

Supervised Learning Applied to the Decoding of SCMA Codewords

S. Vidal-Beltrán, F. Martínez-Piñón, J. López-Bonilla

Abstract—This work puts together two technologies that are in the interest of the scientific community, on the one hand, access methods for fifth generation systems of mobile communications, in this case Sparse Code Multiple Access (SCMA), and on the other hand supervised learning based on neural networks. SCMA is one of the proposed access techniques for fifth generation mobile communication systems. Until now, the detection algorithm in the receiver is based on Message Passing Algorithm (MPA) or minimum Euclidean distance. In this work, a new approach is proposed, which is based on supervised learning using neural networks to decode SCMA codewords. The simulation results show that the receiver based on neural networks learns quickly and obtains 100% accuracy in predictions on channels with high noise. In addition to being simpler in its implementation than its predecessors (MPA and minimum Euclidean distance).

Index Terms—5G, Adam, Neural networks, SCMA, SGD, Supervised Learning.

I. INTRODUCCIÓN

5G mejorará el rendimiento de las redes celulares móviles mediante un avance sustancial en la velocidad, capacidad, latencia y conectividad mientras usa eficientemente los recursos de radio y de potencia. Se espera que esta no sea una evolución de la actual tecnología 4G, sino que se enfoque en afrontar nuevos retos tecnológicos tales como la integración y la creación de nuevos modelos de negocios. 5G deberá centrarse en alcanzar tasas de transmisión del orden de los Gbps, acceso confiable con gran ancho de banda y comunicaciones masivas, una muy baja latencia (<5ms) y una alta confiabilidad (>99.999 %), además de un uso eficiente del espectro [1]-[3].

SCMA (Sparse Code Multiple Access) es una técnica de acceso que usa libros de código (codebooks) de baja densidad para disminuir la complejidad de la detección de los símbolos.

Con este método a cada usuario se le asigna un libro de código y los datos dentro de él son usados para mapear el flujo de bits directamente a un vector de baja densidad (sparse vector) llamado palabra de código (codeword) [4], [5].

S. Vidal-Beltrán, Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica – Zacatenco, Ciudad de México, México (e-mail: svidalb@ipn.mx).

F. Martínez-Piñón, Instituto Politécnico Nacional, Centro de Investigación e Innovación Tecnológica- Azcapotzalco, Ciudad de México, México.

J. L. López-Bonilla, Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica – Zacatenco, Ciudad de México, México.

Libros de código multidimensionales son usados para complementar la modulación QAM y el ensanchamiento depende de este factor. Cada uno contiene el valor cero en las mismas dos dimensiones de todos sus elementos y la posición de los ceros está determinada aleatoriamente para prevenir colisiones entre los usuarios. Actualmente, tomando como referencia [6], [7], [10], [11] para detectar y decodificar las palabras de código SCMA se usan dos técnicas principalmente, por un lado, MPA el cual se basa en un complejo algoritmo de probabilidad conjunta y por otro lado mínima distancia Euclidiana, que, aunque es un algoritmo más sencillo de implementar requiere de más tiempo de máquina para hacer los cálculos.

En los años recientes, las tecnologías basadas en redes neuronales y aprendizaje profundo han traído mejoras en distintas áreas de la tecnología como en el campo de las comunicaciones donde trabajos recientes han demostrado su aplicabilidad en este campo [12].

En este trabajo, se propone un detector de palabras de código SCMA, el cual se basa en redes neuronales. El detector propuesto muestra un buen desempeño aún en presencia de bajos niveles de relación señal a ruido (SNR).

En la sección II de este trabajo, se presenta la teoría fundamental para la construcción de libros de código usando el método de rotación de fase. En la sección III se exhibe un breve resumen de las alternativas para la detección y decodificación de palabras de código SCMA, incluida la descripción del receptor basado en redes neuronales. La sección IV está dedicada a establecer los parámetros de simulación y posteriormente en las secciones V y VI se procede a presentar resultados de la simulación, así como las conclusiones, respectivamente.

