

UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA CIVIL AMBIENTAL



Optimización de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado

**TESIS PARA OPTAR EL TÍTULO DE
INGENIERO CIVIL AMBIENTAL**

AUTOR

Jose Manuel Vasquez Requejo

ASESOR

Gabriel Cachi Cerna

<https://orcid.org/0000-0003-0076-3211>

Chiclayo, 2022

Optimización de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado

PRESENTADA POR

Jose Manuel Vasquez Requejo

A la Facultad de Ingeniería de la
Universidad Católica Santo Toribio de Mogrovejo

para optar el título de

INGENIERO CIVIL AMBIENTAL

APROBADA POR

Segundo Guillermo Cieza Carranza

PRESIDENTE

José Alberto Martínez Acero

SECRETARIO

Gabriel Cachi Cerna

VOCAL

Dedicatoria

A mis padres, Ramiro Vasquez Arevalo y Luz Mari Requejo Aguilar por seguir apoyándome en hacer realidad mis sueños, por motivarme hacer lo que me apasiona y educarme para ser mejor persona.

A mis hermanas, Mayra y María Yaquelin Vasquez Requejo por su motivación y apoyo incondicional para lograr mis metas.

Agradecimientos

A Dios, por permitirme hacer realidad mis metas y por guiarme en el camino

A mis padres y hermanas por la motivación y apoyo y estar siempre conmigo en el alcance de mis metas.

A mi asesor, por compartir su experiencia, apoyo y su valioso tiempo para desarrollar esta tesis.

Índice

Resumen	10
Abstract	11
I. Introducción	12
II. Marco teórico	14
Antecedentes	14
Bases Teóricas.....	17
III. Metodología	41
IV. Resultados y discusión	55
V. Conclusiones	79
VI. Recomendaciones.....	80
VII. Referencias bibliográficas	81
VIII. Anexos.....	87

Lista de tablas

Tabla 1: Espesor mínimo de losas en una dirección macizas y nervadas no preesforzadas	18
Tabla 2: Peralte de losas aligeradas en una dirección.....	19
Tabla 3: Espesor mínimo de losas no preesforzadas de dos direcciones sin vigas interiores (cm) ^[1]	19
Tabla 4: Espesor mínimo de losas no preesforzadas de dos direcciones con vigas entre los apoyos en todos lados (cm)	20
Tabla 5: Comparación de las ventajas y desventajas de tipos de losas de concreto armado.....	24
Tabla 6: Comparación de las cualidades de diferentes técnicas de IA	27
Tabla 7: Resumen y comparación de las ventajas y desventajas de diferentes algoritmos de inteligencia de enjambre	28
Tabla 8: Ventajas y desventajas de las redes neuronales (RNs).....	29
Tabla 9: Redes heteroasociativas y autoasociativas.....	32
Tabla 10: Operacionalización de variables.....	43
Tabla 11: Instrumentos para las técnicas de recolección de datos	44
Tabla 12: Momentos últimos por tramos	58
Tabla 13: Cortantes últimas por tramos	59
Tabla 14: Área de refuerzo por flexión por tramos.....	60
Tabla 15: Momentos resistentes aligerado 30cm	60
Tabla 16: Variables del vector entrada a la red neuronal	62
Tabla 17: Variables del vector de salida de la red neuronal	63
Tabla 18: Número óptimo de neuronas de la capa oculta.	65
Tabla 19: Matriz de umbrales y pesos sinápticos de las neuronas capa oculta.....	68
Tabla 20: Matriz de umbrales y pesos sinápticos de la capa de salida	68
Tabla 21 Magnitudes comunes de los parámetros usados para la simulación	69
Tabla 22 Coeficientes del máximo momento, cortante y deflexión usado para el diseño.....	70
Tabla 23: Comparación de soluciones.....	73
Tabla 24: Verificación de restricciones.....	73
Tabla 25: Datos de entrada a la red entrenada.....	74
Tabla 26: Resultados del diseño en tramos por la red neuronal 1	75
Tabla 27: Resultados del diseño en apoyos por la red neuronal 2	76
Tabla 28: Comparación de resultados de diseño tradicional vs proceso la red neuronal 1	77
Tabla 29: Comparación de resultados de diseño tradicional vs proceso la red neuronal 2	77

Tabla 30: Pesos unitarios de los materiales de construcción	91
Tabla 31: Cargas vivas mínimas distribuidas uniformemente.....	91
Tabla 32: Diámetros de acero corrugado en [mm] y [pulg].....	92
Tabla 33: Data de entrenamiento redes neuronal 1.....	93
Tabla 34: Data de entrenamiento red neuronal 2	95
Tabla 35: Matriz de umbrales y pesos sinápticos capa oculta de la red neuronal 2.....	98
Tabla 36: Matriz de umbrales y pesos sinápticos capa de salida de la red neuronal 2	98

Lista de figuras

Figura 1: Análisis y diseño de losas macizas unidireccionales	21
Figura 2: Características geométricas de losas nervadas armadas en una dirección.....	21
Figura 3: Características geométricas de una losa aligerada	21
Figura 4: Diagrama de flujo de diseño de losas no preesforzadas	23
Figura 5: Publicaciones de investigación sobre el uso de diferentes ramas de IA en ingeniería estructural	26
Figura 6: Revisión de técnicas de IA aplicadas en ingeniería civil.....	26
Figura 7: Principales funciones de transferencia utilizadas	31
Figura 8: Una neurona que recibe tres entradas	32
Figura 9: Jerarquía del lobo gris (el dominio disminuye de arriba hacia abajo)	34
Figura 10: Actualización de posición en GWO.....	36
Figura 11: Diagrama de flujo para el algoritmo GWO.....	36
Figura 12: Comparación entre el proceso de diseño (a) tradicional y (b) óptimo	37
Figura 13: Objetivos de optimización del diseño estructural de concreto armado.....	38
Figura 14: Número de artículos recopilados con técnica de optimización adoptada en cada categoría de tema de investigación	40
Figura 15: Organización de los documento y referencias	45
Figura 16: Ejemplos de la porción de losa que debe incluirse con la viga.....	47
Figura 17: Geometría de los aligerados típicos	47
Figura 18: Posición del bloque de compresión vigas Tee.....	48
Figura 19: Alternancia de carga viva de 200 kg/m^2 en un modelo de losa aligerada unidireccional.....	49
Figura 20 Sección de una losa nervada y el diseño de las variables seleccionadas	50
Figura 21: Plano de distribución de oficinas	55
Figura 22: Estructuración de la planta de distribución de oficinas	57
Figura 23: Diagrama de momentos últimos.....	58
Figura 24: Ejecución del programa 1 diseño losas aligeradas	60
Figura 25: Desarrollo del refuerzo en una losa aligera de 2 tramos continuos.....	61
Figura 26: Arquitectura del modelo de red neuronal.....	61
Figura 27: Representación de las variables de entrada y salida de la red neuronal.....	63
Figura 28: Datos de entrada para entrenamiento red neuronal	64
Figura 29: Datos de entrada para entrenamiento red neuronal – Losas a diseñar	64

Figura 30: Datos de salida para entrenamiento de red neuronal – Refuerzo.....	65
Figura 31: Arquitectura de la red neuronal para diseño refuerzo positivo en losas.	66
Figura 32: Estado del entrenamiento de la red neuronal	67
Figura 33: Mejor valor del error cuadrático red neuronal.....	67
Figura 34: Sección de una losa maciza	70
Figura 35: Convergencia de la función objetivo (a) sin penalizar (b) penalizada -GWO	74
Figura 36: Desarrollo del refuerzo positivo y negativo MOD2 y MOD4	76
Figura 37: Emisiones globales de CO ₂ por sector, 2018.....	87
Figura 38: Producción de cemento a nivel mundial en millones de toneladas	87
Figura 39: Modelo de ficha de análisis documental	88
Figura 40 Momentos en losas continuas por el Método de los Coeficientes ACI.....	88
Figura 41: Consideraciones para el corte de acero en regiones de momento positivo y negativo	88
Figura 42: Entrenamiento de la red neuronal 2	89
Figura 43: Verificación de restricciones losas macizas -gwo.....	90
Figura 44: Ingreso de variables a la red neuronal para el diseño del modelo 3	90

Resumen

Las prácticas actuales de diseño estructural de concreto armado consta en realizar cálculos de diseño manuales o mediante la automatización parcial con software de computadora. Se ha observado que tales prácticas conducen a menudo considerar valores mucho más altos de factor de seguridad y otros supuestos innecesarios. Frente a esa problemática esta investigación se centra en la optimización de losas, teniendo como objetivo principal optimizar el diseño de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado utilizando el algoritmo grey wolf optimization (GWO) y una red neuronal. Para ello, se han diseñado 50 modelos de losas aligeradas con sobrecarga de 250 kg/m² y una carga muerta adicional por tabiquería y acabados de 220 kg/m².

Los modelos diseñados sirvieron para el entrenamiento de dos redes neuronales (1 para diseño refuerzo positivo y 2 refuerzo negativo). Los parámetros de entrada a la red se definieron como la luz de la edificación, número de tramos continuos y apoyos de las losas, la luz libre, número de tramo y apoyo a diseñar. Respecto al algoritmo gwo se obtuvo convergencia temprana y las variables caen en sus límites. Finalmente, las redes neuronales entrenadas si brindan buenos resultados con errores menos de 10% para el diseño de modelos dentro de su campo de entrenamiento, fuera de el para el refuerzo positivo se obtuvo los diámetros de refuerzo igual al diseño tradicional y las longitudes de desarrollo se diferenciaron dentro del 50%.

Palabras clave: Losas, Inteligencia Artificial, Optimización, Redes Neuronales.

Abstract

Current reinforced concrete structural design practice consists of performing manual design calculations or partial automation with computer software. It has been observed that such practices often lead to consider much higher values of factor of safety and other unnecessary assumptions. Faced with this problem, this research focuses on the optimization of slabs, with the main objective of optimizing the design of two types of non-prestressed slabs with large spans in reinforced concrete buildings using the gray wolf optimization (GWO) algorithm and a neural network. For this, 50 models of lightened slabs have been designed with an overload of 250 kg /m² and an additional dead load for partitions and finishes of 220 kg /m².

The designed models were used for the training of two neural networks (1 for positive reinforcement design and 2 for negative reinforcement). The input parameters to the network were defined as the span of the building, number of continuous sections and slab supports, free span, section number and support to be designed. Regarding the gwo algorithm, early convergence was obtained and the variables fall within its limits. Finally, the trained neural networks do provide good results with errors less than 10% for the design of models within their training field, outside of that for positive reinforcement, reinforcement diameters equal to the traditional design and development lengths were obtained. they differed within 50%.

Keywords: Slabs, Artificial Intelligence, Optimization, Neural Networks

I. Introducción

El concreto armado es el material de construcción de mayor uso en las edificaciones e infraestructura civil debido a: su facilidad de conseguir, alta durabilidad, resistencia al fuego y muy resistente a compresión. En consecuencia, la producción anual mundial del concreto armado fue superior a 10 mil millones de m³ en el año 2012 [2], lo que demandó gran cantidad de materias primas y produjo impactos significantes en el medio ambiente [3]. La agencia Internacional de Energía (IEA) y el Programa Ambiental de las Naciones Unidas (UNEP) destacan que la construcción y edificaciones son un actor clave en el cambio climático.

En concordancia con el Global Status Report 2019 de la IEA y UNEP la industria de la construcción y operaciones de la misma representan el 39% de las emisiones globales de CO₂ en el año 2017 (ver Figura 37) [4]. En toda la Unión Europea, la construcción y el uso de edificios representan alrededor de la mitad de los materiales que se extraen y la energía que se utiliza, y aproximadamente un tercio del agua consumida. El uso de estos recursos está determinado en gran medida por las decisiones sobre el diseño y la elección de los materiales de construcción [5].

Las prácticas actuales de diseño estructural de concreto armado consta en realizar cálculos de diseño manuales o mediante la automatización parcial con softwares de computadora, como: CSI ETABS, SAP2000, Tekla Structures y Autodesk RSA, etc. Se ha observado que tales prácticas conducen a menudo considerar valores mucho más altos de factor de seguridad y otros supuestos innecesarios. Además, estos enfoques a veces no satisfacen los requisitos mínimos de los parámetros de restricción en los reglamentos de diseño, afectando así la estructura en general. Todas esas razones conllevan a errores poco económicos y a requerir cantidades mayores de los materiales de construcción, como concreto y acero de refuerzo [1].

Por lo tanto, el diseño es la etapa principal y más importante de la construcción de edificios en la que se diseñan los diferentes elementos estructurales siguiendo especificaciones reglamentarias. Y ante lo expuesto y en vista del desarrollo de la tecnología, la construcción; y la mayor competencia en este sector, es necesario ofrecer soluciones estructurales y de construcción que permitan mayor eficiencia. Debido a ello, la presente investigación utiliza las técnicas de la Inteligencia Artificial (IA) con estrategias de optimización para el diseño de losas aligeradas y macizas de concreto armado.

La optimización de estructuras comenzó con la disminución de peso de las estructuras de madera, pero no es hasta inicios de siglo XIX para ver las primeras aportaciones en el diseño de mínimo peso de arcos y cerchas metálicas. Sin embargo, la optimización tiene mayor

influencia a partir del desarrollo de la programación computacional y los algoritmos genéticos después de los años 70 y 80 [6]. En la actualidad, muchos de los problemas de ingeniería civil y estructural son resueltos con IA, teniendo aplicación en el monitoreo de la salud y gestión estructural, y optimización del diseño [7].

El desarrollo de la presente investigación se justifica en los siguientes aspectos: social, económico y ambiental. En el aspecto social, esta tesis difunde la utilización de la Inteligencia Artificial en el diseño estructural de losas aligeradas y macizas de concreto armado. Además, facilitará la labor profesional de los ingenieros civiles con la automatización de la etapa inicial del diseño. En el aspecto económico, el proyecto es beneficioso por que se logrará minimizar el costo total de la estructura con el ahorro de materiales como el volumen de concreto y la cantidad de acero de refuerzo.

En el aspecto ambiental, el diseño óptimo de losas aligeradas y macizas tendrá como consecuencia minimizar el uso de materiales de construcción. Al reducir estos materiales, se logrará disminuir el uso de materias primas y la producción de nuevos materiales; lo que conlleva a tener menores emisiones de CO₂ por la industria del sector construcción (ver Figura 37). Asimismo, uno de los retos del siglo XXI es la construcción sostenible por lo que se busca mejorar las losas desde el diseño con el ahorro de materiales, ya que la producción del cemento ha tenido crecimiento constante después del año 2000 (ver Figura 38).

La presente tesis, tiene como objetivo principal optimizar el diseño de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado utilizando el algoritmo grey wolf optimization (GWO) y una red neuronal. Asimismo, como objetivos específicos se tiene; diseñar de forma tradicional losas de entrepiso macizas y aligeradas que se encuentran empotradas, simplemente apoyadas y en voladizo de edificaciones de concreto armado; implementar un programa computacional basado en el algoritmo GWO y una red neuronal para el diseño óptimo de losas macizas y aligeradas; y comparar la eficiencia del programa implementado con el diseño tradicional y los resultados de otras investigaciones.

El desarrollo de esta investigación se limitó al estudio de losas aligeradas y macizas no preesforzadas de concreto armado, con resistencia específica a compresión de 210 kg/cm². Dichas losas se encuentran apoyadas de tres formas: empotradas, simplemente apoyadas y en voladizo, y soportan cargas distribuidas.

II. Marco teórico

Antecedentes

- D. Kumar y S. Akhtar, «Design Optimization of Reinforced Concrete Slabs using Various Optimization Techniques,» *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 3, n° 5, pp. 45-58, 2019 [8].

En el presente artículo se buscó encontrar los mejores parámetros de diseño para cuatro diferentes tipos de losa de concreto armado (losa simple apoyada, losa continua en un extremo, continua en ambos extremos y losa en voladizo) usando tres algoritmos metaheurísticos. Estos son: algoritmos genéticos (GA), particle Swarm optimization (PSO) and gray wolf optimization (GWO) ejecutados en MATLAB. Para ello, se plantea minimizar el costo de losa con la optimización de espesor, diámetro y espaciamiento del refuerzo. La función objetivo (costo total) viene dada por el costo del concreto, refuerzo y encofrado, además el diseño cumple las restricciones del código ACI 318. Y para contrastar sus resultados se comparan con tres investigaciones previas que utilizan los algoritmos neural dynamics model, PSO, cuckoo search (CS).

Las conclusiones fueron: GWO converge más rápido para optimización estructuras de concreto, haciendolo en 15 iteraciones y GA con PSO alrededor 25 iteraciones. Para la losa continúa en un extremo GWO y PSO registran mejores resultados y en una losa simple apoyada los resultados de GA, PSO y GWO son parecidos. Por último, los tres métodos registran mejores respuestas que los propuestos en los trabajos anteriormente. Y para cada tipo de losa, la reducción costo es 10.43%, 12.61%, 14.95% y 20.33%, respectivamente.

- S. Fayaz y K. Madhavi, «Design Optimization of Reinforced Concrete Slabs Using Genetic Algorithms,» *International Journal of Civil Engineering and Technology (IJCIET)*, vol. 9, n° 4, pp. 1370-1386, 2018 [9]

El presente artículo tuvo como objetivo principal estudiar la minimización del costo total de una losa de concreto armado simplemente apoyada y en voladizo. La función objetivo incluye el costo del concreto, el costo del reforzamiento y el costo de encofrado con su volumen de materiales, respectivamente. Asimismo, las variables de diseño son: espesor de losa, porcentaje de acero y espaciamiento barras de acero, con restricciones del código iraní IS 456-2000. Para la optimización, utiliza algoritmos genéticos (GA) desarrollado en MATLAB y los resultados obtenidos los compara con resultados publicados en investigaciones previas.

Las conclusiones del estudio muestran que para el aumento del $f'c$, carga y grado de acero fijo, aumenta el costo total de la losa. Al aumentar el grado de acero, disminuye el costo óptimo con carga fija y $f'c$ único. El costo óptimo para la losa se logra mediante un $f'c$ igual 20 N/mm² (204 kg/cm²) y f_y de 500 N/mm² (5100 kg/cm²). La losa en voladizo tiene el costo total máximo y este valor aumenta predo a tramos mayores. Finalmente, los resultados obtenidos se comparó con la literatura habiendo una reducción de costos 6.78% y 18.92% para losa apoyada y en voladizo respectivamente.

- A. Kaveh and S. Bijari, "Optimum cost design of reinforced concrete one-way ribbed slabs using CBO, PSO and democratic PSO algorithms," *Asian J. Civ. Eng.*, vol. 15, no. 6, pp. 788–802, 2014 [10].

El propósito de este artículo científico es optimizar el costo estructural de una losa nervada unidireccional de concreto armada que se encuentra simplemente apoyada en ambos extremos. Para ello utilizó los algoritmos colliding bodies optimization (CBO) y particle swarm optimization (PSO); y una versión mejorada de PSO denominada democratic particle swarm optimization (DPSO). Las variables de diseño consideradas fueron las siguientes: espesor de la losa, espacio entre nervios, ancho inferior y superior del nervio, diámetro del refuerzo y peralte del nervio.

La función objetivo de la optimización incluyó los costos asociados al concreto y el acero, así como el vaciado y montaje de armadura de la losa a minimizarse. Por otro lado, las restricciones del diseño fueron de acuerdo al reglamento ACI 318-08 y el rendimiento de los algoritmos se comparará con harmony search (HS). Los resultados de convergencia indica que CBO y DPSO detienen su análisis en el número 1440 y 3480 respectivamente; sin embargo, HS y PSO no alcanzan finalizar su análisis en 6000 aunque se vuelven rectas en 3000 y 1260 respectivamente. La conclusion del estudio es que los métodos DPSO y CBO son enfoques potentes y eficientes para encontrar la solución óptima a los problemas de optimización estructural.

- E. Ghandi, N. Shokrollahi y M. Nasrolahi, «Optimum Cost Design of Reinforced Concrete Slabs Using Cuckoo Search Optimization Algorithm,» *International Journal Of Optimization In Civil Engineering*, vol. 7, nº 4, pp. 539-564, 2017 [11].

El presente artículo se propone demostrar que el algoritmo evolutivo puede diseñar y optimizar las estructuras de la vida real de manera eficiente. Se estudió dos tipos de losa de concreto armado, en una dirección y plana. La función objetivo equivalió a la suma del costo

de concreto, barras de refuerzo y encofrado con variables independientes como espesor de losa, diámetro barra y espaciamiento. Para la optimización, utiliza cuckoo optimization algorithm (COA) desarrollado en lenguaje Matlab. El diseño y las restricciones de acuerdo al código ACI 318. Las respuestas obtenidos son cotejados con un neural dynamics model and PSO. Las conclusiones fueron: COA tiene una velocidad aceptable de convergencia, a las 1000-2000 iteraciones, y sus resultados comparado a una red neural dinámica y/o PSO son mejores.

- M. Aldwaik y H. Adeli , «Cost optimization of reinforced concrete flat slabs of arbitrary configuration in irregular highrise building structures,» *Structural and Multidisciplinary Optimization*, vol. 54, n° 1, pp. 151-164, 2016 [12].

En este artículo científico se plantea el problema que la optimización de losas planas está limitada a optimizar espesor analizando tramos equivalentes aplicados a diseños regulares de piso rectangular y simétrico. Entonces, plantea construir un modelo de configuración arbitraria con cualquier columna y muro de corte en edificios irregulares y de gran altura de uso común. Para la optimización utiliza el neural dynamics model of Adeli and Park (NDAP) desarrollado en Matlab con las limitantes del código ACI 318-11. Las variables de diseño para la optimización son: espesor de losa, diámetro refuerzo adicional y espaciamiento en x e y, refuerzo en momento positivo y/o negativo según lo requiera y refuerzo por punzonamiento de columnas. Esta metodología fue aplicada a dos losas planas típicas (pisos 12 a 24 y 25 a 32) de edificios de gran altura, 36 niveles. El principal resultado de la investigación indica que para losas irregulares se puede alcanzar ahorros en costo de hasta 9%, empleando la optimización con NDAP).

- M. Afzal, “Evaluation and Development of Automated Detailing Design Optimization Framework for RC Slabs Using BIM and Metaheuristics,” The Hong Kong University of Science and Technology, 2019 [1].

La presente investigación desarrolla una recopilación de las técnicas de optimización de estructuras de concreto armado para reducir la cantidad de materiales de construcción, el costo de materiales y los efectos ambientales. Además, desarrolla un marco holístico basado en BIM utilizando los diferentes algoritmos metaheurísticos (SGA, SGA-SQP y PSO-SQP) para el diseño óptimo de losas sólidas considerando minimización del costo total de construcción. Para ello se estudió el caso de una losa sólida típica de concreto armado en una y dos direcciones con diferentes condiciones de carga y propiedades del material.

Las variables consideradas fueron el refuerzo de acero (diámetros y espaciamentos entre barras de refuerzo) y la selección de resistencia característica apropiada del concreto con las restricciones de diseño y construcción de la norma británica. Concluye que la integración de la tecnología BIM y las estrategias metaheurísticas permite la simplificación y automatización de la etapa de optimización del diseño detallado de la construcción de edificios. También que el modelo desarrollado permite ahorrar el tiempo, el costo y otros recursos importantes gastados en actividades repetitivas de rediseño y alteración. Por último, para futuras investigaciones indica que se podrían considerar otros tipos de losas (losas planas, nervadas y onduladas, losas de núcleo hueco y losas pre / postensado, etc.) y componentes (columna RC, viga, etc.) considerando otros códigos de diseño (ACI, Eurocode, Chinese codes, etc).

- J. De Almeida, “Otimização de lajes nervuradas de concreto armado, utilizando algoritmo genético,” Universidade Federal de Pernambuco, 2018 [13].

La presente investigación tuvo como objetivo desarrollar e implementar un programa informático para el diseño optimizado de losas nervadas utilizando algoritmos genéticos. Las variables de diseño fueron: ancho y peralte del nervio, espacio entre nervios, espesor de la losa y área del refuerzo en las dos direcciones; asimismo, la función objetivo estuvo representada por el costo del concreto y la armadura. Para asegurar la funcionalidad y seguridad del diseño óptimo de losas nervadas siguió las prescripciones de la norma brasilera ABNT NBR 6118.

Los resultados del programa implementado en Matlab se comparó con resultados de losas acanaladas existentes en la literatura y con un ejemplo ejecutado en CYPECAD. También se realizó un estudio paramétrico, variando las longitudes de la losa (cuadrada, rectangular) y la resistencia a compresión del concreto. Con ello analizó el comportamiento del costo de los materiales (hormigón y acero) y las variables de diseño. Concluye que los Algoritmos Genéticos funcionan bien en la optimización y puede destacarse como puntos positivos por su fácil implementación y flexibilidad de tratar problemas de diversas restricciones. Por último, el diseño de losa nervada con el programa implementado presentó un ahorro del 37.36% respecto a la sección dimensionada por CYPECAD.

Bases Teóricas

Losas

Las losas de concreto armado son elementos estructurales cuyas dimensiones en planta en comparación a su peralte son relativamente grandes. En las edificaciones, las losas de entrepiso tienen dos funciones. La primera, soportar las cargas verticales referente a su propio peso,

acabados, divisiones y cargas vivas de acuerdo a su uso. La segunda, en el análisis sísmico las losas se consideran diafragmas rígidos que se desplazan sin deformación de su plano bajo el efecto de las cargas horizontales [14], [15].

Clasificación

Las losas pueden clasificarse de acuerdo a la orientación de sus planos de flexión, a su sistema constructivo y a sus condiciones de apoyo.

▪ Según la orientación de sus planos

a) Losas en una dirección

Según la literatura revisada de losas armadas en una dirección se caracterizan porque la relación entre las dimensiones de sus paños es mayor que dos. Esto representa en el elemento que la deflexión sea más pronunciada en una dirección. Por otro lado, la característica más identificable de la losa unidireccional, si está apoyada únicamente en dos lados opuestos [15]–[17].

Para el análisis y diseño se considera como viga de ancho unitario y bajo peralte o espesor. Este se estima satisfaciendo la máxima deflexión de su plano. Además se debe verificar que el concreto puro sea capaz de resistir los esfuerzos de corte ya que no se emplea refuerzo transversal en losas unidireccionales [17].

El código ACI 318-19 [18] muestra la siguiente Tabla 1 para losas macizas no preesforzadas. Estas no soportan o están ligadas a particiones u tipo de construcción susceptibles a dañarse por deflexiones grandes. Señala que el espesor total de la losa h no debe ser menor que el mínimo; sin embargo, estos límites pueden obviarse si el cálculo de deflexiones demuestra que puede utilizarse menores espesores. Asimismo, la Tabla 2 de la referencia [19] muestra los espesores de losas aligeradas en una dirección para un rango de luces libres.

Tabla 1: Espesor mínimo de losas en una dirección macizas y nervadas no preesforzadas

Condición de apoyo	h mínimo ⁽¹⁾	
	<i>Macizas</i>	<i>Nervadas</i>
Simplemente apoyadas	$l/20$	$l/16$
Un extremo continuo	$l/24$	$l/18.5$
Ambos extremos continuos	$l/28$	$l/21$
En voladizo	$l/10$	$l/8$

⁽¹⁾ Relaciones aplicables para concreto de peso normal y $f_y=4200$ kg/cm².

Fuente: Elaboración propia. En base a ACI 318 2019: p. 90

Tabla 2: Peralte de losas aligeradas en una dirección

Peralte h(cm)	Peso propio (kg/m ²)	Luz máxima (m)
17.0	280.0	$L_n \leq 4.0$
20.0	300.0	$4.0 \leq L_n \leq 5.5$
25.0	350.0	$5.0 \leq L_n \leq 6.5$
30.0	420.0	$6.0 \leq L_n \leq 7.5$

Fuente: Elaboración propia. En base a Blanco 1994: pp. 36-37

b) Losas en dos direcciones

Este tipo de losa se encuentra apoyada perimetralmente y cuya relación entre sus dimensiones es menor que dos. La curva de deflexión que presenta es en las dos direcciones. La losa bidireccional puede reforzarse de tres maneras: incorporando vigas entre las columnas, aumentando espesor de la losa alrededor de la columna (*ábacos*) y ensanchando las columnas debajo de la losa (*capiteles de columna*) [15], [16].

Para las losas no preesforzadas sin vigas interiores en dos direcciones, el espesor total de la losa h se determina de acuerdo a lo requerido por la siguiente Tabla 3 y no deben ser menor que 12.5cm en losas sin ábacos o 10.0cm en losas con ábacos [18]. Los valores de dicha tabla pueden obviarse a menos que se cumplan los límites de deflexiones calculadas según 8.3.2 ACI. Para calcular deflexiones debe tenerse en cuenta: el tamaño y forma del panel, condiciones de apoyo y restricciones en los bordes, y otras condiciones. Asimismo, el espesor mínimo de las losas con vigas entre los apoyos en todos los lados se calcula con la Tabla 4.

Tabla 3: Espesor mínimo de losas no preesforzadas de dos direcciones sin vigas interiores (cm)^[1]

f_y kg/cm ² ^[2]	Sin ábacos			Con ábacos		
	Paneles exteriores		Paneles interiores	Paneles exteriores		Paneles interiores
	Sin vigas de borde	Con vigas de borde		Sin vigas de borde	Con vigas de borde	
2,800	$\frac{l_n}{33}$	$\frac{l_n}{36}$	$\frac{l_n}{36}$	$\frac{l_n}{36}$	$\frac{l_n}{60}$	$\frac{l_n}{40}$
4,200	$\frac{l_n}{30}$	$\frac{l_n}{33}$	$\frac{l_n}{33}$	$\frac{l_n}{33}$	$\frac{l_n}{36}$	$\frac{l_n}{36}$
5,620	$\frac{l_n}{27}$	$\frac{l_n}{30}$	$\frac{l_n}{30}$	$\frac{l_n}{30}$	$\frac{l_n}{33}$	$\frac{l_n}{33}$

^[1] l_n es la luz libre en la dirección larga, medida entre caras de los apoyos (cm).

^[2] Para f_y entre los valores dados en la tabla, el espesor mínimo debe obtenerse por interpolación lineal.

Fuente: Elaboración propia. En base a ACI 318 2019: p. 101

Tabla 4: Espesor mínimo de losas no preesforzadas de dos direcciones con vigas entre los apoyos en todos lados (cm)

α_{fm} ^[1]	Espesor mínimo, h , cm.		
$\alpha_{fm} \leq 0.2$	Se aplica 8.3.1.1 ACI 318-19		(a)
$0.2 < \alpha_{fm} \leq 2.0$	Mayor de:	$\frac{\ln\left(0.8 + \frac{fy}{14\,000}\right)}{36 + 5\beta(\alpha_{fm} - 0.2)}$	(a) ^{[2], [3]}
		12.5	(c)
$\alpha_{fm} > 2.0$	Mayor de:	$\frac{\ln\left(0.8 + \frac{fy}{14\,000}\right)}{36 + 9\beta}$	(d) ^{[2], [3]}
		9.0	(e)

^[1] α_{fm} es el valor promedio de α_f para todas las vigas en el borde de un panel.

^[2] \ln corresponde a la luz libre en la dirección larga, medida cara a cara de las vigas (cm).

^[3] β es la relación de la luz libre en la dirección larga a la luz libre en la dirección corta de la losa

Fuente: Elaboración propia. En base a ACI 318 2019: p. 101

Una losa maciza armada en dos direcciones es muy buena en resistencia y rigidez, requiriendo peraltes reducidos, pudiendose considerar en el orden de 1/40 (luz entre apoyos) o 1/180 (perímetro del paño). Por otro lado Los aligerados armados en dos direcciones se usan generalmente cuando se tienen paños más o menos cuadrados y de luces mayores a los 6.0 m [19]. En estos casos se podrá considerar:

$h = 25$ cm para luces comprendidas entre 6.5 y 7.5 m.

