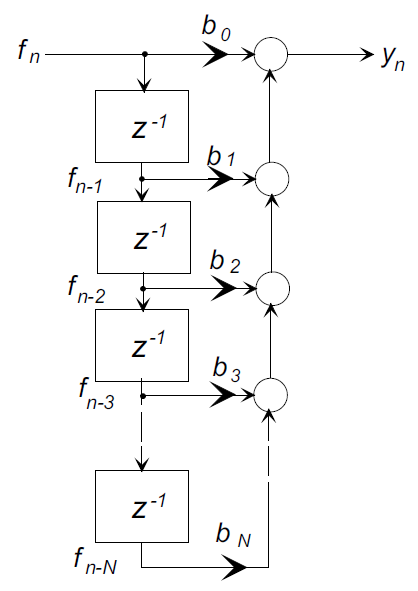
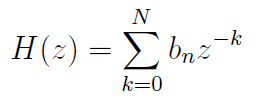
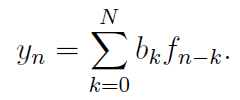
**Лекция 2. Фильтры Винера и LMS**



# 1. ких фильтр

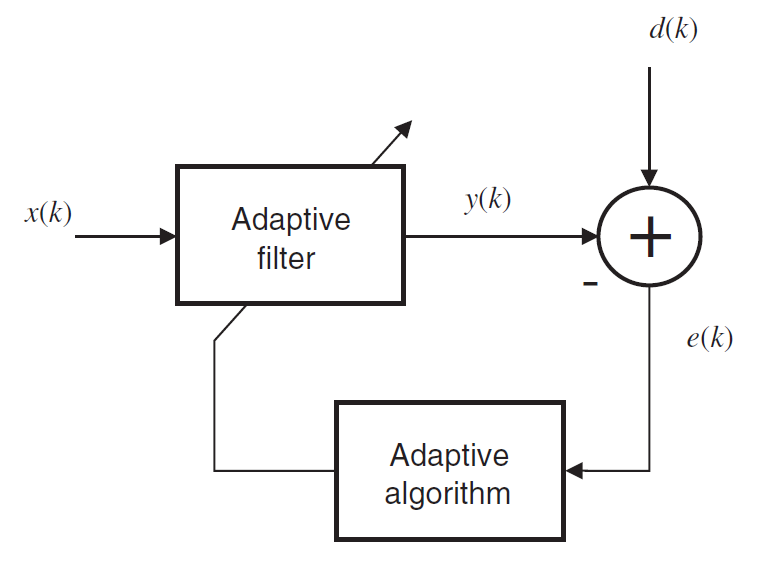
В прошлый раз мы познакомились с КИХ фильтром, узнали, как считать его выход при известных входах и параметрах, как составлять его схему.

Теперь нужно научиться подбирать параметры фильтра для решения некоторых задач цифровой обработки сигналов, т.е. обучать (синтезировать, адаптировать) фильтр.

# 2. Адаптивные фильтры

Если параметры фильтра автоматически подбираются под решение задачи – такой фильтр называют **адаптивным**.

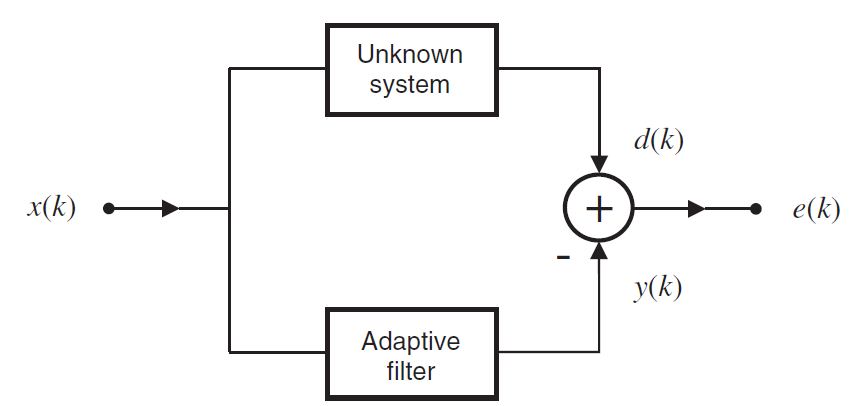


*КНИГА1: Paulo S.R. Diniz. Adaptive Filtering Algorithms and Practical Implementation. Springer, 2008, DOI: 10.1007/978-0-387-68606-6*

