



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERIA MECÁNICA

IDENTIFICACIÓN DE IMPACTOS EN UNA ESTRUCTURA
TRIDIMENSIONAL UTILIZANDO AUTOENCODERS Y UNA
APROXIMACIÓN LINEAL DEL PRINCIPIO DE MÁXIMA ENTROPÍA

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERA CIVIL MECÁNICA

CONY DE LOS ÁNGELES ESPINOZA QUITRAL

PROFESORA GUÍA:
VIVIANA MERUANE NARANJO

MIEMBROS DE LA COMISIÓN:
ALEJANDRO ORTIZ BERNARDIN
ENRIQUE LÓPEZ DROGUETT

SANTIAGO DE CHILE

2018

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERA CIVIL MECÁNICA
POR: CONY DE LOS ÁNGELES ESPINOZA QUITRAL
FECHA: OCTUBRE 2018
PROF. GUÍA: VIVIANA MERUANE NARANJO

IDENTIFICACIÓN DE IMPACTOS EN UNA ESTRUCTURA TRIDIMENSIONAL UTILIZANDO AUTOENCODERS Y UNA APROXIMACIÓN LINEAL DEL PRINCIPIO DE MÁXIMA ENTROPÍA

En general los sistemas con recubrimientos expuestos a impactos son incapaces de detectar problemas en tiempo real y con precisión. La motivación de este trabajo es avanzar en diseñar estructuras que reconozcan las alteraciones que reciban del medio. Una forma de monitorear la integridad de estos sistemas es mediante el desarrollo de algoritmos de detección de impactos basados en la respuesta vibratoria del sistema a alteraciones de tipo impacto. En este contexto se propone utilizar autoencoders (AE). Los AE son un tipo de red neuronal que permite extraer la información más significativa de datos con múltiples variables. Con esto, la estructura puede “aprender” de los impactos que recibe para reconocer eficientemente los impactos futuros.

El objetivo de este trabajo es desarrollar un sistema de identificación de la ubicación y magnitud de impactos recibidos por una estructura. Se elige trabajar con una estructura cilíndrica metálica, emplazada en dos configuraciones a estudiar. Una con la estructura cilíndrica ubicada en posición vertical sobre una superficie lisa. Y otra con la estructura suspendida por medio de una cuerda emplazada a lo largo del cilindro. Ahora, se entrena un AE para extraer el espacio latente de los datos medidos. Luego, se entrena un algoritmo de aprendizaje supervisado basado en el principio de máxima entropía (LME), que permite reconocer y asociar señales de impactos con aquellas de la fase de entrenamiento. Dando como resultado una estimación de la ubicación y magnitud de los impactos recibidos por la estructura.

Para ambas etapas se deben seleccionar las variables que permitan obtener el algoritmo más eficiente y que entreguen los resultados con el menor error asociado posible. El desempeño del AE es bueno si sólo se analiza su capacidad de reconstruir la señal, pero al analizar la estructura del espacio latente, esta resulta representar bien los datos en forma, pero debe mejorar en precisión. Por otro lado, el análisis del comportamiento de la etapa LME según el número de vecinos a utilizar resulta acorde a la interferencia de las condiciones de borde del problema y de cómo afectan las zonas de contacto con elementos externos a los resultados finales. Además de la superficie de contacto con la base, otro elemento que produce un aumento en el error de predicción es la presencia del cordón de soldadura a lo largo de la altura del cilindro. Finalmente, es posible afirmar que el método desarrollado en general cumple su función de identificar la ubicación y magnitud de los impactos efectuados a la estructura. Sin embargo, es necesario trabajar en la precisión de este.

A la Mina, quien sería la persona más feliz y orgullosa de que me titule. Siempre pienso en usted y siempre lo haré.

AGRADECIMIENTOS

Es muy loco pensar que realmente llegó este momento en el que el fin (en el buen sentido) está tan cerca y tengo la oportunidad de pensar en quienes me han acompañado y ayudado a llegar a este día.

En primer lugar, están mis papás, son profesores y desde chica me han tratado de transmitir el gusto por aprender. Y que a pesar de lo estrictos que son, siempre me regalonean y buscan que ponga un poquito más de esfuerzo en lo que sea que esté haciendo. A mi papá que es un oso cariñoso en el traje de un oso feroz, y mi mamá que es la mezcla perfecta entre la mejor conversadora, y la persona más practica y lógica del mundo.

A la Karina que me ha dado nervios de acero, pero siempre me sorprende con algún detalle tierno y kawai. Lo he pasado muy bien contigo, aunque a veces te quiero destruir y tu a mí. Sigue siendo diva y usándome como trofeo con tus amigas porque soy bakán.

Al Nicolás Daniel que es la persona que siempre logra calmarme cuando me pongo nerviosa, quien me hace sentir que todo va a salir bien. Vivimos y sufrimos la U juntos y espero haber sido tan buena contigo como tu has sido conmigo (aunque yo soy la mejor persona jajaja). La Javiera está desde el día 1, y los 84 años restantes. La María Paulina siempre me apaña cuando tengo nuevas ideas (las cuales han sido bastantes), además que siempre tiene los mismos dramas de hermana mayor que yo.

A las chicocas del colegio que siguen estando, que les encanta presentarme en los carretes como “la cabezona”, las adoro, siempre me sacan risas y son un aire fresco en los días cargados de trabajo.

A mis amiguites mecánicos, fue bakán sufrir juntos, saber que estábamos todos pasando por lo mismo. Compartir esas largas noches de estudio o trabajo, tanto en la casa como en la U. Además de tantas conversaciones, cantar reggaetón mientras hacíamos informes, tomar once, preparar presentaciones, planear carretes, etc. Me ayudaron mucho a sobrevivir.

Mis tías y tíos (y primes) siempre pendientes, Nono (padrino siempre alerta, gracias a el tengo estufa y no muero de frio), Nito (y todos los pelolais que los adoro), Tía Edith (que me matriculó, la más motivada con mi entrada a Beauchef), tía Blanca (las mejores sopaipillas de la vida, que me mandaba desde Laja <3), y a todos con los que he compartido tallas, risas, comida y cariño.

A los que ya no están, que los extraño todos los días y amaría abrazarlos de nuevo. En especial a la Mina y el Papo que me malcriaron tanto y me hicieron muy feliz.

Y lo más importante, a la Profe Vivi, que me aceptó como su memorista, y me ayudó en todo. La admiro mucho.

TABLA DE CONTENIDO

1	INTRODUCCIÓN	1
1.1	ANTECEDENTES GENERALES.....	1
1.1.1	Detección de Impactos	1
1.1.2	Análisis de vibraciones aplicado para detección de impactos.....	2
1.1.3	Algoritmos de detección de impactos.....	2
1.2	MOTIVACIÓN.....	3
1.3	OBJETIVOS.....	3
1.3.1	Objetivo General.....	3
1.3.2	Objetivos Específicos.....	3
1.4	ALCANCES.....	4
2	ANTECEDENTES ESPECÍFICOS.....	5
2.1	RESPUESTA VIBRATORIA A UN IMPULSO	5
2.2	PROCESAMIENTO DE SEÑALES.....	7
2.2.1	Método de Ventana de Tiempo	7
2.2.2	Transformada de Hilbert.....	8
2.3	AUTOENCODERS (AE)	10
2.4	PRINCIPIO DE MÁXIMA ENTROPÍA (PME).....	11
2.5	MÉTODO DE APROXIMACIÓN LINEAL BASADO EN EL PRINCIPIO DE MÁXIMA ENTROPÍA	12
2.6	MULTIPLICADORES DE LAGRANGE	16
2.7	EVALUACIÓN DE MÉTODOS DE DETECCIÓN DE IMPACTOS	17
3	METODOLOGÍA.....	20
3.1	MONTAJE EXPERIMENTAL	20
3.1.1	Estructura Cilíndrica	20
3.1.2	Sensores Piezoeléctricos.....	21
3.1.3	Tarjeta de Adquisición.....	22
3.2	ETAPAS DE TRABAJO	23
3.2.1	Lectura de datos	24
3.2.2	Expansión de la base de entrenamiento	24
3.2.3	Aplicación del AE.....	25
3.2.4	Aplicación de LME	26
3.2.5	Determinación de Error.....	26
3.3	PRUEBAS REALIZADAS.....	27
3.3.1	Mediciones impactos de entrenamiento configuración vertical.....	27
3.3.1	Mediciones impactos de entrenamiento configuración suspendida.....	28
3.3.2	Mediciones datos de prueba.....	29

4	RESULTADOS	30
4.1	RESULTADOS ESTRUCTURA VERTICAL	30
4.1.1	Resultados Entrenamiento Autoencoder	30
4.1.2	Resultados LME	31
4.2	RESULTADOS ESTRUCTURA SUSPENDIDA	34
4.2.1	Resultados Entrenamiento Autoencoder	34
4.2.2	Resultados LME	35
4.3	DISCUSIÓN	38
5	CONCLUSIONES	43
6	BIBLIOGRAFÍA	44
7	ANEXOS	45
7.A	CÓDIGO MATLAB – LECTURA DE DATOS DE ENTRENAMIENTO	45
7.B	CÓDIGO MATLAB – LECTURA DE DATOS DE PRUEBA	47
7.C	CÓDIGO MATLAB – EXPANSIÓN BASE DE ENTRENAMIENTO	48
7.D	CÓDIGO PYTHON – IMPLEMENTACIÓN AUTOENCODER	49
7.E	CÓDIGO MATLAB – IMPLEMENTACIÓN LME	51
7.F	CÓDIGO MATLAB – CÁLCULO ERROR	52
7.G	UBICACIÓN SENSORES PIEZOELÉCTRICOS	53
7.H	UBICACIÓN Y MAGNITUD IMPACTOS DE ENTRENAMIENTO CONFIGURACIÓN VERTICAL	53
7.I	UBICACIÓN Y MAGNITUD IMPACTOS DE ENTRENAMIENTO CONFIGURACIÓN SUSPENDIDA	56
7.J	UBICACIÓN Y MAGNITUD IMPACTOS DE PRUEBA CONFIGURACIÓN VERTICAL	60
7.K	UBICACIÓN Y MAGNITUD IMPACTOS DE PRUEBA CONFIGURACIÓN SUSPENDIDA	61

INDICE DE TABLAS

Tabla 3.1 Composición química y propiedades mecánicas de la cañería utilizada. Fuente.	21
Tabla 3.2 Propiedades sensor piezoeléctrico 7BB-20-6L0.	21
Tabla 3.3 Propiedades tarjeta de adquisición Econ-MI7016.	22
Tabla 3.4 Propiedades martillo modal LC-01A. Fuente: Global Sensor Technology.....	23
Tabla 3.5 Dimensiones capas de reducción y espacio latente del AE.....	25
Tabla 4.1 Resultados AE optimizado, estructura vertical.	31
Tabla 4.2 Evaluación algoritmo, estructura vectical.	32
Tabla 4.3 Resultados AE optimizado, estructura suspendida.....	35
Tabla 4.4 Evaluación algortimo, estructura suspendida.	36
Tabla 4.5 Comparación desempeño AE.....	38
Tabla 4.6 Comparación elección Nro. de vecinos.....	39
Tabla 4.7 Comparación método AE+LME con PCA+LME.	41
Tabla 4.8 Desempeño algoritmo para placa plana.	42

INDICE DE ILUSTRACIONES

Figura 1.1: Esquema de un montaje experimental típico de detección de impactos [1].....	1
Figura 2.1 Fuerza tipo impulso en el tiempo.....	5
Figura 2.2 Respuesta de un sistema a un impulso en un tiempo t	7
Figura 2.3 Ventana rectangular.	8
Figura 2.4 Señal original de respuesta a un impacto.	9
Figura 2.5 Envolvente de la señal original con respecto al tiempo.....	9
Figura 2.6 Estructura esquemática de un Autoencoder.....	10
Figura 2.7 Matriz X original (azul) y reconstruido (naranja) por el AE.....	11
Figura 3.1 Sensor piezoeléctrico 7BB-20-6L0.....	21
Figura 3.2 Sistema de Adquisición.....	22
Figura 3.3 Martillo Modal.....	22
Figura 3.4 Diagrama de flujo de estrategia de identificación de impactos.	23
Figura 3.5 Fotografías configuraciones utilizadas. (Izq.) Vertical. (Der.) Suspendida.....	27
Figura 3.6 Ubicación impactos de entrenamiento, configuración vertical.....	28
Figura 3.7 Ubicación impactos de entrenamiento, configuración suspendida.	28
Figura 3.8 Ubicación impactos de prueba.	29
Figura 4.1 Costo AE vs. número de épocas.....	30
Figura 4.2 Costo AE vs. tasa de aprendizaje.....	30
Figura 4.3 Error de área en función del número de vecinos.....	31
Figura 4.4 Error de fuerza en función del número de vecinos.....	31
Figura 4.5 Error de identificación de impacto normalizado, en función de n_v	32
Figura 4.6 Resultados AE+LME estructura vertical, coordenada θ	32
Figura 4.7 Resultados AE+LME estructura vertical, coordenada z	33
Figura 4.8 Resultados AE+LME estructura vertical, magnitud fuerza de impacto.	33
Figura 4.9 Resultados AE+LME estructura vertical, ubicación de impactos.....	33
Figura 4.10 Costo AE vs. número de épocas.	34
Figura 4.11 Costo AE vs. tasa de aprendizaje.....	34
Figura 4.12 Error de área en función del número de vecinos.	35
Figura 4.13 Error de fuerza en función del número de vecinos.	35
Figura 4.14 Error de identificación de impacto normalizado, en función de n_v	36
Figura 4.15 Resultados AE+LME estructura suspendida, coordenada θ	36

Figura 4.16 Resultados AE+LME estructura vertical, coordenada z.....	37
Figura 4.17 Resultados AE+LME estructura suspendida, magnitud fuerza de impacto.....	37
Figura 4.18 Resultados AE+LME estructura suspendida, ubicación de impactos.	37
Figura 4.20 Resultados PCA+LME (Izq.) Configuración vertical y suspendida.	41
Figura 4.21 Fotografía costura presente al interior del cilindro.	42

1 INTRODUCCIÓN

1.1 ANTECEDENTES GENERALES

1.1.1 Detección de Impactos

Monitorear en tiempo real la integridad estructural ha llamado gran interés en diversas aplicaciones de ingeniería. El poder garantizar la integridad de una estructura tiene un gran potencial en aumentar la seguridad de las personas, como también poder obtener mayores beneficios económicos [1].

Por otra parte, en aplicaciones de ingeniería, es habitual el uso de placas y estructuras metálicas compuestas. Este tipo de estructuras destacan debido a que presentan una alta resistencia y versatilidad en el diseño, a la vez que logran minimizar el peso [2]. Para asegurar la integridad de estos sistemas se analiza su respuesta vibratoria a fuerzas externas, particularmente a fuerzas de tipo impacto. Estos eventos son comunes en componentes aeronáuticos y elementos estructurales.

Para evitar que estos eventos de tipo impacto lleguen a causar fallas catastróficas, es fundamental detectar la presencia de daño apenas este es producido. Esto debido a que existen impactos que no dañan de forma severa la estructura, sin embargo, pueden ser precursores de un daño mayor que avance en el tiempo, hasta producirse una falla catastrófica.

El principio de un sistema de identificación de impactos es detectar, localizar y cuantificar una fuerza de impacto, con el uso de un sistema pasivo de sensores, los cuales se encuentran distribuidos sobre la superficie de la estructura. Esto se puede observar en la Figura 1.1, donde se ilustra un montaje experimental típico de detección de impactos, el cual utiliza sensores piezoeléctricos que detectan las ondas de esfuerzos que se propagan por la superficie. Estas ondas son generadas por una fuerza de impacto, producida con un martillo modal el cual golpea la estructura.

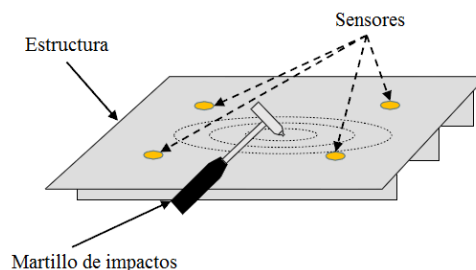


Figura 1.1: Esquema de un montaje experimental típico de detección de impactos [1].

1.1.2 Análisis de vibraciones aplicado para detección de impactos

El estudio de vibraciones basa su análisis en la respuesta vibratoria de un sólido al ser sometido a algún tipo de excitación. Dicha respuesta será característica del sistema y está dada por sus propiedades físicas y geométricas. Las características principales que definen una vibración son la amplitud y la frecuencia. De la amplitud de vibración pueden referirse los parámetros de desplazamiento, velocidad y aceleración.

Por otra parte, al conocer la respuesta vibratoria típica de un sistema, se pueden utilizar distintos algoritmos que permitan determinar características de la excitación inicial que provocó dicha respuesta. Para el caso de excitaciones causadas por un impacto, se pueden estudiar modelos que responden a excitaciones de tipo impulso.

A partir de este tipo de estudio, es posible estimar la localización y magnitud de una fuerza de impacto en base a señales vibratorias en distintos puntos del sólido. Se requieren como mínimo tres sensores para poder determinar la ubicación del evento en una placa plana [1].

1.1.3 Algoritmos de detección de impactos

Existen distintos algoritmos que pueden ser utilizados para determinar la localización y magnitud de impactos realizados sobre una estructura.

Actualmente, se han desarrollado estudios de impactos utilizando algoritmos basados en Redes Neuronales Artificiales (ANN, por sus siglas en inglés), los cuales han obtenido buenos resultados, tanto en estructuras simples como compuestas. El ANN se destaca por procesar datos de manera rápida, sin embargo, necesita un mayor tiempo de aprendizaje y para su aplicación se deben determinar un gran número de parámetros [1].

Otro algoritmo bastante utilizado es el de Máquinas de Vectores de Soporte (SVM, por sus siglas en inglés), el cual presenta una ventaja en cuanto a optimización global y mejores capacidades de generalización que el ANN. También se ha desarrollado el algoritmo de Mínimos Cuadrados de Maquinas de Vectores de Soporte (LSSVM, por sus siglas en inglés) que simplifica la regresión a un problema que puede ser resuelto con un grupo de ecuaciones lineales [1].

Un algoritmo que ha obtenido muy buenos resultados consiste en un método de aproximación lineal, el cual está basado en el Principio de Máxima Entropía (PME), y se denota como Aproximación Lineal con Máxima Entropía (LME, por sus siglas en inglés). Este algoritmo se destaca en comparación con otros algoritmos, ya que no necesita de entrenamiento previo [1], además que sólo se necesita determinar un parámetro para su implementación [2].

Por otra parte, para el procesamiento de datos, también es utilizado comúnmente el método de Análisis de Componentes Principales (PCA). Esta técnica selecciona los patrones o características más significativas de datos con múltiples variables, para simplificar problemas complejos, mientras que retiene la mayor cantidad de información [3].

Se ha observado en algunos estudios, que la incorporación del método PCA como pre procesamiento de datos, en conjunto con otro algoritmo, logra reducir el error en las estimaciones de la ubicación del impacto [3]. Es por esto, que éste trabajo de memoria de título se centra en la evaluación de otro tipo de pre procesamiento.

1.2 MOTIVACIÓN

En la actualidad, cuando se producen impactos de tipo “Barely Visible Impact Damage” (BVID), definido como aquel daño que se encuentra de forma interna en la estructura y no es detectable con una inspección visual simple. Por lo tanto, se utilizan otras técnicas, tales como, ultrasonido o termografía activa, las cuales pueden ser costosas y tardar bastante tiempo al ser aplicadas a la totalidad de una estructura.

