State-space Approach

• In tracking a moving object by remote measurements, we are interested in monitoring how position and velocity of the object change in time.

• The state-space approach to tracking, navigation, and many other application problems is based on describing a time-varying process by a vector of quantities.

• These quantities are collectively called the state of the process.

• The evolution of the process over time is represented as a trajectory in the space of states, i.e., a successive transition from one state to another.
State-space Modelling

• **State**: a vector of measurements for an object describing its behaviour in time
  
  – *Example*: \([p, v, a]\) - the position, velocity, and acceleration of a moving 1D "object" in time:
  
  \[v(t + \Delta t) = v(t) + a(t)\Delta t;\quad p(t + \Delta t) = p(t) + \frac{v(t+\Delta t)+v(t)}{2}\Delta t = p(t) + v(t)\Delta t + \frac{a(t)}{2}\Delta t\]

• **State space**: the space of all possible states

• **Trajectory** of an object in the state space: the evolution of the object’s state in time

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>(t)</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>(a(t))</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(v(t))</td>
<td>0</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>(p(t))</td>
<td>0</td>
<td>2.5</td>
<td>10</td>
<td>20</td>
<td>30</td>
<td>40</td>
</tr>
</tbody>
</table>
1D point trajectory in the 3D state space

- for $k = 0$: $a_{k+1} = a_k; \ v_{k+1} = v_k + a_k; \ p_{k+1} = p_k + v_k + \frac{a_k}{2}$

- for $k = 1, 2, \ldots$: $a_{k+1} = 0; \ v_{k+1} = v_k + a_k; \ p_{k+1} = p_k + v_k + \frac{a_k}{2}$
State-space Trajectory: Vector Description

**State** of the process: an $n \times 1$ vector $x_k$ of quantities describing the process at time $k$, e.g.

$$x_k = \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ x_{3,k} \end{bmatrix} \equiv \begin{bmatrix} p_k \\ v_k \\ a_k \end{bmatrix}; \quad k = 0, 1, 2, \ldots$$

**Observation**, or output: an $m \times 1$ vector $y_k$; $m \leq n$, being a vector or scalar function of the state vector at time $k$: $y_k = C_k(x_k)$

**Process evolution**: a vector function of the state vector at time $k$: $x_{k+1} = A_k(x_k)$
Estimating States: General Case

- **Problem**: Estimate states $x_k$ from observations $y_k$; $k = 0, 1, 2, \ldots$

- **Basic Assumptions**:
  
  - Vector functions $A_k(x_k)$ describing the evolution of states are known for each $k$ but with uncertainty $u_k$:
    \[
    x_{k+1} = A_k(x_k) + u_k
    \]
  
  - How the observation depends on the state vector is known also with measurement noise $v$:
    \[
    y_k = C_k(x_k) + v_k
    \]
  
  - Only statistical properties of the random vectors $u_k$ and $v_k$ are known
Estimating States: Linear Case

- Linear functions $A_k(\ldots)$ and $C_k(\ldots)$:
  - The $n \times n$ state evolution matrices $A_k$
  - The $m \times n$ output matrices $C_k$

- Matrix-vector evolution of the system:
  \[
  x_{k+1} = A_k x_k + u_k \\
  y_k = C_k x_k + v_k; \quad k = 0, 1, 2, \ldots
  \]

- The matrices $A_k$ and $C_k$ can be considered as linear approximations of the non-linear vector functions $A_k(\ldots)$ and $C_k(\ldots)$
Linear Case: an Example

State matrices: \( A_0 = \begin{bmatrix} 1 & 1 & 1/2 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \); \( A_k = \begin{bmatrix} 1 & 1 & 1/2 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \); \( k = 1, 2, \ldots \), and the output matrix \( C_k = [1 \ 0 \ 0] \)

\[\begin{array}{|c|c|c|}
\hline
k & 0 & 1 \\
\hline
x_{1,k} / u_{1,k} & 0.0 / 0.1 & 2.6 / -0.1 \\
x_{2,k} / u_{2,k} & 0.0 / -0.1 & 4.9 / 0.1 \\
x_{3,k} / u_{3,k} & 5.0 / 0.2 & 5.2 / -0.2 \\
y_k / v_k & 0.3 / 0.3 & 2.3 / -0.3 \\
\hline
\end{array}\]

\[\begin{array}{|c|c|c|}
\hline
2 & 3 & 4 \\
\hline
10.0 / 0.1 & 20.1 / -0.1 & 29.8 / 0.1 \\
10.2 / -0.1 & 9.9 / 0.1 & 9.8 / 0.0 \\
-0.2 / -0.2 & -0.2 / 0.0 & 0.0 / -0.2 \\
9.7 / -0.3 & 20.1 / 0.0 & 29.7 / -0.1 \\
\hline
\end{array}\]

