Least Squares Estimation

Model

\[ y(k) = \sum_{i=1}^{n} \phi_i(k - 1) \theta_i \]

Where

- \( y(k) \) observed output
- \( \phi_i(k) \) known and measurable function
- \( \theta_i \) unknown but constant parameter
Least Squares Estimation

Model

\[ y(k) = \phi^T(k - 1) \theta \]

Where

\[ y(k) \] measured output

\[ \phi(k) = \begin{bmatrix} \phi_1(k) \\ \vdots \\ \phi_n(k) \end{bmatrix} \]

\[ \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} \]

\( n \times 1 \) regressor

unknown vector
Batch Least Squares Estimation

Assume that we have collected $k$ data sets:

$y(1), \ldots, y(k)$

$\phi(0), \ldots, \phi(k - 1)$

We want to find the parameter estimate at instant $k$: $\hat{\theta}(k)$

that best fits all collected data in the least squares sense:

$$\min_{\hat{\theta}(k)} \left\{ \frac{1}{2} \sum_{j=1}^{k} \left[ y(j) - \phi^T(j - 1) \hat{\theta}(k) \right]^2 \right\}$$

kept constant in the summation
Batch Least Squares Estimation

Defining the cost functional

\[ V(\hat{\theta}(k)) = \frac{1}{2} \sum_{j=1}^{k} \left[ y(j) - \phi^T(j - 1) \hat{\theta}(k) \right]^2 \]

\( \hat{\theta}(k) \) is obtained by solving

\[ \frac{dV(\hat{\theta}(k))}{d\hat{\theta}(k)} = 0 \]
Batch Least Squares Solution

The least squares parameter estimate \( \hat{\theta}(k) \) which solves

\[
\frac{dV(\hat{\theta}(k))}{d\hat{\theta}(k)} = 0
\]

Satisfies the **normal equation**: 

\[
\left[ \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) \right] \hat{\theta}(k) = \sum_{i=1}^{k} \phi(i-1)y(i)
\]

\[\text{\(n \times n\) matrix} \quad \text{\(n \times 1\) vector}\]
Normal Equation Derivation

\[ V(\hat{\theta}(k)) = \frac{1}{2} \sum_{j=1}^{k} \left[ y(j) - \phi^T(j - 1) \hat{\theta}(k) \right]^2 \]

\[ = \frac{1}{2} \left\| \begin{bmatrix}
y(1) - \phi^T(0) \hat{\theta}(k) \\
\vdots \\
y(k) - \phi^T(k - 1) \hat{\theta}(k) 
\end{bmatrix} \right\|^2 
\]

\[ = \frac{1}{2} \left\| \begin{bmatrix}
y(1) \\
\vdots \\
y(k) 
\end{bmatrix} - \begin{bmatrix}
\phi^T(0) \\
\vdots \\
\phi^T(k - 1) 
\end{bmatrix} \hat{\theta}(k) \right\|^2 
\]

\[ = \frac{1}{2} \left\| Y(k) - \Phi^T(k - 1) \hat{\theta}(k) \right\|^2 
\]
Normal Equation Derivation

\[ V(\hat{\theta}(k)) = \frac{1}{2} \left\| Y(k) - \Phi^T(k - 1)\hat{\theta}(k) \right\|^2 \]

\[ = \frac{1}{2} \left[ Y^T(k)Y(k) + \hat{\theta}^T(k)\Phi(k - 1)\Phi^T(k - 1)\hat{\theta}(k) - 2\hat{\theta}^T(k)\Phi(k - 1)Y(k) \right] \]

Taking the partial derivative with respect to \( \hat{\theta}(k) \)

\[ \frac{\partial V(\hat{\theta}(k))}{\hat{\theta}(k)} = \Phi(k-1)\Phi^T(k-1)\hat{\theta}(k) - \Phi(k-1)Y(k) \]

For optimality, we therefore need

\[ \Phi(k - 1)\Phi^T(k - 1)\hat{\theta}(k) = \Phi(k - 1)Y(k) \]
Normal Equation Derivation

\[ \Phi(k - 1) = \begin{bmatrix} \phi(0) & \cdots & \phi(k - 1) \end{bmatrix} \]

\[ Y(k) = \begin{bmatrix} y(1) & \cdots & y(k) \end{bmatrix}^T \]

For optimality, we need

\[ \Phi(k - 1) \Phi^T(k - 1) \hat{\theta}(k) = \Phi(k - 1) Y(k) \]

\[ \sum_{i=1}^{k} \phi(i - 1) \phi^T(i - 1) \]

\[ \sum_{i=1}^{k} \phi(i - 1) y(i) \]