Un sistema de detección de impactos permite localizar puntos que posiblemente hayan sufrido daño, reduciendo el costo y tiempo de las inspecciones. Para aumentar el desempeño de los sistemas de detección de impactos es importante mejorar las estimaciones de localización e intensidad de dichos eventos y acercar este tipo de estudio a un caso real, como es el caso de impactos en estructuras con una geometría cada vez más compleja.

1.3 OBJETIVOS

1.3.1 Objetivo General

El objetivo del estudio es desarrollar un sistema de detección, localización y cuantificación de la intensidad de los impactos en una estructura tridimensional simple, a partir de un análisis de su respuesta vibratoria, utilizando los métodos de autoencoder (AE) y una aproximación lineal bajo el principio de máxima entropía.

1.3.2 Objetivos Específicos

Para alcanzar el objetivo general, se necesita cumplir con los siguientes hitos:

- Desarrollar una metodología que entregue estimaciones de posición y magnitud de la intensidad de los impactos, utilizando auto codificado y una aproximación lineal.
- Diseño y construcción de un montaje experimental para la detección de impactos en una estructura tridimensional simple.

- Evaluación del desempeño de la metodología desarrollada comparada con datos de impactos conocidos.

1.4 ALCANCES

El presente trabajo está limitado a generar una metodología de identificación de impactos para una estructura metálica cilíndrica. Además del diseño y construcción de un banco de pruebas para su implementación y validación.

2 ANTECEDENTES ESPECÍFICOS

2.1 RESPUESTA VIBRATORIA A UN IMPULSO

Una fuente usual de vibraciones es la aplicación de una fuerza de corta duración denominada impulso. Una excitación tipo impulso es una fuerza de impacto no periódica, aplicada por un periodo muy breve de tiempo (o infinitesimal). La respuesta de un sistema a un impulso es idéntica a la respuesta libre del sistema a una cierta condición inicial.

Consideremos una representación matemática de un impulso, la cual se puede describir de acuerdo con la siguiente relación:

$$F(t) = \begin{cases} 0 & t \leq \tau - \epsilon \\ \frac{\hat{F}}{2\epsilon} & \tau - \epsilon \leq t \leq \tau + \epsilon \\ 0 & t \geq \tau + \epsilon \end{cases}, \quad \epsilon \rightarrow 0 \quad (2.1)$$

Donde τ corresponde al instante en el que se aplica el impulso. La figura 2.1 grafica el comportamiento de esta fuerza.

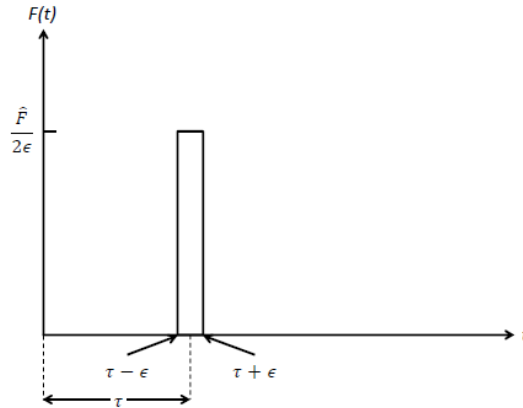


Figura 2.1 Fuerza tipo impulso en el tiempo.

La definición numérica de impulso se obtiene al integrar la expresión para la fuerza en el tiempo, como se presenta a continuación:

$$I(\epsilon) = \int_{-\infty}^{\infty} F(t)dt = \frac{\hat{F}}{2\epsilon} \cdot 2\epsilon = \hat{F} \quad (2.2)$$

Dado que $\epsilon \rightarrow 0$, y mientras no alcance el valor de cero, se tiene que $I(\epsilon) = \hat{F}$.

Si la magnitud de \hat{F} es igual a 1, se denomina impulso unitario $\delta(t)$, también conocido como la función delta de Dirac.

La respuesta un impulso para un sistema de un grado de libertad inicialmente en reposo, se puede determinar aprovechando el hecho que un impulso produce un cambio de

momentum al cuerpo. Para simplificar el análisis, consideremos $\tau = 0$ en la definición del impulso. Este instante de tiempo lo denotaremos como 0^- . Dado que el sistema se encuentra inicialmente en reposo, las condiciones iniciales son ambas cero:

$$\dot{x}(0^-) = \dot{x}(0^-) = 0 \quad (2.3)$$

Por lo tanto, luego del golpe el cambio de momentum es:

$$m\dot{x}(0^+) - m\dot{x}(0^-) = mv_0 \quad (2.4)$$

Donde:

$\dot{x}(0^-)$: Velocidad en el instante inmediatamente anterior al impacto.

$\dot{x}(0^+)$: Velocidad en el instante inmediatamente posterior al impacto.

v_0 : Velocidad inicial del sistema.

De manera que:

$$\hat{F} = F\Delta t = mv_0 - 0 = mv_0 \quad (2.5)$$

Mientras que el desplazamiento inicial se mantiene en cero. Por lo tanto, un impulso aplicado a un sistema de un grado de libertad es lo mismo que aplicar una condición inicial de desplazamiento cero y velocidad inicial de $v_0 = \hat{F}/m$.

Para un sistema con amortiguamiento débil ($0 < \xi < 1$), la respuesta a las condiciones iniciales $x_0 = 0, v_0 = \hat{F}/m$ es de la forma:

$$x(t) = \frac{\hat{F}e^{-\xi\omega_n t}}{m\omega_d} \cdot \text{sen}(w_d t) \quad (2.6)$$

Es conveniente escribir la ecuación anterior como:

$$x(t) = \hat{F}h(t) \quad (2.7)$$

Donde $h(t)$ viene dada por:

$$h(t) = \frac{1}{m\omega_d} \cdot e^{-\xi\omega_n t} \cdot \text{sen}(w_d t) \quad (2.8)$$

Notar que $h(t)$ es la respuesta a un impulso en $t = 0$. Si se aplica en $t = \tau, \tau \neq 0$, se puede escribir como:

$$h(t - \tau) = \frac{1}{m\omega_d} \cdot e^{-\xi\omega_n(t-\tau)} \cdot \text{sen}(w_d(t - \tau)) \quad (2.9)$$

Las funciones $h(t)$ y $h(t - \tau)$ se denomina función de respuesta a un impulso.

En la práctica, una fuerza es considerada un impulso si su duración Δt es muy breve en comparación con el periodo, $T = 2\pi/\omega_n$, asociado con la frecuencia natural de la

estructura. En la figura 2.2 se ilustra una respuesta a un impulso típica para un sistema de un grado de libertad.

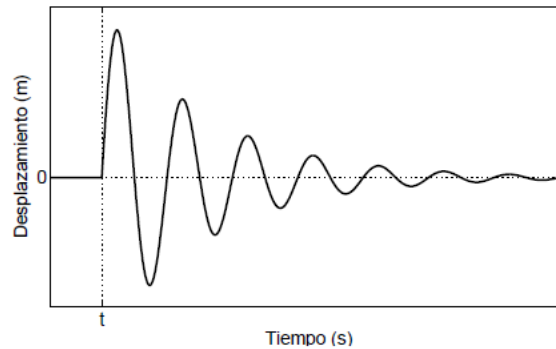


Figura 2.2 Respuesta de un sistema a un impulso en un tiempo t .

En el diseño de sistemas sometidos a fuerzas tipo impulso o escalón, se pueden definir tres parámetros que miden la cantidad de la respuesta:

- El tiempo en alcanzar la amplitud máxima, denominado t_p .
- La máxima amplitud de la respuesta con respecto a la respuesta estacionaria, denominado sobressalto.
- El tiempo de establecimiento t_s , que corresponde al tiempo que tarda la respuesta en permanecer dentro de un cierto porcentaje de la respuesta estacionaria.

2.2 PROCESAMIENTO DE SEÑALES

Al realizar variadas mediciones de un sistema de forma independiente, es necesario procesar las señales obtenidas para lograr obtener la mayor cantidad de información relevante para el problema estudiado. Para esto, resulta de mucha utilidad estudiar las señales en distintos dominios [5].

Existen decenas de métodos de procesamiento de señales, pero se decide utilizar los métodos usualmente utilizados en sistemas de identificación de impactos.

2.2.1 Método de Ventana de Tiempo

El uso de ventanas de tiempo es obligatorio al procesar una señal digital. Estas permiten fijar la zona temporal de datos de interés, especialmente en el caso de respuestas a impulsos, donde solo se necesita estudiar el intervalo de tiempo acorde al impulso.

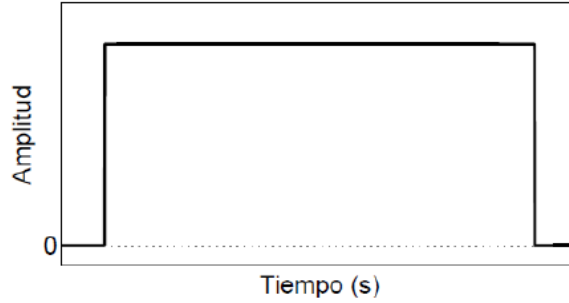


Figura 2.3 Ventana rectangular.

Existen distintos tipos de ventanas, por lo que una buena selección de la ventana a utilizar puede reducir considerablemente errores comunes en la adquisición de datos. En el presente trabajo se utiliza una ventana rectangular como la presentada en la figura 2.3.

Para esto, se define una ventana que incluya 12000 datos de voltaje por cada impacto medido, tomando los 500 datos anteriores al instante del impacto, y los siguientes 11499. Lo cual se traduce a una ventana de tiempo de aproximadamente 0.33 segundos.

2.2.2 Transformada de Hilbert

La transformada de Hilbert de una función real $s(t)$, se calcula de la convolución de las señales $s(t)$ y $1/(\pi t)$, de donde se obtiene la transformada $\hat{s}(t)$. En el presente trabajo se utiliza esta herramienta para identificar de mejor manera los máximos de la señal e identificar el tiempo de desfase entre señales, para el caso del algoritmo LME.

La transformada de Hilbert se define a continuación [6].

$$\hat{s}(t) = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{s(\tau)}{t - \tau} d\tau \quad (2.10)$$

Donde:

$s(t)$: Señal original con respecto al tiempo.

$\hat{s}(t)$: Transformada de Hilbert de la señal $s(t)$.

Luego, la señal analítica se construye con la señal original como la parte real y la transformada de Hilbert como la parte compleja, como se muestra en la ecuación 2.11.

$$a(t) = s(t) + i \cdot \hat{s}(t) \quad (2.11)$$

Donde:

$a(t)$: Señal analítica de $s(t)$.

i : Unidad imaginaria $\sqrt{-1}$.

Con esto se calcula la envolvente de la señal, como se muestra en (2.12). Esta función entrega el valor de los extremos de la amplitud de la señal en función del tiempo.

$$e(t) = \sqrt{s^2(t) + \hat{s}^2(t)} \quad (2.12)$$

Donde:

$e(t)$: Envolvente de la señal $s(t)$.

En la Figura 2.4, se presenta gráficamente la señal de respuesta a un impacto con respecto al tiempo de adquisición, medida con un sensor ubicado sobre la superficie de la estructura. Mientras que en la Figura 2.5 se presenta la envolvente de la señal original.

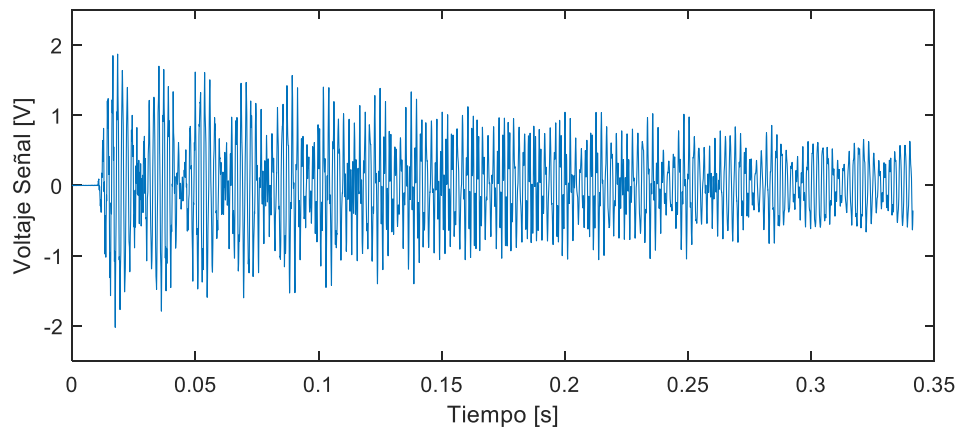


Figura 2.4 Señal original de respuesta a un impacto.

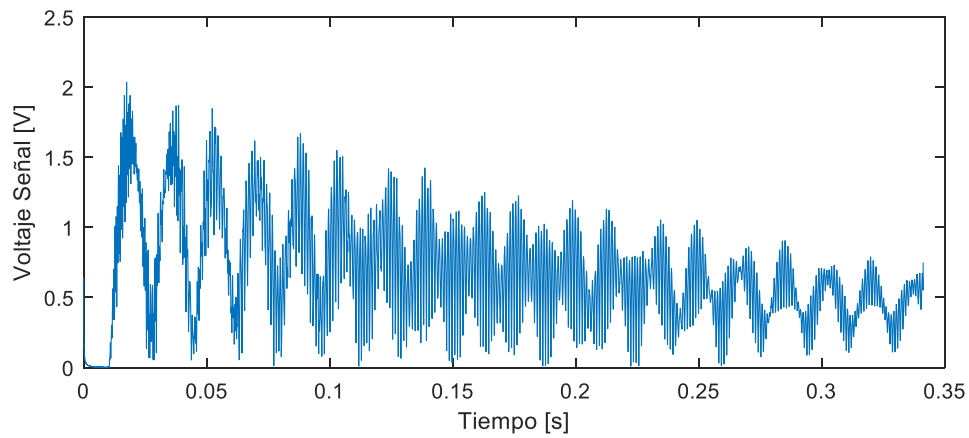


Figura 2.5 Envolvente de la señal original con respecto al tiempo.

2.3 AUTOENCODERS (AE)

Los autoencoders son redes neuronales artificiales capaces de generar una representación eficiente de los datos de entrada sin supervisión, esta representación es llamada espacio latente [3]. Este es por lo general de una dimensión bastante menor que el set de datos de entrada, lo que hace a los autoencoders útiles para problemas donde se necesite aplicar una reducción dimensional que represente las características principales del set de datos original. La Figura 2.6 presenta un esquema de la estructura típica de un AE.

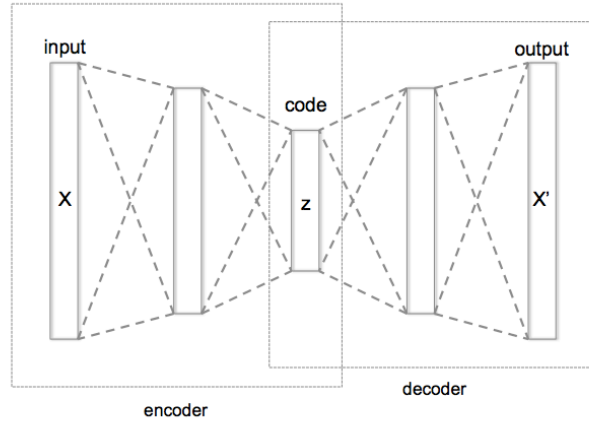


Figura 2.6 Estructura esquemática de un Autoencoder.

Sorpresivamente, los autoencoders trabajan aprendiendo a copiar los datos de entrada en los datos de salida. Para lograr un procesamiento útil, se debe evitar realizar el procedimiento trivial de invertir las operaciones de reducción (encoding) en la etapa de reconstrucción (decoding), si se cae en este error el espacio latente corresponderá simplemente a la matriz identidad (I). Para evitar esto, se deben escoger correctamente los tamaños de las capas del autoencoder, además de utilizar más de un set de datos (datos de entrenamiento y datos de prueba), lo que fuerza a la red a aprender formas eficientes de representar los datos.

El AE tiene como objetivo aprender la función presentada en la ecuación 2.13, es decir que busca una aproximación de la función identidad que permita obtener un \hat{x} lo más similar posible a el x de entrada [3].

$$h_{W,b}(x) \approx x \quad (2.13)$$

Donde W corresponde a los pesos del AE. En casos de AE simétricos (como el del presente trabajo), se le asigna a los pesos (W), las dimensiones de las capas de reducción del AE. Mientras que b corresponde a la dimensión del espacio latente.

En la figura 2.7, se presenta gráficamente la señal entregada al AE en color azul, y la señal reconstruida en color naranja.

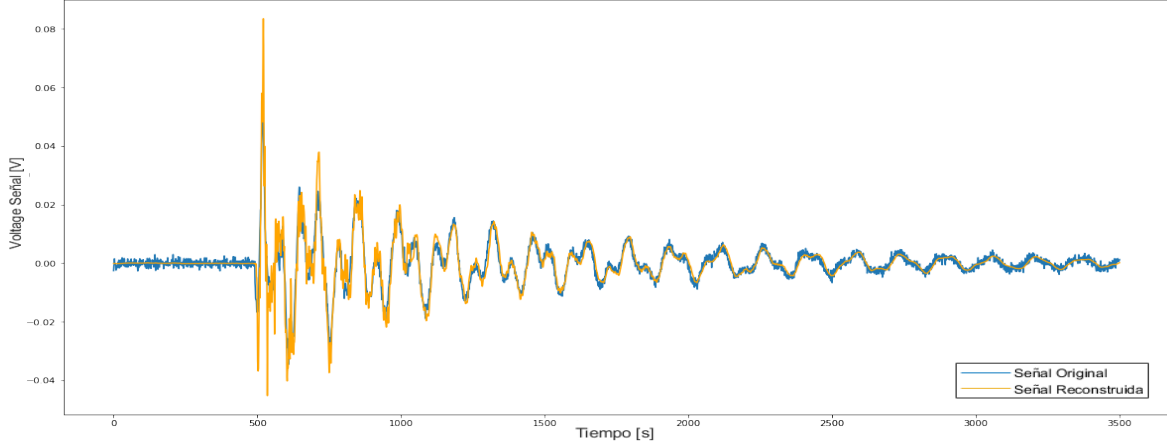


Figura 2.7 Matriz X original (azul) y reconstruido (naranja) por el AE.

Como se menciona anteriormente, para evitar obtener el resultado trivial de función identidad (I), se debe poner atención a las variables de trabajo del AE.

Entre estas se tiene la tasa de aprendizaje (learning rate) de la red, variable que define la cantidad de información que el algoritmo toma en cada iteración, y en general se busca utilizar una tasa baja para aumentar el rendimiento del aprendizaje. Luego, se elige el tamaño de lote (batch size) de datos a tomar por cada sesión de entrenamiento. También, al entrenar el AE, se deben seleccionar la cantidad de épocas (o recursiones) necesarias para optimizar el proceso. Finalmente se debe definir la dimensión de la capas de reducción y del espacio latente.

Para evaluar el desempeño del AE, se calcula el costo de reconstrucción de los datos de entrenamiento, definido como la Raíz del Error Cuadrático Medio (RMSE), presentado en la ecuación 2.14.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - x_{i-rec})^2} \quad (2.14)$$

Donde:

n : Número de datos.

x_i : i -ésimo dato en el vector x .

x_{i-rec} : i -ésimo datos en el vector reconstruido de x .

