

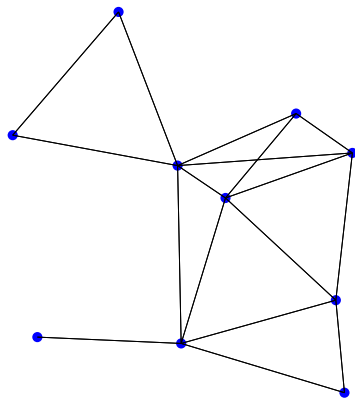
Diffusion Adaptation over Networks

Javier Zazo

7 November 2012

Gráfico de la Red

- ▶ Nodos de la red: cada uno realiza sus propias mediciones
- ▶ Comparten información y colaboran entre sí con sus vecinos
- ▶ Realizan medidas independientes



Mean-Square-Error Estimation

- ▶ Modelo de sistema \Rightarrow Modelo Lineal

$$d_k(i) = u_{k,i}w^0 + v_k(i)$$

- ▶ k es el usuario, i es el tiempo, w_0 es el vector a estimar (columna), u pertenece al modelo lineal (vector fila), $v_k(i)$ es AWGN
- ▶ Cada usuario minimiza el error local

$$\begin{aligned} J_k &= \min_w \mathbb{E} |d_k(i) - u_{k,i}w|^2 \\ &= \min_w J_{k,min} + (w - w^0)^T R_{u,k} (w - w^0) \end{aligned}$$

Non-Cooperative solution

- ▶ Cada usuario usa LMS para encontrar el parámetro w_0 (desconociendo los estadísticos del canal)

$$\begin{aligned}e_k(i) &= d_k(i) - u_{k,i}w_{k,i} \\ w_{k,i} &= w_{k,i-1} + \mu_k u_{k,i}^T e_k(i)\end{aligned}$$

- ▶ Definimos los vectores de error

$$\begin{aligned}\tilde{w}_{k,i} &\triangleq w^0 - w_{k,i} \\ e_{a,k}(i) &\triangleq u_{k,i}\tilde{w}_{k,i} \\ e_k(i) &= d_k(i) - u_{k,i}w_{k,i} \\ &= e_{a,k}(i) + v_k(i)\end{aligned}$$

Solución No Cooperativa

- ▶ Y calculamos sus valores estimados

$$\begin{aligned}\mathbb{E}|e_k(i)|^2 &= \mathbb{E}|e_{a,k}(i)|^2 + \sigma_{v,k}^2 \\ MSE_k &= \lim_{i \rightarrow \infty} \mathbb{E}|e_k(i)|^2 \\ &= EMSE_k + \sigma_{v,k}^2\end{aligned}$$

- ▶ Observamos que LMS produce un error de estimación mayor que el óptimo (cuando se conocen los estadísticos del canal)
- ▶ Este error depende del parámetro de adaptación y de la varianza del error



$$EMSE_K \approx \mu_k \sigma_{v,k}^2 \text{Tr}(R_{u,k})/2$$

- ▶ Hay un incentivo para que los nodos colaboren!

Adaptación Cooperativa mediante Difusión

- ▶ Definimos una nueva función global de coste

$$J^{glob}(w) \triangleq \sum_{k=1}^N J_k(w) = \sum_{k=1}^N \mathbb{E} |d_k(i) - u_{k,i}w|^2$$

- ▶ También redefinimos la función de coste local de cada sensor

$$\begin{aligned} J_l^{loc}(w) &\triangleq \sum_{k \in N_l} c_{k,l} J_k(w) \\ &= J_{l,min}^{loc} + (w - w^0)^T R_{u,l} (w - w^0) \end{aligned}$$

donde $\sum_{l \in N} c_{k,l} = 1$ para todo $k \in N$.

