11 Efectos de precisión finita

- La teoría de filtrado adaptivo discutida enteriormente supone la utilización de un **modelo analógico** (o sea, de precisión infinita) para las muestras de los datos de entrada y los cálculos internos de los algoritmos.

- En particular, en una implementación digital de un algoritmo de filtrado adaptivo como encontrada en la práctica, los datos de entrada y los cálculos internos del algoritmo se realizan con **precisión finita**.

- En consecuencia, el proceso de cuantización tiene el efecto de desviar el desempeño de la implementación digital del algoritmo de la versión teórica. Esta desviación está influenciada por:
  - El tipo de detalles de diseño del algoritmo de filtrado adaptivo adoptado.
  - El grado de condicionamiento (dispersión de autovalores) de la matriz correlación que caracteriza los datos de entrada.
  - La forma de realizar los cálculos numéricos (punto fijo o punto flotante).

- Es importante entender las propiedades numéricas de los algoritmos de filtrado adaptivo dado que esto ayudará en la busqueda de cumplimiento de especificaciones de diseño.

- Además, el costo de la implementación digital de un algoritmo está influenciado por el **número de bits** (o sea, la precisión) disponible para efectuar los cálculos numéricos asociados al algoritmo.

- Genericamente hablando, el costo de la implementación aumenta con el número de bits empleados.

11.1 Errores de cuantización

En una implementación digital de un filtro adaptivo existen esencialente dos fuentes de errores de cuantización a ser consideradas:

1. **Conversión analógica - digital**.
   - Dado que los datos están en forma analógica, es posible utilizar un conversor A/D para obtener su representación numérica.
   - Suponemos que el proceso de cuantización tiene un **tamaño de paso uniforme** δ y un conjunto de **niveles de cuantización**, posicionados en $0, \pm\delta, \pm 2\delta, \cdots$.
   - Consideramos una muestra particular en la entrada del cuantizador, con una amplitud en el rango $i\delta - (\delta/2)$ a $i\delta + (\delta/2)$, donde i es un entero (positivo o negativo, incluyendo el cero) y $i\delta$ define la **salida del cuantizador**.
   - El proceso de cuantización introduce una región de incertidumbre de ancho δ, centrada en $i\delta$.
   - Si η es el error de cuantización, la salida del cuantizador es $i\delta + \eta$, donde η está acotado entre $-(\delta/2) \leq \eta \leq (\delta/2)$.
   - Cuando la cuantización es suficientemente fina (o sea, el número de niveles de cuantización es mayor a 64) y si el espectro de la señal es suficientemente rico, la distorsión producida por el proceso de cuantización puede modelarse como una fuente independiente de ruido blanco de media cero y varianza determinada por el tamaño del paso del cuantizador δ.
   - Es usual suponer que el error de cuantización η está **uniformemente distribuido** en un rango $-\delta/2 \leq \eta \leq \delta/2$. La varianza del error de cuantización está entonces dada por

Figura 17.1 pag xxx

$$\sigma^2 = \int_{-\delta/2}^{\delta/2} \frac{1}{\delta}\eta^2 d\eta = \frac{\delta^2}{12}$$

- Suponemos que la entrada al cuantizador está apropiadamente escalada tal que esté dentro del intervalo $(-1, +1]$, con cada nivel de cuantización representado por B bits más signo, el tamaño del paso del cuantizador es

$$\delta = 2^{-B}$$

de forma que el error de cuantización resultante de la representación digital de los datos analógicos tiene una varianza