Therefore, we need

\[ \hat{\theta}(k) = \sum_{i=1}^{k} \phi(i - 1) y(i) \]
Batch Least Squares Estimation

The solution of the normal equation

\[ \sum_{i=1}^{k} \phi(i - 1) \phi^T(i - 1) \] \( \hat{\theta}(k) = \sum_{i=1}^{k} \phi(i - 1) y(i) \)

Is given by:

\[ \hat{\theta}(k) = \left[ \sum_{i=1}^{k} \phi(i - 1) \phi^T(i - 1) \right] \# \sum_{i=1}^{k} \phi(i - 1) y(i) \]

Pseudoinverse
Moore-Penrose pseudoinverse

• Let $A$ have the singular value decomposition

$$A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$$

$\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_r)$  \hspace{1cm} $\sigma_1 \geq \cdots \geq \sigma_r > 0$

• Then the Moore-Penrose pseudoinverse of $A$ is

$$A^\# = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix}$$

In MATLAB: \texttt{pinv}(A)
Moore-Penrose pseudoinverse

Let \( A \in \mathcal{R}^{n \times m} \) and \( A^\# \) be its Moore-Penrose pseudoinverse.

Then, \( A^\# \) has the dimension of \( A^T \) and satisfies:

- \( A A^\# A = A \)
- \( A^\# A A^\# = A^\# \)
- \( A^\# A \) and \( A A^\# \) are Hermitian

In this case, since \( A = \Phi \Phi^T \)

\[
\Phi = \begin{bmatrix} \phi(0) & \cdots & \phi(k-1) \end{bmatrix}
\]

\[
A A^\# \Phi = \Phi
\]
Batch Least Squares Estimation

Assume that we have collected sufficient data and the data has sufficient richness so that

$$\sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) = \phi(0)\phi^T(0) + \phi(1)\phi^T(1) + \cdots + \phi(k-1)\phi^T(k-1)$$

has full rank.

Then,

$$\hat{\theta}(k) = \left[ \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) \right]^{-1} \sum_{i=1}^{k} \phi(i-1) y(i)$$

$$F(k)$$
Recursive Least Squares (RLS)

Assume that we have collected \( k-1 \) sets of data and have computed \( \hat{\theta}(k-1) \) using

\[
\hat{\theta}(k-1) = \left[ \sum_{i=1}^{k-1} \phi(i-1)\phi^T(i-1) \right]^{-1} \sum_{i=1}^{k-1} \phi(i-1)y(i)
\]

\[
F(k-1)
\]

Then, given a new set of data: \( y(k) \) \( \phi(k-1) \)

We want to find \( \hat{\theta}(k) \) in a recursive fashion:

\[
\hat{\theta}(k) = \hat{\theta}(k-1) + [ \text{correction term} ]
\]
Recursive Least Squares Algorithm

Define the \textit{a-priori} output estimate:

\[
\hat{y}^o(k) = \phi^T(k - 1)\hat{\theta}(k - 1)
\]

and the \textit{a-priori} output estimation error:

\[
e^o(k) = y(k) - \phi^T(k - 1)\hat{\theta}(k - 1)
\]

The RLS algorithm is given by:

\[
\hat{\theta}(k) = \hat{\theta}(k - 1) + F(k)\phi(k - 1)e^o(k)
\]

where \(F(k)\) has the recursive relationship on the next slide
Recursive Least Squares Gain

The RLS gain $F(k)$ is defined by

$$F^{-1}(k) = \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)$$

Therefore,

$$F^{-1}(k) = F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

Using the matrix inversion lemma, we obtain

$$F(k) = F(k-1) - \frac{F(k-1) \phi(k-1) \phi(k-1)^T F(k-1)}{1 + \phi(k-1)^T F(k-1) \phi(k-1)}$$
Recursive Least Squares Derivation

Define the least squares gain matrix \( F(k) \)

\[
F(k) = \left[ \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) \right]^{-1}
\]

Therefore,

\[
\hat{\theta}(k) = F(k) \sum_{i=1}^{k} \phi(i-1)y(i)
\]
Recursive Least Squares Derivation

