



UNIVERSIDAD
INTERNACIONAL
SEK
SER MEJORES

UNIVERSIDAD INTERNACIONAL SEK

FACULTAD DE ARQUITECTURA E INGENIERÍA CIVIL

**Trabajo de Titulación Previo a la Obtención del Título de
INGENIERO CIVIL**

**Implementación de un algoritmo genético evolutivo para la modelación paramétrica y
optimización de estructuras aporticadas de acero, mediante un lenguaje de
programación visual**

Brito Caviedes Cristhián Nicolás

Quito, Diciembre de 2024



DECLARACION JURAMENTADA

Yo, BRITO CAVIEDES CRISTHIÁN NICOLÁS, con cédula de identidad número 172455305-0, declaro bajo juramento que el trabajo aquí desarrollado es de mi autoría, que no ha sido previamente presentado para ningún grado a calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo mis derechos de propiedad intelectual que correspondan relacionados a este trabajo, a la UNIVERSIDAD INTERNACIONAL SEK, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

D. M. Quito, Diciembre de 2024

Brito Caviedes Cristhián Nicolás

C.I. 172455305-0

Correo electrónico: ccbrito.civ@uisek.edu.ec



ECUADOR
UNIVERSIDAD
INTERNACIONAL
SEK
SER MEJORES

DECLARATORIA

El presente trabajo de investigación titulado:

“Implementación de un algoritmo genético evolutivo para la modelación paramétrica y optimización de estructuras aporticadas de acero, mediante un lenguaje de programación visual”

Realizado por:

BRITO CAVIEDES CRISTHIÁN NICOLÁS

como Requisito para la obtención del Título de:

INGENIERO CIVIL

ha sido dirigido por el profesor

Ing. Hugo Marcelo Otáñez Gómez, Msc.

quien considera que constituye un trabajo original de su autor

Ing. Hugo Marcelo Otáñez Gómez, Msc.

TUTOR



Implementación de un algoritmo genético evolutivo para la modelación paramétrica y optimización de estructuras aperticadas de acero, mediante un lenguaje de programación visual

Por

Brito Caviedes Cristhián Nicolás

Agosto 2024

Aprobado:

Primer Nombre, Inicial, Primer Apellido, Inicial, Tutor

Primer Nombre, Inicial, Primer Apellido, Inicial, Presidente del Tribunal

Primer Nombre, Inicial, Primer Apellido, Inicial, Miembro del Tribunal

Primer Nombre, Inicial, Primer Apellido, Inicial, Miembro del Tribunal

Aceptado y Firmado: _____ día, mes, año
Primer Nombre, Inicial, Primer Apellido, Inicial.

Aceptado y Firmado: _____ día, mes, año
Primer Nombre, Inicial, Primer Apellido, Inicial.

Aceptado y Firmado: _____ día, mes, año
Primer Nombre, Inicial, Primer Apellido, Inicial.

_____ día, mes, año

Primer Nombre, Inicial, Primer Apellido, Inicial.

Presidente(a) del Tribunal

Universidad Internacional SEK



ECUADOR
UNIVERSIDAD
INTERNACIONAL
SEK
SER MEJORES

Dedicatoria

A Jesús, mi señor y salvador.

A mis padres, Cristian Brito y Sandra Caviedes, quienes me han mostrado su amor
incondicional.

A mis hermanos, Mateo y Kristel.



Agradecimiento

A Dios, por darme la vida, la fuerza y la sabiduría para completar esta etapa tan importante. Sin su gracia, este logro no habría sido posible.

A mis padres, Cristian Brito y Sandra Caviedes, por su amor, paciencia y apoyo incondicional, quienes siempre creyeron en mí, incluso cuando yo dudaba. Gracias por enseñarme el valor del esfuerzo y la dedicación.

A mis profesores; a mi tutor, el Ing. Marcelo Otañez, por su guía académica, sus enseñanzas y su dedicación en este proceso. Su orientación fue clave para el desarrollo de esta tesis.

A mis amigos, compañeros y hermanos, por acompañarme en este camino, por cada conversación, consejo y por ser un pilar de apoyo en los momentos difíciles.

A todos aquellos que, de alguna manera, contribuyeron a que este proyecto sea una realidad, les extiendo mi más sincero agradecimiento.



Este trabajo de tesis fue realizado bajo el Programa de Investigación:

Nombre de Programa

FACULTAD DE ARQUITECTURA E INGENIERÍA CIVIL

Y con el financiamiento de

(colocar el que aplique)

Proyecto de Investigación de la Dirección de Investigación e Innovación

DII-UISEK-PXXXXXX_X.X



Resumen

Al momento de realizar el diseño estructural de una edificación, es necesario tomar decisiones que influirán en el resultado y costo final del diseño. Aunque hay varias alternativas que garantizan que la estructura cumple con las condiciones de servicio y resistencia, algunas son más eficientes que otras.

Un algoritmo de modelación paramétrica y optimización basado en la evolución genética permite encontrar la mejor solución posible de un diseño estructural, en la cual se establecen parámetros fijos y el algoritmo comparará todas las variables posibles, descartando las soluciones menos favorables y evolucionando hacia aquellas más eficientes hasta llegar a la mejor solución, en este caso, la estructura con el menor costo.

La propuesta incluye la implementación de un algoritmo que agilice que la fase de modelación del proyecto, facilitando la modificación de parámetros en el modelo, ahorrando tiempo. Además, el programa permitirá visualizar los resultados más eficientes del modelo y compararlos entre ellos, de esta manera se puede trabajar conjuntamente en la fase de diseño arquitectónico, evitando así reprocesos y escogiendo la opción que más se adecúe a las necesidades del proyecto y los recursos disponibles.

El algoritmo se desarrollará mediante un lenguaje de programación visual en el software "Rhino" y extensiones especializadas como "Grasshopper", "Karamba 3d" y "Galapagos". Su uso se enfoca en la modelación de estructuras aporricadas de acero con geometría rectangular y su aplicación principal es para viviendas de interés social o con presupuestos muy limitados.

Palabras Clave: Algoritmo, modelación, optimización, evolución genética, pórticos.



Abstract

When carrying out the structural design of a building, it is necessary to make decisions that will influence the result and final cost of the design. Although there are several alternatives that guarantee that the structure meets the conditions of service and resistance, some are more efficient than others.

A parametric modeling and optimization algorithm based on genetic evolution allows finding the best possible solution for a structural design, in which fixed parameters will be established and the algorithm will compare all possible variables, discarding the least favorable solutions and evolving towards the most efficient ones until reaching the best solution, in this case, the structure with the lowest cost.

The proposal includes the implementation of an algorithm that speeds up the modeling phase of the project, facilitating the modification of parameters in the model, saving time. In addition, the program will allow you to view the most efficient results of the model and compare them between them, in this way you can work together in the architectural design phase, thus avoiding reprocessing and choosing the option that best suits the needs of the project and resources available.

The algorithm will be developed using a visual programming language in the "Rhino" software and specialized extensions such as "Grasshopper", "Karamba 3D" y "Galapagos". Its use focuses on the modeling of steel porticoed structures with rectangular geometry and its main application is for social housing or housing with very limited budgets.

Keywords: *Algorithm, modeling, optimization, genetic evolution, frames.*

Tabla de Contenidos

| | | |
|-----------|---|------------|
| 1. | Lista de Tablas..... | xv |
| 2. | Lista de Figuras | xvi |
| 1. | CAPÍTULO 1 | 1 |
| 1.1 | Introducción..... | 1 |
| 1.2 | Justificación | 1 |
| 1.3 | Hipótesis | 2 |
| 1.4 | Antecedentes..... | 2 |
| 1.5 | Objetivos..... | 3 |
| | 1.5.1 Objetivo general..... | 3 |
| | 1.5.2 Objetivos específicos | 3 |
| 1.6 | Alcance | 3 |
| 1.7 | Limitaciones | 4 |
| 1.8 | Planteamiento del problema de investigación | 4 |
| 1.9 | Metodología..... | 5 |
| | 1.9.1 Variables dependientes | 6 |
| | 1.9.2 Variables independientes | 6 |
| 2. | CAPÍTULO 2 (FUNDAMENTO TEÓRICO) | 7 |
| 2.1 | Diseño paramétrico..... | 7 |
| | 2.1.1 Objetivos del diseño paramétrico | 7 |
| | 2.1.2 Proceso de diseño paramétrico | 8 |
| | 2.1.3.1 Definición de condiciones iniciales | 9 |
| | 2.1.3.2 Parámetros | 9 |
| | 2.1.3.3 Preparación del procedimiento paramétrico | 10 |

| | | |
|-----------|--|-----------|
| 2.1.3.4 | Ejecución del procedimiento paramétrico | 10 |
| 2.1.3.5 | Selección e interpretación de resultados..... | 11 |
| 2.1.3 | Ventajas del diseño paramétrico | 11 |
| 2.2 | Herramientas de diseño paramétrico | 12 |
| 2.2.1 | Rhino..... | 12 |
| 2.2.1.2 | Características especiales: | 13 |
| 2.2.2 | Grasshopper | 14 |
| 2.2.2.1 | Interface | 15 |
| 2.2.3 | LunchBox..... | 17 |
| 2.3 | Algoritmo | 17 |
| 2.3.1 | Lenguaje de programación visual..... | 18 |
| 2.4 | Análisis estructural | 19 |
| 2.4.1 | Método de elementos finitos..... | 20 |
| 2.4.2 | Herramientas de Análisis estructural | 21 |
| 2.4.2.1 | Karamba 3D..... | 21 |
| 2.5 | Optimización Estructural..... | 22 |
| 2.5.1 | Algoritmos genéticos evolutivos | 24 |
| 2.5.1.1 | Algoritmos evolutivos..... | 24 |
| 2.5.1.2 | Algoritmos genéticos | 24 |
| 2.5.2 | Herramientas de optimización estructural | 25 |
| 2.5.2.1 | Galapagos..... | 25 |
| 2.5.2.2 | Karamba 3D..... | 26 |
| 3. | CAPÍTULO 3 (CASOS DE ESTUDIO) | 27 |

| | | |
|---------|---|----|
| 3.1 | Modelación paramétrica | 31 |
| 3.1.1 | Datos de entrada..... | 31 |
| 3.1.2 | Nodos y líneas..... | 34 |
| 3.1.3 | Vigas Secundarias..... | 37 |
| 3.1.4 | Columnas perimetrales | 40 |
| 3.1.5 | Columnas interiores | 41 |
| 3.1.6 | Identificación de vigas por planta..... | 41 |
| 3.2 | Ensamble del modelo estructural..... | 44 |
| 3.2.1 | Líneas a vigas..... | 45 |
| 3.2.2 | Apoyos | 47 |
| 3.2.3 | Cargas | 48 |
| 3.2.4 | Materiales y secciones transversales..... | 51 |
| 3.2.4.1 | Materiales..... | 51 |
| 3.2.4.2 | Secciones transversales | 52 |
| 3.3 | Análisis estructural | 53 |
| 3.4 | Parámetros de modelación..... | 55 |
| 3.4.1 | Descripción de los datos geométricos..... | 55 |
| 3.4.2 | Materiales..... | 56 |
| 3.4.3 | Análisis de cargas | 56 |
| 3.4.3.1 | Carga muerta..... | 57 |
| 3.4.3.2 | Carga viva | 57 |
| 3.4.3.3 | Sobrecargas | 57 |
| 3.4.4 | Secciones transversales..... | 58 |
| 3.5 | Caso 1 | 59 |

| | | |
|-----------|---|-----------|
| 3.5.1 | Optimización estructural de forma y tamaño con el algoritmo genético | |
| | 59 | |
| 3.5.2 | Resultados de la primera optimización..... | 66 |
| 3.6 | Caso 2..... | 68 |
| 3.6.1 | Optimización estructural de forma y tamaño con el optimizador de Karamba 3D | 68 |
| 3.6.2 | Resultados de la segunda optimización | 71 |
| 3.6.3 | Racionalización del caso 2..... | 73 |
| 3.6.3.1 | Resultados de la racionalización del caso 2..... | 73 |
| 3.7 | Caso 3..... | 75 |
| 3.7.1 | Optimización topológica..... | 75 |
| 3.7.2 | Resultados de la tercera optimización | 76 |
| 3.7.3 | Racionalización del caso 3..... | 79 |
| 3.7.3.1 | Resultados de la racionalización del caso 3..... | 79 |
| 4. | CAPÍTULO 4 ANÁLISIS DE RESULTADOS | 82 |
| 5. | CAPÍTULO 5 (CONCLUSIONES Y RECOMENDACIONES) | 86 |
| 6. | REFERENCIAS | 89 |
| 7. | ANEXOS | 91 |
| 7.1 | Anexo 1. Catálogo de secciones transversales disponibles para columnas del caso de estudio..... | 91 |
| 7.2 | Anexo 2. Catálogo de secciones transversales disponibles para vigas principales y secundarias del caso de estudio | 92 |
| 7.3 | Anexo 3. Código en Visual Basic para limitar el desplazamiento máximo del modelo93 | |

| | | |
|------|---|-----|
| 7.4 | Anexo 4. Código en Visual Basic para limitar los radios de utilización..... | 94 |
| 7.5 | Anexo 5. Código en Visual Basic para convertir a 10000 todos los valores de la lista si algún radio de optimización excede el máximo aceptable..... | 95 |
| 7.6 | Anexo 6. Cálculo de la masa de las columnas de la estructura del caso 2 racionalizado..... | 96 |
| 7.7 | Anexo 7. Cálculo de la masa de las vigas principales de la estructura del caso 2 racionalizado..... | 97 |
| 7.8 | Anexo 8. Cálculo de la masa de las vigas secundarias de la estructura del caso 2 racionalizado..... | 98 |
| 7.9 | Anexo 9. Cálculo de la masa de las columnas de la estructura del caso 3 racionalizado..... | 99 |
| 7.10 | Anexo 10. Cálculo de la masa de las vigas principales de la estructura del caso 3 racionalizado..... | 100 |
| 7.11 | Anexo 11. Cálculo de la masa de las vigas secundarias de la estructura del caso 3 racionalizado..... | 101 |

1. Lista de Tablas

| | |
|--|----|
| Tabla 1 Taxonomía de parámetros. | 10 |
| Tabla 2 Posición de los ejes de la estructura | 56 |
| Tabla 3 Tabla resumen de cargas aplicadas | 58 |
| Tabla 4 Masa de la estructura con la segunda optimización racionalizada | 75 |
| Tabla 5 Posiciones de los ejes de la estructura resultantes de la optimización 3 | 77 |
| Tabla 6 Masa de la estructura con la segunda optimización racionalizada | 81 |
| Tabla 7 Tabla comparativa de las optimizaciones realizadas..... | 85 |