$$\sigma^2 = \frac{2^{-2B}}{12}$$

2. **Aritmética de longitud finita de palabra.**

- Los cálculos aritméticos internos en una máquina digital se almacenan comunmente con una longitud de palabra finita.
- Suponiendo que no existe overflow durante el transcurso de un cálculo, las sumas no introducen error (si se utiliza aritmética de punto fijo), mientras que cada multiplicación introduce un error despues que el producto ha sido truncado.
- La caracterización estadística de los errores aritméticos de longitud finita de palabra pueden ser totalmente diferentes a los asociados al proceso de conversión AD. Los errores de aritmética de longitud finita de palabra pueden tener **media no nula**, lo cual resulta del redondeo o truncamiento de la salida de un multiplicador para alcanzar la longitud de palabra preestablecida.
- La presencia de aritmética de longitud finita de palabra introduce un problema serio en la implementación digital de un filtro adaptivo, particularmente cuando los coeficientes del filtro se adaptan en forma contínua.
- La versión digital del filtro exibe una **respuesta** específica o **propagación** de tales errores, haciendo que el desempeño se desvie de la forma ideal (de precisión infinita).
- Además, es posible que la desviación sea de naturaleza catastrófica en el sentido que los errores resultantes del uso de aritmética de precisión finita se acumulen sin limite.
- Si tal situación persiste el filtro entrará en una condición de overflow y el algoritmo se comportará en forma **numericamente inestable**. Un filtro adaptivo se dice **numericamente estable** si la utilización de aritmética de precisión finita resulta en desviaciones **acotadas** de la forma de precisión infinita.

- Es importante tener encuenta que la estabilidad numérica es una característica inherente al filtro adaptivo. En otras palabras, si un filtro adaptivo es numericamente estable, entonces incrementar el número de bits utilizados en la implementación digital no cambiará la condición de estabilidad de la implementación.
- Otro aspecto que requiere atención en la implementación digital de un filtro adaptivo es la **precisión numérica**. A diferencia de la estabilidad numérica, la precisión numérica de un filtro adaptivo está determinada por el número de bits utilizados para implementar los cálculos internos. Cuanto mayor es el número de bits usados, menor será la desviación del desempeño ideal y más precisa será entonces la implementación digital del filtro.
- En términos prácticos es pertinente hablar de precisión numérica de un filtro adaptivo si es numericamente estable.

11.2 Algoritmo LMS

Para simplificar la discusión de los efectos de longitud finita sobre el desempeño del algoritmo LMS se supondrá que la entrada y en consecuencia los coeficientes del filtro son **reales**.

Un diagrama en bloques del algoritmos LMS de precisión finita se muestra en la figura 11.2. El operador Q representa un **cuantizador** que introduce error de redondeo en si mismo. Es posible describir las relaciones entrada - salida de los cuantizadores de esa figura como sigue:

Figura 17.2 pag xxx

1. Para el cuantizador de la entrada se tiene que

$$\boldsymbol{u}_q(n) = Q[\boldsymbol{u}(n)] = \boldsymbol{u}(n) + \boldsymbol{\eta}_u(n)$$

   donde $\boldsymbol{\eta}_u(n)$ es el vector de **error de cuantización de la entrada**.

2. Para el cuantizador de la respuesta deseada se tiene que

$$\boldsymbol{d}_q(n) = Q[d(n)] = d(n) + \eta_d(n)$$

   donde $\eta_d(n)$ es el vector de **error de cuantización de la respuesta deseada**.

3. Para el cuantizador de los coeficientes se tiene que

$$\hat{\boldsymbol{w}}_q(n) \;=\; Q\left[\hat{\boldsymbol{w}}(n)\right] = \hat{\boldsymbol{w}}(n) + \Delta\hat{\boldsymbol{w}}(n)$$

   donde $\Delta\hat{\boldsymbol{w}}(n)$ es el vector de **error de cuantización de los coeficientes**.

4. Para el cuantizador de la salida del filtro FIR se tiene que

$$\boldsymbol{y}_q(n) \;=\; Q\left[\boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n)\right] = \boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n) + \eta_y(n)$$

   donde $\eta_y(n)$ es el vector de **error de cuantización de la salida del filtro**.

El algoritmo LMS de precisión finita se describe por las siguientes relaciones

$$e_q(n) \;=\; d_q(n) - y_q(n) \tag{152}$$

$$\hat{\boldsymbol{w}}_q(n+1) \;=\; \hat{\boldsymbol{w}}_q(n) + Q\left[\mu e_q(n)\boldsymbol{u}_q(n)\right] \tag{153}$$

- El producto $\mu e_q(n)\boldsymbol{u}_q(n)$ representa una versión escalada de la estimación del vector gradiente, está cuantizado **antes** de la suma al vector de coeficientes.