**Goal:** Given the matrices \( A_k, C_k \), statistics of \( u_k, v_k \), and observations \( y_k \) for \( k = 0, 1, \ldots \), estimate the hidden state vectors \( x_k, k = 0, 1, \ldots \).
Evolution of a Periodic Signal – 1

- Scalar noisy observations $y_k$ of a periodic signal represented with a finite Fourier series plus a noise term:

$$y_k = c_1 e^{j2\pi f_1 k} + c_2 e^{j2\pi f_2 k} + \ldots + c_n e^{j2\pi f_n k}$$

where the coefficients $c_i$ are complex numbers

- For this periodic function, each frequency is the state component:

$$x_k = \begin{bmatrix} e^{j2\pi f_1 k} \\ e^{j2\pi f_2 k} \\ \vdots \\ e^{j2\pi f_n k} \end{bmatrix}$$

$$x_{i,k+1} = e^{j2\pi f_i (k+1)}$$

$$= e^{j2\pi f_i} e^{j2\pi f_i k} = e^{j2\pi f_i x_{i,k}}$$

Evolution of a state component
Evolution of a Periodic Signal – 2

• The state evolution: \( x_{k+1} = A_k x_k \) where \( A_k \) is the diagonal \( n \times n \) matrix:

\[
A_k \equiv A = \begin{bmatrix}
  e^{j2\pi f_1} & 0 & \cdots & 0 \\
  0 & e^{j2\pi f_2} & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & e^{j2\pi f_n}
\end{bmatrix}
\]

• The observation \( y_k = C_k x_k + v_k \) where \( C_k \) is the \( 1 \times n \) vector-row:

\[
C_k \equiv C = [c_1 \ c_2 \ \cdots \ c_n]
\]

• In this example, there is no uncertainty in the state evolution: \( u_k = 0 \)
Estimation of States from Observations

Let $\hat{x}_k$ denote the state estimated from all the known at time $k$ observations $y_t; t = 0, 1, \ldots, k$:

$$\hat{x}_k \equiv \hat{x}_k(y_0, \ldots, y_k)$$

At time $k$, the estimator has to minimise the average squared error

$$e_k = \sum_{i=1}^{n} |x_{i,k} - \tilde{x}_{i,k}|^2 \equiv \sum_{i=1}^{n} |x_{i,k} - \tilde{x}_k(y_0, \ldots, y_k)|^2$$

under the simplifying assumptions:

- the state uncertainty $u_k$ is totally uncorrelated with the measurement noise $v_k$ and

- each pair of vectors $(u_k, u_l)$ or $(v_k, v_l)$ are totally uncorrelated for $k \neq l$
Basic Notation – 1

- An $n$-dimensional (or $n \times 1$) column vector $x$ of states has generally complex-valued components $x_1, \ldots, x_n$.

- The conjugate, or Hermite transpose of $x$, denoted $x^H$, is the $1 \times n$ row vector of complex-conjugate components $[x_1^* \ldots x_n^*]$. If $x = a + jb$, then $x^* = a - jb$ where $a$ and $b$ are the real and imaginary components of the complex $x$.

- The inner product between two complex vectors $x$ and $y$ of the same dimension is defined as $x^H y = \sum_{i=1}^{n} x_i^* y_i$.
  - Two vectors are perpendicular if $x^H y = 0$.
  - The vector length is computed as $\| x \| = \sqrt{x^H x}$.
Basic Notation – 2

- **Conjugate transposition** $H$ of an $m \times n$ matrix $A$ with complex elements $a_{\alpha,\beta}$ is the $n \times m$ matrix $A^H$ such that $a^H(\beta, \alpha) = a^*(\alpha, \beta)$