2. Lista de Figuras

| | |
|---|----|
| Figura 1 Proceso general de actividades en el diseño paramétrico | 8 |
| Figura 2 Casos reales de estructuras paramétricas. | 12 |
| Figura 3 Logo de Rhinoceros. | 12 |
| Figura 4 Logo de Grasshopper. | 14 |
| Figura 5 Ventana Principal de Grasshopper. | 15 |
| Figura 6 Logo de LunchBox..... | 17 |
| Figura 7 Ejemplo de una estructura..... | 19 |
| Figura 8 Logo de Karamba 3D..... | 21 |
| Figura 9 Optimización por tamaño, por forma y topológica respectivamente. | 23 |
| Figura 10 Algoritmo genético | 25 |
| Figura 11 Logo de Galapagos..... | 25 |
| Figura 12 Esquema de procesos del algoritmo de modelación paramétrica y análisis estructural | 29 |
| Figura 13 Esquema de procesos del algoritmo de modelación estructural | 30 |
| Figura 14 Sliders de entrada de datos geométricos | 31 |
| Figura 15 Herramienta "Gene List" para ingreso de posiciones de ejes | 32 |
| Figura 16 Componente "Value List" para seleccionar la dirección de las vigas secundarias | 32 |
| Figura 17 Bloque de todos los parámetros de entrada..... | 33 |
| Figura 18 Bloque para generar nodos y líneas | 34 |
| Figura 19 Algoritmo para mantener ejes en el perímetro de la estructura | 34 |
| Figura 20 Creación de la malla guía de la estructura | 35 |
| Figura 21 Comando "Move" para replicar las vigas en las demás plantas | 35 |
| Figura 22 Replicar los nodos y vigas en las demás plantas..... | 36 |

| | |
|--|----|
| Figura 23 Modelo geométrico de la estructura con vigas y columnas | 36 |
| Figura 24 Bloque para generar vigas secundarias | 37 |
| Figura 25 División de las vigas principales para ubicar las vigas secundarias | 38 |
| Figura 26 Nodos creados al dividir la viga principal | 38 |
| Figura 27 Eliminación de los nodos superpuestos con los nodos de las vigas principales | 39 |
| Figura 28 Modelo de la geometría incluyendo las vigas secundarias | 40 |
| Figura 29 Generar las vigas secundarias en las demás plantas | 40 |
| Figura 30 Bloque para identificar las columnas perimetrales | 40 |
| Figura 31 Bloque para definir las columnas interiores..... | 41 |
| Figura 32 Bloque para identificar las vigas por planta..... | 42 |
| Figura 33 Identificar vigas en la primera planta..... | 43 |
| Figura 34 Identificar vigas en las plantas intermedias | 43 |
| Figura 35 Identificar vigas en la cubierta | 44 |
| Figura 36 Componente para ensamblar el modelo estructural | 44 |
| Figura 37 Bloque para crear elementos estructurales a partir de líneas | 45 |
| Figura 38 Componente "LineToBeam" | 46 |
| Figura 39 Algoritmo para dividir las vigas principales | 47 |
| Figura 40 Componente Support para definir apoyos..... | 47 |
| Figura 41 Bloque para definir las cargas | 48 |
| Figura 42 Componentes para generar una carga | 49 |
| Figura 44 Algoritmo para definir cargas en la primera planta | 50 |
| Figura 43 Superficie donde se aplica la sobrecarga | 50 |
| Figura 45 Bloque para definir los materiales y las secciones transversales | 51 |
| Figura 46 Componentes para definir los materiales | 51 |

| | |
|---|----|
| Figura 47 Componente "Cross Section Material Selection" para definir las secciones transversales | 52 |
| Figura 48 Componente "Cross Section Selector" para seleccionar una sección transversal..... | 53 |
| Figura 49 Bloque para realizar el análisis estructural del modelo..... | 53 |
| Figura 50 Vista general del algoritmo completo | 55 |
| Figura 51 Visualización de la geometría de la estructura..... | 56 |
| Figura 52 Disposición de las sobrecargas en las plantas de la estructura | 58 |
| Figura 53 Masa y deflexión máxima obtenidas de la estructura sin optimizar | 59 |
| Figura 54 Estructura inicial sin optimizar | 59 |
| Figura 55 Componente "Galapagos" | 60 |
| Figura 56 "Number Sliders" conectados al parámetro "Genome" | 60 |
| Figura 57 Componente "Evaluate" para crear la función objetivo..... | 61 |
| Figura 58 Componente "VB Script" con el código para filtrar el desplazamiento máximo..... | 62 |
| Figura 59 Componente "Utilization" y componente VB "Script" para filtrar los radios de optimización | 63 |
| Figura 60 Componente "VB Script" para convertir todos los datos a 10000..... | 63 |
| Figura 61 Bloque para realizar la optimización estructural con el algoritmo genético "Galapagos"..... | 64 |
| Figura 62 Ventana para configurar las opciones de Galapagos | 64 |
| Figura 63 Ventana del solucionador de Galapagos para la primera optimización | 67 |
| Figura 64 Estructura con la primera optimización | 68 |
| Figura 65 Masa y desplazamiento máximo obtenidos de la estructura con la primera optimización | 68 |

| | |
|---|----|
| Figura 66 Componente “Optimize Cross Section” | 70 |
| Figura 67 Bloque para la optimización en Karamba 3D | 71 |
| Figura 68 Ejemplo de elementos optimizados con Karamba3D | 72 |
| Figura 69 Estructura con la segunda optimización..... | 72 |
| Figura 70 Masa y desplazamiento máximo obtenidos de la estructura con la segunda optimización | 73 |
| Figura 71 Secciones transversales establecidas para el caso 2 racionalizado | 74 |
| Figura 72 Parámetros conectados al componente "Galapagos" | 76 |
| Figura 73 Ventana del solucionador de Galapagos para la tercera optimización..... | 76 |
| Figura 74 Lista de la posición de los ejes con la optimización de "Karamba 3D" | 77 |
| Figura 75 Esquema en planta de la posición de los ejes con la optimización topológica | 78 |
| Figura 76 Estructura con la tercera optimización..... | 78 |
| Figura 77 Masa y desplazamiento máximo obtenidos de la estructura con la tercera optimización | 79 |
| Figura 78 Secciones transversales establecidas para el caso 3 racionalizado | 80 |

1. CAPÍTULO 1

1.1 Introducción

En la era actual de la ingeniería y el diseño arquitectónico, la modelación paramétrica es una herramienta esencial para abordar la complejidad y diversidad de los problemas estructurales. El proceso de generar y evaluar múltiples alternativas de diseño que cumplen con ciertos parámetros establecidos de antemano permite a los ingenieros encontrar soluciones de forma precisa y efectiva.

Este enfoque combina la flexibilidad de la modelación paramétrica con la robustez de los algoritmos evolutivos, que modelan el proceso de selección natural para encontrar soluciones óptimas a problemas complejos. Los algoritmos genéticos aplican los principios de variación y selección para mejorar iterativamente un conjunto de soluciones candidatas; por lo tanto, permiten encontrar una configuración estructural óptima de manera eficiente.

El objetivo de la tesis es explorar la modelación paramétrica y la optimización estructural basada en algoritmos genéticos y demostrar su impacto en la eficiencia estructural y el consiguiente ahorro de costos. Se espera contribuir al desarrollo de metodologías de diseño y optimización avanzadas que respondan a las exigencias actuales dentro de la ingeniería estructural.

1.2 Justificación

Dentro de la industria de la construcción y la ingeniería civil, se conocen diversas técnicas para el diseño estructural de edificaciones. Sin embargo, los métodos tradicionales carecen de eficiencia y precisión en términos de diseño, lo que incrementa los costos finales de la estructura.

A pesar de los avances tecnológicos en los últimos años, que permiten realizar procesos cada vez más complejos en las computadoras, aún existen vacíos en cuanto a la aplicación de

algoritmos de modelación y optimización en estructuras. En nuestro país, principalmente, no se ha apostado por el uso de estas herramientas y una de las razones es que no se ha estudiado a fondo estos temas.

El proyecto propuesto busca contribuir al estudio de implementación de algoritmos de modelación paramétrica y optimización estructural. Se proporcionará un algoritmo específico para estructuras de acero aporticadas que permita modelar la estructura de forma paramétrica y optimizar las secciones transversales de los elementos y la ubicación de columnas. Esto resultará innovador y, sobre todo, útil para que los diseñadores optimicen el proceso de diseño y finalmente, los recursos para la construcción de la estructura.

1.3 Hipótesis

La implementación de un algoritmo de modelación paramétrica y optimización mejora la eficiencia del proceso de diseño de estructuras aporticadas de acero y reduce los costos de su construcción al optimizar el peso de la estructura.

1.4 Antecedentes

Hasta la segunda mitad del siglo XX, los métodos de optimización y modelación que existían, consistían en procedimientos exactos aplicados específicamente para cada problema con una limitada utilidad a sectores específicos.

Con el desarrollo tecnológico, y el avance en la computación en las últimas décadas, se han establecido nuevas técnicas llamadas metaheurísticas, para obtener soluciones de manera más eficiente y menos limitada en su aplicación. Dentro de estas técnicas se encuentran los algoritmos de modelación paramétrica y algoritmos genéticos evolutivos para optimizar una solución. Estas técnicas presentan una notable facilidad en su aplicación. (Casado Bravo, 2022)

Actualmente, estos algoritmos son cada vez más utilizados por profesionales en el campo de la ingeniería civil, sin embargo, el tema presenta limitados estudios y aplicaciones que comparen y analicen la eficiencia de los diferentes tipos de algoritmos y sus aplicaciones dentro del campo de la ingeniería civil.

1.5 Objetivos

1.5.1 Objetivo general

Desarrollar un algoritmo para la modelación paramétrica y optimización de estructuras aporticadas de acero mediante un lenguaje de programación visual.

1.5.2 Objetivos específicos

- Implementar un algoritmo de modelación paramétrica desarrollado en “Rhino” y “Grasshopper” para facilitar procesos.
- Realizar el análisis estructural dentro del algoritmo mediante el uso del software complementario “Karamba 3D” para conocer el comportamiento de la estructura.
- Generar la optimización de la estructura, incluyendo la posición de las columnas dentro del algoritmo con el fin de encontrar el modelo más eficiente.
- Validar la eficiencia y rendimiento del algoritmo mediante pruebas en casos de estudio.

1.6 Alcance

Elaboración de un algoritmo de modelación paramétrica y optimización para estructuras aporticadas de acero dentro del software “Rhino” y “Grasshopper” que cumpla con lo siguiente:

- Reducir el tiempo del diseño estructural: El algoritmo permite agilizar la modelación facilitando la entrada y modificación de parámetros.

- Disminución de costos totales de la estructura: El algoritmo selecciona la opción más eficiente combinando todas las variables posibles.

1.7 Limitaciones

El resultado es un algoritmo pensado principalmente para estructuras aporricadas, de acero, cuya geometría sea ortogonal, con plantas de geometría regular. Se excluye del análisis los elementos de conexiones, cimentaciones y detalles constructivos. Únicamente se realizará un análisis estructural de elementos finitos para una superestructura; columnas y vigas a partir de cargas y esfuerzos dados.

1.8 Planteamiento del problema de investigación

La modelación paramétrica y la optimación estructural son aspectos fundamentales a la hora de diseñar una estructura, ya que precisamente lo que se busca es encontrar de manera rápida la solución más óptima. Los métodos tradicionales de análisis estructural empleados actualmente en nuestro entorno no son los más eficientes, principalmente porque no se evalúan todas las soluciones posibles para la estructura.

Por una parte, durante el proceso de diseño suelen surgir cambios en los parámetros de la estructura, como la geometría de los espacios o los materiales. Este tipo de modificaciones son laboriosas y aumentan el tiempo del proceso de diseño ya que no siempre es rápido realizar modificaciones en los softwares de diseño y análisis estructural.

Por otra parte, existen variables que no se toman en cuenta al diseñar una estructura, como la ubicación y cantidad de columnas. Esto se debe principalmente a que estas variables están determinadas por la arquitectura. Sin embargo, es posible tomar en cuenta estos parámetros incluso antes del diseño arquitectónico. Al trabajar conjuntamente en el diseño estructural y arquitectónico se pueden encontrar soluciones más eficientes.

Dadas estas consideraciones, resulta pertinente implementar un método de diseño estructural más eficiente, como un algoritmo de modelación paramétrica y optimización estructural. Un diseño estructural óptimo resulta en ahorro de recursos, evitar reprocesos, pérdida de tiempo y reducción de costos totales en la estructura.

1.9 Metodología

El tipo de proyecto a realizarse se enmarca dentro del desarrollo tecnológico, utilizando una investigación exploratoria para profundizar el tema de algoritmos de modelación paramétrica y optimización estructural.

La investigación adoptará una modalidad centrada en la obtención de resultados tangibles para abordar problemas y necesidades específicas, empleando un enfoque teórico de modelación.

El procedimiento implica el desarrollo del algoritmo en el software “Rhino” y sus extensiones especializadas como: “Grasshopper” para la gestión de datos en un ambiente de diseño paramétrico, “Karamba 3D” para el análisis estructural, y “Galapagos” para la optimización.

El proyecto se realizará en un período a aproximado de 8 meses y se delimitará el estudio a estructuras aporticadas de acero con geometría rectangular y ortogonal. Dentro del algoritmo se establecerán las restricciones como las deformaciones máximas de los elementos y el cumplimiento de la resistencia y radios de optimización. El algoritmo permitirá el ingreso de datos como: Las dimensiones del área del proyecto, la cantidad de plantas y su altura, el número y posición de ejes, el material de los elementos y las cargas actuantes. Mediante el algoritmo de optimización, el programa encontrará la solución más eficiente, es decir, la que tenga el menor peso, seleccionando las variables óptimas, en este caso: Las secciones de los

elementos y la posición de las columnas. En este sentido se puede delimitar las variables:

1.9.1 Variables dependientes

Secciones de elementos, posición de las columnas, peso de la estructura, deflexión y radios de optimización.

1.9.2 Variables independientes

Restricciones de servicio y resistencia, sistema estructural, materiales, dimensiones del área del proyecto, cantidad y altura de pisos.

2. CAPÍTULO 2 (FUNDAMENTO TEÓRICO)

2.1 Diseño paramétrico

El diseño paramétrico implica la conceptualización de una idea mediante la utilización de principios geométricos y matemáticos. Este enfoque nos posibilita ajustar nuestro diseño de manera precisa, facilitando la obtención de resultados óptimos. (Molinare, 2011).

2.1.1 Objetivos del diseño paramétrico

- Elaborar un proceso en lugar de un resultado específico: En vez de centrarse en un único resultado, se concibe un proceso que desarrolla relaciones geométricas y matemáticas, creando sistemas y métodos (algoritmos) que permiten explorar múltiples resultados con ciertas premisas de diseño preestablecidas.
- Capacidad para interrelacionar variables o parámetros: Al centrarse en un proceso de diseño en lugar de una forma fija, se permite la manipulación en tiempo real de sus variables y propiedades, facilitando la comparación de diferentes resultados. Esto conduce a un producto final más eficiente.
- Resultados paramétricos y/o adaptativos según condiciones previamente establecidas: Mediante el diseño paramétrico, es posible crear diseños que sean inteligentes y adaptativos basándose en un conjunto de criterios de diseño (exploración de formas). Esto permite que el diseño se adapte a diversas situaciones, contextos, o condicionantes tectónicos, integrando parámetros o variables en el proceso de diseño para generar soluciones específicas y receptivas a problemas concretos. (Navarrete, 2014).

2.1.2 Proceso de diseño paramétrico

En el diseño paramétrico, el proceso comienza con la definición de las relaciones entre los distintos elementos. A partir de estas relaciones, se construye el diseño, el cual puede ser modificado evaluando y seleccionando los resultados obtenidos. Este enfoque permite explorar diferentes variantes sin necesidad de rehacer completamente el trabajo de representación cada vez. Este método requiere un cambio en los métodos de trabajo del diseñador, quien debe incluir la definición de las relaciones y la lógica que mantiene la coherencia del diseño como una fase esencial en su proceso creativo. En un modelo paramétrico, cada elemento tiene parámetros que controlan sus propiedades geométricas, como longitud, anchura, altura, radio, entre otras. Además, estos parámetros determinan la posición de los elementos en el modelo y cómo se relacionan entre ellos. El operador puede ajustar estos parámetros para generar la geometría deseada (Navarrete, 2014)

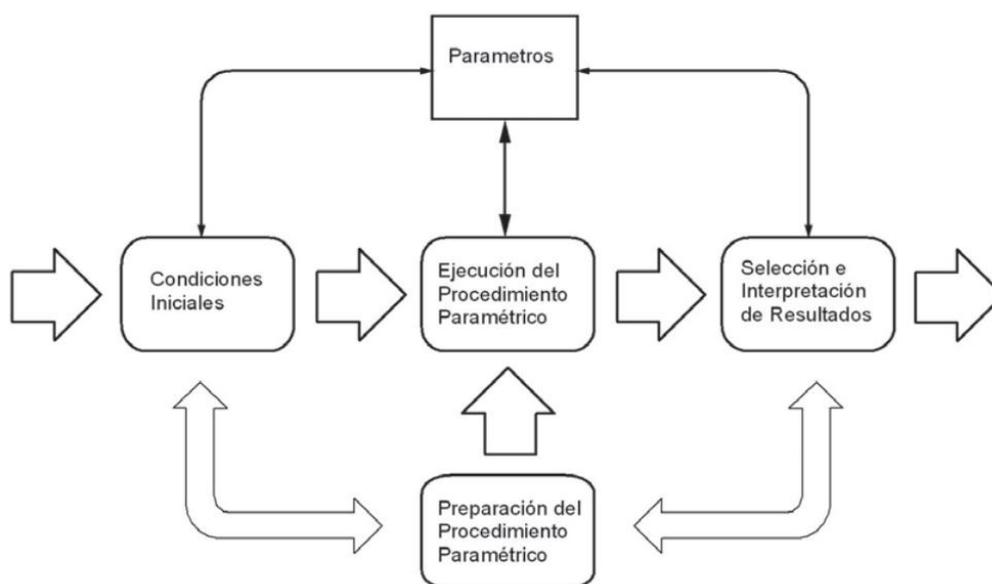


Figura 1 Proceso general de actividades en el diseño paramétrico

Tomado de: Diseño paramétrico en Arquitectura, por García, R, 2013, ResearchGate

2.1.3.1 Definición de condiciones iniciales

En el diseño paramétrico, las condiciones iniciales son factores del proyecto o situación que se consideran durante el proceso. Estos incluyen fórmulas para generar formas o figuras base, incluso de fuentes externas. Algunas condiciones se identifican tras obtener resultados, descartando opciones no viables. Pueden abarcar límites conceptuales, de desempeño, y características formales como curvaturas o extensiones, expresadas en valores numéricos o seleccionadas intuitivamente. Estas condiciones deben formar un sistema de relaciones geométricas que se puede definir en una plataforma de modelado o programación gráfica (García Alvarado & Lyon Gottlieb, 2013).

2.1.3.2 Parámetros

Los parámetros sirven para definir intervalos, límites y configuraciones concretas. Dependiendo de cómo se ajusten estos parámetros, un mismo modelo paramétrico puede generar resultados distintos. La diversidad de parámetros puede clasificarse en una taxonomía que abarca diferentes escalas como se puede observar en la tabla 1. Estos parámetros pueden incluir magnitudes geométricas generales o parciales, propiedades materiales que pueden ser independientes de la forma o relacionadas con las dimensiones, así como relaciones formales. Estos parámetros pueden estar definidos de manera explícita o implícita, pero algunos se aplican con rangos numéricos específicos (García Alvarado & Lyon Gottlieb, 2013).