$h = 30$ cm para luces comprendidas entre 7.0 y 8.5 m.

▪ Según el sistema constructivo

a) Losas macizas

Son aquellas losas que no tienen vacíos en su interior y se diseñan como vigas de ancho unitario. Es el sistema más convencional en la construcción de losas; sin embargo, no son convenientes si se trata de grandes luces. Debido a su poco espesor que tienen poca rigidez y vibran demasiado. Esto lo convierte en pesadas y antieconómicas ya que requiere aumentar el peralte para incrementar su rigidez [15], [17].

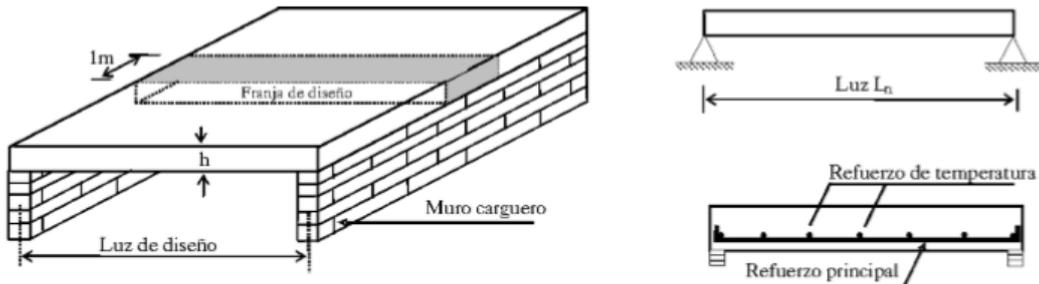


Figura 1: Análisis y diseño de losas macizas unidireccionales

Fuente: Rochel 2002: p. 255

b) Losas aligeradas o nervadas

Consiste en una losa maciza de concreto, de 5 a 10 cm de espesor, apoyada en nervios o viguetas espaciados a distancias que oscilan entre 40 y 100 cm. Los nervios o viguetas llamadas vigas T permite salvar los problemas de una losa maciza, poca rigidez y vibración, siendo más ligera. Esto les permite ser más eficiente para cubrir grandes luces [15], [17].

Las losas aligerada es recomendable para edificaciones pequeñas. Su construcción no requiere de un encofrado metálico para las viguetas a diferencia de una losa nervada. Esto porque el ladrillo de techo actúa como encofrado lateral [17].

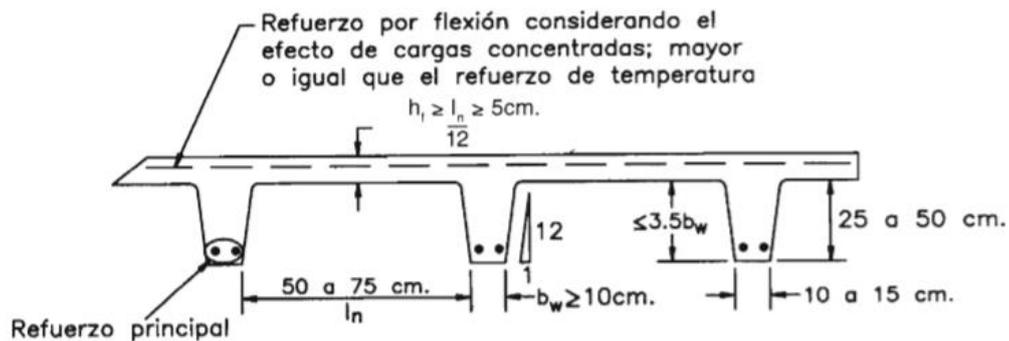


Figura 2: Características geométricas de losas nervadas armadas en una dirección

Fuente: Harmsen 2002: p. 227

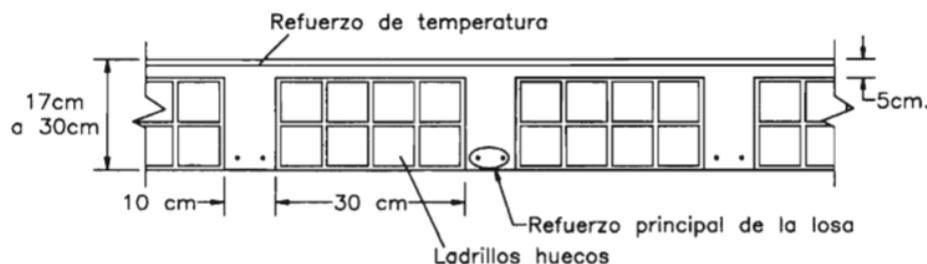


Figura 3: Características geométricas de una losa aligerada

Fuente: Harmsen 2002: p. 228

▪ Según los esfuerzos en el concreto

a) Losas no preesforzadas

Las losas no preesforzadas consiste en elementos de concreto reforzado que este consiste en una combinación de concreto y acero en la que el refuerzo de acero proporciona la resistencia a la tensión que carece el concreto. El acero de refuerzo es también capaz de resistir fuerzas de compresión y se usa en columnas, así como en otros miembros estructurales y en situaciones [16]. La principal ventaja del concreto simplemente reforzado es que no requiere mano de obra calificada y es un material económico, en comparación con otros materiales como el acero estructural [17]. Sin embargo, cuando se trate de cubrir grandes luces con elementos no preesforzados se requieren grandes secciones; por lo que se recurre a pretensados y postensados.

b) Losas preesforzadas

El preesfuerzo puede definirse como la imposición a una estructura de esfuerzos internos que son de carácter opuesto a los causados por las cargas de servicio o de trabajo [16]. Este tipo de estructuras están constituidas por elementos de concreto sometidas a una fuerza de precompresión provista por cables de acero [17]. La finalidad es reducir los esfuerzos de tracción en la sección del elemento de concreto ya que este es un material muy resistente a esfuerzos de compresión. De modo que, el criterio básico de diseño de elementos preesforzados consiste en determinar la fuerza de precompresión requerida y su ubicación en la sección.

Según el proceso constructivo, los dos métodos de preesforzar elementos de concreto son el pretensado y postensado. En el pretensado los tendones (cables) se tensan antes de vaciar el concreto, y luego de fraguado del concreto se cortan los cables y la fuerza de preesfuerzo se transmite al concreto por adherencia. En la construcción postensada los tendones se tensan después de vaciar concreto cuando haya adquirido la resistencia deseada.

La ventaja principal que representa el preesfuerzo es la posibilidad de usar secciones pequeñas para mayores luces. Ello optimiza el peso de los elementos y por lo tanto se reduce considerablemente las cargas totales de diseño en las estructuras de concreto. Sin embargo, el concreto preesforzado resulta costoso porque requiere el uso de concretos y aceros de alta resistencia. También es necesario un control de calidad más estricto en la fabricación lo que genera un mayor costo de mano de obra [16].

Diseño de losas no preesforzadas

El diseño de losas no preesforzadas inicia con el pre dimensionamiento dependiendo de la luz libre y el material. Escogido el espesor de la losa, se calcula las cargas y sobrecargas para

después idealizar la estructura según las condiciones de borde o apoyos. En la idealización se deben combinar y amplificar las cargas para obtener resistencias requeridas (U) y estas aproximarlas a la resistencia de diseño (ϕS_n) de la losa. El diagrama de flujo que se muestra, Figura 4, describe el proceso de diseño de losas no preesforzadas.

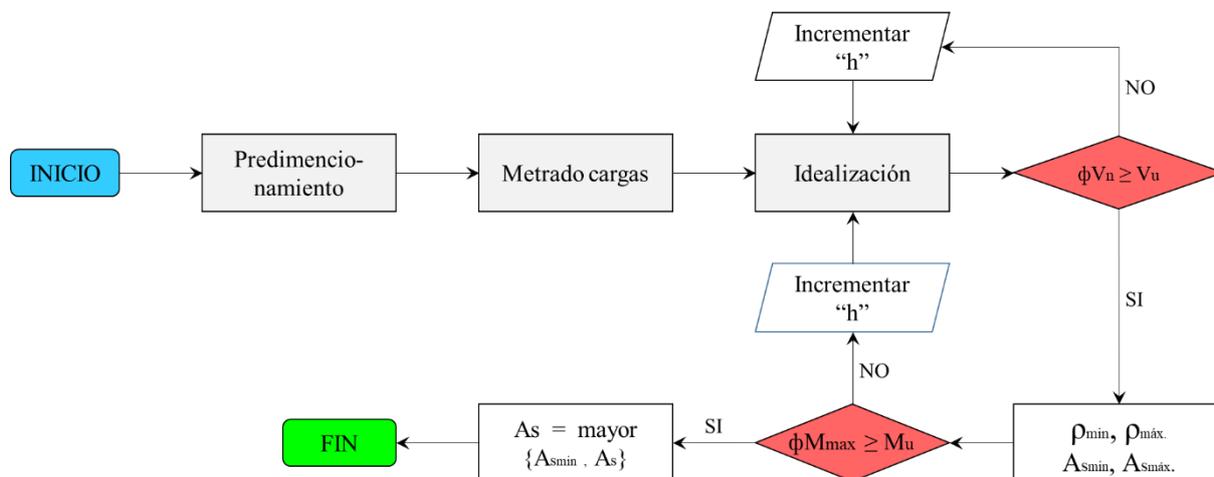


Figura 4: Diagrama de flujo de diseño de losas no preesforzadas

Fuente: Elaboración propia

Las cargas mínimas a emplear en el diseño de losas son obtenidas del reglamento ASCE 7-16. Contempla las cargas permanentes como: el peso de los materiales, dispositivos de servicio, tabiques y otros soportados por la edificación, incluye su peso propio de la estructura. También las sobrecargas mínimas a repartir por ocupación o uso de las edificaciones, como son: peso de los ocupantes, materiales, equipos, muebles y otros elementos móviles [20].

Las combinaciones de carga, factores de carga, reducción de resistencia, diseño y construcción para losas se considera el reglamento ACI 318-19. Este también contempla los métodos de análisis estructural, límites de las deflexiones, restricciones de diseño, desarrollo y empalme del refuerzo, y más requisitos de diseño y detallado para sistemas estructurales y miembros individuales. De esta manera, permite asegurar su funcionamiento y durabilidad de las losas durante su vida útil [18].

Ventajas y desventajas de losas

La Tabla 5 compara las ventajas y desventajas de las losas aligeradas, macizas y nervadas de concreto armado [19].

Tabla 5: Comparación de las ventajas y desventajas de tipos de losas de concreto armado

Losa	Ventajas	Desventajas
Maciza	<ul style="list-style-type: none"> ▪ Espesor depende del diseño ▪ Mayor Resistencia e inercia ▪ Puede especificarse menor espesor ▪ Puede eliminarse el tarrajeo 	<ul style="list-style-type: none"> ▪ Falso techo para embeber tuberías ▪ Mayor concreto y peso ▪ Recomendable en luces pequeñas ▪ Encofrado metálico
Aligerada	<ul style="list-style-type: none"> ▪ Fácil de construir ▪ Ahorro en el encofrado ▪ Permite embeber tuberías ▪ Menos costo de encofrado 	<ul style="list-style-type: none"> ▪ Espesor depende del ladrillo ▪ Peso del ladrillo ▪ Requiere tarrajeo o enlucido ▪ Económicas hasta 8m de luz
Nervada	<ul style="list-style-type: none"> ▪ Espesor depende del diseño ▪ Espacios libres entre viguetas ▪ Menos concreto y peso ▪ Luces mayores de 8m (casetones) 	<ul style="list-style-type: none"> ▪ Encofrado lateral de viguetas ▪ Mayor costo de encofrado ▪ Falso techo para embeber tuberías ▪ Se pierde rigidez

Fuente: Elaboración propia. En base a Blanco 1992

Inteligencia artificial

El término inteligencia artificial (IA) fue conocido por primera vez en 1956 en la conferencia celebrada en la universidad de Dartmouth. Desde su origen, la IA ha tenido que lidiar con el conflicto de que no existía una definición clara y única de inteligencia. De tal manera que en la actualidad existen distintas definiciones de inteligencia artificial, con similitudes entre ellas haciendo énfasis en diferentes aspectos [21]. La IA es una rama de la informática, que participa en la investigación, diseño y aplicación de la computadora inteligente [22].

Para Salehi y Burgueño [23] la IA es un método computacional que intenta imitar la capacidad cognitiva humana a través de la manipulación de símbolos y bases de conocimiento estructurados simbólicamente. Ha sido desarrollada en base a la interacción de disciplinas como: informática, cibernética, lingüística, neurofisiología y neurociencia. En otras palabras, la IA se refiere a la capacidad de una máquina o artefacto pueda simular las mismas funciones que una mente humana.

Diversas ramas de IA se han desarrollado desde que fuese conocida por primera vez en 1956. Estas técnicas pueden dividirse en cinco periodos: de incubación (antes de 1956), de formación (1956-1970), periodo oscuro (1966-1974), de aplicación del conocimiento (1970-1988), y el periodo de desarrollo integrado (1986-presente) [7]. Pero, es a finales de la década de 1980 que la IA cada vez ha venido adquiriendo mayor campo de aplicación. Esto gracias a los avances de la tecnología, con el desarrollo de la red neuronal artificial, aprendizaje automático, reconocimiento de patrones y aprendizaje profundo.

La IA es utilizada para resolver muchos problemas de diversas áreas como en ingeniería, economía, medicina, militar, marina y otros sectores. Lu, Chen y Zheng [22] describe que la IA

en el campo de ingeniería civil está principalmente en el diseño de ingeniería, la gestión de la construcción y la toma de decisiones del programa. También, en la última década, la IA se ha utilizado en monitoreo de salud estructural de infraestructuras y la evaluación y predicción [7].

Técnicas de IA en ingeniería estructural

Salehi y Rigoberto [23] realizaron una revisión de las técnicas de IA utilizadas en ingeniería estructural en la última década (ver figura 5). Aquellos métodos de IA encontrados son aprendizaje automático (machine learning), reconocimiento de patrones (pattern recognition), redes neuronales (neural networks), lógica difusa (fuzzy logic), computación evolutiva (evolutionary computation), aprendizaje profundo (deep learning), sistemas expertos (expert systems), teoría de probabilidad (probability theory), análisis discriminante (discriminant analysis), optimización de enjambre (swarm optimization), optimización metaheurística (metaheuristic optimization) y árboles de decisión (decisión tree).

La mayoría de los métodos anteriores ha aumentado su uso durante los últimos diez años aun cuando las técnicas como la computación evolutiva, la lógica difusa y los sistemas expertos no tuvieron un cambio notable. Se observa también que las redes neuronales han atraído una gran atención por los investigadores y el aumento significativo de los estudios que presentan el uso de machine learning y Deep learning. Por otro lado, la Figura 6 muestra las técnicas de la inteligencia artificial y su desarrollo en la ingeniería civil en los últimos años.

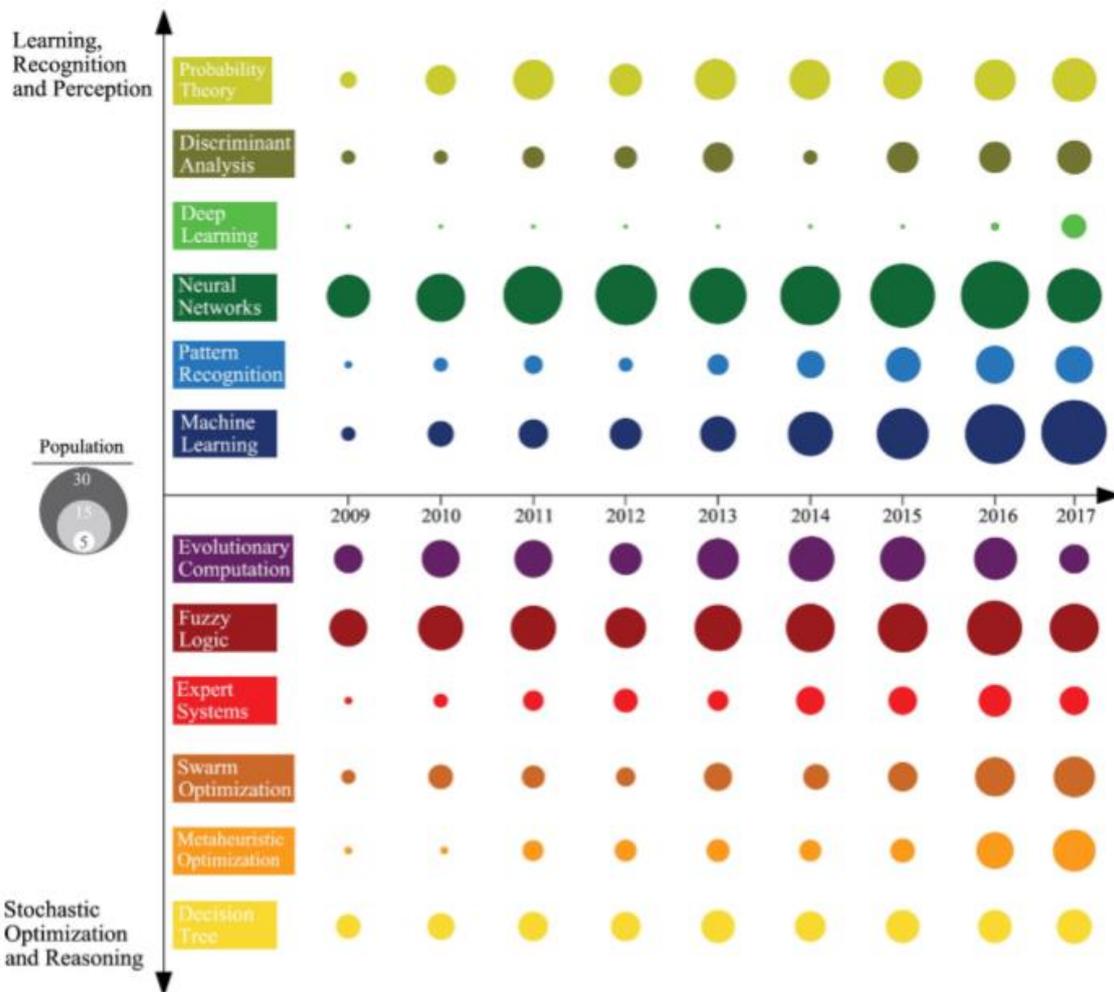


Figura 5: Publicaciones de investigación sobre el uso de diferentes ramas de IA en ingeniería estructural

Fuente: Saleh y Burgeño 2018: p. 173

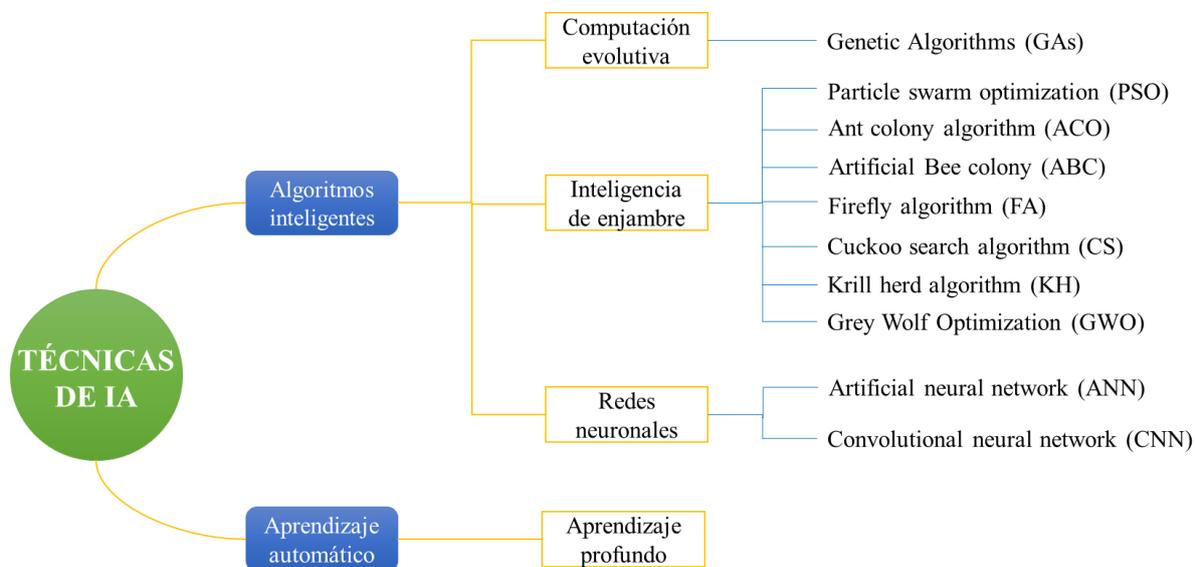


Figura 6: Revisión de técnicas de IA aplicadas en ingeniería civil

Fuente: Elaboración propia. En base a Huang, Li y Fu 2019: pp. 845-875

Tabla 6: Comparación de las cualidades de diferentes técnicas de IA

Computación evolutiva	Inteligencia de enjambre	Redes neuronales artificiales	Aprendizaje profundo
Utiliza proceso iterativo	Utiliza proceso iterativo	Entrenadas con datos disponibles	Se basa en red neuronales
Técnica muy antigua	Técnica activa actualmente	Se sigue mejorando en la actualidad	Gran interés en las investigaciones
Regular aplicación en ingeniería estructural	Creciente uso en ingeniería estructural	Muy aplicada en ingeniería estructural	Poco aplicada en ingeniería estructural
Convergencia prematura ^[1]	Convergencia según algoritmo	Convergencia según arquitectura	Convergencia lenta
Fácil de implementar	Complejidad según algoritmo	Complejidad según arquitectura	Complejidad alta por número capas
Uso decreciente	Uso creciente	Uso constante	Uso reciente

^[1] Cuando aparece una solución con un nivel de adaptación muy superior al resto de la población haciendo que los resultados sean sub óptimos

Fuente: Elaboración propia

Las cualidades de cada técnica de IA comparadas en la Tabla 6 sustenta la elección del método para este estudio. Las redes neuronales han tenido gran influencia en las investigaciones de la última década y los algoritmos de inteligencia de enjambre su uso ha sido creciente en los últimos años. Además, existe un potencial para utilizar estas técnicas de IA porque se han considerado en pocos temas de investigación en optimización.

Ventajas y desventajas de la IA

La técnica de inteligencia de enjambre (SI) fue propuesto por primera vez en 1992 por Hackwood et al. inspirados en el comportamiento colectivo de los insectos sociales y otros animales [7]. Las redes neuronales es una técnica muy antigua, incluso antes que se conociera el término de inteligencia artificial. Pero las RNs en la actualidad han tenido gran influencia por que en ellas se basa otras técnicas. Entonces la tabla 7 resume algunos algoritmos SI desarrollados con sus ventajas y desventajas.

Tabla 7: Resumen y comparación de las ventajas y desventajas de diferentes algoritmos de inteligencia de enjambre

Algoritmo	Creador (es)	Ventajas	Desventajas	Ref.
<i>PSO</i>	Kennedy y Eberhart (1995)	<ul style="list-style-type: none"> ▪ Fácil de implementar ▪ Pocos parámetros ▪ Tiempo de cálculo corto 	<ul style="list-style-type: none"> ▪ Problemas de dispersión sin solución ▪ Difícil definir parámetros iniciales de diseño ▪ Convergencia prematura, atrapado en el óptimo local y poca precisión 	[24] [25]
<i>ACO</i>	Dorigo et al. (1999)	<ul style="list-style-type: none"> ▪ Buscar entre una población en paralelo ▪ Convergencia garantizada ▪ Rápido descubrimiento de buenas soluciones 	<ul style="list-style-type: none"> ▪ Análisis teórico difícil, requiere más investigación experimental ▪ Tiempo incierto para la convergencia ▪ No es recomendable para problemas de gran escala 	[24] [25] [26]
<i>ABC</i>	Karaboga (2005)	<ul style="list-style-type: none"> ▪ Fácil de implementar ▪ Óptima búsqueda global ▪ Fácil agregar o eliminar abejas 	<ul style="list-style-type: none"> ▪ Velocidad de convergencia lenta ▪ Lento cuando se aplica a problemas complejos ▪ Puede atascarse en óptimos locales 	[25] [27]
<i>FA</i>	Yang (2008)	<ul style="list-style-type: none"> ▪ Optimización global y local ▪ Tiempo de cálculo rápido ▪ Puede combinarse con otros algoritmos 	<ul style="list-style-type: none"> ▪ Rendimiento moderado relacionado con el tiempo de procesamiento ▪ Convergencia prematura ▪ No tiene memoria y no hay información extraída durante la búsqueda 	[7] [28]
<i>CSA</i>	Yang et al (2009)	<ul style="list-style-type: none"> ▪ Fácil de implementar ▪ No es fácil de alcanzar el óptimo local ▪ Versátil para asociarse con otro algoritmo 	<ul style="list-style-type: none"> ▪ Carece del análisis matemático ▪ Hay un salto en proceso de búsqueda así en el local no es cuidadosa ▪ Convergencia lenta 	[26] [28] [29]
<i>KH</i>	Gandomi y Alavi (2012)	<ul style="list-style-type: none"> ▪ Fácil de implementar ▪ Puede converger a solución satisfactoria 	<ul style="list-style-type: none"> ▪ No puede implementar la búsqueda global ▪ Cae en óptimos locales y no puede escapar 	[30]
<i>GWO</i>	Mirjalili et al. (2014)	<ul style="list-style-type: none"> ▪ Fácil de implementar ▪ Alto valor de investigación teórica ▪ Convergencia garantizada ▪ Alta precisión de optimización 	<ul style="list-style-type: none"> ▪ Algoritmo nuevo ▪ Investigación y desarrollo de la teoría aún no son perfectos ▪ Poca exploración e investigación 	[8] [25] [31]

Fuente: Elaboración propia

La desventaja común de estos algoritmos son la velocidad de convergencia lenta, la precisión de optimización es baja y es fácil de caer en el óptimo local. La causa de esta deficiencia se por una inadecuada exploración y explotación en cada fase de búsqueda [32]. La exploración se refiere al proceso de visitar regiones completamente nuevas de un espacio de búsqueda, mientras que la explotación es la capacidad de volver a visitar cuidadosamente en esas regiones [33]. También en muchos algoritmos de inteligencia de enjambre, la mejor solución se busca

solo con una sola solución; sin embargo, con GWO la solución óptima puede evaluarse mediante tres soluciones (lobos) [25].

Las redes neuronales tienen muchas aplicaciones y ventajas que hacen sean más adecuadas para ciertos problemas y situaciones. Respecto a técnicas de modelado no lineal convencionales las RNs proporcionan mayor precisión con el mismo número de datos [34]. Otra característica importante es que tienen la capacidad de aprender mediante un entrenamiento previo, con ejemplos (datos). Y cabe resaltar que las RNs son la base de otras técnicas de IA como machine learning o aprendizaje automático [7], [23]. A continuación, la Tabla 8 resume las ventajas y desventajas de las redes neuronales artificiales.

Tabla 8: Ventajas y desventajas de las redes neuronales (RNs)

	Descripción	Ref.
Ventajas	▪ Capacidad de tratar con conocimiento incompleto	[1]
	▪ Pueden desarrollarse utilizando múltiples algoritmos de entrenamiento diferente	[35]
	▪ Adaptación de la experiencia, capacidad de aprendizaje, organización de datos, almacenamiento de conocimiento y tolerante a fallos	[36]
Desventajas	▪ Comportamiento de red inexplicable	[1]
	▪ Requiere mayores recursos computacionales	[35]
	▪ Tiempo de convergencia, según aplicaciones requiere más datos; esto equivale a mayor tiempo de cálculo	[37]

Fuente: Elaboración propia

Redes Neuronales Artificiales

Las redes neuronales artificiales (RNAs) son modelos computacionales que surgieron de la inspiración en el sistema nervioso humano que percibe y procesa la información. Las RNAs imitan funcionamiento del cerebro humano, basándose en el aprendizaje a través de la experiencia y la red neural aprende mediante el entrenamiento [38]. El cerebro está formado por unos cien mil millones de neuronas, en cambio una red neuronal consta de un número finito de neuronas artificiales interconectadas para formar una red de información [39], [40].

Para Osipov, Panov y Yakovlev [41] las RNAs son modelos informáticos paralelos que exhiben un comportamiento excelente para resolver problemas complejos de inteligencia artificial. Aunque muchos investigadores se niegan a usarlas y lo consideran una “caja negra” debido a que es difícil explicar por qué una red neuronal toma una decisión específica. Pero eso no ha sido un obstáculo; ya que se han utilizado en diversas áreas y en especial han sido

utilizadas ampliamente en varios campos de la ingeniería civil. En los últimos años, las RNAs tienen gran interés de investigación con el desarrollo del aprendizaje profundo [7], [23].

Existen dos tipos básicos de problemas que las RNAs pueden tratar de resolver: clasificación y regresión. En el primero, el objetivo consiste en crear un procedimiento mediante el cual un nuevo caso representado por unos atributos observados o características que constituyen los datos de entrada a la RNA. Un ejemplo de problemas de clasificación sería el reconocimiento de rostros. Por otro lado, los problemas de regresión el objetivo es obtener un número en función de los atributos de entrada a la red. Por ejemplo, poder predecir la evolución de los consumos energéticos de un país a lo largo del año en función de ciertos datos históricos [42].

Las RNAs tienen la capacidad de adquirir y mantener conocimientos (basados en información) con un previo entrenamiento. De ello, se concluye que RNA es un conjunto de unidades de procesamiento, representadas por neuronas artificiales interconectadas por muchas interconexiones (sinapsis artificiales), implementadas por vectores y matrices de pesos sinápticos [36, p. 5].

▪ **Arquitectura de una red neuronal**

La arquitectura de las RNAs hace referencia a la organización y disposición de neuronas en la red formando capas interconectadas entre sí. Las capas representan el conjunto de neuronas finitas constituyendo así varias capas de una red neuronal. Por lo general, una RNA se puede dividir en tres partes o capas: una capa de entrada (input layer), capas ocultas o intermedias (hidden or intermediate layers) y la capa de salida (output layer); como se muestra en la Figura 8. Además, [38] las arquitecturas neuronales pueden clasificarse atendiendo los siguientes criterios: estructura de capas, flujo de datos en la red y tipo de respuesta de la red.

Capa de entrada: Esta capa es responsable de recibir información (datos), señales, características o mediciones del entorno externo. Estas entradas por lo general se normalizan dentro de los límites producidos por las funciones de activación. Esta normalización da como resultado una mejor precisión numérica para las operaciones matemáticas realizadas por la red [36, p. 21]. El número y tipo de neuronas que constituyen esta capa depende de los datos del problema.

Capas intermedias: Se constituye de una o más capas (dependiendo de la complejidad del problema) compuestas por un conjunto de neuronas. Estas son responsables de recibir la información (datos de entrada) y entregar como resultado los datos de salida. Las capas ocultas realizan la mayor parte del procesamiento interno desde una red [36, p. 22].

Capa de salida: La capa también está compuesta de neuronas y estas son responsables de producir y presentar las salidas finales de la red, que son resultados del procesamiento realizado en la capas anteriores [36, p. 22]. El número de neuronas de esta capa depende del formato esperado de salida.

Pesos sinápticos: El peso sináptico w_{ij} define la fuerza de conexión entre dos neuronas, la neurona presináptica i y la neurona postsináptica j (ver Figura 8). Estos pesos pueden tomar valores negativos, positivos o ceros. En una entrada positiva, un peso positivo actúa de excitador (neurona activa) mientras un peso negativo actúa de inhibidor (neurona inactiva). Un peso cero significa que no hay conexión entre el par de neuronas. El ajuste peso sináptico w y el umbral w_o sucede etapa del entrenamiento done la red es capaz de adaptarse a cualquier entorno y dar solución a un problema determinado.