2.4 PRINCIPIO DE MÁXIMA ENTROPÍA (PME)

La noción de entropía en la información es explicada por Shannon como una medida de la incertidumbre. El Principio de Máxima Entropía sostiene que la distribución de probabilidad menos sesgada que se le puede atribuir a un sistema es aquella que maximiza

la entropía, o incertidumbre, es decir, aquel momento en el que la desinformación es máxima [4].

Si se considera una serie de N eventos discretos $\{x_1, x_2, \dots, x_N\}$, donde la probabilidad de cada evento está dada por $p_a = p(x_a) \in [0,1]$, la incertidumbre asociada a cada p_a es $-\ln(p_a)$. La entropía de Shannon (ecuación 2.15), es la cantidad de incertidumbre representada por la incertidumbre $\{p_1, \dots, p_N\}$. La distribución de probabilidad menos sesgada y la que tiene mayor probabilidad de ocurrir es obtenida a partir del problema de optimización presentado en la ecuación 2.16.

$$H(p) = - \sum_{i=1}^N p_i \cdot \ln(p_i) \quad (2.15)$$

$$\max_{p \in \mathbb{R}_+^N} \left[H(p) = - \sum_{i=1}^N p_i \cdot \ln(p_i) \right] \quad (2.16)$$

Bajo las restricciones:

$$\sum_{i=1}^N p_i = 1 \quad (2.17)$$

$$\sum_{i=1}^N p_i \cdot g_r(x_i) = \langle g_r(x) \rangle \quad (2.18)$$

Donde:

\mathbb{R}_+^N : Ortante no negativo.

$\langle g_r(x) \rangle$: Valor esperado para la función de densidad $g_r(x) (r = 0, 1, \dots, m)$, con $g_0(x) = 1$ como condición de normalización.

2.5 MÉTODO DE APROXIMACIÓN LINEAL BASADO EN EL PRINCIPIO DE MÁXIMA ENTROPÍA (LME)

Para abordar el problema de detección de impactos, este se resuelve como una regresión lineal. Para realizar esto se define un vector de datos observados, presentado a continuación en la ecuación 2.19.

$$Y^j = \{y_1^j, y_2^j, y_3^j\} \quad (2.19)$$

Donde j representa al j -ésimo impacto sobre la estructura, mientras y_1 e y_2 corresponden a las coordenadas cilíndricas $\hat{\theta}$ y \hat{z} de la localización de la fuerza de impacto, la coordenada $\hat{\rho}$ se mantiene constante ya que los impactos se producen sobre la superficie

de la estructura cilíndrica. Por otro lado y_3 corresponde a la magnitud en Newton de la fuerza de impacto.

Luego, se define el vector X^j (ecuación 2.20) que representa un grupo de parámetros de respuestas asociadas al impacto Y^j . Los parámetros de respuesta considerados varían según la estrategia de identificación utilizada.

$$X^j = \{x_1^j, x_2^j, \dots, x_m^j\} \quad (2.20)$$

Para el caso de LME, los valores x_m corresponden al máximo valor de la envolvente, al tiempo de llegada de la envolvente y el tiempo del primer máximo.

Ahora, para el caso de LME con un procesamiento previo con AE, se utiliza el espacio latente que contiene la mayor información posible de los datos de entrenamiento, obtenidos a partir de los valores de amplitud de la envolvente de la señal de cada sensor. La envolvente entrega una gran cantidad de datos, los cuales se reducen utilizando AE para extraer únicamente m datos.

El objetivo en la identificación de impactos es estimar el correspondiente impacto Y en base al vector de características observadas X . Para lograr esto, se construye una base de datos de J impactos, formada por parejas $(X^1, Y^1), (X^2, Y^2), \dots, (X^j, Y^j)$. Luego, el vector X se representa como una combinación lineal de los N vecinos X^j más cercanos en la base de datos, ponderado por un vector w con los elementos w_j que cumplen con la condición.

$$X = \sum_{j=1}^N w_j(X) X^j \quad (2.21)$$

$$\sum_{j=1}^N w_j(X) = 1 \quad (2.22)$$

Donde:

$w_1(X), w_2(X), \dots, w_N(X)$: Funciones de ponderación.

X^1, X^2, \dots, X^N : Vectores de características dentro de la base de datos.

La ecuación 2.23, puede escribirse de manera matricial (ecuación 2.24).

$$A \cdot w = b \quad (2.23)$$

$$A = \begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^N \\ x_2^1 & x_2^2 & \dots & x_2^N \\ \vdots & \vdots & \ddots & \vdots \\ x_n^1 & x_n^2 & \dots & x_n^N \\ 1 & 1 & \dots & 1 \end{bmatrix}_{(n+1) \times N} \quad (2.24)$$

$$b = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \\ 1 \end{bmatrix}_{(n+1) \times 1} \quad (2.25)$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}_{N \times 1} \quad (2.26)$$

Obteniendo el vector w de la ecuación 2.26, es posible estimar Y . Esta estimación se presenta a continuación en la ecuación 2.27.

$$\hat{Y} = \sum_{j=1}^N w_j(X)Y^j \quad (2.27)$$

Típicamente este sistema de ecuaciones es indeterminado, ya que N solo considera los valores cercanos a la vecindad de X . Sin embargo, el sistema puede ser resuelto aplicando del principio de máxima entropía para obtener los valores del vector w .

Ahora, al retomar el problema de optimización lineal presentado en la ecuación 2.16, bajo las restricciones presentadas en las ecuaciones 2.17 y 2.18. Se observa que es posible aplicar LME al sistema con una distribución de probabilidad previa, lo que permite reducir la incertidumbre con respecto a la probabilidad p_i . Suponiendo que esta toma una distribución previa de m_i , esta reduce su incertidumbre a $-\ln(p_i) + \ln(m_i) = -\ln(p_i/m_i)$, lo que modifica la expresión del problema a la expresión presentada a continuación.

$$\max_{p \in \mathbb{R}_+^N} \left[H(p) = - \sum_{i=1}^N p_i \cdot \ln \left(\frac{p_i}{m_i} \right) \right] \quad (2.28)$$

Bajo las mismas restricciones.

Gracias a los anterior, se reemplaza la probabilidad p_i por la función de ponderación w_i de vector w , proveniente de la primera combinación lineal (ecuación 2.29). Para resolver este sistema se utiliza el método de los multiplicadores de Lagrange.

$$\max_{p \in \mathbb{R}_+^N} \left[H(p) = - \sum_{i=1}^N w_i \cdot \ln \left(\frac{w_i}{m_i} \right) \right] \quad (2.29)$$

Bajo las restricciones:

$$\sum_{i=1}^N w_i = 1 \quad (2.30)$$

$$\sum_{i=1}^N w_i \cdot (X^i - X) = 0 \quad (2.31)$$

Para m_i se utiliza comúnmente la distribución de Gauss.

$$m_i(X) = \exp\left(-\beta_i \|X^i - X\|^2\right) \quad (2.32)$$

$$\beta_i = \frac{\gamma}{h_i^2} \quad (2.33)$$

Donde:

β_i : Depende de γ .

γ : Parámetro que controla la curva de aproximación Gaussiana.

h_i : Distancia euclidiana n-dimensional característica entre los vecinos de la base de datos, para cada X^i .

Finalmente, las matrices X e Y para para el caso de LME sin procesamiento previo, son las presentadas en las ecuaciones 2.34 y 2.35 respectivamente.

$$X = \begin{bmatrix} A_1^1 & \cdots & A_1^n \\ T_1^1 & \cdots & T_1^n \\ t_1^1 & \cdots & t_1^n \\ \vdots & \ddots & \vdots \\ A_s^1 & \cdots & A_s^n \\ T_s^1 & \cdots & T_s^n \\ t_s^1 & \cdots & t_s^n \end{bmatrix} \quad (2.34)$$

$$Y = \begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \\ F_1 & \cdots & F_n \end{bmatrix} \quad (2.35)$$

Donde:

A_m^n : Amplitud máxima de la deformación detectada en el n-ésimo impacto, por el sensor s.

T_m^n : Tiempo de llegada de la envolvente de la señal en el n-ésimo impacto, para el sensor s.

t_m^n : Tiempo que transcurre entre el máximo global de la fuerza y un máximo (global o local) de la deformación en el sensor s, para el n-ésimo impacto.

x_n : Posición en la coordenada $\hat{\theta}$ del n-ésimo impacto.

y_n : Posición en la coordenada \hat{z} del n-ésimo impacto.

F_n : Máximo global de la señal de fuerza del n-ésimo impacto.

Por otra parte, la matriz X para el caso de LME con procesamiento previo (en este caso utilizando AE), corresponde a la presentada en la ecuación 2.36. La matriz Y se mantiene igual a la presentada en la ecuación 2.35.

$$X = \begin{bmatrix} x_{11}^1 & \cdots & x_{11}^n \\ x_{21}^1 & \cdots & x_{21}^n \\ \vdots & \ddots & \vdots \\ x_{ms}^1 & \cdots & x_{ms}^n \end{bmatrix} \quad (2.36)$$

Donde

x_{ms}^n : M-ésimo componente principal de la señal del sensor s.

2.6 MULTIPLICADORES DE LAGRANGE

Para determinar los valores de w_j se utiliza el método de los multiplicadores de Lagrange, el cual permite encontrar mínimos o máximos de una función con una serie de restricciones. En este caso, se busca maximizar la función de la ecuación 2.29, aplicando las siguientes restricciones.

$$\sum_{j=1}^N w_j = 1 \quad (2.37)$$

$$\sum_{j=1}^N w_j X^i - X = 0 \quad (2.38)$$

$$\sum_{j=1}^N w_j Y^i - Y = 0 \quad (2.39)$$

Luego, se define el Lagrangiano como se muestra en la ecuación 2.40.

$$\mathcal{L}(x, \lambda) = f(x) + \sum_{k=1}^M \lambda_k g_k(x) \quad (2.40)$$

Donde:

$f(x)$: Función objetivo.

$g_k(x)$: Representa las restricciones aplicadas.

Para encontrar el mínimo se utiliza el gradiente de \mathcal{L} (ecuación 2.40). En el caso particular ecuación 2.29, el término lagrangiano corresponde a la expresión de la ecuación 2.41, donde los multiplicadores correspondientes λ_r ($r = 0, 1, 2$) están asociados a cada una de las restricciones del problema. Luego, se aplica la derivada parcial con respecto a w_j .

$$\nabla[\mathcal{L}] = 0 \quad (2.41)$$

$$\delta \left[- \sum_{j=1}^N w_j \cdot \log \left(\frac{w_j}{m_j} \right) + \lambda_0 \left(1 - \sum_{j=1}^N w_j \right) + \lambda_1 \left(X - \sum_{j=1}^N w_j X^j \right) + \lambda_2 \left(Y - \sum_{j=1}^N w_j Y^j \right) \right] = 0 \quad (2.42)$$

La ecuación 2.43 presenta el resultado de esta expresión.

$$-1 - \log\left(\frac{w_j}{m_j}\right) - \lambda_0 - \lambda_1 X^j - \lambda_2 Y^j = 0 \quad (2.43)$$

A continuación, se asume $\lambda_0 = \log(Z) - 1$. Donde Z corresponde a la función de partición. De este modo la ecuación puede ser descrita como se presenta a continuación.

$$\log\left(\frac{w_j}{m_j}\right) + \log Z = -\lambda_1 X^j - \lambda_2 Y^j \quad (2.44)$$

$$w_j = \frac{e^{-\lambda_1 X^j - \lambda_2 Y^j}}{Z} \quad (2.45)$$

Tomando la expresión anterior, y debido a que $\sum_{j=1}^N w_j = 1$, se obtiene:

$$Z(\lambda_1, \lambda_2) = \sum_{j=1}^N e^{-\lambda_1 X^j - \lambda_2 Y^j} \quad (2.46)$$

Esta expresión puede ser escrita como una función de peso.

$$w_j = \frac{e^{-\lambda_1 X^j - \lambda_2 Y^j}}{\sum_{j=1}^N e^{-\lambda_1 X^j - \lambda_2 Y^j}} = \frac{Z_j}{\sum_{j=1}^N Z_i} \quad (2.47)$$

Al aplicar las restricciones 2.38 y 2.39 a la expresión de w_j (ecuación 2.47), es posible obtener un sistema de ecuaciones no lineales, el cual se presenta a continuación.

$$\sum_{j=1}^N e^{-\lambda_1 X^j - \lambda_2 Y^j} \cdot X^j - X = 0 \quad (2.48)$$

$$\sum_{j=1}^N e^{-\lambda_1 X^j - \lambda_2 Y^j} \cdot Y^j - Y = 0 \quad (2.49)$$

Con este sistema es posible obtener los parámetros λ_1 y λ_2 . Este tipo de sistema se resuelve mediante un método iterativo.

2.7 EVALUACIÓN DE MÉTODOS DE DETECCIÓN DE IMPACTOS

Se desea comparar el desempeño de los distintos métodos de detección de impactos, por lo tanto, se definen las siguientes funciones que terminan el error de estos métodos al estimar la fuerza y la posición de los impactos [1].

Se define el error promedio en la localización de los impactos en las coordenadas $\hat{\theta}$ y \hat{z} , como las funciones presentadas en las ecuaciones 2.50 y 2.51 respectivamente. Estas

funciones comparan cada coordenada de posición estimada por el algoritmo con la posición real del impacto, y luego obtienen un promedio de esta diferencia para n impactos.

De forma similar se obtiene el error porcentual de área con la función presentada en la ecuación 2.53. Este se calcula al multiplicar los errores de cada coordenada y dividir por el área total de la superficie estudiada, obteniendo finalmente el error de la localización del impacto como un error de área porcentual.

Para el caso del error en la estimación de la fuerza del impacto, se realiza un procedimiento análogo, utilizando la diferencia entre el valor estimado por el algoritmo y el valor experimental medido por el martillo modal. Sin embargo, en este caso se divide por la fuerza experimental, como se muestra en la ecuación 2.52, para obtener el error porcentual.

$$E_{\theta} = \frac{1}{n} \sum_{j=1}^n |\hat{Y}_1^j - Y_1^j| \quad (2.50)$$

$$E_z = \frac{1}{n} \sum_{j=1}^n |\hat{Y}_2^j - Y_2^j| \quad (2.51)$$

$$E_F = \frac{1}{n} \sum_{j=1}^n \frac{|\hat{Y}_3^j - Y_3^j|}{Y_3^j} \times 100 \quad (2.52)$$

$$E_A = \frac{E_{\theta}[\text{mm}] \times E_z[\text{mm}]}{A[\text{mm}^2]} \times 100 = \frac{\sum_{j=1}^n |\hat{Y}_1^j - Y_1^j| \cdot \sum_{j=1}^n |\hat{Y}_2^j - Y_2^j|}{n^2 A} \times 100 \quad (2.53)$$

Donde:

n : Número de elementos en la base de datos de prueba.

A : Área de la placa [mm^2].

Y_1^j : Coordenada $\hat{\theta}$ real del j -ésimo impacto [$^{\circ}$].

\hat{Y}_1^j : Coordenada $\hat{\theta}$ estimada del j -ésimo impacto [$^{\circ}$].

Y_2^j : Coordenada \hat{z} real del j -ésimo impacto [mm].

\hat{Y}_2^j : Coordenada \hat{z} estimada del j -ésimo impacto [mm].

Y_3^j : Fuerza experimental y del j -ésimo impacto [N].

\hat{Y}_3^j : Fuerza estimada del j -ésimo impacto [N].

E_{θ} : Error promedio en la estimación del impacto en la coordenada $\hat{\theta}$ [$^{\circ}$].

E_z : Error promedio en la estimación de impacto en la coordenada \hat{z} [mm].

E_F : Porcentaje de error en la estimación de la magnitud de la fuerza [%].

E_A : Porcentaje de error en la localización del área del impacto [%].

Es importante considerar que el error promedio en la coordenada $\hat{\theta}$ se calcula en unidad de ángulo [°], por lo tanto a continuación se presenta el error de área con las unidades correctas, en la ecuación 2.55.

$$E_A = \frac{(E_\theta \frac{2\pi \cdot r}{360}) \times E_z}{n^2 \cdot 2\pi \cdot r \cdot h} \times 100 = \frac{E_\theta \times E_z}{n^2 \cdot h \cdot 360} \times 100 \quad (2.54)$$

$$E_A = \frac{\sum_{j=1}^n |\hat{Y}_1^j - Y_1^j| \cdot \sum_{j=1}^n |\hat{Y}_2^j - Y_2^j|}{n^2 \cdot h \cdot 360} \times 100 \quad (2.55)$$

Donde:

h : Altura del cilindro [mm].

El error de identificación de impacto normalizado se define como:

$$E_I = \frac{E_A}{E_A^0} \times \frac{E_F}{E_F^0} \quad (2.56)$$

Donde:

E_A^0 : Error de área para un valor inicial de nv [%].

E_F^0 : Error de fuerza para un valor inicial de nv [%].

El algoritmo de identificación de impactos se evalúa para diferentes valores de $nv = \{nv_1, nv_2, \dots, nv_{nk}\}$. Hasta que finalmente, se selecciona el parámetro nv^* que entrega el menor error de identificación de impacto normalizado.

3 METODOLOGÍA

El presente trabajo requiere de una base de datos de entrenamiento que sea capaz de describir de manera uniforme y lo más completa posible la superficie de la estructura. Además de una base de datos de prueba que se distribuya aleatoriamente en la superficie. Es por esto que es importante diseñar un montaje experimental que permita obtener mediciones representativas del comportamiento de la estructura.

Para el desarrollo de este trabajo se dispone de los recursos del Laboratorio de Vibraciones Mecánicas del Departamento de Ingeniería Mecánica de la Universidad de Chile.

3.1 MONTAJE EXPERIMENTAL

El objetivo del montaje experimental es construir un banco de pruebas para realizar ensayos de respuestas vibratorias a impactos en una estructura tridimensional de forma cilíndrica. Las respuestas entregadas por el sistema deben ser medidas por sensores y ser digitalizadas para su posterior procesamiento.

El montaje necesario para la realización del presente trabajo está compuesto por los siguientes sistemas y elementos:

1. Estructura metálica cilíndrica, de radio 150 mm, altura 300 mm y espesor 5 mm.
2. 6 sensores piezoeléctricos adheridos a la superficie de la estructura.
3. Martillo modal, para aplicar los impactos y medir directamente la fuerza de estos.
4. Tarjeta de adquisición de al menos 7 canales.
5. Computador personal con software de adquisición de datos, además de los software Matlab y Python.

A continuación se presentan en mayor detalle los elementos y equipos que forman parte del montaje experimental a utilizar.

3.1.1 Estructura Cilíndrica

La estructura corresponde a una sección de cañería con costura de acero de norma ASTM A53 (propiedades presentadas en la Tabla 3.1).

Tabla 3.1 Composición química y propiedades mecánicas de la cañería utilizada. Fuente

Composición Química								
%C	%Mn	%P	%S	%Cu	%Ni	%Cr	%Mo	%V
0.25	0.95	0.05	0.045	0.4	0.4	0.4	0.15	0.08
Propiedades Mecánicas								
Prop.	Resistencia a la Tracción		Esfuerzo de Fluencia			Elongación		
Magn.	330		205			36		
Unidad	[MPa]		[MPa]			[%]		

Se divide la superficie de la estructura en secciones de 30 mm de alto por 30° de ancho con respecto al centro del cilindro. De esta forma se forma un mallado con 120 secciones y 132 puntos, en los cuales se aplicarán los impactos de entrenamiento.

