

# Estimación distribuida con redes de difusión: algoritmo con combinadores adaptativos y agentes que preservan estimaciones locales

Jesus Fernandez-Bes<sup>(1)</sup>, Luis A. Azpicueta-Ruiz<sup>(1)</sup>, Magno T. M. Silva<sup>(2)</sup>, Jerónimo Arenas-García<sup>(1)</sup>  
jesusfbes@tsc.uc3m.es, lazpicueta@tsc.uc3m.es, magno@lps.usp.br, jarenas@tsc.uc3m.es.

<sup>(1)</sup>Dpto. de Teoría de la Señal y Comunicaciones. Universidad Carlos III de Madrid.  
Avda. Universidad 30, 28911 Leganés, Madrid.

<sup>(2)</sup>Dpto. de Sistemas Electrónicos. Universidad de São Paulo. São Paulo 05508-010, Brasil.

**Abstract**—Adaptive diffusion networks have become an attractive approach to estimate an unknown set of parameters in a distributed manner employing the information sensed by a group of agents. Adaptive diffusion schemes are mainly based on two steps: adaptation and combination. The way in which each node calculates a combined estimation using the estimations received by its neighborhood is crucial, making adaptive combination schemes necessary in scenarios where, for instance, each node obtains measurements with different quality. In this paper, we describe an adaptive diffusion scheme that presents two main characteristics: 1) each node preserves a purely local estimation based on its local measurements; and 2) each node adaptively combines this estimation with those received from its neighbors, minimizing a least-squares cost function. We experimentally show that our scheme outperforms the state-of-the-art algorithm for diffusion networks with adaptive combiners.

## I. INTRODUCCIÓN

Recientemente, las redes de difusión adaptativas han suscitado un gran interés para la estimación distribuida de una serie de parámetros explotando la información recogida por diferentes agentes [1], [2]. En este artículo consideraremos la identificación de un vector de parámetros desconocido  $\mathbf{w}_0(n)$  mediante una red de  $N$  sensores, asumiendo que cada nodo tiene acceso a una medida de la señal deseada  $d_k(n)$  y del regresor  $\mathbf{u}_k(n)$ , relacionados según el conocido modelo

$$d_k(n) = \mathbf{u}_k^T(n)\mathbf{w}_0(n) + v_k(n), \quad \text{con } k = 1, \dots, N, \quad (1)$$

siendo  $v_k(n)$  el ruido de medida. Nótese que el vector de parámetros  $\mathbf{w}_0(n)$  es común a las ecuaciones (1) de todos los nodos de la red, y precisamente el objetivo de la red consiste en mejorar las estimaciones de cada nodo mediante el intercambio de información entre nodos vecinos.

Aunque existen otras soluciones para la estimación colaborativa de  $\mathbf{w}_0(n)$ , las redes de difusión adaptativas resultan una opción muy atractiva gracias a que el procesamiento es totalmente distribuido (a diferencia de los algoritmos centralizados, no se requiere de un nodo central con gran capacidad de cómputo que reciba y procese las medidas de los  $N$  sensores). Además, esta estrategia proporciona a la red capacidad de adaptación cuando el vector a estimar varía en el tiempo, ya que, a diferencia de las soluciones basadas en consenso, las redes

de difusión adaptan sus estimaciones continuamente, lo que permite un buen comportamiento ante cambios abruptos o sostenidos de  $\mathbf{w}_0(n)$ .

Actualmente, se han propuesto soluciones basadas en redes de difusión adaptativas para diferentes ámbitos, entre los que destacan la localización y seguimiento de blancos [2], y el sensado espectral en redes de telefonía móvil [2], entre otros.

Uno de los esquemas más relevantes es el conocido como ATC (*adapt-then-combine*) [2]. El algoritmo se basa en dos etapas: la primera de ellas, denominada *adaptación*, donde la estimación de cada nodo se actualiza utilizando las variables sensadas  $\mathbf{u}_k(n)$  y  $d_k(n)$  siguiendo un algoritmo adaptativo específico; y la segunda de ellas, denominada *combinación*, donde se combina esta estimación recientemente actualizada con las estimaciones recibidas de los nodos vecinos, para conformar una estimación mejorada del parámetro  $\mathbf{w}_0(n)$ .