  $1 \leq \alpha \leq m$ – rows and $1 \leq \beta \leq n$ – columns in $A$

- **Law of composition** for $H$: $(AB)^H = B^H A^H$

  for matrices $A$ and $B$

- **Outer product** $xy^H$ of an $n \times 1$ vector $x$ and an $m \times 1$ vector $y$ is the $n \times m$ matrix of pairwise vector component products:

  $$
  \begin{bmatrix}
  x_1 \\
  x_2 \\
  \vdots \\
  x_n
  \end{bmatrix}
  \begin{bmatrix}
  y_1^* \\
  y_2^* \\
  \vdots \\
  y_m^*
  \end{bmatrix}
  =
  \begin{bmatrix}
  x_1 y_1^* & x_1 y_2^* & \cdots & x_1 y_m^* \\
  x_2 y_1^* & x_2 y_2^* & \cdots & x_2 y_m^* \\
  \vdots & \vdots & \ddots & \vdots \\
  x_n y_1^* & x_n y_2^* & \cdots & x_n y_m^*
  \end{bmatrix}
  $$
• **Average** or expected value of a continuous random variable: \( \mathbb{E}\{x\} = \int_{-\infty}^{\infty} xp(x) \, dx \)

  - \( p(x) \): a probability density function (p.d.f.) of \( x \)
  - \( \mathbb{E}\{\ldots\} \) denotes the mathematical expectation
  - Expected vector \( \mathbb{E}\{x\} \) of random variables: the vector of expected elements \( \mathbb{E}\{x_i\}; \; i = 1, \ldots, n \)
  - Expected vector sum: \( \mathbb{E}\{x + y\} = \mathbb{E}\{x\} + \mathbb{E}\{y\} \)
  - Expected matrix \( A \): the matrix of expected elements \( \mathbb{E}\{A(\alpha, \beta)\} \)

• **Correlation** between two random variables \( x \) and \( y \): \( \mathbb{E}\{xy^*\} = \int_{-\infty}^{\infty} xy^*p(x, y) \, dx \)

  - \( p(x, y) \) is a joint p.d.f. of \( x \) and \( y \)
Probability Concepts – 2

- **Correlation matrix** of two vectors $x$ and $y$ of random variables is the expected outer product matrix $xy^H$

- Entries of the correlation matrix are expected pairwise products of the scalar vector entries $E\{x_\alpha y^*_\beta\}$

- The correlation matrix of the error $x_k - \hat{x}_k$ is the matrix $E\{(x_k - \hat{x}_k)(x_k - \hat{x}_k)^H\}$

- Pair of vectors $x$ and $y$ are **uncorrelated** if $E\{xy^H\} = 0$ where $0$ – the matrix of appropriate dimensions with zero entries
State / Observation Statistics
Known by Assumption:

the $n \times n$ correlation matrix $U_k$ for uncertainty $u_k$ and the $m \times m$ correlation matrix $V_k$ for measurement noise $v_k$ for all $k, l = 0, \ldots, K$:

$$
\mathbb{E}\{u_k u_l^T\} = \begin{cases} U_k & \text{if } k = l \\
0 & \text{otherwise} \end{cases} \\
\mathbb{E}\{v_k v_l^T\} = \begin{cases} V_k & \text{if } k = l \\
0 & \text{otherwise} \end{cases}; \quad \mathbb{E}\{u_k v_l^T\} = 0
$$

Components of the latter expected matrices are expected pairwise products of vector components such as $\mathbb{E}\{u_k,\alpha u_l,\beta\}; \alpha, \beta = 1, \ldots, n$, $\mathbb{E}\{v_k,\alpha v_l,\beta\}; \alpha, \beta = 1, \ldots, m$, or $\mathbb{E}\{u_k,\alpha v_l,\beta\}; \alpha = 1, \ldots, n; \beta = 1, \ldots, m$

Both the uncertainty and measurement noise are centred: $\mathbb{E}\{u_k\} = \mathbb{E}\{v_k\} = 0; \ k = 0, 1, \ldots, K$
Rudolf Kalman’s Approach

The search for a linear estimator:

\[ \hat{x}_k = \sum_{t=0}^{k} G_t y_t \]

where \( G_k; k = 0, 1, \ldots, K \), are \( n \times m \) gain matrices to be determined.

The desired gain matrices have to minimise the mean error \( \mathbb{E}\{\| x_k - \hat{x}_k \|^2 \} \).