Tabla 1.

| Parámetros Ambientales (PA) | Parámetros Globales (PG) | Parámetros Locales (PL) | Parámetros de Ejecución (PE) |
|--|--|---|---|
| <i>Datos geográficos:</i> topografía, vistas, tipos de suelo, etc. | <i>Dimensiones o proporciones generales:</i> rangos mínimos y máximos para largo, ancho, profundidad, curvatura, etcétera. | <i>Dimensiones o proporciones de componentes:</i> rangos mínimos y máximos para largo, ancho, profundidad, cantidad, etc. | <i>Dimensiones de producción:</i> tamaño de materiales y máquinas de ejecución. |
| <i>Datos climáticos:</i> orientación, temperatura, humedad, radiación, vientos, etc. | <i>Requerimientos funcionales:</i> prestaciones de confort, ergonomía, accesibilidad. | <i>Interacción con otros componentes:</i> condiciones de borde y respuesta a configuraciones adyacentes. | <i>Propiedades materiales:</i> rangos de resistencia o flexión. |
| <i>Situación contextual:</i> restricciones normativas de situación urbana, materialidad, tipología (aislado, pareado, torre, placa, etc.). | <i>Distribución global:</i> relaciones y topología interna. | <i>Respuesta a valores de análisis:</i> profundidad o espesor de las piezas según asoleamiento o solicitudes estructurales. | <i>Características del producto:</i> color, textura, terminación, etc. |
| <i>Relaciones del entorno:</i> flujos peatonales, vehiculares, presencia de singularidades, referencias, etc. | <i>Condiciones expresivas:</i> Configuración de fachadas y materialidad. | <i>Condiciones formales:</i> Variación gradual entre componentes. | <i>Valores de aplicación:</i> costos del proyecto. |
| <i>Dimensiones del sitio:</i> Ancho y profundidad del lote, pendiente, límites de edificación. | <i>Restricciones técnicas:</i> crujiás y voladizos según sistema estructural. | <i>Requerimientos de montaje:</i> tipos de ensamblaje, unión y dilatación entre componentes. | <i>Dimensiones para transporte:</i> magnitudes de vehículos y operación. |

Tabla 1 Taxonomía de parámetros.

Tomado de Taxonomía de parámetros, de García. R & Lyon. A, 2013, Diseño paramétrico en Arquitectura; método, técnicas y aplicaciones.

2.1.3.3 Preparación del procedimiento paramétrico

La preparación del procedimiento paramétrico se refiere a la configuración del algoritmo, es una acción secuencial en la cual, a partir de datos numéricos y operaciones matemáticas se genera una geometría. El algoritmo puede incluir análisis de elementos finitos, algoritmos de optimización, etc.

Estos procedimientos se realizan usualmente en softwares de programación visual como “Grasshopper” incluido en “Rhino”, además se suelen usar extensiones especializadas para cada caso. En resumen, la preparación consiste en elaborar un algoritmo en un programa definiendo antecedentes y acciones de ejecución y comprobación.

2.1.3.4 Ejecución del procedimiento paramétrico

La ejecución del procedimiento paramétrico consiste en ejecutar el algoritmo previamente configurado, en esta etapa se definen los datos específicos del caso. Se ejecuta el

programa las veces necesarias modificando y ajustando valores para poder obtener variedad de resultados.

2.1.3.5 Selección e interpretación de resultados.

Finalmente se obtiene como resultado una forma, y los datos de esta. En el caso de una estructura, se obtiene el modelo y los resultados de análisis estructural. Pocas veces la solución del algoritmo se tomará como diseño final, es necesario contrastar el diseño con aspectos como la factibilidad constructiva y de servicialidad arquitectónica. Es posible descartar soluciones o realizar modificaciones

Es importante también comprobar los resultados del análisis estructural con otro programa especializado de elementos finitos. Y el archivo geométrico puede ser transportado a un programa de diseño para completar trazados faltantes o detalles constructivos.

2.1.3 Ventajas del diseño paramétrico

- Mayor libertad: Posibilidad de crear formas irregulares que resultan llamativas, asemejándose a figuras encontradas en la naturaleza, lo cual sería complicado de realizar con métodos tradicionales.
- Eficiencia: Automatiza el proceso de diseño, evitando reprocesos y ahorrando recursos.
- Flexibilidad: Facilita modificaciones por cambios en los requisitos y condiciones. (Seguí, 2023).

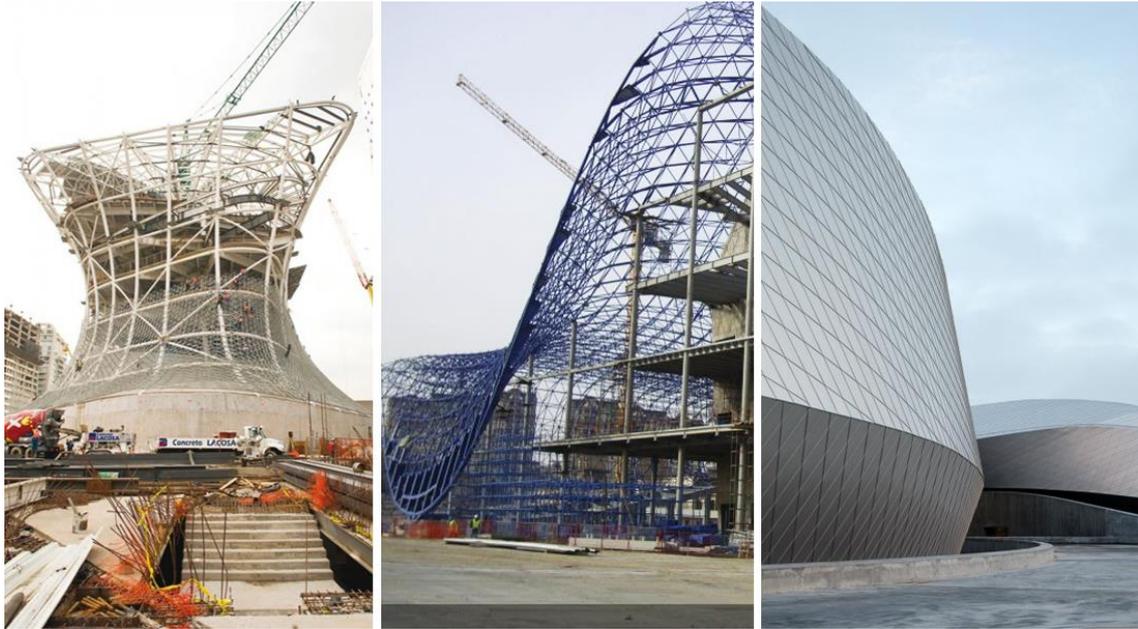


Figura 2 Casos reales de estructuras paramétricas.

Tomado de: En Construcción: Diseño paramétrico, 2014, ArchDaily

2.2 Herramientas de diseño paramétrico

2.2.1 Rhino



Figura 3 Logo de Rhinoceros.

Adaptado de Rhinoceros, McNeel, R, 2024, McNeel & Associates.

Rhino es un software avanzado para modelado tridimensional, basado en la tecnología NURBS, ampliamente utilizado en ingeniería, arquitectura, diseño industrial, naval, de productos, y automotriz. Desarrollado por Robert McNeel & Associates, Rhino evolucionó de ser una extensión de AutoCAD a convertirse en una herramienta independiente para modelado

3D. Este programa se distingue por su facilidad de uso, su costo accesible y su capacidad para generar diseños visualmente impactantes.

Este software de modelado 3D es robusto y versátil, permitiendo a los usuarios crear, modificar, analizar, documentar, renderizar, traducir y animar no solo curvas NURBS, sino también superficies, sólidos, nubes de puntos, geometrías de subdivisión, y mallas poligonales. (Gonzales, 2021)

Las NURBS son modelos matemáticos de geometría tridimensional que pueden representar con exactitud una amplia variedad de formas, desde figuras básicas como líneas, círculos, arcos y curvas bidimensionales, hasta sólidos complejos y superficies orgánicas tridimensionales de formas libres. Por su versatilidad y precisión, los modelos NURBS son aplicables en una variedad de procesos, abarcando desde la ilustración y la animación hasta la fabricación. (McNeel, 2024)

2.2.1.2 Características especiales:

- Herramientas ilimitadas para modelado 3D con formas libres: Ofrece un extenso conjunto de herramientas para crear modelos 3D de cualquier forma.
- Precisión exacta para diseño y fabricación: Proporciona la exactitud necesaria para procesos de diseño, creación de prototipos, ingeniería, análisis y producción de cualquier producto.
- Compatibilidad amplia: Funciona con la mayoría de los programas de diseño asistido por computadora (CAD), dibujo, CAM, ingeniería, creación de prototipos, análisis, renderizado, animación e ilustración.
- Capacidad para leer y reparar mallas complejas y archivos IGES: Puede manejar y corregir archivos complicados de mallas y formatos IGES.

- Accesible y fácil de aprender: Diseñado para ser intuitivo y de fácil uso, lo que facilita su aprendizaje y utilización.
- Alta velocidad sin requerimientos especiales de hardware: Opera rápidamente y no necesita equipos especiales para funcionar.
- Compatible con múltiples plataformas: Es un modelador 3D extremadamente versátil, disponible tanto para Windows como para macOS. (McNeel, 2024)

2.2.2 Grasshopper



Figura 4 Logo de Grasshopper.

Tomado de Grasshopper3d, Davidson, S, 2024, Grasshopper

Grasshopper es un complemento de programación visual que se integra en el software de diseño asistido por computadora (CAD) Rhinoceros 3D. Las herramientas de programación visual, como Grasshopper, permiten construir algoritmos mediante una interfaz intuitiva basada en "arrastrar y soltar" componentes en el área de trabajo. Al interconectar estos componentes, se forma una secuencia de instrucciones destinadas a realizar acciones específicas para obtener resultados determinados.

Principalmente, Grasshopper se utiliza como una herramienta para gestionar datos en entornos de diseño paramétrico. En esencia, esto implica aplicar lógica algorítmica al diseño, estructurando el proceso en una serie de pasos coordinados que guían la evolución de formas geométricas y las interrelaciones entre diversos parámetros.

Esta metodología abre oportunidades para explorar múltiples alternativas de diseño, favoreciendo la búsqueda de soluciones inteligentes que optimicen y automaticen los procesos. Además, facilita la incorporación de datos numéricos provenientes de distintas fuentes, utilizándolos como factores clave en el desarrollo del proyecto. (Echeverri Montes, 2021)

2.2.2.1 Interface

El diálogo principal:

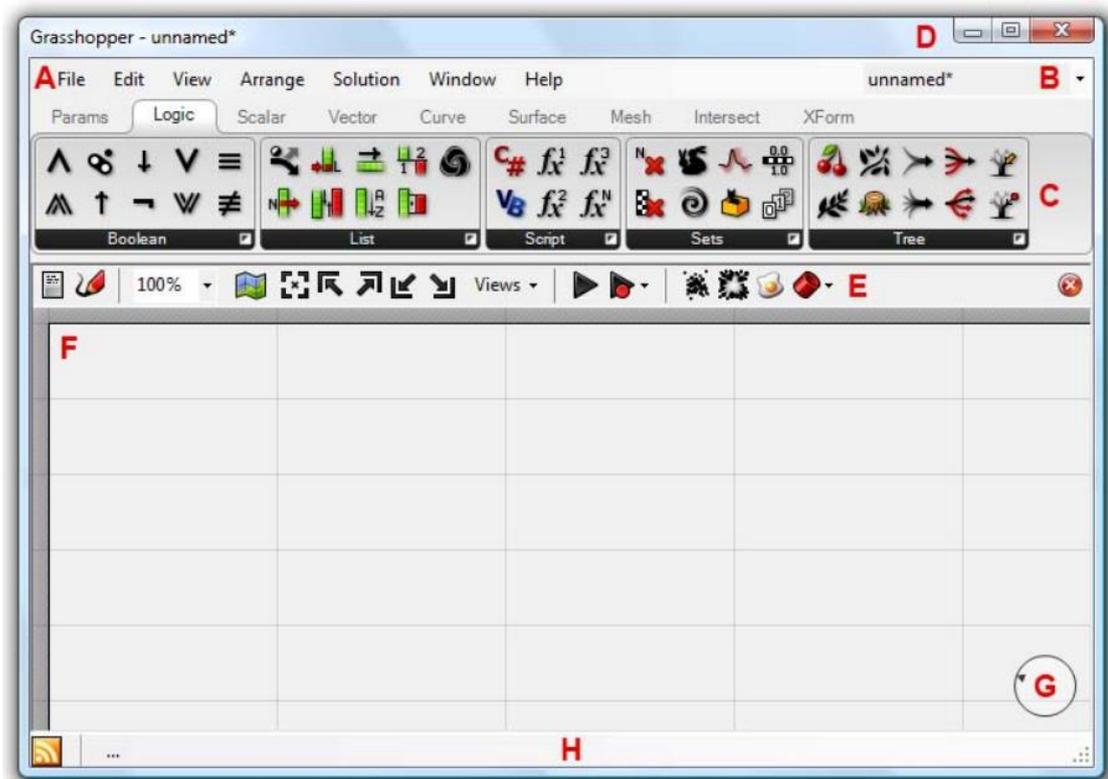


Figura 5 Ventana Principal de Grasshopper.

Tomado de Grasshopper Primer, Payne, A & Issa, Brajaa

A: La barra de menú principal: Incluye opciones generales de la interface de Grasshopper.

B: Cuadro desplegable de explorador de archivos: Se puede cambiar entre distintos documentos cargados.

C: Panel de comandos: En esta sección se muestra todas las categorías de los comandos. A cada uno de los comandos le corresponde una categoría específica como parámetros, curvas, y extensiones instaladas. Para ver toda la colección de comandos de una categoría específica se debe hacer clic en la parte inferior del panel. Para insertar un determinado componente es necesario arrastrarlo hasta el lienzo de trabajo o digitando el nombre del comando dando doble clic sobre el lienzo.

D: Barra de título: Si se hace doble clic sobre la barra de título, se minimizará o maximizará la ventana de Grasshopper a la parte derecha de la pantalla. De esta manera se facilita el cambio entre Rhino y Grasshopper para el usuario.

E: Barra de herramientas de lienzo: Permite el acceso a funciones de uso frecuente para manejar de mejor manera el lienzo de trabajo.

F: El lienzo de trabajo: Es el espacio en el cual se alberga y edita los diferentes componentes. Los objetos dentro del lienzo utilizan un código de color para indicar información sobre su estado.

G: Dispositivos de interfaz de usuario: Se dispone una brújula ubicada en la esquina inferior derecha, la brújula muestra la ubicación de los elementos que se encuentran en el lienzo en relación a la vista actual.

H: La barra de estado: Provee información acerca de errores o advertencias sobre un conjunto seleccionado.

2.2.3 LunchBox



Figura 6 Logo de LunchBox.

Tomado de Lunchbox Food4Rhino, Miller, N. 2024, Food4Rhino.

LunchBox es un complemento de Grasshopper para explorar formas matemáticas, estructuras y gestión de datos. Contiene componentes que permiten generar geometrías sofisticadas como paneles, y redes de puntos que conforman cerchas cuadrangulares, diamantadas y triangulares de manera paramétrica (McNeel, 2024).

LunchBox incluye LunchBoxML, un conjunto de herramientas de aprendizaje automático generalizado para implementar modelos que incluyen regresión, redes neuronales, clasificadores probabilísticos y agrupación (McNeel, 2024).

2.3 Algoritmo

Un algoritmo es una secuencia explícita, precisa, inequívoca y ejecutable mecánicamente de instrucciones elementales, generalmente destinadas a lograr un objetivo específico (Erickson, 2019).

Informalmente, un algoritmo es un procedimiento computacional definido que toma un valor o un conjunto de valores como entrada y genera un valor o conjunto de valores como

salida en un período de tiempo. Un algoritmo es, entonces, una secuencia de pasos que transforman una entrada en salida para resolver un problema específico.

Un algoritmo es correcto sí, para cada para cada instancia del problema proporcionada como entrada, detiene su cálculo y genera la solución correcta. Un algoritmo incorrecto puede que no se detenga en algunas instancias de entrada o que se detenga con una respuesta incorrecta. Los algoritmos incorrectos pueden ser útiles si se controla su tasa de error (Cormet, Leiserson, Rivest, & Stein, 2022).

2.3.1 Lenguaje de programación visual

En la programación visual, los componentes del lenguaje de programación se presentan como bloques diseñados gráficamente, lo que le ha valido el nombre de programación gráfica. Estos bloques tienen una apariencia y etiquetas que indican claramente qué función pueden desempeñar en el flujo del programa. Este tipo de programación no requiere estructuras muy complejas.

Este enfoque básico de programación tiene las siguientes características:

- Visual: Los elementos se integran en el flujo del programa mediante arrastrar, soltar y conectar.
- Orientado a eventos: Cada paso del programa se activa cuando ocurre un evento definido previamente.
- Imperativo: La programación se desarrolla como una secuencia de comandos.
- Orientado a objetos: Se trabaja con objetos individuales a los que se asignan funciones específicas.

