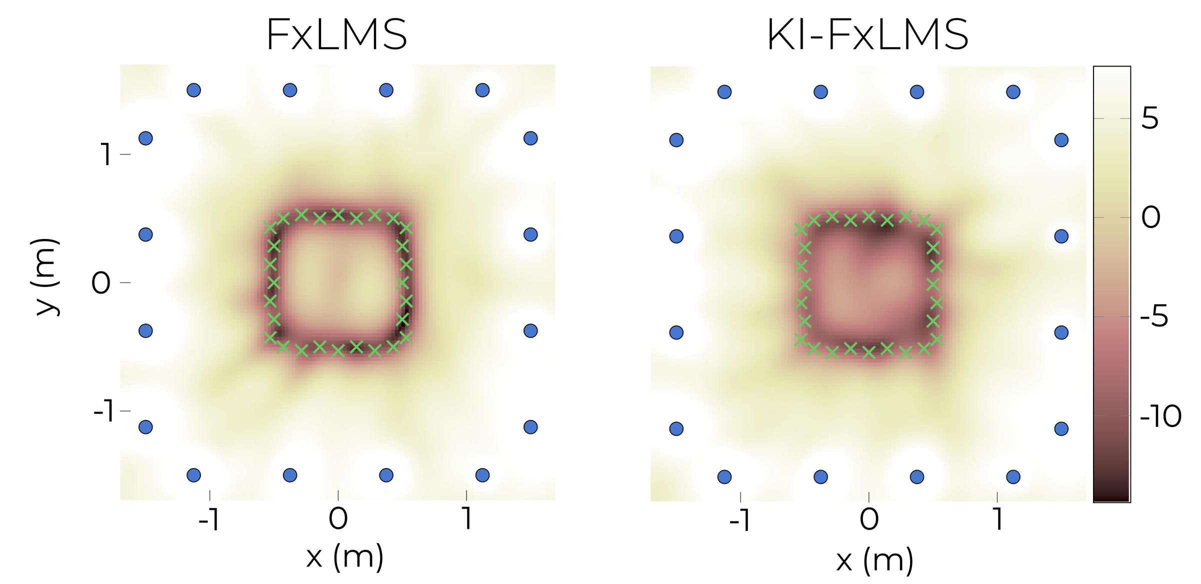
$$P_{\mathrm{reg}}(n) = \frac{\sum_{\nu}\sum_{\tau} u(n-\tau,\boldsymbol{r}_{\nu})^2}{\sum_{\nu}\sum_{\tau} \underbrace{u_{\mathrm{p}}(n-\tau,\boldsymbol{r}_{\nu})}_{\text{Primary noise sound pressure}}{}^2}$$

Results

Source type Noise



- Power distribution in dB after 600 seconds



All source types

- Average regional power reduction in dB between 120 and 240 seconds

	Noise	Song	Instrumental
FxLMS	-3.81	-3.28	-3.13
Fast Block FxLMS	-3.88	-3.17	-3.12
KI-FxLMS	-7.88	$\mathbf{-3.99}$	$\mathbf{-4.39}$
Fast Block KI-FxLMS	$\mathbf{-8.00}$	-3.60	-4.08