Initial estimate \( \hat{x}_0 \) and correlation matrix \( P_0 \) of estimation error are assumed to be known.

The Kalman’s observation was that this linear estimate should **evolve recursively** just as the system’s states are evolving themselves (!!)

This brilliant observation became a cornerstone of the most popular at present approach to linear filtering called **Kalman filtering**.
Constructing a Kalman Filter – 1

Suppose an optimal linear estimate $\hat{x}_{k-1}$ based on observations $y_0, y_1, \ldots, y_{k-1}$ is already constructed.

Then $\hat{x}_i^k \overset{\text{def}}{=} A_{k-1} \hat{x}_{k-1}$ is the best guess of $\hat{x}_k$ before making the observation $y_k$ at time $k$.

It is the natural evolution of the estimated state vector $\hat{x}_{k-1}$ by the linear system dynamics in Slide 6.

The superscript “i” indicates this is an intermediate estimate before constructing $\hat{x}_k$.

$y_i^k = C_k \hat{x}_i^k$ is the best prediction of $y_k$ before the actual measurement.

**Kalman’s proposal**: the optimal solution for $\hat{x}_k$ should be a linear combination of $\hat{x}_i^k$ and the difference between $y_k$ and $y_i^k$:

$$\hat{x}_k = \hat{x}_i^k + G_k (y_k - C_k \hat{x}_i^k)$$
Constructing a Kalman Filter – 2

If \( y_k = y^i_k \), then \( \hat{x}_k = \hat{x}^i_k = A_{k-1} \hat{x}_{k-1} \), i.e. the estimate evolves purely by what is known about the process.

**Optimal gain matrix** \( G_k \) has to minimise the mean error \( \mathbb{E}\{\|x_k - \hat{x}_k\|^2\} \) in Slide 16:

\[
\mathbb{E}\left\{\| (x_k - \hat{x}^i_k) - G_k (y_k - C_k \hat{x}^i_k) \|^2 \right\}
\]

Solution: by taking and setting to zero the derivative w.r.t. to the matrix entries.

**Theorem 1:** Let \( a \) and \( b \) be random vectors. Then the matrix \( G \) minimising \( \mathbb{E}\{\| a - Gb \|^2 \} \) is as follows:

\[
G = \mathbb{E}\left\{ab^H\right\} \left( \mathbb{E}\left\{bb^H\right\} \right)^{-1}
\]

providing the correlation matrix \( \mathbb{E}\{bb^H\} \) is invertible.
Proof of Theorem 1 – (a)

Derivative of a scalar function $f$ w.r.t. an $n \times m$ matrix $Q$ is defined as

$$\frac{\partial f}{\partial Q} = \begin{bmatrix} \frac{\partial f}{\partial Q_{1,1}} & \frac{\partial f}{\partial Q_{2,1}} & \cdots & \frac{\partial f}{\partial Q_{n,1}} \\ \frac{\partial f}{\partial Q_{1,2}} & \frac{\partial f}{\partial Q_{2,2}} & \cdots & \frac{\partial f}{\partial Q_{n,2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial Q_{1,m}} & \frac{\partial f}{\partial Q_{2,m}} & \cdots & \frac{\partial f}{\partial Q_{n,m}} \end{bmatrix}$$

For a function $f = t^H Q s$ where $t$ and $s$ are arbitrary $n \times 1$ and $m \times 1$ vectors, respectively, the derivative is

$$\frac{\partial}{\partial Q} (t^H Q s) = st^H$$

The right hand side matrix is of the dimension $m \times n$

Each its $(\beta, \alpha)$-entry $t_\alpha^* s_\beta$ is precisely what is obtained by differentiating the scalar function $f$ w.r.t. the $(\alpha, \beta)$-entry $Q_{\alpha,\beta}$ of $Q$
Proof of Theorem 1 – (b)

Expanding $\mathbb{E}\{\| a - Gb \|^2 \}$ gives

\[
\begin{align*}
\mathbb{E} \left\{ (a - Gb)^H (a - Gb) \right\} & = \mathbb{E} \left\{ (a^H - b^H G^H) (a - Gb) \right\} \\
& = \mathbb{E} \{ a^H a - b^H G^H a - a^H Gb + b^H G^H Gb \} \\
& = \mathbb{E} \{ a^H a \} - \mathbb{E} \{ b^H G^H a \} - \mathbb{E} \{ a^H Gb \} + \mathbb{E} \{ b^H G^H Gb \}
\end{align*}
\]

Differentiating this with respect to the matrix $G$ may seen difficult because both $G$ and $G^H$ are appearing.