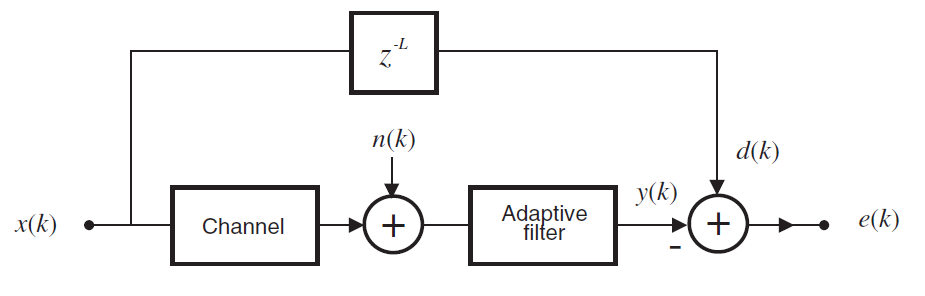
*КНИГА2: Poularikas A. D., Ramadan Z. M. Adaptive filtering primer with MATLAB. – CRC Press, 2006.*

# 3. Некоторые постановки задач цифровой обработки сигналов

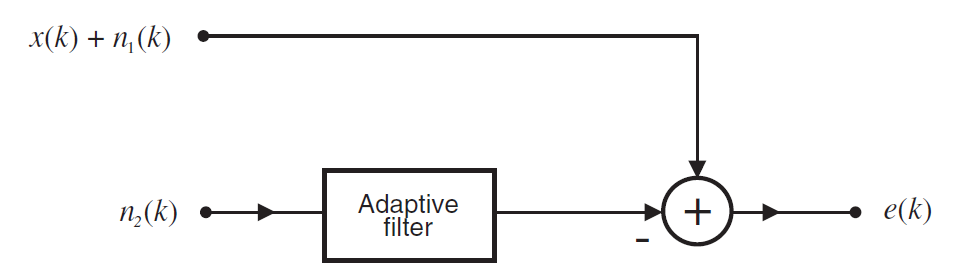
## А) Идентификация системы



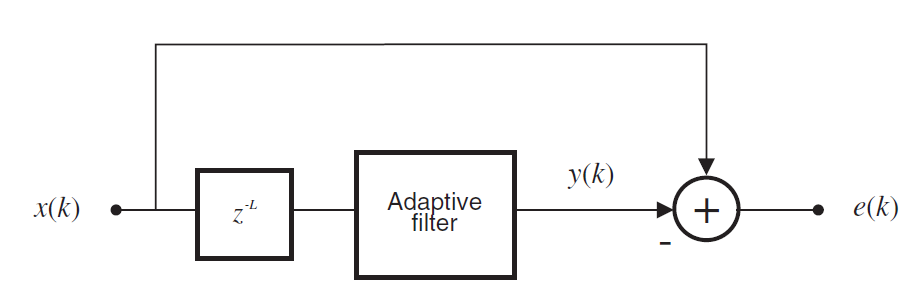
## Б) Эквалайзер (выравниватель) канала связи