Función de activación: También función de transferencia calcula el estado de actividad de una neurona. Transforma la entrada global en un valor de activación cuyo rango normalmente va de (-1 a 1) dependiendo del tipo de función. Entonces una neurona puede estar totalmente inactiva (0 o -1) o activa (1). En la siguiente figura se muestran algunas funciones de activación.

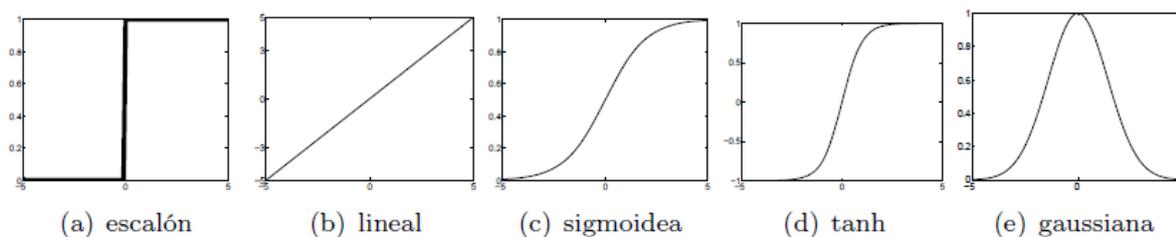


Figura 7: Principales funciones de transferencia utilizadas

Fuente: Stuar and Peter 2007, p621.

a) Según su estructura en capas

Existen las *redes monocapa* que están compuestas por una única capa de neuronas, en las que establecen conexiones laterales (ver Figura 8). Por otro lado, están las *redes multicapa* cuyas neuronas se organizan en más de una capa oculta o intermedia.

b) Según el flujo de datos en la red

La propagación de la red neuronal puede darse en cualquiera de los dos sentidos: hacia adelante o hacia atrás. Cuando es hacia adelante (*feedforward*) conocida como redes unidireccionales donde ninguna salida neuronal es entrada. La información circula en un único sentido, desde las neuronas de la capa de entrada hacia las neuronas de la capa de salida. En cambio, las redes de propagación hacia atrás (*feedback*) las salidas de las neuronas pueden

servir de entradas. Las redes *feedback* que presentan circuitos cerrados se denominan sistemas recurrentes.

c) Según el tipo de respuesta de la red

Se distinguen las redes *heteroasociativas* que son aquellas entrenadas con un patrón A, el sistema responde con otro diferente B. Estas redes presentan al menos dos capas, una entrada para captar información y una salida con la información asociada. Por otro lado, están las redes *autoasociativas* que son entrenadas para que asocien un patrón consigo mismo, es decir, ante la presentación de un patrón A' (afectado por ruido) su respuesta sea el patrón original A. Estas redes pueden implementarse con una única capa de neuronas. A continuación, laElaboración propia. En base a Kim Phil 2017, p20.

Tabla 9 resume los modelos más conocidos de redes heteroasociativas y autoasociativas desarrollados.

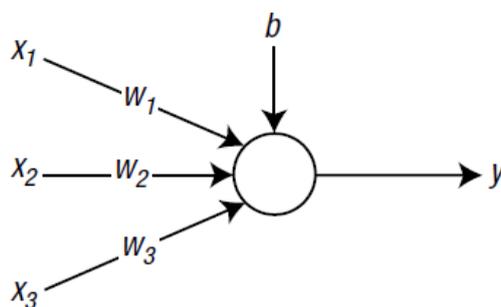


Figura 8: Una neurona que recibe tres entradas

Fuente: Elaboración propia. En base a Kim Phil 2017, p20.

Tabla 9: Redes heteroasociativas y autoasociativas

Tipo de red	Modelos	
Redes heteroasociativas	<ul style="list-style-type: none"> ▪ Perceptron ▪ Adaline/Madaline ▪ Backpropagation ▪ Linear Rewar Penalty ▪ Associative Reward Penalty ▪ Adaptive Heuristic Critic ▪ Boltzmann Machine ▪ Cauchy Machine ▪ Learning Matrix ▪ Temporal Associative 	<ul style="list-style-type: none"> ▪ Linear Associative Memory ▪ Optimal Linear Associative Memory ▪ Drive-Reinforcement ▪ Fuzzy Associative Memory ▪ Counterpropagation ▪ Bidirectional Associative Memory (BAM) ▪ Adaptive BAM ▪ Cognitron/Neocognitron ▪ Learning Vector Quantizer ▪ Adaptive Resonance Theory
Redes autoasociativas	<ul style="list-style-type: none"> ▪ Brain-State-in-a-box ▪ Hopfield ▪ Optimal Linear Associative Memory ▪ Additive Grossberg ▪ Shunting Grossberg ▪ Self-Organization Feature Map (SOFM) 	

Fuente: Elaboración propia. En base a Levy, Flores y Fernández 2008: p. 31

▪ **Aprendizaje y entrenamiento de una red neuronal**

Una de las principales características de una red neuronal es su capacidad de aprendizaje. Este permite que la red modifique su propia estructura (matriz de pesos) adaptándola hasta conseguir un algoritmo de ejecución. Tal algoritmo se consigue a partir de los datos obtenidos de la solución de ejemplos con los que se realiza el entrenamiento de la red [39]. Entonces es necesario hacer una distinción entre los conceptos de aprendizaje y entrenamiento.

El aprendizaje es un procedimiento externo a la red que consiste en hacer cambios en los pesos entre las conexiones de las neuronas de la red hasta conseguir hasta conseguir la respuesta deseada. En cambio, el entrenamiento es el procedimiento por la cual la red aprende, siendo un proceso interno [39]. Por tanto, el objetivo del entrenamiento es enseñar con datos a la red y esta aprenda como resultado final de ese proceso. En ese contexto, se consideran dos categorías de entrenamiento.

a) Aprendizaje supervisado

Es el tipo de tarea de aprendizaje automático más desarrollado que utiliza redes neuronales cuyo objetivo es aprender cierto mapeo de señales de entrada y salida correspondiente basado en datos disponibles [43]. Para ello, requiere una tabla con datos de entrada/salida también llamada tabla de atributos/valores que representa el proceso y su comportamiento, todo ello con el fin de que las estructuras neuronales formulen una “hipótesis” sobre el sistema que se está aprendiendo. Su aplicación depende de la disponibilidad de la tabla mencionada y se comporta como si un entrenador estuviera enseñando a la red cuál es la respuesta correcta para cada muestra presentada para su entrada [36]. La primera estrategia de aprendizaje supervisado fue propuesta por Donald Hebb en 1949, inspirada en observaciones neurofisiológicas.

b) Aprendizaje no supervisado

Es el tipo de aprendizaje que requiere la menor cantidad de supervisión; es decir, el proceso de aprendizaje no es inspeccionado ni controlado por ningún agente externo, cuyo objetivo es encontrar la subyacente estructura en los datos. Esto quiere decir que, a diferencia del aprendizaje supervisado, la aplicación de un algoritmo no supervisado no requiere ningún conocimiento de los resultados deseados (datos o vector de salida) partiendo únicamente de un vector de entrada. Se necesita gran cantidad de vectores de entrada, de tal manera que la red dispondrá de números suficiente de patrones y se obtendrá resultados de calidad [38].

Grey Wolf Optimization

El algoritmo grey wolf optimization (GWO) fue desarrollado por Mirjalili, Mirjalili y Lewis [31], está inspirado en lobos grises (grey wolves), dado que imita la jerarquía de liderazgo y el mecanismo de casa de lobo gris en la naturaleza. Se emplean cuatro tipos de lobos en diferentes niveles, de acuerdo con la jerarquía se denominan alpha wolf (α), beta wolf (β), the delta wolf (δ) and the omega wolf (ω) como se muestra en la Figura 9.

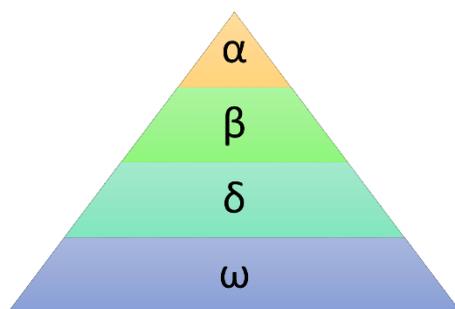


Figura 9: Jerarquía del lobo gris (el dominio disminuye de arriba hacia abajo)

Fuente: Elaboración propia

En términos matemáticos el modelo GWO se basa en las siguientes suposiciones: ⁽¹⁾ α , β y δ denotan la mejor solución, la segunda mejor solución y la tercera mejor solución, respectivamente. ⁽²⁾ ω , representa el resto de soluciones. ⁽³⁾ Los tres mejores lobos ⁽¹⁾ tienen un mejor conocimiento sobre la ubicación potencial de la presa que los lobos ω . ⁽⁴⁾ ω , siguen a los tres mejores lobos. Por lo tanto, como el ω permanece en el nivel más bajo, no tiene seguidores. La caza, ahora está liderada por lobos alfa, beta, omega y lobos tranquilos (omega).

El movimiento de toda población se guía por las tres mejores soluciones del problema de optimización. El proceso de GWO incluye los siguientes pasos.

Paso 1.- Encierro de la presa, el primer paso de caza es rodear la presa El proceso de cerco de Gray Wolf es equivalente a la solución óptima que está rodeada por toda la población y está dada por:

$$\vec{D} = |\vec{C} \cdot \vec{X}_{prey}(i)| - \vec{X}_{wolf}(i) \quad 2.1$$

$$\vec{X}_{wolf}(i+1) = \vec{X}_{prey}(j) - \vec{A} \cdot \vec{D} \quad 2.2$$

Donde:

i representa el número de iteraciones

\vec{A} y \vec{D} son el lobo y vectores de posición de presa respectivamente. Los vectores coeficientes \vec{A} y \vec{C} se calculan a continuación:

$$A = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad 2.3$$

$$A = 2\vec{r}_2 \quad 2.4$$

Los valores del vector \vec{a} disminuye linealmente de 2 a 0 con las iteraciones y \vec{r}_1, \vec{r}_2 vectores aleatorios dentro del intervalo de [0, 1]. La Figura 10 muestra cómo un agente de búsqueda actualiza su posición de acuerdo con alfa, beta y delta en un espacio de búsqueda 2D. Se puede observar que la posición final estaría en un lugar aleatorio dentro de un círculo el cual está definido por las posiciones de α, β y δ en el espacio de búsqueda. En otras palabras, alfa, beta y delta estiman la posición de la presa, y otros lobos actualizan sus posiciones al azar alrededor de la presa.

Paso 2.- Caza, los lobos tienen la capacidad de ubicar su presa y rodearlos, pero la solución óptima no se conoce en el problema de optimización. Por lo tanto, los autores de GWO suponen que el lobo alfa, el lobo beta y el lobo delta tienen un mejor conocimiento de la ubicación potencial de la presa para simular. Matemáticamente, esta búsqueda obtiene primero las tres mejores soluciones y luego actualiza otras soluciones.

$$\vec{X}_{wolf}(i+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad 2.5$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha), \quad \vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}_{wolf}| \quad 2.6$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta), \quad \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}_{wolf}| \quad 2.7$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta), \quad \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}_{wolf}| \quad 2.8$$

Paso 3.- Atacar las presas (explotación), el lobo comienza apretar el agarre para cazar y entonces el movimiento de la presa es más y más pequeño a medida que los lobos se mueven. Termina, la presa deja de moverse, cuando el lobo realiza el ataque final. En un modelo matemático, el valor del vector (a) se reduce de dos a cero por cada iteración en el movimiento lineal de la presa y el lobo (posición de la población) y, finalmente, la presa (óptima) obtiene el movimiento de la presa. La Figura 11 muestra un diagrama de flujo del algoritmo GWO.

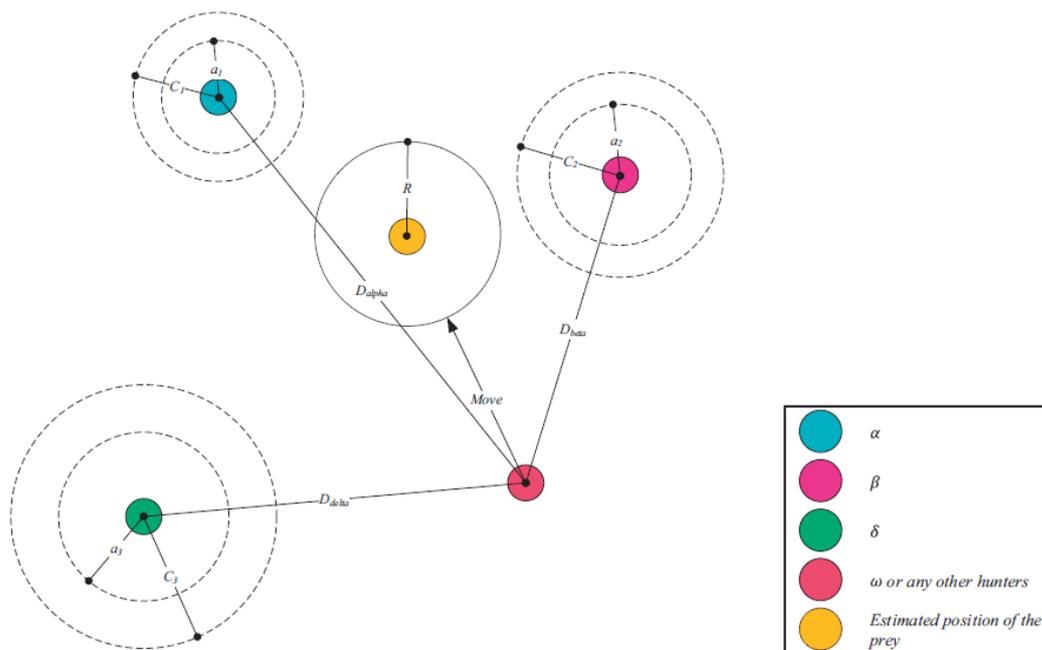


Figura 10: Actualización de posición en GWO

Fuente: Mirjalili, Mohammad y Lewis 2014: p. 50

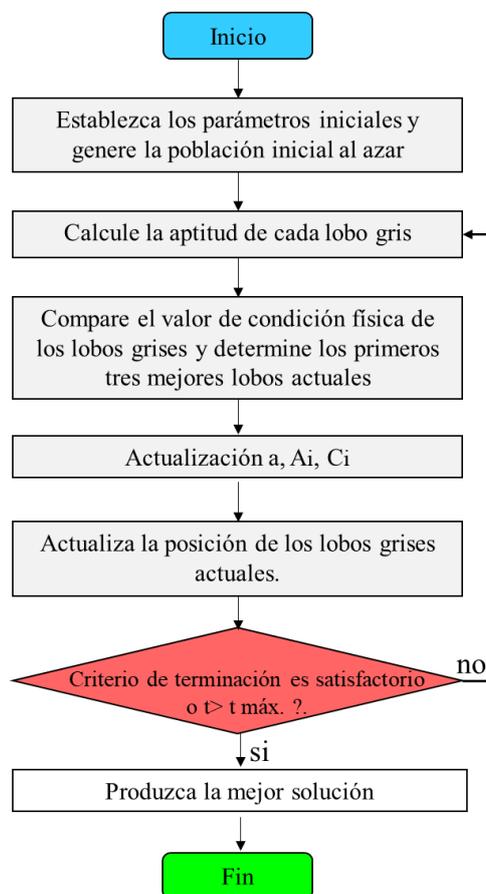


Figura 11: Diagrama de flujo para el algoritmo GWO

Fuente: Elaboración propia. En base a Kumar y Akhata 2019: p. 54

Optimización estructural

La optimización se basa en lograr el mejor resultado de una operación dada al mismo tiempo que se satisface ciertas restricciones. Su estructura está compuesta por una función objetivo que depende de una o más variables y pueden estar sujetas a restricciones que limitan su espacio de búsqueda [44]. Entonces el diseño óptimo de estructuras es lograr el mejor diseño posible de acuerdo a una medida de efectividad preseleccionada [45].

La Figura 12 compara el diseño tradicional con el diseño óptimo; el primero, busca comprobar los criterios de desempeño y, en caso de no ser satisfactorios, el diseño es alterado basado en la experiencia hasta que se comprueben dichos criterios. Sin embargo, el proceso se limita únicamente a cumplir los criterios mas no a encontrar una solución óptima. El segundo, si la convergencia de soluciones óptimas no es satisfactoria, se altera el diseño mediante un algoritmo de optimización.

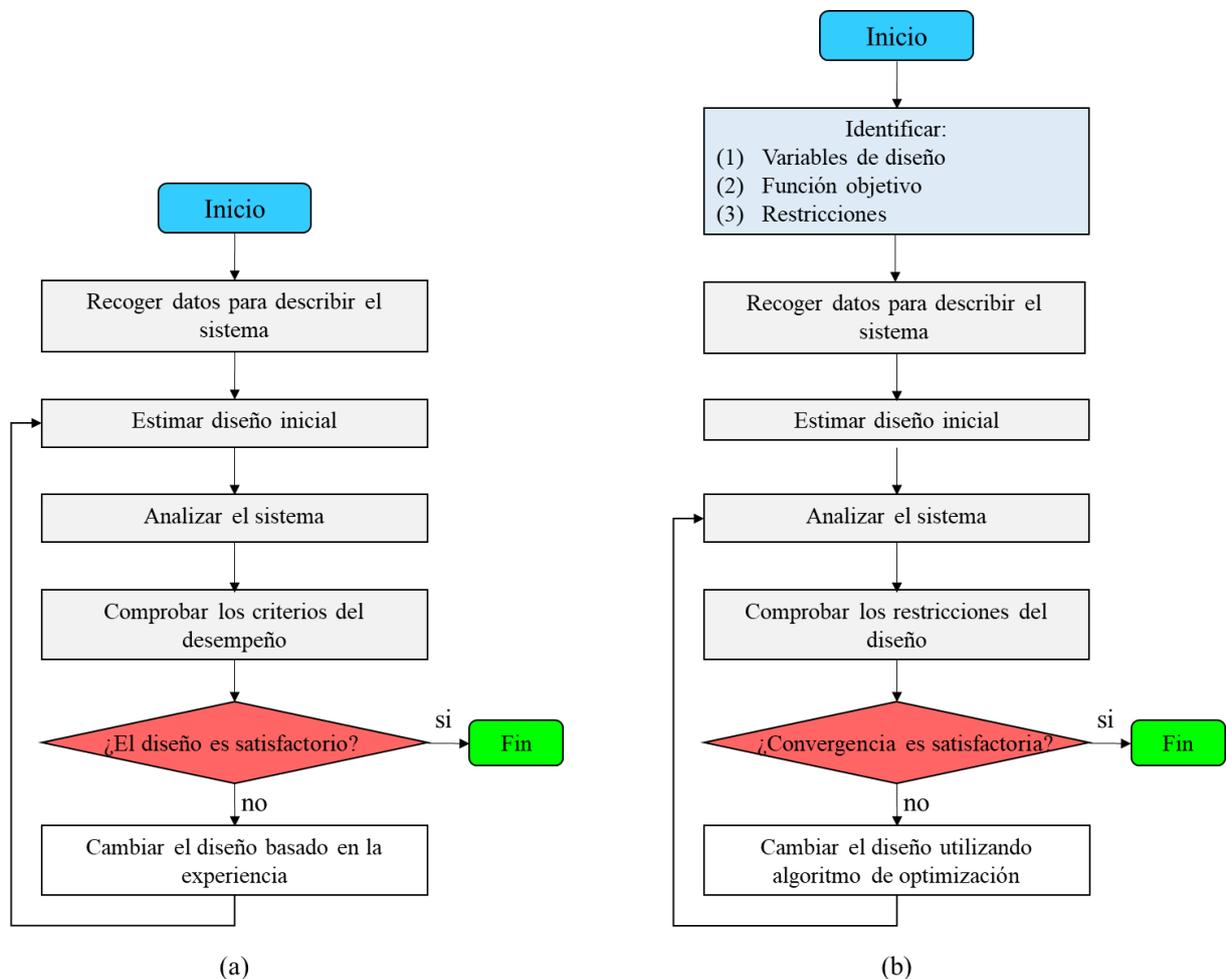


Figura 12: Comparación entre el proceso de diseño (a) tradicional y (b) óptimo

Fuente: Elaboración propia. En base a Fernández 2017: p. 45

Definición de elementos del problema de optimización

- *Variables del diseño*: son aquellas que cambian durante el proceso de optimización. El cambio potencial generalmente se expresa en términos de rangos permitidos [46]. En el campo de la ingeniería algunas variables pueden ser el área de refuerzo y las dimensiones de los elementos estructurales a optimizar .
- *Función objetivo*: representa lo que se pretende optimizar (minimizar o maximizar) y puede ser utilizado como eficiencia del diseño. La optimización con un objetivo se denomina optimización simple, en cambio cuando tenga más de dos objetivos se llama optimización multiobjetiva [46]. Las funciones objetivos del diseño de estructuras de concreto armado pueden ser las presentadas en la figura 12.
- *Restricciones*: son funciones que describen situaciones no deseadas y pueden ser igualdades o desigualdades. Las restricciones imponen límites inferiores y superiores reduciendo la población de soluciones de un problema de optimización [46]. Cuando se trate de hormigón armado, por ejemplo, las restricciones son valores límites para las áreas de refuerzo, ancho, altura, esfuerzos, entre otras limitaciones. Estas se encuentran en las diferentes normativas y códigos para diseñar.
- *Espacio de búsqueda*: es la población de soluciones posibles y viables al problema, con sus límites impuestos por las restricciones. Aquí es donde se debe encontrar la solución óptima.

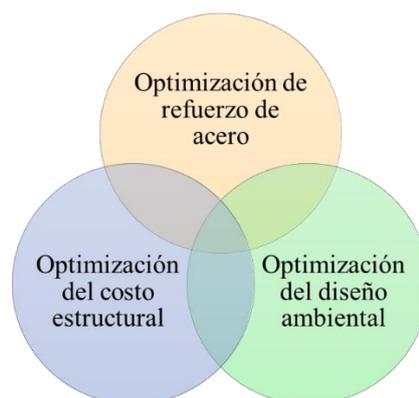


Figura 13: Objetivos de optimización del diseño estructural de concreto armado

Fuente: Elaboración propia. En base a Afzal, et al. 2020

Representación del problema de optimización

Un problema de **optimización simple**, matemáticamente puede ser definido de la siguiente forma [46]:

Minimizar $f(x)$:

Tal que $g_j(x) \geq 0, \quad j = 1, \dots, n_g, \quad 2.9$

$h_k(x) = 0, \quad k = 1, \dots, n_e,$

Donde:

x es un vector de las variables de diseño con las componentes $x_i, i=1, \dots, n$

$h_k(x)$ y $g_j(x)$ son las restricciones de igualdad y desigualdad, respectivamente.

Un problema de **optimización lineal** se representa matemática como la siguiente ecuación, donde función de variables y restricciones son lineales. Este tipo de problemas son solucionados por una rama de la programación matemática llamada programación lineal [46].

$$f(x) = c_1x_1 + c_2x_2 + \dots + c_nx_n = c^T x \quad 2.10$$

Asimismo, un problema de **optimización multiobjetivo** donde hay más de una función objetivo a la que se puede llamar vector puede formularse de la siguiente manera [47]:

Minimizar $f(x) = \{f_1(x), f_2(x), \dots, f_o(x)\}$

Tal que $g_j(x) \geq 0, \quad j = 1, \dots, n_g, \quad 2.11$

$h_k(x) = 0, \quad k = 1, \dots, n_e,$

Donde:

x es un vector de las variables de diseño con las componentes $x_i, i=1, \dots, n$

$h_k(x)$ y $g_j(x)$ son las restricciones de igualdad y desigualdad, respectivamente.

De acuerdo con los problemas de optimización estructural, este divide en tres clases. **Optimización del tamaño:** las variables son las dimensiones y la forma de las secciones transversales de los elementos estructurales. **Optimización de forma:** se refieren a los cambios en el posicionamiento nodal y la superficie / curvatura del elemento. **Optimización topológica:** se refiere a la colocación de elementos estructurales y material en relación con el flujo de fuerzas y niveles de tensión. Este último, es el tipo de optimización más amplio y el modelo consta de tres formulaciones: *función objetivo, restricciones y espacio de búsqueda* [48].

Técnicas y métodos de optimización

En la actualidad, existen varias técnicas y métodos de optimización disponibles para resolver problemas de ingeniería estructural. Hasta los años 80 solamente se resolvían problemas de

optimización a pequeña escala debido a la falta de tecnología y herramientas poco eficientes y robustas. Después de esa década se introdujeron nuevos avances en los métodos de optimización, tales como: programación lineal, programación no lineal y programación no lineal secuencial que han estado por muchos años en práctica [49]. También otras técnicas desarrolladas son los algoritmos evolutivos, algoritmos de inteligencia de enjambre (ver Tabla 6 y 7).

La elección de la técnica de optimización depende de la naturaleza del problema. Las técnicas de optimización metaheurística se han utilizado en más ocasiones en la literatura, y sobre todo los algoritmos genéticos. De acuerdo a la revisión de Afzal, et al. [49] la preferencia de las técnicas de optimización por los investigadores es la que se muestra en la Figura 14. Además, identifica cuatro categorías del proceso de optimización de estructuras de concreto armado según el objetivo común que persigue cada estudio (ver Figura 13).

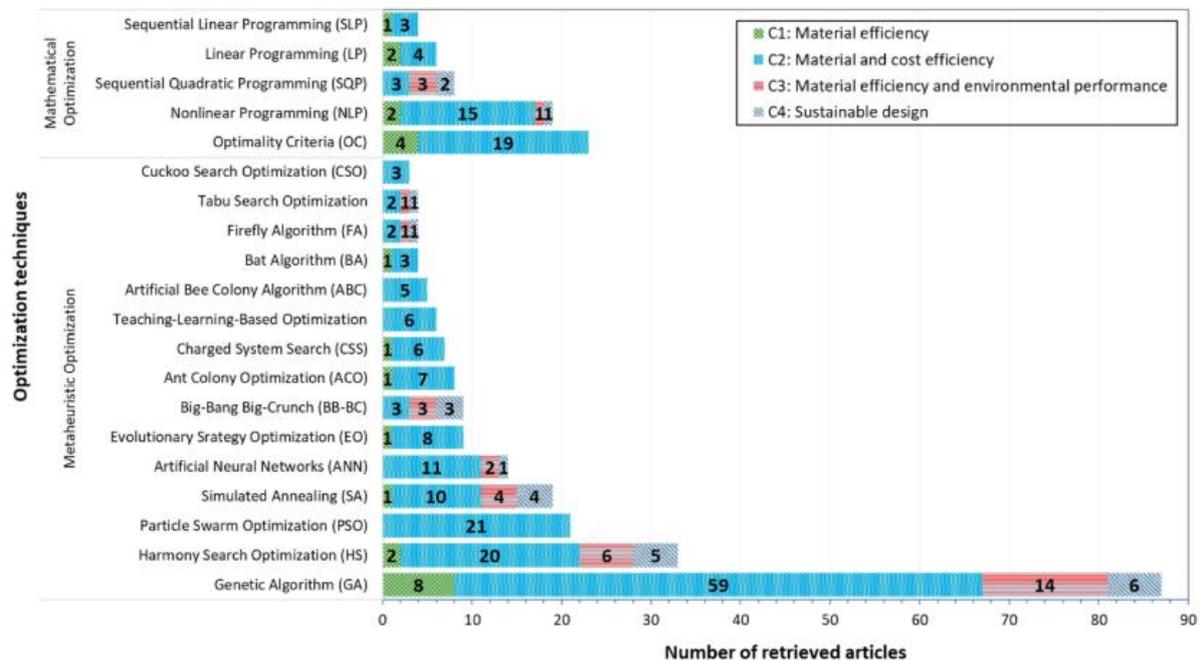


Figura 14: Número de artículos recopilados con técnica de optimización adoptada en cada categoría de tema de investigación

Fuente: Afzal, et al. 2019: p. 67

III. Metodología

Tipo y nivel de investigación

El tipo de investigación será básica o pura de acuerdo a la finalidad perseguida, puesto que está orientada a aportar datos y características para construir losas aligeradas y macizas no preesforzadas con apoyos simples, empotrados y en voladizo. Tales losas sean óptimas y tengan un comportamiento estructural adecuado a diversas cargas distribuidas que son sometidas según el uso de la edificación. De esta forma, se buscó encontrar respuestas que puedan dar solución a una problemática del sector de la construcción civil.

La investigación es cuantitativa de acuerdo al enfoque del estudio, puesto que utiliza la estadística y se prueba una hipótesis mediante un procedimiento secuencial y probatorio [50]. De manera que, se busca corroborar si se logra optimizar el diseño de losas no preesforzadas de concreto armado aplicando la metodología propuesta. Para ello, previamente, el estudio de este tipo de estructuras ha sido delimitado a dos tipologías (losas macizas y aligeradas) con resistencia específica a compresión del concreto de 210 kg/cm².

La investigación es no experimental de acuerdo al diseño, puesto que la investigación se realiza sin manipular deliberadamente variables [50]. El manejo de las variables está limitado a las especificaciones de los reglamentos de diseño, ACI 318 y ASCE 7. De modo que, en la optimización de losas no se manipula la información si no que se resuelve utilizando técnicas de optimización. Entonces lograr un eficiente comportamiento estructural de las losas consideradas por la presente investigación.

El nivel de investigación es correlacional de acuerdo al alcance del estudio, cuyo propósito es conocer la relación o grado de asociación entre dos o más conceptos o variables en un contexto específico [50]. Las variables independiente y dependiente con sus dimensiones e indicadores respectivos, considerados, son las que afirman lo anterior. Como variable independiente se tiene la optimización de losas cuyas dimensiones son: variables de diseño y restricciones. Y de variable dependiente es el comportamiento de la losa cuyas dimensiones son: tipo de losa, tipo de apoyo y el tipo de carga.

Diseño de investigación

La hipótesis que se plantea es la siguiente: se logra optimizar el diseño de losas aligeradas y macizas no preesforzadas en un 20% utilizando el algoritmo GWO y una red neuronal artificial

El diseño a emplearse en la presente investigación es **no experimental transeccional** o transversal, cuyo propósito es describir variables y analizar su incidencia e interrelación en un

momento dado. Además, porque se describe la relación entre las variables, la investigación presenta un diseño **transeccional correlacional** [50]. De tal manera que, se asocia la optimización de losas (variable independiente) con el comportamiento estructural adecuado (variable dependiente).

Población, muestra, muestreo

La población de estudio será todo tipo de techos o losas no preesforzadas de concreto armado con distintas resistencias específicas a compresión. Losas con diferentes configuraciones geométricas que soportan cargas de acuerdo al uso de la edificación y cumplen la función de entrepiso de una edificación y de transmitir cargas a las vigas y columnas. Son también toda aquella bibliografía que traten del tema de losas no preesforzadas de concreto armado.

La muestra de esta investigación viene dada por dos tipos de losas no preesforzadas de concreto armado cuyas características son:

- Tipo de losa : aligeradas y macizas
- Tipo de apoyo : simplemente apoyadas, empotradas y en voladizo
- Tipo de carga : distribuida
- Tipo de material : concreto de resistencia específica a compresión de 210 kg/cm²
- Configuración : rectangulares

El tamaño de la muestra es de cincuenta modelos de losas con las características antes descritas que serán diseñadas según la normativa vigente, E-060, ACI 318-19 y ASCE 7-16. Con los resultados (datos) del diseño tradicional se entrena la técnica de optimización utilizada en la presente investigación y esta permita crear diseños óptimos de losas. La muestra también son aquellos papers y tesis desarrollados del año 2016 en adelante que traten el tema de optimización de losas para cotejar con los resultados de este estudio.