II. CONSTRUCCIÓN DE LIBROS DE CÓDIGO SCMA

Para el diseño y construcción de los libros de código, se inicia seleccionado un conjunto de vectores Gray [6], [7], el cual representa cada punto de la Constelación Madre (M_C), entonces se define un subconjunto S_1 de enteros Gaussianos con la siguiente estructura (1):

$$S_1 = \{A_m(1 + i) \mid A_m = 2m - 1 - M, m = 1, \dots, M\} \quad (1)$$

A continuación, se asignan valores a cada punto S_{1m} de S_1 . Considerando que $M = 4$ (M es el número de palabras de código), el mapeo de Gray toma la siguiente forma de (2):

$$\begin{aligned} S_{11} &= -3(1 + i) \\ S_{12} &= -(1 + i) \\ S_{13} &= (1 + i) \\ S_{14} &= 3(1 + i) \end{aligned} \quad (2)$$

S_1 , representa la primera dimensión, en este trabajo los libros de código se construyen con 2 dimensiones ($N = 2$). Las siguientes dimensiones se obtienen considerando que $S_l = S_1 U_l$, donde $U_l = I_{M \times M} e^{i\theta_{l-1}}$, $\theta_{l-1} = \frac{(l-1)\pi}{MN}$, $l = 1, \dots, N$, por lo que M_C está dada por (3):

$$M_C = (S_1, \dots, S_N)^T = \begin{bmatrix} S_{11} & S_{12} & \dots & S_{1M} \\ S_{21} & S_{22} & \dots & S_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N1} & S_{N2} & \dots & S_{NM} \end{bmatrix} \quad (3)$$

Una de las principales propiedades de SCMA es que los libros y palabras de códigos son dispersos (sparse), de esta forma se logra conectividad masiva de usuarios. En SCMA la longitud de las palabras de código se decide con base en el número de subportadoras disponibles K (Factor de esparcimiento). Las palabras de código se consideran dispersas dado que el número de elementos no-zero en cada palabra de código N es menor que K . Cuando los valores de K & N son determinados es posible determinar la cantidad de libros de código posibles (J). La cantidad de libros de código está dada por el coeficiente binomial dado en (4):

$$J = \binom{K}{N} = \frac{K}{N} \binom{K-1}{N-1} \quad (4)$$

Considerando $K = 4$, $N = 2$, entonces $J = 6$. En los sistemas OFDMA, cada subportadora transmite solo a un usuario, lo cual no permite sobrecarga. SCMA transmite más de un usuario en una subportadora. Considerando, $K = 4$, $N = 2$, se generan $J = 6$ libros de código.

La cantidad de usuarios conectados a una subportadora (d_f) está dado por (5):

$$d_f = \binom{K-1}{N-1} = \frac{JN}{K} \quad (5)$$

Por otro lado, el factor de sobrecarga está dado por $\lambda = \frac{J}{K} = \frac{d_f}{N}$, considerando 6 usuarios ($J = 6$), y cada usuario usa un libro de código diferente el factor de sobrecarga (λ) es 1.5 y d_f es igual a 3. Por lo tanto, una subportadora SCMA transmite más usuarios que una portadora OFDMA.

En este trabajo, los parámetros principales para la construcción de los libros (codebooks) y palabras de código (codewords) SCMA:

- Cantidad de palabras: 4 ($M = 4$: 00,01,11,10)
- Número de usuarios: 6 ($J = 6$)
- Cantidad de Subportadoras disponibles: 4 ($K = 4$).
- Factor de sobrecarga: 1.5 ($\lambda = 1.5$)
- Usuarios por subportadora: 3 ($d_f = 3$)
- Dimensiones de código: 2 ($N = 2$)

Los libros de código para diferentes usuarios se obtienen a partir de la constelación madre M_C . Los ángulos de rotación para los distintos usuarios se obtienen a partir de d_f & e_u los cuales representan el número de usuarios que se transportan

por un recurso de radio (en este caso 3), por lo que el ángulo de rotación φ_u para los distintos usuarios se define por (6):

$$\varphi_u = (u-1) \frac{2\pi}{Md_f} + e_u \frac{2\pi}{M}, \forall u = 1, \dots, d_f \quad (6)$$