Uno de los elementos más importantes en el algoritmo es el procedimiento para combinar la información recibida de los vecinos, es decir, para ajustar el peso que utiliza el nodo  $k$  para ponderar la estimación recibida de su  $\ell$ -ésimo vecino. Aunque existen algoritmos que utilizan pesos estáticos [2], [3], parece sensato pensar que se pueden obtener ventajas en el esquema de difusión si éstos se calculan adaptativamente. De esta forma, se puede lograr que un nodo  $k$  no utilice la estimación recibida de su vecino  $\ell$ , si ésta se ha basado en datos con baja relación señal a ruido (SNR), sin más que asignarle un peso nulo  $c_{\ell k}(n) = 0$ . Sin embargo, la mayoría de algoritmos de difusión adaptativa que implementan pesos adaptativos se basan en asunciones que son válidas principalmente en estado estacionario [3]–[5]. Esto hace que, si se diseña una red con nodos que presentan diferentes capacidades de adaptación (por ejemplo diferentes velocidades de convergencia), los nodos rápidos retrasen su adaptación debido a que incorporan, con un valor  $c_{\ell k}(n)$  alto, la estimación de los nodos lentos (de peor calidad en convergencia pero menor error en estado estacionario), degradando el comportamiento global de la red en términos de convergencia.

En este artículo se describe un algoritmo reciente de difusión adaptativa con pesos adaptativos que soluciona el problema antes mencionado, dotando al algoritmo de un buen

funcionamiento en convergencia, estacionario y seguimiento [6], [7]. Para ello, en la Sección 2 se incluye una descripción detallada del esquema de difusión propuesto, así como del algoritmo utilizado para adaptar los pesos en la etapa de combinación. La Sección 3 detalla la evaluación experimental de nuestro esquema comparando con el algoritmo considerado actualmente como estado del arte. El artículo finaliza con las conclusiones de nuestro trabajo así como una discusión de las posibles líneas de investigación futuras.

## II. DESCRIPCIÓN DEL ALGORITMO ADN-LS

En la presente sección se incluye una descripción detallada del esquema propuesto. Primeramente, se procede a relatar la estrategia de difusión desarrollada, la cual puede considerarse una variante del algoritmo ATC. Posteriormente, se describe el algoritmo utilizado para actualizar los pesos de la etapa de combinación, el cual se basa en la minimización de una suma de errores cuadráticos ponderados, de forma similar a un algoritmo tipo mínimos cuadrados. De esta forma, el algoritmo que presentamos se denomina ADN-LS (*Adaptive Diffusion Network with Least-Squares combiners*).

### A. Esquema de difusión

Al igual que el algoritmo ATC, nuestro esquema de difusión también se puede descomponer en:

- Una etapa de adaptación:

$$\psi_k(n) = \psi_k(n-1) + \tilde{\mu}_k(n)\mathbf{u}_k(n)e_k(n), \quad (2)$$

donde se ha utilizado el algoritmo NLMS (*Normalized Least-Mean Squares*) [8], siendo  $\tilde{\mu}_k(n) = \frac{\mu_k}{\delta + \|\mathbf{u}_k(n)\|^2}$ , con  $\mu_k$  el paso de adaptación del NLMS y  $\delta$  una pequeña constante para evitar división por cero. Cada nodo adapta  $\psi_k(n)$  buscando minimizar la potencia de su error de estimación local:

$$e_k(n) = d_k(n) - \psi_k^T(n-1)\mathbf{u}_k(n) = d_k(n) - y_k(n), \quad (3)$$

siendo  $y_k(n)$  la estimación de  $d_k(n)$  que realiza cada agente basada en su vector de pesos local  $\psi_k(n)$ .

- Una etapa de combinación:

$$\mathbf{w}_k(n) = c_{kk}(n)\psi_k(n) + \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n)\mathbf{w}_\ell(n-1), \quad (4)$$

donde  $\bar{\mathcal{N}}_k$  representa el conjunto de los  $\bar{N}_k$  nodos vecinos al nodo  $k$ , excluyendo al propio nodo.

Sin embargo, el algoritmo ADN-LS presenta una importante diferencia con respecto al algoritmo ATC, ya que nuestro algoritmo conserva una estimación puramente local del vector de pesos. Por el contrario, el algoritmo ATC realimenta en (2) el vector de pesos combinado, i.e., sustituyendo el primer sumando de la parte derecha de (2) por  $\mathbf{w}_k(n-1)$  [2].