- ▶ Y lo incluimos en la función global

$$J^{glob}(w) = J_k^{local} + \sum_{l \neq k} J_{l,min}^{loc} + (w - w^0)^T R_{u,l} (w - w^0)$$

Aproximaciones

- ▶ $\sum_{l \neq k} J_k(w) \rightarrow \sum_{l \in N_k / \{k\}} J_k(w)$
- ▶ $(w - w^0)^T R_{u,l} (w - w^0) \approx b_{lk} \|w - w^0\|^2$
- ▶ Función a optimizar por cada nodo k

$$J_k^{glob''}(w) = \sum_{l \in N_k} c_{lk} J_l(w) + \sum_{l \in N_k / \{k\}} b_{lk} \|w - w^0\|^2$$

donde $\sum_{l \in N} c_{k,l} = 1$ para todo $l \in N$.

Steepest Descent

$$w_{k,i} = w_{k,i-1} - \mu_k \nabla_w J_k^{glob''} (w_{k,i-1})$$

- ▶ Desarrollando la expresión obtenemos

$$w_{k,i} = w_{k,i-1} - \underbrace{\mu_k \sum_{l \in N_k} c_{lk} \nabla_w J_l (w_{k,i-1})}_{\varphi_{k,i}} - \mu_k b_{lk} (w_{k,i-1} - w^0)$$

- ▶ Para el caso del LMS

$$\nabla_w J_l (w_{k,i-1}) \approx -u_{l,i}^T (d_l(i) - u_{l,i} w_{k,i})$$

- ▶ Aproximamos $w^0 \approx \varphi_{l,i}$ de los nodos vecinos, y además, una vez calculado $\varphi_{k,i}$ sustituimos $\varphi_{k,i} \rightarrow w_{k,i-1}$

$$a_{lk} = \mu_k b_{lk}, \quad a_{kk} = 1 - \mu_k \sum_{l \in N_k} b_{lk}$$

Adapt-then-Combine (ATC) Diffusion Strategy

$$\varphi_{k,i} = w_{k,i} + \mu_k \sum_{l \in N_k} c_{lk} \nabla_w J_l(w_{k,i-1})$$

$$w_{k,i} = \sum_{l \in N_k} a_{lk} \varphi_{l,i}$$

sujeto a $c_{lk} \geq 0$, $\sum_{k \in N} c_{lk} = 1$ y $a_{lk} \geq 0$, $\sum_{k \in N} a_{lk} = 1$.

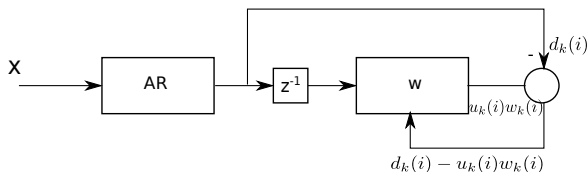
- ▶ Existen otras estrategias como Combine-then-Adapt, y más variantes.

Modelo AR

- ▶ Sistema AR, con distintas excitaciones para cada nodo
- ▶ Ecuación

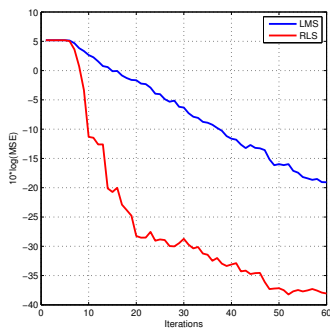
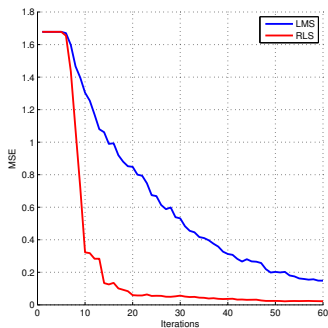
$$d_k(i) = \sum_{m=1}^M \beta_m d_k(i - m) + v_k(i)$$

- ▶ $w^0 = [\beta_1, \dots, \beta_M]$
- ▶ $u_k(i) = [d_k(i - 1), \dots, d_k(i - M)]$