El valor para φ_u se asigna a las posiciones no-cero del factor de gráfica [8]; considerando $M = 4$ & $N = 2$, el factor de gráfica está dado por (7):

$$F = \begin{bmatrix} 0 & \varphi_1 & \varphi_2 & 0 & \varphi_3 & 0 \\ \varphi_3 & 0 & \varphi_1 & 0 & 0 & \varphi_2 \\ 0 & \varphi_2 & 0 & \varphi_3 & 0 & \varphi_1 \\ \varphi_2 & 0 & 0 & \varphi_2 & \varphi_3 & 0 \end{bmatrix} \quad (7)$$

El factor de gráfica define cuáles portadoras (filas) son empleadas por cada usuario (columna). De (7) se observa que los datos del usuario 1 se transportan por las subportadoras 2 y 4. Haciendo que f_j sea la j -ésima columna de F sin elementos cero. Se define un operador para el j -ésimo usuario tal como $\Delta_j = \text{diag}(f_j) = f_j, |j = 1, 2, \dots, 6$. Los libros de código para los diferentes usuarios se obtienen aplicando (8):

$$X_j = V_j \Delta_j M_C, |j = 1, 2, \dots, J \quad (8)$$

donde V_j es una matriz de dispersión de dimensión 2×4 donde dos filas son cero. Para este trabajo se generan 6 libros de código ($J = 6$), donde los libros de código X_j tienen la forma de una matriz de 4×4 ; las columnas representan las palabras de código mientras que las filas a los recursos (subportadoras). Las palabras de código modulan a las subportadoras (SC) las cuales son sumadas con las demás y se envían a la antena para producir una señal que será enviada al aire. La Fig. 1, muestra el diagrama general de los libros de código y la selección de las palabras de código:

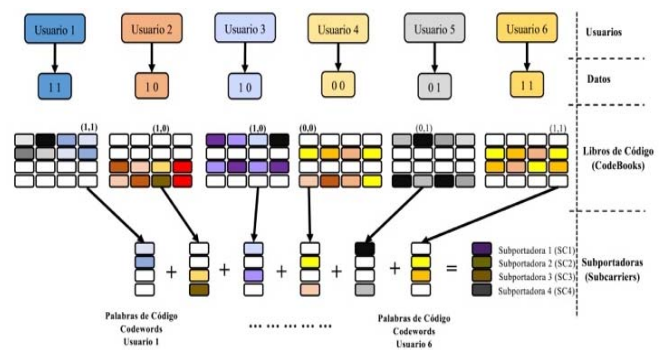
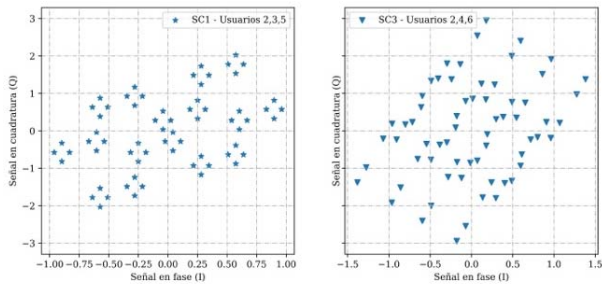


Fig. 1. Diagrama de la selección de libros de código.

La información de cada usuario viaja en dos subportadoras de radio, razón por la cual la complejidad en la recepción se incrementa ya que la información se debe buscar en dos lugares simultáneamente. Por ejemplo, para el usuario 2, y con base en el factor de gráfica dado por (7), se tiene que los datos se envían por la SC1 y la SC3. La SC1 transporta datos de los usuarios 2, 3 y 5; la SC 3 transporta datos de los usuarios 2, 4,

6. Considerando que la SC1 transporta datos de 3 usuarios cada par de bits del usuario 2 puede ser representado hasta por 16 posibles combinaciones, ya que el valor de un par de bits del usuario 2 debe ser sumado con los 4 posibles valores del usuario 3 y los 4 posibles valores del usuario 5, resultando que 4 pares de bits pueden ser representados hasta por 64 combinaciones. La Fig. 2, muestra la constelación en el plano IQ para la SC1, que transporta datos de los usuarios 2, 3 y 5; asimismo se presenta la constelación para la SC3.



(a) Constelación para SC1 (b) Constelación para SC3
Fig. 2. Constelación de referencia en el plano IQ para la SC1 y SC3.