El muestreo que ha sido considerado para la presente investigación corresponde a una muestra no probabilística, puesto que la elección de los elementos se ha realizado de acuerdo al juicio y conveniencia del investigador.

Criterios de selección

El criterio para seleccionar el tamaño de muestra de la presente investigación se ha realizado, al tratarse de una muestra no probabilística, por conveniencia y juicio del investigador. Por lo que se ha tomado en cuenta el tiempo y costo del desarrollo del proyecto; y lo revisado en la literatura.

Operacionalización de variables

La Tabla 10 muestra la operacionalización de las variables independiente y dependiente consideradas en la presente investigación.

Tabla 10: Operacionalización de variables

VARIABLE	TIPO	DIMENSIÓN	INDICADOR	INSTRUMENTO
Optimización de losas	Independiente	Variables de diseño	Espesor losa	ACI 318, E-060 MATLAB, SAP 2000
			Diámetro acero de refuerzo	
			Espaciamiento del refuerzo	
		Restricciones	Deflexión	
			Cortante	
			Flexión	
			Serviciabilidad	
Comportamiento de la losa	Dependiente	Tipo de losa	Aligerada	ACI 318, AISC 7, E-060, E-020
			Maciza	
		Tipo de apoyo	Empotrada	
			Simplemente apoyada	
			Voladizo	
		Tipo de carga	Distribuida	

Fuente: Elaboración propia

Métodos, técnicas e instrumentos de recolección de datos

Métodos

El método de diseño se considera según lo estipulado en la norma E.060, donde el método de diseño para estructuras de concreto armado es el **diseño por resistencia**. La resistencia de un elemento en términos de momento, fuerza axial, cortante, torsión y aplastamiento, debe tomarse como la resistencia nominal R_n multiplicada por el factor de reducción de resistencia ϕ . Y las resistencias requeridas, R_u , son calculadas para las cargas según lo especifica la norma E.020 y las fuerzas amplificadas por las combinaciones del ACI 318 y E.060.

$$\phi R_n \geq R_u$$

La resistencia requerida para el diseño de losas se obtiene del análisis estructural con las combinaciones de carga específica en el ACI 318 – 19 y la norma E-060. Asimismo, el factor de reducción de resistencia por flexión es 0.9

$$\text{Ecuación 5.3.1a ACI 318-19 : } w_u = 1.4DL \quad 3.1$$

$$\text{Ecuación 9.1 E-060 R.N.E : } w_u = 1.4DL + 1.7LL \quad 3.2$$

Método de optimización se considera la metaheurística, utilizando un algoritmo de inteligencia de enjambre y una red neuronal. El algoritmo que se ha elegido es el Grey Wolf Optimization (GWO) porque la solución óptima puede evaluarse mediante tres soluciones, a diferencia de los otros que la mejor solución se busca solo con una sola solución. También este algoritmo tiene un alto valor de investigación por ser un desarrollo reciente y no estar aplicado en la optimización de losas (ver Tabla 7). Asimismo, la arquitectura de la red neuronal se elige luego de reconocer todas las variables que participan en el diseño y optimización de losas.

Técnicas de recolección de datos

- Análisis de documentos mediante la interpretación de la información recopilada de diferentes fuentes como: journals, revistas, base de datos, repositorios institucionales y otras fuentes bibliográficas. Los documentos obtenidos son: papers, conferencias, libros, reglamentos, tesis y otra información de alto impacto.
- Observación directa o sistemática mediante la observación de las variables en su contexto natural para obtener la información necesaria para la elaboración del proyecto de investigación.

Instrumentos de recolección de datos

Los instrumentos para las técnicas de recolección de datos se presentan en la siguiente tabla:

Tabla 11: Instrumentos para las técnicas de recolección de datos

<i>Técnica</i>	Instrumento	Elementos de la población
<i>Análisis de documentos</i>	Fichas de análisis (ver Figura 39)	Bibliografía
<i>Observación directa</i>	Programa de diseño (ver Anexo)	Losas aligeradas Losas Macizas

Fuente: elaboración propia

Fuentes y softwares

- Fuentes:
 - ASCE 7 : cargas mínimas de diseño
 - ACI 318 : especificaciones del diseño
 - Journals : Asian J. Civ. Eng, Engineering Structures, otros
 - Editoriales : Elseiver, Springer, IEEE, Hindawi, otros
 - Base datos : scopus, scielo, repositorios de tesis, otros
 - Planos : plantas de edificaciones de concreto armado

○ Softwares:

- Google : para la búsqueda de información
- Autocad : para visualizar, acotar y editar los planos
- SAP 200 : para el modelamiento y análisis estructural
- Excel 2016 : para elaboración de hojas de cálculo
- Matlab : para la programación de la técnica de optimización
- Word 2016 : para la redacción y edición del informe
- Mendeley : para la elaboración de citas

Procedimientos

Recopilación de información

Inicia con la búsqueda de información en las diferentes fuentes electrónicas: base de datos, repositorios institucionales, editoriales, revistas, journals, entre otras. Se ha buscado en estas fuentes las siguientes palabras: “RC slab optimization”, “RC slab optimization”, “Design optimization”, “Review of optimization”, “Review IA in civil engineering”, “Structural optimization”, “Swarm intelligence”. Para organizar toda aquella información se realizó en Mendeley que es una aplicación web y de escritorio; permite gestionar y compartir referencias bibliográficas y documentos de investigación, encontrar nuevas referencias y documentos. Para el análisis de esta información se ha realizado fichas de análisis (ver Figura 39)

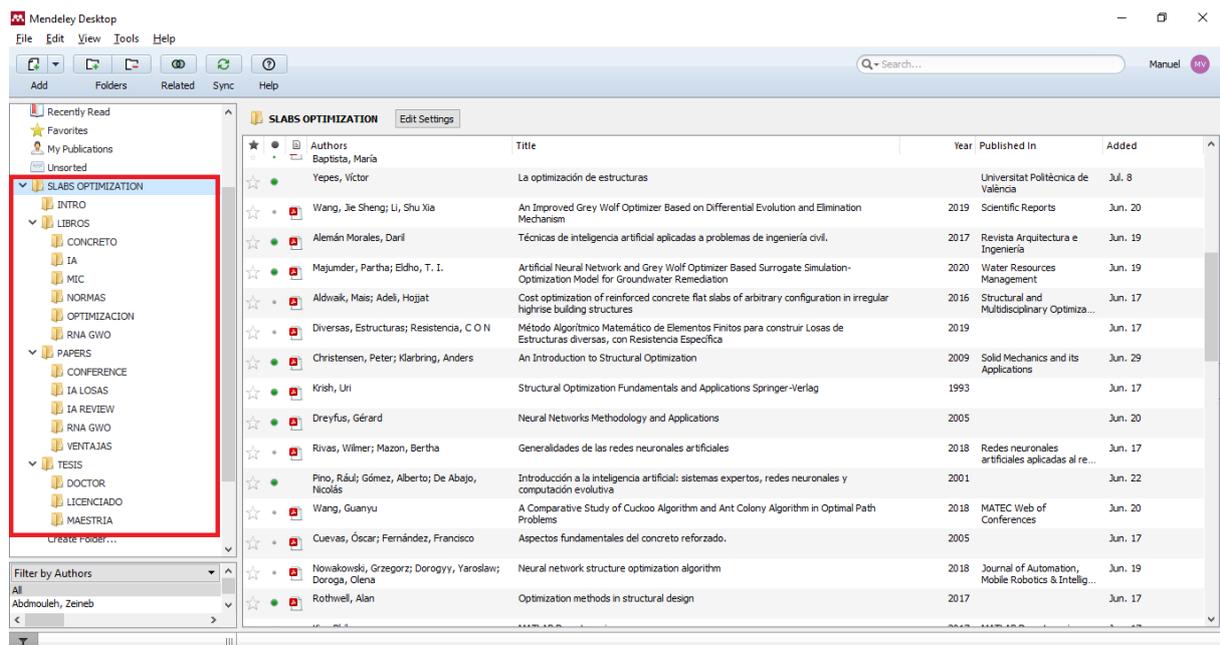


Figura 15: Organización de los documento y referencias

Fuente: Elaboración propia

Análisis Estructural de losas de entepiso

a. Normativa

- Norma NTE E – 020 : Cargas
- Norma NTE E – 060 : Concreto Armado.

b. Definición de las variables constantes

- Tipo de apoyo : Apoyada, apoyada-continua, continua, voladizo
- l_n : Distancia libre entre apoyos
- f_c : Resistencia a compresión del concreto
- f_y : Resistencia a la fluencia del acero
- rec : Recubrimiento de la estructura
- w_g : Peso del concreto por unidad de volumen
- w_s : Peso del acero por unidad de volumen
- DL : Carga muerta
- LL : Carga viva

c. Metrado de y combinación cargas

La carga muerta (DL) o permanente está definida por el peso propio de la estructura, el peso de los acabados y una carga mínima repartida que representa a la tabiquería. La carga viva (LL) o sobrecargas varían su magnitud de acuerdo al uso de la edificación. Estas pueden ser de uso privado, como viviendas, hasta el uso público como son los colegios, bibliotecas, otros. En la norma NTE E-020 se puede encontrar los pesos unitarios de los materiales y las cargas vivas mínimas repartidas (Ver Tabla 30: Pesos unitarios de los materiales de construcción y Tabla 31: Cargas vivas mínimas distribuidas uniformemente).

Según la sección de la losa se calcula el área y esta se multiplica por su peso unitario del concreto para obtener el peso de la estructura por un metro lineal. En la Figura 16(a) el caso de una vigueta Tee en extremo y para la Figura 16(b) una viga Tee central. Asimismo, la geometría de los aligerados típicos en nuestro medio se resume en la Figura 17: Geometría de los aligerados típicos El espaciamiento libre entre vigas Tee de 30.0 cm obedece a las dimensiones del ladrillo para techo que se fabrica en nuestro medio [51].

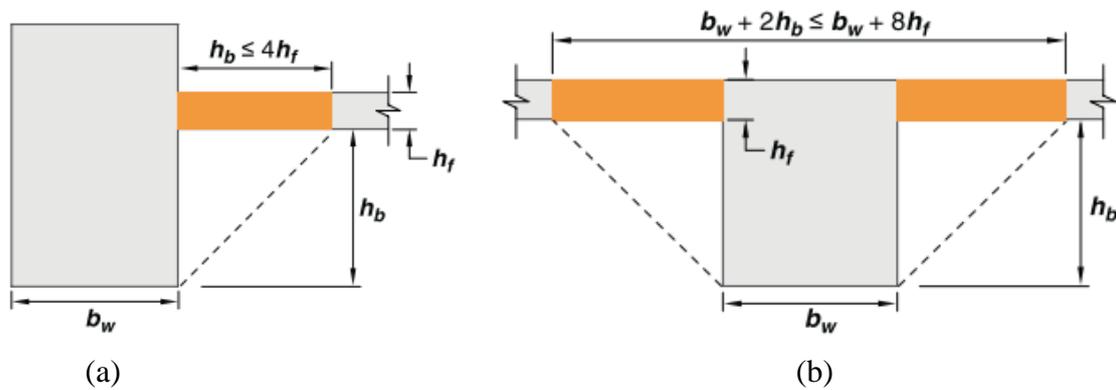


Figura 16: Ejemplos de la porción de losa que debe incluirse con la viga

Fuente: ACI 318 2019: p. 104

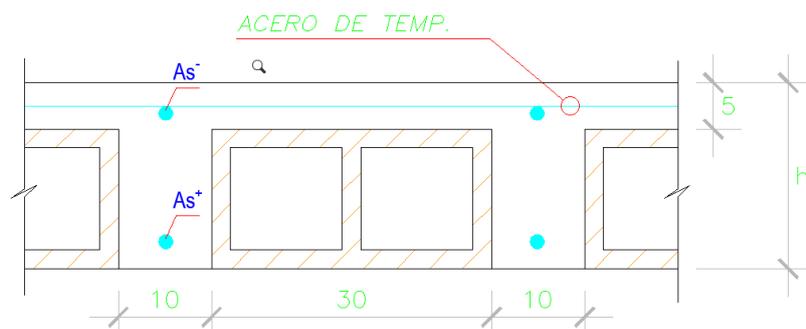


Figura 17: Geometría de los aligerados típicos

Fuente: Elaboración propia

d. Restricciones por la NTE E - 060

▪ Flexión: $\phi M_n \geq M_u$ 3.3

En el caso de secciones Tee para momentos positivos, al estar en compresión existen dos posibilidades de análisis de esfuerzos utilizando la sección transformada agrietada. La primera posibilidad, Figura 18(a), es que el bloque de compresión sea en el espesor del ala ($a \leq h_f$), entonces la sección se analiza como si fuera rectangular de ancho ($b_w = b$). La otra posibilidad, Figura 18(b), el bloque de compresión esté en el alma ($a > h_f$), se analiza como viga T [51]. Para determinar la profundidad del bloque en compresión (a) [17] se asume, inicialmente, que sólo el ala de la sección contribuye a la resistencia y se tiene que:

$$A_s = \frac{M_u}{\phi f_y \left(d - \frac{h_f}{2} \right)} \quad 3.4$$

$$a \approx \frac{A_s f_y}{0.85 f'_c b} \quad 3.5$$

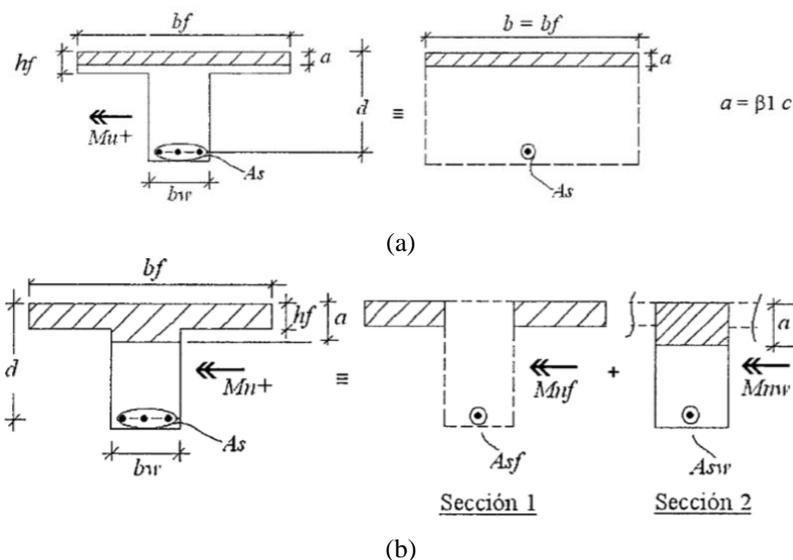


Figura 18: Posición del bloque de compresión vigas Tee

Fuente: Ottazzi 2015: p. 229

Caso a)
Conozco M_u , calculo A_s

$$0 = \frac{a^2}{2} - ad + \left[\frac{M_u}{\phi \cdot 0.85 \cdot f'_c \cdot b} \right] \quad 3.6$$

$$A_s = \left[\frac{M_u}{\phi \cdot f_y \left(d - \frac{a}{2} \right)} \right] \quad 3.7$$

Caso b)
Conozco M_u , calculo A_s

$$a = A_s \cdot \frac{f_y}{\phi \cdot 0.85 \cdot f'_c \cdot b} \quad 3.8$$

$$M_n = A_s \cdot f_y \left(d - \frac{a}{2} \right) \quad 3.9 \quad \phi = 0.9$$

▪ Cortante: $\phi V_n \geq V_u \quad 3.10$

El cortante mayorado, V_u , debe ser resistido por la capacidad a corte que aporta el concreto de la losa V_c , debido a que no tiene estribos. La V_c se calcula según la ecuación 22.5.5.1(a) ACI 318-19. Al tratarse las losas de elementos en flexión pura, la carga axial $N_u = 0 \text{ kg}$ y $\lambda = 1.0$ para un concreto de peso normal. La V_u se obtiene de las combinaciones de carga, considerándose el mayor resultado. Por lo tanto, la sección de la losa debe cumplir $\phi V_c \geq V_u$ de lo contrario se recomienda ensanches por cortante, incrementar peralte y/o mayor resistencia a la compresión. En el caso de nervaduras el V_c de 3.11 podrá considerarse 10% mayor (8.11.8 E-060).

$$\phi V_n = \phi V_c = \phi 0.53x \sqrt{f'_c} b_w d \quad 3.11 \quad \phi_V = 0.85$$

▪ Servicio $\rho_{min} \geq \rho \geq \rho_{max} \quad 3.12$

$$A_s \geq A_{min} \quad 3.14$$

$$\rho_{min} = \begin{cases} 0.0020 ; & \text{barras corrugadas } f_y < 4200 \frac{\text{kg}}{\text{cm}^2} \\ 0.0018 ; & \text{barras corrugadas } f_y \geq 4200 \frac{\text{kg}}{\text{cm}^2} \end{cases} \quad 3.15$$

$$\rho_{max} = 0.75\rho_b = 0.75\beta_1 \frac{f'_c}{f_y} \left[\frac{6000}{(6000+f_y)} \right] \quad 3.16$$

$$\beta_1 = \begin{cases} 0.85 & ; f'_c < 280 \text{ kg/cm}^2 \\ 0.85 - \frac{0.55(f'_c-280)}{70} & ; f'_c < 550 \text{ kg/cm}^2 \\ 0.65 & ; f'_c > 550 \text{ kg/cm}^2 \end{cases} \quad 3.17$$

- Deflexión

De acuerdo a la NTE E-060 el peralte mínimo para las losas está en función de la luz, condiciones de apoyo, el esfuerzo de fluencia del acero, la orientación de sus planos (en una o dos direcciones) y el tipo de sistema constructivo (Ver Tablas 1, 2, 3 y 4). Asimismo, de la Figura 15 el espesor de la losa superior (hf) no debe ser menor que la doceava parte de la separación de nervaduras o 5.00 cm.

- Otras restricciones

En losas nervadas se considera lo siguiente:

- El ancho de las nervaduras no debe ser menor de 100 mm y debe tener una altura no mayor de 3,5 veces su ancho mínimo.
- El espaciamiento libre entre las nervaduras no debe exceder de 750 mm [18].

e. Modelamiento asistido por computadora

Se modelan diferentes modelos de losas en el software SAP 2000 para obtener sus esfuerzos últimos, como el momento de flexión y cortante producidos por la alternancia de cargas vivas, el peso propio y otras cargas muertas de acabados y/o tabiquería (DLs). En la Figura 19: Alternancia de carga viva de 200 kg/m² en un modelo de losa aligerada unidireccional. se muestra un ejemplo de alternancia de carga viva de 200 kg/m², que se distribuye en un aligerado unidireccional para un ancho tributario de 0.40 m.

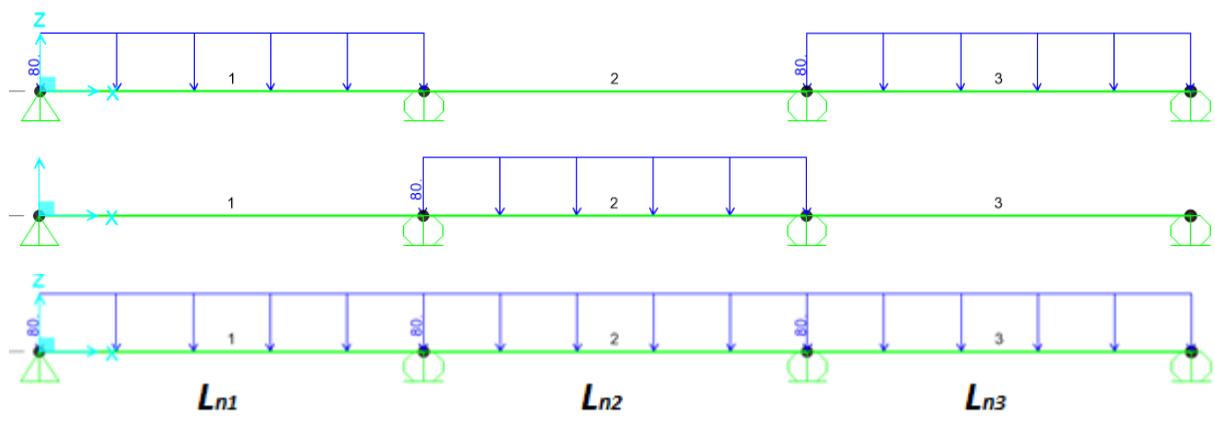


Figura 19: Alternancia de carga viva de 200 kg/m² en un modelo de losa aligerada unidireccional.

Fuente: Elaboración propia

Criterios de optimización

Algoritmo Grey Wolf Optimizer

a. Función objetivo $f(x_i)$

$$f(x_i) = \text{mínimo (Costo (C))}$$

$$C = \text{Costo concreto (Cc)} + \text{Costo acero (Cs)} + \text{Costo construcción (Ccon)}$$

$$Cc = (\text{Volumen de Concreto (Vc)}) (\text{Precio Unitario del concreto (PUc)})$$

$$Cr = (\text{Peso del Acero (Ps)}) (\text{Precio Unitario del acero (PUs)})$$

$$Ccon = \text{Ratio de Construcción}$$

b. Variables x_n

$$x_1 = \text{Espesor del ala de la viga (hf)}$$

$$x_2 = \text{Peralte del alma de la viga (hb)}$$

$$x_3 = \text{Ancho del alma de la viga (bw)}$$

$$x_4 = \text{Separación de las vigas (S)}$$

$$x_5 = \text{Diámetro del refuerzo (db)}$$

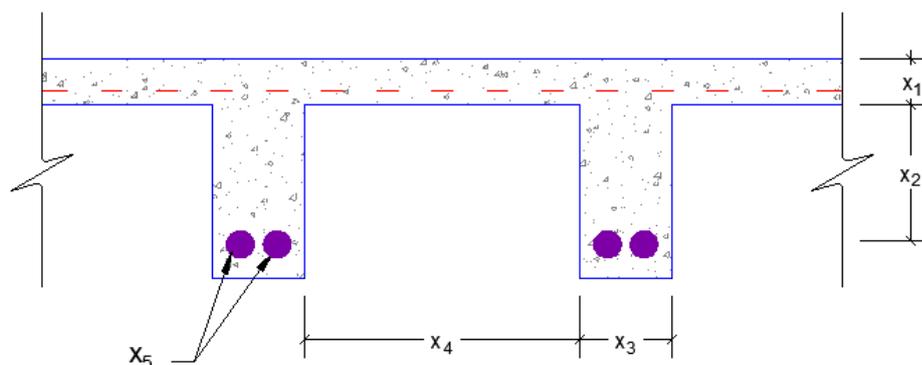


Figura 20 Sección de una losa nervada y el diseño de las variables seleccionadas

Fuente: Elaboración propia

c. Restricciones $g_j(x) \leq 0$, $h_k(x) = 0$

De Flexión

De Cortante

De Servicio

De Deflexión

Otras

Red Neuronal

a. Entradas y salidas

Son los datos del diseño convencional de losas aligeradas. Se obtendrá un vector de variables constantes que son las limitaciones y tienen un único valor, un vector de variables de entrada que son variables con un rango de valores y un vector de salida que son el resultado de las variables de entrada.

b. Arquitectura de la red neuronal

Al tratarse de un problema de regresión, se utiliza redes neuronales simples de ya que solo se entrenará con valores numéricos.

c. Algoritmo de entrenamiento

Algoritmos *backpropagation (BP)* entrena a una red neuronal de retro alimentación (*feedforward*), permite actualizar los pesos sinápticos moviéndose hacia adelante y hacia atrás hasta que la función de error quede atrapada en su mínimo local. El entrenamiento de la red será el supervisado, ya que ingresamos un vector con valores y se busca obtener de salida otro vector deseado.

d. Número de neuronas

El número de neuronas se determinará mediante en entrenamiento de la red hasta conseguir resultados eficientes.

Plan de procesamiento y análisis de datos

La secuencia del desarrollo del proyecto consta de cuatro fases que empieza con la recopilación de información, diseño tradicional, programación del algoritmo y finalmente el análisis de resultados y conclusiones.

FASE I: Recopilación de información, en esta fase se recopila toda la información relacionado con el tema de investigación. Haciendo una revisión de las técnicas de inteligencia artificial aplicado a la ingeniería estructural. Asimismo, se obtendrá los planos arquitectónicos de las losas de entrepiso. Y con el estudio de normativa nacional e internacional se definen las variables de diseño de este tipo de estructuras.

FASE II: Diseño de losas de concreto armado, se inicia con la elaboración de una hoja de cálculo en Excel y el pre-dimensionamiento de estos elementos estructurales. Con los resultados del análisis estructural en SAP 2000 se diseñan tradicionalmente las losas macizas y aligeradas.

FASE III: Desarrollo del programa de optimización, esta fase inicia con la definición de la arquitectura de la red neuronal y la definición de la función objetivo para la optimización por el algoritmo grey wolf optimization para posterior empezar a escribir el programa en lenguaje Matlab. Se entrena el modelo propuesto con los diseños de la anterior fase y se valida mediante la depuración y pruebas.

FASE IV: Análisis de resultados y conclusiones, luego de ejecutar el programa, en esta fase se discuten los resultados y se comparan con investigaciones previas para finalmente redactar las conclusiones y recomendaciones de la investigación.

Matriz de consistencia

FORMULACIÓN DEL PROBLEMA	OBJETIVOS	VARIABLES Y DIMENSIÓN	INDICADORES	METODOLOGÍA TÉCNICAS E INSTRUMENTOS
¿Cuánto sería el porcentaje de optimización del diseño de losas no preesforzadas de grandes luces utilizando el algoritmo GWO y una red neuronal?	<p>Objetivo General: Optimizar el diseño de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado utilizando el algoritmo GWO y una red neuronal.</p> <p>Objetivos Específicos: O1.- Diseñar de forma tradicional losas de entrepiso macizas y aligeradas que se encuentran empostradas, simplemente apoyadas y en voladizo de edificaciones de concreto armado. O2.- Implementar un programa computacional basado en el algoritmo GWO y una red neuronal para el diseño óptimo de losas macizas y aligeradas. O3.- Comparar la eficiencia del programa implementado con el diseño tradicional y los resultados de otras investigaciones.</p>	<p>VARIABLE DEPENDIENTE Comportamiento de la losa</p> <p>Dimensiones Tipo de losa Tipo de apoyo Tipo de carga</p> <p>VARIABLE INDEPENDIENTE Optimización de losas</p> <p>Dimensiones Variables de diseño Restricciones</p>	<ul style="list-style-type: none"> - Aligerada - Maciza - Empotrada - Simplemente apoyada - Voladizo - Distribuida - Espesor losa - Diámetro acero de refuerzo - Espaciamiento del refuerzo - Deflexión - Cortante - Flexión - Serviciabilidad 	<p>Tipo y Diseño de Investigación: Fin que persigue: Básica Enfoque: Cuantitativo Diseño: No experimental Nivel: Correlacional</p> <p>Población y Muestra: La Población: Todo tipo de losas no preesforzadas de concreto armado. Losas de diferentes geometrías que soportan cargas según su uso. La Muestra: Losas aligeradas y macizas.</p> <p>No probabilístico. De acuerdo a juicio y conveniencia propia. Diseño de cincuenta losas rectangulares con carga distribuida</p> <p>Técnica: Análisis de documentos Observación directa</p> <p>Instrumento: - Hojas de registro de datos - Reglamentos - Softwares</p>

Consideraciones éticas

Para realizar la investigación “Optimización de dos tipos de losas no preesforzadas de grandes luces en edificaciones de concreto armado” se tuvo en cuenta desde el primer instante que no exista una investigación anterior respecto al tema. Por lo que, la presente tesis es auténtica y original. Me comprometo a desarrollar el tema de tesis según los objetivos explicados y detallados en el proyecto. Los resultados que se obtendrán del análisis de documentos serán de fuentes veraces y comprometiéndome a cometer plagio con respecto a la información de otros autores. Los antecedentes y marco teórico fueron realizados en base a otras investigaciones sobre el tema, respetando los derechos de autor citando el contenido

IV. Resultados y discusión

Diseño de modelos de entrenamiento de la red neuronal

Para la presente investigación, se realizó el diseño de 50 losas de entrepiso de edificaciones con configuración rectangular. El diseño de estos modelos se utiliza para el aprendizaje de la red neuronal en la etapa de entrenamiento.

Descripción general de los modelos

Las losas de entrepiso de las edificaciones estudiadas son del tipo aligerados y macizas en una dirección y de configuración rectangular. Las edificaciones que se proyectan son para uso de oficinas y aulas como se muestra en la Figura 21; asimismo, al tratarse de la optimización de losas de gran luz, se ha utilizado las losas aligeradas con peralte de 30cm y macizas con peralte según la Tabla 1.