### B. Adaptación de los combinadores de la red

Al igual que otros algoritmos de difusión con pesos adaptativos, nuestro esquema considera que cada nodo adapta  $\bar{N}_k$  pesos  $c_{\ell k}(n)$ , y calcula  $c_{kk}(n) = 1 - \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n) = 1 - \mathbf{1}^T \bar{\mathbf{c}}_k(n)$ , donde  $\mathbf{1}$  es un vector unitario de igual longitud que el vector  $\bar{\mathbf{c}}_k(n)$ , el cual almacena los pesos  $c_{\ell k}(n) \forall \ell \in \bar{\mathcal{N}}_k$ .

La actualización de los pesos de combinación de cada nodo se lleva a cabo con el objetivo de minimizar el error cuadrático medio de la red, definido como

$$\text{MSE}(n) = \frac{1}{N} \sum_{k=1}^N \text{MSE}_k(n) = \frac{1}{N} \sum_{k=1}^N \mathbb{E}\{\check{e}_k^2(n)\}, \quad (5)$$

donde  $\mathbb{E}\{\cdot\}$  representa la esperanza matemática, y

$$\check{e}_k(n) = d_k(n) - \mathbf{w}_k^T(n-1)\mathbf{u}_k(n) = d_k(n) - \check{y}_k(n), \quad (6)$$

representa el error combinado, siendo  $\check{y}_k(n)$  la salida combinada. Sin embargo, puesto que cada término  $\text{MSE}_k(n)$  en (5) sólo depende de los pesos  $\bar{\mathbf{c}}_k(n)$ , la adaptación de dichos pesos puede realizarse de forma local.

Existen diferentes algoritmos para poder aproximar la minimización de la esperanza matemática en (5) con los datos disponibles en cada nodo. Sin embargo, en este esquema adaptamos los pesos de la combinación buscando minimizar una suma ponderada de errores al cuadrado:

$$J_k(n) = \sum_{i=1}^n \lambda(n,i) \check{e}_k^2(n,i) = \sum_{i=1}^n \lambda(n,i) [d_k(i) - \check{y}_k(n,i)]^2, \quad (7)$$

siendo  $\lambda(n,i)$  una ventana temporal, y donde

$$\begin{aligned} \check{y}_k(n,i) &= \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n) \check{y}_{k,\ell}(i) + \left[ 1 - \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n) \right] y_k(i) \\ &= y_k(i) + \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n) [\check{y}_{k,\ell}(i) - y_k(i)] \end{aligned} \quad (8)$$

es la salida combinada del nodo  $k$  utilizando los pesos en el instante  $n$  y el regresor  $i$ -ésimo, con  $\check{y}_{k,\ell}(i) = \mathbf{w}_{\ell}^T(i-1)\mathbf{u}_k(i)$ .

Considerando (8), el error  $\check{e}_k(n,i)$  se puede expresar como

$$\check{e}_k(n,i) = e_k(i) + \sum_{\ell \in \bar{\mathcal{N}}_k} c_{\ell k}(n) [y_k(i) - \check{y}_{k,\ell}(i)]. \quad (9)$$

Sustituyendo (9) en la función de coste (7), derivando con respecto a cada peso de la combinación  $c_{mk}(n)$ , con  $m \in \bar{\mathcal{N}}_k$ , igualando a cero dichas derivadas y tras varias manipulaciones algebraicas se llega a la siguiente ecuación para cada vecino de cada nodo

$$\sum_{i=1}^n \sum_{\ell \in \bar{\mathcal{N}}_k} \lambda(n,i) c_{\ell k}(n) \check{y}_{k,\ell}(i) \check{y}_{k,m}(i) = \sum_{i=1}^n \lambda(n,i) e_k(i) \check{y}_{k,m}(i), \quad (10)$$

donde  $\check{y}_{k,m}(n) \triangleq \check{y}_{k,m}(n) - y_k(n)$ , con  $m \in \bar{\mathcal{N}}_k$ , por conveniencia notacional.

