Adaptive Filtering for Audio Interferer Removal

Wiener, Steepest Descent, LMS and NLMS

Fabio Antonacci

Feb. 19, 2026

Audio interferer removal: problem statement

- Observed (primary) signal:

$$d(n) = s(n) + v(n)$$

- $s(n)$: desired audio (speech, music)
- $v(n)$: interference (noise, hum, disturbance)

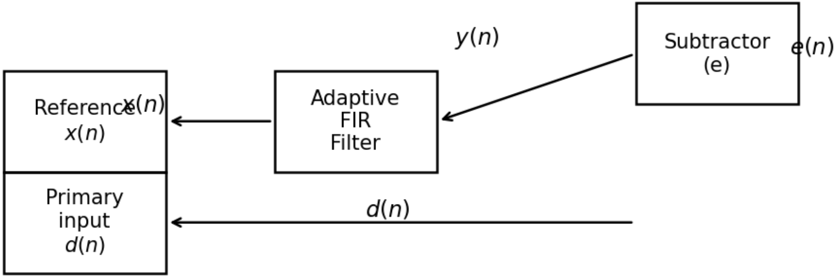
Goal: recover an estimate of $s(n)$ from the observed mixture $d(n)$.

Reference signal and key assumption

- A reference signal is available:
$$x(n) \approx v(n)$$
- The reference is correlated with the interference
- The reference is *uncorrelated* with the desired signal:
$$E\{s(n)x(n-k)\} = 0$$

This assumption underlies adaptive noise cancellation.

Adaptive noise canceller structure



- Adaptive FIR filter models the interference path
- Filter output estimates the interference
- Subtraction yields enhanced audio

Signal model (vector form)

Reference vector:

$$\boldsymbol{x}(n) = \begin{bmatrix} x(n) & x(n-1) & \cdots & x(n-M+1) \end{bmatrix}^T$$

Filter output:

$$y(n) = \boldsymbol{w}^T \boldsymbol{x}(n)$$

Error (system output):

$$e(n) = d(n) - y(n)$$

Common objective: MMSE

All approaches minimize the same cost:

$$J(\boldsymbol{w}) = E\{e^2(n)\} = E\{(d(n) - \boldsymbol{w}^T \boldsymbol{x}(n))^2\}$$

This defines a quadratic surface in the coefficient space.

Second-order statistics

Define:

$$\boldsymbol{R}_{xx} = E\{\boldsymbol{x}(n)\boldsymbol{x}^T(n)\} \quad \boldsymbol{r}_{xd} = E\{\boldsymbol{x}(n)d(n)\}$$

These quantities summarize the correlation structure of the data.

Wiener filter: optimal solution

Setting the gradient of $J(\boldsymbol{w})$ to zero:

$$\boldsymbol{R}_{xx}\boldsymbol{w}_{\text{opt}} = \boldsymbol{r}_{xd}$$

Closed-form solution:

$$\boldsymbol{w}_{\text{opt}} = \boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xd}$$

Wiener filter: interpretation

- Optimal linear estimator in the MMSE sense
- Requires knowledge (or estimation) of statistics
- Assumes stationarity over the estimation interval

Useful as a theoretical reference, but impractical for many real-time audio problems.

Why iterative adaptive methods?

- Statistics are often unknown or time-varying
- Audio environments change (sources, paths, gains)
- We want online, low-latency adaptation

Idea: minimize $J(\boldsymbol{w})$ iteratively using gradient information.

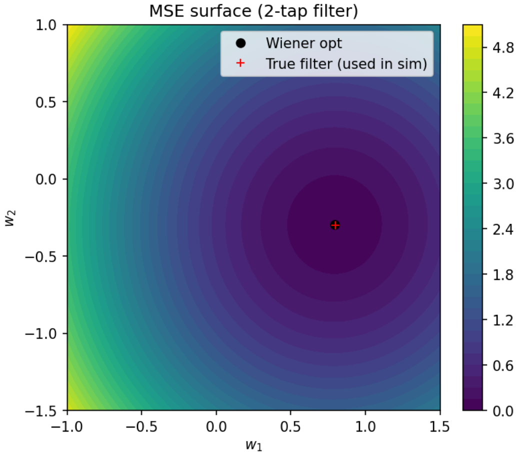


Figure: Surface contour of an examplary $J(\mathbf{w})$)

Steepest descent algorithm

Gradient of the MSE cost:

$$\nabla J(\boldsymbol{w}) = -2\boldsymbol{r}_{xd} + 2\boldsymbol{R}_{xx}\boldsymbol{w}$$

Update rule:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \mu(\boldsymbol{r}_{xd} - \boldsymbol{R}_{xx}\boldsymbol{w}(k))$$

Steepest descent: role and limitations

- Converges to the Wiener solution
- Requires full second-order statistics
- Convergence speed depends on eigenvalue spread of $\boldsymbol{R}_{xx}$

Main value: conceptual bridge to LMS.

From steepest descent to LMS

Replace expectations by instantaneous estimates:

$$\boldsymbol{r}_{xd} \to d(n)\boldsymbol{x}(n), \quad \boldsymbol{R}_{xx}\boldsymbol{w} \to \boldsymbol{x}(n)\boldsymbol{x}^T(n)\boldsymbol{w}$$

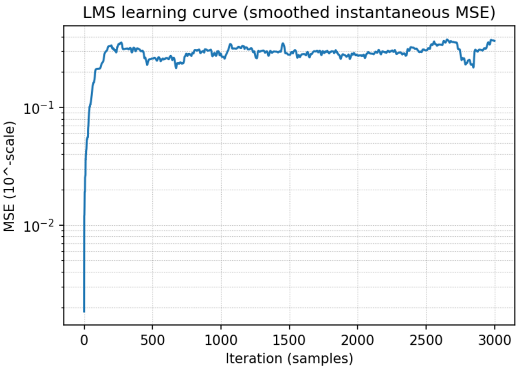
Instantaneous gradient approximation:

$$\nabla J(\boldsymbol{w}) \approx -2e(n)\boldsymbol{x}(n)$$

LMS update equation

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu e(n)\boldsymbol{x}(n)$$

- Simple
- Online
- Low computational cost



LMS behavior and step size

- Step size μ controls:
  - convergence speed
  - steady-state misadjustment
- Stability depends on input signal power

LMS can become unstable when the reference amplitude changes.

Motivation for NLMS

Problem with LMS:

- Same μ used regardless of input energy
- Large input power $\Rightarrow$ too aggressive updates
- Small input power $\Rightarrow$ slow learning

Solution: normalize by instantaneous input energy.

NLMS update equation

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \frac{\alpha}{\varepsilon + \|\boldsymbol{x}(n)\|^2} e(n)\boldsymbol{x}(n)$$

- α: normalized step size
- ε: regularization term

NLMS: interpretation

- Effective step size adapts automatically
- Robust to sudden changes in reference level
- Often faster and more stable than LMS in practice

LMS vs NLMS

	LMS	NLMS
Step size	fixed μ	normalized
Sensitivity to input power	high	low
Typical convergence	slower	faster
Complexity	$O(M)$	$O(M)$

Take-home messages

- All adaptive filters minimize the same MSE objective.
- Wiener filtering gives the optimal offline solution.
- Steepest descent explains how adaptation reaches Wiener.
- LMS enables online, low-cost adaptation.
- NLMS improves robustness by normalizing the update.
- For audio applications, NLMS is often the safest default.