Se utilizan bloques predefinidos del lenguaje de programación visual que emplean. Estos bloques pueden manejar información, que se organiza de manera lógica y modificar su apariencia o significado. Los bloques de programación están diseñados para encajar entre sí como piezas de un rompecabezas. Si los elementos no se conectan lógicamente, el editor lo

detecta y lo señala con colores. Los bloques suelen estar agrupados por colores, lo que ayuda a los usuarios a seleccionarlos correctamente (IONOS, 2020).

2.4 Análisis estructural

Cervera & Blanco (2002), define a una estructura como la forma en que se organizan parte de un sistema u objeto. En la ingeniería, una estructura se relaciona al sector de la construcción; son entonces, ejemplos de estructuras, los edificios, puentes, presas, etc. Se puede entender, de una forma más específica, per estructura, a la parte de una obra que soporta todo el sistema, es capaz de resistir esfuerzos de peso propio, cargas externas vivas y muertas, movimientos sísmicos, viento, lluvia, impactos, etc. La ingeniería estructural debe tomar en cuenta la funcionalidad, seguridad, economía y estética de una estructura para que se puedan desarrollar las actividades civiles.



Figura 7 Ejemplo de una estructura

Tomado de Estructuras de Ingeniería de Arqhys.com

El análisis estructural se inicia con la creación de un modelo matemático que idealiza la geometría de la estructura, las condiciones de apoyo y las fuerzas que actúan sobre ella. Esto genera un conjunto de ecuaciones que permiten determinar la respuesta y el comportamiento de la estructura. El modelo debe reflejar adecuadamente el comportamiento de la estructura en función de la rigidez de sus componentes como barras, placas y láminas; el comportamiento del material, ya sea elástico, elastoplástico, isótropo, ortótropo, anisótropo; las conexiones entre elementos, los efectos de segundo orden y las interacciones con el terreno u otros sistemas estructurales. La formulación del modelo de una estructura se basa en las ecuaciones de equilibrio que relacionan las fuerzas externas con los esfuerzos locales (Romero, Galvín, & Solís, 2020).

2.4.1 Método de elementos finitos

El Método de los Elementos Finitos (MEF) es una herramienta de análisis fundamental y ha sido ampliamente utilizada en la ingeniería desde los años 70. Aunque sus principios se remontan a más de 150 años atrás, la aparición de las computadoras ha permitido su implementación extensiva en diversas ramas de la ingeniería. El método ha tenido un significativo desarrollo en la ingeniería civil.

Nápoles, Gonzáles, & Olivares, (2015) establecen que el método de los elementos finitos se basa en representar un cuerpo mediante un conjunto de subdivisiones conocidas como elementos. Estos elementos están conectados entre sí en puntos denominados nodos. Se discretiza el elemento o sistema dividiéndolo en un sistema equivalente compuesto de cuerpos pequeños, cuyo conjunto representa el cuerpo total. La solución obtenida para cada una de estas unidades pequeñas se combina para formar la solución completa. En problemas estructurales, esta solución implica determinar los desplazamientos de los nodos, y a partir de estos desplazamientos, calcular las deformaciones y tensiones en el sistema analizado. Las propiedades de los elementos que conectan los nodos dependen del material asignado a cada

elemento, lo que define su rigidez, así como la geometría de la estructura que se modela. Según las Leyes de la Elástica, las deformaciones y las fuerzas externas están relacionadas mediante la rigidez y las relaciones constitutivas del elemento. Al trabajar en el régimen elástico, las ecuaciones que describen el sistema pueden expresarse en forma matricial, como se indica a continuación:

$$[K]u^{\vec{}} = F$$

Donde:

- $[K]$ = matriz rigidez del sistema
- $\{u\}$ = vector de desplazamiento
- $\{F\}$ = vector de cargas

2.4.2 Herramientas de Análisis estructural

2.4.2.1 Karamba 3D



Figura 8 Logo de Karamba 3D

Tomado de Karamba Food4Rhino

Karamba3D es una herramienta interactiva de ingeniería paramétrica que permite realizar análisis de elementos finitos (FEA) de manera rápida y precisa. Está especialmente diseñada para satisfacer las necesidades de los profesionales en las etapas iniciales del proceso de diseño.

Karamba3D se integra en el entorno paramétrico de Grasshopper dentro del software de modelado 3D “Rhino3D”. Esta integración facilita la combinación de modelos geométricos parametrizados, cálculos de elementos finitos y algoritmos de optimización.

Además, Karamba3D puede utilizarse como una biblioteca independiente .NET, lo que permite incorporar la funcionalidad de elementos finitos en scripts personalizados (Karamba 3D).

2.5 Optimización Estructural

La optimización estructural se enfoca en elegir las variables de diseño adecuadas para lograr el mejor desempeño posible dentro de los límites establecidos para el comportamiento estructural, la geometría y otros factores. El objetivo es cumplir con la función objetivo específica bajo las cargas o condiciones generales estipuladas.

En términos generales, un problema de optimización se compone de los siguientes elementos (Casado, 2022):

- Variables de diseño o variables de optimización, que son los parámetros que definen el problema a resolver.
- Función objetivo o de desempeño, la cual depende de las variables de optimización. La solución óptima se encuentra en el máximo o mínimo global de esta función.
- Restricciones, que se expresan mediante igualdades o desigualdades y limitan el universo de soluciones posibles para el problema.

Según Sánchez (2012), la optimización en estructuras puede abordarse desde diversos enfoques principales:

- Optimización del tamaño: Se enfoca en encontrar las dimensiones óptimas de los elementos que conforman una estructura previamente definida. Es el método más sencillo, ya que el número de variables a considerar es relativamente limitado.
- Optimización de la forma: Consiste en buscar la forma óptima de una estructura con una topología fija. A partir de una forma predefinida, se optimiza modificando la conectividad entre elementos o eliminando aquellos que menos esfuerzo soportan.
- Optimización de la topología: Persigue la distribución óptima del material en una estructura mediante la eliminación de elementos y/o la modificación de la conectividad y/o las coordenadas nodales. Este enfoque es el más complejo debido al elevado número de variables involucradas. Todos los estudios realizados hasta ahora parten siempre de una morfología previa.

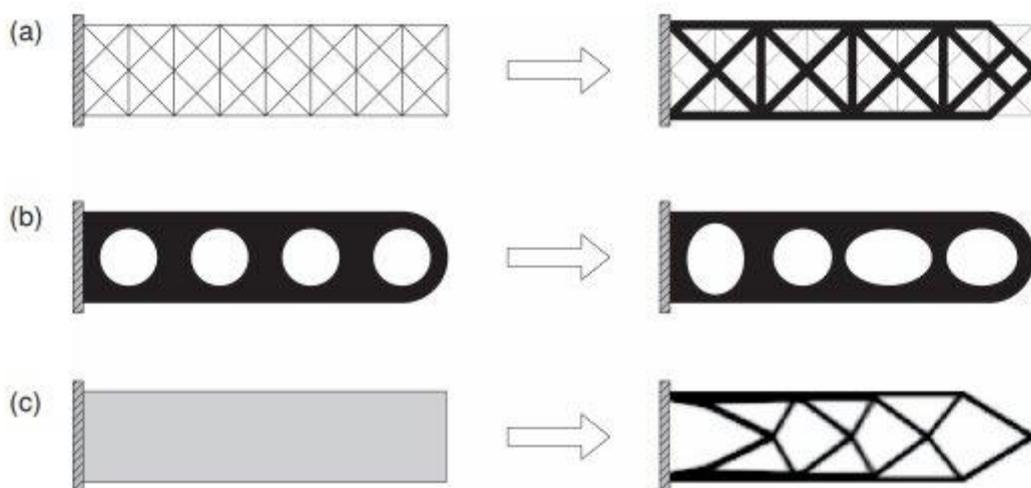


Figura 9 Optimización por tamaño, por forma y topológica respectivamente.

Tomado de: *Diferentes tipos de optimización estructural*, de Gil, J, 2018, ResearchGate

2.5.1 Algoritmos genéticos evolutivos

2.5.1.1 Algoritmos evolutivos

Estos algoritmos se fundamentan en los principios de evolución, adaptación y selección natural. Una de las características distintivas de los algoritmos evolutivos es que inician con una población aleatoria de individuos. Cada uno de estos individuos representa una posible solución al problema a optimizar, compuesto por conjuntos de valores del espacio de búsqueda (Casado Bravo, 2022).

En los algoritmos evolutivos, se incluyen los siguientes elementos:

Población de individuos: un grupo de soluciones candidatas que cumplen con las restricciones del problema y sobre las cuales se evalúa la función objetivo para determinar su aptitud.

Operadores de transformación y selección: procedimientos que generan nuevos individuos para la siguiente generación, basándose en la selección de los más aptos de la generación anterior.

2.5.1.2 Algoritmos genéticos

Los Algoritmos Genéticos, basados en la evolución genética, son una clase de algoritmo evolutivo. Este enfoque mezcla las características más aptas de los individuos de la población actual para generar nuevas soluciones superiores a las anteriores. Así, con cada generación, se logran resultados cada vez más próximos al óptimo.

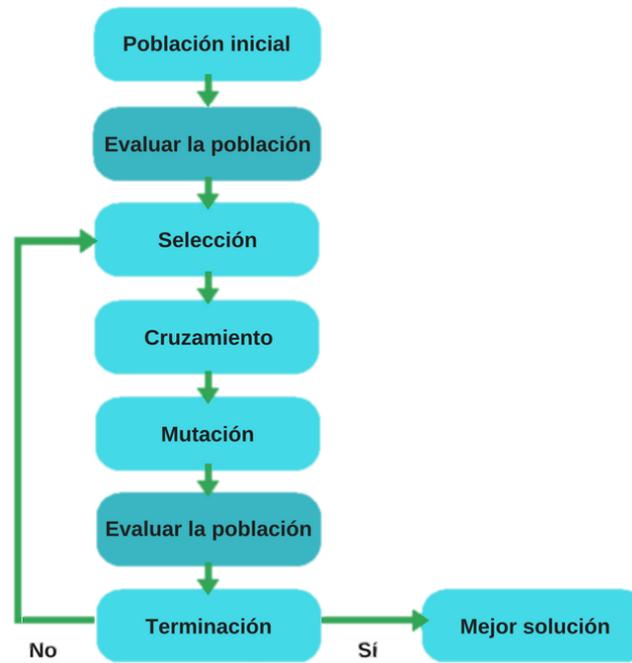


Figura 10 Algoritmo genético

Adaptado de Algoritmo genético, de García, M, 2017, Coding or not

2.5.2 Herramientas de optimización estructural

2.5.2.1 Galapagos



Figura 11 Logo de Galapagos.

Tomado de Galapagos Evolutionary Solver, de Rutten, D, 2023, Grasshopper

Galápagos es un optimizador que emplea algoritmos genéticos dentro del entorno de Grasshopper. Esta herramienta requiere que el usuario especifique la cantidad de generaciones y el valor de fitness para alcanzar el individuo óptimo después de varios ciclos de cruces y

mutaciones. Sirve para explorar el conjunto de soluciones de un problema de diseño estructural y genera gráficos visuales del proceso de optimización.

Como solucionador evolutivo, Galápagos resuelve problemas planteados matemáticamente, siempre que cuenten con una función objetivo o variables claramente definidas. Es capaz de realizar tanto optimización genética como recocido simulado, utilizando específicamente un algoritmo evolutivo (Landeta, 2021).

2.5.2.2 Karamba 3D

La herramienta “Karamba 3D” cuenta con un componente de optimización estructural llamado “Optimize Cross Section”, este componente está diseñado para la selección automatizada de secciones transversales óptimas para elementos tipo viga o láminas, teniendo en cuenta la capacidad de carga de las secciones transversales y, opcionalmente la deflexión máxima permitida de la estructura.

3. CAPÍTULO 3 (CASOS DE ESTUDIO)

El algoritmo para la modelación paramétrica y optimización será desarrollado en “Grasshopper”, una extensión del software “Rhino”. Para el análisis estructural se empleará el complemento “Karamba 3D”, y su componente “Optimize Cross Section” será utilizado para la optimización por tamaño y forma de secciones. Además, la herramienta “Galapagos” se usará para llevar a cabo una optimización tanto de tamaño y forma como topológica.

El caso de estudio incluye un algoritmo de modelación paramétrica y optimización con los siguientes alcances y limitaciones:

- Modelación paramétrica de estructuras aporticadas de acero, con vigas y columnas posicionadas ortogonalmente respecto a los ejes "X", "Y" y "Z", sin incluir voladizos o irregularidades en planta o elevación.
- La estructura tiene un límite de 10 ejes por lado, y los espaciamientos entre ejes pueden definirse individualmente.
- Modelación de vigas secundarias, permitiendo modificar su dirección global y ajustar la cantidad de vigas secundarias de forma individual en cada luz.
- Capacidad para ingresar cualquier número de plantas, con alturas de entrepiso personalizables. La altura de la planta baja se puede modificar independientemente de las demás.
- Posibilidad de asignar cargas vivas y muertas en el área de la planta, además de definir una sobrecarga en áreas específicas. Estas cargas pueden aplicarse de manera independiente en la primera planta, plantas intermedias y la cubierta.
- Inclusión de una lista de secciones IPE, HEB y HSS, basadas en los catálogos comerciales disponibles en el país, para su uso en los diferentes elementos estructurales.
- Análisis estructural mediante la creación de un modelo de elementos finitos.

- Implementación de diversas optimizaciones estructurales que busquen reducir el peso de la estructura, cumpliendo con los requisitos mínimos de deflexión y los radios de optimización adecuados. La optimización topológica en este caso consiste en buscar la mejor posición de las columnas.

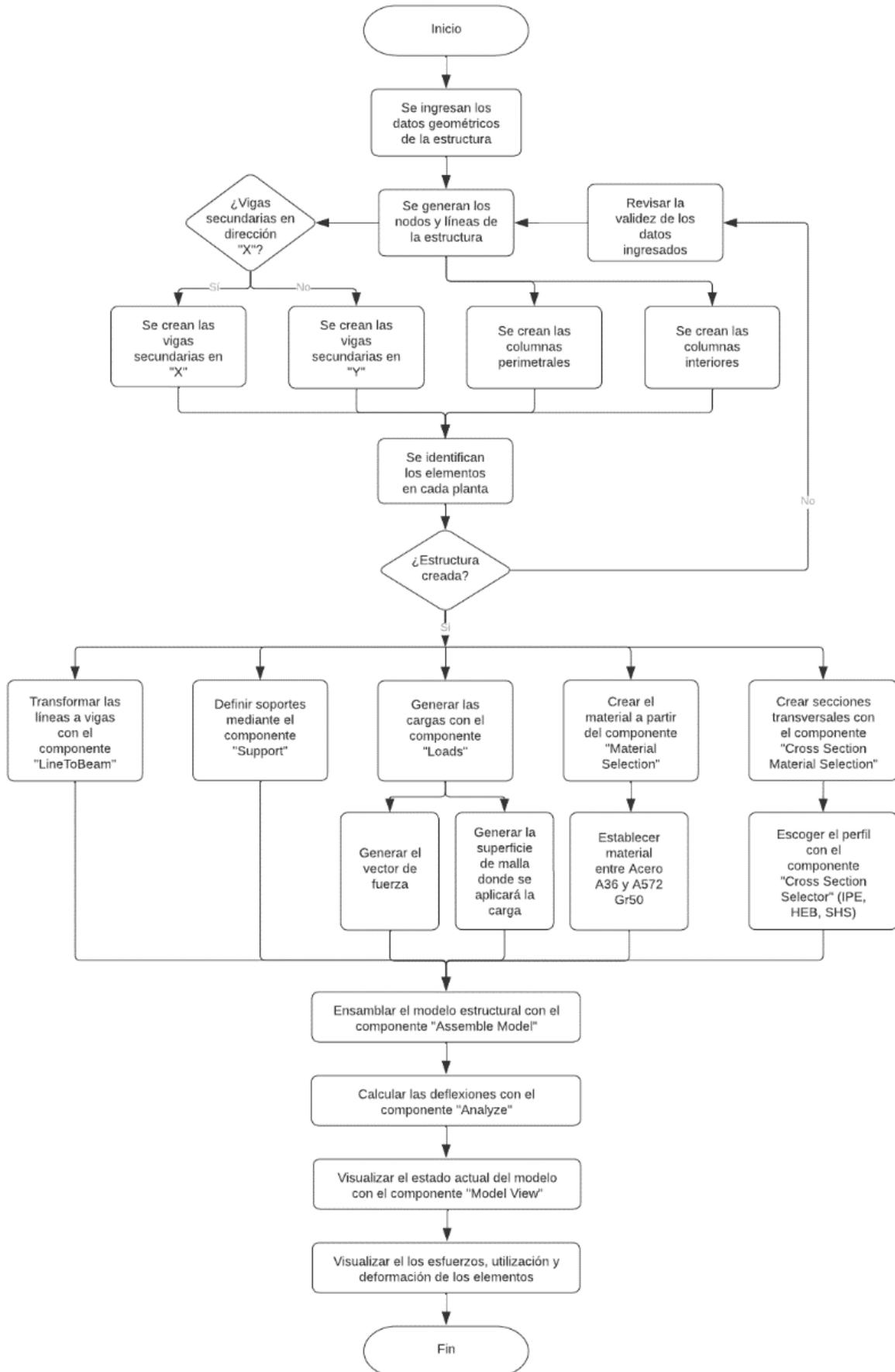


Figura 12 Esquema de procesos del algoritmo de modelación paramétrica y análisis estructural