III. RECEPTOR SCMA

La detección y decodificación de las palabras de código SCMA es afectada por el diseño del código, el rendimiento del canal y el ruido. MPA y mínima distancia Euclidiana son las dos técnicas usadas principalmente, las cuales son descritas brevemente a continuación y posteriormente se describe el receptor propuesto en este trabajo.

A. Message Passing Algorithm (MPA)

MPA usa el algoritmo sum-product para calcular la probabilidad marginal de diferentes palabras de código. Se emplean métodos de probabilidad Bayesiana para calcular la probabilidad de la palabra de código dada una probabilidad inicial dentro del libro de código [10].

Considerando 6 usuarios, 4 palabras de código en cada libro de código y asumiendo que todos los usuarios transmiten dentro de la misma trama de datos. Dado que no se tiene conocimiento previo de los datos a transmitir, se asume que cada palabra de código seleccionada tiene la misma probabilidad. Es posible establecer la probabilidad inicial de un usuario escogiendo una palabra de código igual a 1 dividido por la cardinalidad M del conjunto de palabras de código en el libro de código de cada usuario [10]. Para este caso, todas las palabras de código tienen una probabilidad inicial $1/4$. Dado que MPA es un algoritmo que trabaja con probabilidades conjuntas, resulta ser difícil de implementar y consume tiempo de máquina para hacer sus cálculos.

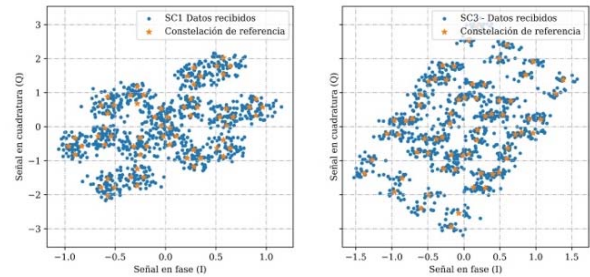
B. Mínima Distancia Euclidiana

La detección coherente, el transmisor y el receptor acuerdan la misma constelación de referencia para modular y demodular la información del usuario [11]. El primer paso en la detección IQ es calcular la distancia Euclidiana entre dos vectores dados (el arreglo de referencia y los símbolos recibidos con ruido). Cada símbolo en el vector recibido debe ser comparado con cada símbolo del arreglo de referencia, y entonces calcular la distancia Euclidiana mínima.

Considerando que $x = (x_1, x_2, \dots, x_p)$, $y = (y_1, y_2, \dots, y_p)$, sean dos vectores en el plano-p. La distancia Euclidiana está dada por (10):

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_p - y_p)^2} \quad (10)$$

La Fig. 3 muestra la constelación de referencia (puntos rojos) y los datos aleatorios (puntos azules afectados por una SNR = -12dB).



(a) Constelación para SC1 (b) Constelación para SC3
Fig. 3. Constelación en el plano IQ para las SC1 y SC3 (SNR = -12 dB).

Para cada dato recibido (puntos azules), se calcula la distancia mínima a cada punto de constelación (puntos rojos) de referencia, con lo cual se decodifican los datos. Este proceso, aunque es más sencillo de implementar que MPA, resulta en una gran cantidad de tiempo de máquina, dado que cada dato recibido es comparado con 64 valores posibles de la constelación de referencia.

C. Receptor basado en Redes Neuronales

A diferencia de sus predecesores (MPA y mínima distancia Euclidiana), en el receptor, el problema de detección y decodificación se analiza como un problema de clasificación a través del uso de aprendizaje supervisado mediante redes neuronales [12], [13] para decodificar los datos aún en presencia de ruido [4]. La Fig. 4, muestra la arquitectura general del receptor propuesto.

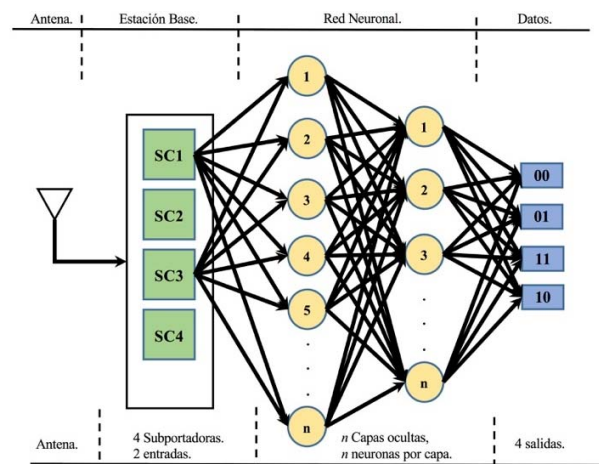


Fig. 4. Diagrama general de la red neuronal.