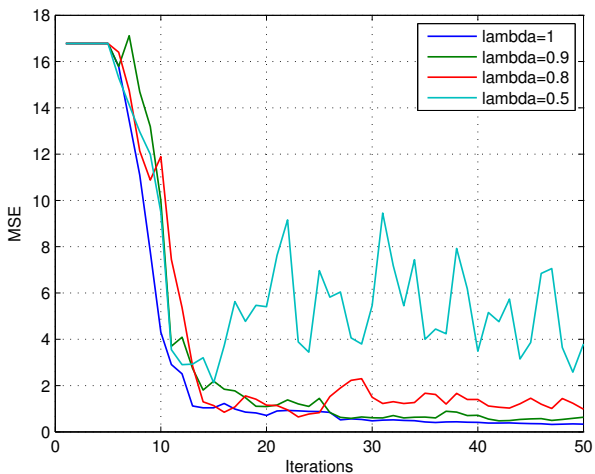
Simulaciones: LMS y RLS

- ▶ $M=5$, muestras = 60, $w^0 = [\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}]$
- ▶ Averaged over 10 iterations



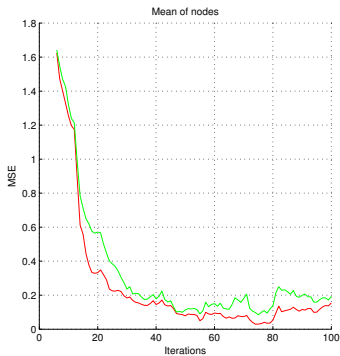
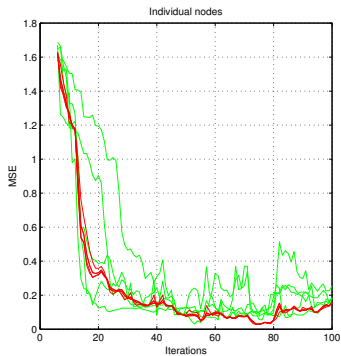
Factor de Olvido en RLS

- ▶ Averaged over 10 iterations



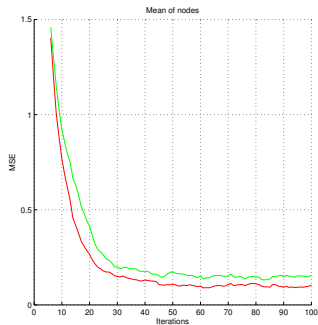
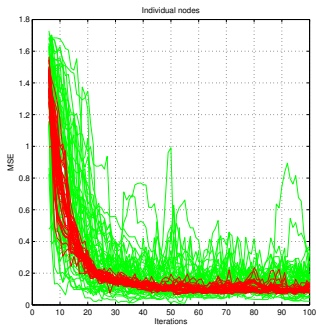
Simulaciones: Diffusion LMS

- ▶ 5 Nodes, 100 samples



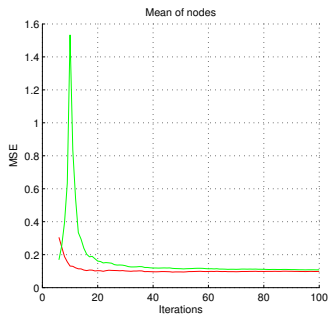
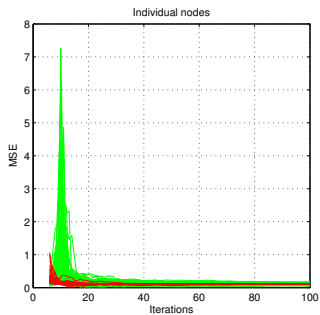
Simulaciones: Diffusion LMS

- ▶ 50 Nodes, 100 samples



Simulaciones: Diffusion RLS

- ▶ 50 Nodes, 100 samples



Diffusion RLS and LMS

- ▶ 50 Nodes, 100 samples