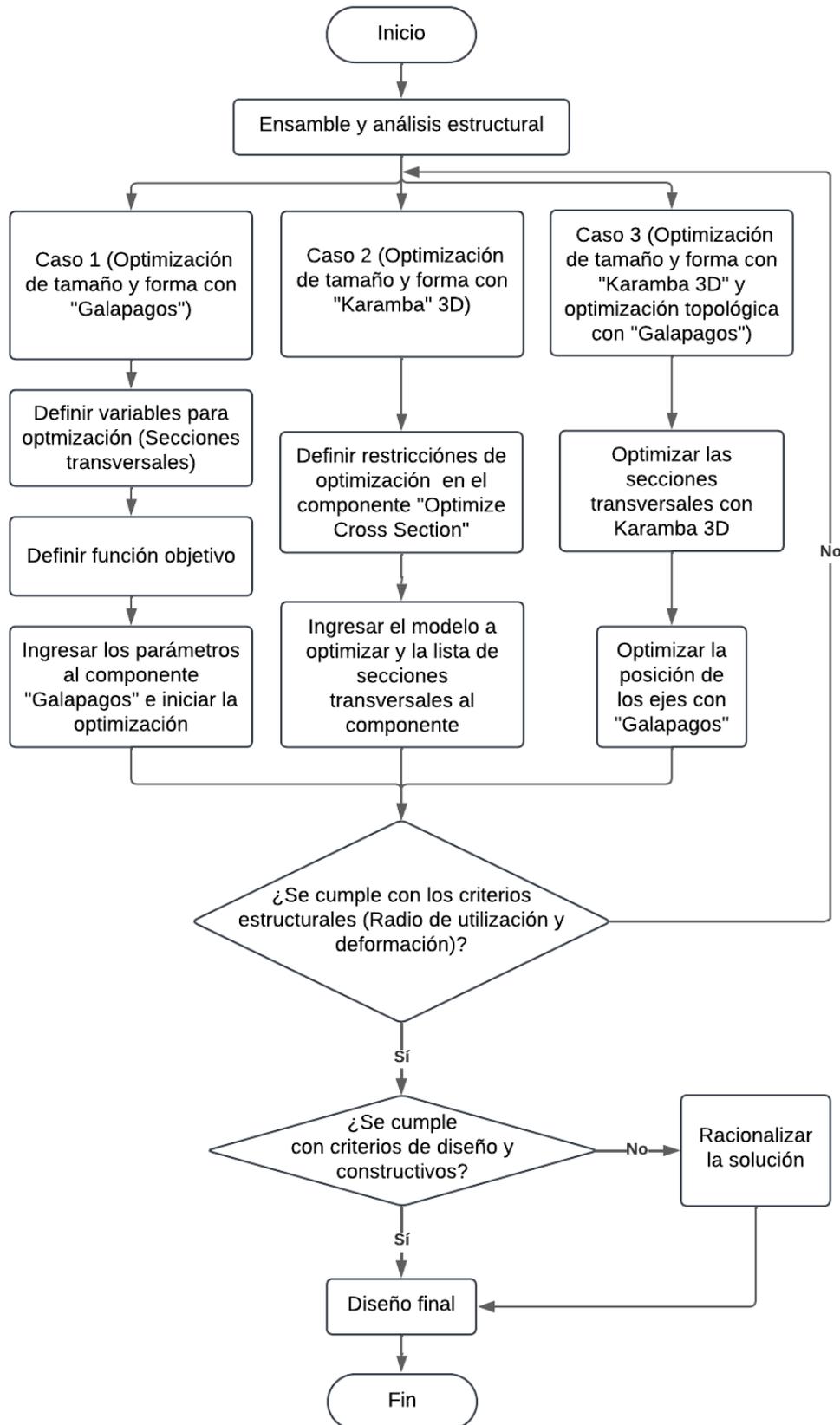


Figura 13 Esquema de procesos del algoritmo de modelación estructural

3.1 Modelación paramétrica

El algoritmo de modelación paramétrica tiene como objetivo representar la geometría de la estructura mediante líneas y puntos, que corresponderán a los diferentes elementos estructurales, como columnas, vigas principales y vigas secundarias.

3.1.1 Datos de entrada

Primero, se crearon componentes que permiten al usuario ingresar los datos necesarios para definir la geometría de la estructura. Para ello, se utilizó el componente “Number Slider”, que permite ingresar los siguientes datos:

- Longitud en “X”
- Longitud en “Y”
- Número de ejes en “X”
- Número de ejes en “Y”
- Altura del entrepiso de la planta baja
- Altura de los pisos superiores
- Número de plantas

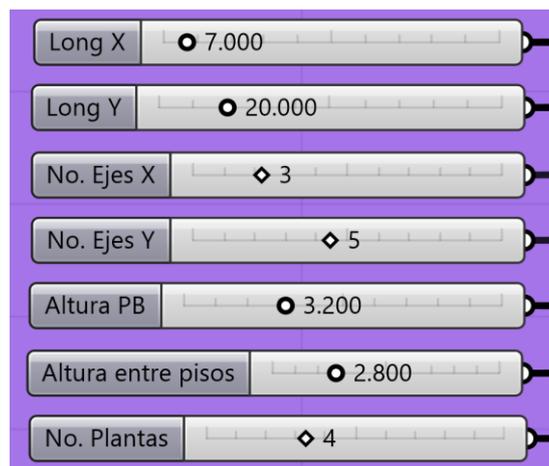


Figura 14 Sliders de entrada de datos geométricos

Adicionalmente, con el componente “Gen List”, es posible definir las posiciones globales de los ejes en “X” y “Y”. El usuario deberá configurar previamente estas listas para definir la cantidad de ejes y la longitud total de cada eje. Cabe destacar que este componente puede conectarse a la herramienta “Galapagos”, lo que permite que el algoritmo genético varíe estos datos para optimizar la posición de las columnas.



Figura 15 Herramienta "Gene List" para ingreso de posiciones de ejes

También se ofrece al usuario la posibilidad de seleccionar la dirección general de todas las vigas secundarias (en “X” o en “Y”) mediante el componente “Value List”.



Figura 16 Componente "Value List" para seleccionar la dirección de las vigas secundarias

DATOS

Long X

Long Y

No. Ejes X

No. Ejes Y

Altura PB

Altura entre pisos

No. Plantas

EJES X

| | |
|---|-----|
| 0 | 2.8 |
| 1 | 3.5 |
| 2 | 2.9 |

EJES Y

| | |
|---|------|
| 0 | 13.8 |
| 1 | 14.5 |
| 2 | 8.2 |
| 3 | 3.0 |
| 4 | 7.0 |

Vigas secundarias en Y ▼

Figura 17 Bloque de todos los parámetros de entrada

3.1.2 Nodos y líneas

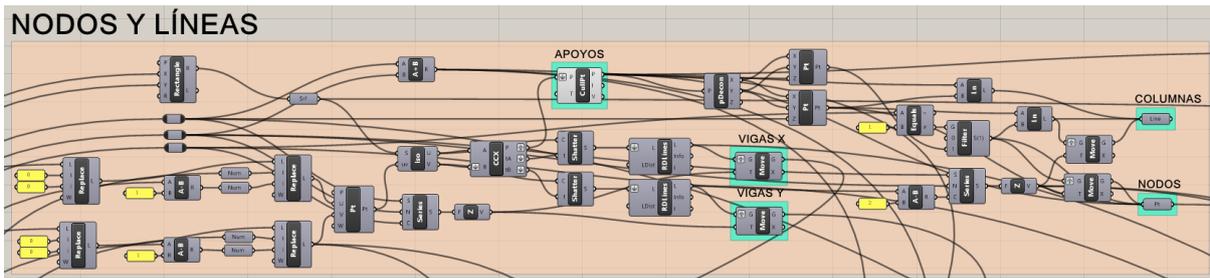


Figura 18 Bloque para generar nodos y líneas

Para modelar la geometría de la estructura, primero es necesario establecer la cuadrícula (grid) donde se intersecan los ejes. Para esto fue necesario tomar los datos de las posiciones de cada eje, y con el componente “Replace Items” se tomó el primer valor de la lista y se reemplazó con 0, y posteriormente se tomó el ultimo valor de la lista y se reemplazó con la longitud total del eje. Este proceso se realiza con las dos listas de ejes, y es necesario realizarlo para que al momento que el algoritmo genético varíe las posiciones de los ejes, siempre se mantengan ejes en el perímetro de la estructura. Finalmente conectan las listas a la caja “Point Oriented” para definir las coordenadas.

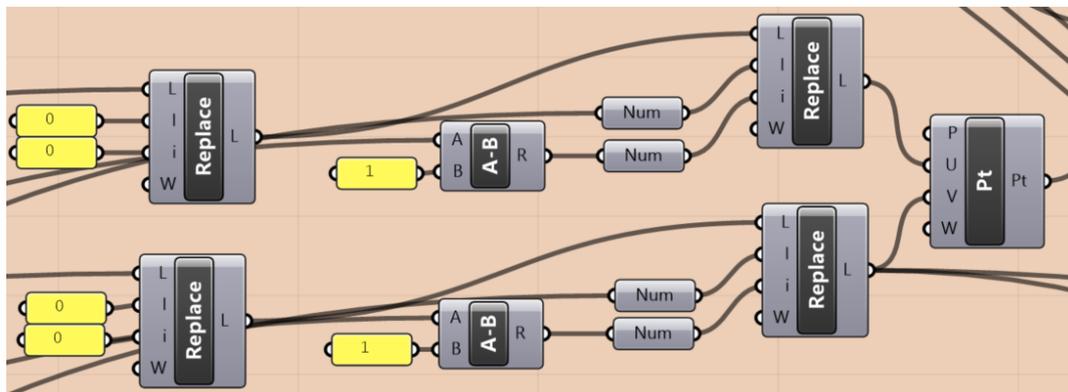


Figura 19 Algoritmo para mantener ejes en el perímetro de la estructura

Debido a la forma en la que el complemento “Point Oriented” gestiona los datos, no se generan todos los puntos de la malla, por lo tanto, es necesario hacer lo siguiente: Conectar los puntos generados al complemento “Iso Curve”, el cual crea líneas intersecando puntos dados dentro de una superficie. De esta manera se tienen las líneas de la malla, y con la herramienta

“Curve | Curve”, se generan puntos a partir de la intersección de líneas que serán los puntos de apoyo de la estructura.

El comando “Shatter” divide las líneas en función de los puntos de intersección, generando así las líneas individuales de cada paño de la estructura. Finalmente se incorporan los comandos “Remove Duplicate Lines (Karamba3D)” y “Cull Duplicates” para eliminar líneas y puntos que se encuentren en la misma posición.

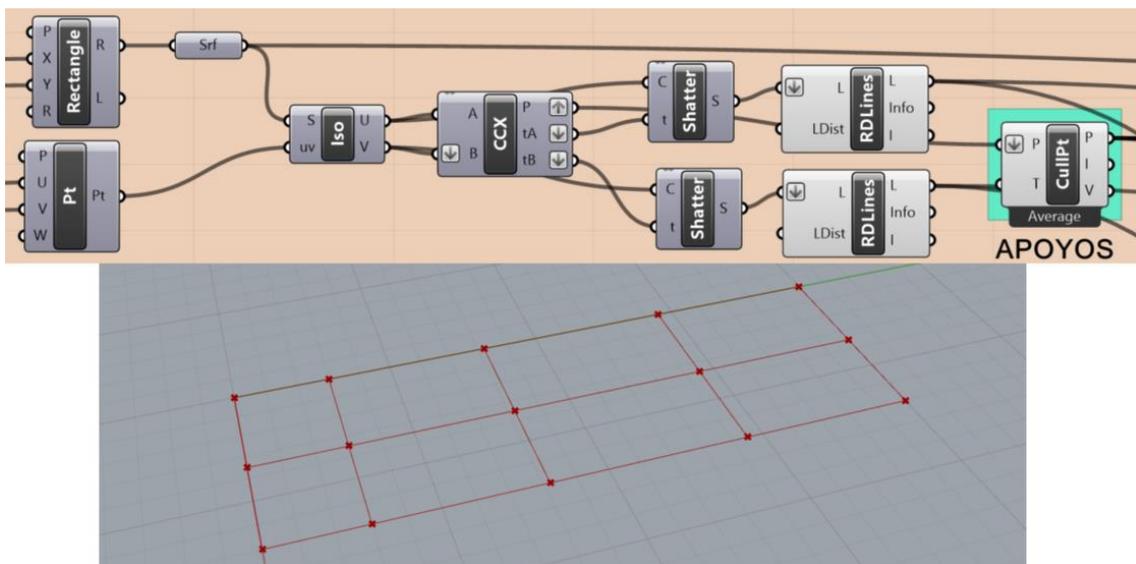


Figura 20 Creación de la malla guía de la estructura

Para replicar esta malla en cada planta es necesario mover los elementos en las posiciones correspondientes en el eje “Z”. Se movió las vigas con el comando “Move” conectando en la entrada del vector de transición una serie que define la posición de partida, el paso y el número de veces que se debe mover las vigas.

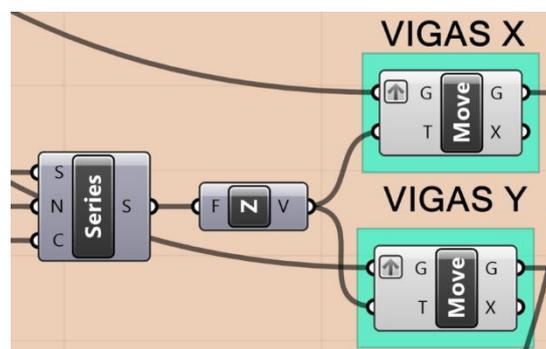


Figura 21 Comando “Move” para replicar las vigas en las demás plantas

Se sigue un proceso similar para mover los demás elementos. Los nodos se han replicado en las demás plantas con el comando “Deconstruct” y “Construct Point”, reemplazando el eje “Z” con el valor de la altura de las plantas.

Se ha usado la herramienta “Stream Filter” estableciendo una condición en la cual, si el número de plantas es uno, entonces solo se moverán los elementos una sola vez. Es necesario realizar este proceso para que los elementos se repliquen adecuadamente en cada planta.

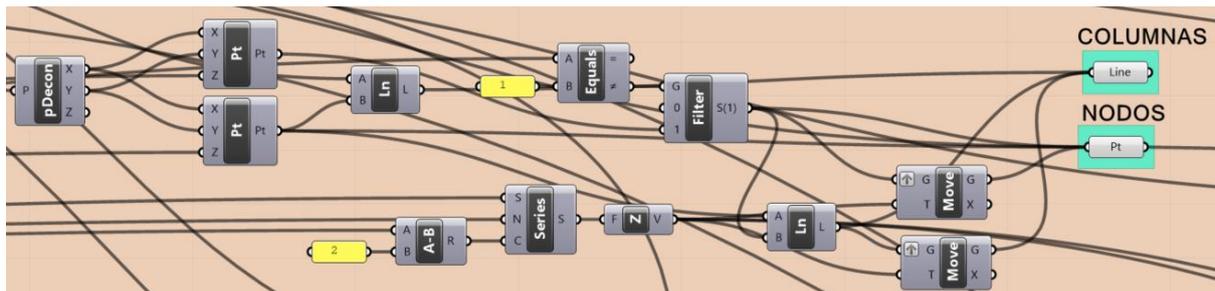


Figura 22 Replicar los nodos y vigas en las demás plantas

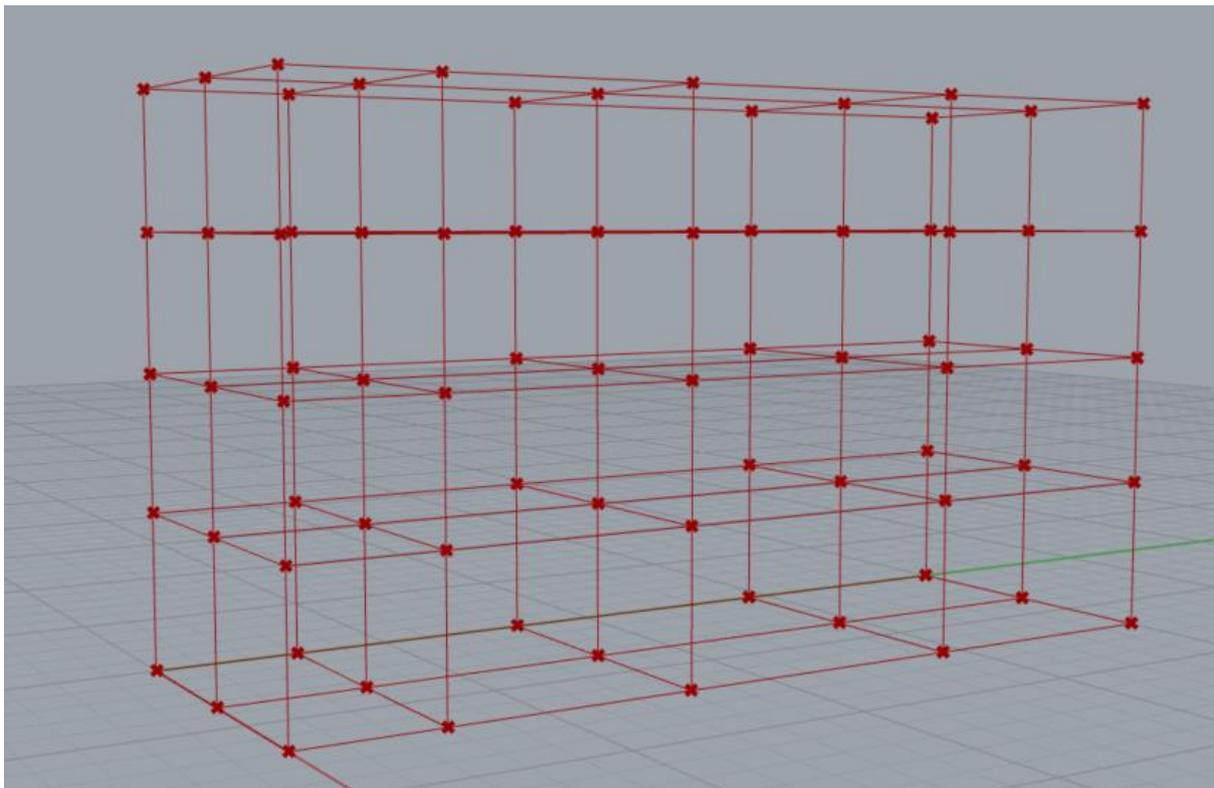


Figura 23 Modelo geométrico de la estructura con vigas y columnas

3.1.3 Vigas Secundarias

Se han generado dos algoritmos para crear las vigas secundarias, uno para modelar vigas secundarias en “X” y otro para vigas secundarias en “Y”. Solo uno de los algoritmos es funcional dependiendo de la elección del usuario en cuanto a la dirección de las vigas secundarias.