Notice that

\[ \hat{\theta}(k) = F(k) \sum_{i=1}^{k} \phi(i - 1)y(i) \]

\[ = F(k) \left[ \phi(k - 1)y(k) + \sum_{i=1}^{k-1} \phi(i - 1)y(i) \right] \]

\[ F^{-1}(k - 1)\hat{\theta}(k - 1) \]

\[ F^{-1}(k - 1) = F^{-1}(k) - \phi(k - 1)\phi^T(k - 1) \]
Recursive Least Squares Derivation

Therefore plugging the previous two results,

\[
\hat{\theta}(k) = F(k) \left[ \left( F(k)^{-1} - \phi(k - 1) \phi^T(k - 1) \right) \hat{\theta}(k - 1) \right.
\]

\[
+ \left. \phi(k - 1) y(k) \right]
\]

And rearranging terms, we obtain

\[
\hat{\theta}(k) = \hat{\theta}(k - 1)
\]

\[
+ F(k) \phi(k - 1) \left[ y(k) - \phi^T(k - 1) \hat{\theta}(k - 1) \right]
\]

\[
\underbrace{\varepsilon_o(k)}_{e_o(k)}
\]
Recursive Least Squares Estimation

Define the \textit{a-priori} output estimate:

\[
\hat{y}^o(k) = \phi^T(k - 1)\hat{\theta}(k - 1)
\]

and the \textit{a-priori} output estimation error:

\[
e^o(k) = y(k) - \phi^T(k - 1)\hat{\theta}(k - 1)
\]

The RLS algorithm is given by:

\[
\hat{\theta}(k) = \hat{\theta}(k - 1) + F(k)\phi(k - 1) e^o(k)
\]
Recursive Least Squares Estimation

Recursive computation of $F(k)$

$$F^{-1}(k) = \sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)$$

Therefore,

$$F^{-1}(k) = F^{-1}(k-1) + \phi(k-1)\phi^T(k-1)$$

Using the matrix inversion lemma, we obtain

$$F(k) = F(k-1) - \frac{F(k-1) \phi(k-1) \phi(k-1)^T F(k-1)}{1 + \phi(k-1)^T F(k-1) \phi(k-1)}$$
Recursive Least Squares Estimation

Matrix inversion lemma:

\[
F^{-1}(k) = F^{-1}(k - 1) + \phi(k - 1)\phi^T(k - 1)
\]

- Multiply by \(F(k - 1)\) on the right and \(F(k)\) on the left:

\[
F(k - 1) = F(k) + F(k)\phi(k - 1)\phi(k - 1)^T F(k - 1)
\]

- Multiply by \(\phi(k - 1)\) on the right:

\[
F(k - 1)\phi(k - 1) = F(k)\phi(k - 1) + F(k)\phi(k - 1)\phi(k - 1)^T F(k - 1)\phi(k - 1)
\]

[scalar]
Recursive Least Squares Estimation

Matrix inversion lemma:

• Rearranging terms,

\[ F(k - 1)\phi(k - 1) = \left[ 1 + \phi(k - 1)^T F(k - 1)\phi(k - 1) \right] F(k)\phi(k - 1) \]

• Solving for \( F(k)\phi(k - 1) \)

\[ F(k)\phi(k - 1) = \frac{F(k - 1)\phi(k - 1)}{\left[ 1 + \phi(k - 1)^T F(k - 1)\phi(k - 1) \right]} \]
Recursive Least Squares Estimation

Matrix inversion lemma:

- Plug

\[ F(k) \phi(k - 1) = \frac{F(k - 1) \phi(k - 1)}{1 + \phi(k - 1)^T F(k - 1) \phi(k - 1)} \]

into

\[ F(k) = F(k - 1) - \frac{F(k - 1) \phi(k - 1) \phi(k - 1)^T F(k - 1)}{1 + \phi(k - 1)^T F(k - 1) \phi(k - 1)} \]

to obtain
RLS Estimation Algorithm

A-priori version:

\[
e^{o}(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k)
\]

\[
\hat{\theta}(k + 1) = \hat{\theta}(k) + F(k + 1)\phi(k)e^{o}(k + 1)
\]

\[
F(k + 1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)}
\]

Initial conditions:

\[
F(0) = F^T(0) > 0 \quad \hat{\theta}(0)
\]
RLS Estimation Algorithm

A-posteriori version (used to prove that $e(k) \to 0$):

$$e^o(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k)$$

$$e(k + 1) = \frac{e^o(k + 1)}{1 + \phi^T(k)F(k)\phi(k)}$$

$$\hat{\theta}(k + 1) = \hat{\theta}(k) + F(k)\phi(k)e(k + 1)$$

$$F(k + 1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)}$$
Define the parameter estimation error:

\[ \tilde{\theta}(k) = \theta - \hat{\theta}(k) \]