- Debido a restricciones de hardware, esta forma de implementación digital es preferible antes que el método de operar con el acumulador de coeficientes en precisión doble y luego la cuantización del mismo en precisión simple.

En el análisis estadístico del algoritmo LMS de precisión finita es posible realizar las siguientes suposiciones

1. La entrada está escalada apropiadamente para **prevenir overflow** de los elementos de $\hat{\boldsymbol{w}}_q(n)$ y la salida cuantizada $y_q(n)$, durante la operación del filtro.

2. Cada muestra se representa por B_D bits más signo, y cada coeficiente se representa por B_W bits más signo. Así, el error de cuantización asociado con un número de B_D bits más signo, tiene como varianza

$$\sigma_D^2 = \frac{2^{-2B_D}}{12}$$

y en forma similar, el error de cuantización asociado a un número de B_W bits más signo tiene como varianza

$$\sigma_W^2 = \frac{2^{-2B_W}}{12}$$

3. Los elementos de $\boldsymbol{\eta}_u(n)$ y $\boldsymbol{\eta}_d(n)$ son secuencias de **ruido blanco**, independientes de las respectivas señales y entre sí. Además tienen media cero y varianza σ_D^2.

4. $\eta_y(n)$ es una secuencia de ruido blanco, independiente de las señales de entrada y otros ruidos de cuantización. Tiene media cero y varianza $c\sigma_D^2$, donde c es una constante que depende en que se calcule $\boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n)$:

   - Si los productos escalares individuales de $\boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n)$ se calculan sin cuantización, luego se suman y el resultado final se cuantiza con B_D bits más signo, la constante $c = 1$ y $\eta_y(n) = \sigma_D^2$.

- Si los productos escalares individuales de $\boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n)$ se cuantizan y luego se suman, la constante $c = M$ y $\eta_y(n) = M\sigma_D^2$, donde M es el orden del filtro adaptivo.

5. Se asumirá la teoría de la independencia del capítulo de LMS con precisión infinita.

Error medio cuadrático total de salida

El **error total de salida** es igual a

$$\begin{aligned} e_{total}(n) &= d(n) - y_q(n) \\ &= d(n) - \boldsymbol{u}_q^T(n)\hat{\boldsymbol{w}}_q(n) - \eta_y(n) \end{aligned}$$

sustituyendo la descripción de los vectores cuantizados e ignorando términos de error de cuantización mayores de primer orden se tiene que

$$e_{total}(n) = \underbrace{[d(n) - \boldsymbol{u}^T(n)\hat{\boldsymbol{w}}(n)]}_{\text{LMS precisión infinita}} - \underbrace{[\Delta\hat{\boldsymbol{w}}^T(n)\boldsymbol{u}(n) + \boldsymbol{\eta}_u^T(n)\hat{\boldsymbol{w}}(n) - \eta_y(n)]}_{\text{errores de cuantización}}$$

Con base en las suposiciones anteriores y suponiendo mu sufientemente pequeño es posible mostrar que el **error medio cuadrático total** producido por el algoritmo de precisión finita tiene la siguiente forma de estado estacionario

$$E[e_{total}^2(n)] = \underbrace{J_{min}(1+\mathcal{M})}_{\text{MSE precisión infinita}} + \underbrace{\xi_1(\sigma_w^2, \mu)}_{\text{debido a } \Delta\hat{\boldsymbol{w}}} + \underbrace{\xi_2(\sigma_D^2)}_{\text{debido a } \boldsymbol{\eta}_u(n) \text{ y } \eta_y(n)} \quad (154)$$

donde J_{min} es el **error medio cuadrático mínimo** del filtro de Wiener óptimo y $\mathcal{M}$ es el **desajuste** del algoritmo LMS de precisión infinita. $\xi_1(\sigma_w^2, \mu)$ es **inversamente proporcional al factor de convergencia** μ. $\xi_2(\sigma_D^2)$ es basicamente independiente de μ.