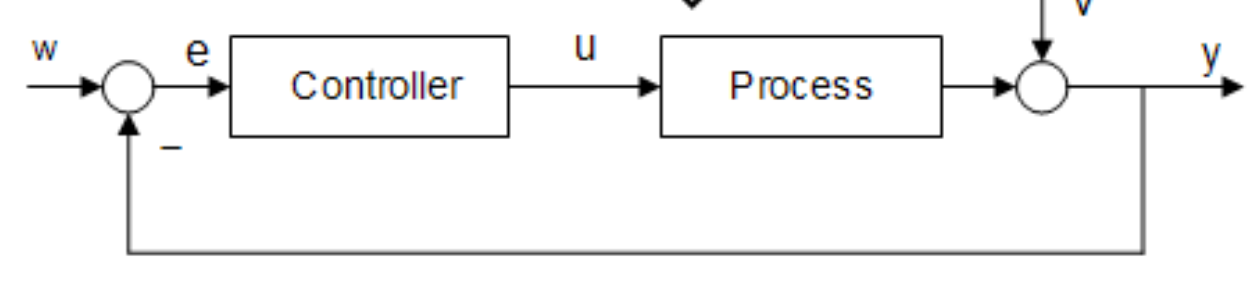
## В) Подавление шума



## Г) Предсказание сигнала



## Д) Управление



И другие…

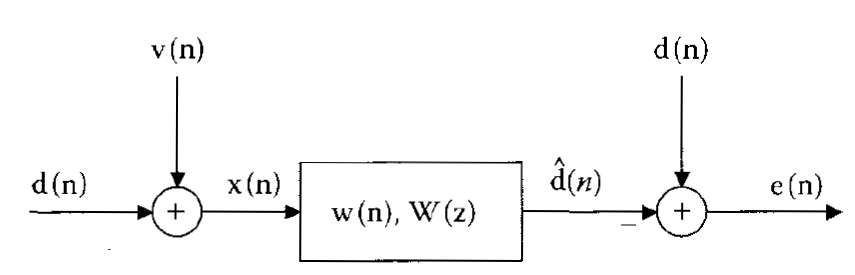
# 4. Фильтр Винера

Задача любого обучения – подобрать параметры так, чтобы уменьшить ошибку.

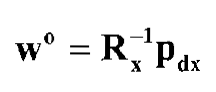
Будем рассматривать сигналы на входе и выходе объекта как случайные процессы. Тогда можем потребовать, чтобы **средний квадрат отклонения** выхода фильтра от выхода модели был как можно меньше для всех заданных входов – это функция ошибки.

Выход фильтра – взвешенная сумма отсчетов входного сигнала, с некоторыми, неизвестными весами. Из условия минимизации ошибки можно вывести «оптимальные» значения весов (параметров КИХ фильтра).

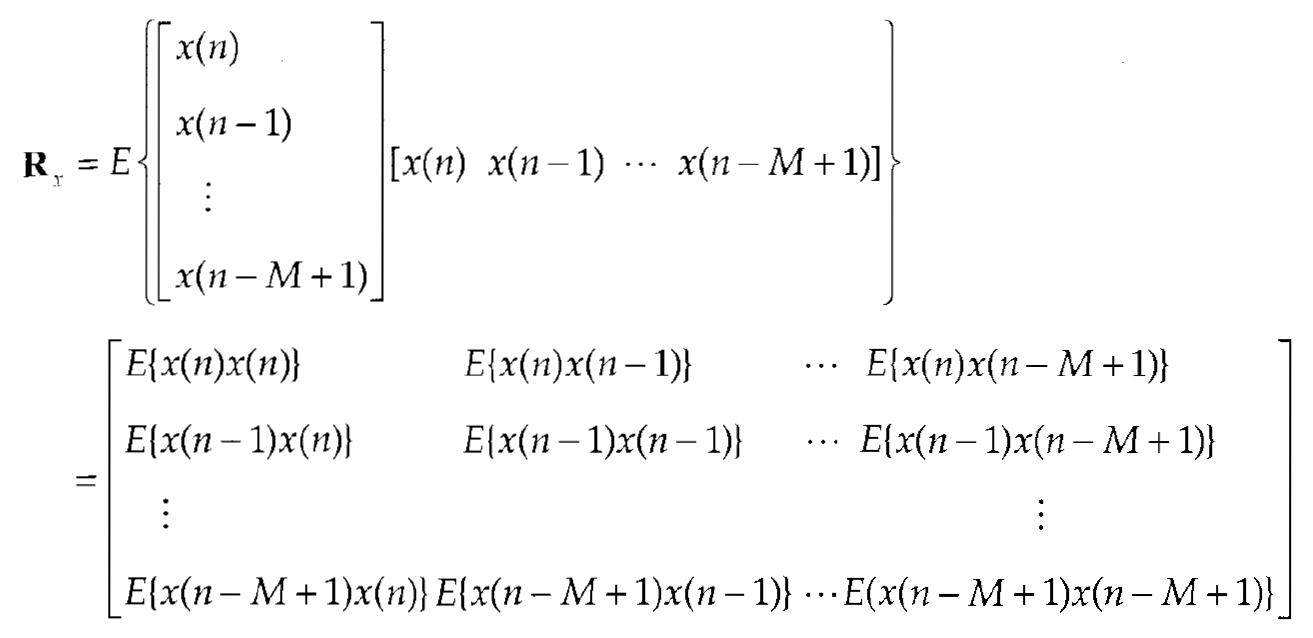
Посмотрите здесь <https://www.youtube.com/watch?v=LYvAgAeAGcQ> или в п.4.2 Книги2.

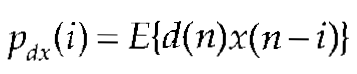


Получится, что оптимальные веса, должны рассчитываться как:



здесь p – вектор взаимной корреляции между входным и желаемым выходным сигналом фильтра, R- матрица автокорреляции входного сигнала.