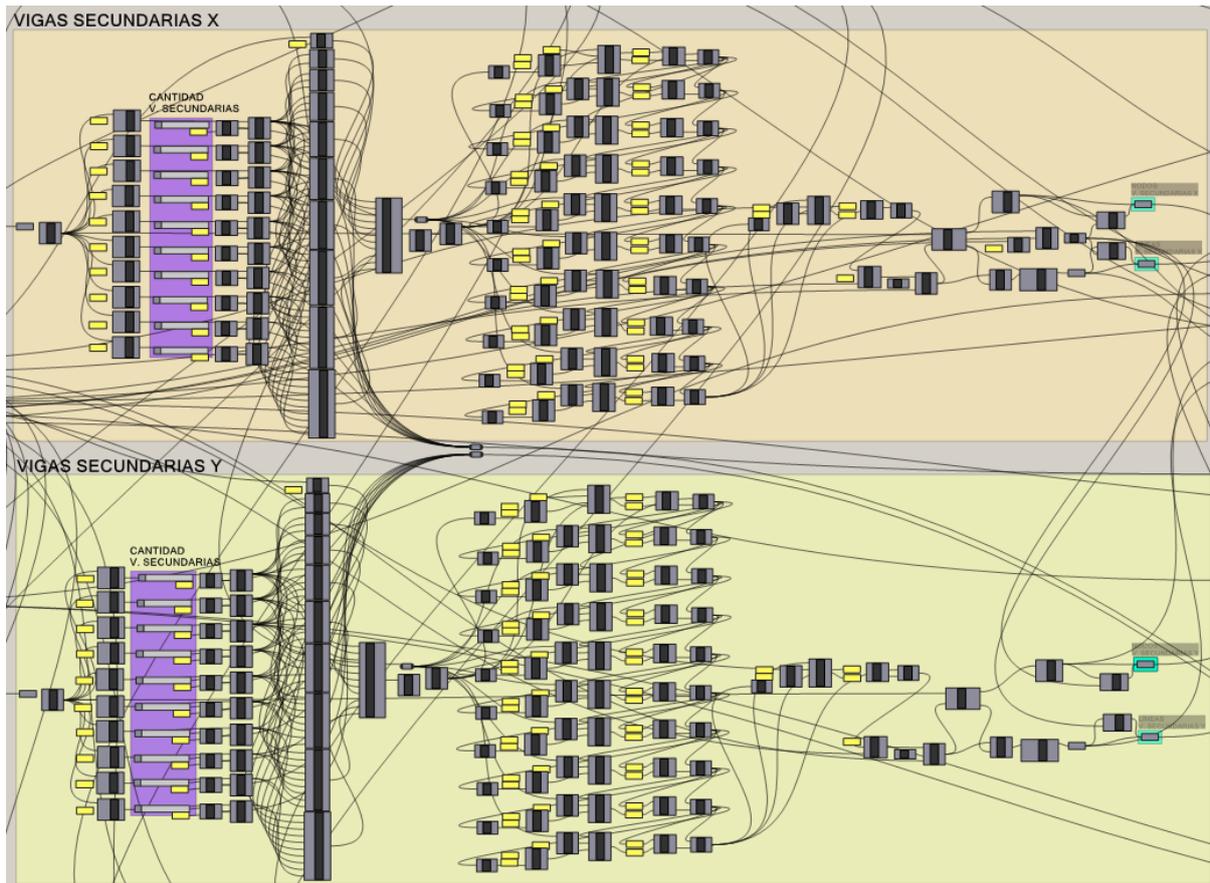


Figura 24 Bloque para generar vigas secundarias

Este algoritmo permite generar las vigas secundarias en hasta 10 luces utilizando el componente “List Item”, que extrae las vigas de cada luz; posteriormente la viga seleccionada es dividida según la cantidad de vigas secundarias escogida por el usuario. Este proceso se ejecuta para las demás luces y mediante el componente “Merge” y “Stream Filter” se limita las divisiones a las luces existentes en la estructura.

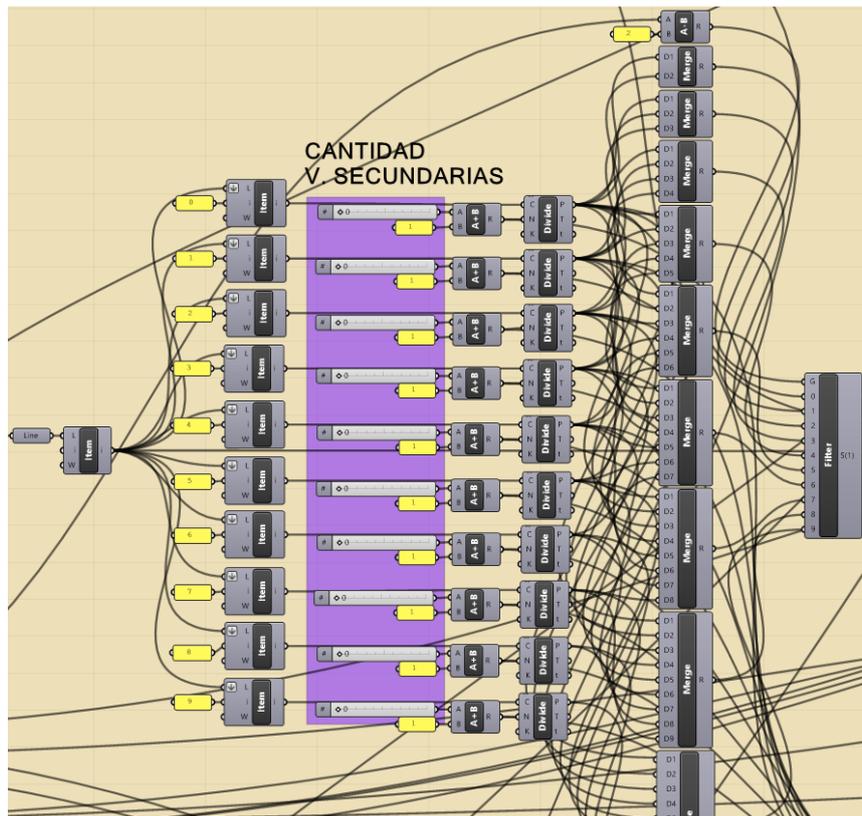


Figura 25 División de las vigas principales para ubicar las vigas secundarias

Al dividir las vigas, se generan los nodos donde se ubicarán las vigas secundarias. Sin embargo, es necesario eliminar los puntos que coinciden con los puntos de las vigas principales.

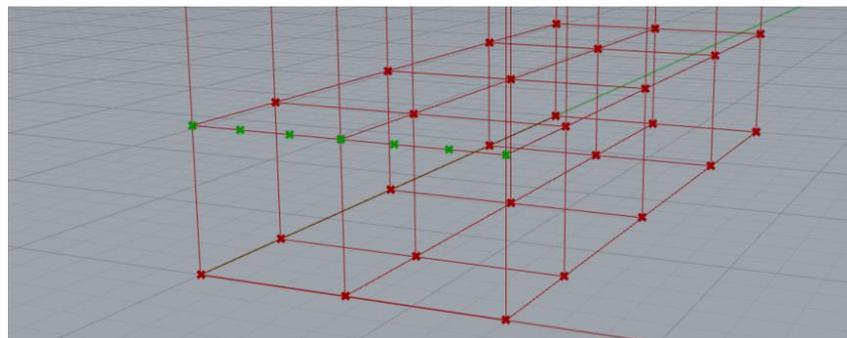


Figura 26 Nodos creados al dividir la viga principal

Se explicará cómo se eliminan los nodos que no se necesitan, tomando como ejemplo que estamos generando vigas secundarias en el eje “X”: Se crea una lista que reemplace todas las coordenadas en “X” de los nodos con la coordenada en eje “X” del primer eje, y con el

componente “Equality” se evalúa la igualdad entre la esta lista y la lista original; la herramienta entrega un patrón de datos booleanos.

Posteriormente, con la herramienta “Cull Pattern” se eliminan los datos redundantes. Se han eliminado entonces las coordenadas en “X” que estén en la misma posición del primer eje, este proceso se repite para eliminar nodos superpuestos en hasta 10 ejes.

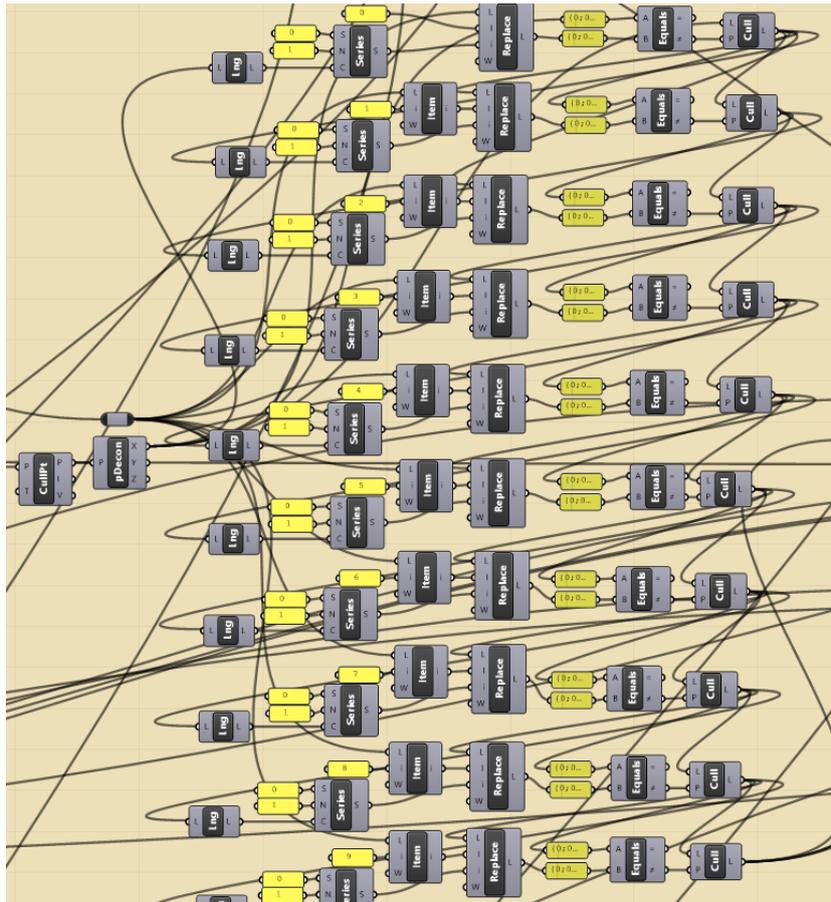


Figura 27 Eliminación de los nodos superpuestos con los nodos de las vigas principales

Finalmente, se generan las vigas secundarias utilizando “Line SDL”, que traza las líneas desde los puntos de inicio corregidos, con una dirección y longitud establecida. Estas vigas se recortan para trabajar de manera independiente en cada paño y, posteriormente, se eliminan las duplicidades con los comandos “Remove Duplicate Lines (Karamba3D)” y “Cull Duplicates”. Las vigas secundarias también se replican en las demás plantas de manera similar al proceso de replicación de las vigas principales.

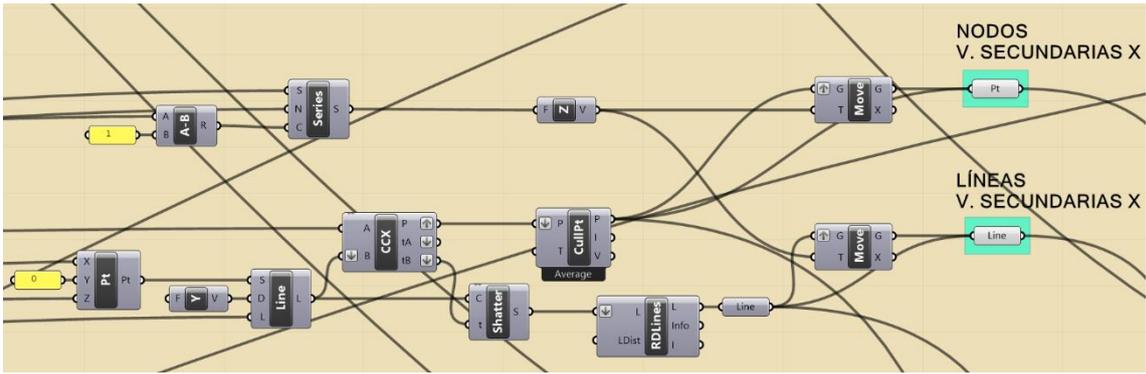


Figura 29 Generar las vigas secundarias en las demás plantas

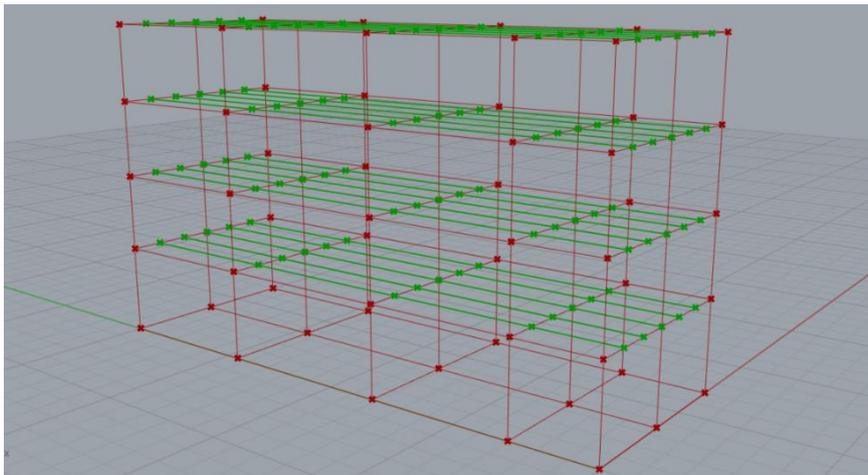


Figura 28 Modelo de la geometría incluyendo las vigas secundarias

3.1.4 Columnas perimetrales

Aunque las columnas de la estructura ya están modeladas, es pertinente diferenciar entre las columnas perimetrales y las interiores, ya que trabajar con estos dos tipos de elementos por separado, a cada uno se le puede atribuir una sección diferente.

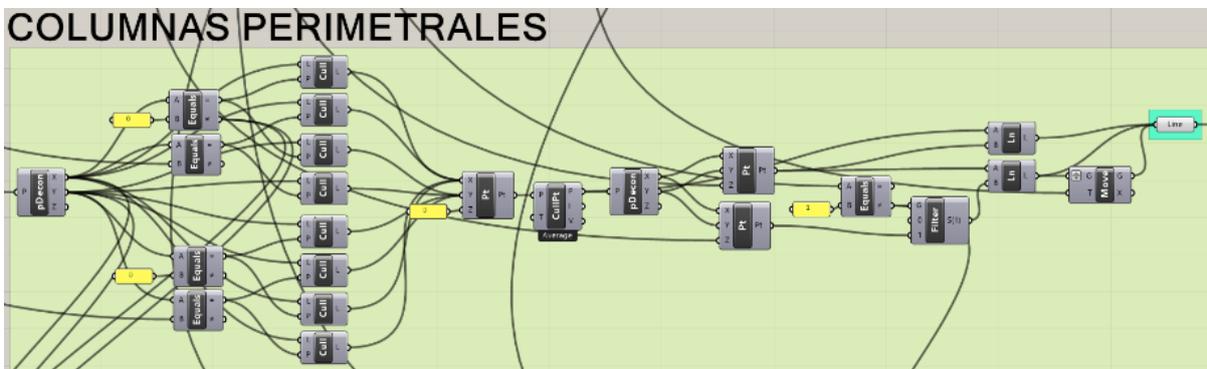


Figura 30 Bloque para identificar las columnas perimetrales

- Elementos en las plantas intermedias
- Elementos en la cubierta

El modelo funciona de la siguiente forma:

- Si solo existe una planta, los elementos se consideran de cubierta.
- Si la estructura tiene dos plantas, habrá elementos en la primera planta y en la cubierta.
- Si la estructura tiene tres plantas, existirán elementos en la planta baja, en la planta intermedia y en la cubierta.
- Para estructuras con más de tres plantas, todas las plantas intermedias se tratan como un solo tipo de elemento, compartiendo la misma sección.