Por ejemplo, para decodificar los datos del usuario 1, la estación base conoce el libro de código correspondiente y

busca la información en las subportadoras 1 y 3 las cuales son la entrada a la red neuronal, la cual está compuesta por n número de capas densas, cada una con n número de neuronas, 4 salidas representan los datos a decodificar (00, 01, 11, 10).

IV. PARÁMETROS DE LA CONFIGURACIÓN DE LA SIMULACIÓN

Para la detección y decodificación de las palabras de código SCMA usando redes neuronales se utilizaron los siguientes parámetros de configuración.

A. Libros y Palabras de Código SCMA

Se usa el método de rotación de fase de la constelación madre (M_c) para la transmisión de datos. Según se muestra en la sección II, se contemplan 6 usuarios ($J = 6$), usando 4 recursos de radio ($K = 4$), 4 palabras de código ($M = 4$: 00, 01, 11, 10), hasta 3 usuarios se transportan por cada recurso de radio ($d_f = 3$) con dos dimensiones de código ($N = 2$). Para este trabajo, los recursos de radio (subportadoras), se asignan estáticamente acorde al factor de gráfica dado en (7). Una vez que las palabras de código son seleccionadas, se envían a la antena para la transmisión inalámbrica, ruido aleatorio se añade a la señal, la magnitud del ruido depende de la SNR [4]. Las simulaciones de este trabajo consideraron señales afectadas por un canal AWGN (Additive white Gaussian noise) con relación señal a ruido desde -18 dB hasta 18 dB.

B. Configuración de la Red Neuronal

Para decodificar los datos de cada usuario, se construye una red neuronal con 4 entradas y tres capas densas ocultas, cada una con 64, 32 y 16 nodos; las cuales producen 4 salidas. La tarea principal del aprendizaje supervisado es obtener la mejor estimación de la función objetivo (f), que mapea los datos de entrada (X), en variables de salida (Y) [14], [15]. Para llegar a la mejor estimación se recurre a dos algoritmos de optimización; se utilizó Stochastic Gradient Descent (SGD) [16] y Adaptive Moment Estimation (Adam) [16].

C. Entrenamiento de la Red Neuronal

El algoritmo de aprendizaje supervisado depende de un conjunto de datos de entrenamiento, el cual funciona como un supervisor del proceso de aprendizaje. Dado que se conoce la respuesta correcta, el algoritmo iterativamente hace predicciones en los datos de entrenamiento y en caso de fallas en la clasificación, el supervisor hace correcciones. El aprendizaje se detiene cuando el algoritmo logra un nivel aceptable de desempeño en las predicciones. El conjunto de datos de entrenamiento se generó usando (11):

$$P_t = M^{d_f} \times \text{Rango de dB} \quad (11)$$

donde P_t , representa el conjunto inicial de datos de entrenamiento, considerando $M = 4$, $d_f = 3$ y un rango de -18 dB a 18 dB, resultan 2034 valores que son utilizados como el conjunto de datos de entrenamiento de la red neuronal. Para este trabajo se generaron datos de entrenamiento de 5 veces P_t .

D. Predicciones

Una vez que la red neuronal ha sido entrenada, para probar el funcionamiento de la red neuronal se generaron 1024 bits aleatorios, afectados por diferentes SNR (desde -18 dB hasta 18 dB). Por ejemplo, en la Fig. 5, los puntos marcados en color rojo representan la constelación IQ de referencia, mientras que los puntos en azul son los datos generados aleatoriamente cuando éstos son afectados por una relación señal a ruido (SNR) de -9 dB, se presentan las constelaciones para la SC1 y la SC3.