Мат. ожидания можно оценить как средние значения, зная входной сигнал и желаемый выходной.

Пример: пусть в фильтре 3 веса и

x=1, 2, 3, 4, 5, 6, 7, 8, 9

d=9, 8, 7, 6, 5, 4, 3, 2, 1

pdx(0) = (9\*1+8\*2+7\*3+6\*4+5\*5+4\*6+3\*7+2\*8+1\*9)/9

pdx(1) = (8\*1+7\*2+6\*3+5\*4+4\*5+3\*6+2\*7+1\*8)/8

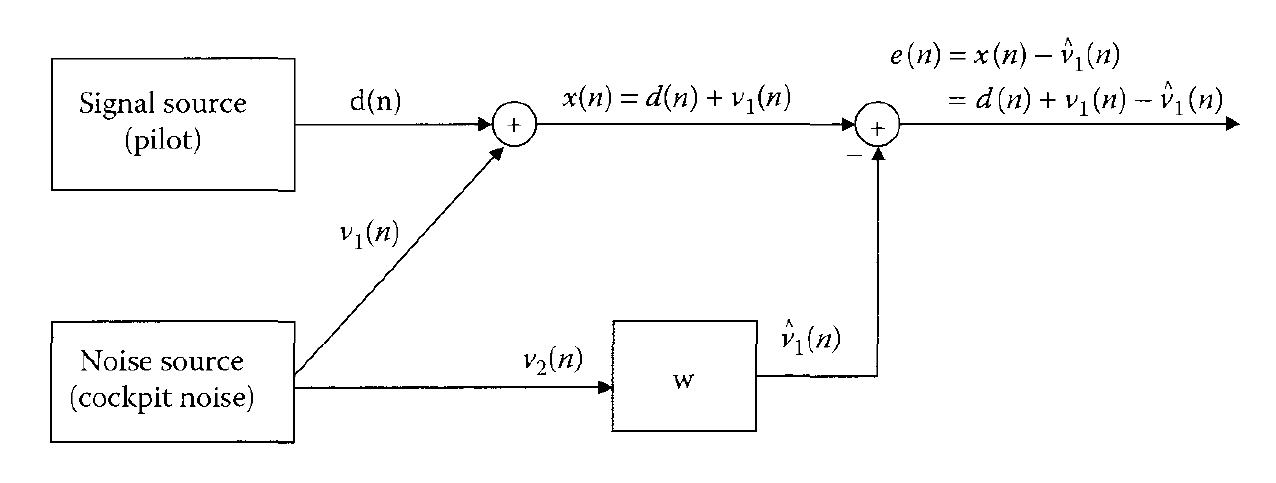
pdx(2) = (7\*1+6\*2+5\*3+4\*4+3\*5+2\*6+1\*7)/7

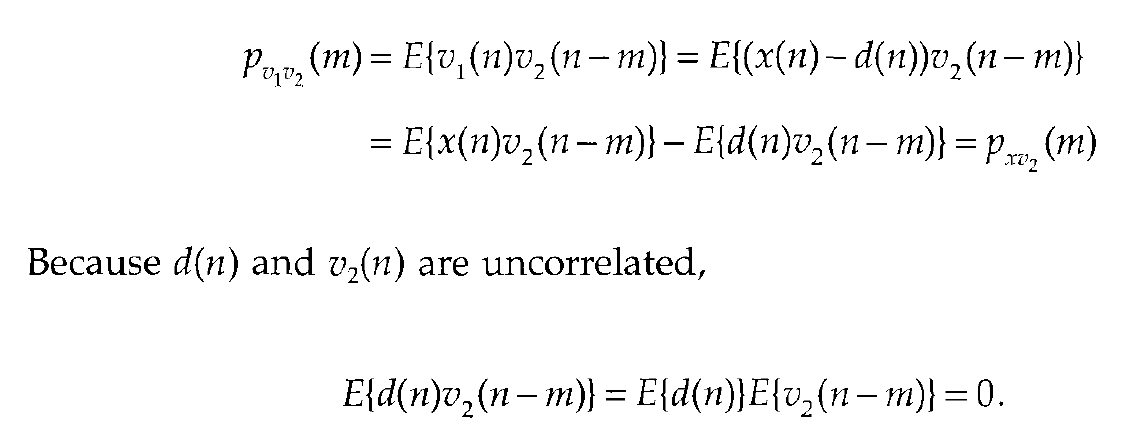
R(1,1)=(1\*1+2\*2+3\*3+4\*4+5\*5+6\*6+7\*7+8\*8+9\*9)/9

R(1,2)=R(2,1)=(2\*1+3\*2+4\*3+5\*4+6\*5+7\*6+8\*7+9\*8)/8

….