3.1.2 Sensores Piezoeléctricos

Los sensores piezoeléctricos (Figura 3.2) utilizados en el sistema se caracterizan por ser muy livianos, delgados y de bajo costo. Estas características evitan afectar la respuesta entregada por el sistema. El fabricante además detalla que entregan una señal limpia, con bajo ruido, consumen poca potencia en voltaje y son altamente reutilizables. Sus especificaciones se presentan en la Tabla 3.2.

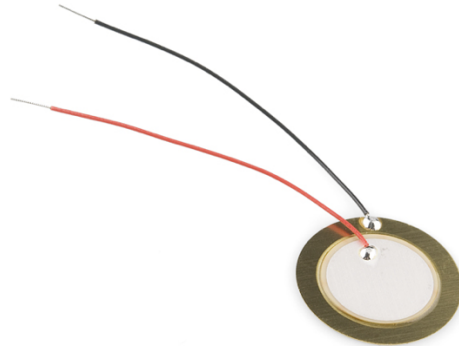


Figura 3.1 Sensor piezoeléctrico 7BB-20-6L0.

Tabla 3.2 Propiedades sensor piezoeléctrico 7BB-20-6L0. Fuente: Murata Manufacturing Co.

Propiedad	Magnitud	Unidad
Frecuencia de Resonancia	6.00	[kHz]
Diámetro Placa	20.00	[mm]
Diámetro Piezoeléctrico	14.00	[mm]
Espesor Placa	0.20	[mm]
Espesor Total	0.42	[mm]

3.1.3 Tarjeta de Adquisición

En la Figura 3.2 se presenta una fotografía del sistema de adquisición utilizado. Este sistema permite digitalizar las señales provenientes de los sensores piezoeléctricos y del matrillo modal. La tarjeta de adquisición es conectada por medio de USB al computador personal, donde se tiene el software de adquisición de datos desarrollado por ECON-GROUP. Con este software es posible ajustar la sensibilidad de las señales, la frecuencia de muestreo y el rango de tiempo de las ventanas rectangulares en las que se toma la señal.



Figura 3.2 Sistema de Adquisición.

Se utiliza el martillo modal presentado en la Figura 3.3 para realizar los impactos sobre la superficie de la estructura. El martillo además permite medir la magnitud de la fuerza de impacto en función del tiempo, ya que posee un dinamómetro. Al igual que los sensores piezoeléctricos, el martillo se conecta a la tarjeta de adquisición de datos.



Figura 3.3 Martillo Modal.

En la Tabla 3.3 se presentan las características de la tarjeta de adquisición, mientras que en la Tabla 3.4 las características del martillo modal.

Tabla 3.3 Propiedades tarjeta de adquisición Econ-MI7016. Fuente: Econ-group, MI7016 Manual of Operation.

Propiedad	Magnitud	Unidad
Número de Canales	16	-
Resolución	24-bit	-
Voltaje	± 10	[VPEAK]
Precisión de Frecuencia	0.0075	[%]

Tabla 3.4 Propiedades martillo modal LC-01A. Fuente: Global Sensor Technology.

Propiedad	Magnitud	Unidad
Tipo de Transductor	Fuerza	-
Material Punta	Nylon	-
Sensibilidad	4	[pC/N]
Máxima Fuerza de Choque	2	[kN]
Diámetro Cabeza	16	[mm]
Masa Cabeza	40	[g]
Masa Cuerpo	100	[g]

3.2 ETAPAS DE TRABAJO

En la figura 3.6 se presenta un diagrama explicativo del método de identificación de impactos desarrollado en este trabajo.

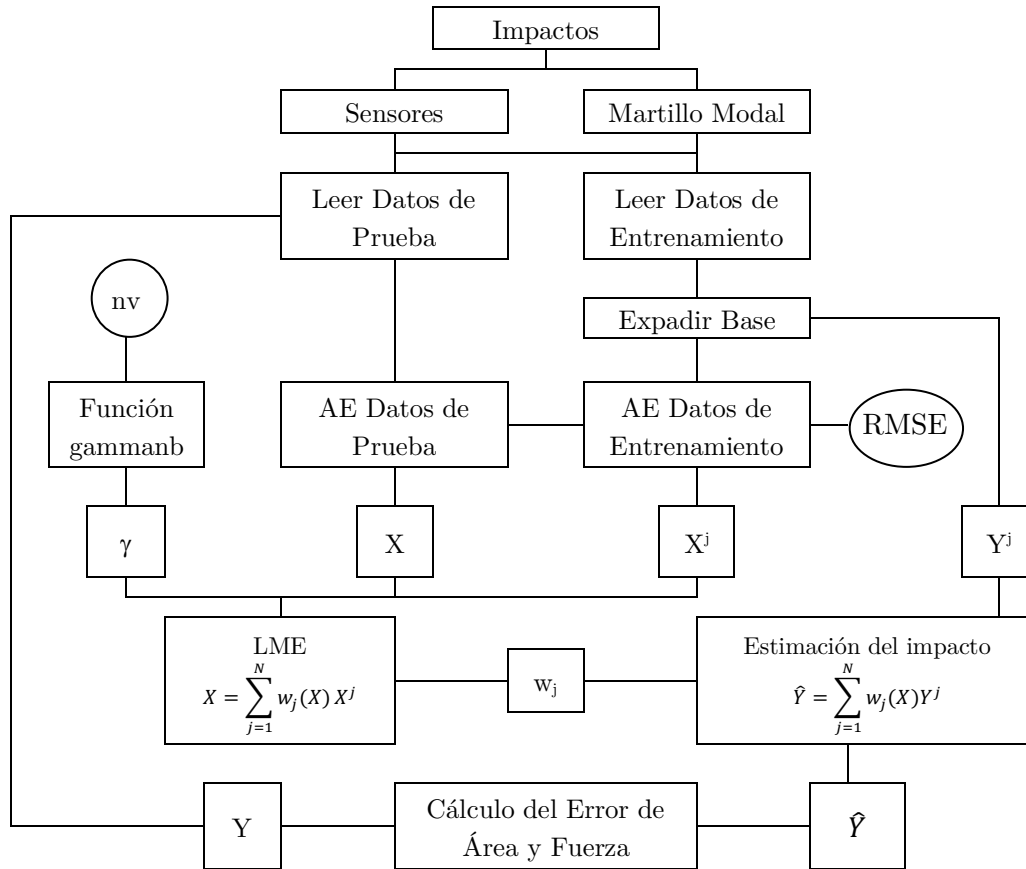


Figura 3.4 Diagrama de flujo de estrategia de identificación de impactos.

A continuación se detallan las etapas de trabajo.

3.2.1 Lectura de datos

Los impactos realizados sobre la superficie son tomados por los sensores y el martillo modal, y enviados a la tarjeta de adquisición para ser transformados en señales de voltaje y fuerza con respecto al tiempo. Luego, estas son tomadas por el software de procesamiento de señales, para finalmente ser analizado por el algoritmo de identificación de impactos generado.

Para detectar la ubicación y magnitud de los impactos es necesario implementar dos sets de datos. En primer lugar se debe implementar un set de datos de entrenamiento, con el cual se caracteriza la respuesta del sistema. Para esto, se utiliza una malla de puntos distribuida uniformemente sobre la superficie del cilindro, donde en cada punto se realiza un único impacto en base al set de datos de entrenamiento.

Al realizar un impacto, un sensor piezoeléctrico adherido a la estructura recibe las ondas de esfuerzos superficiales, los cuales entregan estas ondas en forma de una señal de voltaje. Esta señal contiene una gran cantidad de datos, por lo que se extrae la parte de la señal que entrega información relevante sobre el impacto. Para esto, en primer lugar se aplica la Transformada de Hilbert, obteniendo la envolvente de la señal. Luego, se determina el instante en el que ocurre el impacto usando la señal de fuerza tomada por el martillo modal, y se extrae una sección con los 500 datos anteriores y los 11499 datos posteriores al impacto. Esto entrega, para cada impacto, un total de 12000 datos.

La etapa de lectura de datos concluye al obtener los vectores X^i con la señal de voltaje extraído de la señal de los impactos pertenecientes a la base de entrenamiento para cada sensor. Mientras los vectores Xp^i almacenan la señal de voltaje de los impactos de prueba para cada sensor. Por otro lado, se crean las matrices referentes a la ubicación de los golpes. Se define Y^i , matriz que almacena la información de ubicación y magnitud de impacto de los datos de entrenamiento. Y finalmente se define Y , matriz que almacena la información de ubicación y magnitud de impacto de los datos de prueba.

La implementación de esta etapa en el entorno Matlab, se presenta en el Anexo 7.A para los datos de entrenamiento, y en el Anexo 7.B para los datos de prueba.

3.2.2 Expansión de la base de entrenamiento

Dado que la estructura presenta la propiedad de ser lineal, es posible determinar la respuesta del sistema frente a impactos de diferentes magnitudes. Esto se obtiene al multiplicar la respuesta medida por un factor de escala. La expansión de la base de datos de respuesta se realiza únicamente a los datos de entrenamiento, dado que los datos de prueba sólo se utilizan para evaluación.

La expansión se obtiene mediante la expresión presentada en la ecuación 3.1.

$$X^a = X^j \cdot \frac{Y_3^a}{Y_3^j} \quad (3.1)$$

Donde:

Y_3^a : fuerza definida arbitrariamente.

Y_3^j : fuerza medida para el impacto j.

X^j : datos de respuesta obtenidos por un sensor para un impacto j.

X^a : respuesta de base expandida el mismo impacto, pero con magnitud Y_3^a .

La implementación de esta etapa, se presenta en el Anexo 7.C.

3.2.3 Aplicación del AE

Una vez obtenidas la base de datos de prueba y la base expandida de datos de entrenamiento, se entra en la etapa del Autoencoder (AE) con el fin de extraer el espacio latente de los datos, el cual concentra la mayor cantidad de información relevante de los impactos. Es importante normalizar los datos de entrenamiento y prueba, dado que cada impacto presenta amplitudes de vibraciones diferentes con respecto a la magnitud del impacto recibido, y con la ubicación del impacto. Por esto se aplica una normalización de valores entre 0 y 1 para cada impacto.

Para implementar el AE, se utiliza la biblioteca de código abierto Tensorflow en el ambiente de programación Python. Tensorflow fue desarrollado por Google para satisfacer necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar patrones y correlaciones específicos [5]. Gracias a esto, es posible generar un AE con gran capacidad de entrenamiento y reconstrucción de datos.

Con respecto a la reducción a realizar por el encoder, al revisar trabajos similares de extracción de propiedades características de los datos utilizando PCA [2], para definir la dimensión del espacio latente, se prueba el utilizar el usar un espacio latente de largo 100 y 200. Se decide definir la dimensión del espacio latente en 200, ya que entrega un RMSE promedio menor, y el tiempo de procesamiento también disminuye. Luego, se aplica una reducción de cuatro capas con reducción homogénea a la señal de entrada (Input). Estas dimensiones se presentan a continuación en la Tabla 3.5.

Tabla 3.5 Dimensiones capas de reducción y espacio latente del AE.

Capa	Input	Capa 1	Capa 2	Capa 3	Capa 4	Espacio Latente
Dimensión	12000	9640	7280	4920	2560	200

Se deben encontrar los valores de tasa de aprendizaje y épocas de entrenamiento que optimizan la reconstrucción de la matriz inicial. Para esto, se comienza con una configuración

estándar, para luego comenzar a mover estas variables hasta obtener la combinación más eficiente en función del RMSE de reconstrucción, calculado a partir de la ecuación 2.14.

Una vez elegidas todas las variables relevantes para el entrenamiento del AE, se aplica la misma configuración a los datos entregados por cada sensor. Los cuales son concatenados posteriormente para ser la matriz de entrada al LME.

La implementación de esta etapa se presenta en el Anexo 7.D.

3.2.4 Aplicación de LME

Una vez obtenido el espacio latente de los datos se procede a aplicar el método LME. Este proceso utiliza un software basado en un código de máxima entropía llamado “Maxent Basis Functions for Matlab”, desarrollado por el proesor co-guía de este trabajo, Alejandro Ortiz-Bernardin del Departamento de Ingeniería Mecánica de la Universidad de Chile. Este cuenta con algoritmos que permiten utilizar el principio de máxima entropía dentro del entorno Matlab.

Como se observa en el diagrama presentado en la Figura 3.4, los datos de entrada para este metodo son la matriz X^j que corresponde al espacio latente de los datos de entrenamiento, la matriz X que corresponde al espacio latente de los datos de prueba y el parámetro γ , el cual se determina usando la ecuación 2.32. El parámetro γ controla la influencia que poseen los elementos de la base al realizar la estimación de los valores de \hat{Y} (matriz resultante de impactos identificados por el algoritmo). Un valor de γ alto implica que la expresión $\|X^j - X\|$ tenga poca influencia, es decir, los elementos de la base que se encuentran a mayor distancia euclidiana aportan cada vez menos. Por lo tanto, para determinar el valor de γ , se utiliza la función Gammanb (perteneciente al software LME) la cual lo calcula en función del número de vecinos (denotado como nv).

Tanto el número de vecino (nv), como los parámetros de entrenamiento del Autoencoder influyen en el desempeño del algoritmo, por lo que se debe realizar un análisis para determinar su valor óptimo.

Finalmente, el método LME entrega los pesos w_j , con los que se calcula el impacto estimado \hat{Y} . Además, la implementación de esta etapa se presenta en el Anexo 7.E.

3.2.5 Determinación de Error

Para determinar el desempeño del algoritmo completo, se utilizan las funciones de error presentadas en las ecuaciones 2.50, 2.51, 2.52 y 2.55. Lo que entrega las magnitudes de error de localización de los impactos. Además de la determinación del mínimo error de Identificación de impactos normalizado, usando la ecuación 2.56. La implementación de esta etapa se presenta en el Anexo 7.F.

3.3 PRUEBAS REALIZADAS

Dado que se tiene una estructura diferente a las referencias bibliográficas consultadas, es importante analizar como influyen las condiciones de borde (fijaciones) en el resultado final del desempeño del algoritmo. Es por esto que se definen dos configuraciones diferentes, una en la que la estructura se dispone de forma vertical sobre una superficie plana (Figura 3.5 Izquierda). Y una segunda donde se suspende el cilindro utilizando una cuerda (Figura 3.5 Derecha).



Figura 3.5 Fotografías configuraciones utilizadas. (Izq.) Vertical. (Der.) Suspendida.

En ambos casos, se aplica un mallado homogéneo en la superficie del cilindro. En las Figuras 3.6 y 3.7 se presentan las coordenadas $\hat{\theta}$ y \hat{z} de cada uno de los impactos de entrenamiento para cada configuración, mientras que las coordenadas en las que se encuentran los sensores están en rojo. Además, se realizan 50 impactos al azar (iguales en ambas configuraciones), para obtener los datos de impactos de prueba. También se presentan las ubicaciones de los impactos de prueba.

3.3.1 Mediciones impactos de entrenamiento configuración vertical

Para la construcción de la base de datos de entrenamiento, se realizan 120 impactos en total, los cuales se distribuyen de forma homogénea en la superficie del cilindro. Gracias a la expansión de la base, los datos de entrenamiento son aumentados a 1560 impactos, con magnitudes entre 1 y 50 [N]. En el Anexo 7.H se presenta el detalle de las coordenadas y magnitudes de cada impacto en la base de entrenamiento.

En la Figura 3.6 se presentan los impactos de la base de entrenamiento, la posición de los sensores, y la ubicación de la zona de contacto con la base.

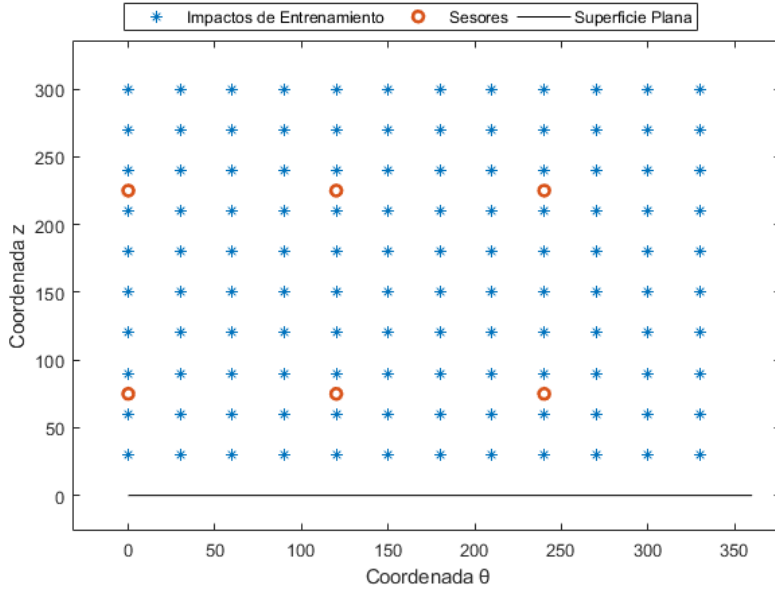


Figura 3.6 Ubicación impactos de entrenamiento, configuración vertical.

3.3.1 Mediciones impactos de entrenamiento configuración suspendida

Para la construcción de la base de datos de entrenamiento, se realizan 132 impactos en total, los cuales se distribuyen de forma homogénea en la superficie del cilindro. Gracias a la expansión de la base, los datos de entrenamiento son aumentados a 1716 impactos, con magnitudes entre 1 y 50 [N]. En la Figura 3.7 se presentan los impactos de la base de entrenamiento, la posición de los sensores, y la ubicación de la zona de contacto con la cuerda.

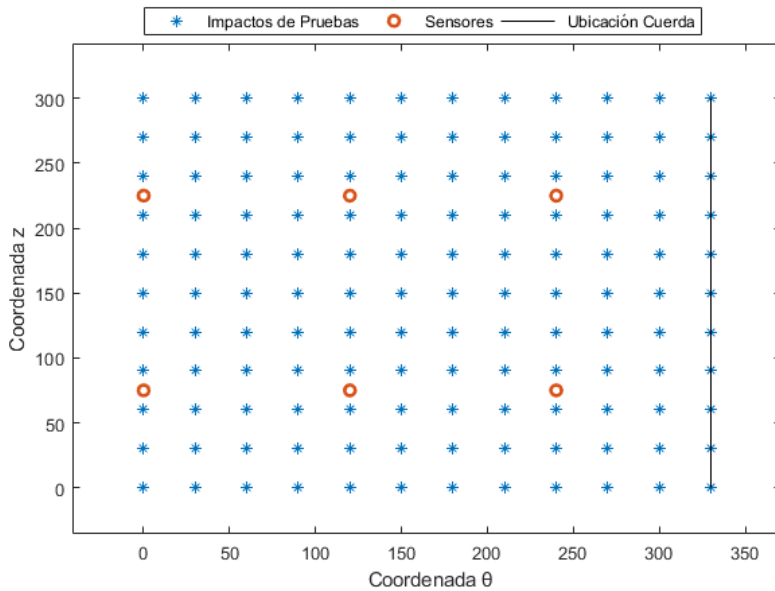


Figura 3.7 Ubicación impactos de entrenamiento, configuración suspendida.

En el Anexo 7.I se presenta la tabla con el detalle de las coordenadas y magnitudes de cada impacto en la base de entrenamiento para esta configuración.

3.3.2 Mediciones datos de prueba

Para probar el algoritmo, es necesario tener una base de datos de prueba. Para esto, se realiza una medición de 50 impactos distribuidos de manera aleatoria sobre la superficie del cilindro. La Figura 3.8 presenta la ubicación de los impactos de prueba.