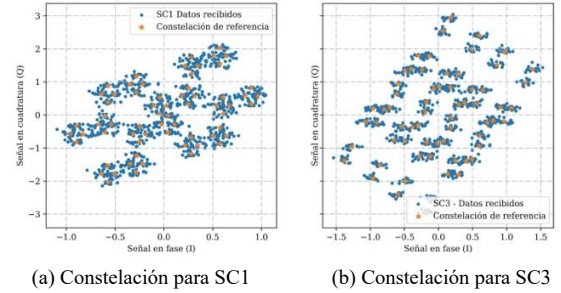


Fig. 5. Constelación en el plano IQ para las SC1 y SC3 (SNR = -9 dB).

E. Parámetros de Evaluación

Para verificar el desempeño de la red neuronal haciendo predicciones en datos nuevos, se evalúa lo siguiente:

i. Precisión del algoritmo de optimización.

Las redes neuronales y aprendizaje de máquina usan algoritmos de optimización que iteran sobre los datos introducidos para encontrar los mejores resultados. Valores cercanos a 1 indican un alto grado de precisión mientras que valores cercanos a cero indican un bajo desempeño del optimizador.

ii. Función de pérdida.

El grado de aprendizaje de una red neuronal, está dado por la función de pérdida, es un método mediante el cual se evalúa qué tan bien un algoritmo modela los datos dados. Si las predicciones se alejan mucho de los resultados, la función de pérdida tiene valores grandes, un valor cercano a cero indica un mejor desempeño de las predicciones.

iii. Cantidad de errores en las predicciones.

Para comprobar el desempeño del receptor se presentan los resultados la cantidad de errores en las predicciones cuando la SNR cambia entre -18 dB y 18 dB.

Un parámetro importante en el desempeño de los algoritmos de optimización es la cantidad de *epochs* (iteraciones) consumidos para que un algoritmo converja. Un *epoch* se define como una iteración completa del algoritmo de optimización sobre el total del conjunto de datos dados. Si un algoritmo de optimización logra valores altos de precisión en un número pequeño de *epochs*, indica que tiene una más rápida tasa de aprendizaje que aquellos que requieren un mayor número de *epochs*. En la siguiente sección se muestran los resultados de este trabajo.

V. RESULTADOS

Para la simulación, se emplearon 10 iteraciones, tamaño del batch de 10 y un rango de validación de datos de 30 % para ambos optimizadores; para el optimizador SGD se usó una tasa de aprendizaje de 0.01 ($\eta = 0.01$). La función de activación de las capas ocultas de la red neuronal fue Relu, a excepción de la última capa donde se utilizó la función de activación SOFTMAX. La Fig. 6, muestra la precisión de los algoritmos de optimización, se observa que Adam alcanza una precisión de 100 %, mientras que SGD en el mejor de los casos alcanza el 92.5 %. A la función de pérdida se le conoce también como función objetivo (objective function), como se muestra en la Fig. 7, Adam llega a cero en el valor de la función de pérdida, mientras que SGD presenta pérdidas cercanas a 0.2.

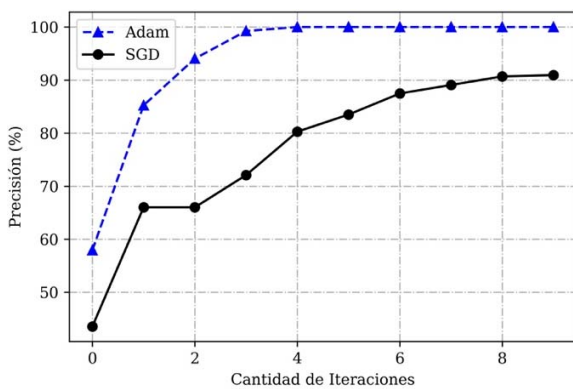


Fig. 6. Precisión del algoritmo de optimización.

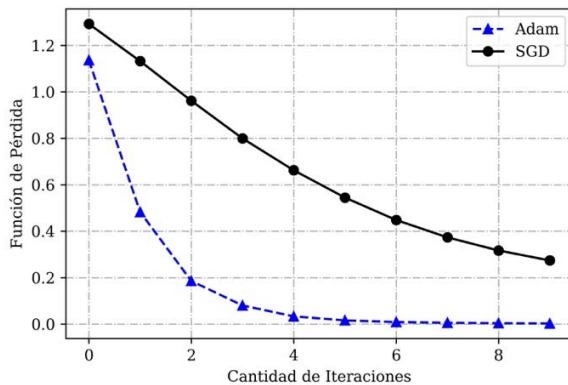


Fig. 7. Función de pérdida de los algoritmos de optimización.