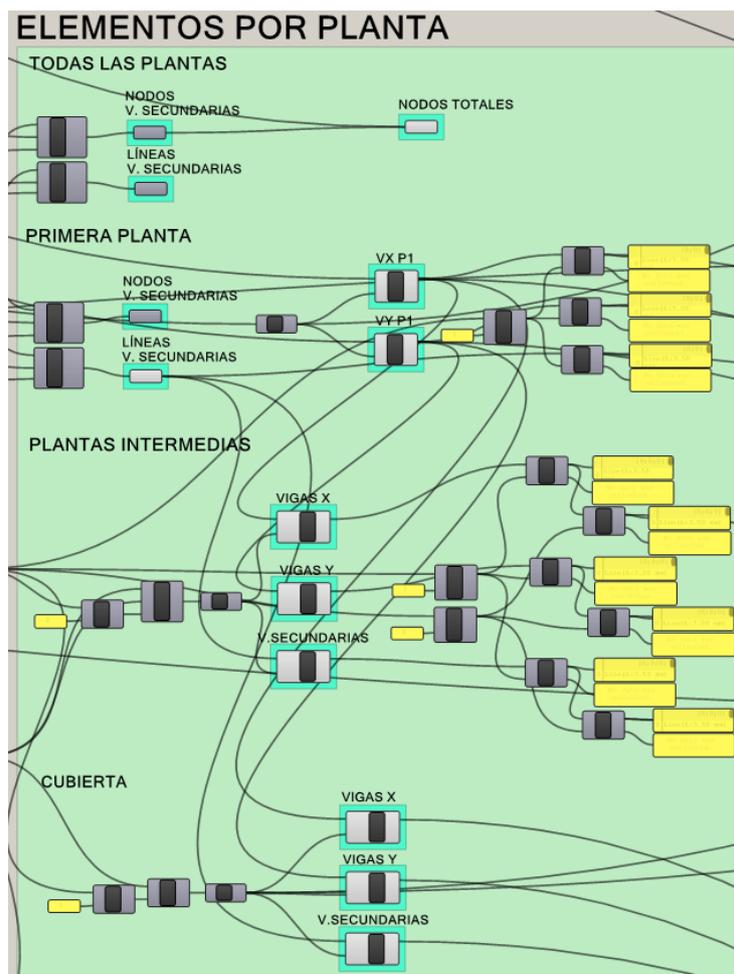


Figura 32 Bloque para identificar las vigas por planta

El proceso para identificar las vigas en la primera planta es el siguiente:

Se utiliza el componente “Stream Filter” para filtrar las vigas secundarias generadas en dirección “X” y “Y” y seleccionar solo aquellas con la dirección definida por el usuario. Luego, se toman las líneas que representan las vigas principales en “X” y “Y” de la cuadrícula guía, y utilizando el componente “Move”, se ubican las vigas en la primera planta. Así, se disponen las vigas principales y secundarias en la primera planta.

Con los componentes “Equality” y “Stream Gate”, se evalúa si el número de plantas es igual a 1. Si es así, no se envía la información, ya que en este caso solo existirían elementos de cubierta. Si el número de plantas es mayor a 1, el algoritmo envía esta información para su inclusión en el modelo estructural.

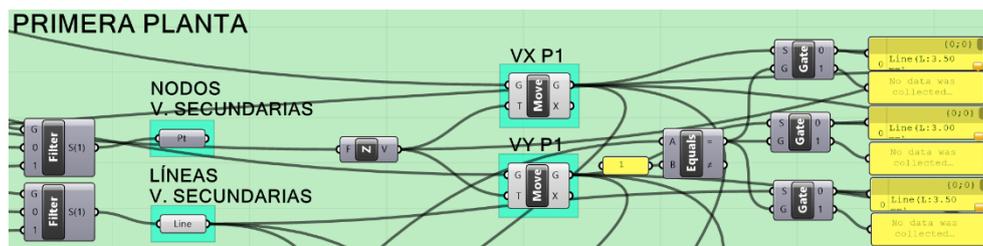


Figura 33 Identificar vigas en la primera planta

Para generar las vigas en las plantas intermedias se sigue el mismo proceso y los mismos componentes que el de las vigas en la primera planta. El contenido se envía al ensamble estructural únicamente si el número de plantas es mayor a 2, ya que solo en este caso existen plantas intermedias.

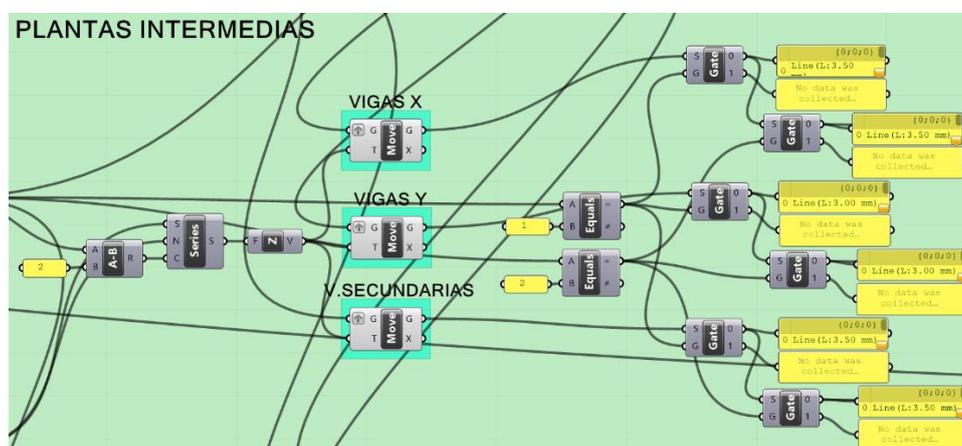


Figura 34 Identificar vigas en las plantas intermedias

Para generar las vigas en la cubierta, se utiliza nuevamente el componente “Move” para posicionar correctamente las vigas. En este caso, no es necesario aplicar un filtro, ya que la cubierta siempre será parte del modelo estructural, independientemente del número de plantas.

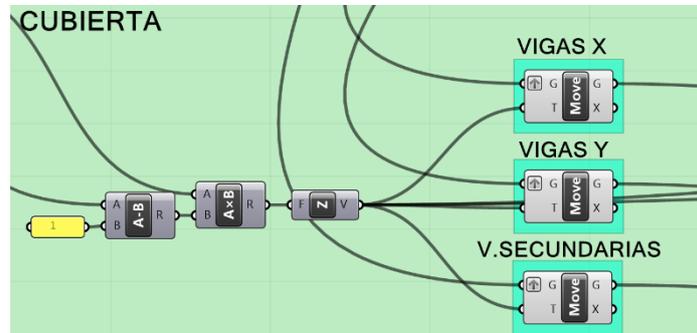


Figura 35 Identificar vigas en la cubierta

3.2 Ensamble del modelo estructural

Una vez completado el algoritmo para la modelación paramétrica de la geometría de la estructura, se procede al ensamblaje del modelo estructural. Para ello, se utiliza el componente “Assemble Model” de la extensión “Karamba 3D”, el cual permite generar un modelo de elementos finitos a partir de entidades previamente definidas. Los parámetros de entrada necesarios para este proceso son:

- Elementos
- Apoyos
- Cargas
- Sección transversal
- Material

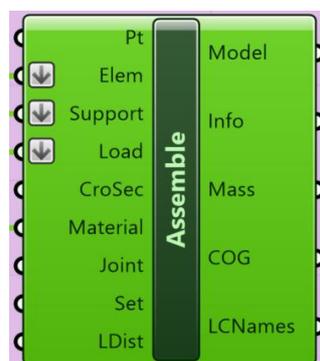


Figura 36 Componente para ensamblar el modelo estructural

3.2.1 Líneas a vigas

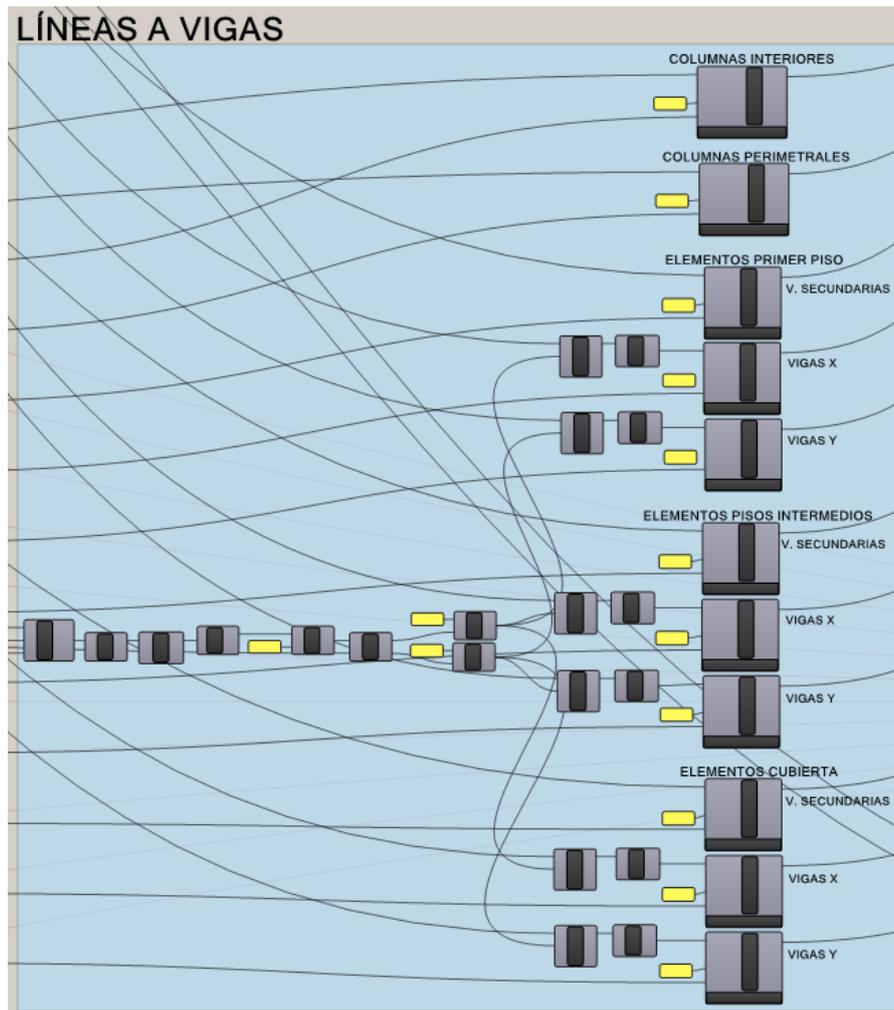


Figura 37 Bloque para crear elementos estructurales a partir de líneas

Para ingresar los elementos en el componente “Assemble Model”, es necesario transformar las líneas de la geometría en elementos con propiedades estructurales específicas. Este proceso se realiza utilizando el componente “LineToBeam”, el cual se ha utilizado once veces, uno para cada tipo de elemento estructural.

En cada caja “LineToBeam” se ha asignado un identificador con la abreviación del nombre del elemento que aparecerá reflejado en la vista del modelo en la ventana de “Rhino”. Además, se debe ingresar la sección transversal correspondiente para cada tipo de elemento. El algoritmo para la generación de estas secciones transversales se explicará más adelante.

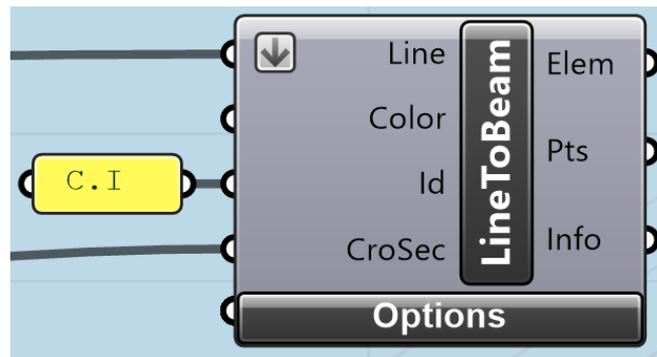


Figura 38 Componente "LineToBeam"

Los elementos estructurales generados son los siguientes:

- Columnas interiores
- Columnas perimetrales

Elementos de la primera planta:

- Vigas secundarias
- Vigas principales "X"
- Vigas principales "Y"

Elementos plantas intermedias

- Vigas secundarias
- Vigas principales "X"
- Vigas principales "Y"

Elementos en la cubierta:

- Vigas secundarias
- Vigas principales "X"
- Vigas principales "Y"

Cabe mencionar que se realizó un proceso previo para el manejo de las vigas principales en las direcciones "X" y "Y". Para que "Karamba 3D" pueda realizar correctamente el análisis estructural, es fundamental que las vigas principales a las que se conectan las vigas secundarias estén divididas en segmentos ubicados entre las vigas secundarias.

Este proceso de división de las vigas principales se llevó a cabo utilizando la herramienta “Divide”. Dado que el número de divisiones varía en función de la longitud de cada luz, se seleccionó un divisor en común del número de vigas secundarias en cada tramo. Este número es el número de divisiones que se ha aplicado uniformemente a todas las vigas principales, resultando en un proceso eficiente y efectivo.

Finalmente, con el componente “Stream Gate” se filtró la división de las vigas principales en “X” y “Y” para aplicarla únicamente a aquellas que están conectadas a las vigas secundarias.

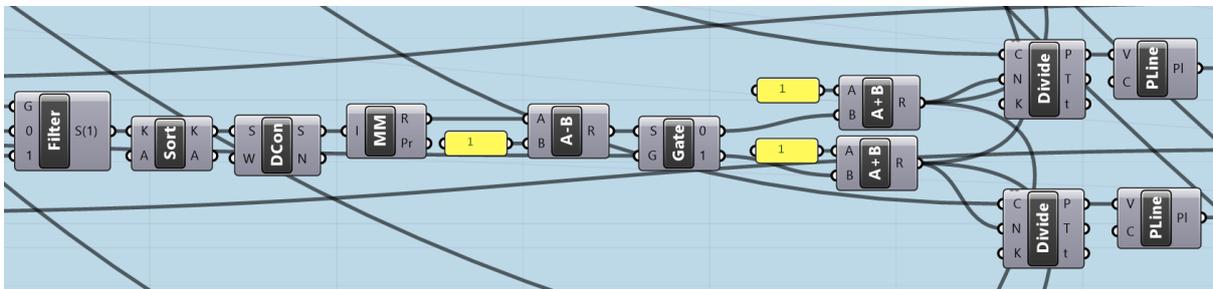


Figura 39 Algoritmo para dividir las vigas principales

3.2.2 Apoyos



Figura 40 Componente Support para definir apoyos

Para la definición de los apoyos, se utilizaron los puntos previamente modelados, los cuales fueron ingresados en el componente “Support” de la extensión “Karamba 3D”. Este

componente permite crear apoyos en los puntos especificados y seleccionar, mediante casillas, los grados de libertad que se desean restringir para cada uno de ellos.