Notice that, since

\[ y(k) = \phi^T(k - 1)\theta \]

And the a-priori error is

\[ e^o(k) = y(k) - \phi^T(k - 1)\hat{\theta}(k - 1) \]

We obtain,

\[
e^o(k) = \phi^T(k - 1)\theta - \phi^T(k - 1)\hat{\theta}(k - 1)
\]

\[
= \phi^T(k - 1) \left[ \theta - \hat{\theta}(k - 1) \right]
\]

\[
= \phi^T(k - 1) \tilde{\theta}(k-1)
\]
RLS Estimation Algorithm

Thus, the a-priori output estimation error can be written as

\[ e^o(k) = \phi^T(k - 1)\tilde{\theta}(k - 1) \]

Similarly, define the \textbf{a-posteriori output and estimation error}:

\[ \hat{y}(k) = \phi^T(k - 1)\hat{\theta}(k) \]

\[ e(k) = y(k) - \hat{y}(k) \]

then,

\[ e(k) = \phi^T(k - 1)\tilde{\theta}(k) \]
RLS Estimation Algorithm

Derivation of the RLS A-posteriori version:

\[ \hat{\theta}(k) = \hat{\theta}(k - 1) + F(k)\phi(k - 1) e^o(k) \]

\[ e^o(k) = y(k) - \phi^T(k - 1)\hat{\theta}(k - 1) \]

Remember that,

\[ F(k)\phi(k - 1) = \frac{F(k - 1)\phi(k - 1)}{1 + \phi(k - 1)^T F(k - 1)\phi(k - 1)} \]

Thus,

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)} e^o(k + 1) \]
RLS Estimation Algorithm

Multiplying by \( \phi^T(k) \) to the left of

\[
\tilde{\theta}(k + 1) = \tilde{\theta}(k) - \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)} e^o(k + 1)
\]

to obtain,

\[
\frac{\phi^T(k)\tilde{\theta}(k + 1)}{e(k+1)} = \frac{\phi^T(k)\tilde{\theta}(k)}{e^o(k+1)} - \frac{\phi^T(k)F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)} e^o(k + 1)
\]

Thus,

\[
e(k + 1) = e^o(k + 1) - \frac{\phi^T(k)F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)} e^o(k + 1)
\]
\[
= \frac{e^o(k + 1)}{1 + \phi^T(k)F(k)\phi(k)}
\]
Therefore, from

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k)\phi(k) \frac{e^o(k + 1)}{1 + \phi^T(k)F(k)\phi(k)} \]

We obtain,

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k)\phi(k) e(k + 1) \]

\[ e(k + 1) = \frac{e^o(k + 1)}{1 + \phi^T(k)F(k)\phi(k)} \]
RLS with forgetting factor

The inverse of the gain matrix in the RLS algorithm is given by:

\[ F^{-1}(k) = F^{-1}(k - 1) + \phi(k - 1)\phi^T(k - 1) \]

Its trace is given by:

\[ \text{tr} \left[ F^{-1}(k) \right] = \text{tr} \left[ F^{-1}(k - 1) \right] + \|\phi(k - 1)\|^2 \]

which always increases when \( \|\phi(k - 1)\| \neq 0 \)
RLS with forgetting factor

Similarly, the trace of the gain matrix is given by

\[
\text{tr} [F(k)] = \text{tr} [F(k - 1)] - \frac{\|F(k - 1)\phi(k - 1)\|^2}{1 + \phi^T(k - 1)F(k - 1)\phi(k - 1)}
\]

always decreases when \( \|F(k - 1)\phi(k - 1)\| \neq 0 \)

Problem: RLS eventually stops updating
RLS with forgetting factor

We can modify cost function to “forget” old data

\[ V(\hat{\theta}(k)) = \frac{1}{2} \sum_{j=1}^{k} \lambda^{(k-j)} \left[ y(j) - \phi^T(j - 1) \hat{\theta}(k) \right]^2 \]

\[ 0 < \lambda \leq 1 \]

Key idea: Discount old data, e.g. the term

\[ \lambda^{(k-1)} \left[ y(1) - \phi^T(0) \hat{\theta}(k) \right]^2 \]

is small when \( k \) is large since \( \lim_{m \to \infty} \lambda^m = 0 \)
RLS with forgetting factor

A-priori version:

\[ e^o(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k) \]