Desviaciones durante el período de convergencia

Durante el período de convergencia la situación es compleja.

Es posible mostrar que las propiedades del transitorio de adaptación pueden analizarse a través de una ecuación general del desajuste de los coeficientes ó **perturbación** con respecto a la solución obtenida con el algoritmo de precisión infinita.

El desajuste de los coeficientes se define por

$$\mathcal{W}(n) = E[\Delta\hat{\boldsymbol{w}}^T(n)\Delta\hat{\boldsymbol{w}}(n)]$$

donde el vector de error de los coeficientes está definido por $\Delta\hat{\boldsymbol{w}}(n) = \hat{\boldsymbol{w}}_q(n) - \hat{\boldsymbol{w}}(n)$, donde $\hat{\boldsymbol{w}}_q(n)$ y $\hat{\boldsymbol{w}}(n)$ hacen referencia a los coeficientes de precisión finita e infinita, respectivamente, obtenidos con el algoritmo LMS.

Para determinar $\mathcal{W}(n)$, la ecuación de adaptación se puede escribir como

$$\hat{\boldsymbol{w}}_q(n+1) = \hat{\boldsymbol{w}}_q(n) + \mu e_q(n)\boldsymbol{u}_q(n) + \boldsymbol{\eta}_w(n)$$

donde $\boldsymbol{\eta}(n)$ es el **vector de error de cuantización del gradiente**, resultante de la cuantización del producto $\mu e_q(n)\boldsymbol{u}_q(n)$. Cada componente de $\boldsymbol{\eta}_w(n)$ se suponen decorrelacionados en tiempo y entre sí y tienen varianza σ_w^2.

Utilizando la transformación de coordenadas típica de análisis a $\hat{\boldsymbol{w}}_q$, las características del vector de desajuste $\mathcal{W}(n)$ durante la adaptación y el estado estacionario pueden resumirse en

1. Los coeficientes en el algoritmo LMS son los **más sensibles** de todos los parámetros a la cuantización. Para el caso de **datos de entrada decorrelacionados**, la varianza σ_w^2 es proporcional a $\frac{1}{r(0)\mu}$. Para el caso de **datos de entrada correlacionados**, la varianza σ_w^2 es proporcional a $\frac{1}{\lambda_{min}\mu}$.

2. Para datos de entrada decorrelacionados, las constantes de tiempo de adaptación del vector de desajuste $\mathcal{W}(n)$ son fuertemente dependientes de μ.

3. Para datos de entrada correlacionados, las constantes de tiempo de adaptación del vector de desajuste $\mathcal{W}(n)$ son fuertemente dependientes de la interacción entre μ y λ_{min}.

Algoritmo LMS con pérdidas

Para hacer más robusta la estabilidad de la implementación digital del algoritmo LMS es posible utilizar una técnica denominada como **leakage** (pérdidas).

Basicamente, el leakage evita la ocurrencia de overflow en un contexto de precisión finita introduciendo un compromiso entre minimizar el error medio cuadrático y limitar la energía de la respuesta impulsiva del filtro adaptivo.

Para el algoritmo LMS **leaky** (con pérdidas)

$$J(n) = e^2(n) + \alpha \|\hat{\boldsymbol{w}}(n)\|^2$$

donde α es un parámetro positivo de control. Para datos reales la ecuación de adaptación será

$$\hat{\boldsymbol{w}}(n+1) = (1-\mu\alpha)\hat{\boldsymbol{w}}(n) + \mu e(n)\boldsymbol{u}(n)$$

donde α debe satisfacer la condición $0 \leq \alpha < \frac{1}{\mu}$.

La inclusión $(1-\mu\alpha)$ es equivalente de sumar a la entrada $u(n)$ una secuencia de ruido blanco de media cero y varianza α.