Una vez evaluado el optimizador, se analizó la cantidad de errores existentes al momento de hacer las predicciones sobre los nuevos datos; como se mencionó anteriormente, se generaron 1024 nuevos datos afectados por SNR entre -18 dB y 18 dB, la Fig. 8, muestra el comportamiento de ambos algoritmos de optimización.

De la Fig. 8, se observa que Adam a partir de -15 dB no presenta errores en la decodificación de datos, mientras que SGD en el mejor de los casos presenta aproximadamente 140 errores por cada 1024 datos nuevos. En la Fig. 9, se muestra la precisión en las predicciones tanto de SGD como de Adam.

De la Fig. 9, se observa que para SNR de -18 dB la red neuronal usando Adam presenta un 99 % de precisión en las

predicciones sobre datos nuevos y para -15 dB la precisión es de 100 %.

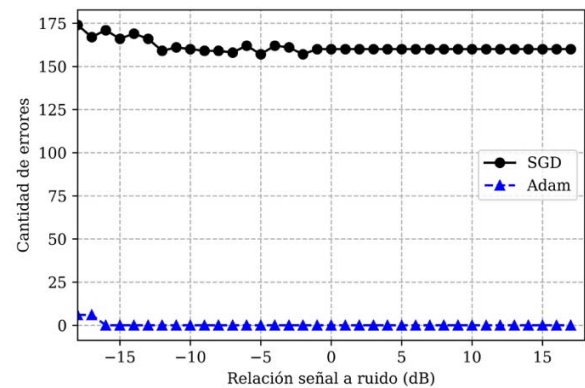


Fig. 8. Cantidad de errores en función de la SNR.

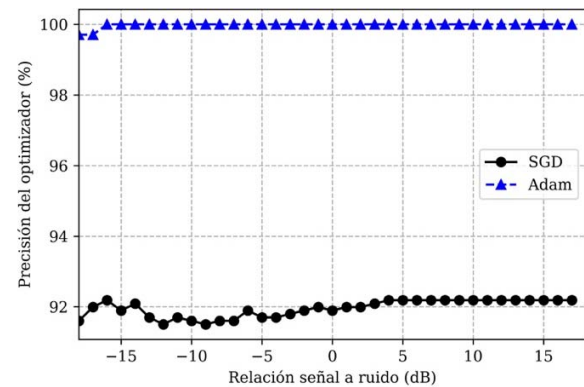


Fig. 9. Precisión de SGD y Adam.

VI. CONCLUSIÓN

En este trabajo se propone y evalúa un receptor para palabras de código SCMA basado en aprendizaje supervisado usando redes neuronales. Las principales contribuciones se resumen en:

- i. Se demuestra que el aprendizaje supervisado puede ser usado en la detección y decodificación de palabras de código SCMA.
- ii. El detector propuesto basado en redes neuronales utiliza dos algoritmos de optimización para hacer las predicciones; se presentan los resultados de precisión y pérdida para SGD y Adam.
- iii. La red neuronal propuesta, solo se requiere de 3 capas ocultas (64, 32 y 16 neuronas respectivamente), para que Adam logre una precisión de 100 % con solo 10 *epochs* (iteraciones) en los datos de entrenamiento.
- iv. La red neuronal usando Adam, como algoritmo de optimización, a partir de -15 dB de SNR, presenta 0 errores en las predicciones con los nuevos datos.

El receptor propuesto requiere pocos datos de entrenamiento (10,170 datos) y es rápido en su aprendizaje dado que solo 10 *epoch* son necesarios para lograr un aprendizaje de 100 % y una función de pérdida de 0. Con estos resultados se propone que el aprendizaje supervisado con redes neuronales, con Adam como algoritmo de optimización,

es una buena opción para un receptor de palabras de código SCMA.

AGRADECIMIENTOS

Los autores agradecen al Instituto Politécnico Nacional, el soporte recibido para llevar a cabo este trabajo. Este trabajo se desarrolló bajo el proyecto 20190057, financiado por el Instituto Politécnico Nacional. Asimismo, nuestro agradecimiento a los editores y revisores que nos ayudaron a mejorar la calidad de la presentación y estructura de este trabajo.