It can be proven that the elements of $G$ can be treated as independent from the elements of $G^H$ although they are not of course

Setting the derivative of the above expression w.r.t. $G^H$ equal to zero produces the equation

\[-\mathbb{E} \{ ab^H \} + G \mathbb{E} \{ bb^H \} = 0\]

It gives the solution $G = \mathbb{E} \{ ab^H \} \left( \mathbb{E} \{ bb^H \} \right)^{-1}$
Constructing a Kalman Filter – 3

To optimise the gain matrix \( G_k \), \( a = x_k - \hat{x}_k^i \) and \( b = y_k - C_k \hat{x}_k^i \), so that

\[
E \left\{ ab^H \right\} = E \left\{ (x_k - \hat{x}_k^i) (y_k - C_k \hat{x}_k^i)^H \right\} \\
= E \left\{ (x_k - \hat{x}_k^i) (C_k x_k + v_k - C_k \hat{x}_k^i)^H \right\} \\
= E \left\{ (x_k - \hat{x}_k^i) (x_k - \hat{x}_k^i)^H C_k^H \right\} \\
+ E \left\{ (x_k - \hat{x}_k^i) v_k^H \right\}
\]

The last expectation on the right is zero as the intermediate estimate \( \hat{x}_k^i \) depends only on \( y_0, y_1, \ldots, y_{k-1} \) including only the noise terms \( v_i \) and uncertainties \( u_i \) for \( i < k \) that are uncorrelated with the “new” noise \( v_k \).

Thus, \( E \left\{ ab^H \right\} = E \left\{ (x_k - \hat{x}_k^i) (x_k - \hat{x}_k^i)^H C_k^H \right\} = E \left\{ (x_k - \hat{x}_k^i) (x_k - \hat{x}_k^i)^H \right\} C_k^H \equiv P_k^i C_k^H \)

where \( P_k^i = E \left\{ (x_k - \hat{x}_k^i) (x_k - \hat{x}_k^i)^H \right\} \) denotes the correlation matrix for the “intermediate” error \( x_k - \hat{x}_k^i \).
Constructing a Kalman Filter – 4

Similar considerations result in a following simple form for

\[
\mathbb{E} \{ \mathbf{b}\mathbf{b}^H \} = \mathbb{E} \left\{ \left( \mathbf{y}_k - \mathbf{C}_k \hat{\mathbf{x}}_k^i \right) \left( \mathbf{y}_k - \mathbf{C}_k \hat{\mathbf{x}}_k^i \right)^H \right\} \\
= \mathbb{E} \left\{ \left( \mathbf{C}_k \mathbf{x}_k + \mathbf{v}_k - \mathbf{C}_k \hat{\mathbf{x}}_k^i \right) \left( \mathbf{C}_k \mathbf{x}_k + \mathbf{v}_k - \mathbf{C}_k \hat{\mathbf{x}}_k^i \right)^H \right\} \\
= \mathbb{E} \left\{ \left( \mathbf{C}_k \left( \mathbf{x}_k - \hat{\mathbf{x}}_k^i \right) + \mathbf{v}_k \right) \left( \left( \mathbf{x}_k - \hat{\mathbf{x}}_k^i \right)^H \mathbf{C}_k^H + \mathbf{v}_k^H \right) \right\} \\
= \mathbf{C}_k \mathbf{P}_k^i \mathbf{C}_k^H + \mathbf{V}_k
\]

where \( \mathbf{V}_k = \mathbb{E} \left\{ \mathbf{v}_k \mathbf{v}_k^H \right\} \) is the measurement noise correlation matrix.

By Theorem 1, the optimal gain matrix is

\[
\mathbf{G}_k = \mathbf{P}_k^i \mathbf{C}_k^H \left( \mathbf{C}_k \mathbf{P}_k^i \mathbf{C}_k^H + \mathbf{V}_k \right)^{-1}
\]

assuming that the inverse on the right hand side exists.