Fenómeno de Cerrojo (stalling)

Este fenómeno ocurre cuando la estimación del gradiente no es suficientemente ruidosa. Especificamente, una implementación digital del algoritmo LMS **para de adaptar** siempre que $\mu e_q(n)u_q(n-i)$ es menor en magnitud que el **bit menos significativo** (LSB) del coeficiente, ó

$$|\mu e_q(n_0)u_q(n_0 - i)| \leq LSB$$

donde n_0 esel instante en el cual el i-ésimo coeficiente deja de adaptar. Suponiendo que esta condición que esta condición es satisfecha, es posible aproximar, $u_q(n_0 - i)$ por su valor **rms**, A_{rms}. Usando este valor la adaptación cesa en

$$|e_q(n)| \leq \frac{LSB}{\mu A_{rms}} = e_D(\mu)$$

que se denomina **error digital residual**.

Para evitar el fenómeno de cerrojo debido a la implementación digital, $e_D(\mu)$ debe harcerse lo más pequeño posible, lo que puede satisfacerse de dos formas

1. El bit menos significativo (LSB) se reduce (cantidad suficientemente grande de bits).

2. Se incrementa el factor de convergencia μ tanto cuanto sea posible.

11.3 Algoritmo RLS

- El algoritmo RLS ofrece una alternativa al algoritmo LMS como herramienta para la solución de problemas de filtrado adaptivo. De sus propiedades se concluye que está caracterizado por una velocidad de convergencia rápida y es relativamente insensible a la dispersión de autovalores de $/symbR(n)$, y tiene un desajuste despreciable (nulo para un ambiente estacionario sin perturbaciones).

- Además, a pesar que es computacionalmente más complejo (con un order de complejidad M^2), la formulación y la implementación del algoritmo RLS es relativamente simple.

- Sin embargo, existe un problema de inestabilidad numérica a ser considerado cuando se implementa este algoritmo con aritmética de precisión finita.

- La **inestabilidad numérica** ó **divergencia explosiva** del algoritmo RLS es de la misma naturaleza que la experimentada por el filtro Kalman.

- Además, el problema puede asociarse a que $\boldsymbol{P}(n)$ se calcula como la diferencia entre dos matrices definidas no negativas. La divergencia explosiva del algoritmo ocurre cuando $\boldsymbol{P}(n)$ pierde la propiedad de ser positiva definida o hermitiana simétrica.

Inicialización
$\boldsymbol{P}(0) = \delta^{-1}\boldsymbol{I}$, δ una constante positiva pequeña.
$\hat{\boldsymbol{w}}(0) = \boldsymbol{0}$
Para cada n, $n = 1, 2, \cdots$, calcular
$\boldsymbol{\pi}(n) = \boldsymbol{P}(n-1)\boldsymbol{u}(n)$
$r(n) = \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n)}$
$\boldsymbol{k}(n) = r(n)\boldsymbol{\pi}(n)$
$\xi(n) = d(n) - \hat{\boldsymbol{w}}^H(n-1)\boldsymbol{u}(n)$
$\hat{\boldsymbol{w}}(n) = \hat{\boldsymbol{w}}(n-1) + \boldsymbol{k}(n)\xi^*(n)$
$\boldsymbol{P}(n) = Tri\{\lambda^{-1}[\boldsymbol{P}(n-1) - r(n)\boldsymbol{\pi}(n)]\}$

Tabla 5: Algoritmo RLS que preserva la simetría de $\boldsymbol{P}(n)$

- Una solución eficiente que mantiene la simetría de $\boldsymbol{P}(n)$ se muestra en la tabla.
- La eficiencia de este algoritmo se logra porque calcula solo la parte triangular inferior/superior de la matriz $\boldsymbol{P}(n)$, definida por el operador $Tri\{.\}$ y luego completa el resto de la matriz para preservar la simetría.