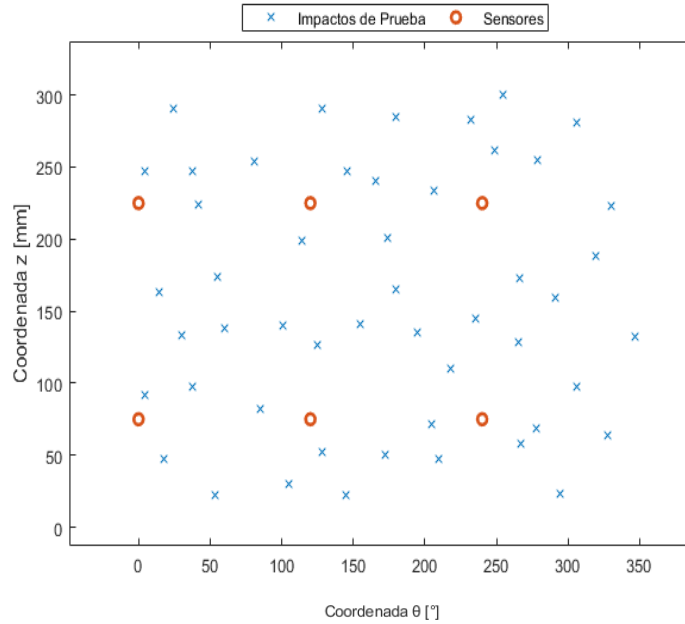


Figura 3.8 Ubicación impactos de prueba.

Esta medición de prueba se realiza en ambas configuraciones. El Anexo 7.J tiene las coordenadas y magnitud de los datos de entrenamiento para la configuración vertical, mientras que el Anexo 7.K presenta las coordenadas y magnitud para la configuración suspendida.

4 RESULTADOS

4.1 RESULTADOS ESTRUCTURA VERTICAL

4.1.1 Resultados Entrenamiento Autoencoder

Luego de cargar los archivos y aplicar la expansión de la base de entrenamiento, se deben seleccionar los parámetros de tasa de aprendizaje y número de épocas que minimizan el RMSE del AE. Al analizar el desempeño del algoritmo para variados valores de estos parámetros, se obtienen los gráficos presentados en las Figuras 4.1 y 4.2.

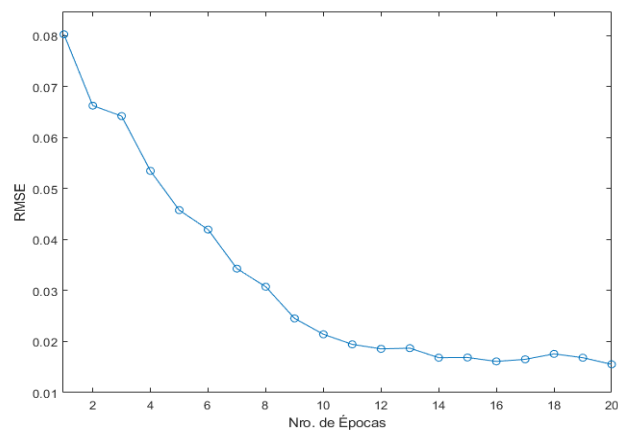


Figura 4.1 Costo AE vs. número de épocas.

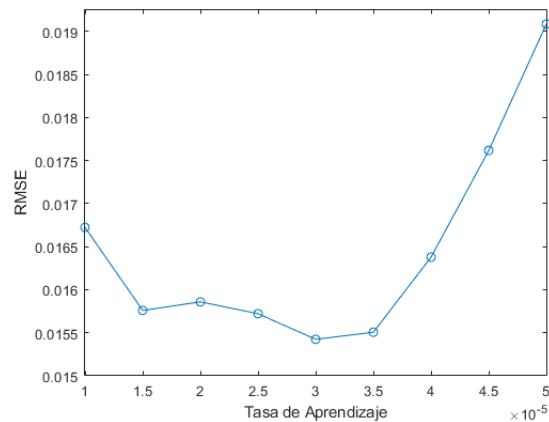


Figura 4.2 Costo AE vs. tasa de aprendizaje.

A partir de esto, se selecciona una tasa de aprendizaje de $3E-5$, y un número de épocas de 20. La Tabla 4.1, presenta los resultados de RMSE para cada sensor al aplicar esta configuración. También se presenta el RMSE promedio de la prueba.

Tabla 4.1 Resultados AE optimizado, estructura vertical.

Sensor	Tasa de Aprendizaje	Épocas	RMSE
1	3.00E-05	20	0.015424
2	3.00E-05	20	0.027882
3	3.00E-05	20	0.015422
4	3.00E-05	20	0.028565
5	3.00E-05	20	0.016539
6	3.00E-05	20	0.022066
RMSE Promedio			0.020983

4.1.2 Resultados LME

Luego de obtener el espacio latente mediante el AE, se debe evaluar el desempeño del LME para variados valores de n_v (nro. de vecinos), con el fin de seleccionar su valor óptimo para el algoritmo en la presente configuración.

A continuación, se presenta la evaluación del error de área y fuerza según el número de vecinos, en las Figuras 4.3 y 4.4 respectivamente.

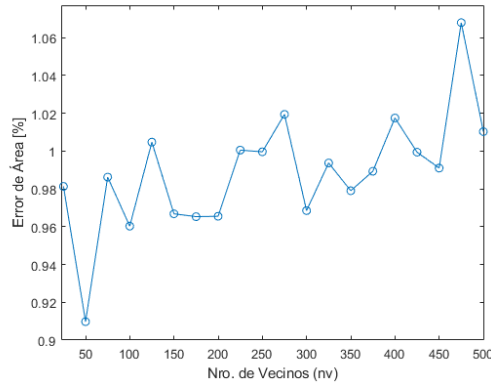


Figura 4.3 Error de área en función del número de vecinos.

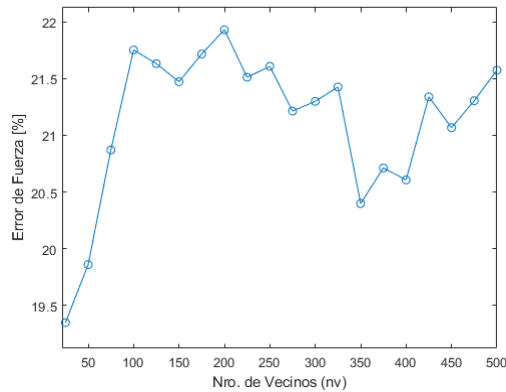


Figura 4.4 Error de fuerza en función del número de vecinos.

Es posible observar que mientras que el error de área encuentra su valor mínimo en los 50 vecinos, el error de área se minimiza en los 10 vecinos. Por lo tanto, para seleccionar el número de vecinos que entregue los mejores resultados, se debe calcular el error de identificación de impactos normalizado utilizando la ecuación 3.56.

En la Figura 4.5, se presenta el error de identificación según el número de vecinos.

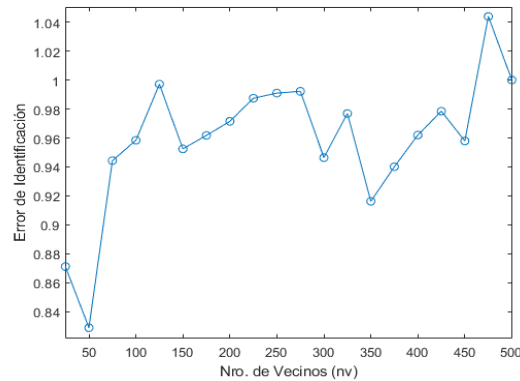


Figura 4.5 Error de identificación de impacto normalizado, en función del número de vecinos.

Considerando este análisis, se selecciona el número de vecino $nv = 50$, ya que con esto se obtiene el menor error de Identificación en la presente configuración. Con esto, se obtienen los resultados óptimos del algoritmo, presentados en la Tabla 4.2.

Tabla 4.2 Evaluación algoritmo, estructura vectical.

Error	Error de Área	Error de Fuerza	Error Promedio $\hat{\theta}$	Error Promedio \hat{z}
Unidad	[%]	[%]	[°]	[mm]
Valor	0.9098	19.8599	30.6104	32.1017

Estos resultados son presentados de manera gráfica en las figuras 4.6, 4.7, 4.8 y 4.9.

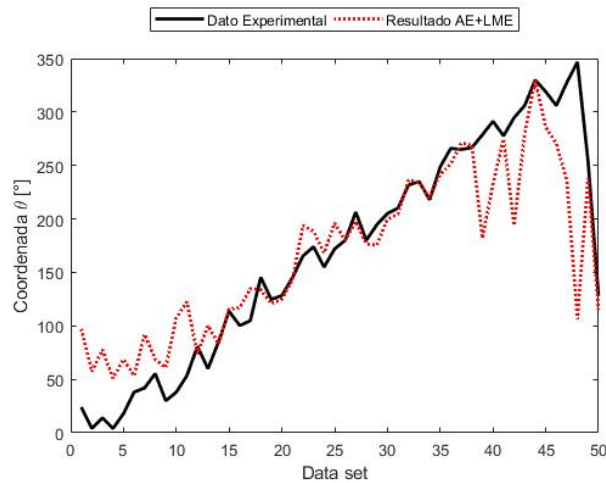


Figura 4.6 Resultados AE+LME estructura vertical, coordenada θ .

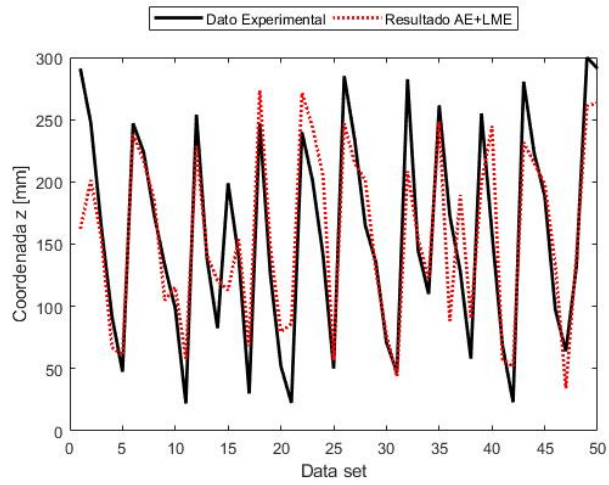


Figura 4.7 Resultados AE+LME estructura vertical, coordenada z.

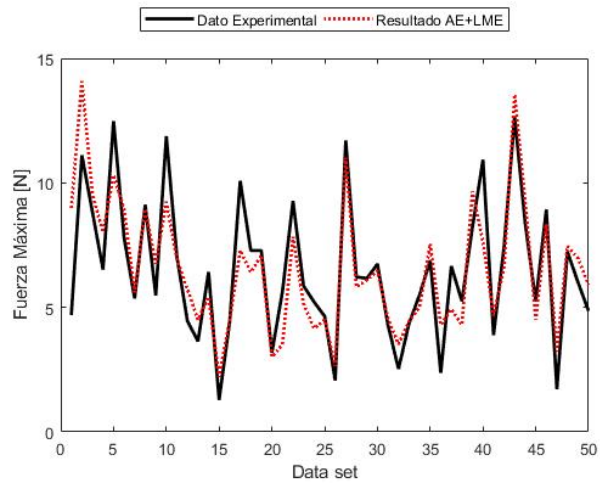


Figura 4.8 Resultados AE+LME estructura vertical, magnitud fuerza de impacto.

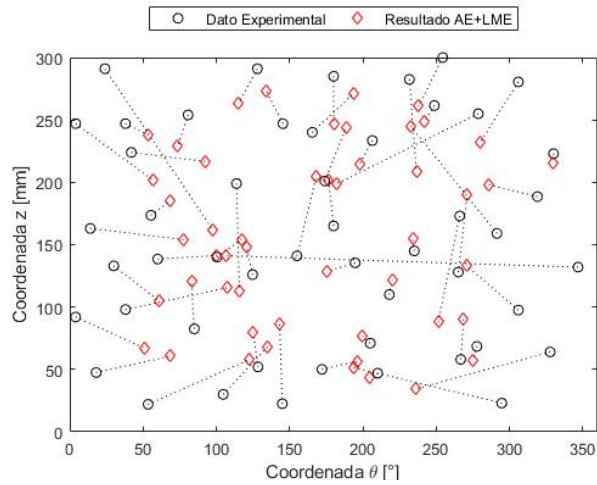


Figura 4.9 Resultados AE+LME estructura vertical, ubicación de impactos en la superficie del cilindro.

4.2 RESULTADOS ESTRUCTURA SUSPENDIDA

4.2.1 Resultados Entrenamiento Autoencoder

El set de datos para la configuración de la estructura suspendida tiene un tamaño 132×12000 expandidos a 1716×12000 , los cuales son reducido por el AE a un espacio latente de dimensión 1716×200 .

Para esta configuración también se deben seleccionar los parámetros de tasa de aprendizaje y número de épocas que minimizan el RMSE del AE. Al analizar el desempeño del algoritmo para variados valores de estos parámetros, se obtienen los gráficos presentados en las Figuras 4.10 y 4.11.

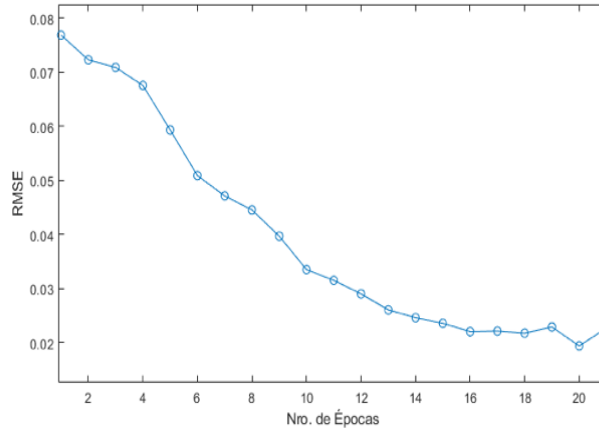


Figura 4.10 Costo AE vs. número de épocas.

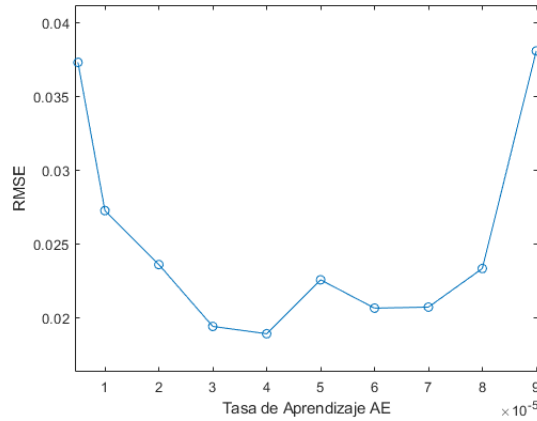


Figura 4.11 Costo AE vs. tasa de aprendizaje.

A partir de esto, se define una tasa de aprendizaje de $4E-5$, y un número de épocas de 20. La Tabla 4.3, presenta los resultados de RMSE para cada sensor al aplicar esta configuración. También se presenta el RMSE promedio de la prueba.

Tabla 4.3 Resultados AE optimizado, estructura suspendida.

Sensor	Tasa de Aprendizaje	Épocas	RMSE
1	4.00E-05	20	0.017633
2	4.00E-05	20	0.017856
3	4.00E-05	20	0.018998
4	4.00E-05	20	0.019629
5	4.00E-05	20	0.018095
6	4.00E-05	20	0.016802
RMSE Promedio			0.018169

4.2.2 Resultados LME

Luego de obtener el espacio latente mediante el AE, se debe evaluar el desempeño del LME para variados valores de nv (nro. de vecinos), con el fin de seleccionar su valor óptimo para el algoritmo en la presente configuración.

A continuación, se presenta la evaluación del error de área y fuerza según el número de vecinos, en las Figuras 4.12 y 4.13 respectivamente.

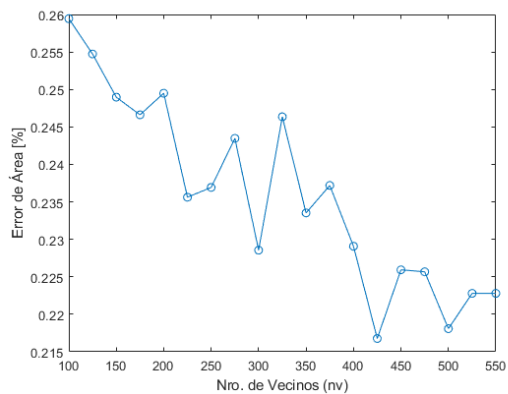


Figura 4.12 Error de área en función del número de vecinos.

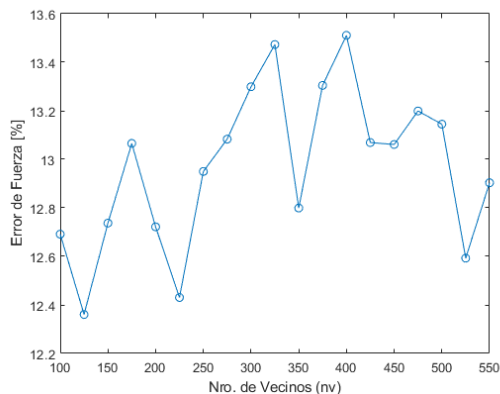


Figura 4.13 Error de fuerza en función del número de vecinos.

Es posible observar que mientras que el error de área encuentra su valor mínimo en los 425 vecinos, el error de área se minimiza en los 125 vecinos. Por lo tanto, para seleccionar el número de vecino que entregue los mejores resultados, se debe calcular el error de identificación de impactos normalizado utilizando la ecuación 2.56.

En la Figura 4.14, se presenta el error de identificación según el número de vecinos.

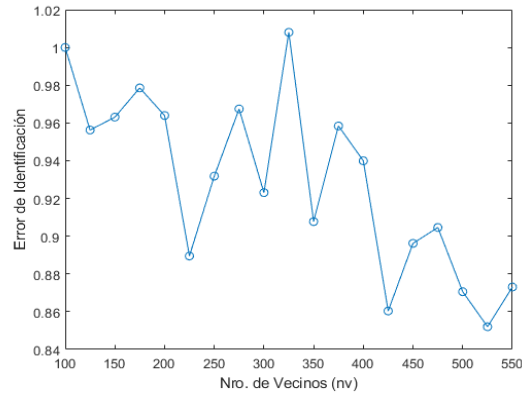


Figura 4.14 Error de identificación de impacto normalizado, en función del número de vecinos.

Considerando este análisis, se selecciona el número de vecinos que entrega el menor error de identificación de impactos, $nv = 525$. Con este nv , se obtienen los resultados de desempeño del algoritmo para esta configuración, presentados a continuación en la tabla 4.4.

Tabla 4.4 Evaluación algoritmo, estructura suspendida.

Error	Error de Área	Error de Fuerza	Error Promedio $\hat{\theta}$	Error Promedio \hat{z}
Unidad	[%]	[%]	[°]	[mm]
Valor	0.2228	12.5926	32.1056	7.4912

Estos resultados son presentados de manera gráfica en las Figuras 4.15 a la 4.18.

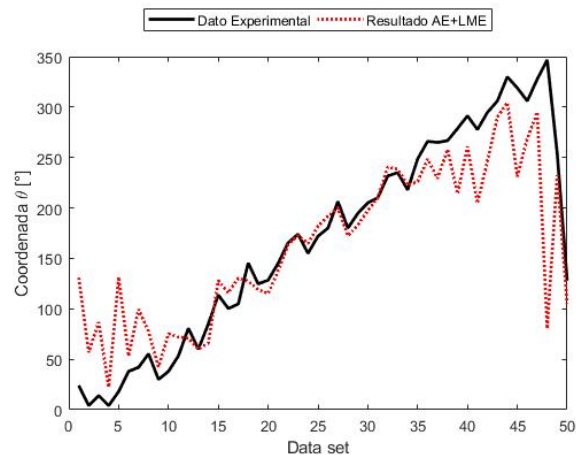


Figura 4.15 Resultados AE+LME estructura suspendida, coordenada θ .