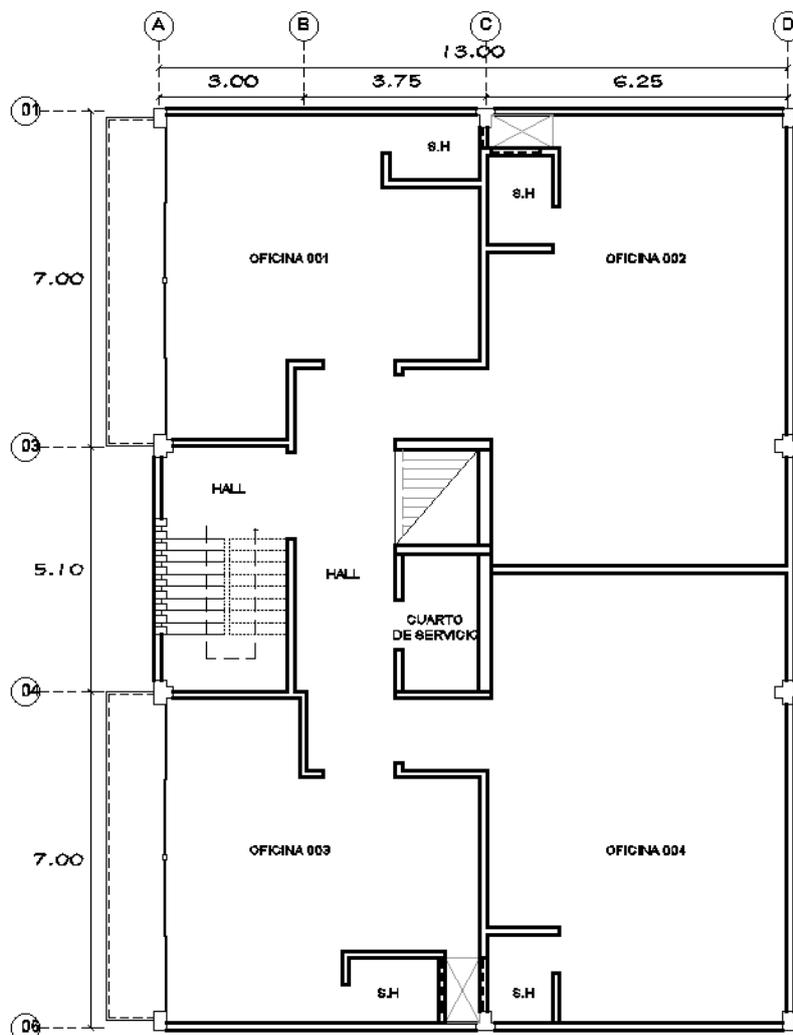


Figura 21: Plano de distribución de oficinas
Fuente: Elaboración propia

- a. Propiedades de los materiales
- Resistencia a compresión del concreto: 210.00 kg/cm²
 - Resistencia a fluencia del acero: 4200.0 kg/cm²
- b. Cargas
- Peso volumétrico concreto: 2400 kg/m³
 - Peso volumétrico de albañilería: 1350 kg/m³
 - Peso aligerado de 30 cm: 420 kg/m²
 - Acabados: 120.00 kg/m²
 - Albañilería: 100.00 kg/cm²
 - Sobrecarga: 250.00 kg/cm²
 - Corredores y escaleras: 400 kg/m²

Estructuración

De acuerdo con el plano de distribución se realiza la estructuración el modelo como se muestra en la Figura 22 para determinar lo siguiente:

- Dirección de las losas aligeradas y/o macizas
- Condiciones de contorno de cada paño
- El número de tramos continuos
- Longitud total de los tramos continuos
- La luz libre de cada tramo de losa
- Modelamiento estructural

Metrado de cargas

En la losa aligerada: para un ancho tributario de 0.40m de acuerdo a la sección típica de esta, presenta las siguientes cargas:

$$\text{CARGA MUERTA (DL)} = 0.40\text{m} \times (420 + 120 + 100) \text{ kg/m}^2$$

$$P = 0.40\text{m} \times 0.15\text{m} \times 1.20\text{m} \times 1350 \text{ kg/m}^3$$

$$DL = 256.00 \text{ kg/m}$$

$$\text{CARGA VIVA (LL)} = 0.40\text{m} \times 250.00 \text{ kg/cm}^2$$

$$LL = 100.00 \text{ kg/m}$$

$$\text{CARGA ÚLTIMA (w}_u\text{)} = 1.4DL + 1.7LL$$

$$W_u = 528.40 \text{ kg/m}$$

$$P_u = 110.00 \text{ kg}$$

En la losa maciza: para una losa de 20.00cm y ancho tributario de 1.00 m

$$DL = 1.00\text{m} \times (120 + 2400.0 \times 0.20) \text{ kg/m}^2$$

$$DL = 600.00 \text{ kg/m}$$

$$LL = 400.00 \text{ kg/m}$$

$$W_u = 1520.0 \text{ kg/m}$$

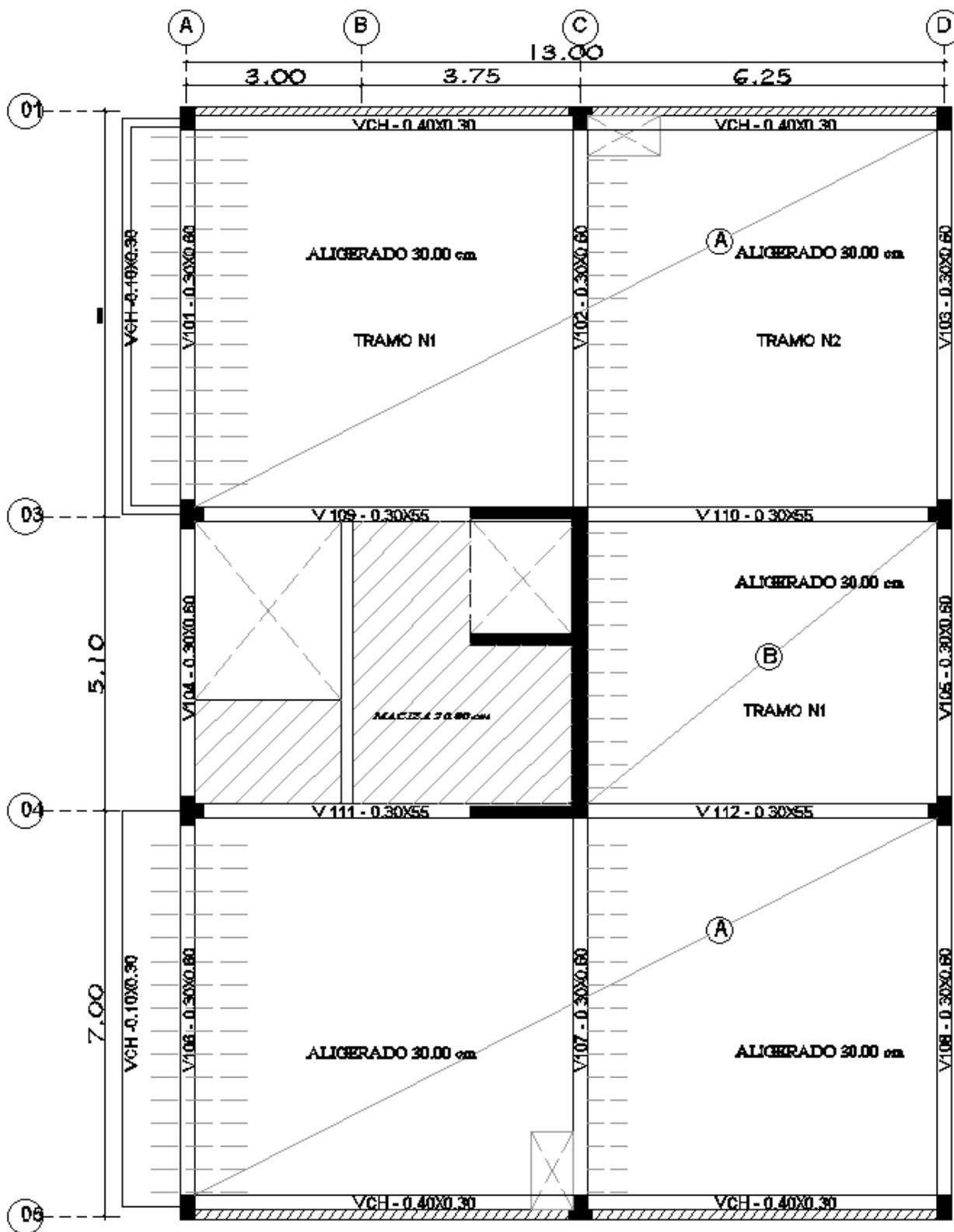


Figura 22: Estructuración de la planta de distribución de oficinas

Fuente: elaboración propia

Diseño de losas de entrapiso

Se realizó de acuerdo la metodología de diseño por resistencia de acuerdo a las limitaciones y especificaciones de la norma NTE E-060 de concreto armado. Se emplearán las fórmulas de las ecuaciones 3.6 a 3.16.

Cálculo de momentos flectores

Los momentos flectores se obtienen de la modelación de los modelos en el programa SAP 2000. Para conseguir los momentos máximos positivos se ha definido una envolvente de las alternancias de carga viva como lo muestra la Figura 19: Alternancia de carga viva de 200 kg/m² en un modelo de losa aligerada unidireccional. Por lo tanto, los momentos para el tramo N1, tramo N2 y el voladizo N0 del modelo ejemplo se muestran la imagen. Se ha considerado un brazo rígido de 0.15 m a cada lado del apoyo para un factor de uno.

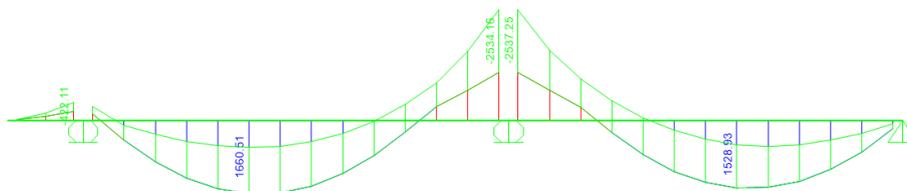


Figura 23: Diagrama de momentos últimos

Fuente: Elaboración propia.

Tabla 12: Momentos últimos por tramos

ALIGERADO		Ln (m)	Mu- i. kg - m	Mu+ kg - m	Mu- j. kg - m
A	Voladizo 0	1.20	0	0	-422.11
	Tramo 1	6.45	-315	1660.5	-2534.16
	Tramo 2	5.95	-2537.25	1528.93	0
B	Tramo 1	5.95	-2453.3	1377.76	0

Fuente: Elaboración propia

Verificación por cortante

Por la ecuación 3.11 se tiene que el cortante que aporta el concreto cuya resistencia específica a compresión de $f'c = 210 \text{ kg/cm}^2$ es:

- Cortante nominal en losas aligeradas

$$\phi V_n = 0.85 \left(1.1 \times 0.53 \sqrt{210} \times 10 \times 27 \right)$$

$$\phi V_n = 1938.9 \text{ kg}$$

- Cortante nominal en losas macizas

$$\phi V_n = 0.85 \left(0.53 \sqrt{210} \times 100 \times 20 \right)$$

$$\phi V_n = 13056.7 \text{ kg}$$

- Cortante último en losas aligeradas

Tabla 13: Cortantes últimas por tramos

ALIGERADO		Ln (m)	Vu- i. kg - m	Vu- j. kg - m
A	Voladizo 0	1.20	0	650.1
	Tramo 1	6.45	-1400	-2048.2
	Tramo 2	5.95	-2027.6	1196.9
B	Tramo 1	5.95	-2013.8	1130.2

Fuente: Elaboración propia

De acuerdo a la tabla de cortantes últimos y cortante resistente por el concreto no se cumple en ciertos tramos que $\phi V_n \geq V_u$. Por lo tanto, para cumplir dicha condición se puede: ensanchar viguetas del aligerado o aumentar la resistencia a compresión del concreto.

$$\text{Caso 1: } b_w = 12 \text{ cm} \quad \rightarrow \phi V_n = 2326.7 \text{ kg}$$

$$\text{Caso 2: } f'_c = 280 \text{ kg/cm}^2 \quad \rightarrow \phi V_n = 2238.8 \text{ kg}$$

En ambos casos el cortante resistente resulta mayor que los cortantes últimos de cada tramo. Sin embargo, al ensanchar viguetas se está aumentando el volumen de concreto y en consecuencia el peso del aligerado. Esto último, conlleva a realizar un nuevo metrado de cargas y recálculo de los nuevos momentos y cortantes últimos, por lo tanto, se optó por la segunda opción.

Cálculo del refuerzo por flexión

Con los momentos exportados del SAP 2000 a una hoja Excel se filtra solamente los momentos últimos que son útiles para calcular el área del refuerzo. Estos momentos se carga al código de Matlab del ANEXO 04: Programa 1 e ingresando las propiedades del material y características de la sección del aligerado se ejecuta el programa. Se guarda el cálculo del área de acero y su respectivo momento en un archivo Excel.

- El As_{min_A} : acero mínimo en los apoyos en cm^2 , por el momento de agrietamiento (num 10-5-1 E-060).
- El As_{min_T} : acero mínimo en los tramos en cm^2

```

Command Window
Ingresar resist compresi3n de concreto en kg/cm2: 280
Ingresar l3mite de fluencia del acero en kg/cm2: 4200
Ingresar ancho del ala de vigueta TEE en cm: 40
Ingresar ancho del alma de vigueta TEE en cm: 10
Ingresar peralte de la vigueta TEE en cm: 30
Asmin_A =
    1.4664
Asmin_T =
    0.7530
Mmax_A =
    4.7578e+03
Mmax_T =
    1.1139e+04
fx >>

```

Figura 24: Ejecuci3n del programa 1 dise1o losas aligeradas

Fuente: Elaboraci3n propia

Tabla 14: 1rea de refuerzo por flexi3n por tramos

ALIGERADO		Ln (m)	As- i. kg - m	As+ kg - m	As- j. kg - m
A	Voladizo 0	1.20	0.00	0.75	0.42
	Tramo 1	6.45	0.31	1.65(1#3)	2.73(1#4)
	Tramo 2	5.95	2.73(1#4)	1.52(1#3)	1.47
B	Tramo 1	5.95	2.63(1#3)	1.37(1#3)	1.47
As min			1#5	1#4	1#5

Fuente: Elaboraci3n propia

Longitud de desarrollo de bastones

El desarrollo de un bast3n de refuerzo debe llegar hasta donde indica la Figura 41: Consideraciones para el corte de acero en regiones de momento positivo y negativo. Para ello, se calcul3 con las f3rmulas de las ecuaciones 3.8 y 3.9 el momento resistente para las barras de acero de #4 y #5.

Tabla 15: Momentos resistentes aligerado 30cm

Di1metro #	Di1metro [pulg]	1rea [cm ²]	M _{r-} [kg-m]	M _{r+} [kg-m]
4	1/2	1.27	1242.37	1282.71
5	5/8	1.98	1907.79	2007.85

Fuente: Elaboraci3n propia

En la siguiente figura se muestra el desarrollo de las barras de refuerzo de una losa aligerada de 2 tramos continuos. El refuerzo positivo m3nimo es 1 barra de 1/2 (#4) y el refuerzo negativo

mínimo es una barra de 5/8 (#5), también podría ser 1 de 1/2 + 1 de 3/8 siempre que sea superior o igual a 1.47 cm².

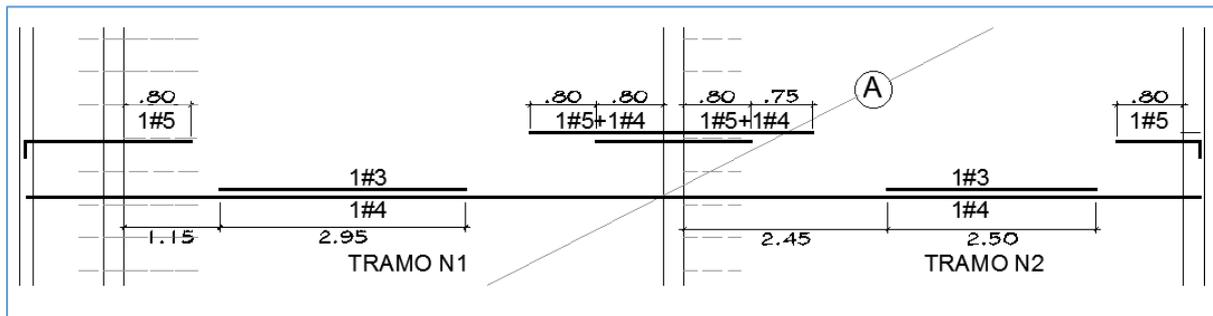


Figura 25: Desarrollo del refuerzo en una losa aligerada de 2 tramos continuos

Fuente: Elaboración propia

Desarrollo de la Red Neuronal

Modelo de red neuronal

El modelo de red neuronal artificial que se utilizará es el del perceptrón multicapa (MLP), ya que resuelve el tipo de problemas de aproximación de funciones con ajuste de entrada-salida y es el más usual. Emplea una arquitectura “feedforwad” con una o más capas ocultas y función de transferencia sigmoidea y una capa de salida con función identidad (ver Figura 26: Arquitectura del modelo de red neuronal. Esta red, emplea los algoritmos de entrenamiento tipo “backpropagation” como: “Bayesian regulation” y “Levenberg Marquardt”. El número de neuronas en la capa oculta se obtendrá del entrenamiento de varias redes y se utilizará la que tiene mejor eficiencia.

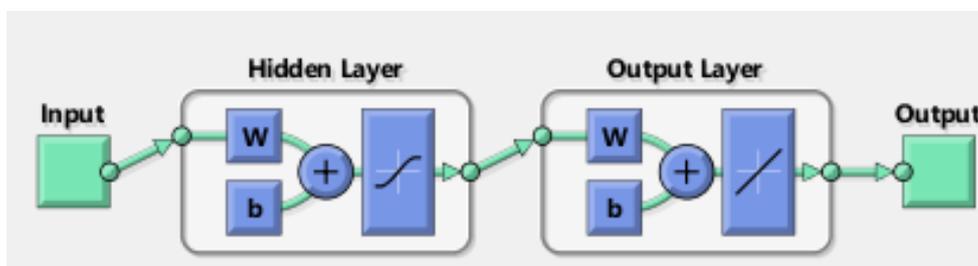


Figura 26: Arquitectura del modelo de red neuronal

Fuente: Mathworks

Parámetros de entrada y salida

El diseño estructural de losas aligeradas de entrepiso mediante redes neuronales consiste en la estimación del refuerzo positivo por flexión, su ubicación y longitud de desarrollo en un tramo analizado cumpliendo con los límites de resistencia y servicio.

▪ Entradas

Las redes neuronales con entrenamiento supervisado son más eficientes para el ajuste de funciones, es decir, se tiene que emplear una matriz de datos para entrenamiento de la red. En consecuencia, para poder obtener una data valores de entrada se fijaron las siguientes variables.

▪ Variables constantes

- Se diseñará un aligerado de 30.00 cm
- El peso propio de la estructura es de 420 kg/m^2
- El peso de acabados y tabiquería es de 220 kg/m^2
- La sobrecarga es de 250 kg/m^2
- Resistencia a compresión del concreto de 280 kg/cm^2
- El esfuerzo a fluencia del acero es de 4200 kg/cm^2

La red neuronal entrenada solo trabajará dentro del rango de estas variables constantes, de no cumplirse estos parámetros se corre el riesgo de obtener resultados incorrectos.

▪ Vector de entrada de la RNA

Los datos de la Tabla 16: Variables del vector entrada a la red neuronal son los valores de entrada para la red neuronal. Estos datos se pueden presentar en un vector de tal forma que cada vector representa el diseño de una losa aligerada.

Tabla 16: Variables del vector entrada a la red neuronal

Inputs para el diseño en un tramo				
N°	Variable	Descripción	Und.	Rango
1	L	Longitud edificación	m	De 12 a 28.50
2	T	Total, tramos continuos		De 2 a 4
3	n	N° tramo a diseñar		
4	Ln	Luz libre del tramo a diseñar	m	De 6.00 a 7.50

Inputs para el diseño en un apoyo				
N°	Variable	Descripción	Und.	Rango
1	L	Longitud losa continua	m	De 12 a 28.50
2	T	Total, apoyos		De 3 a 5
3	n	N° apoyo a diseñar		
4	Lni	Luz libre anterior al apoyo	m	De 6.00 a 7.50
5	Lnj	Luz libre posterior al apoyo	m	

Fuente: Elaboración propia

▪ Salidas

Los valores que se desea obtener mediante la red neuronal son el acero de refuerzo, la ubicación y longitud del bastón del paño interior o exterior de una losa aligerada (ver Figura 27). Con estos datos se podrá optimizar los diseños de losas aligeradas cumpliendo las limitaciones de resistencia y servicio. La siguiente tabla muestra las variables del vector de salida de la red neuronal.

Tabla 17: Variables del vector de salida de la red neuronal

Outputs del diseño en un tramo			
Nº	Variable	Descripción	Und
1	db	Diámetro del refuerzo corrido +	#
2	db _b	Diámetro de bastón	#
3	x _b	Distancia del apoyo al bastón	m
4	L _b	Longitud de desarrollo del bastón	m

Outputs de diseño en un apoyo			
Nº	Variable	Descripción	Und
1	db	Diámetro del refuerzo mínimo -	#
2	db _b	Diámetro de bastón	#
3	L _{d_i}	Longitud de desarrollo de db a la izquierda del apoyo	m
4	L _{dbi}	Longitud de desarrollo de db _b a la izquierda del apoyo	
	L _{dd}	Ld de db a la derecha del apoyo	
4	L _{db_s}	Ld de db _b a la derecha del apoyo	m

Fuente: Elaboración propia

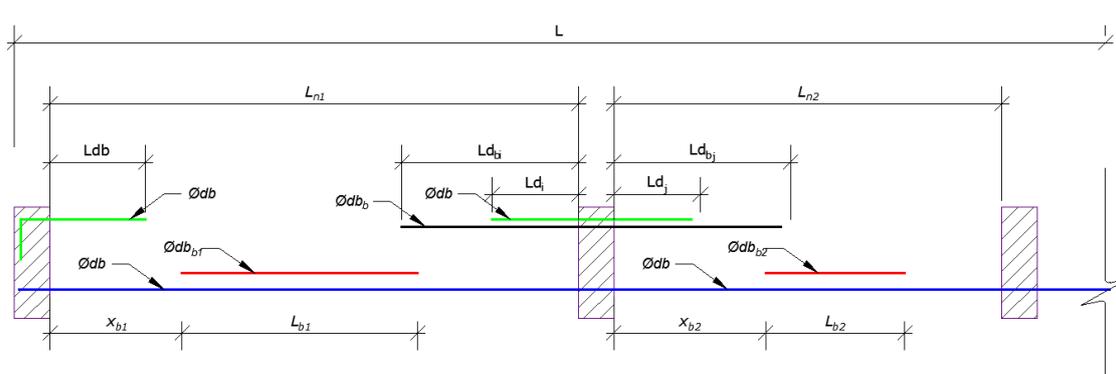


Figura 27: Representación de las variables de entrada y salida de la red neuronal

Fuente: Elaboración propia

Entrenamiento de la red neuronal

Se realizó el entrenamiento de dos redes neuronales, ya que la cantidad de variables de salida es mayor que las variables de entrada y la red pierde capacidad de aprendizaje. Por lo tanto, la primera red a entrenar es para el diseño en los tramos de una losa aligerada, siendo la otra red para el diseño en apoyo. Los siguientes datos de entrada y salida se utilizaron para el entrenamiento de la red neuronal artificial en Matlab.

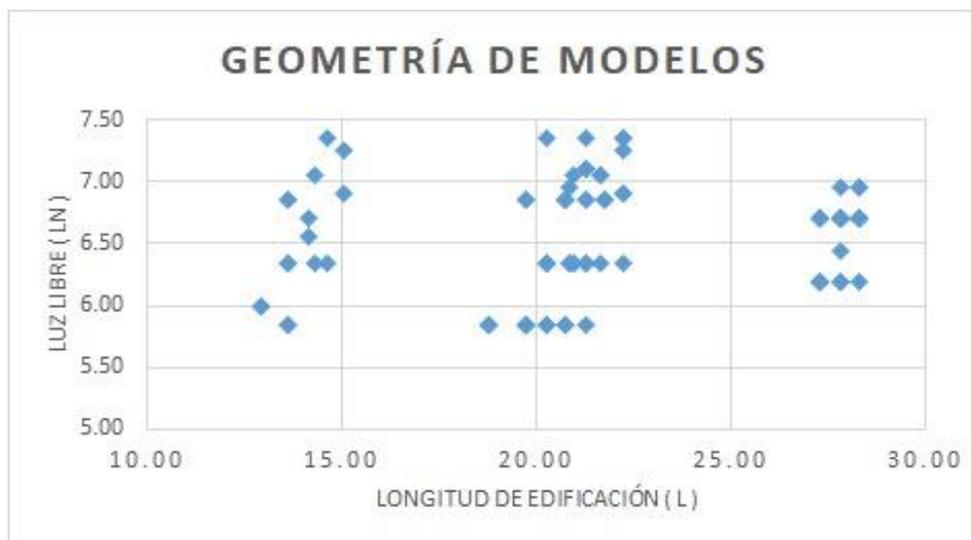


Figura 28: Datos de entrada para entrenamiento red neuronal

Fuente: elaboración propia

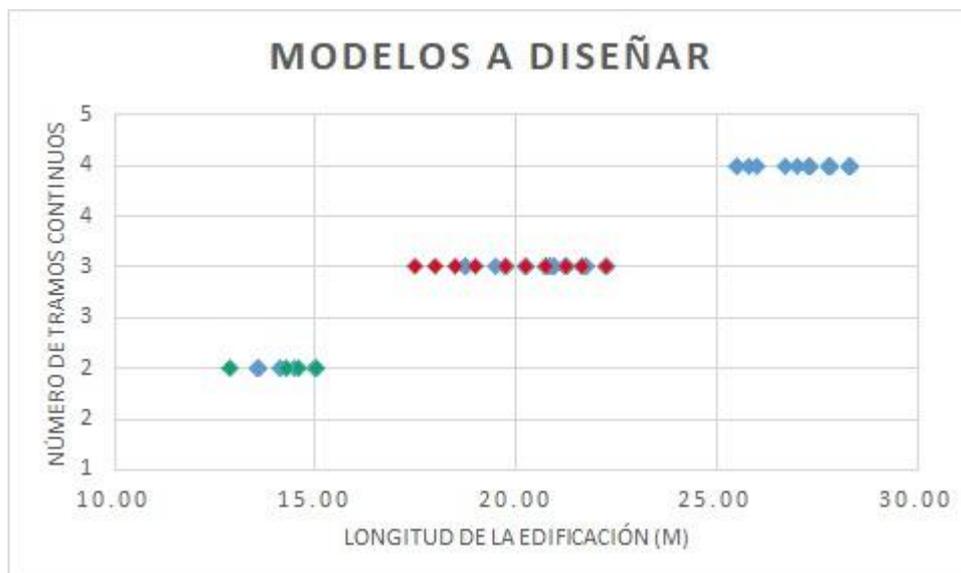


Figura 29: Datos de entrada para entrenamiento red neuronal – Losas a diseñar

Fuente: elaboración propia

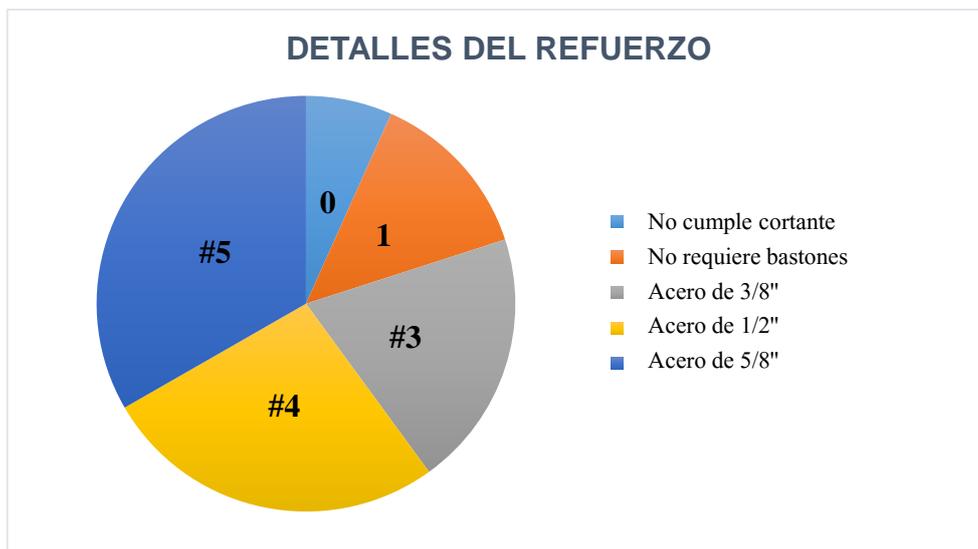


Figura 30: Datos de salida para entrenamiento de red neuronal – Refuerzo

Fuente: elaboración propia

▪ Número de neuronas de la capa oculta

El número óptimo de neuronas en la capa oculta se determinó realizando entrenamientos a la red neuronal en un rango de neuronas de 5 a 50. Se hicieron entrenamientos elevando el número de neuronas de 5 en 5 hasta conseguir una red neuronal eficiente. El mejor rendimiento se obtuvo en el rango de 5 a 10 neuronas por lo que se profundizó el entrenamiento en ese espacio encontrándose 8 neuronas como número óptimo. El código de la arquitectura de la red neuronal que se usó se muestra en el Anexo 03.

Tabla 18: Número óptimo de neuronas de la capa oculta.

Neuronas	Performance	Gradiente
5	0.299	7.74E-05
10	0.0935	7.83E-04
15	0.024	7.53E-03
20	0.0122	1.44E-02
25	0.00489	0.00139
30	0.00749	0.00111
35	0.000769	0.00075
40	0.0068	0.00259
45	0.00687	0.00393
50	0.00688	0.00076

Neuronas	Performance	Gradiente
5	0.325	0.00251
5	0.224	6.38E-04
6	0.056	0.00254
6	0.087	1.25E-04
8	0.0065	0.00078
8	0.15462	0.05748
8	0.034884	8.62E-05
10	0.0536	3.82E-04
10	0.02456	1.68E-03
10	0.2456	3.57E-05

Fuente: Elaboración propia

Como se puede apreciar en la tabla anterior, la red neuronal tuvo mayor eficiencia con 8 neuronas en la capa oculta. Por lo tanto, la arquitectura final de la red neuronal elegida para el diseño del refuerzo positivo de losas aligeradas tiene: 04 neuronas en la capa de entrada, 8 neuronas en la capa oculta y 4 neuronas en capa de salida. Las funciones de transferencia que utiliza la red neurona son: en la capa oculta *tansig* y en la capa de salida la función *lineal*. La Figura 31, muestra la arquitectura de la red neuronal elegida para diseño del refuerzo positivo (tramos) de losas

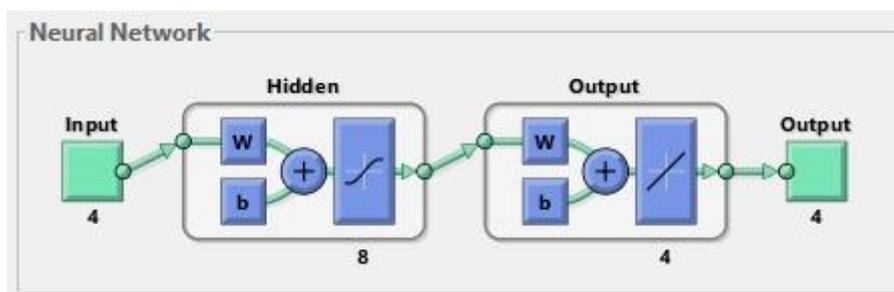


Figura 31: Arquitectura de la red neuronal para diseño refuerzo positivo en losas.

Fuente: Elaboración propia

Resultados del entrenamiento

En el estado de entrenamiento de la red se detuvo en la época 433 con un gradiente del error de $8.6197e-05$, aunque no alcanzó el gradiente de error asignado $1e-29$. Sin embargo, se validó a lo largo de 150 iteraciones más donde el gradiente de error siguió ascendiendo y se detuvo el entrenamiento (ver figura 32). Por otro lado, el menor error medio cuadrático de validación se obtuvo en la época 283 cuyo valor es 0.034884 (ver figura 33).

La red neuronal 2, de diseño en un apoyo, se consiguió mejor aprendizaje de la red con 5 neuronas en la capa oculta. El proceso de entrenamiento se detuvo en la época 259 validándose con 153 donde el gradiente continuó subiendo. La arquitectura final de la red 2 se muestra en el anexo de figuras.