Modelo de propagación del error

De acuerdo al algoritmo de la tabla anterior, las recursividades involucradas en el cálculo de $\boldsymbol{P}(n)$ son

$$\begin{aligned}
\boldsymbol{\pi}(n) &= \boldsymbol{P}(n-1)\boldsymbol{u}(n) \\
r(n) &= \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n)} \\
\boldsymbol{k}(n) &= r(n)\boldsymbol{\pi}(n) \\
\boldsymbol{P}(n) &= Tri\{\lambda^{-1}[\boldsymbol{P}(n-1) - r(n)\boldsymbol{\pi}(n)]\}
\end{aligned}$$

Consideramos la propagación de un **único** error de cuantización en el instante $n-1$ hacia los cálculos siguientes y suponemos que no existen otros errores de cuantización. En particular, es posible modelar esto como

$$\boldsymbol{P}_q(n-1) = \boldsymbol{P}(n-1) + \boldsymbol{\eta}_p(n-1)$$

donde $\boldsymbol{\eta}_p(n-1)$ es la **matriz de error** asociada a la cuantización de $\boldsymbol{P}(n-1)$ y además

$$\boldsymbol{\pi}_q(n) = \boldsymbol{\pi}(n) + \boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n)$$

es el valor cuantizado de $\boldsymbol{\pi}(n)$. Si $r_q(n)$ es el valor cuantizado de $r(n)$, usando la definición de $r(n)$ es posible escribir

$$\begin{aligned}
r_q(n) &= \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}_q(n)} \qquad (155) \\
&= \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n) + \boldsymbol{u}^H(n)\boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n)} \\
&= \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n)} \left(1 + \frac{\boldsymbol{u}^H(n)\boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n)}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n)}\right)^{-1}
\end{aligned}$$

$$= \frac{1}{\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n)} - \frac{\boldsymbol{u}^H(n)\boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n)}{(\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n))^2} + O(\boldsymbol{\eta}_p^2)$$

$$= r(n) - \frac{\boldsymbol{u}^H(n)\boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n)}{(\lambda + \boldsymbol{u}^H(n)\boldsymbol{\pi}(n))^2} + O(\boldsymbol{\eta}_p^2) \quad (156)$$

donde $O(\boldsymbol{\eta}_p^2)$ denota el orden de magnitud $\|\boldsymbol{\eta}_p\|^2$.

En una situación práctica, la cantidad escalar $r(n)$ de precisión infinita es **no negativa** y toma valores entre cero y $1/\lambda$.

Por otro lado, si $\boldsymbol{u}^H\boldsymbol{\pi}(n)$ es pequeño comparado con λ y $\lambda << 1$, entonces de acuerdo a las ecuaciones anteriores es posible que en un contexto de precisión finita $r_q(n)$ tome valores **negativos** mayores que $1/\lambda$. Cuando esto sucede, el algoritmo RLS exhibe divergencia explosiva.

El valor cuantizado del vector de ganancia $\boldsymbol{k}(n)$ puede escribirse como

$$\boldsymbol{k}_q(n) = r_q(n)\boldsymbol{\pi}_q(n) = \boldsymbol{k}(n) + \boldsymbol{\eta}_k(n)$$

donde $\boldsymbol{\eta}_k(n)$ es ek **error de cuantización del vector ganancia**, definido por

$$\boldsymbol{\eta}_k(n) = r(n)(\boldsymbol{I} - \boldsymbol{k}(n)\boldsymbol{u}^H(n))\boldsymbol{\eta}_p(n-1)\boldsymbol{u}(n) + O(\boldsymbol{\eta}_p^2)$$

Finalmente, usando la ecuación (156), es posible obtener el error de cuantización obtenido al calcular $\boldsymbol{P}(n)$,

$$\boldsymbol{\eta}_p(n) = \lambda^{-1}(\boldsymbol{I} - \boldsymbol{k}(n)\boldsymbol{u}^H(n))\boldsymbol{\eta}_p(n-1)(\boldsymbol{I} - \boldsymbol{k}(n)\boldsymbol{u}^H(n))^H \quad (157)$$

donde se ha despreciado el término $O(\boldsymbol{\eta}_p^2)$. Sobre la base de esta ecuación, si es posible asumir que $\boldsymbol{\eta}_p^H(n-1) = \boldsymbol{\eta}_p(n-1)$, entonces el algoritmo anterior conserva la propiedad de simetría, o lo que es lo mismo, es estable numericamente.