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k + 1)\phi(k)e^o(k + 1) \]

\[ F(k + 1) = \frac{1}{\lambda} \left[ F(k) - \frac{F(k)\phi(k)\phi(k)^T F(k)}{\lambda + \phi(k)^T F(k)\phi(k)} \right] \]

\[ F^{-1}(k + 1) = \lambda F^{-1}(k) + \phi(k)\phi^T(k) \]
RLS with forgetting factor

A-posteriori version (used to prove that $e(k) \longrightarrow 0$):

$$e^o(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k)$$

$$e(k + 1) = \frac{\lambda e^o(k + 1)}{\lambda + \phi^T(k)F(k)\phi(k)}$$

$$\hat{\theta}(k + 1) = \hat{\theta}(k) + \frac{1}{\lambda}F(k)\phi(k)e(k + 1)$$

$$F(k + 1) = \frac{1}{\lambda} \left[ F(k) - \frac{F(k)\phi(k)\phi(k)^T F(k)}{\lambda + \phi(k)^T F(k)\phi(k)} \right]$$
The gain of the RLS with FF may blow up

\[
\text{tr } [F(k)] = \frac{1}{\lambda} \text{tr } [F(k-1)] - \frac{\|F(k-1)\phi(k-1)\|^2}{\lambda^2 + \lambda \phi^T(k-1)F(k-1)\phi(k-1)}
\]

if \( \phi(k) \) is not persistently exciting

(more on this later)
General PAA gain formula

\[ F^{-1}(k + 1) = \lambda_1(k) F^{-1}(k) + \lambda_2(k) \phi(k)\phi^T(k) \]

\[
0 < \lambda_1(k) \leq 1 \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad 0 \leq \lambda_2(k) < 2
\]

- **Constant adaptation gain:** \( \lambda_1(k) = 1, \lambda_2(k) = 0 \)
  
  *(We talked about this case in the previous lecture)*

- **RLS:** \( \lambda_1(k) = 1, \lambda_2(k) = 1 \)

- **RLS with forgetting factor:** \( \lambda_1(k) < 1, \lambda_2(k) = 1 \)
General PAA gain formula

\[ F^{-1}(k + 1) = \lambda_1(k) F^{-1}(k) + \lambda_2(k) \phi(k)\phi^T(k) \]

\[ 0 < \lambda_1(k) \leq 1 \quad 0 \leq \lambda_2(k) < 2 \]

\[ F(k+1) = \frac{1}{\lambda_1(k)} \left[ F(k) - \lambda_2(k) \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k) + \lambda_2(k)\phi^T(k)F(k)\phi(k)} \right] \]

\[ F(0) = F^T(0) > 0 \]
General PAA

A-priori version:

\[ e^o(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k) \]

\[ \hat{\theta}(k+1) = \hat{\theta}(k) + \frac{1}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)}F(k)\phi(k)e^o(k+1) \]

\[ F(k+1) = \frac{1}{\lambda_1(k)} \left[ F(k) - \lambda_2(k) \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k) + \lambda_2(k)\phi^T(k)F(k)\phi(k)} \right] \]

When \( \lambda_2(k) = 1 \), the parameter estimate equation simplifies to

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k + 1)\phi(k)e^o(k + 1) \]
General PAA

A-posteriori version (used to prove that \( e(k) \rightarrow 0 \)):

\[
e^o(k + 1) = y(k + 1) - \phi^T(k)\hat{\theta}(k)
\]

\[
e(k + 1) = \frac{\lambda_1(k)}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)} e^o(k + 1)
\]

\[
\hat{\theta}(k + 1) = \hat{\theta}(k) + \frac{1}{\lambda_1(k)} F(k)\phi(k)e(k + 1)
\]

\[
F(k+1) = \frac{1}{\lambda_1(k)} \left[ F(k) - \lambda_2(k) \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k) + \lambda_2(k)\phi^T(k)F(k)\phi(k)} \right]
\]
Additional Material
(you are not responsible for this)

• The Matrix Inversion Lemma

• Relationships for the General PAA

(these will be included in the next version...)
Matrix Inversion Lemma (simplified version)

• Since $\det(I + RL) = \det(I + LR)$, we know that

$$I + RL \text{ is invertible}$$

$$\iff$$

$$I + LR \text{ is invertible}$$

• The matrix inversion lemma (simplified version) states that

$$(I + RL)^{-1} = I - R(I + LR)^{-1}L$$
Matrix Inversion Lemma
(simplified version)

\[(I + RL)^{-1} = I - R(I + LR)^{-1}L\]