Вернемся к задачам…………



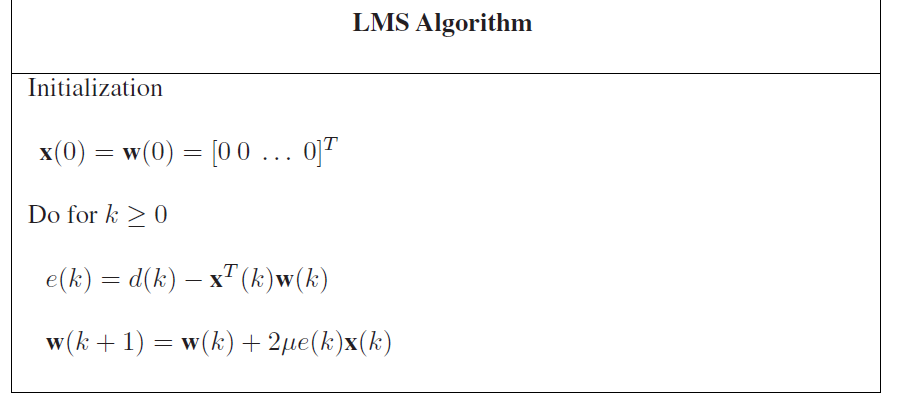


# 5. LMS фильтр

Если вместо среднего квадрата отклонений считать квадрат мгновенного отклонения, то придем к идее LMS фильтра.

Будем подбирать параметры итеративно, чтобы уменьшать такую ошибку.

По сути, это метод градиентного спуска.



<https://www.youtube.com/watch?v=TykEeAM6v9U>

<https://www.youtube.com/watch?v=2JgoeuM7iVM>

Пример: >> openExample('dsp/SystemIdentificationWithLMSFilterExample')

Отличия фильтра Винера от LMS:

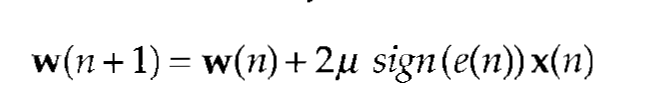
(напишите)

- ошибка разная

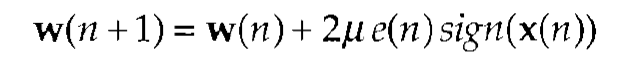
- веса изменяются сразу – винер, по шагам - лмс

# Модификации LMS фильтра

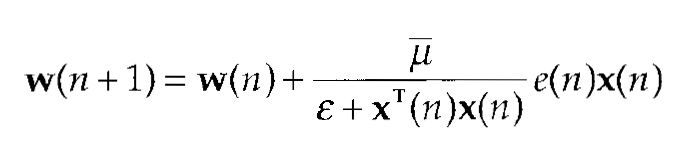
Знак ошибки

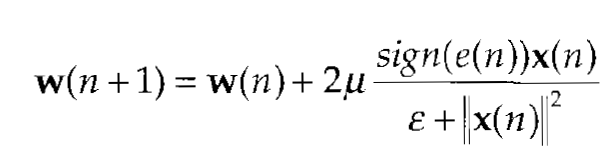


Знак входа

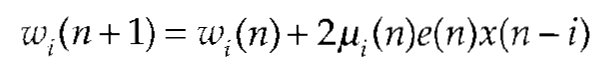


Нормализованный

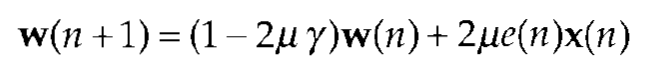


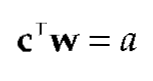


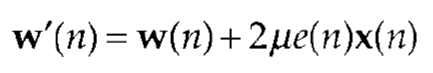
С изменяемым шагом

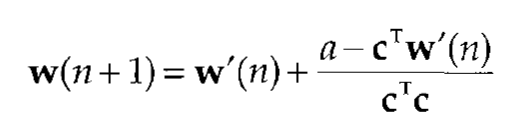


С «утечкой»

С ограничениями 





И прочие, прочие, прочие, см. Книгу 2.

На этом мы закончим с линейными фильтрами, а в следующий раз познакомимся с нелинейными.