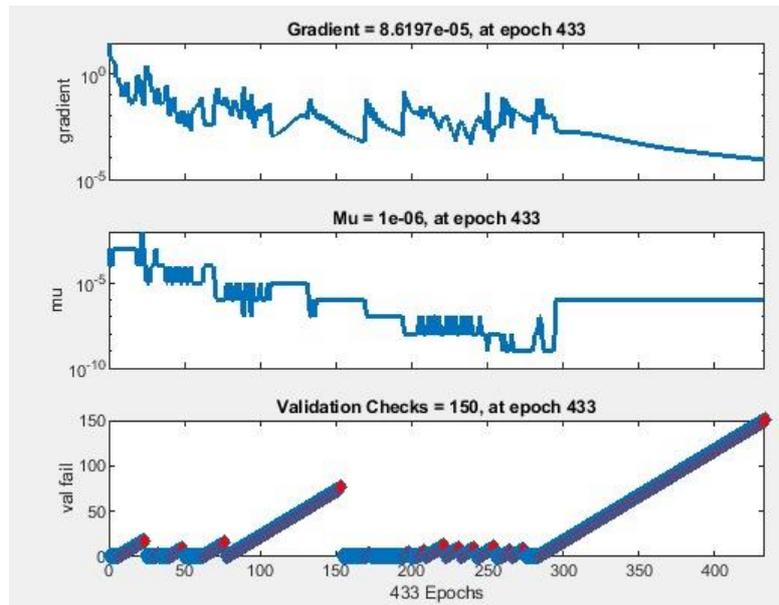


Figura 32: Estado del entrenamiento de la red neuronal

Fuente: Elaboración propia

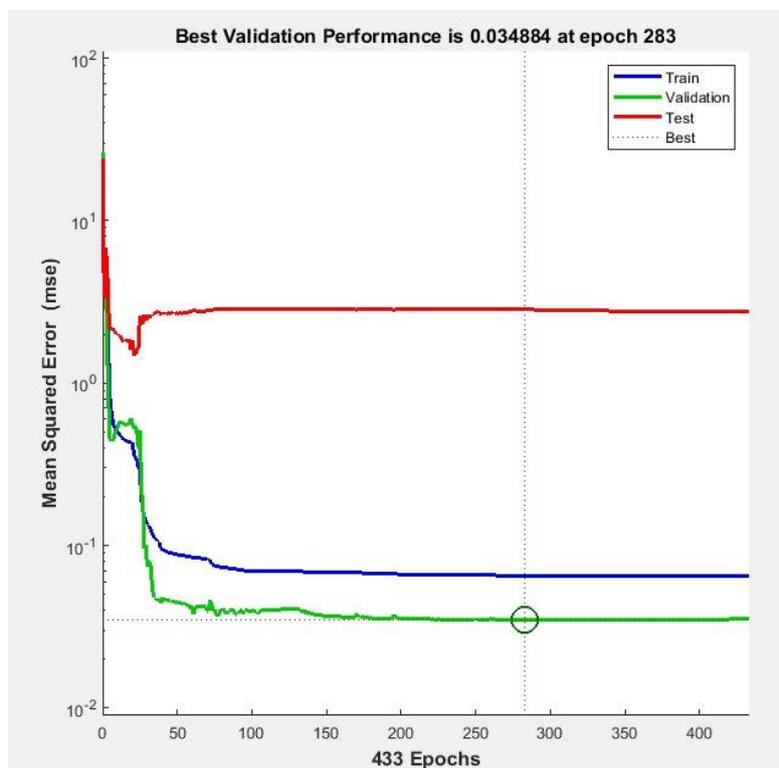


Figura 33: Mejor valor del error cuadrático red neuronal

Fuente: Elaboración propia

Las siguientes tablas 19 y 20 muestran las matrices de los umbrales y pesos sinápticos que se obtuvieron del entrenamiento de la red neuronal 1 para el diseño del refuerzo positivo.

Tabla 19: Matriz de umbrales y pesos sinápticos de las neuronas capa oculta

FILA/ COL	1	FILA/ COL	1	2	3	4
1	-14.8531	1	-24.5325	-82.0815	132.1805	-41.0358
2	5.5161	2	0.0620	-0.0664	5.4865	-0.3311
3	-39.9234	3	40.3729	-61.6162	-3.6082	72.9533
4	-16.9253	4	-3.7433	13.6823	0.2194	21.3468
5	0.8370	5	0.4937	4.1321	-7.0228	0.0396
6	-43.6119	6	-32.0979	-2.9967	-29.8674	-15.3861
7	-0.7634	7	-0.7067	-4.0385	7.2294	0.0052
8	-32.0594	8	-3.7305	26.4762	0.2110	38.8041

Fuente: Elaboración propia

Tabla 20: Matriz de umbrales y pesos sinápticos de la capa de salida

FILA/ COL	1	FILA/ COL	1	2	3	4	5	6	7	8
1	1.2161	1	0.9212	0.0079	-0.9868	564.7159	-1.8886	-0.0135	-2.8255	- 563.4780
2	1.3636	2	1.3682	-1.3052	-1.0376	633.8651	64.8052	0.0521	63.3175	- 632.2044
3	0.1480	3	0.4747	-0.2479	-0.2366	183.7533	-14.4540	0.0841	-14.4626	- 183.1629
4	1.1333	4	1.2334	-1.6785	-0.9395	605.1395	73.0517	-0.0113	71.7825	- 603.6935

Fuente: Elaboración propia

Algoritmo Grey Wolf Optimization - GWO

Parámetros de ingreso

Son las variables que se muestran en la Tabla 21 Magnitudes comunes de los parámetros usados para la simulación, para la optimización de la sección y diseño de losas macizas utilizando el algoritmo GWO.

Tabla 21 Magnitudes comunes de los parámetros usados para la simulación

Nombre	Descripción	Magnitudes	Unidades
ln	Longitud entre apoyos	3.5	m
DL _s ^[1]	Carga muerta, excluyendo peso propio	100	kg/m ²
LL	Carga viva	250.00	kg/m ²
f _c	Resistencia específica a compresión del concreto	280.00	kg/cm ²
f _y	Resistencia a fluencia del refuerzo	4200.00	kg/cm ²
rec	Recubrimiento	2.00	cm
db	Diámetro del refuerzo mínimo	9.5	mm
wg	Peso concreto por unidad de volumen	2400.00	kg/m ³
ws	Peso del acero por unidad de volumen	7850.00	kg/m ³
Cuc ^[2]	Precio del concreto por unidad de volumen	420.00	S./m ³
Cur	Precio de las barras de refuerzo por unidad de peso	4.47	S./kg
Cc	Costo vaciado concreto	42.7	S/m ³
Cr	Costo habilitación de acero	4.40	S/kg

[1]. El peso propio se computa en variables de optimización.

[2]. El precio para concreto varías según su resistencia a compresión.

[3]. Referencia de costos: <https://www.premezcla2.com/precios.htm>

Fuente: Elaboración propia

Tabla 22 Coeficientes del máximo momento, cortante y deflexión usado para el diseño

Caso Apoyo / Coeficiente (k _i)	Simples	Simple-Continuo	Continuos	Voladizo
Momento	1/8	1/10	1/11	1/2
Cortante	1	1.15	1	2
Deflexión ^[1]	1/16	1/18.5	1/21	1/8
Deflexión ^[2]	1/20	1/24	1/28	1/10

[1] Coeficientes para el peralte mínimo de losas nervadas en una dirección.

[2] Coeficientes para el peralte mínimo de losas macizas en una dirección.

Fuente: elaboración propia

Función objetivo

La función objetivo para la optimización de losas nervadas es el costo total asociado al costo del concreto con el refuerzo y construcción los cuales deben minimizarse. Esta función se escribe de la siguiente forma para determinar el valor óptimo por decisión de las variables independientes consideradas (ver Figura 34: Sección de una losa maciza).

$$c_t = c_c + c_r + c_{con}$$

$$c_t = v_c (c_{uc} + c_c) + w_r (c_{ur} + c_r)$$

Considerando: $\bar{c} = \frac{c_t}{c_{uc} + c_c}$

$$\text{Mínimo } f(x_n) = \bar{c} = \left(v_c + \frac{w_r (c_{ur} + c_r)}{c_{uc} + c_c} \right) / b \quad (1)$$

Variables de diseño

Volumen concreto $C_c = bx_1 l_n C_{uc}$ (2)

Peso del refuerzo $w_r = w_s A_s l_s = w_s \frac{\pi}{4} \left(\frac{x_3^2}{1000^2} l_d \right) \left(\frac{b}{x_2} \right)$ (3)

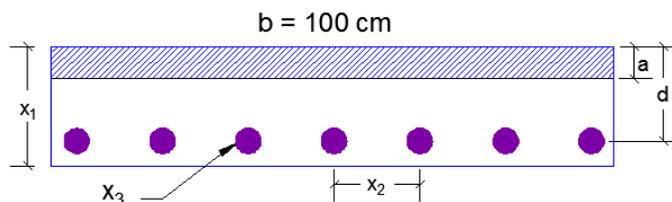


Figura 34: Sección de una losa maciza

Fuente: Elaboración propia

Restricciones

▪ Restricción de flexión

Se debe cumplir ecuación 3.3 entonces en función de las variables de x_1 a x_3 se escribe de la siguiente manera:

$$g_1(x_n) = \frac{M_u}{\phi M_n} - 1 \leq 0 \quad (4)$$

$$M_u = k_\alpha w_u l_n^2 \quad (5)$$

$$w_u = 1.4 \{w_g x_1 b + DL_s b\} + 1.7 LLb \quad (6)$$

$$\phi M_n = \phi A_s f_y \left\{ (x_1 - rec - 1) - \frac{1}{2} \left(\frac{A_s f_y}{0.85 f_c' b} \right) \right\} \quad (7)$$

$$A_s = \frac{\pi}{4} \left(\frac{x_3^2}{100} \right) \left(\frac{b}{x_2} \right) \quad (8)$$

Donde k_m es el coeficiente de momento según Tabla 22 y $\phi = 0.9$.

▪ Restricción de cortante

Se debe cumplir ecuación 3.10, entonces en función de las variables de x_1 a x_3 se escribe de la siguiente manera:

$$g_2(x_n) = \frac{V_u}{\phi V_n} - 1 \leq 0 \quad (9)$$

$$V_u = k_v w_u \frac{l_n}{2} \quad (10)$$

$$\phi V_n = \phi V_c = \phi 0.53 \sqrt{f_c'} (x_1 - rec - 1) b \quad (11)$$

Donde k_v es el coeficiente de cortante según Tabla 22 y $\phi = 0.85$

▪ Restricciones de servicio

Se debe cumplir ecuación 3.12, entonces en función de las variables de x_1 a x_3 se escribe de la siguiente manera:

$$g_3(x_n) = \frac{\rho_{\min}}{\rho} - 1 \leq 0 \quad (12)$$

$$g_4(x_n) = \frac{\rho}{\rho_{\max}} - 1 \leq 0 \quad (13)$$

$$g_5(x_n) = \frac{s}{s_{\max}} - 1 \leq 0 \quad (14)$$

$$g_6(x_n) = \frac{s_{\min}}{s} - 1 \leq 0 \quad (15)$$

$$\rho = \frac{A_s}{bx_1} \quad (16)$$

$$A_{sd_b} = \frac{\pi}{4} \left(\frac{d_b}{10} \right)^2 \left(\frac{b}{x_2} \right) \quad (17)$$

Donde $\rho_{\min}=0.0018$ y ρ_{\max} se obtiene según la ecuación 3.16. $s_{\max} = \min\{3h, 40.0cm\}$,
 $s_{\min} = \max\{db, 2.54cm\}$

El espaciamiento máximo del refuerzo no debe exceder tres veces el espesor de la losa ni 40 cm.

▪ Restricciones de deflexión

Se debe cumplir que el espesor de la losa debe ser mayor que: la luz libre por un factor k_d según la Tabla 22.

$$g_7(x_n) = \frac{k_d l_n}{x_1} - 1 \leq 0 \quad (18)$$

▪ Límites de las variables

$$\begin{aligned} 5.00 &\leq x_1 \leq 65.00 \text{ cm} \\ 2.54 &\leq x_2 \leq 40.00 \text{ cm} \\ 9.50 &\leq x_3 \leq 19.10 \text{ mm} \end{aligned} \quad (19)$$

Penalización de la función objetivo

Al tratarse de un problema de optimización con restricciones no lineales, el espacio de búsqueda de las variables independientes se divide en dos espacios. Uno donde las restricciones son factibles y el otro en el cual las restricciones son inviables, es decir $g_j(x_n) > 0$. Por lo tanto, para este tipo problemas, existen tres técnicas de manejo de restricciones en computación evolutiva. Una de estas técnicas es la penalización de la función objetivo y el método más común para penalizar es el de Lagrange modificado [44], [52].

$$F(X) = f(x) + \sum_{j=1}^{l+m} \lambda_j g_j(X) \quad (20)$$

$$\begin{aligned} X &= \{x_1, x_2, \dots, x_5\} \\ g_j(x) &\leq 0; & j &= 1, 2, \dots, l \\ h_k(x) &= 0; & k &= 1, 2, \dots, m \end{aligned}$$

Donde λ es el parámetro de penalización y tiene valores muy altos como: 10^7 , 10^{10} , otros.

El código se muestra en el Anexo 03

Verificación de la penalización

Sean los valores de variables constantes de la Tabla 21 Magnitudes comunes de los parámetros usados para la simulación, considerando bordes continuos de la losa maciza; las mejores soluciones obtenidas por el algoritmo GWO para la de la función objetivo sin penalizar y penalizada son los que se muestran en la siguiente tabla. Las soluciones logradas deberán convalidarse con las restricciones del problema.

Como se puede observar, en la función objetivo sin penalizar sus soluciones caen en los límites de cada variable y se violan las restricciones de flexión, servicio y deflexión. Por otro lado, la función penalizada obtiene mejores resultados y no cumple solamente la restricción de servicio (cuantía mínima y máxima).

Tabla 23: Comparación de soluciones

Variables	Función Objetivo sin Penalizar	Función Objetivo Penalizada
x_1	5	26.7956
x_2	40	38.4903
x_2	9.500	10.2725
$F(x)$	91.7689	2.4800e+07
Mejor Costo	91.7689	447.0601

Fuente: elaboración propia

Tabla 24: Verificación de restricciones

Restricciones	Función Objetivo sin Penalizar (1)	Función Objetivo Penalizada (20)
De flexión (4)	7.2925	0.9992
De cortante (9)	-0.0454	-0.8318
De servicio (12)	-0.4921	1.2400
(13)	8.8794	1.2400
(14)	1.6667	-0.0377
(15)	-0.3650	-0.3401
De deflexión (18)	1.5000	-0.5335

Fuente: elaboración propia

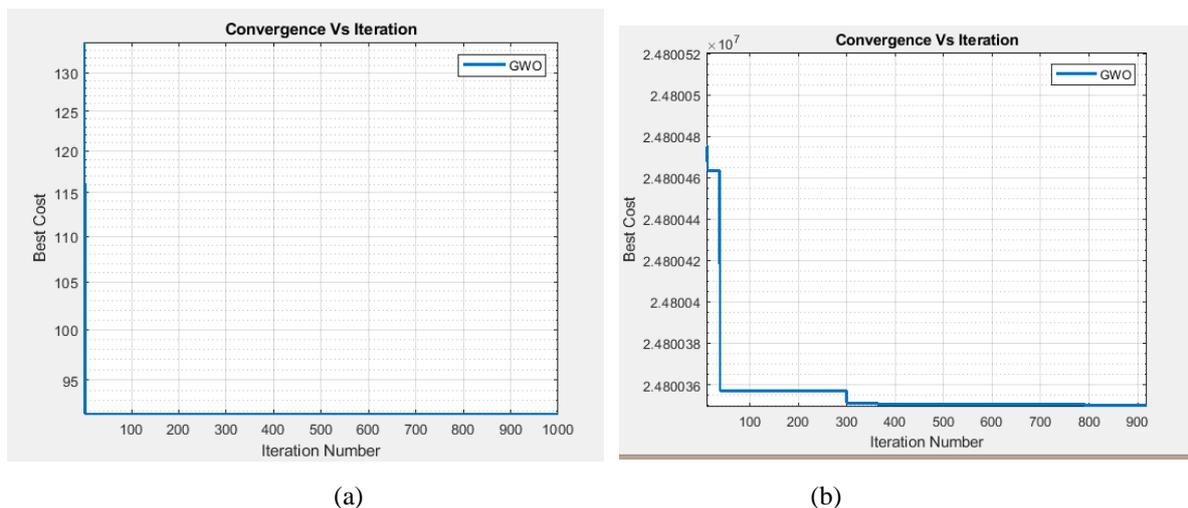


Figura 35: Convergencia de la función objetivo (a) sin penalizar (b) penalizada -GWO

Fuente: elaboración propia

Validación de la red neuronal y algoritmo GWO

A continuación, se realiza una prueba de la red neuronal artificial entrenada y el algoritmo GWO. Primero, se realiza el diseño de modelos de losas aligeradas y macizas de manera tradicional donde se utiliza herramientas como el Sap 2000 y hojas de Excel, y cabe resaltar que estos modelos no han sido incluidos en el entrenamiento de la red. Posteriormente, se realiza el diseño estructural utilizando la red neuronal artificial.

Modelos a diseñar

A continuación, se realiza 4 diseños, dos de los modelos de entrenamiento de la red neuronal (MOD1 y MOD3) y los otros fuera de la data de modelos de entrenamiento. Los datos de entrada a la red son los que se muestra en la siguiente tabla.

Tabla 25: Datos de entrada a la red entrenada

Modelos diseñados por la red entrenada					
MODELO	Longitud Edificación (m)	Nro Tramos	Nro Apoyos	# Paño a diseñar	Luz libre del tramo a diseñar (m)
MOD1	22.25	3	4	1, 2 y 3	6.35, 7.35 y 7.35
MOD2	21.20	3	4	1, 2 y 3	6.5, 6.75 y 6.75
MOD3	14.30	2	4	1 y 2	7.05 y 6.35
MOD4	13.40	2	4	1 y 2	6 y 6.5

Fuente: Elaboración propia

Operación de la red neuronal

La red neuronal 1 y 2 operan de manera similar, se requiere de las variables de entrada definidas en la Tabla 16: Variables del vector entrada a la red neuronal . Con en el ingreso de los datos anteriores se elige el vector de entrada según el número de tramos y apoyos que tenga el modelo y se ejecuta el programa. En la figura de los anexos se muestra el script de Matlab para el diseño del modelo 3.

El diseño del refuerzo positivo obtenido por la red neuronal 1 se muestra en la tabla 26. El acero mínimo, los bastones en cada tramo, la distancia de la cara del apoyo al inicio de la longitud de desarrollo del bastón. Un resultado equivalente a: [0 0 0 0] significa que para la luz libre de ese tramo no pasaría por cortante y [4 1 0 0] significa que para ese tramo no requiere bastones. Otros resultados como negativos, diámetros de refuerzo (alejados de #3 y #4) es obtenido naturalmente por la red entrenada.

El diseño del refuerzo negativo por la red neuronal 2 se muestra en la siguiente tabla 27. El acero mínimo, los bastones en cada apoyo, las longitudes de desarrollo a la izquierda y derecha del apoyo. Un resultado obtenido como: [5 1 0 0 0.80 0] significa que no requiere bastones y longitud de desarrollo a la izquierda, únicamente una Ld de 0.80m a la derecha del apoyo.

Para valores del diámetro obtenidos por las redes neuronales lo aproximaremos al valor entero más cercano para comparar el error con el diseño tradicional en las tablas 28 y 29.

Tabla 26: Resultados del diseño en tramos por la red neuronal 1

Valores del diseño por la red neuronal 1					
MODELO	Tramo	Diámetro (#) As min (db)	Diámetro bastón (dbb)	Posición bastón (xi)	Longitud de desarrollo del bastón (Ldb)
MOD1	1	4	3	0.67	3.75
	2	4	1	-0.02	-0.01
	3	0	0	0.05	-0.03
MOD2	1	3.99	3.41	0.66	3.78
	2	3.99	0.99	-0.01	-0.02
	3	4.01	3.79	1.75	4.24
MOD3	1	0	0	0.00	0.00
	2	4	3	4.40	2.04
MOD4	1	3.94	3.00	1.02	2.76
	2	3.99	3.56	2.37	3.59

Fuente: Elaboración propia

Tabla 27: Resultados del diseño en apoyos por la red neuronal 2

Valores del diseño por la red neuronal 2							
MODELO	Tramo	Diámetro (#) As min (db)	Diámetro bastón	L _{di}	L _{d_{bi}}	L _{da}	L _{d_{bd}}
MOD1	1	5	1	0.00	0.00	0.80	0.00
	2	5	3	0.80	1.35	0.80	1.65
	3	5	4	0.95	1.85	0.00	0.00
	4	5	1	0.80	0.00	0.00	0.00
MOD2	1	5.00	1.02	-0.12	-0.19	0.80	0.22
	2	5.01	3.33	0.50	0.73	0.69	1.49
	3	5.00	3.27	0.98	1.21	0.79	0.87
	4	4.97	-2.16	0.28	-1.13	0.00	-1.47
MOD3	1	5	1	0.00	0.00	0.80	0.00
	2	5	5	0.00	0.00	0.90	1.70
	3	5	1	0.80	0.00	0.00	0.00
MOD4	1	5.00	0.78	-0.11	-0.29	0.74	0.12
	2	4.99	0.01	0.14	-0.34	0.58	-0.48
	3	4.97	-2.18	0.28	-1.13	-0.01	-1.48

Fuente: Elaboración propia

A continuación, se compara los diseños obtenidos por las redes neuronales con el diseño tradicional que se muestra en la figura del desarrollo del refuerzo para los modelos MOD2 y MOD4, que fueron modelos que no están la data de entrenamiento de la red neuronal.

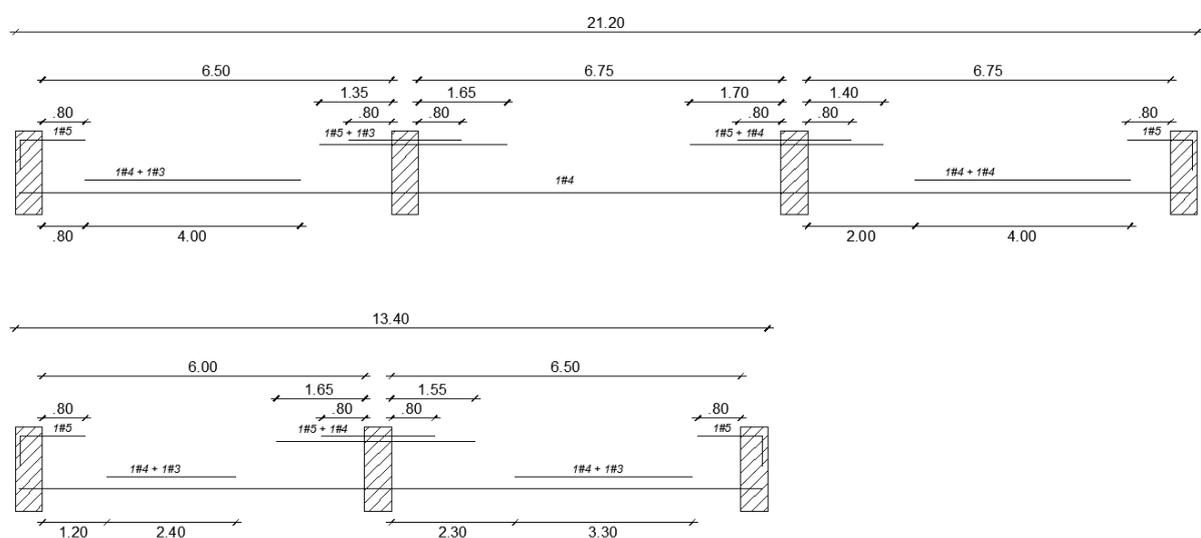


Figura 36: Desarrollo del refuerzo positivo y negativo MOD2 y MOD4

Fuente: Elaboración propia

Tabla 28: Comparación de resultados de diseño tradicional vs proceso la red neuronal 1

DISEÑO:		Red entrenada				Tradicional				%Max Error
MODELO	Tramo	db	dbb	xi	Ldbb	db	dbb	xi	Ldbb	
MOD1	1	4	3	0.67	3.75	4	3	0.70	3.75	3.00%
	2	4	1	-0.02	-0.01	4	1	0.00	0.00	2.00%
	3	0	0	0.05	-0.03	0	0	0.00	0.00	5.00%
MOD2	1	4	3	0.66	3.78	4	3	0.80	4.00	22.00%
	2	4	1	0	0	4	1	0	0	0.00%
	3	4	4	1.75	4.24	4	4	2.00	4.00	25.00%
MOD3	1	0	0	0.00	0.00	0	0	0.00	0.00	0.00%
	2	4	3	4.40	2.04	4	3	4.40	2.04	0.00%
MOD4	1	4	3	1.02	2.76	4	3	1.20	2.40	36.00%
	2	4	4	2.37	3.59	4	3	2.30	3.30	29.00%

Fuente: Elaboración propia

Tabla 29: Comparación de resultados de diseño tradicional vs proceso la red neuronal 2

DISEÑO:		Red entrenada				Tradicional				%Error
MODELO	Tramo	db	dbb	Ldi	Ldbi	db	dbb	Ldi	Ldbi	
MOD1	1	5	1	0.00	0.00	5	1	0.00	0.00	0.00%
	2	5	3	0.80	1.37	5	3	0.80	1.35	2.00%
	3	5	4	0.92	1.78	5	4	0.95	1.85	3.00%
	4	5	1	0.85	0.17	5	1	0.80	0.00	5.00%
MOD2	1	5	1	-0.12	-0.19	5	1	0.80	0	0.00%
	2	5	3	0.50	0.73	5	3	0.80	1.35	0.00%
	3	5	3	0.98	1.21	5	3	0.8	1.70	49.00%
	4	5	0	0.28	-1.13	5	1	0.8	0	0.00%
MOD3	1	5	1	0.00	0.00	5	1	0.00	0.00	0.00%
	2	5	5	0.00	0.00	5	5	0.00	0.00	0.00%
	3	5	1	0.80	0.00	5	1	0.80	0.00	0.00%
MOD4	1	5	1	-0.11	-0.29	5	1	0.8	0.0	0.00%
	2	5	0	0.14	-0.34	5	4	0.80	1.65	0.00%
	3	5	0	0.28	-1.13	5	1	0.80	0.0	0.00%

Fuente: Elaboración propia

Se observa que el diseño por redes neuronales artificiales de nuevos modelos de losas que no estuvieron incluidas en el entrenamiento presenta algunos resultados incongruentes, como longitudes de desarrollo negativas sobre todo en el diseño del refuerzo negativo, ya que la red neuronal 1 si muestra una eficiencia debajo del 50%.

Los diámetros del refuerzo positivo obtenidos por la red neuronal 1 son los correctos en comparación al diseño tradicional de losas, tanto para los modelos que se usó en el entrenamiento (MOD 1 y MOD3) como para los otros modelos distintos al entrenamiento de las redes neuronales. Asimismo, la posición y longitudes de desarrollo de los bastones positivos varía en 25% en el modelo 2, es decir el resultado por la red neuronal es 25cm menos de longitud de desarrollo con respecto al diseño tradicional.

Por otro lado, para los modelos que fue entrenado la red neuronal muestra resultados con errores menores al 10%. Es posible mejorar los resultados para los nuevos modelos entrenando redes neuronales con una data mucho mayor para que la red neuronal artificial aprenda a estimar nuevos modelos con menos variación respecto al diseño tradicional.

V. Conclusiones

Se realizó el diseño de 50 modelos de losas aligeradas no preesforzadas con luces libres que van de los 6.00 a 7.50 m en edificaciones para uso de oficinas y/o aulas con una sobrecarga de 250kg/m^2 . Asimismo, se consideró para carga de acabados 120kg/m^2 y una carga de tabiquería 100kg/m^2 .

En el proceso del diseño estructural se encontró que, por las cargas consideradas, el cortante último superó al resistente aportado por el concreto de $f'c = 210\text{kg/cm}^2$ en luces menores a los 6.25m. Se optó por incrementar la resistencia a compresión a los 280kg/cm^2 para no variar la sección típica de los aligerados ensanchando el alma de la vigueta.

Las redes neuronales están entrenadas con los 50 modelos generados, donde se identificó 06 parámetros de entrada (longitud edificación, número de tramos de continuos, número de apoyos, luz libre de un tramo y número del tramo y apoyo a diseñar) 04 y 06 parámetros de salida para diseño de un tramo y apoyo, respectivamente.

Se entrenaron dos redes neuronales, la primera para el diseño del refuerzo positivo y la otra para el diseño del refuerzo negativo. En el proceso de entrenamiento de la red neuronal 1 se encontró que el número óptimo de neuronas en la capa oculta es de 8, mientras para la red neuronal 2 es de 5 neuronas.

Se definió como función objetivo para optimización de losas macizas por el grey wolf optimization (gwo) el costo que depende de las variables de el espesor de la losa, el espaciamiento del refuerzo y el diámetro del mismo.

El algoritmo gwo cae tempranamente sus iteraciones en los límites de las variables si no se penaliza la función objetivo y no se cumple más del 50% de sus restricciones. Por otro lado, cuando se penalizó la función las soluciones mejoraron y solo hubo una variable (el diámetro del refuerzo) que cayó en su límite inferior, por esta razón no se cumplió las restricciones de servicio.

Comparando el diseño por redes neuronales y el diseño tradicional se observó que algunos de los resultados obtenidos presentan incongruencias como longitud de desarrollo negativo, esto se vio en el diseño del refuerzo negativo. Para el diseño del refuerzo positivo si mostró mejores valores la red, estimando los diámetros de al acero mínimo y bastones correctamente, en las longitudes y posiciones de estos bastones se tuvo una diferencia menor al 50% con respecto al diseño tradicional.

Las redes neuronales si mostraron buenos resultados en el diseño de los modelos que sirvieron de data de entrenamiento de las mismas, presentando errores menores del 10%.

VI. Recomendaciones

Se sugiere realizar el estudio de losas de entrepiso de luces menores y mayores a los modelos de esta investigación con el fin de ampliar la data de entrenamiento para el mejor aprendizaje de la red neuronal.

Se sugiere incrementar la cantidad de modelos de entrenamiento y que estos presenten irregularidades en su configuración; asimismo, diseñar modelos de mayor cantidad de tramos.

Es necesario que se revise por un profesional los resultados de diseño que se obtienen por la red neuronal entrenada, ya que estas nos brindan un diseño preliminar.

Se recomienda incluir en el estudio de losas el análisis de vigas chatas.

Se recomienda realizar el estudio de losas de entrepiso preesforzadas, puesto que este tipo de losas pueden cubrir mayores luces que los aligerados y macizas no preesforzadas.