La ecuación (157) define el **mecanismo de propagación del error** para el algoritmo RLS sobre la base de un único error de cuantización en $\boldsymbol{P}(n-1)$. La matriz $\boldsymbol{I}-\boldsymbol{k}(n)\boldsymbol{u}^H(n)$ describe como el error de cuantización $\boldsymbol{\eta}_p(n-1)$ se propaga a través del algoritmo. Usando la definición original del vector ganancia se tiene que

$$\boldsymbol{k}(n)=\boldsymbol{\Phi}^{-1}(n)\boldsymbol{u}(n)$$

y es posible escribir

$$\boldsymbol{I}-\boldsymbol{k}(n)\boldsymbol{u}^H(n)=\boldsymbol{I}-\boldsymbol{\Phi}^{-1}(n)\boldsymbol{u}(n)\boldsymbol{u}^H(n) \qquad (158)$$

Teniendo en cuenta además que

$$\boldsymbol{\Phi}(n)=\lambda\boldsymbol{\Phi}(n-1)+\boldsymbol{u}(n)\boldsymbol{u}^H(n)$$

que multiplicada por $\boldsymbol{\Phi}^{-1}(n)$ y reordenando permite obtener

$$\boldsymbol{I}-\boldsymbol{\Phi}^{-1}(n)\boldsymbol{u}(n)\boldsymbol{u}^H(n)=\lambda\boldsymbol{\Phi}^{-1}(n)\boldsymbol{\Phi}(n-1) \qquad (159)$$

Comparando las ecuaciones (158) y (159) es posible deducir que

$$\boldsymbol{I}-\boldsymbol{k}(n)\boldsymbol{u}^H(n)=\lambda\boldsymbol{\Phi}^{-1}(n)\boldsymbol{\Phi}(n-1)$$

Considerando el efecto del error de cuantización $\boldsymbol{\eta}_p(n_0)$ inducido en el instante $n_0\leq n$, cuando se utiliza el algoritmo RLS de la tabla 5 y de acuerdo al modelo de propagación del error, se tiene que

$$\boldsymbol{\eta}_p(n)=\lambda^{-(n-n_0}\boldsymbol{\varphi}(n,n_0)\boldsymbol{\eta}_p(n_0)\boldsymbol{\varphi}^H(n,n_0), \quad n\geq n_0 \qquad (160)$$

donde $\boldsymbol{\varphi}(n,n_0)$ es una **matriz de transición** definida por

$$\boldsymbol{\varphi}(n, n_0) = (\boldsymbol{I} - \boldsymbol{k}(n)\boldsymbol{u}^H(n)) \cdots (\boldsymbol{I} - \boldsymbol{k}(n_0+1)\boldsymbol{u}^H(n_0+1))$$

de donde es posible obtener

$$\boldsymbol{\varphi}(n, n_0) = \lambda^{n-n_0}\boldsymbol{\Phi}^{-1}(n)\boldsymbol{\Phi}(n_0)$$

En base a $\boldsymbol{\Phi}(n) = \sum_{i=1}^{n}\lambda^{n-i}\boldsymbol{u}(i)\boldsymbol{u}^H(i)$, $\boldsymbol{u}(n)$ se dice de **excitación persistente uniforme** si, para n grande, existe algún $a > 0$ y $n > 0$ tal que la siguiente condición es satisfecha

$$\boldsymbol{\Phi}(n) \geq a\boldsymbol{I}, \quad \text{para} n \geq N$$

lo que es lo mismo que decir que $\boldsymbol{\Phi}(n)$ es positiva definida. La condición de excitación persistente no solo garantiza la positividad de $\boldsymbol{\Phi}(n)$ sino también que la norma de la matriz está uniformemente aotada para $n \geq N$, como mostrado por

$$\|\boldsymbol{\Phi}^{-1}(n)\| \leq \frac{1}{a}, \quad \text{para} n \geq N$$

Volviendo a $\boldsymbol{\varphi}(n, n_0)$, y como $\|\boldsymbol{A}\boldsymbol{B}\| \leq \|\boldsymbol{A}\|\|\boldsymbol{B}\|$, se tiene que

$$\begin{aligned} \|\boldsymbol{\varphi}(n, n_0)\| &= \lambda^{n-n_0}\|\boldsymbol{\Phi}^{-1}(n)\| \, \|\boldsymbol{\Phi}(n_0)\| \\ &= \frac{\lambda^{n-n_0}}{a}\|\boldsymbol{\Phi}^{-1}(n)\| \end{aligned}$$