Puesto que cada nodo cuenta con  $\bar{N}_k$  ecuaciones similares a (10), éstas se pueden expresar de forma matricial

$$\mathbf{P}_k(n) \bar{\mathbf{c}}_k(n) = \mathbf{z}_k(n), \quad (11)$$

siendo  $\mathbf{P}_k(n)$  una matriz cuadrada de tamaño  $\bar{N}_k$  con componentes

$$[\mathbf{P}_k(n)]_{f,g} = \sum_{i=1}^n \lambda(n,i) \tilde{y}_{k,\bar{b}_k^{(f)}}(i) \tilde{y}_{k,\bar{b}_k^{(g)}}(i), \quad (12)$$

donde  $f, g = 1, 2, \dots, \bar{N}_k$  son los índices de la matriz; y  $\mathbf{z}_k(n)$  es un vector columna con  $\bar{N}_k$  elementos, que vienen dados por

$$z_k^{(f)}(n) = \sum_{i=1}^n \lambda(n,i) e_k(i) \tilde{y}_{k,\bar{b}_k^{(f)}}(i), \quad (13)$$

para  $f = 1, 2, \dots, \bar{N}_k$ . El vector  $\bar{\mathbf{b}}_k$  contiene la denominación de los vecinos del nodo  $k$ , siendo  $\bar{b}_k^{(g)}$  en (12) y (13) su  $g$ -ésimo elemento, i.e., el índice del  $g$ -ésimo vecino del nodo  $k$ .

Por lo tanto, la solución a nuestro problema puede obtenerse utilizando (11), sin más que

$$\bar{\mathbf{c}}_k(n) = \mathbf{P}_k^{-1}(n) \mathbf{z}_k(n). \quad (14)$$

Puede comprobarse que el formato de la solución es el habitual en los problemas de mínimos cuadrados ponderados; para una explicación más detallada puede consultarse [9].

El algoritmo ADN-LS, tal y como se ha descrito, no impone ninguna restricción adicional al valor de los pesos  $c_{\ell k}(n)$ , salvo la condición de que todos los pesos involucrados en la combinación (4) deben sumar uno (combinación afín). Sin embargo, otros algoritmos de la literatura incluyen la condición de convexidad en el cálculo de sus pesos de combinación. En nuestro caso, se puede incorporar una restricción adicional al algoritmo, haciendo que tras cada iteración en la que se calculan los pesos en (14), éstos se proyecten al simplex unitario, dando lugar a una combinación convexa. Ambas variantes del algoritmo serán evaluadas experimentalmente en la siguiente sección.

### III. EVALUACIÓN EXPERIMENTAL

En esta sección se evalúa experimentalmente el funcionamiento del algoritmo ADN-LS, tanto en su versión afín como convexa, en un escenario estacionario de identificación de planta. Para ello, consideramos una red formada por 15 nodos interconectados según la topología mostrada en la Figura 1.

Las señales involucradas, tanto los regresores  $\mathbf{u}_k(n)$  como el ruido de medida  $v_k(n)$ , son temporal y espacialmente independientes. Con respecto al ruido de medida, éste se considera blanco, gaussiano, de media nula y con varianza ajustada para obtener una SNR diferente en cada nodo, entre 20 dB y 30 dB. En relación a la señal de entrada  $\mathbf{u}_k(n)$ , se evalúan dos casos diferentes: el primero, en el cual se ha generado un ruido blanco obtenido de una gaussiana multidimensional de medias nulas y matriz de covarianza igual a la identidad; y el segundo, en el cual se ha generado la señal de entrada como la salida de un proceso autoregresivo de primer orden con función de transferencia  $0,6/(1-0,8z^{-1})$ , cuya entrada es un ruido blanco de media nula y varianza unidad.

Para la fase de adaptación, cada nodo utiliza un filtro NLMS de 50 coeficientes, de igual longitud que la planta desconocida  $\mathbf{w}_0(n)$ . Sin embargo, con el objetivo de incorporar diversidad

en la adaptación de los diferentes nodos, éstos utilizan pasos de adaptación diferentes, seleccionando  $\mu_k = 1$  para los nodos sombreados en la Figura 1 y  $\mu_k = 0,1$  para los demás. Para la fase de combinación del algoritmo ADN-LS se ha elegido una ventana  $\lambda(n,i)$  con forma rectangular. El tamaño de dicha ventana impone el conocido compromiso *vs* varianza [9], seleccionando un tamaño de 2500 muestras con el objetivo de tener un buen comportamiento en estado estacionario.

En la evaluación experimental se incluye la comparación con un algoritmo de difusión que adapta los pesos de la combinación utilizando estimaciones de varianza relativa adaptativa, propuesto en [5] y que denominamos como ARV (Adaptive Relative Variance). Este algoritmo de actualización de los combinadores se utilizará con un esquema ATC estándar usando filtros NLMS con los mismos ajustes en la fase de adaptación, y donde la fase de combinación se controla mediante un parámetro  $\nu$ . Dicho parámetro ha sido ajustado empíricamente seleccionando  $\nu = 0,01$  con el objetivo de obtener el mejor funcionamiento del algoritmo en estado estacionario.