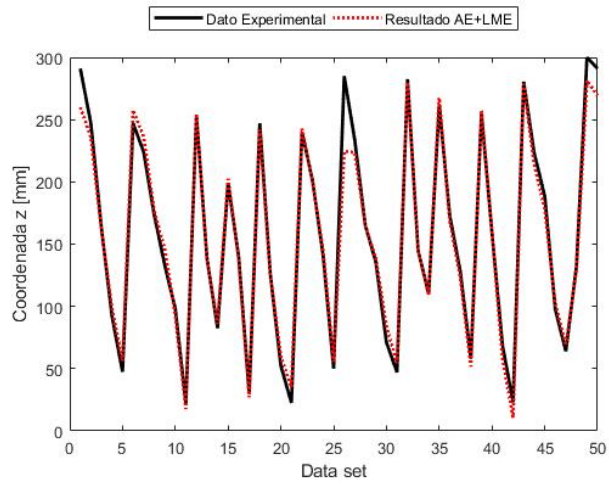


Figura 4.16 Resultados AE+LME estructura vertical, coordenada z.

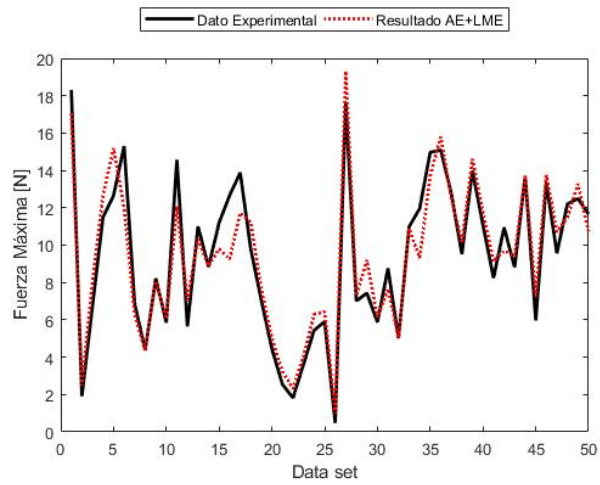


Figura 4.17 Resultados AE+LME estructura suspendida, magnitud fuerza de impacto.

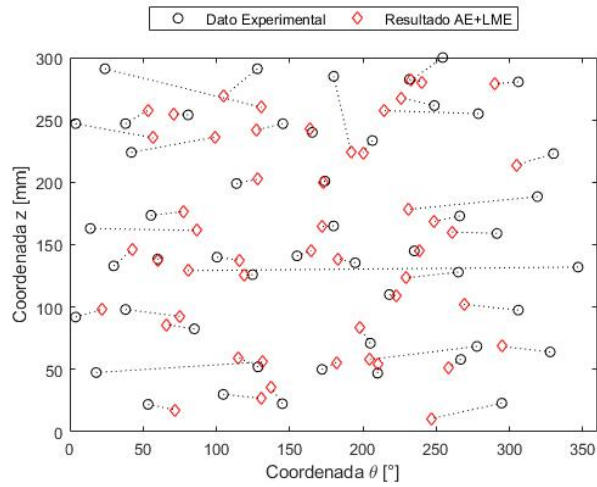


Figura 4.18 Resultados AE+LME estructura suspendida, ubicación de impactos en la superficie del cilindro.

4.3 DISCUSIÓN

SOBRE EL DESEMPEÑO DEL AUTOENCODER DESARROLLADO

Para analizar el desempeño del AE generado, es necesario analizar tanto la selección de variables relevantes para la construcción y trabajo del algoritmo; como el qué tan bien el espacio latente obtenido representa las características de los datos de entrada.

Al observar la sensibilidad de los datos a la variación del número de épocas de entrenamiento, se selecciona trabajar con 20 épocas para ambas configuraciones. Además, en las Figuras 4.1 y 4.10 es posible observar que, para ambas configuraciones, el RMSE presenta la tendencia de disminuir al aumentar el número de épocas.

Según la teoría es esperable que el algoritmo sea capaz de aprender más fácilmente si se le permite hacer más recursiones (épocas en este caso), hasta que llegue un punto en el que el RMSE se estanca e incluso puede comenzar a subir, dado que estas recursiones extras no resultan útiles y pueden ser contraproducentes para el desempeño del AE. Es por esto que se considera correcta la selección de 20 épocas de trabajo para el algoritmo.

Por otro lado, con respecto a la selección de la tasa de aprendizaje se hace un análisis independiente para cada configuración. Cuando las Figuras 4.2 y 4.11, se obtienen los valores para la tasa de entrenamiento que minimizan el RMSE de la prueba. En la Tabla 4.5 se presenta el resumen de esta selección, donde se observa que la tasa de aprendizaje para la configuración vertical es menor que la necesaria para la suspendida, esto tiene sentido al tomar en cuenta que la base de datos de entrenamiento que se le entrega al AE tiene una dimensión menor, y por lo tanto su reducción es también menor.

Como se indica en la teoría [3], en general una tasa de aprendizaje baja se resume en un bajo RMSE. Ya que de esta forma el algoritmo toma datos en bajas cantidades para aprender sus características con mayor precisión. Aunque también se debe considerar no utilizar una tasa demasiado pequeña, puesto que esto aumenta el tiempo de trabajo sin aportar en un baja del RMSE. Es por esto que el análisis de sensibilidad realizado permite justificar correctamente la selección realizada.

Tabla 4.5 Comparación desempeño AE.

Configuración	Épocas	Tasa de Aprendizaje	RMSE Promedio
Vertical	20	3.00E-05	0.020983
Suspendida	20	4.00E-05	0.018169

Con esto, además de la Figura 2.7, podemos concluir que el AE es muy eficiente al reconstruir la señal de entrada.

SOBRE LA DETERMINACIÓN DEL NUMERO DE VECINOS EN EL LME

Ya que se elije dejar fija la reducción de los datos realizada por el AE, sólo es necesario buscar el número de vecinos que minimice el error de identificación del algoritmo para cada configuración. En la Tabla 4.6 se presenta el resumen de los valores de n_v elegidos y el error de identificación obtenido en cada caso. Se observa que, si bien el valor de n_v es bastante, el error de identificación es bastante similar. Además, el nro. de vecinos de referencia para el cálculo del error de identificación (n_{v_0} para utilizar la ecuación 2.56), también es casi contrario. Esto implica que la evolución del error de identificación según el valor de n_v es bastante similar en cuanto a forma.

Tabla 4.6 Comparación elección Nro. de vecinos.

Configuración	Nro. de Vecinos	Nro. de Vecinos n_{v_0}	Error de Identificación
Vertical	50	500	0.829004
Suspendida	525	100	0.852037

Para la configuración vertical, que presenta mayores restricciones dadas por el contacto con la base plana, se tiene que el disminuir el valor de n_v entrega mejores resultados, es decir que el algoritmo trabaja mejor tomando una cantidad de datos baja en cada análisis. Lo cual tiene sentido, dado que, al tomar los datos en grupos reducidos, se evita entrar en conflictos relacionados a las fijaciones del sistema, es comparable con el afinamiento de mallados para el análisis de elemento finito en zonas cercanas a un cambio o discontinuidad de la geometría.

En cuanto a la configuración suspendida, que presenta un mayor grado de libertad (dado que solo esta fijo su movimiento en la dirección vertical), se obtiene que el error de identificación disminuye al aumentar el nro. de vecinos. Esto puede implicar, que al tomar una mayor cantidad de datos, se tiene una mejor representación de la estructura como un elemento independiente del exterior.

Por lo tanto, se puede afirmar que el valor de n_v depende de forma importante de las características físicas y de geometría del sistema. Pero que, al mismo tiempo, el algoritmo es ajustable a estas diferencias de configuración.

SOBRE LOS RESULTADOS DE LA CONFIGURACION VERTICAL

Como se menciona anteriormente, para el caso de la estructura en posición vertical la base plana actúa como apoyo simple y por lo tanto aplica una restricción de movimiento al sistema, tanto en la coordenada z como en θ . Además, interfiere en la propagación de las vibraciones provocadas por cada impacto, puede tanto detenerlas o reflejarlas. Por lo tanto, el contacto del cilindro con la base entrega ruido y discontinuidades a las mediciones iniciales, lo que se traduce en un error mayor al momento de identificar los impactos nuevos.

En la Figura 4.6 se puede observar que el algoritmo acierta bastante bien en la coordenada θ en la zona entre los 60° y los 230° aproximadamente. Y destaca el hecho de que en los casos en los que la estimación es errónea, presentan un error muy alto, lo que puede implicar problemas en la geometría o estructura interna del cilindro en estas zonas.

Mientras que la Figura 4.7 presenta que para la coordenada z , el algoritmo reconoce mejor los impactos realizados a una altura cercana a la altura en la que se encuentran los sensores (75 y 225 [mm]), lo cual tiene sentido dado que en estas alturas la señal es medida con una menor interferencia externa.

SOBRE LOS RESULTADOS DE LA CONFIGURACIÓN SUSPENDIDA

Para la configuración con el cilindro suspendido, la estructura se encuentra con mayor libertad de movimiento y al mismo tiempo con una superficie de contacto con un agente externo (la cuerda es este caso) mucho menor en comparación con la estructura en posición vertical. En esta configuración, prácticamente no hay alteraciones a lo largo de la coordenada z , mientras que para θ la cuerda actúa como obstáculo para la propagación uniforme de las vibraciones provocadas por los impactos.

Al observar la Figura 4.15 se tiene un resultado bastante similar al vertical, aunque en este caso la zona donde la estimación es más acertada es más amplia, abarcando desde los 25° hasta los 250° . En este caso se puede atribuir el error en la estimación en la zona restante a la cercanía con la cuerda de soporte, aunque en este caso se esperaría una zona más simétrica, lo que hace continuar con la presunción de la existencia de una zona con discontinuidades geométricas o de estructura interna.

En la Figura 4.16, se presenta que la identificación de la coordenada z es muy acertada, e uniforme a lo largo de todo el cilindro, lo que permite confirmar que el error mayor en la configuración vertical está dado por el contacto con la base.

COMPARACIÓN RESULTADOS AE+LME CON PCA+LME [2]

Si bien la Figura 4.19 hace pensar que el espacio latente entregado por el AE no representa de la mejor manera al set de datos inicial, es necesario hacer un análisis mayor para evaluar correctamente el desempeño del método utilizado.

Al tomar un algoritmo PCA+LME disponible en la literatura [2], ideado para una placa plana y ajustarlo a la geometría del presente trabajo, es posible comparar ambos métodos. En la Figura 4.20 se presenta la detección de impactos obtenida con PCA+LME para cada configuración, utilizando los mismos valores para la reducción y nro. de vecinos elegidos para el algoritmo AE+LME.

Es posible observar que estos resultados son bastante parecidos en cuanto a forma a los obtenidos con AE+LME, dado que para la configuración vertical se mantiene la variabilidad tanto en los resultados de coordenada θ como el los de coordenada z . Mientras que para la configuración se obtienen resultados acertados para la coordenada z , y más error asociado a la coordenada θ .

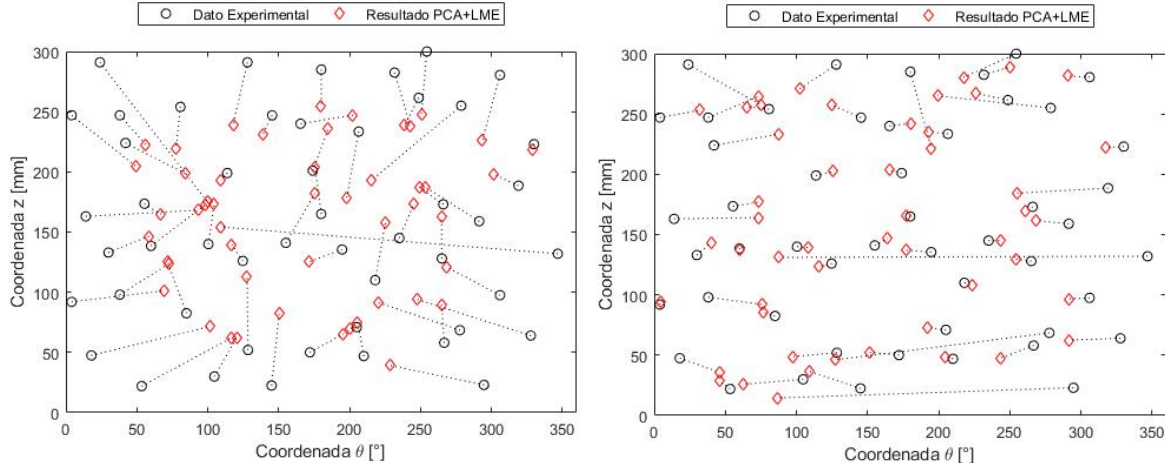


Figura 4.19 Resultados PCA+LME (Izq.) Configuración vertical. (Der.) Configuración suspendida.

En cuanto a una comparación cuantitativa, la Tabla 4.7 presenta los resultados para ambos métodos en cada configuración.

Tabla 4.7 Comparación método AE+LME con PCA+LME.

Configuración	Método	Error de Área	Error de Fuerza	Error Promedio $\hat{\theta}$	Error Promedio \hat{z}
	Unidad	[%]	[%]	[°]	[mm]
Vertical	PCA+LME [1]	0.6752	14.2472	24.5792	29.6653
	AE+LME	0.9098	19.8599	30.6104	32.1017
Suspendida	PCA+LME [1]	0.1867	9.4159	28.3627	7.1124
	AE+LME	0.2228	12.5926	32.1056	7.4912

Para ambas configuraciones el algoritmo PCA+LME entrega mejores resultados, especialmente para la configuración vertical, dado que el error de área disminuye en un 25% y el error de fuerza en un 28%, lo que es una variación considerable. Mientras que para la configuración suspendida el método PCA+LME disminuye el error de área en un 16% y el error de fuerza en un 25%.

Esto permite afirmar que el espacio latente entregado por el AE, si bien permite obtener resultados cualitativamente correctos según las características del problema, debe ser mejorado para aumentar la precisión de la identificación final de coordenadas y magnitudes.

SOBRE LA GEOMETRIA DE LA ESTRUCTURA

Si bien los resultados del método PCE+LME resultan mejores para este problema, al analizar el método AE+LME desarrollado sobre una placa simple, se obtienen los resultados presentados en la Tabla 4.8.

Tabla 4.8 Desempeño algoritmo para placa plana.

Método	Error de Área	Error de Fuerza	Error Promedio x	Error Promedio y
Unidad	[%]	[%]	[mm]	[mm]
PCA+LME	0.0449	10.789	12.932	9.725
AE+LME	0.0346	13.665	10.681	9.062

Para esta geometría bidimensional, el algoritmo AE+LME presenta un error de área 22% menor. Mientras que el error de fuerza sigue siendo mejor al usar el método PCA-LME. Esto demuestra que el algoritmo puede ser ajustado a diferentes geometrías.

SOBRE LA DISCONTINUIDAD EN LA GEOMETRÍA ESTUDIADA

Como se menciona repetidas veces, el aumento del error en la identificación de la coordenada θ en general bajo los 30° y sobre los 250° no es explicado simplemente por el contacto con la cuerda en la configuración suspendida, sino que denota una discontinuidad en la geometría del cilindro utilizado para las mediciones. De hecho, la estructura presenta una costura de soldadura a lo largo de la altura del cilindro (Figura 4.21), ubicada en $\theta = 290^\circ$ aproximadamente. Además de ser una discontinuidad en la geometría, esta costura implica una discontinuidad en la estructura interna del material, la cual puede inferir en la propagación de las vibraciones generadas por los impactos.



Figura 4.20 Fotografía costura presente al interior del cilindro.

5 CONCLUSIONES

Este trabajo presenta la elaboración, aplicación y evaluación de un método de identificación de impactos que utiliza un Autoencoder para extraer el espacio latente de los datos de impactos reales, para luego utilizar una aproximación lineal del Principio de Máxima Entropía para reconocer y reconocer la ubicación y magnitud de impactos de prueba.

Se desarrollan dos configuraciones para permitir una evaluación más completa, una con la estructura elegida (un cilindro metálico) se posiciona verticalmente sobre una superficie plana, y otro donde se suspende el cilindro por medio de una cuerda. Para ambos casos, se realizan mediciones de entrenamiento y prueba. En general, es posible afirmar que el desempeño del método generado es mejor en la configuración suspendida, dado que en esta la estructura se encuentra con menores perturbaciones externas. Mientras que la configuración vertical presenta una importante zona de contacto con la base, la cual influye en la propagación de ruido en las mediciones.

El desempeño del AE en el algoritmo es, por un lado, bastante correcta dado que su capacidad de reconstruir la señal inicial desde el espacio latente es buena. Pero por otro lado, al analizar la estructura del espacio latente, y los resultados finales de método comparado con otro que utiliza PCA, si bien se obtienen resultados similares en forma, se puede mejorar en cuanto a precisión.

El análisis del comportamiento de la etapa LME según el número de vecinos a utilizar resulta reveladora sobre la interferencia de las condiciones de borde del problema, dado que la configuración vertical se desempeña mejor al tomar un número reducido de datos, puesto que de esta manera evita las dispersiones generadas por la interferencia de la zona de contacto con la base. Por otro lado, la configuración suspendida se comporta más como un conjunto por lo que se desempeña mejor con un nro. de vecinos mayor.

Al comparar el desempeño del método en una placa plana se obtiene que el error de área disminuye, lo que implica que el algoritmo puede ser ajustado y aplicado a diversas geometrías.

Además de la superficie de contacto con la base, otro elemento que produce un aumento en el error de predicción es la presencia del cordón de soldadura a lo largo de la altura del cilindro. Esta discontinuidad es reconocida por todas las pruebas realizadas, con ambas configuraciones y con ambos métodos utilizados.

Finalmente, es posible afirmar que el método desarrollado en general cumple su función de identificar la ubicación y magnitud de los impactos efectuados a la estructura. Sin embargo, es necesario trabajar en la precisión de este.

6 BIBLIOGRAFÍA

- [1] Sanchez, N., Meruane, V., & Ortiz-Bernardin, A. (2016). A novel impact identification algorithm based on a linear approximation with maximum entropy. *Smart Materials and Structures*, 25(9), 095050.
- [2] Meruane, V., Véliz, P., López Droguett, E., & Ortiz-Bernardin, A. (2017). Impact location and quantification on an aluminum sandwich panel using principal component analysis and linear approximation with maximum entropy. *Multidisciplinary Digital Publishing Institute, Entropy Journal*, 2017.
- [3] Géron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. " O'Reilly Media, Inc.", Chapter 15.
- [4] Fu, H., & Xu, Q. (2013). Locating impact on structural plate using principal component analysis and support vector machines. *Mathematical Problems in Engineering*, 2013.
- [5] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Kudlur, M. (2016, November). Tensorflow: a system for large-scale machine learning. In *OSDI* (Vol. 16, pp. 265-283).
- [6] Ratnaparkhi, A. (1997). A simple introduction to maximum entropy models for natural language processing. *IRCS Technical Reports Series*, 81.
- [7] Meruane, V., & Ortiz-Bernardin, A. (2015). Structural damage assessment using linear approximation with maximum entropy and transmissibility data. *Mechanical Systems and Signal Processing*, 54, 210-223.
- [8] Véliz, P., (2016). Identificación de impactos en una placa compuesta utilizando el principio de máxima entropía y análisis de componentes principales. Departamento de Ingeniería Mecánica, Universidad de Chile, 2016.
- [9] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, 689-696.
- [10] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504-507.
- [11] Haywood, J., Coverley, P. T., Staszewski, W. J., & Worden, K. (2004). An automatic impact monitor for a composite panel employing smart sensor technology. *Smart Materials and Structures*, 14(1), 265.

