

PROYECTO LMS

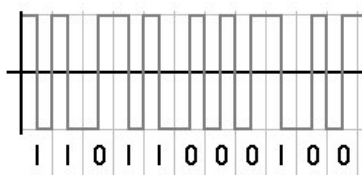
El primer proyecto consiste en la investigación y desarrollo de un problema con LMS. Deberán determinar el algoritmo óptimo (a partir de los algoritmos vistos en clase) y sus (meta)parámetros óptimos; pueden investigar otras variantes de LMS.

Es importante que en el trabajo justifiquen todas las decisiones que tomen: enfoque al problema, criterios de calidad (curva de aprendizaje, tracking de coeficientes, desajuste, robustez, velocidad de convergencia, modos, noise amplification, etc.), algoritmos y parámetros elegidos, y comparaciones entre alternativas. Para las simulaciones utilicen, según corresponda, el método de Monte Carlo o el set de datos propuesto. Deben evaluar la complejidad de las simulaciones y justificar toda simplificación que hagan.

Los proyectos son grupales (en equipos de dos), y deberán presentarse en la fecha de presentación anunciada. Una semana después de la presentación deberán entregar un informe en formato IEEE que debe incluir una sección de introducción (con el contexto del trabajo y referencias bibliográficas relevantes), una sección de desarrollo (con actividades realizadas y resultados obtenidos), una sección de conclusiones (con el análisis de los resultados y trabajo a futuro) y una sección de bibliografía.

Temas sugeridos:

- *Cancelación de ecos en un celular con manos libres.* Para simular las respuestas al impulso de las habitaciones pueden utilizar `gpuRIR` (<https://github.com/DavidDiazGuerra/gpuRIR>) o `pyroomacoustics` (<https://pypi.org/project/pyroomacoustics/>). Recomendaciones: simulen sólo un lado del circuito de comunicación; consideren habitaciones típicas; simulen usando una base de datos con conversaciones reales; asuman una distancia entre parlante y micrófono de 10 cm y una velocidad de desplazamiento del celular máxima de 0.1 m/s.
- *Detección de anomalías cardíacas con un predictor LMS.* Para las simulaciones pueden usar la *MIT-BIH Arrhythmia Database* (<https://physionet.org/content/mitdb/1.0.0/>).
- *Ecualización de señales en un enlace digital de comunicaciones.* Los datos consisten en una secuencia de datos pseudoaleatoria codificada con Manchester (https://es.wikipedia.org/wiki/Codificación_Manchester) a 250 bps y amplitud ± 1.0 :



La señal recibida es muestreada a 4 kHz. El canal puede modelarse con el siguiente fragmento de código en Python (si tienen problemas con el tracking pueden disminuir `alpha` y/o `noise`, justificando las decisiones tomadas):

```
alpha = 0.01
noise = 0.1
def tx_channel(x):
    y = np.array([])
    for i in range(0, len(x)):
        pp = np.exp(tx_channel.freq * 1j) * [0.9, 0.95]
        p = [pp[0], np.conj(pp[0]), pp[1], np.conj(pp[1])]
        b, a = signal.zpk2tf([], p, 1)
        y_, tx_channel.zi = signal.lfilter(b, a, np.array([x[i]]),
                                           zi=tx_channel.zi)
        tx_channel.freq += alpha * np.random.uniform(-1, 1, 2)
        y = np.append(y, y_[0])
    return y + noise * np.random.randn(len(x))
tx_channel.freq = np.random.uniform(0., np.pi, 2)
tx_channel.zi = np.zeros(4)
```