Cabe citar que no se ha incluido en los experimentos los resultados del algoritmo de *Hasting* [3], ni del esquema presentado en [10]. El primero no fue diseñado considerando la posibilidad de diferentes pasos de adaptación en cada nodo de la red adaptativa, lo que da lugar a un funcionamiento incorrecto en este escenario. El segundo, al considerar combinadores fijos en la etapa de convergencia, da lugar a una degradación de la misma y no puede considerarse completamente como un algoritmo con pesos adaptativos.

La figura de mérito empleada para comparar el funcionamiento de los algoritmos es la desviación cuadrática media de la red

$$\text{MSD}(n) = \frac{1}{N} \sum_{k=1}^N \text{MSD}_k(n) = \frac{1}{N} \sum_{k=1}^N \mathbb{E}\{\|\mathbf{w}_k(n) - \mathbf{w}_0(n)\|^2\},$$

estimada mediante un promedio sobre 500 realizaciones independientes de los algoritmos.

Las Figuras 2 y 3 muestran el comportamiento de los algoritmos ADN-LS (tanto en su versión afín como convexa), el algoritmo ARV y el de un algoritmo que no realiza difusión alguna, en un escenario estacionario utilizando, respectivamente, señal de entrada blanca y coloreada. En este escenario, con el objetivo de evaluar la capacidad de reconvergencia de los esquemas, se ha introducido un cambio abrupto en la planta a identificar  $\mathbf{w}_0(n)$  en la mitad del experimento.

Tal y como se aprecia en ambas figuras, todos los algoritmos de difusión mejoran los resultados con respecto al caso de no cooperación (donde cada nodo funciona como un filtro adaptativo independiente sin realizar difusión alguna). Sin embargo, el algoritmo ADN-LS (en sus dos versiones) presenta una convergencia más rápida que el algoritmo ARV. Esto es debido a que los pesos de combinación calculados mediante el algoritmo ARV otorgan una gran importancia a la estimación de los nodos lentos, incluso en convergencia, degradando la estimación de los nodos rápidos. Centrándonos en la comparación de las dos versiones de nuestro esquema,

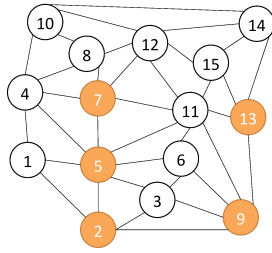


Fig. 1. Topología de red utilizada en los experimentos.

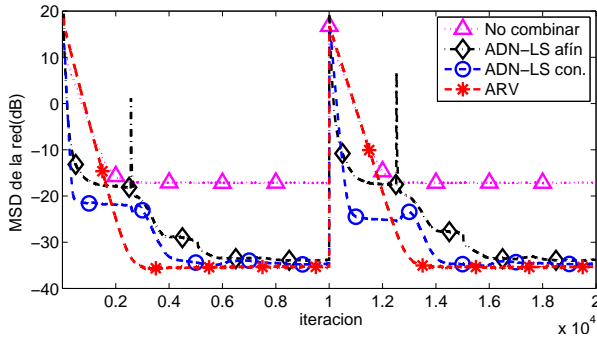


Fig. 2. MSD( $n$ ) de los algoritmos usando señal de entrada blanca.

se puede comprobar cómo la versión convexa resulta en un comportamiento más estable, principalmente fruto de posibles inestabilidades en la inversión de la matriz  $\mathbf{P}_k(n)$  en (14) para el caso afín. Por lo tanto, parece sensato señalar que, en caso de utilizar el algoritmo ADN-LS con ventana rectangular, es ventajoso imponer la condición de convexidad, al igual que hacen otros algoritmos con combinadores adaptativos.

Con todo, merece la pena considerar que, aunque la mayoría de los algoritmos sugieren la utilización de combinaciones convexas, éstos han sido testados en situaciones donde los regresores son totalmente independientes entre los diferentes nodos. Sin embargo, resultados del campo de la combinación adaptativa de filtros adaptativos sugieren que, en caso de existir correlaciones espaciales entre las señales de los nodos, una combinación afín que permitiese pesos negativos pudiese proporcionar ganancias adicionales [11].