REFERENCIAS

- [1] L. Dai, B. Wang, Y. Yuan, S. Han, and Z. Wang, "Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends," *IEEE Communications Magazine*, Vol. 53, no. 9, pp. 74–81, Sept. 2015.
- [2] J. Bao, Z. Ma, Z. Ding, G. K. Karagiannidis and Z. Zhu, "On the design of multiuser codebooks for uplink SCMA systems", *IEEE Communications Letters*, Vol. 20, no. 10, pp. 1920–1923, Oct. 2016.
- [3] E. A. Degollado, S. Vidal, and J. L. López, "Simplified Equation for Received Power Calculation of the LTE Signal in 2.13 GHz," *Applied Mathematics & Information Sciences*, Vol. 12, no. 6, pp. 1177–1182, 2018.
- [4] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005.
- [5] M. Taherzadeh, H. Nikopur, A. Bayasteh and H. Baligh, "SCMA codebook design," in *IEEE Vehicular Technology Conference (VTC Fall)*. Vancouver, BC, Canada, 2014.
- [6] J. L. López, S. Vidal, I. Santiago, and F. Martínez, "Construction of SCMA CodeBooks using the phase rotation method," in *IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC 2018)*. Ixtapa, México, 2018.
- [7] L. Yu, X. Lei, P. Fan, and D. Chen, "An optimized design of SCMA codebook based on star-QAM signaling constellations," in *2015 International Conference on Wireless Communications Signal Processing (WCSP)*, Nanjing, China, Oct 2015.
- [8] R. C. Minnick, B. Elspas, R. A. Short, "Symmetric Latin Squares". *IEEE Transactions on Electronic Computers*. Vol. EC-12, no: 2, pp. 130-131, Apr. 1963.
- [9] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
- [10] A. Bayesteh, H. Nikopour, M. Taherzadeh, H. Baligh, and J. Ma. "Low complexity techniques for SCMA detection". In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, San Diego, CA, USA, Dec 2015.
- [11] S. Vidal-Beltrán, J. L. López-Bonilla, F. Martínez-Piñón, "The Euclidean distance -based detection method applied on SCMA codewords", *International Journal of Latest Research in Science and Technology*, vol. 7, no. 3, pp. 14-18, May-June 2018
- [12] O. Simeone, "A Very Brief Introduction to Machine Learning with Applications to Communication Systems", *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 648-664, Dec. 2018.
- [13] A. S. Brandao, D. C. Jorge, "Artificial Neural Networks Applied to Image Steganography", *IEEE Latin America Transactions*, vol. 14, no. 3, pp. 1361-1366, Mar. 2016.
- [14] J. Wu, I. Bisio, C. Gniady, E. Hossain, M. Valla, H. Li, "Context-aware networking and communications: Part 1", *IEEE Communications Magazine*, vol. 52, no. 6, June 2014.
- [15] R. Atat, L. Liu, J. Wu, G. Li, C. Ye, Y. Yang, "Big Data Meet Cyber-Physical Systems: A Panoramic Survey", *IEEE Access*, vol. 6, pp. 73603-73636, November 2018.
- [16] D. P. Kingma, J. Lei-Ba, "Adam: A method for stochastic optimization," in the 2nd International Conference on Learning Representations, San Diego, USA, 2015.



Sergio Vidal-Beltrán. Profesor e Investigador Científico en la Maestría en Ciencias en Ingeniería de Telecomunicaciones del Instituto Politécnico Nacional, México. Especialista en Interconectividad de redes y en sistemas de comunicaciones móviles.



Fernando Martínez-Piñón. Profesor e Investigador científico del Centro de Investigación e Innovación Tecnológica (CIITEC) del Instituto Politécnico Nacional, México. Especialista en Fotónica (Fibras Ópticas para Telecomunicaciones y Sensores), Dispositivos láser, Dispositivos electro-ópticos y Radiocomunicaciones



José Luis López-Bonilla. Profesor e Investigador científico de la Escuela Superior de Ingeniería Mecánica y Eléctrica (ESIME) del Instituto Politécnico Nacional, México. Especialista en Métodos Matemáticos Aplicados a la Ingeniería y Física Teórica.