Proof:

Define \( \Phi = I - R(I + LR)^{-1}L \)

We want to show that \((I + RL)\Phi = I\)

\[(I + RL)\Phi = (I + RL) - (I + RL)R(I + LR)^{-1}L\]

\[\underbrace{R + RLR = R(I + LR)}\]

\[(I + RL)\Phi = I + RL - R(I + LR)(I + LR)^{-1}L\]

\[= I + RL - RL\]
Matrix Inversion Lemma

If $A$, $C$, and $(A+UCV)$ are invertible, then

$$(A+UCV)^{-1} = A^{-1} - A^{-1}U \left(C^{-1} + VA^{-1}U\right)^{-1} VA^{-1}$$

Proof:

$$(A + UCV)^{-1} = \left[(I + UCVA^{-1})A\right]^{-1}$$

$$= A^{-1} \left(I + UCVA^{-1}\right)^{-1}$$

$$= A^{-1} \left[I - UC \left(I + VA^{-1}UC\right)^{-1} VA^{-1}\right]$$

$$= A^{-1} \left[I - U \left[(I + VA^{-1}UC)C^{-1}\right]^{-1} VA^{-1}\right]$$

$$= A^{-1} - A^{-1}U \left(C^{-1} + VA^{-1}U\right)^{-1} VA^{-1}$$
Relationships for General PAA

Proof: We know that

\[ F^{-1}(k + 1) = \lambda_1(k) F^{-1}(k) + [\lambda_2(k) \phi(k)] \phi^T(k) \]

By the Matrix Inversion Lemma

\[ F(k + 1) = \frac{1}{\lambda_1(k)} F(k) \]

\[ - \left[ \frac{1}{\lambda_1(k)} F(k) \right] [\lambda_2(k) \phi(k)] \left[ \frac{1}{1 + \phi^T(k) \left[ \frac{1}{\lambda_1(k)} F(k) \right] [\lambda_2(k) \phi(k)]} \right] \phi^T(k) \left[ \frac{1}{\lambda_1(k)} F(k) \right] \]

This simplifies to the stated expression for \( F(k+1) \)
Relationships for General PAA

\[ F(k+1)\phi(k) = \frac{1}{\lambda_1(k) + \lambda_2(k)\phi^T(k)F(k)\phi(k)} F(k)\phi(k) \]

**Proof:**

\[ F^{-1}(k+1) = \lambda_1(k)F^{-1}(k) + \lambda_2(k)\phi(k)\phi^T(k) \]

\[ \Downarrow \]

\[ F(k+1) \left[ F^{-1}(k+1) \right] F(k)\phi(k) \]

\[ = F(k+1) \left[ \lambda_1(k)F^{-1}(k) + \lambda_2(k)\phi(k)\phi^T(k) \right] F(k)\phi(k) \]

\[ \Downarrow \]

\[ F(k)\phi(k) = \lambda_1(k)F(k+1)\phi(k) \]

\[ + \lambda_2(k)F(k+1)\phi(k)\phi^T(k)F(k)\phi(k) \]

\[ = F(k+1)\phi(k) \left[ \lambda_1(k) + \lambda_2(k)\phi^T(k)F(k)\phi(k) \right] \]
Relationships for General PAA

\[ e(k + 1) = \frac{\lambda_1(k)}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)}e^o(k + 1) \]

**Proof:**

\[ \tilde{\theta}(k + 1) = \tilde{\theta}(k) - \frac{1}{\lambda_1(k)}F(k)\phi(k)e(k + 1) \]

\[ \phi^T(k)\tilde{\theta}(k+1) = \phi^T(k) \left[ \tilde{\theta}(k) - \frac{1}{\lambda_1(k)}F(k)\phi(k)e(k + 1) \right] \]

\[ = \phi^T(k)\tilde{\theta}(k) - \frac{1}{\lambda_1(k)}\phi^T(k)F(k)\phi(k)e(k + 1) \]
**Relationships for General PAA**

\[ e(k + 1) = \frac{\lambda_1(k)}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)} e^o(k + 1) \]

**Proof (continued):**

From the previous slide,

\[ e(k+1) = e^o(k+1) - \frac{1}{\lambda_1(k)}\phi^T(k)F(k)\phi(k)e(k+1) \]

\[ \Downarrow \]

\[ [\lambda_1(k) + \phi^T(k)F(k)\phi(k)] e(k+1) = \lambda_1(k)e^o(k+1) \]