Finalmente, usando la ecuación de propagación del error, es posible usar la norma de $\boldsymbol{\eta}_p(n)$ como

$$\|\boldsymbol{\eta}_p(n)\| \leq \lambda^{-(n-n_0)}\|\boldsymbol{\varphi}(n, n_0)\| \, \|\boldsymbol{\eta}_p(n-1)\| \, \|\boldsymbol{\varphi}^H(n, n_0)\|$$

la que puede reescribirse como

$$\|\boldsymbol{\eta}_p(n)\| \leq \lambda^{-(n-n_0)} M, \qquad n \geq n_0 \tag{161}$$

donde M es un número positivo definido por

$$M = \frac{1}{a^2} \|\boldsymbol{\Phi}(n_0)\|^2 \|\boldsymbol{\eta}_p(n-1)\|$$

La ecuación (161) determina que el algoritmo RLS de la tabla es **exponencialmente estable**, en el sentido que un único error de cuantización $\boldsymbol{\eta}_p(n_0)$ en la estimación de $\boldsymbol{P}(n_0)$ decae exponencialmente con $\lambda < 1$ (o sea tiene memoria finita).

Sin embargo, la propagación de error de propagación único para el caso de memoria creciente (o sea $\lambda = 1$) **no es contractiva**.

La razón de esto último es que cuando $\lambda = 1$ ni $\boldsymbol{\varphi}(n, n_0) \leq \boldsymbol{I}$, ni $\|\boldsymbol{\varphi}(n, n_0)\| \leq 1$ se mantienen, aún si $\boldsymbol{u}(n)$ es de excitación persistente.

En consecuencia, la acumulación de errores numéricos puede causar que el algoritmo se inestabilize.

Fenomeno de cerrojo

En forma similar al caso del algoritmo LMS, una segunda forma de divergencia, denominada como **fenómeno de cerrojo**, ocurre cuando los coeficientes del algoritmo RLS dejan de adaptar.

En particular, esto ocurre cuando los elementos cuantizados de $\boldsymbol{P}(n)$ se vuelven muy pequeños, tal que la multiplicación por $\boldsymbol{P}(n)$ es equivalente a la multiplicación por una matriz cero.

Claramente, este fenómeno es independiente de la forma de implementación del algoritmo RLS.

El fenómeno de cerrojo está directamente relacionado con λ y σ_u^2. Suponiendo que λ es próximo a 1, ya se discutió que

$$E[\mathbf{\Phi}(n)] \cong \frac{\boldsymbol{R}}{1-\lambda}, \qquad n \text{ grande}$$

entonces

$$E[\boldsymbol{P}(n)] = E[\mathbf{\Phi}^{-1}(n)] \cong (E[\mathbf{\Phi}(n)])^{-1}$$

de donde

$$E[\boldsymbol{P}(n)] \cong (1-\lambda)\boldsymbol{R}^{-1}$$

Suponiendo que $u(n)$ es ESA, es posible definir la **matriz correlación normalizada** $\mathcal{R}$, con elementos en la diagonal iguales a 1 y elementos fuera de la diagonal menores que 1 y σ_u^2 la varianza, donde

$$E[\boldsymbol{P}(n)] \cong \left(\frac{1-\lambda}{\sigma_u^2}\right)\mathcal{R}^{-1}, \qquad \text{para } n \text{ grande}$$

que muestra que el algoritmo RLS puede no llegar a la convergencia si λ es próximo a 1 y/o σ_u^2 es grande. En consecuencia, puede evitarse el fenómeno de cerrojo en el algoritmo RLS estándar utilizando un número suficientemente grande de bits de representación para el cálculo de $\boldsymbol{P}(n)$.