VII. Referencias bibliográficas

- [1] M. Afzal, “Evaluation and Development of Automated Detailing Design Optimization Framework for RC Slabs Using BIM and Metaheuristics,” The Hong Kong University of Science and Technology, 2019.
- [2] S. A. Miller, A. Horvath, and P. J. M. Monteiro, “Readily implementable techniques can cut annual CO2 emissions from the production of concrete by over 20%,” *Environ. Res. Lett.*, vol. 11, no. 7, 2016, doi: 10.1088/1748-9326/11/7/074029.
- [3] H. Van Damme, “Concrete material science: Past, present, and future innovations,” *Cement and Concrete Research*, vol. 112, no. January. Elsevier, pp. 1–20, 2018, doi: 10.1016/j.cemconres.2018.05.002.
- [4] IEA and UNEP, “2019 Global Status Report for Buildings and Construction: Towards a zero-emissions, efficient and resilient buildings and construction sector,” 2019. [Online]. Available: <https://www.iea.org/reports/global-status-report-for-buildings-and-construction-2019>.
- [5] European Commission, “On Resource Efficient Opportunities in the Building Sector,” vol. COM(2014), pp. 1–10, 2014, [Online]. Available: <https://ec.europa.eu/environment/eussd/pdf/SustainableBuildingsCommunication.pdf>.
- [6] V. Yepes, “La optimización de estructuras,” *Universitat Politècnica de València*. <https://victoryepes.blogs.upv.es/2016/07/14/la-optimizacion-de-estructuras/> (accessed Jul. 08, 2020).
- [7] Y. Huang, J. Li, and J. Fu, “Review on application of artificial intelligence in civil engineering,” *CMES - Computer Modeling in Engineering and Sciences*, vol. 121, no. 3. Tech Science Press, pp. 845–875, 2019, doi: 10.32604/cmes.2019.07653.
- [8] D. Kumar and S. Akhtar, “Design Optimization of Reinforced Concrete Slabs using Various Optimization Techniques,” *Int. J. Trend Sci. Res. Dev.*, vol. 3, no. 5, pp. 45–58, 2019, doi: 10.31142/ijtsrd25231.
- [9] S. Basha and M. Latha, “Design optimization of reinforced concrete slabs using genetic

- algorithms,” *Int. J. Civ. Eng. Technol.*, vol. 9, no. 4, pp. 1370–1386, 2018, [Online]. Available: http://www.iaeme.com/ijciet/IJCIET_Paper.asp?sno=11200.
- [10] A. Kaveh and S. Bijari, “Optimum cost design of reinforced concrete one-way ribbed slabs using CBO, PSO and democratic PSO algorithms,” *Asian J. Civ. Eng.*, vol. 15, no. 6, pp. 788–802, 2014, [Online]. Available: www.SID.ir.
- [11] E. Ghandi and N. Shokrollahi, “Optimum Cost Design of Reinforced Slabs USING Cuckoo Search Optimization Algorithm,” *Int. J. Optim. Civ. Eng.*, vol. 7, no. 4, pp. 539–564, 2017, [Online]. Available: <https://www.researchgate.net/publication/316846968>.
- [12] M. Aldwaik and H. Adeli, “Cost optimization of reinforced concrete flat slabs of arbitrary configuration in irregular highrise building structures,” *Struct. Multidiscip. Optim.*, vol. 54, no. 1, pp. 151–164, Jul. 2016, doi: 10.1007/s00158-016-1483-5.
- [13] J. De Almeida, “Otimização de lajes nervuradas de concreto armado, utilizando algoritmo genético,” Universidade Federal de Pernambuco, 2018.
- [14] Ó. Cuevas and F. Fernández, *Aspectos fundamentales del concreto reforzado.*, 4ta ed. México, 2005.
- [15] R. Rochel, *Hormigón Reforzado*, 1ra ed. Medellín, 2007.
- [16] J. McCormac and R. Beown, *Diseño de Concreto Armado*, 8va ed. Mexico, 2011.
- [17] T. Harmsen, *El concreto y sus componenetes*, 3ra ed. 2002.
- [18] ACI 318-19, *Building Code Requirements for Structural Concrete and Commentary*. American Concrete Institute, 2019.
- [19] A. Blanco, *Estructuración y Diseño de Edificaciones de Concreto armado*, 2nd ed. Lima, 1994.
- [20] ASCE 7-16, *Minimum Design Loads and Associated Criteria for Buildings and Other Structures*, no. 798. American Society of Civil Engineers, 2016.
- [21] J. Ponce, A. Torres, F. Quezada, A. Silva, E. Martínez, and A. Casali, *Inteligencia*

- Artificial*, vol. 1, no. 5. 2014.
- [22] P. Lu, S. Chen, and Y. Zheng, “Artificial intelligence in civil engineering,” *Mathematical Problems in Engineering*, vol. 2012. pp. 1–23, 2012, doi: 10.1155/2012/145974.
- [23] H. Salehi and R. Burgueño, “Emerging artificial intelligence methods in structural engineering,” *Engineering Structures*, vol. 171. Elsevier Ltd, pp. 170–189, Sep. 15, 2018, doi: 10.1016/j.engstruct.2018.05.084.
- [24] Z. Abdmouleh, A. Gastli, L. Ben-Brahim, M. Haouari, and N. A. Al-Emadi, “Review of optimization techniques applied for the integration of distributed generation from renewable energy sources,” *Renew. Energy*, vol. 113, pp. 266–280, 2017, doi: 10.1016/j.renene.2017.05.087.
- [25] J. S. Wang and S. X. Li, “An Improved Grey Wolf Optimizer Based on Differential Evolution and Elimination Mechanism,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–21, 2019, doi: 10.1038/s41598-019-43546-3.
- [26] G. Wang, “A Comparative Study of Cuckoo Algorithm and Ant Colony Algorithm in Optimal Path Problems,” in *MATEC Web of Conferences*, 2018, vol. 232, doi: 10.1051/mateconf/201823203003.
- [27] H. Sun, C. Y. Yang, C. W. Lin, J. S. Pan, V. Snasel, and A. Abraham, “Artificial Bee Colony Using Opposition-Based Learning,” in *Advances in Intelligent Systems and Computing*, 2015, vol. 329, no. 289, pp. 3–10, doi: 10.1007/978-3-319-12286-1.
- [28] Q. Pan, C. Darabos, J. Moore, and X. Yang, *Cuckoo Search and Firefly Algorithm Theory and Applications*, vol. 516. 2014.
- [29] M. Shehab, A. T. Khader, and M. A. Al-Betar, “A survey on applications and variants of the cuckoo search algorithm,” *Appl. Soft Comput. J.*, vol. 61, no. March, pp. 1041–1059, 2017, doi: 10.1016/j.asoc.2017.02.034.
- [30] G. G. Wang, A. H. Gandomi, and A. H. Alavi, “An effective krill herd algorithm with migration operator in biogeography-based optimization,” *Appl. Math. Model.*, vol. 38, no. 9–10, pp. 2454–2462, May 2014, doi: 10.1016/j.apm.2013.10.052.

- [31] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
- [32] R. S. Parpinelli and H. S. Lopes, “New inspirations in swarm intelligence: A survey,” *Int. J. Bio-Inspired Comput.*, vol. 3, no. 1, pp. 1–16, 2011, doi: 10.1504/IJBIC.2011.038700.
- [33] M. Crepinsek, S. H. Liu, and M. Mernik, “Exploration and exploitation in evolutionary algorithms: A survey,” *ACM Comput. Surv.*, vol. 45, no. 3, 2013, doi: 10.1145/2480741.2480752.
- [34] G. Dreyfus, *Neural Networks Methodology and Applications*, vol. 2, no. 10. 2005.
- [35] J. V. Tu, “Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes,” *J. Clin. Epidemiol.*, vol. 49, no. 11, pp. 1225–1231, 1996, doi: 10.1016/S0895-4356(96)00002-9.
- [36] I. Nunes, D. Hernane, R. Andrade, L. Bartocci, and S. dos Reis, “A Practical Course,” in *Artificial Neural Networks*, 1st ed., 2017, p. 307.
- [37] W. Rivas and B. Mazon, “Generalidades de las redes neuronales artificiales,” in *Redes neuronales artificiales aplicadas al reconocimiento de patrones*, E. UTMACH, Ed. 2018, pp. 1–29.
- [38] J. Levy, R. Florez, and J. Fernandez, “Las redes neuronales artificiales. Fundamentos teoricos y aplicaciones practicas,” 2008.
<https://books.google.com.pe/books?id=X0uLwi1Ap4QC&pg=PA7&lpg=PA7&dq=Las+redes+neuronales+artificiales++fundamentos+teóricos+y+aplicaciones+prácticas&source=bl&ots=gNQympkvXd&sig=ACfU3U0e2qmQPYYgXxaZ7zCEeiHTHq13ig&hl=es-419&sa=X&ved=2ahUKEwjnm4fC5pXqAhW> (accessed Jun. 22, 2020).
- [39] R. Pino, A. Gómez, and N. De Abajo, “Introducción a la inteligencia artificial: sistemas expertos, redes neuronales y computación evolutiva,” 2001.
<https://books.google.es/books?hl=es&lr=&id=RKqLMCw3IUkC&oi=fnd&pg=PA10&dq=computación+evolutiva&ots=iGMAh3u47U&sig=DDN1KSQnxhqMCtQdfzMM29i8vHQ#v=onepage&q&f=false> (accessed Jun. 22, 2020).

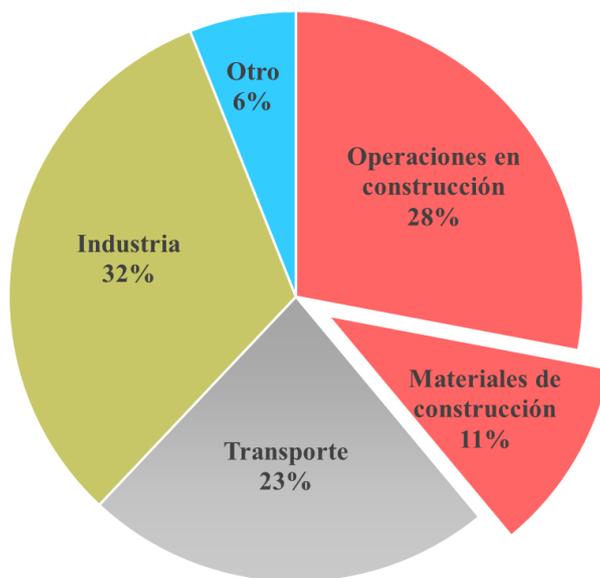
- [40] P. Majumder and T. I. Eldho, “Artificial Neural Network and Grey Wolf Optimizer Based Surrogate Simulation-Optimization Model for Groundwater Remediation,” *Water Resour. Manag.*, vol. 34, no. 2, pp. 763–783, 2020, doi: 10.1007/s11269-019-02472-9.
- [41] G. Osipov, A. Panov, and K. Yakovlev, *Artificial Intelligence: 5th RAAI Summer School*, vol. 11866. Cham: Springer International Publishing, 2019.
- [42] R. Stuart and N. Peter, *Inteligencia Artificial*, vol. 2, no. 6. 2007.
- [43] N. Zheng and P. Mazumder, “Fundamentals and Learning of Artificial Neural Networks,” in *Learning in Energy-Efficient Neuromorphic Computing: Algorithm and Architecture Co-Design*, 2020, pp. 11–60.
- [44] A. Rothwell, *Optimization methods in structural design*. 2017.
- [45] U. Krish, “Structural Optimization Fundamentals and Applications Springer-Verlag,” 1993. [Online]. Available: <https://link-springer-com.ezproxy.lib.monash.edu.au/content/pdf/10.1007%2F978-3-642-84845-2.pdf>.
- [46] R. T. Haftka, Z. Gürdal, and M. P. Kamat, *Elements of Structural Optimization*, vol. 1. 1992.
- [47] S. Mirjalili and J. Song, *Multi-Objective Optimization using Artificial Intelligence Techniques*. 2020.
- [48] P. Christensen and A. Klarbring, *An Introduction to Structural Optimization*, 1st ed., vol. 153. Springer Netherlands, 2009.
- [49] M. Afzal, Y. Liu, J. C. P. Cheng, and V. J. L. Gan, “Reinforced concrete structural design optimization: A critical review,” *J. Clean. Prod.*, vol. 260, p. 120623, 2020, doi: 10.1016/j.jclepro.2020.120623.
- [50] R. Hernández, C. Fernández, and M. Baptista, *Metodología de la Investigación Científica*, 6th ed. 2018.
- [51] G. Ottazzi, “Apuntes de concreto armado I.” Pontificia Universidad Católica del Perú,

Lima, pp. 1–530, 2015.

- [52] W. Long, X. Liang, S. Cai, J. Jiao, and W. Zhang, “A modified augmented Lagrangian with improved grey wolf optimization to constrained optimization problems,” *Neural Comput. Appl.*, vol. 28, no. s1, pp. 421–438, 2017, doi: 10.1007/s00521-016-2357-x.

VIII. Anexos

ANEXO 01: FIGURAS

Figura 37: Emisiones globales de CO₂ por sector, 2018

Fuente: Elaboración propia. En base a IEA and UNEP 2019: p. 12

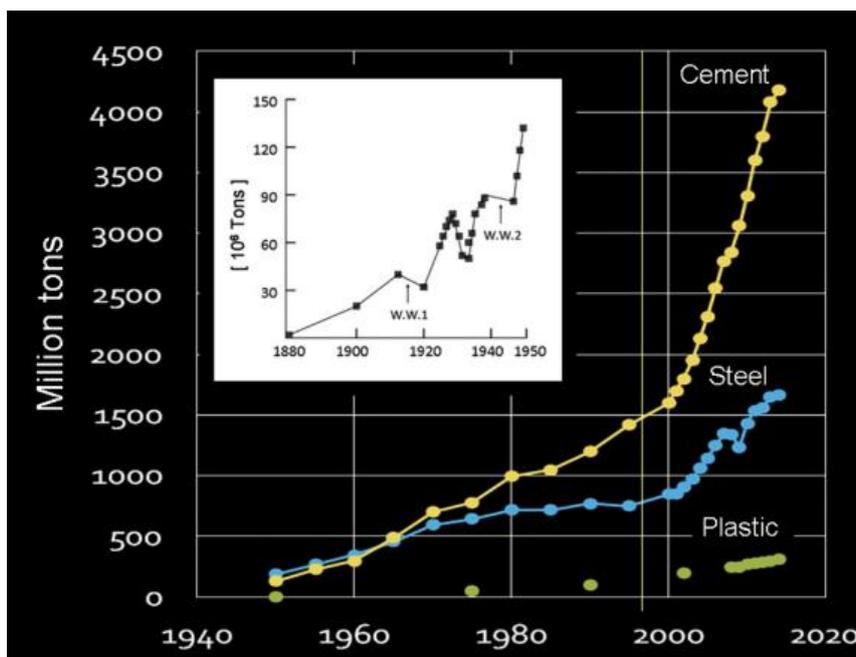


Figura 38: Producción de cemento a nivel mundial en millones de toneladas

Fuente: Van Damme 2018: p.2

Ficha N°:	----
TITULO	
AUTOR	FECHA / LUGAR
INFORME	
PROBLEMA	
OBJETIVO	
METODOLOGÍA	VARIABLES
SOLUCION	
CONCLUSIONES	
FUTURAS INV.	
FUENTE	URL

Figura 39: Modelo de ficha de análisis documental

Fuente: Elaboración propia

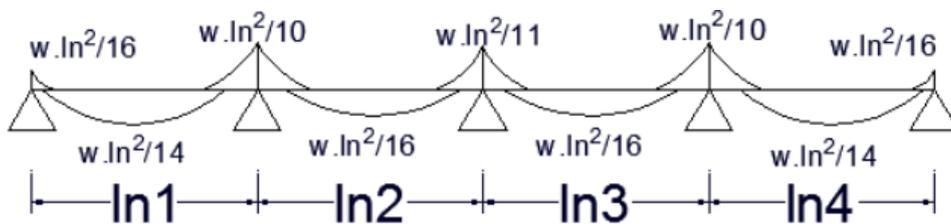


Figura 40 Momentos en losas continuas por el Método de los Coeficientes ACI

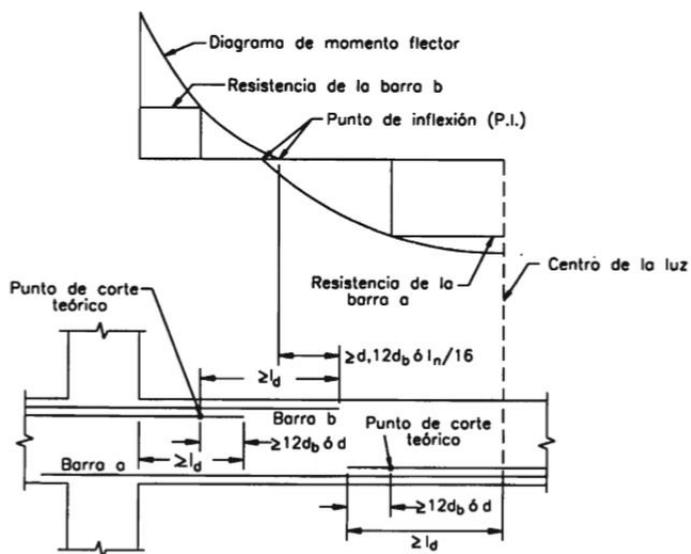
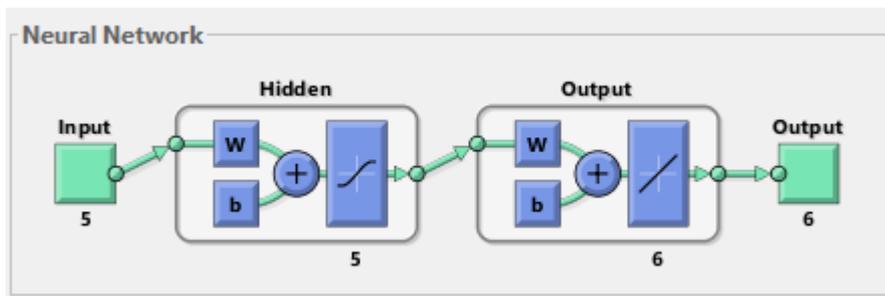
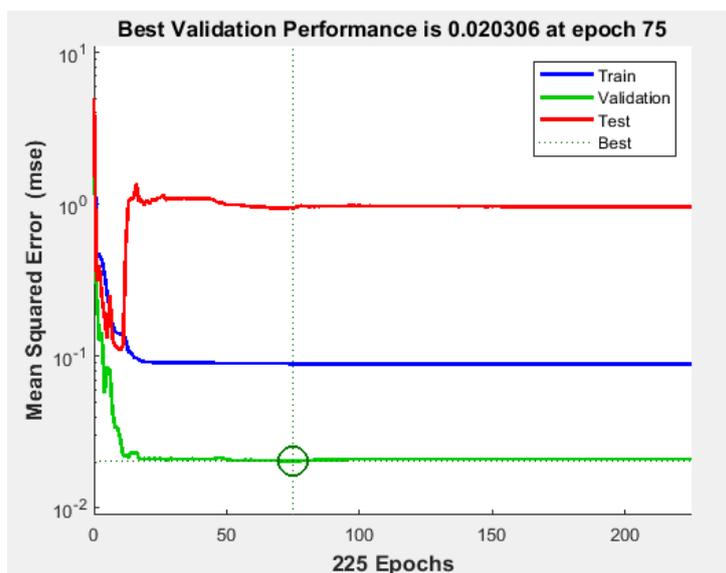


Figura 41: Consideraciones para el corte de acero en regiones de momento positivo y negativo

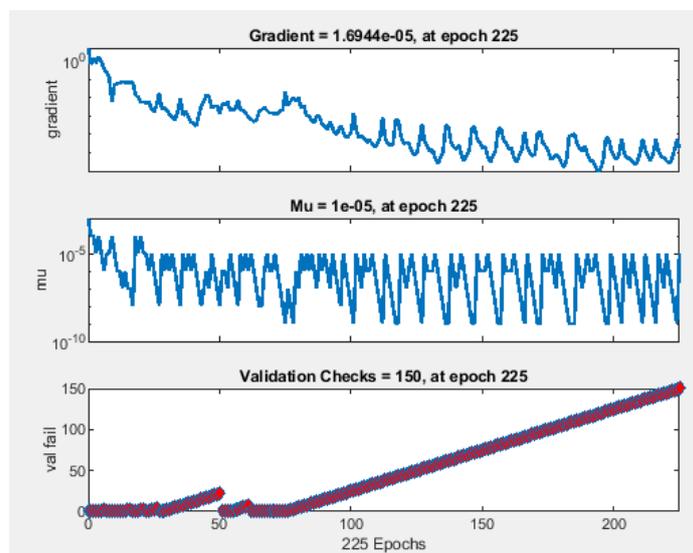
Fuente: Harmsen 2002: p. 106



Arquitectura red neuronal 2



Mejor valor del error medio cuadrático



Gradiente del error

Figura 42: Entrenamiento de la red neuronal 2

Fuente: Elaboración propia

```

Command Window

>> x1, x2, x3
x1 =
    26.7956
x2 =
    38.4903
x3 =
    10.2725
>> g1 = (km*Ln.^2)*(1.4*(wg*(b.*x1)/10000+DLs*b/100)+1.7*LL*b/100)./...
(0.9*(pi()/400*(x3.^2).*b./x2).*fy.*(x1-rec-1/2-...
0.5*(pi()/400*(x3.^2).*b./x2).*fy./(0.85.*fc.*b))*0.01);
>> g2 = 0.5*kv*Ln*(1.4*(wg*(b.*x1)/10000+DLs*b/100)+1.7*LL*b/100)./...
(0.85*0.53.*sqrt(fc)).*(x1-rec-1/2)*b)-1;
>> g3 = (0.0018*b.*x1)./(pi()/400*(x3.^2)*b./x2)-1;
>> g4 = ((pi()/400*(x3.^2)*b./x2)./(b.*x1))./(0.75.*pb)-1;
>> g5 = x2./min(3.*x1,40)-1;
>> g6 = max(x3./10,25.4)./x2-1;
>> g7 = (100*kd*Ln)./x1-1;
>> g = [g1 g2 g3 g4 g5 g6 g7];
>> g
g =
    0.9992    -0.8318     1.2400     1.2400    -0.0377    -0.3401    -0.5335
fx

```

Figura 43: Verificación de restricciones losas macizas -gwo

Fuente: Elaboracion propia

```

Editor - C:\Users\Usuario\Documents\MATLAB\pruebas red neuronal\MATLAB\RNA_AsPos -\Run.m
Red_AsPos.m x Red_AsNeg.m x Run.m x +
5 %% Carga red entrenada
6 load('Red1_AsPos.mat'); % Red entrenada 1
7 load('Red2_AsNeg.mat'); % Red entrenada 2
8
9 %% Ingrese variables de entrada
10 L = 21.2; %Longitud de la edificacion
11 T = 3; %Numero de tramos continuos ..de 2 a 4
12 A = 4; %Numero de apoyos ... de 3 a 5
13 Ln1 = 6.5; %Luz de primer tramo..entre 6.00 y 7.50 m
14 Ln2 = 6.75; %Luz de segundo tramo..
15 Ln3 = 6.75;
16 Ln4 = 0;
17
18 %% Activar el vector de entrada
19 %x = [L T 1 Ln1; L T 2 Ln2]'; %Para 2 tramos
20 x = [L T 1 Ln1; L T 2 Ln2; L T 3 Ln3]'; %para 3 tramos
21 % x = [L T 1 Ln1; L T 2 Ln2; L T 3 Ln3; L 4 4 Ln4]'; %para 4 tramos
22
Command Window

```

Figura 44: Ingreso de variables a la red neuronal para el diseño del modelo 3

Fuente: Elaboracion propia

ANEXO 02: TABLAS

Tabla 30: Pesos unitarios de los materiales de construcción

Materiales	Peso (kgf/m³)
Albañilería	
<i>Unidades de arcilla cocida sólidas</i>	1800.0
<i>Unidades de arcilla cocida huecas</i>	1350.0
Concreto armado	2400.0
Acero	7850
Enlucido o Revoque	
<i>Mortero de cemento</i>	2000.0
<i>Mortero de cal y cemento</i>	1850.0

Fuente: Elaboración propia. En base a E-060 y ASCE 7-16

Tabla 31: Cargas vivas mínimas distribuidas uniformemente

Ocupación o uso	Carga repartida (kgf/m²)
Almacenaje	500.0
Bibliotecas	
<i>Sala de lectura</i>	300.0
<i>Corredores y escaleras</i>	400.0
Centros de educación	
<i>Aulas</i>	250.0
<i>Talleres</i>	350.0
<i>Laboratorios</i>	300.0
<i>Corredores y escaleras</i>	400.0
Hoteles	
<i>Cuartos</i>	200.0
<i>Almacenaje y servicios</i>	500.0
<i>Corredores y escaleras</i>	400.0
Oficinas	
<i>Exceptuando salas de archivo y computación</i>	250.0
<i>Salas de archivo</i>	500.0
<i>Salas de computación</i>	250.0
<i>Corredores y escalera</i>	400.0
Tiendas	500.0
<i>Corredores y escaleras</i>	500.0
Viviendas	200.0
<i>Corredores y escaleras</i>	200.0

Fuente: Elaboración propia. En base a E-020 y ASCE 7-16

Tabla 32: Diámetros de acero corrugado en [mm] y [pulg]

Diámetro #	[Pulg]	[mm]	Área [cm2]	Peso [kg/m]
2	1/4	6.4	0.32	0.207
3	3/8	9.5	0.71	0.559
4	1/2	12.7	1.27	0.994
5	5/8	15.9	1.98	1.554
6	3/4	19.1	2.84	2.237
8	1	25.4	5.1	3.978

Fuente: Elaboración propia

Tabla 33: Data de entrenamiento redes neuronal 1

L	$\#Tramos$	$Paño$	L_n	A_{smin}	db_b	x_b	L_b
13.60	2	1	5.85	4	3	0.90	2.90
13.60	2	2	5.85	4	3	2.80	1.80
18.75	3	1	5.85	4	3	0.85	3.05
18.75	3	2	5.85	4	1	0.00	0.00
18.75	3	3	5.85	4	3	1.95	3.05
19.75	3	1	5.85	4	3	0.95	2.75
19.75	3	3	5.85	4	3	2.15	2.75
20.75	3	2	5.85	4	1	0.00	0.00
20.25	3	1	5.85	4	3	1.00	2.55
20.25	3	3	5.85	4	3	2.30	2.55
21.25	3	2	5.85	4	1	0.00	0.00
20.75	3	1	5.85	4	3	0.90	2.95
19.75	3	2	5.85	4	1	0.00	0.00
19.75	3	3	5.85	4	3	1.90	3.15
12.90	2	1	6.00	4	3	1.00	2.60
12.90	2	2	6.00	4	3	2.70	2.60
27.30	4	1	6.20	4	3	0.80	3.35
27.30	4	4	6.20	4	3	2.05	3.35
27.30	4	2	6.20	4	1	0.00	0.00
27.30	4	3	6.20	4	1	0.00	0.00
27.80	4	4	6.20	4	3	2.05	3.35
27.80	4	4	6.20	4	3	1.00	4.40
27.80	4	4	6.20	4	3	2.05	3.35
13.60	2	1	6.35	4	3	0.90	3.15
13.60	2	2	6.35	4	3	2.60	3.15
14.60	2	1	6.35	4	3	0.90	2.85
14.30	2	2	6.35	4	3	2.50	2.95
20.25	3	1	6.35	4	3	0.70	3.80
20.25	3	2	6.35	4	1	0.00	0.00
20.25	3	3	6.35	4	3	1.85	3.80
21.25	3	1	6.35	4	3	1.20	3.10
21.25	3	3	6.35	4	3	2.05	3.10
21.65	3	2	6.35	4	1	0.00	0.00
20.85	3	1	6.35	4	3	0.70	3.70
20.85	3	3	6.35	4	3	1.95	3.70
21.25	3	2	6.35	4	1	0.00	0.00
22.25	3	1	6.35	4	3	0.70	3.75
20.95	3	2	6.35	4	1	0.00	0.00
20.95	3	3	6.35	4	4	1.75	3.95
27.80	4	1	6.45	4	3	0.70	3.70
14.15	2	2	6.55	4	3	5.90	0.00
14.15	2	1	6.70	0	0	0.00	0.00
27.30	4	2	6.70	4	1	0.00	0.00
27.30	4	3	6.70	4	1	0.00	0.00
27.30	4	1	6.70	4	4	0.60	4.15
27.30	4	4	6.70	4	4	1.95	4.15
28.30	4	1	6.70	4	4	0.60	4.15

28.30	4	2	6.70	4	1	0.00	0.00
28.30	4	3	6.70	4	1	0.00	0.00
28.30	4	4	6.70	4	4	1.95	4.15
27.80	4	1	6.70	4	4	0.60	4.15
27.80	4	2	6.70	4	1	0.00	0.00
27.80	4	3	6.70	4	1	0.00	0.00
27.80	4	3	6.70	4	1	0.00	0.00
27.80	4	1	6.70	4	4	0.60	3.70
27.80	4	2	6.70	4	1	0.00	0.00
27.80	4	3	6.70	4	1	0.00	0.00
13.60	2	2	6.85	0	0	0.00	0.00
13.60	2	1	6.85	0	0	0.00	0.00
21.75	3	1	6.85	4	4	0.65	4.45
21.75	3	2	6.85	4	1	0.00	0.00
21.75	3	3	6.85	4	4	1.85	4.45
19.75	3	2	6.85	4	1	0.00	0.00
20.75	3	1	6.85	4	4	0.55	4.60
20.75	3	3	6.85	4	4	1.70	4.60
21.25	3	1	6.85	4	4	0.55	4.55
21.25	3	3	6.85	4	4	1.75	4.55
20.75	3	2	6.85	4	1	0.00	0.00
20.75	3	3	6.85	4	4	1.90	4.35
19.75	3	1	6.85	4	4	0.55	4.50
15.05	2	1	6.90	0	0	0.00	0.00
22.25	3	1	6.9	0	0	0.00	0.00
22.25	3	3	6.9	0	0	0.00	0.00
27.80	4	2	6.95	4	3	2.90	1.50
20.85	3	2	6.95	4	1	0.00	0.00
14.30	2	1	7.05	0	0	0.00	0.00
21.65	3	1	7.05	0	0	0.00	0.00
21.65	3	3	7.05	0	0	0.00	0.00
20.95	3	1	7.05	0	0	0.00	0.00
21.25	3	1	7.1	4	4	0.50	4.90
21.25	3	3	7.1	4	4	1.70	4.90
15.05	2	2	7.25	0	0	0.00	0.00
22.25	3	2	7.25	4	1	0.00	0.00
14.60	2	2	7.35	0	0	0.00	0.00
21.25	3	2	7.35	4	3	2.75	1.85
20.25	3	2	7.35	4	3	2.55	2.25
22.25	3	2	7.35	4	1	0.00	0.00
22.25	3	3	7.35	0	0	0.00	0.00
28.30	4	1	6.95	0	0	0.00	0.00
28.30	4	2	6.70	4	1	0.00	0.00
28.30	4	3	6.95	4	3	0.00	0.00
28.30	4	4	6.20	4	3	2.10	3.30

Tabla 34: Data de entrenamiento red neuronal 2

<i>L</i>	<i>#Apoyos</i>	<i>Apoyo</i>	<i>L_{ni}</i>	<i>L_{nj}</i>	<i>A_{smin}</i>	<i>db bas</i>	<i>Ld_db_i</i>	<i>Ld_b_i</i>	<i>Ld_db_j</i>	<i>Ld_b_j</i>
12.90	2	1	0.00	6.00	5	1	0.00	0.00	0.80	0.00
12.90	2	2	6.00	6.00	5	3	0.80	1.55	0.80	1.55
12.90	2	3	6.00	0.00	5	1	0.80	0.00	0.00	0.00
13.60	2	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
13.60	2	2	6.35	6.35	5	4	0.80	1.60	0.80	1.60
13.60	2	3	6.35	0.00	5	1	0.80	0.00	0.00	0.00
14.15	2	1	0.00	6.70	5	1	0.00	0.00	0.80	0.00
14.15	2	2	6.70	6.55	0	0	0.00	0.00	0.00	0.00
14.15	2	3	6.55	0.00	5	1	0.80	0.00	0.00	0.00
15.05	2	1	0.00	6.90	5	1	0.00	0.00	0.80	0.00
15.05	2	2	6.90	7.25	0	0	0.00	0.00	0.00	0.00
15.05	2	3	7.25	0.00	5	1	0.80	0.00	0.00	0.00
13.60	2	1	0.00	5.85	5	1	0.00	0.00	0.80	0.00
13.60	2	2	5.85	6.85	5	4	0.80	1.00	0.00	0.00
13.60	2	3	6.85	0.00	5	1	0.80	0.00	0.00	0.00
14.60	2	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
14.60	2	2	6.35	7.35	0	0	0.00	0.00	0.00	0.00
14.60	2	3	7.35	0.00	5	1	0.80	0.00	0.00	0.00
13.60	2	1	0.00	6.85	5	1	0.00	0.00	0.80	0.00
13.60	2	2	6.85	5.85	5	4	0.00	0.00	0.80	1.70
13.60	2	3	5.85	0.00	5	1	0.80	0.00	0.00	0.00
14.30	2	1	0.00	7.05	5	1	0.00	0.00	0.80	0.00
14.30	2	2	7.05	6.35	5	5	0.00	0.00	0.90	1.70
14.30	2	3	6.35	0.00	5	1	0.80	0.00	0.00	0.00
20.75	3	1	0.00	5.85	5	1	0.00	0.00	0.80	0.00
20.75	3	2	5.85	6.85	5	4	0.80	1.25	0.80	1.55
20.75	3	3	6.85	6.85	5	4	0.80	1.80	0.80	1.45
20.75	3	4	6.85	0.00	5	1	0.80	0.00	0.00	0.00
22.25	3	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
22.25	3	2	6.35	7.35	5	3	0.80	1.35	0.80	1.65
22.25	3	3	7.35	7.35	5	4	0.95	1.85	0.00	0.00
22.25	3	4	7.35	0.00	5	1	0.80	0.00	0.00	0.00
19.75	3	1	0.00	6.85	5	1	0.00	0.00	0.80	0.00
19.75	3	2	6.85	5.85	5	3	0.80	1.40	0.80	1.85
19.75	3	3	5.85	5.85	5	3	1.55	0.00	1.10	0.00
19.75	3	4	5.85	0.00	5	1	0.80	0.00	0.00	0.00
20.95	3	1	0.00	7.05	5	1	0.00	0.00	0.80	0.00
20.95	3	2	7.05	6.35	5	4	0	0.00	0.80	1.85
20.95	3	3	6.35	6.35	5	3	0.8	1.65	0.80	1.25
20.95	3	4	6.35	0.00	5	1	0.8	0.00	0.00	0.00
18.75	3	1	0.00	5.85	5	1	0	0.00	0.80	0.00
18.75	3	2	5.85	5.85	5	1	1.2	0.00	1.55	0.00
18.75	3	3	5.85	5.85	5	1	1.05	0.00	1.05	0.00
18.75	3	4	5.85	0.00	5	1	0.8	0.00	0.00	0.00