7 ANEXOS

7.A CÓDIGO MATLAB – LECTURA DE DATOS DE ENTRENAMIENTO

```
clear all;
close all;
clc

load Yi

for i = 1:132

    fuerza = ['Cony\p' num2str(i) '\Position_E1.txt'];
    sensor1 = ['Cony\p' num2str(i) '\Position_R2.txt'];
    sensor2 = ['Cony\p' num2str(i) '\Position_R3.txt'];
    sensor3 = ['Cony\p' num2str(i) '\Position_R4.txt'];
    sensor4 = ['Cony\p' num2str(i) '\Position_R5.txt'];
    sensor5 = ['Cony\p' num2str(i) '\Position_R6.txt'];
    sensor6 = ['Cony\p' num2str(i) '\Position_R7.txt'];

    fid1 = fopen(fuerza);
    C1 = textscan(fid1, '%f %f', 'headerlines', 15);
    fclose(fid1);

    fid2 = fopen(sensor1);
    C2 = textscan(fid2, '%f %f', 'headerlines', 15);
    fclose(fid2);

    fid3 = fopen(sensor2);
    C3 = textscan(fid3, '%f %f', 'headerlines', 15);
    fclose(fid3);

    fid4 = fopen(sensor3);
    C4 = textscan(fid4, '%f %f', 'headerlines', 15);
    fclose(fid4);

    fid5 = fopen(sensor4);
    C5 = textscan(fid5, '%f %f', 'headerlines', 15);
    fclose(fid5);

    fid6 = fopen(sensor5);
    C6 = textscan(fid6, '%f %f', 'headerlines', 15);
    fclose(fid6);

    fid7 = fopen(sensor6);
    C7 = textscan(fid7, '%f %f', 'headerlines', 15);
    fclose(fid7);

    %localizar peaks de fuerza
    [pks,locs] = findpeaks(abs(C1{1,2}), 'minpeakheight', max(abs(C1{1,2}))/4);

    Yi(3,i) = max(pks);

    pos = locs(1);

    %datos puros
    Xd1(:,i) = detrend(C1{1,2});
```

```

Xd2(:,i) = detrend(C2{1,2});
Xd3(:,i) = detrend(C3{1,2});
Xd4(:,i) = detrend(C4{1,2});
Xd5(:,i) = detrend(C5{1,2});
Xd6(:,i) = detrend(C6{1,2});
t(:,i) = C2{1,1};

X1(:,i) = C1{1,2};
X2(:,i) = C2{1,2};
X3(:,i) = C3{1,2};
X4(:,i) = C4{1,2};
X5(:,i) = C5{1,2};
X6(:,i) = C6{1,2};

%transformada de hilbert
C1{1,2} = abs(hilbert(detrend(C1{1,2})));
C2{1,2} = abs(hilbert(detrend(C2{1,2})));
C3{1,2} = abs(hilbert(detrend(C3{1,2})));
C4{1,2} = abs(hilbert(detrend(C4{1,2})));
C5{1,2} = abs(hilbert(detrend(C5{1,2})));
C6{1,2} = abs(hilbert(detrend(C6{1,2})));
C7{1,2} = abs(hilbert(detrend(C7{1,2})));

%hilbert sin ventana de tiempo
Xh1(:,i) = C1{1,2};
Xh2(:,i) = C2{1,2};
Xh3(:,i) = C3{1,2};
Xh4(:,i) = C4{1,2};
Xh5(:,i) = C5{1,2};
Xh6(:,i) = C6{1,2};

%ventana de tiempo
Xi1(:,i) = C2{1,2}(pos-500:pos+11500-1);
Xi2(:,i) = C3{1,2}(pos-500:pos+11500-1);
Xi3(:,i) = C4{1,2}(pos-500:pos+11500-1);
Xi4(:,i) = C5{1,2}(pos-500:pos+11500-1);
Xi5(:,i) = C6{1,2}(pos-500:pos+11500-1);
Xi6(:,i) = C7{1,2}(pos-500:pos+11500-1);
force(:,i) = C1{1,2}(pos-500:pos+11500-1);

tiempo(:,i) = C2{1,1}(pos-500:pos+11500-1);

end

save Xi1 Xi1;
save Xi2 Xi2;
save Xi3 Xi3;
save Xi4 Xi4;
save Xi5 Xi5;
save Xi6 Xi6;
save force force;
save tiempo tiempo;
save Yi Yi

```

7.B CÓDIGO MATLAB – LECTURA DE DATOS DE PRUEBA

```
clear all;
close all;
clc

load Yt

for i = 1:50

    fuerza = ['Cony\t' num2str(i) '\Position_E1.txt'];
    sensor1 = ['Cony\t' num2str(i) '\Position_R2.txt'];
    sensor2 = ['Cony\t' num2str(i) '\Position_R3.txt'];
    sensor3 = ['Cony\t' num2str(i) '\Position_R4.txt'];
    sensor4 = ['Cony\t' num2str(i) '\Position_R5.txt'];
    sensor5 = ['Cony\t' num2str(i) '\Position_R6.txt'];
    sensor6 = ['Cony\t' num2str(i) '\Position_R7.txt'];

    fid1 = fopen(fuerza);
    C1 = textscan(fid1, '%f %f', 'headerlines', 15);
    fclose(fid1);

    fid2 = fopen(sensor1);
    C2 = textscan(fid2, '%f %f', 'headerlines', 15);
    fclose(fid2);

    fid3 = fopen(sensor2);
    C3 = textscan(fid3, '%f %f', 'headerlines', 15);
    fclose(fid3);

    fid4 = fopen(sensor3);
    C4 = textscan(fid4, '%f %f', 'headerlines', 15);
    fclose(fid4);

    fid5 = fopen(sensor4);
    C5 = textscan(fid5, '%f %f', 'headerlines', 15);
    fclose(fid5);

    fid6 = fopen(sensor5);
    C6 = textscan(fid6, '%f %f', 'headerlines', 15);
    fclose(fid6);

    fid7 = fopen(sensor6);
    C7 = textscan(fid7, '%f %f', 'headerlines', 15);
    fclose(fid7);

    %localizar peaks de fuerza
    [pks, locs] = findpeaks(abs(C1{1,2}), 'minpeakheight', max(abs(C1{1,2}))/4);

    pos = locs(1);

    Yt(3,i) = max(pks);

    C1{1,2} = abs(hilbert(detrend(C1{1,2})));
    C2{1,2} = abs(hilbert(detrend(C2{1,2})));
    C3{1,2} = abs(hilbert(detrend(C3{1,2})));
    C4{1,2} = abs(hilbert(detrend(C4{1,2})));
    C5{1,2} = abs(hilbert(detrend(C5{1,2})));
    C6{1,2} = abs(hilbert(detrend(C6{1,2})));
```



```

C7{1,2} = abs(hilbert(detrend(C7{1,2})));

%ventana de tiempo
Xi1_t(:,i) = C2{1,2}(pos-500:pos+11500-1);
Xi2_t(:,i) = C3{1,2}(pos-500:pos+11500-1);
Xi3_t(:,i) = C4{1,2}(pos-500:pos+11500-1);
Xi4_t(:,i) = C5{1,2}(pos-500:pos+11500-1);
Xi5_t(:,i) = C6{1,2}(pos-500:pos+11500-1);
Xi6_t(:,i) = C7{1,2}(pos-500:pos+11500-1);
force_t(:,i) = C1{1,2}(pos-500:pos+11500-1);

tiempo_t(:,i) = C2{1,1}(pos-500:pos+11500-1);

end

save Xi1_t Xi1_t;
save Xi2_t Xi2_t;
save Xi3_t Xi3_t;
save Xi4_t Xi4_t;
save Xi5_t Xi5_t;
save Xi6_t Xi6_t;
save force_t force_t;
save tiempo_t tiempo_t;
save Yt

```

7.C CÓDIGO MATLAB – EXPANSIÓN BASE DE ENTRENAMIENTO

```

clear all;
close all;
clc

%cargar datos de entrenamiento
load Xi1;
load Xi2;
load Xi3;
load Xi4;
load Xi5;
load Xi6;
load Yi;

[nn,mm] = size(Xi1); %nn: respuesta, mm: numero del impacto

%expandir base
mf = [1 2:2:10 14:4:32 40 50];

Xi1_E = [];
Xi2_E = [];
Xi3_E = [];
Xi4_E = [];
Xi5_E = [];
Xi6_E = [];
Yi_E = [];

for k = 1:length(mf)
    for i = 1:mm
        af = Yi(3,i);

        Xa1(:,i) = Xi1(:,i)*mf(k)/af;
    end
end

```

```

        Xa2(:,i) = Xi2(:,i)*mf(k)/af;
        Xa3(:,i) = Xi3(:,i)*mf(k)/af;
        Xa4(:,i) = Xi4(:,i)*mf(k)/af;
        Xa5(:,i) = Xi5(:,i)*mf(k)/af;
        Xa6(:,i) = Xi6(:,i)*mf(k)/af;
        Ya(1:2,i) = Yi(1:2,i);
        Ya(3,i) = mf(k);
    end

    Xi1_E = [Xi1_E Xa1];
    Xi2_E = [Xi2_E Xa2];
    Xi3_E = [Xi3_E Xa3];
    Xi4_E = [Xi4_E Xa4];
    Xi5_E = [Xi5_E Xa5];
    Xi6_E = [Xi6_E Xa6];
    Yi_E = [Yi_E Ya];
end

%Guardar datos expandidos
save Xi1_E Xi1_E;
save Xi2_E Xi2_E;
save Xi3_E Xi3_E;
save Xi4_E Xi4_E;
save Xi5_E Xi5_E;
save Xi6_E Xi6_E;
save Yi_E Yi_E;

```

7.D CÓDIGO PYTHON – IMPLEMENTACIÓN AUTOENCODER

```

import scipy.io
import numpy as np
import tensorflow as tf
import matplotlib.pyplot as plt
import numpy.matlib
from sklearn.preprocessing import MinMaxScaler

x_train = scipy.io.loadmat('Xi1_E.mat')['Xi1_E']
x_train = x_train.transpose()
t = scipy.io.loadmat('Xi1_t.mat')['Xi1_t']
x_test = t.transpose()
x_val = x_test

#normalize
sc = MinMaxScaler(feature_range=(0,1))
x_train = sc.fit_transform(x_train)
x_val = sc.transform(x_val)
x_test = sc.transform(x_test)

#build the Autoencoder
def deep_ae(x_train,x_val,x_test,H,d_z):
    #reset default graph
    tf.reset_default_graph()
    #unpack parameters
    h1 = H[0]
    h2 = H[1]
    h3 = H[2]
    h4 = H[3]
    n_input = x_train.shape[1]

```

```

learning_rate = 4E-5
batch_size = 100
epochs = 20
display_step = 1

#Placeholders for X
x = tf.placeholder(tf.float32, (None, n_input)) #None helps the placeholder
to adjust to the batch size

#Building the graph

layer_1 = tf.layers.dense(x, h1, activation = tf.nn.relu, name='layer_1')
layer_2 = tf.layers.dense(layer_1, h2, activation = tf.nn.relu, name='layer_2')
layer_3 = tf.layers.dense(layer_2, h3, activation = tf.nn.relu, name='layer_3')
layer_4 = tf.layers.dense(layer_3, h4, activation = tf.nn.relu, name='layer_4')

z = tf.layers.dense(layer_4, d_z, activation = tf.nn.tanh, name='z')

layer_5 = tf.layers.dense(z, h4, activation = tf.nn.relu, name='layer_5')
layer_6 = tf.layers.dense(layer_5, h3, activation = tf.nn.relu, name='layer_6')
layer_7 = tf.layers.dense(layer_6, h2, activation = tf.nn.relu, name='layer_7')
layer_8 = tf.layers.dense(layer_7, h1, activation = tf.nn.relu, name='layer_8')

x_reconstruction = tf.layers.dense(layer_8, n_input, activation = None, name='x_reconstruction')

#loss function
cost = tf.sqrt(tf.reduce_mean(tf.square(x-x_reconstruction)))

#Optimization
optimizer = tf.train.AdamOptimizer(learning_rate)
training_op = optimizer.minimize(cost)

# TRAINING
with tf.Session() as sess:

    init = tf.global_variables_initializer()
    sess.run(init) # initialize all the variables before training

    n_batch = len(x_train)//batch_size # the network is fed as many times
as necessary to train with all the data

    for epoch in range(epochs): #we train for all the epochs we want

        for i in range(n_batch): #we train for all the batches in the dataset

            batch_x = x_train[np.random.choice(x_train.shape[0], batch_size,
replace=False),:]

            sess.run(training_op, feed_dict = {x: batch_x})
            #print(i, ' ', )

```

```

        if epoch % display_step == 0:
            train_cost = sess.run(cost, feed_dict={x: x_train})
            val_cost = sess.run(cost, feed_dict={x: x_val})
            print('Training epoch {}/{}. Train Cost = {}, Val cost = {}'.format(epoch,epochs,train_cost,val_cost))

        #TRANSFORM DATASETS
        z_train = sess.run(z, feed_dict={x: x_train})
        z_val = sess.run(z, feed_dict={x: x_val})
        z_test = sess.run(z, feed_dict={x: x_test})

        x_rec = sess.run(x_reconstruction, feed_dict={x: x_train}) #x reconst
ruido

        test_cost = sess.run(cost, feed_dict={x: x_test})
        print('Test cost = {}'.format(test_cost))

    return [z_train,z_val,z_test,x_rec]

#layer and z dimention
H = [9640,7280,4920,2560]
d_z = 200

Z = deep_ae(x_train,x_val,x_test,H,d_z)

#Save latent space
numpy.savetxt('AE1t10.txt' , Z[0].transpose() , delimiter=',')
numpy.savetxt('AEt1t10.txt' , Z[2].transpose() , delimiter=',')

```

7.E CÓDIGO MATLAB – IMPLEMENTACIÓN LME

```

clear all;
close all;
clc

addpath('LME');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%BUILD DATABASES%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ni = 200;
nv = 500;

%Read AE data
AE1 = load('Prueba10\AE1t10.txt');
AE2 = load('Prueba10\AE2t10.txt');
AE3 = load('Prueba10\AE3t10.txt');
AE4 = load('Prueba10\AE4t10.txt');
AE5 = load('Prueba10\AE5t10.txt');
AE6 = load('Prueba10\AE6t10.txt');

%Read AE testing data
AEt1 = load('Prueba10\AEt1t10.txt');
AEt2 = load('Prueba10\AEt2t10.txt');
AEt3 = load('Prueba10\AEt3t10.txt');
AEt4 = load('Prueba10\AEt4t10.txt');
AEt5 = load('Prueba10\AEt5t10.txt');
AEt6 = load('Prueba10\AEt6t10.txt');

```

```

% Build matrices Si
AE = [AE1;AE2;AE3;AE4;AE5;AE6];

%PCs for the testing data
Aet = [Aet1;Aet2;Aet3;Aet4;Aet5;Aet6];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%ESTIMATE IMPACTS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%read observation vector
load Yt; %testing data
load Yi_E; %traing data expanded

for k = 1:50
    k
    gamma = gammanb(AEt(:,k),AE,nv); %determinate parameter gamma in eq. (10)
    [phi,contribute] = lme(AEt(:,k),AE,gamma); %solve LME
    Ye(:,k) = Yi_E(:,contribute)*phi;
end

save Ye Ye

```

7.F CÓDIGO MATLAB – CÁLCULO ERROR

```

clear all;
close all;
clc

load Yt; %testing data
load Ye; %lme result

n = 50;
E_x = 0;
E_y = 0;
E_f = 0;

for k = 1:50
    E_x = E_x + abs(Ye(1,k)-Yt(1,k));
    E_y = E_y + abs(Ye(2,k)-Yt(2,k));
    E_f = E_f + abs((Ye(3,k)-Yt(3,k))/(Yt(3,k)));
end

E_A = (100*(E_x*E_y))/((n^2)*(360*300));

E_F = (E_f*100)/n;

E_theta = E_x/n;
E_z = E_y/n;

x1 = [Yt(1,:); Ye(1,:)];
y1 = [Yt(2,:); Ye(2,:)];

save E_A E_A
save E_F E_F
save E_theta E_theta
save E_z E_z

```

7.G UBICACIÓN SENSORES PIEZOELÉCTRICOS

Sensor	Coordenada θ	Unidad	Coordenada z	Unidad
1	0	[°]	7.5	[mm]
2	0	[°]	22.5	[mm]
3	120	[°]	7.5	[mm]
4	120	[°]	22.5	[mm]
5	240	[°]	7.5	[mm]
6	240	[°]	22.5	[mm]

7.H UBICACIÓN Y MAGNITUD IMPACTOS DE ENTREMIENTO CONFIGURACIÓN VERTICAL

Impacto	Coord. θ	Unidad	Coord. z	Unidad	Magnitud F	Unidad
1	0	[°]	300	[mm]	17.9532	[N]
2	0	[°]	270	[mm]	23.9543	[N]
3	0	[°]	240	[mm]	27.4503	[N]
4	0	[°]	210	[mm]	13.2925	[N]
5	0	[°]	180	[mm]	13.2386	[N]
6	0	[°]	150	[mm]	26.7597	[N]
7	0	[°]	120	[mm]	14.4950	[N]
8	0	[°]	90	[mm]	16.5558	[N]
9	0	[°]	60	[mm]	8.0260	[N]
10	0	[°]	30	[mm]	8.3495	[N]
11	30	[°]	300	[mm]	14.7446	[N]
12	30	[°]	270	[mm]	9.4116	[N]
13	30	[°]	240	[mm]	13.1236	[N]
14	30	[°]	210	[mm]	14.8183	[N]
15	30	[°]	180	[mm]	10.8516	[N]
16	30	[°]	150	[mm]	12.6722	[N]
17	30	[°]	120	[mm]	9.8883	[N]
18	30	[°]	90	[mm]	10.7208	[N]
19	30	[°]	60	[mm]	10.0713	[N]
20	30	[°]	30	[mm]	16.7215	[N]
21	60	[°]	300	[mm]	12.2558	[N]
22	60	[°]	270	[mm]	9.1135	[N]
23	60	[°]	240	[mm]	7.2780	[N]
24	60	[°]	210	[mm]	5.3558	[N]
25	60	[°]	180	[mm]	7.3966	[N]