#### IV. CONCLUSIONES Y TRABAJO FUTURO

El diseño de la etapa de combinación en un algoritmo de difusión adaptativa es clave para un buen funcionamiento del mismo, especialmente si los nodos de la red presentan diferentes capacidades de adaptación y/o sensores con distintas SNRs. En este artículo hemos repasado el esquema ADN-LS, el cual, preservando una estimación puramente local y utilizando un algoritmo basado en mínimos cuadrados para adaptar los pesos, obtiene un buen comportamiento tanto en estado estacionario como en convergencia. Se han revisado dos variantes del algoritmo: convexa y afín, evidenciando empíricamente la mayor estabilidad numérica de la variante convexa, al menos en escenarios estacionarios con señales de entrada incorrelacionadas.

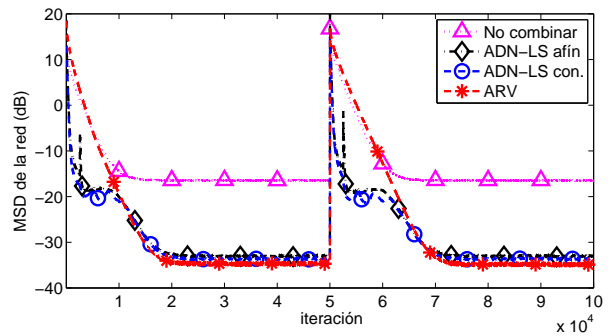


Fig. 3. MSD( $n$ ) de los algoritmos usando señal de entrada coloreada.

Sin embargo, considerando los resultados en otras líneas de investigación similares, merece la pena considerar los casos en los que exista cierta correlación entre los regresores, donde una combinación afín podría ofrecer ventajas con respecto a una convexa. De igual forma, como trabajo futuro también nos centraremos en el diseño de otras estrategias de adaptación de los pesos, así como en el estudio teórico del esquema de difusión utilizando el método de conservación de la energía.

#### AGRADECIMIENTOS

El trabajo de Azpicueta-Ruiz, Fernandez-Bes y Arenas-García ha sido parcialmente soportado por los proyectos del MICINN TEC2011-22480 y PRI-PIBIN-2011-1266. El trabajo de Fernandez-Bes también ha sido soportado por el programa FPU del MECED. El trabajo de Silva ha sido parcialmente apoyado por CNPq bajo la beca 302423/2011-7 y por FAPESP bajo la beca 2012/24835-1.

#### REFERENCES

- [1] A. H. Sayed, S.-Y. Tu, J. Chen, X. Zhao, and Z. J. Towfic, "Diffusion strategies for adaptation and learning over networks," *IEEE Signal Processing Mag.*, vol. 30, no. 3, pp. 155–171, 2013.
- [2] A. H. Sayed, "Diffusion adaptation over networks," *Academic Press Library in Signal Processing*, vol. 3, pp. 323–454, 2013.
- [3] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over LMS adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5107–5124, 2012.
- [4] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least mean-squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, 2010.
- [5] S.-Y. Tu, A. H. Sayed, "Optimal combination rules for adaptation and learning over networks," in *Proc. of 4th IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Process. (CAMSAP)*, 2011, pp. 317–320.
- [6] J. Fernandez-Bes, L. A. Azpicueta-Ruiz, M. T. M. Silva, and J. Arenas-García, "A novel scheme for diffusion networks with least-squares adaptive combiners," in *2012 IEEE Intl. Workshop Mach. Learning for Signal Process.*, 2012, pp. 1–6.
- [7] —, "Improved least-squares-based combiners for diffusion networks," in *2013 Intl. Symp. Wireless Commun. Sys. (ISWCS 2013)*, 2013.
- [8] A. H. Sayed, *Adaptive filters*, Wiley, 2008.
- [9] L. A. Azpicueta-Ruiz, M. Zeller, A. R. Figueiras-Vidal, and J. Arenas-García, "Least-squares adaptation of affine combinations of multiple adaptive filters," in *Proc. of IEEE ISCAS*, 2010, pp. 2976–2979.
- [10] C.-K. Yu, A. H. Sayed, "A strategy for adjusting combination weights over adaptive networks," in *Proc. of IEEE Intl. Conf. Acoustics, Speech, and Signal Process. (ICASSP)*, 2013, pp. 4579–4583.
- [11] R. Candido, M. T. M. Silva, V. H. Nascimento, "Transient and Steady-State Analysis of the Affine Combination of Two Adaptive Filters," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4064–4078, 2010.