# Introduction to Adaptive Noise Cancellation

Christopher Bekos (mpekchri@auth.gr)

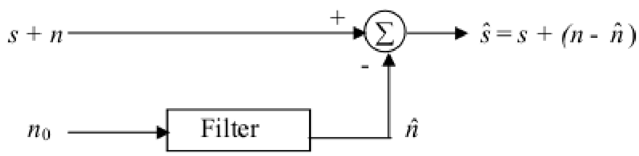
September 1, 2017

## I. Filtering

You can imagine a filter as a black box which may have multiple inputs and outputs . In noise cancellation we will deal with filter with signal input/output .
Our filter has an input signal s + $\nu$ and it's goal is to construct an outputsignal $\tilde{s}$,
in a way that is valid : $s \approx \tilde{s}$ .
Adaptive filters also use their output as feedback in order to adapt their characteristics.

## II. Noise Cancellation

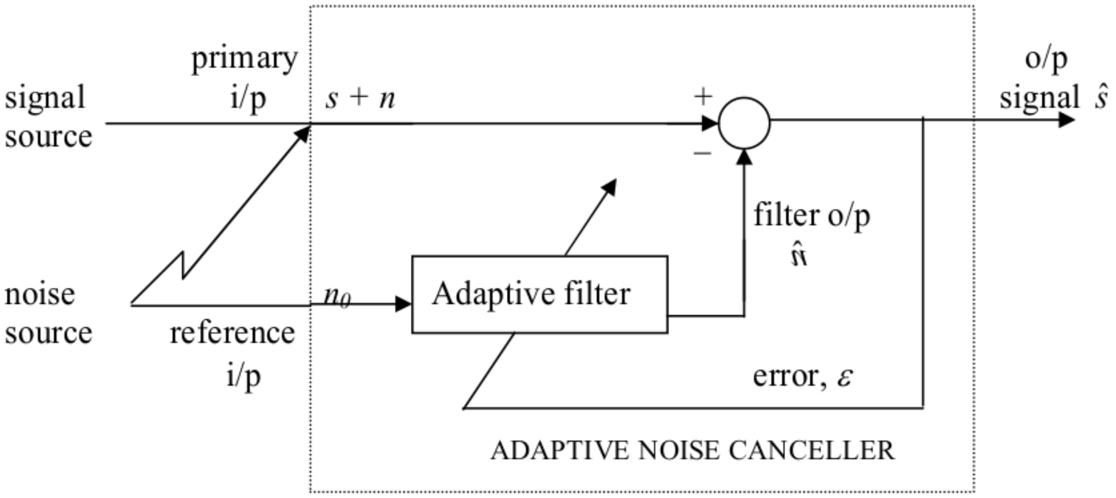
Noise Cancellation is a variation of optimal filtering that involves producing an estimate of the noise by filtering the reference input and then subtracting this noise estimate from the primary input containing both signal and noise.
It makes use of an auxiliary or reference input which contains a correlated estimate of the noise to be cancelled. The reference can be obtained by placing one or more sensors in the noise field where the signal is absent or its strength is weak enough. Subtracting noise from a received signal involves the risk of distorting the signal and if done improperly, it may lead to an increase in the noise level. This requires that the noise estimate $\tilde{n}$ should be an exact replica of n. If it were possible to know the relationship between n and $\tilde{n}$, or the characteristics of the channels transmitting noise from the noise source to the primary and reference inputs are known, it would be possible to make $\tilde{n}$ a close estimate of n by designing a fixed filter. However, since the characteristics of the transmission paths are not known and are unpredictable, filtering and subtraction are controlled by an adaptive process. Hence an adaptivefilter is used that is capable of adjusting its impulse response to minimize an error signal, which is dependent on the filter output. The adjustment of the filter weights, and hence the impulse response, is governed by an adaptive algorithm. With adaptive control, noise reduction can be accomplished with little risk of distorting the signal. In fact, Adaptive Noise Canceling makes possible attainment of noise rejection levels that are difficult or impossible to achieve by direct filtering.




The error signal to be used depends on the application. The criteria to be used may be the minimization of the mean square error, the temporal average of the least squares error etc. Different algorithms are used for each of the minimization criteria e.g. the Least Mean Squares (LMS) algorithm, the Recursive Least Squares (RLS) algorithm etc.

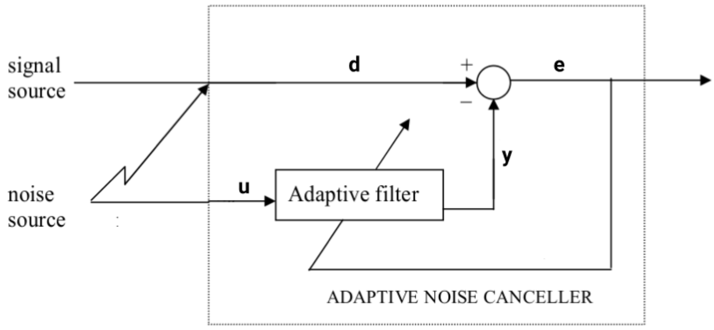
## III. Adaptive Noise Cancellation

An adaptive noise canceler is showed in figure bellow . As said before , in cases that we are able to obtain an estimate of noise , $\hat{n}$ , then we use this signal as input to our filter . Filter's output is added to desired signal ( s + n ) and the result (error) is used to train our filter .




**An Adaptive Noise Canceler**

**Filter Training**
Training our filter aims to find the optimal filter parameters , which can be done by solving Wiener-Hopf equations . Let's examine a FIR filter and the system as showed in the above figure . we name the signal **s + n** as **d (desired)** , filter's input as **u** , filter;s output as **y** and we consider the error signal as **e = d - y** . Thus , our system is now described by the figure :




In case of 1-D signals , we may think of d,u,y and e as vectors of N elements . Also our filter can be defined as a vector w of length M . Our goal is to define the $w_0, w_1, ..., w_{M-1}$ parameters in a way that MSE(e) becomes minimum . Filter is implemented using the equation :

$$y(n) = \sum_{k=0}^{M-1} w_k * u(n-k)$$

In such a case **Wiener-Hopf equation** looks like : R * $\boldsymbol{w_o}$ = **p** .
Here , $\boldsymbol{w_o}$ is the optimal selection of our filter parameters , and of course it's a vector of length M .
Let us consider $r_u(a-b)$ = E[u(n-b)$u^*(n-a)$] and $p(-k)$ = E[$d^*(n)$u(n-k)] ,
then vector p and matrix R , are defined as :

$$\mathbf{p} = \begin{bmatrix} p(0) & p(-1) & p(-2) & \ldots & p(1-M) \end{bmatrix}, \mu\varepsilon \quad \mathbf{R} = \begin{bmatrix} r_u(0) & r_u(1) & \ldots & r(M-2) & r(M-1) \\ r_u(1) & r_u(0) & \ldots & r(M-3) & r(M-2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ r_u(M-2) & r_u(M-3) & \ldots & r(0) & r(1) \\ r_u(M-1) & r_u(M-2) & \ldots & r(1) & r(0) \end{bmatrix}$$

Concluding , we may find an optimal filter $w_o$ , by solving $w_o = R^{-1}$ * p ,if R can be reversed.
Although , estimating matrix R and vector p , and reversing R matrix turn out to cost much time and cpu power . In order to overcome such problems , smarter algorithms (as mentioned above) have been constructed . All those algorithms use some techniques in order to converge to a solution way too close to $w_o$ , but as said before , if there isn't such a solution (R cannot be reversed) , they fail to converge .