20.25	3	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
20.25	3	2	6.35	6.35	5	3	0.80	1.30	0.80	1.50
20.25	3	3	6.35	6.35	5	3	0.80	1.80	0.80	1.00
20.25	3	4	6.35	0.00	5	1	0.80	0.00	0.00	0.00
21.75	3	1	0.00	6.85	5	1	0.00	0.00	0.80	0.00
21.75	3	2	6.85	6.85	5	4	0.80	1.40	0.80	1.60
21.75	3	3	6.85	6.85	5	4	0.80	1.90	0.80	1.10
21.75	3	4	6.85	0.00	5	1	0.80	0.00	0.00	0.00
22.25	3	1	0.00	6.90	5	1	0.00	0.00	0.80	0.00
22.25	3	2	6.90	7.25	5	4	0.00	0.00	0.80	1.75
22.25	3	3	7.25	6.90	5	4	0.8	1.75	0.00	0.00
22.25	3	4	6.90	0.00	5	1	0.80	0.00	0.00	0.00
19.75	3	1	0.00	5.85	5	1	0.00	0.00	0.80	0.00
19.75	3	2	5.85	6.85	5	3	0.8	1.35	0.8	1.55
19.75	3	3	6.85	5.85	5	3	0.8	1.55	0.8	1.35
19.75	3	4	5.85	0.00	5	1	0.80	0.00	0.00	0.00
21.25	3	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
21.25	3	2	6.35	7.35	5	4	0.8	1.45	0.8	1.65
21.25	3	3	7.35	6.35	5	4	0.8	1.65	0.8	1.45
21.25	3	4	6.35	0.00	5	1	0.80	0.00	0.00	0.00
20.75	3	1	0.00	6.85	5	1	0.00	0.00	0.80	0.00
20.75	3	2	6.85	5.85	5	3	0.8	1.3	0.8	1.85
20.75	3	3	5.85	6.85	5	3	0.8	1.85	0.8	1.3
20.75	3	4	6.85	0.00	5	1	0.80	0.00	0.00	0.00
21.65	3	1	0.00	7.05	5	1	0.00	0.00	0.80	0.00
21.65	3	2	7.05	6.35	5	4	0	0	0.8	1.85
21.65	3	3	6.35	7.05	5	4	0.8	1.85	0	0
21.65	3	4	7.05	0.00	5	1	0.80	0.00	0.00	0.00
20.25	3	1	0.00	5.85	5	1	0.00	0.00	0.80	0.00
20.25	3	2	5.85	7.35	5	3	0.8	1.4	0.8	1.55
20.25	3	3	7.35	5.85	5	3	0.8	1.55	1.4	0.8
20.25	3	4	5.85	0.00	5	1	0.80	0.00	0.00	0.00
20.85	3	1	0.00	6.35	5	1	0.00	0.00	0.80	0.00
20.85	3	2	6.35	6.95	5	3	0.8	1.35	0.8	1.6
20.85	3	3	6.95	6.35	5	3	0.8	1.6	0.8	1.35
20.85	3	4	6.35	0.00	5	1	0.80	0.00	0.00	0.00
21.25	3	1	0.00	6.85	5	1	0.00	0.00	0.80	0.00
21.25	3	2	6.85	6.35	5	3	0.8	1.35	0.8	1.75
21.25	3	3	6.35	6.85	5	3	0.8	1.75	0.8	1.35
21.25	3	4	6.85	0.00	5	1	0.80	0.00	0.00	0.00
21.25	3	1	0.00	7.10	5	1	0.00	0.00	0.80	0.00
21.25	3	2	7.10	5.85	5	3	0.8	1.35	0.8	2
21.25	3	3	5.85	7.10	5	3	0.8	2	0.8	1.35
21.25	3	4	7.10	0.00	5	1	0.8	0	0	0
27.3	4	1	0.00	6.20	5	1	0	0	0.8	0
27.3	4	2	6.20	6.70	5	3	0.8	1.4	0.8	1.8

27.3	4	3	6.70	6.70	5	3	1.15	0	1.15	0
27.3	4	4	6.70	6.20	5	3	0.8	1.95	0.8	1.4
27.3	4	5	6.20	0.00	5	1	0.8	0	0	0
27.3	4	1	0.00	6.70	5	1	0	0	0.8	0
27.3	4	2	6.70	6.20	5	4	0.8	1.5	0.8	1.6
27.3	4	3	6.20	6.20	5	4	0.95	0	0.95	0
27.3	4	4	6.20	6.70	5	4	0.8	1.85	0.8	1.35
27.3	4	5	6.70	0.00	5	1	0.8	0	0	0
28.3	4	1	0.00	6.70	5	1	0	0	0.8	0
28.3	4	2	6.70	6.70	5	4	0.8	1.45	0.8	0.8
28.3	4	3	6.70	6.70	5	4	1.1	0	1.1	0
28.3	4	4	6.70	6.70	5	4	0.85	1.85	0.8	1.45
28.3	4	5	6.70	0.00	5	1	0.8	0	0	0
27.8	4	1	0.00	6.70	5	1	0	0	0.8	0
27.8	4	2	6.70	6.70	5	4	0.8	1.45	0.85	1.85
27.8	4	3	6.70	6.70	5	4	1.3	0	1.25	0
27.8	4	4	6.70	6.20	5	4	0.8	1.6	0.8	1.4
27.8	4	5	6.20	0.00	5	1	0.8	0	0	0
27.8	4	1	0.00	6.45	5	1	0	0	0.8	0
27.8	4	2	6.45	6.95	5	4	0.8	1.45	0.8	1.7
27.8	4	3	6.95	6.70	5	4	1.3	0	1.3	0
27.8	4	4	6.70	6.20	5	3	0.8	1.5	0.8	1.35
27.8	4	5	6.20	0.00	5	1	0.8	0	0	0
28.3	4	1	0.00	6.95	5	1	0	0	0.8	0
28.3	4	2	6.95	6.70	5	4	0	0	0.8	1.85
28.3	4	3	6.70	6.95	5	4	1.5	0	1.25	0
28.3	4	4	6.95	6.20	5	3	0.8	1.6	0.8	1.4
28.3	4	5	6.20	0.00	5	1	0.8	0	0	0

Tabla 35: Matriz de umbrales y pesos sinápticos capa oculta de la red neuronal 2

FILA/ COL	1	FILA/ COL	1	2	3	4	
1	-21	1	-8.5795	-53.3535	47.5968	-9.2166	-7.6097
2	11.2	2	0.2939	-0.2341	11.7818	-3.2990	-0.5505
3	6.92	3	-8.3286	0.3688	0.4042	-9.7731	-4.1407
4	-0.7	4	-1.1659	0.9083	-0.9925	0.8724	-2.3698
5	279	5	-137.0815	193.2812	-29.3478	-51.2946	-223.1627

Fuente: Elaboración propia

Tabla 36: Matriz de umbrales y pesos sinápticos capa de salida de la red neuronal 2

FILA/ COL	1	FILA/ COL	1	2	3	4	5
1	0.077	1	0.007	-0.084	-0.002	-0.005	1.009
2	4.465	2	0.638	-5.372	-0.555	-0.589	0.814
3	-22.956	3	0.319	22.830	-0.393	-0.066	0.263
4	-13.877	4	0.540	13.065	-0.554	-0.681	0.453
5	-1.454	5	-0.014	0.468	-0.011	-0.552	0.524
6	25.531	6	0.728	-26.898	-0.724	-0.493	0.834

Fuente: Elaboración propia

ANEXO 03: CÓDIGOS MATLAB

Código 1: Programa de diseño de losas aligeradas en una dirección

```

%           DISEÑO DE LOSAS ALIGERADAS EN UNA DIRECCIÓN
%-----
%fc : Resistencia a compresión del concreto
%fy : Límite de fluencia del acero
%b  : Ancho del ala de la viga TEE
%bw : Ancho del alma de la viga TEE
%h  : Peralte de la viga TEE
%Mu : Momento requerido por flexión

%-----

function [Asmin_A Asmin_T Mmax_A Mmax_T As_A As_T] = Losa_alig_1D(fc, fy,
b, bw, h, Mu)

if fc<280
    beta1=0.85;
elseif fc<550
    beta1=0.85-0.55*(fc-280)/70;
elseif fc>550
    beta1=0.65;
end
hf = 5;           %Espesor del ala
d  = h - 3;      %Peralte efectivo/ recubrimiento = 2cm

%-----
%           DISEÑO EN LOS APOYOS
%-----

%Cálculo del área de acero mínimo en los apoyos
Yg = (b*hf*(h-hf/2)+0.5*bw*(h-hf).^2)/(hf*b+bw*(h-hf));
Yt = h-Yg;
Ig1 = 1/12*b*hf.^3 + b*hf*(Yt-hf/2).^2;
Ig2 = 1/12*bw*(h-hf).^3 + bw*(h-hf)*(Yg-0.5*(h-hf)).^2;
Ig = Ig1 + Ig2;
fr = 2*sqrt(fc); %kg/cm2
Mcr = fr*Ig/Yt/100; %kg-m
phiMn = 1.2*Mcr*100; %kg-cm
Asmin_A = phiMn/(0.9*fy*(d-0.5*(d-sqrt(d^2-2*phiMn/(0.85*0.9*fc*bw)))));

%Cálculo del momento resistente máximo negativo
Asb_A = 0.85*beta1*fc/fy*(6000/(6000+fy))*bw*d;
Asmax_A = 0.75*Asb_A;
Mmax_A = 0.9*Asmax_A*fy*(d-Asmax_A*fy/(2*0.85*fc*bw)); %kg-cm

%Momentos resistentes negativos
Mu_A = Mu (Mu<0) * (-100); %kg-cm
[NoMu_A,~] = size (Mu_A);
As_A = zeros (NoMu_A, 2);

for k = 1:NoMu_A
    if Mu_A(k)>Mmax_A
        As_A(k,1) = Mu_A(k)./(-100); %kg-m
        As_A(k,2) = 0; %Requiere As en compresión

        disp('Requiere As en compresión')
    end
end

```

```

%Cálculo del área de acero requerida en los apoyos
else
    As_a = Mu_A(k)/(0.9*fy*(d-0.5*(d-sqrt(d^2-2*Mu_A(k)/(0.85*0.9*fc*bw)))));
    As_A(k,1) = Mu_A(k)./(-100);
    As_A(k,2) = As_a;
end
end

%-----
%
%                               DISEÑO EN LOS TRAMOS
%-----

%Cálculo del área de acero mínimo en los tramos
pmin    = 0.70.*sqrt(fc)/fy; %RNE E.060
Asmin_T = pmin.*bw.*d;

%Cálculo del momento resistente máximo positivo
ab      = beta1*d*(6000/(6000+fy));
Asb_T   = 0.85*fc/fy*((b-bw)*hf+ab*bw);
Asmax_T = 0.75*Asb_T;
Mmax_T  = 0.9*Asmax_T*fy*(d-Asmax_T*fy/(2*0.85*fc*b)); %kg-cm

%Momentos resistentes positivos
Mu_T    = Mu(Mu>0)*100; %kg-cm
[NoMu_T,~] = size(Mu_T);
As_T    = zeros(NoMu_T,2);

%Verificación del ala en compresión
a = max(Mu_T)/(0.9*(d-hf/2)*0.85*fc*b);
if a>hf;
    disp('Viga TEE')
else
    for i = 1:NoMu_T
        if Mu_T(i)>Mmax_T
            As_T(i,1) = Mu_T(i)./(100); %kg-m
            As_T(i,2) = 0; %Requiere As en compresión
            disp('Requiere As en compresión')

            %Cálculo del área de acero requerida en los tramos
            else
                As_t = Mu_T(i)/(0.9*fy*(d-0.5*(d-sqrt(d^2-2*Mu_T(i)/(0.85*0.9*fc*b)))));
                As_T(i,1) = Mu_T(i)./(100);
                As_T(i,2) = As_t;
            end
        end
    end
end

Mmax_A = Mmax_A/100; %kg-m
Mmax_T = Mmax_T/100;

end

%-----
%===== & VASQUEZ REQUEJO JOSE MANUEL & =====%

```

Ejecutar programa

```

%          DISEÑO DE LOSAS ALIGERADAS EN UNA DIRECCIÓN
%-----
clear all; clc; format compact;

%% -----
%   INGRESA PROPIEDADES MATERIAL Y SECCION
%-----
fc = input('Ingresar resist compresión de concreto en kg/cm2: ');
fy = input('Ingresar límite de fluencia del acero en kg/cm2: ');
b  = input('Ingresar ancho del ala de vigueta TEE en cm: ');
bw = input('Ingresar ancho del alma de vigueta TEE en cm: ');
h  = input('Ingresar peralte de la vigueta TEE en cm: ');

%-----
%   IMPORTAR MOMENTOS ULTIMOS
%-----
% Cargar momentos últimos en kg-m con su propio signo + -
Mu = xlsread('ALIG_1DIR_LL=200_DLs=120.xlsx','Max_Min','H3:H50');

%% -----
%   PROGRAMA DE DISEÑO DE LOSAS ALIGERADAS EN 1D
%-----

[Asmin_A Asmin_T Mmax_A Mmax_T As_A As_T] = Losa_alig_1D(fc,fy,b,bw,h,Mu);

%% -----
%   EXPORTA MOMENTOS ÚLTIMOS Y ÁREAS DE ACERO
%-----
% Guardar en un archivo Excel el diseño

[nombre, direccion] = uinputfile({'*.xlsx','Archivo de Excel'},'Guardar
como');
Tabla = [As_A As_T];
xlswrite([direccion, nombre],Tabla);

%% -----
Asmin_A    %cm2   Área de acero mínimo en los apoyos para M<0
Asmin_T    %cm2   Área de acero mínimo en los tramos para M>0
Mmax_A     %kg-m  Momento resistente máximo negativo
Mmax_T     %kg-m  Momento resistente máximo positivo

%% -----
%===== & VASQUEZ REQUEJO JOSE MANUEL & =====%

```

Código 2: Algoritmo grey wolf optimization

```

%
% ----- %
% Grey Wolf Optimizer (GWO) source codes version 1.0 %
%
% Developed in MATLAB R2011b(7.13) %
%
% Author and programmer: Seyedali Mirjalili %
%
% e-Mail: ali.mirjalili@gmail.com %
%         seyedali.mirjalili@griffithuni.edu.au %
%
% Homepage: http://www.alimirjalili.com %
%
% Main paper: S. Mirjalili, S. M. Mirjalili, A. Lewis %
%             Grey Wolf Optimizer, Advances in Engineering %
%             Software , in press, %
%             DOI: 10.1016/j.advengsoft.2013.12.007 %
%
% ----- %
% Grey Wolf Optimizer
function
[Alpha_score,Alpha_pos,Convergence_curve]=GWO(SearchAgents_no,Max_iter,lb,ub,dim,fobj)
% initialize alpha, beta, and delta_pos
Alpha_pos=zeros(1,dim);
Alpha_score=inf; %change this to -inf for maximization problems
Beta_pos=zeros(1,dim);
Beta_score=inf; %change this to -inf for maximization problems
Delta_pos=zeros(1,dim);
Delta_score=inf; %change this to -inf for maximization problems
%Initialize the positions of search agents
Positions=initialization(SearchAgents_no,dim,ub,lb);
Convergence_curve=zeros(1,Max_iter);
l=0;% Loop counter
% Main loop
while l<Max_iter
    for i=1:size(Positions,1)

        % Return back the search agents that go beyond the boundaries of the search
space
        Flag4ub=Positions(i,*)>ub;
        Flag4lb=Positions(i,*)<lb;
        Positions(i,*)=(Positions(i,*).*(~(Flag4ub+Flag4lb)))+ub.*Flag4ub+lb.*Flag4lb;

        % Calculate objective function for each search agent
        fitness=fobj(Positions(i,));

        % Update Alpha, Beta, and Delta
        if fitness<Alpha_score
            Alpha_score=fitness; % Update alpha
            Alpha_pos=Positions(i,);
        end

        if fitness>Alpha_score && fitness<Beta_score
            Beta_score=fitness; % Update beta
            Beta_pos=Positions(i,);
        end

        if fitness>Alpha_score && fitness>Beta_score && fitness<Delta_score
            Delta_score=fitness; % Update delta
            Delta_pos=Positions(i,);
        end
    end
end

```

```

end

a=2-1*((2)/Max_iter); % a decreases linearly from 2 to 0

% Update the Position of search agents including omegas
for i=1:size(Positions,1)
    for j=1:size(Positions,2)

        r1=rand(); % r1 is a random number in [0,1]
        r2=rand(); % r2 is a random number in [0,1]

        A1=2*a*r1-a; % Equation (3.3)
        C1=2*r2; % Equation (3.4)

        D_alpha=abs(C1*Alpha_pos(j)-Positions(i,j)); % Equation (3.5)-part 1
        X1=Alpha_pos(j)-A1*D_alpha; % Equation (3.6)-part 1

        r1=rand();
        r2=rand();

        A2=2*a*r1-a; % Equation (3.3)
        C2=2*r2; % Equation (3.4)

        D_beta=abs(C2*Beta_pos(j)-Positions(i,j)); % Equation (3.5)-part 2
        X2=Beta_pos(j)-A2*D_beta; % Equation (3.6)-part 2

        r1=rand();
        r2=rand();

        A3=2*a*r1-a; % Equation (3.3)
        C3=2*r2; % Equation (3.4)

        D_delta=abs(C3*Delta_pos(j)-Positions(i,j)); % Equation (3.5)-part 3
        X3=Delta_pos(j)-A3*D_delta; % Equation (3.5)-part 3

        Positions(i,j)=(X1+X2+X3)/3;% Equation (3.7)

    end
end
l=l+1;
Convergence_curve(l)=Alpha_score;
end

% This function initialize the first population of search agents
function Positions=initialization(SearchAgents_no,dim,ub,lb)
Boundary_no= size(ub,2); % numnber of boundaries
% If the boundaries of all variables are equal and user enter a single
% number for both ub and lb
if Boundary_no==1
    Positions=rand(SearchAgents_no,dim).*(ub-lb)+lb;
end
% If each variable has a different lb and ub
if Boundary_no>1
    for i=1:dim
        ub_i=ub(i);
        lb_i=lb(i);
        Positions(:,i)=rand(SearchAgents_no,1).*(ub_i-lb_i)+lb_i;
    end
end
end

```

Código 3: Optimización losas macizas

Script condiciones de borde

```
% CONDICIONES DE APOYO DE UNA LOSA MACIZA UNIDIRECCIONAL
```

```
function [km, kv, kd] = Tipo_Apoyo_maciza(A)
```

```
switch A
    case 'A1'
        % SIMPLEMENTE APOYADA
        km = 1/8; % Momento máximo
        kv = 1; % Cortante máxima
        kd = 1/20; % Deflexion máxima

    case 'A2'
        % APOYADA - CONTINUA
        km = 1/10;
        kv = 1.15;
        kd = 1/24;

    case 'A3'
        % CONTINUA - CONTINUA
        km = 1/11;
        kv = 1;
        kd = 1/28;

    case 'A4'
        % VOLADIZO
        km = 1/2;
        kv = 2;
        kd = 1/10;

end

end
```

Script de parámetros constantes

```
%%% ESCRIBIR VALOR DE PARAMETROS CONSTANTES
```

```
function [Ln,DLs,LL,fc,fy,db,rec,pb,wg,ws,PUc,PUc,Cc,Cr,km, kv, kd] =
Parametros_maciza
```

```
Name          = 'A3'; % Tipo de apoyos (A1:A4)_APOYADA, APOYADA-CONTINUA,
CONTINUA Y VOLADIZO
Luz           = 3.5;  % en metros, luz libre
CargaMuerta   = 250;  % en kg/m2, carga muerta adicional: acabados,
tabiqueria
CargaViva     = 300;  % en kg/m2, carga viva
fc            = 280;  % en kg/cm2, resistencia a compresion del concreto
fy           = 4200;  % en kg/cm2, resistencia a la fluencia del acero
Recubrimiento = 2;    % en cm, recubrimiento de la losa
db           = 9.5;  % en mm, diámetro del acero mínimo
PesoConcreto  = 2400; % en kg/m3, peso del concreto por unidad de
volumen
PesoAcero     = 7850; % en kg/me, peso del acero por unidad de volumen
PUconcreto    = 420;  % en s/m3, costo del concreto por unndad de
volumen
PUacero       = 4.47; % en s/kg, costo del acero por unidad de peso
CostoVaciado  = 42.7; % en s/m3, costo mano obra por unidad de volumen
CostoAcero    = 4.40; % en s/kg, costo de habilitacion de acero
```

```
[km, kv, kd] = Tipo_Apoyo_maciza(Name);
```

```
Ln=Luz; DLs=CargaMuerta; LL=CargaViva; fc=fc; fy=fy; rec=Recubrimiento;
wg=PesoConcreto;...
ws=PesoAcero; PUc=PUconcreto; PUc=PUacero; Cc=CostoVaciado; Cr=CostoAcero;
%%% Calculamos pb
    if fc<280
        beta1 = 0.85;
    elseif fc<550
        beta1 = 0.85-0.55*(fc-280)/70;
    elseif fc>550
        beta1 = 0.6;
    end
pb = 0.85*beta1*fc/fy*(6000/(600000+fy));

end
```

Script de función objetivo y restricciones

```

%FUNCION OBJETIVO

function Y = Fobj_macizas(X)

    x1 = X(:,1);
    x2 = X(:,2);
    x3 = X(:,3);
%% PARAMETROS CONSTANTES

[Ln,DLs,LL,fc,fy,db,rec,pb,wg,ws,PUc,PUs,Cc,Cr,km, kv, kd] =
Parametros_maciza;
b = 100;
Ld=0.25*Ln;
pb;
%% FUNCION OBJETIVO

fcos =
(Ln*b*x1/10000*(PUc+Cc)+ws*(pi()/ (4*1000.^2) .* (Ld*x3.^2) *b./x2) *(PUs+Cr) ) ./
(0.01*b);

%% RESTTRICCIONES DE DESIGUALDAD
%% Restricciones de FLEXION
g(:,1) =
(km*Ln.^2) *(1.4*(wg*(b.*x1)/10000+DLs*b/100)+1.7*LL*b/100) ./...
(0.9*(pi()/400*(x3.^2) .*b./x2) .*fy.*(x1-rec-1/2-...
0.5*(pi()/400*(x3.^2) .*b./x2) .*fy./(0.85.*fc.*b))*0.01)-
1;

%% Restricciones de CORTANTE
g(:,2) =
0.5*kv*Ln*(1.4*(wg*(b.*x1)/10000+DLs*b/100)+1.7*LL*b/100) ./...
(0.85*0.53.*sqrt(fc) .* (x1-rec-1/2) *b)-1;

%% Restricciones de SERVICIO
g(:,3) = (0.0018*b.*x1) ./ (pi()/400*(x3.^2) *b./x2)-1;
g(:,4) = ((pi()/400*(x3.^2) *b./x2) ./ (b.*x1)) ./ (0.75.*pb)-1;
%p/pmax-1<=0 /pmax = 0.75pb
g(:,5) = x2./min(3.*x1,40)-1;
g(:,6) = max(x3./10,25.4) ./x2-1;

%% Restricciones de deflexion
g(:,7) = (100*kd*Ln) ./x1-1;

%% DEFINE TERMINO DE PENALIZACION
pp = 10^7;
for i=1:size(g,1)
    for j=1:size(g,3)
        if g(i,j)> 0
            penalty(i,j) = pp.*g(i,j); %Lagranje modificado
        else
            penalty(i,j) = 0;
        end
    end
end
end
%% Compute la funcion objetivo

```

```

Y = fcos + sum(penalty,2); % + sum(penalty) for minF(X)
Script de ejecutable optimización losas macizas

%%% REALIZAR OPTIMIZACIONES

%clear all
clc
format compact

%%% PARAMETROS INICIALES
fobj = @Fobj_macizas; %Funcion Objetivo
SearchAgents_no = 50; %Population size
dim = 3; %Variables number
lb = [5 2.54 9.5]; %lower limits
ub = [30 40 25.4]; %upper limits
Max_iteration = 1000; %Max.Iterations

%%% =====GWO: Grey Wolf Optimizer=====

%%% INICIO DE OPTIMIZACION
[Best_score,Best_pos,GWO_cg_curve]=GWO(SearchAgents_no,Max_iteration,lb,ub,
dim,fobj);

%%% GRAFICA DE CONVERGENCIA ALGORITMO
semilogy(GWO_cg_curve, 'LineWidth',2);
xlabel('Iteration Number');
ylabel('Best Cost');
title('Convergence Vs Iteration');
axis tight
grid on
box on
legend('GWO')

%%% IMPRESION DE LA MEJOR SOLUCION Y OPTIMIZACION
display(['La mejor solución obtenida por GWO es : ', num2str(Best_pos)]);
display(['El mejor valor óptimo de la funcion objetivo obtenido por GWO es
: ', num2str(Best_score)]);
x1 = Best_pos(:,1)
x2 = Best_pos(:,2)
x3 = Best_pos(:,3)

```

Código 4: Arquitectura de red neuronal 1

Script red entrenamiento de neuronal 1

```

%% RED NEURONAL 1 ACERO POSITIVO
x = xlsread('Database.xlsx','AsPos','A2:D93');
inputs = x';
%%
t = xlsread('Database.xlsx','AsPos','F2:I93');
targets = t';
%%
neuronas = 8; % Numero neuronas en capa oculta

%%
red1 = fitnet(neuronas,'trainlm'); % Tipo de red neuronal (neuronas,
funcion de entrenamiennto)
red1.trainParam.epochs = (1000); % Numero máximo de épocas
red1.trainParam.max_fail = 150; % Verificación de fallas locales
red1.trainParam.min_grad = 1e-29; % Error máximo permitido
%red.layers{1}.transferFcn = 'logsig'; % Cambiar funcion de activacion de
las neuronas capa 1
%red.layers{2}.transferFcn = 'softmax'; % Cambiar funcion de activacion de
las neuronas capa 2

%%
figure(red1,inputs,targets);
red1.divideFcn = 'dividerand'; % Dividir los datos de manera aleatoria
red1.divideMode = 'sample';
red1.divideParam.trainRatio = 85/100; % % de Datos para entrenamiento
red1.divideParam.valRatio = 7.5/100; % % de Datos para validacion
red1.divideParam.testRatio = 7.5/100; % % de Datos para prueba

%% Entrenamiento
[red1 tr]=train(red1,inputs,targets);

%% Test de la red neuronal
y = red1(inputs);
y = y';

%% Maximo error
Error = abs((t-y));
MaxError = max(Error);
maxError = max(MaxError)

%% Conseguir pesos y bías de la red
wb = formwb(red1,red1.b,red1.iw,red1.lw);
[b,iw,lw] = separatewb(red1,wb);

%% Arquitectura de red neuronal
view(red1)

```

Código 5: Arquitectura de red neuronal 2

Script red entrenamiento de neuronal 2

```

%%RED NEURONAL 2 ACERO NEGATIVO
x = xlsread('Database.xlsx','AsNeg','A2:E119');
inputs = x';
%%
t = xlsread('Database.xlsx','AsNeg','G2:L119');
targets = t';
%%
neuronas = 5;

red2 = fitnet(neuronas,'trainlm'); % Tipo de red neuronal (neuronas,
funcion de entrenamiennto)
red2.trainParam.epochs = (1000); % Numero máximo de épocas
red2.trainParam.max_fail = 150; % Verificación de fallas locales
red2.trainParam.min_grad = 1e-29; % Error máximo permitido

%red.layers{1}.transferFcn = 'purelin'; % Cambiar funcion de activacion de
las neuronas capa 1
%red.layers{2}.transferFcn = 'tansig';% Cambiar funcion de activacion de
las neuronas capa 2

%%
configure(red2,inputs,targets);
red2.divideFcn = 'dividerand'; % Dividir los datos de manera aleatoria
red2.divideMode = 'sample';
red2.divideParam.trainRatio = 85/100; % % de Datos para entrenamiento
red2.divideParam.valRatio = 7.5/100; % % de Datos para validacion
red2.divideParam.testRatio = 7.5/100; % % de Datos para prueba

%% Entrenamiento
[red2 tr]=train(red2,inputs,targets);

%% Test de la red neuronal
y = red2(inputs);
y = y';

%% Maximo error
Error = abs((t-y));
MaxError = max(Error);
maxError = max(MaxError)

%% Conseguir pesos y bías de la red
wb = formwb(red2,red2.b,red2.iw,red2.lw);
[b,iw,lw] = separatewb(red2,wb);

%% Arquitectura de red neuronal
view(red2)

```

Código 6: Diseño de losas aligeradas con redes neuronales

Script de nuevos diseños por red neuronales

```
%DISEÑAR LOSAS ALIGERADAS CON RED NEURONAL
```

```
clc; clear all; close all;
```

```
%% Carga red entrenada
```

```
load('Red1_AsPos.mat');      % Red entrenada 1
load('Red2_AsNeg.mat');      % Red entrenada 2
```

```
%% Ingrese variables de entrada
```

```
L = 21.2; %Longitud de la edificacion
T = 3; %Numero de tramos continuos ..de 2 a 4
A = 4; %Numero de apoyos ... de 3 a 5
Ln1 = 7.1; %Luz de primer tramo..entre 6.00 y 7.50 m
Ln2 = 7.35; %Luz de segundo tramo..
Ln3 = 7.1;
Ln4 = 0;
```

```
%% Activar el vector de entrada
```

```
%%% Para 2 tramos
```

```
%x1 = [L T 1 Ln1; L T 2 Ln2]';
%x2 = [L A 1 0 Ln1; L A 2 Ln1 Ln2;L A 3 Ln3 0]';
```

```
%%% Para 3 tramos
```

```
x1 = [L T 1 Ln1; L T 2 Ln2; L T 3 Ln3]';
x2 = [L A 1 0 Ln1; L A 2 Ln1 Ln2;L A 3 Ln2 Ln3; L A 4 Ln3 0]';
```

```
%%% para 4 tramos
```

```
%x1 = [L T 1 Ln1; L T 2 Ln2; L T 3 Ln3; L 4 4 Ln4]';
%x2 = [L A 1 0 Ln1; L A 2 Ln1 Ln2;L A 3 Ln2 Ln3; L A 4 Ln3 Ln4;L A 4 Ln4
0]';
```

```
%% Obtener vector salida por la red entrenada
```

```
y1 = red1(x1);
y2 = red2(x2);
Tramos = y1'
Apoyos = y2'
```