The correlation matrix \( \mathbf{P}_k^i \) is also computed recursively starting from the matrix \( \mathbf{P}_0 \) known by assumption.
Constructing a Kalman Filter – 5

Since \( x_k = A_{k-1} x_{k-1} + u_{k-1} \) and \( \hat{x}^i_k = A_{k-1} \hat{x}_{k-1} \),

\[
P^i_k = \mathbb{E} \left\{ (x_k - \hat{x}^i_k) (x_k - \hat{x}^i_k)^H \right\} = \mathbb{E} \left\{ (A_{k-1} x_{k-1} + u_{k-1} - \hat{x}^i_k) (A_{k-1} x_{k-1} + u_{k-1} - \hat{x}^i_k)^H \right\} = \mathbb{E} \left\{ (A_{k-1} (x_{k-1} - \hat{x}_{k-1}) + u_{k-1}) (A_{k-1} (x_{k-1} - \hat{x}_{k-1}) + u_{k-1})^H \right\}
\]

After some rearrangement and elimination of zero-valued expectations:

\[
P^i_k = A_{k-1} P_{k-1} A_{k-1}^H + U_{k-1}
\]

where \( P_{k-1} = \mathbb{E} \left\{ (x_{k-1} - \hat{x}_{k-1}) (x_{k-1} - \hat{x}_{k-1})^H \right\} \) denotes the correlation matrix of estimation errors and \( U_{k-1} \) is the correlation matrix of uncertainties at time \( k - 1 \). Substituting the formula for \( \hat{x}_k \) to the definition of \( P_k \) and with some amount of algebra, one obtains that

\[
P_k = P^i_k - G_k C_k P^i_k
\]
How the Kalman Filter Works

Known values: \( y_i, V_i, \) and \( U_i, A_i, \) and \( C_i \) for \( 0 \leq i \leq k \) at each time \( k \)

- **Initialisation** \( k = 0 \): Choose or guess suitable \( \hat{x}_0 \) and \( P_0 \)

- **Iteration** \( k = 1, 2, \ldots \): Given \( \hat{x}_{k-1} \) and \( P_{k-1} \), compute:

  1. \( P_i^k = A_{k-1}P_{k-1}A_{k-1}^H + U_{k-1} \)

  2. \( G_k = P_i^kC_k^H \left( C_kP_i^kC_k^H + V_k \right)^{-1} \)

  3. \( \hat{x}_i^k = A_{k-1}\hat{x}_{k-1} \)

  4. \( \hat{x}_k = \hat{x}_i^k + G_k \left( y_k - C_k\hat{x}_i^k \right) \)

  5. \( P_k = P_i^k - G_kC_kP_i^k \)
Example: 1D Process

Fixed state $x_{k+1} = x_k$

Noisy measurements $y_k = x_k + v_k$

$\mathbb{E}\{v_k\} = 0; \mathbb{E}\{v_k^2\} = \sigma^2$ for all $k$

$\mathbb{E}\{x_0\} = \hat{x}_0 = 0; \mathbb{E}\{x_0^2\} = P_0 > 0$

$\Rightarrow A_k = C_k = 1; U_k = 0$, and $V_k = \sigma^2$ for all $k$

In this case, $\hat{x}_k^i = \hat{x}_{k-1}^i$, $P_k^i = P_{k-1}$ for all $k$ so that the intermediate steps are unnecessary (the state is not changing):

\[
G_k = \frac{P_{k-1}}{P_{k-1} + \sigma^2}
\]

\[
P_k = P_{k-1} - \frac{P_{k-1}^2}{P_{k-1} + \sigma^2} = \frac{P_{k-1}\sigma^2}{P_{k-1} + \sigma^2}
\]

\[
\hat{x}_k = \hat{x}_{k-1} + \frac{P_{k-1}}{P_{k-1} + \sigma^2} (y_k - \hat{x}_{k-1})
\]

Case 1: $\sigma = 0$ (no measurement noise) $\rightarrow \hat{x}_k = y_k$

Case 2: $\sigma > 0$; $P_0 = 0$ (so all $x_k = 0$) $\rightarrow G_k = 0; P_k = 0$, and $\hat{x}_k = 0$ for all $k$

Case 3: $\sigma > 0$; $P_0 > 0 \rightarrow P_k < P_{k-1}$ (decreasing error variance), and since $P_0 > 0$, in the limit $\lim_{k \to \infty} P_k = 0$