26	60	[°]	150	[mm]	10.5369	[N]
27	60	[°]	120	[mm]	8.5997	[N]
28	60	[°]	90	[mm]	5.5617	[N]
29	60	[°]	60	[mm]	10.4502	[N]
30	60	[°]	30	[mm]	8.5402	[N]
31	90	[°]	300	[mm]	5.7694	[N]
32	90	[°]	270	[mm]	4.1516	[N]
33	90	[°]	240	[mm]	5.1920	[N]
34	90	[°]	210	[mm]	11.3793	[N]
35	90	[°]	180	[mm]	10.6316	[N]
36	90	[°]	150	[mm]	4.1177	[N]
37	90	[°]	120	[mm]	11.3413	[N]
38	90	[°]	90	[mm]	6.4804	[N]
39	90	[°]	60	[mm]	2.7441	[N]
40	90	[°]	30	[mm]	5.4428	[N]
41	120	[°]	300	[mm]	4.4554	[N]
42	120	[°]	270	[mm]	5.8470	[N]
43	120	[°]	240	[mm]	2.7860	[N]
44	120	[°]	210	[mm]	3.0573	[N]
45	120	[°]	180	[mm]	4.9039	[N]
46	120	[°]	150	[mm]	4.6381	[N]
47	120	[°]	120	[mm]	5.5649	[N]
48	120	[°]	90	[mm]	2.7302	[N]
49	120	[°]	60	[mm]	3.1009	[N]
50	120	[°]	30	[mm]	8.2237	[N]
51	150	[°]	300	[mm]	3.1881	[N]
52	150	[°]	270	[mm]	8.7148	[N]
53	150	[°]	240	[mm]	5.3948	[N]
54	150	[°]	210	[mm]	2.6159	[N]
55	150	[°]	180	[mm]	6.2908	[N]
56	150	[°]	150	[mm]	4.2922	[N]
57	150	[°]	120	[mm]	2.8317	[N]
58	150	[°]	90	[mm]	2.8401	[N]
59	150	[°]	60	[mm]	2.3575	[N]
60	150	[°]	30	[mm]	2.9917	[N]
61	180	[°]	300	[mm]	7.3597	[N]
62	180	[°]	270	[mm]	3.4369	[N]
63	180	[°]	240	[mm]	6.4946	[N]
64	180	[°]	210	[mm]	7.7132	[N]
65	180	[°]	180	[mm]	7.2822	[N]

66	180	[°]	150	[mm]	3.6724	[N]
67	180	[°]	120	[mm]	5.4578	[N]
68	180	[°]	90	[mm]	4.4206	[N]
69	180	[°]	60	[mm]	4.8323	[N]
70	180	[°]	30	[mm]	8.3279	[N]
71	210	[°]	300	[mm]	4.4942	[N]
72	210	[°]	270	[mm]	7.3358	[N]
73	210	[°]	240	[mm]	4.9653	[N]
74	210	[°]	210	[mm]	3.7682	[N]
75	210	[°]	180	[mm]	5.1362	[N]
76	210	[°]	150	[mm]	7.9450	[N]
77	210	[°]	120	[mm]	4.6675	[N]
78	210	[°]	90	[mm]	6.2991	[N]
79	210	[°]	60	[mm]	4.8528	[N]
80	210	[°]	30	[mm]	5.5929	[N]
81	240	[°]	300	[mm]	4.1710	[N]
82	240	[°]	270	[mm]	7.5281	[N]
83	240	[°]	240	[mm]	8.0129	[N]
84	240	[°]	210	[mm]	6.4433	[N]
85	240	[°]	180	[mm]	5.3956	[N]
86	240	[°]	150	[mm]	9.6889	[N]
87	240	[°]	120	[mm]	7.5679	[N]
88	240	[°]	90	[mm]	4.3667	[N]
89	240	[°]	60	[mm]	6.5641	[N]
90	240	[°]	30	[mm]	7.9338	[N]
91	270	[°]	300	[mm]	5.3452	[N]
92	270	[°]	270	[mm]	8.9269	[N]
93	270	[°]	240	[mm]	5.0111	[N]
94	270	[°]	210	[mm]	5.5060	[N]
95	270	[°]	180	[mm]	4.7376	[N]
96	270	[°]	150	[mm]	9.0120	[N]
97	270	[°]	120	[mm]	6.1131	[N]
98	270	[°]	90	[mm]	5.1672	[N]
99	270	[°]	60	[mm]	5.9580	[N]
100	270	[°]	30	[mm]	5.5877	[N]
101	300	[°]	300	[mm]	4.4128	[N]
102	300	[°]	270	[mm]	6.2341	[N]
103	300	[°]	240	[mm]	8.8207	[N]
104	300	[°]	210	[mm]	3.1367	[N]
105	300	[°]	180	[mm]	3.7268	[N]

106	300	[°]	150	[mm]	2.7160	[N]
107	300	[°]	120	[mm]	5.0030	[N]
108	300	[°]	90	[mm]	3.9751	[N]
109	300	[°]	60	[mm]	2.8389	[N]
110	300	[°]	30	[mm]	2.6876	[N]
111	330	[°]	300	[mm]	5.2080	[N]
112	330	[°]	270	[mm]	8.4266	[N]
113	330	[°]	240	[mm]	3.2551	[N]
114	330	[°]	210	[mm]	13.6798	[N]
115	330	[°]	180	[mm]	5.0244	[N]
116	330	[°]	150	[mm]	7.7619	[N]
117	330	[°]	120	[mm]	10.2348	[N]
118	330	[°]	90	[mm]	6.9463	[N]
119	330	[°]	60	[mm]	7.0051	[N]
120	330	[°]	30	[mm]	6.1735	[N]

7.I UBICACIÓN Y MAGNITUD IMPACTOS DE ENTREMIENTO CONFIGURACIÓN SUSPENDIDA

Impacto	Coord. θ	Unidad	Coord. z	Unidad	Magnitud F	Unidad
1	0	[°]	300	[mm]	9.9890	[N]
2	0	[°]	270	[mm]	13.6560	[N]
3	0	[°]	240	[mm]	14.4717	[N]
4	0	[°]	210	[mm]	16.6287	[N]
5	0	[°]	180	[mm]	12.2999	[N]
6	0	[°]	150	[mm]	18.9742	[N]
7	0	[°]	120	[mm]	15.4144	[N]
8	0	[°]	90	[mm]	20.0503	[N]
9	0	[°]	60	[mm]	16.0135	[N]
10	0	[°]	30	[mm]	11.1358	[N]
11	30	[°]	300	[mm]	5.8037	[N]
12	30	[°]	270	[mm]	8.8428	[N]
13	30	[°]	240	[mm]	7.4996	[N]
14	30	[°]	210	[mm]	4.5960	[N]
15	30	[°]	180	[mm]	8.0374	[N]
16	30	[°]	150	[mm]	8.6462	[N]
17	30	[°]	120	[mm]	8.1588	[N]
18	30	[°]	90	[mm]	8.7731	[N]
19	30	[°]	60	[mm]	7.9736	[N]

20	30	[°]	30	[mm]	7.8558	[N]
21	60	[°]	300	[mm]	6.3512	[N]
22	60	[°]	270	[mm]	6.4561	[N]
23	60	[°]	240	[mm]	9.1400	[N]
24	60	[°]	210	[mm]	7.2680	[N]
25	60	[°]	180	[mm]	9.7003	[N]
26	60	[°]	150	[mm]	8.6767	[N]
27	60	[°]	120	[mm]	9.8616	[N]
28	60	[°]	90	[mm]	8.9228	[N]
29	60	[°]	60	[mm]	10.6655	[N]
30	60	[°]	30	[mm]	7.2272	[N]
31	90	[°]	300	[mm]	9.8537	[N]
32	90	[°]	270	[mm]	11.6931	[N]
33	90	[°]	240	[mm]	13.2893	[N]
34	90	[°]	210	[mm]	14.4572	[N]
35	90	[°]	180	[mm]	9.2482	[N]
36	90	[°]	150	[mm]	10.0538	[N]
37	90	[°]	120	[mm]	11.6901	[N]
38	90	[°]	90	[mm]	11.2339	[N]
39	90	[°]	60	[mm]	5.8942	[N]
40	90	[°]	30	[mm]	16.9739	[N]
41	120	[°]	300	[mm]	9.5528	[N]
42	120	[°]	270	[mm]	4.4366	[N]
43	120	[°]	240	[mm]	6.8142	[N]
44	120	[°]	210	[mm]	9.9207	[N]
45	120	[°]	180	[mm]	7.1580	[N]
46	120	[°]	150	[mm]	9.3797	[N]
47	120	[°]	120	[mm]	6.1563	[N]
48	120	[°]	90	[mm]	9.3080	[N]
49	120	[°]	60	[mm]	7.6809	[N]
50	120	[°]	30	[mm]	8.3990	[N]
51	150	[°]	300	[mm]	7.0150	[N]
52	150	[°]	270	[mm]	9.1477	[N]
53	150	[°]	240	[mm]	14.8131	[N]
54	150	[°]	210	[mm]	14.3098	[N]
55	150	[°]	180	[mm]	5.0806	[N]
56	150	[°]	150	[mm]	9.5197	[N]
57	150	[°]	120	[mm]	5.5908	[N]
58	150	[°]	90	[mm]	7.9537	[N]
59	150	[°]	60	[mm]	14.4600	[N]

60	150	[°]	30	[mm]	10.4608	[N]
61	180	[°]	300	[mm]	5.3414	[N]
62	180	[°]	270	[mm]	12.1389	[N]
63	180	[°]	240	[mm]	8.5001	[N]
64	180	[°]	210	[mm]	5.8729	[N]
65	180	[°]	180	[mm]	5.4719	[N]
66	180	[°]	150	[mm]	8.3294	[N]
67	180	[°]	120	[mm]	8.9947	[N]
68	180	[°]	90	[mm]	7.8010	[N]
69	180	[°]	60	[mm]	8.0267	[N]
70	180	[°]	30	[mm]	11.2414	[N]
71	210	[°]	300	[mm]	11.6091	[N]
72	210	[°]	270	[mm]	15.1259	[N]
73	210	[°]	240	[mm]	9.2044	[N]
74	210	[°]	210	[mm]	8.7918	[N]
75	210	[°]	180	[mm]	17.9541	[N]
76	210	[°]	150	[mm]	12.5648	[N]
77	210	[°]	120	[mm]	17.9181	[N]
78	210	[°]	90	[mm]	10.7360	[N]
79	210	[°]	60	[mm]	13.1436	[N]
80	210	[°]	30	[mm]	10.6362	[N]
81	240	[°]	300	[mm]	13.2026	[N]
82	240	[°]	270	[mm]	12.9270	[N]
83	240	[°]	240	[mm]	7.9767	[N]
84	240	[°]	210	[mm]	15.1917	[N]
85	240	[°]	180	[mm]	10.4231	[N]
86	240	[°]	150	[mm]	10.2623	[N]
87	240	[°]	120	[mm]	8.3130	[N]
88	240	[°]	90	[mm]	11.4435	[N]
89	240	[°]	60	[mm]	9.5398	[N]
90	240	[°]	30	[mm]	13.1214	[N]
91	270	[°]	300	[mm]	9.1957	[N]
92	270	[°]	270	[mm]	7.2584	[N]
93	270	[°]	240	[mm]	6.5300	[N]
94	270	[°]	210	[mm]	12.1625	[N]
95	270	[°]	180	[mm]	11.5994	[N]
96	270	[°]	150	[mm]	11.7767	[N]
97	270	[°]	120	[mm]	12.3540	[N]
98	270	[°]	90	[mm]	12.9238	[N]
99	270	[°]	60	[mm]	0.8554	[N]

100	270	[°]	30	[mm]	11.1559	[N]
101	300	[°]	300	[mm]	7.2487	[N]
102	300	[°]	270	[mm]	11.2105	[N]
103	300	[°]	240	[mm]	11.8016	[N]
104	300	[°]	210	[mm]	8.2215	[N]
105	300	[°]	180	[mm]	14.3786	[N]
106	300	[°]	150	[mm]	12.9597	[N]
107	300	[°]	120	[mm]	15.4624	[N]
108	300	[°]	90	[mm]	9.1526	[N]
109	300	[°]	60	[mm]	9.1954	[N]
110	300	[°]	30	[mm]	10.4750	[N]
111	330	[°]	300	[mm]	9.8070	[N]
112	330	[°]	270	[mm]	15.3111	[N]
113	330	[°]	240	[mm]	14.1350	[N]
114	330	[°]	210	[mm]	13.5966	[N]
115	330	[°]	180	[mm]	19.3717	[N]
116	330	[°]	150	[mm]	20.9288	[N]
117	330	[°]	120	[mm]	15.6052	[N]
118	330	[°]	90	[mm]	17.1608	[N]
119	330	[°]	60	[mm]	13.8304	[N]
120	330	[°]	30	[mm]	13.4128	[N]
121	0	[°]	0	[mm]	11.4741	[N]
122	30	[°]	0	[mm]	7.2086	[N]
123	60	[°]	0	[mm]	8.3031	[N]
124	90	[°]	0	[mm]	14.6030	[N]
125	120	[°]	0	[mm]	8.3217	[N]
126	150	[°]	0	[mm]	11.6325	[N]
127	180	[°]	0	[mm]	8.6704	[N]
128	210	[°]	0	[mm]	6.4618	[N]
129	240	[°]	0	[mm]	10.7189	[N]
130	270	[°]	0	[mm]	9.9447	[N]
131	300	[°]	0	[mm]	6.9711	[N]
132	330	[°]	0	[mm]	5.4099	[N]

7.J UBICACIÓN Y MAGNITUD IMPACTOS DE PRUEBA CONFIGURACIÓN VERTICAL

Impacto	Coord. θ	Unidad	Coord. z	Unidad	Magnitud F	Unidad
1	24	[°]	291	[mm]	4.6914	[N]
2	4	[°]	247	[mm]	11.1096	[N]
3	14	[°]	163	[mm]	8.8284	[N]
4	4	[°]	92	[mm]	6.5198	[N]
5	18	[°]	47.5	[mm]	12.4804	[N]
6	38	[°]	247	[mm]	7.7903	[N]
7	42	[°]	224	[mm]	5.3592	[N]
8	54.4	[°]	173.5	[mm]	9.1215	[N]
9	30	[°]	133	[mm]	5.4805	[N]
10	38	[°]	98	[mm]	11.8766	[N]
11	53.4	[°]	22	[mm]	7.0504	[N]
12	80.7	[°]	254	[mm]	4.4509	[N]
13	60	[°]	138.5	[mm]	3.6378	[N]
14	85	[°]	82.5	[mm]	6.4202	[N]
15	113.7	[°]	199	[mm]	1.2813	[N]
16	100.4	[°]	140	[mm]	4.4094	[N]
17	104.7	[°]	30	[mm]	10.0793	[N]
18	145.4	[°]	247	[mm]	7.2857	[N]
19	124.7	[°]	126	[mm]	7.2807	[N]
20	128.4	[°]	52	[mm]	3.1781	[N]
21	145	[°]	22.5	[mm]	5.6252	[N]
22	165.4	[°]	240	[mm]	9.2755	[N]
23	174	[°]	201	[mm]	5.8635	[N]
24	155	[°]	141	[mm]	5.2156	[N]
25	172	[°]	50	[mm]	4.6423	[N]
26	180	[°]	285	[mm]	2.0653	[N]
27	206.4	[°]	233.5	[mm]	11.7024	[N]
28	180	[°]	165	[mm]	6.2250	[N]
29	194.7	[°]	135.5	[mm]	6.1628	[N]
30	205	[°]	71	[mm]	6.7440	[N]
31	210	[°]	47	[mm]	4.2828	[N]
32	231.7	[°]	282.5	[mm]	2.5320	[N]
33	235	[°]	145	[mm]	4.3631	[N]
34	218	[°]	110	[mm]	5.5139	[N]
35	248.7	[°]	261.5	[mm]	6.8850	[N]

36	266	[°]	173	[mm]	2.3675	[N]
37	265	[°]	128	[mm]	6.6588	[N]
38	266.7	[°]	58	[mm]	5.2458	[N]
39	278.7	[°]	255	[mm]	8.2126	[N]
40	291.4	[°]	159	[mm]	10.9296	[N]
41	277.7	[°]	68.5	[mm]	3.8865	[N]
42	294.7	[°]	23	[mm]	7.4451	[N]
43	306	[°]	280.5	[mm]	12.6558	[N]
44	330	[°]	223	[mm]	8.3705	[N]
45	319	[°]	188.5	[mm]	5.2374	[N]
46	306	[°]	97.5	[mm]	8.9321	[N]
47	327.7	[°]	64	[mm]	1.7140	[N]
48	346.7	[°]	132	[mm]	7.2417	[N]
49	254.4	[°]	300	[mm]	6.0081	[N]
50	128	[°]	291	[mm]	4.8560	[N]

7.K UBICACIÓN Y MAGNITUD IMPACTOS DE PRUEBA CONFIGURACIÓN SUSPENDIDA

Impacto	Coord. θ	Unidad	Coord. z	Unidad	Magnitud F	Unidad
1	24	[°]	291	[mm]	18.3108	[N]
2	4	[°]	247	[mm]	1.9077	[N]
3	14	[°]	163	[mm]	6.7095	[N]
4	4	[°]	92	[mm]	11.4920	[N]
5	18	[°]	47.5	[mm]	12.6566	[N]
6	38	[°]	247	[mm]	15.3007	[N]
7	42	[°]	224	[mm]	6.8717	[N]
8	54.4	[°]	173.5	[mm]	4.3629	[N]
9	30	[°]	133	[mm]	8.2163	[N]
10	38	[°]	98	[mm]	5.8479	[N]
11	53.4	[°]	22	[mm]	14.5747	[N]
12	80.7	[°]	254	[mm]	5.6491	[N]
13	60	[°]	138.5	[mm]	11.0019	[N]
14	85	[°]	82.5	[mm]	8.8405	[N]
15	113.7	[°]	199	[mm]	11.2006	[N]
16	100.4	[°]	140	[mm]	12.6671	[N]
17	104.7	[°]	30	[mm]	13.8833	[N]
18	145.4	[°]	247	[mm]	9.7797	[N]
19	124.7	[°]	126	[mm]	7.0515	[N]

20	128.4	[°]	52	[mm]	4.4439	[N]
21	145	[°]	22.5	[mm]	2.5436	[N]
22	165.4	[°]	240	[mm]	1.8192	[N]
23	174	[°]	201	[mm]	3.6238	[N]
24	155	[°]	141	[mm]	5.4372	[N]
25	172	[°]	50	[mm]	5.9107	[N]
26	180	[°]	285	[mm]	0.4768	[N]
27	206.4	[°]	233.5	[mm]	17.6191	[N]
28	180	[°]	165	[mm]	7.0162	[N]
29	194.7	[°]	135.5	[mm]	7.4350	[N]
30	205	[°]	71	[mm]	5.8710	[N]
31	210	[°]	47	[mm]	8.7554	[N]
32	231.7	[°]	282.5	[mm]	5.0142	[N]
33	235	[°]	145	[mm]	10.9842	[N]
34	218	[°]	110	[mm]	11.9553	[N]
35	248.7	[°]	261.5	[mm]	14.9700	[N]
36	266	[°]	173	[mm]	15.0922	[N]
37	265	[°]	128	[mm]	12.7591	[N]
38	266.7	[°]	58	[mm]	9.5176	[N]
39	278.7	[°]	255	[mm]	13.9824	[N]
40	291.4	[°]	159	[mm]	11.0718	[N]
41	277.7	[°]	68.5	[mm]	8.2394	[N]
42	294.7	[°]	23	[mm]	10.9455	[N]
43	306	[°]	280.5	[mm]	8.8185	[N]
44	330	[°]	223	[mm]	13.7065	[N]
45	319	[°]	188.5	[mm]	5.9666	[N]
46	306	[°]	97.5	[mm]	13.3443	[N]
47	327.7	[°]	64	[mm]	9.5622	[N]
48	346.7	[°]	132	[mm]	12.2052	[N]
49	254.4	[°]	300	[mm]	12.4918	[N]
50	128	[°]	291	[mm]	11.6624	[N]