3.2.3 Cargas

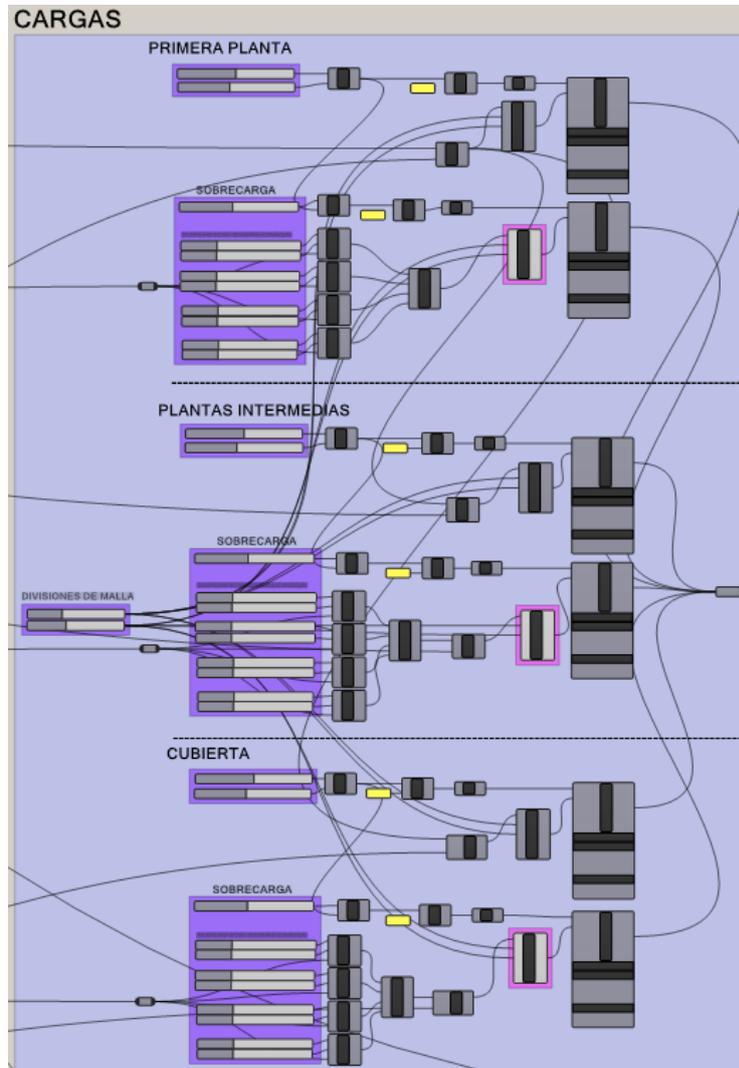


Figura 41 Bloque para definir las cargas

El algoritmo ha sido diseñado para permitir al usuario insertar diferentes tipos de cargas para cada planta: carga viva, carga muerta y sobrecargas. Estas cargas pueden variar en función de la ubicación dentro de la estructura, permitiendo que las cargas para la primera planta, las plantas intermedias y la cubierta sean definidas de manera independiente. Adicionalmente, el usuario puede aplicar sobrecargas en áreas específicas de cada planta, lo que es particularmente

útil en casos donde existen diferentes usos de servicio, como jardines, bibliotecas, piscinas, entre otros.

El componente "Loads" de la extensión "Karamba 3D" fue utilizado para generar estas cargas, permitiendo la creación de cualquier tipo de carga estructural. En este caso, se aplicó una carga de malla constante, que corresponde a una carga en kN/m^2 distribuida sobre un área específica. Esta carga es transferida a los elementos estructurales mediante el concepto de áreas tributarias. Para lograrlo, es necesario seleccionar la opción "Line loads", de modo que la carga sea aplicada como una carga uniformemente distribuida sobre las vigas.

El componente "Loads" requiere ingresar un vector de fuerza y una malla, creada con el componente "Mesh Surface", que define la superficie sobre la que se aplicará la carga. Además, se debe definir el número de subdivisiones de la malla en ambas direcciones. Es importante tomar en cuenta que un mayor número de subdivisiones permite una distribución más precisa de la carga sobre los elementos estructurales. Este parámetro puede ser ajustado por el usuario.

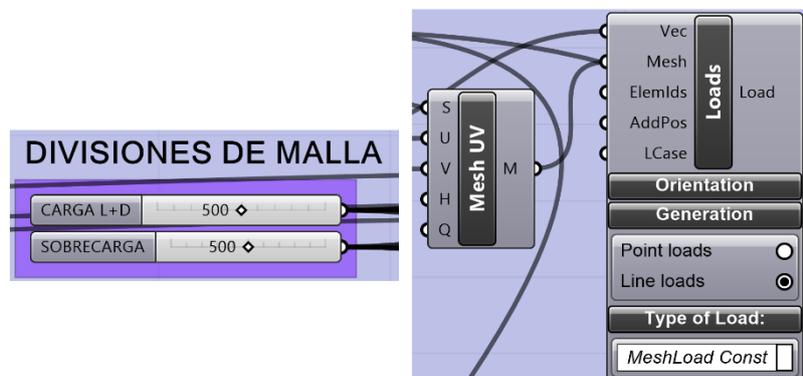


Figura 42 Componentes para generar una carga

Para cada planta, se generó un algoritmo similar que permite definir las cargas. Dos "Number Slider" permiten al usuario ingresar las magnitudes de la carga viva y la carga muerta en kN/m^2 , que luego se aplican en toda la planta. Estas magnitudes se suman a un vector Z, que se conecta al componente "Loads".

El proceso para generar sobrecargas sigue el mismo principio, pero en este caso, la superficie sobre la cual se aplicará la carga es variable. Se utiliza el componente "4Point Surface", que crea una superficie a partir de cuatro puntos, cuyas coordenadas son definidas por el usuario a través de componentes "Number Slider".

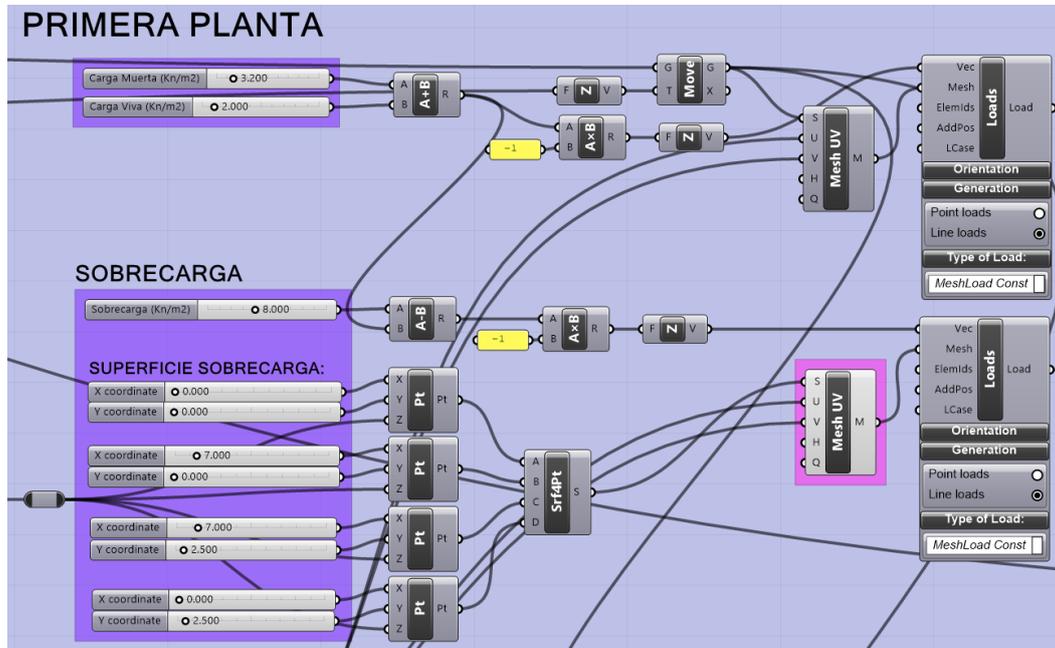


Figura 43 Algoritmo para definir cargas en la primera planta

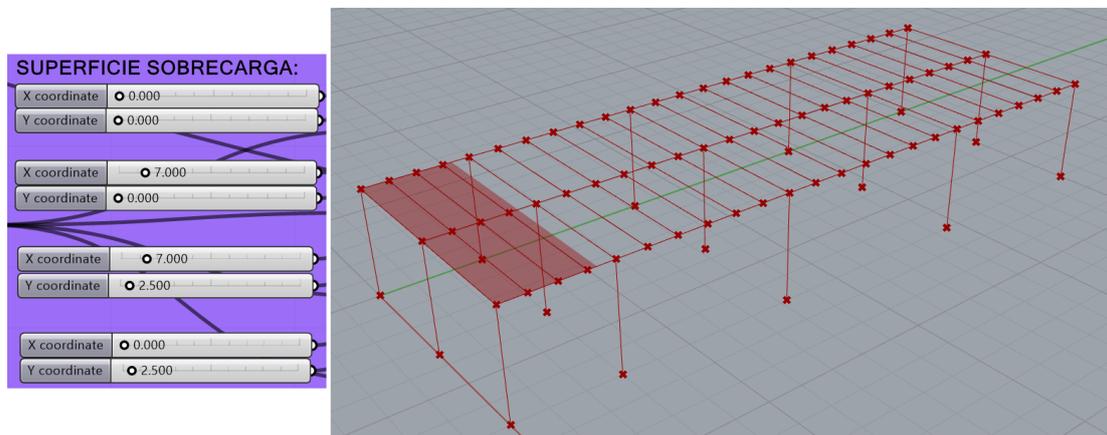


Figura 44 Superficie donde se aplica la sobrecarga

Este algoritmo se repite para todas las plantas, variando únicamente la altura en la que se aplicarán las cargas.

3.2.4 Materiales y secciones transversales

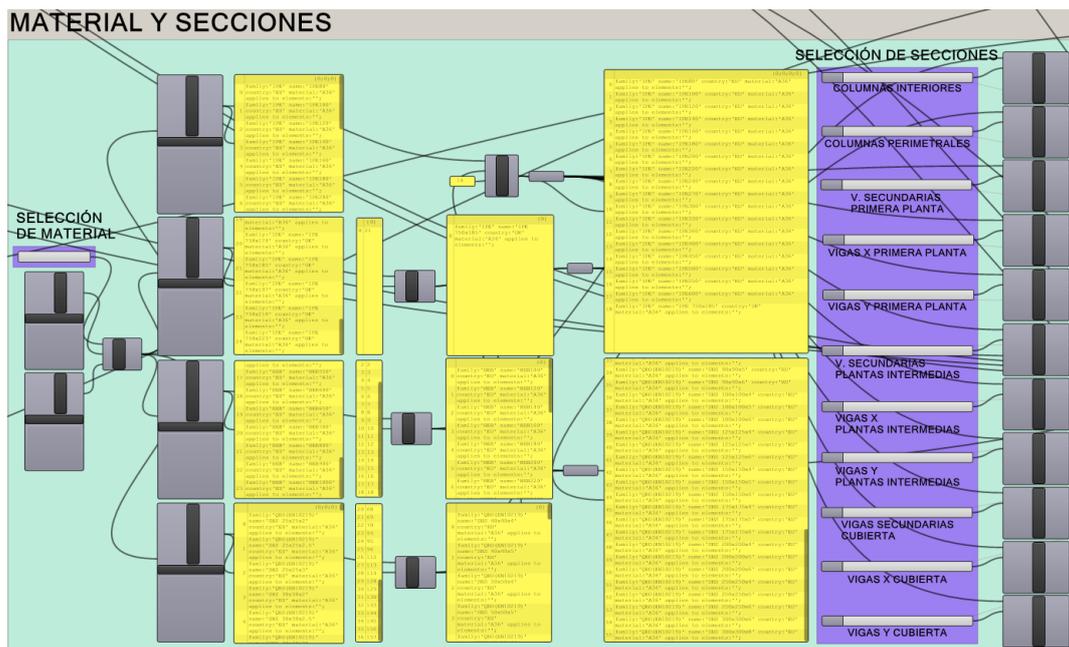


Figura 45 Bloque para definir los materiales y las secciones transversales

3.2.4.1 Materiales

El componente “Material Selection” de la herramienta “Karamba 3D” permite seleccionar materiales a partir de una lista clasificada por familia y nombre. En este caso, se han definido dos materiales comunes en la región: el Acero A36 y el Acero A572GR50. El usuario puede seleccionar cualquiera de estos materiales a través de una lista generada con el componente “Value List”.

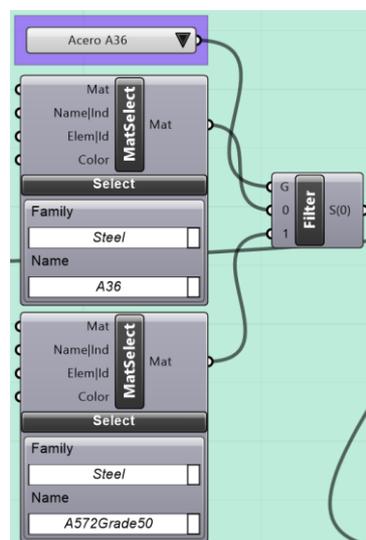


Figura 46 Componentes para definir los materiales

3.2.4.2 Secciones transversales

Para la selección de las secciones transversales, se utilizó el componente “Cross Section Material Selection” de “Karamba 3D”, que permite escoger secciones según país, forma o familia. En este caso, se han implementado tres componentes con las siguientes secciones:

- Secciones IPE para las vigas principales y secundarias.
- Secciones HEB para columnas.
- Secciones cuadradas SHS para columnas.

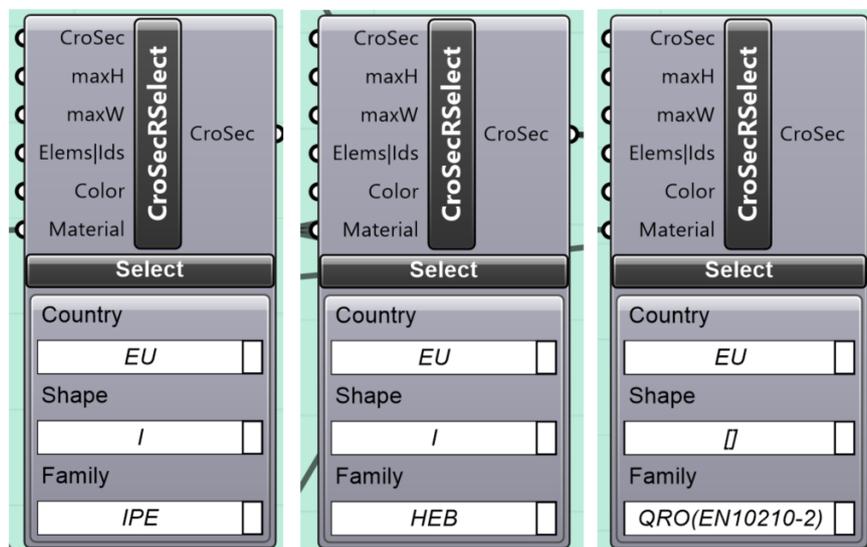


Figura 47 Componente “Cross Section Material Selection” para definir las secciones transversales

Con el componente “List Item” se han filtrado las secciones disponibles en catálogos comerciales del país, los cuales están documentados en los anexos 1 y 2.

Una vez definidos los catálogos con los perfiles seleccionados, estos se vinculan al componente “Cross Section Selector” de “Karamba 3D”, que permite elegir la sección transversal mediante un “Number Slider”. Se ha implementado este proceso para cada tipo de elemento. Finalmente, las secciones transversales se conectan al componente “LineToBeam” correspondiente para asignar las propiedades estructurales a los elementos.

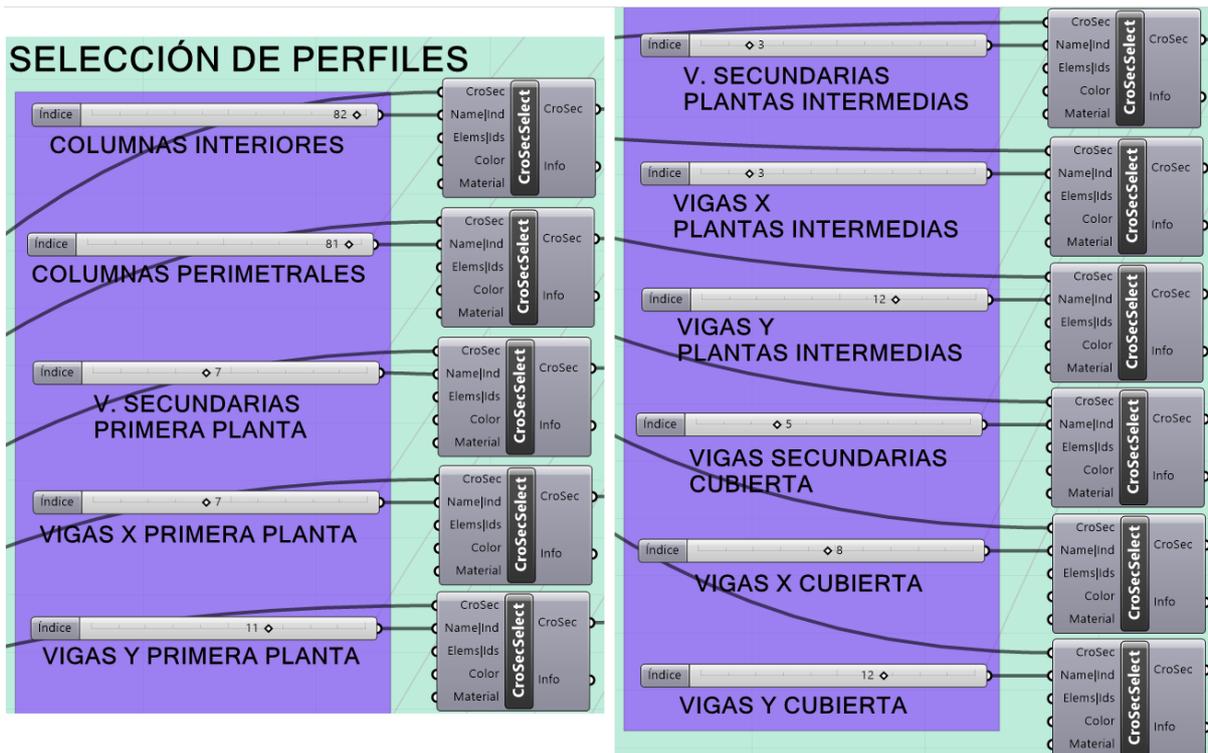


Figura 48 Componente “Cross Section Selector” para seleccionar una sección transversal

3.3 Análisis estructural

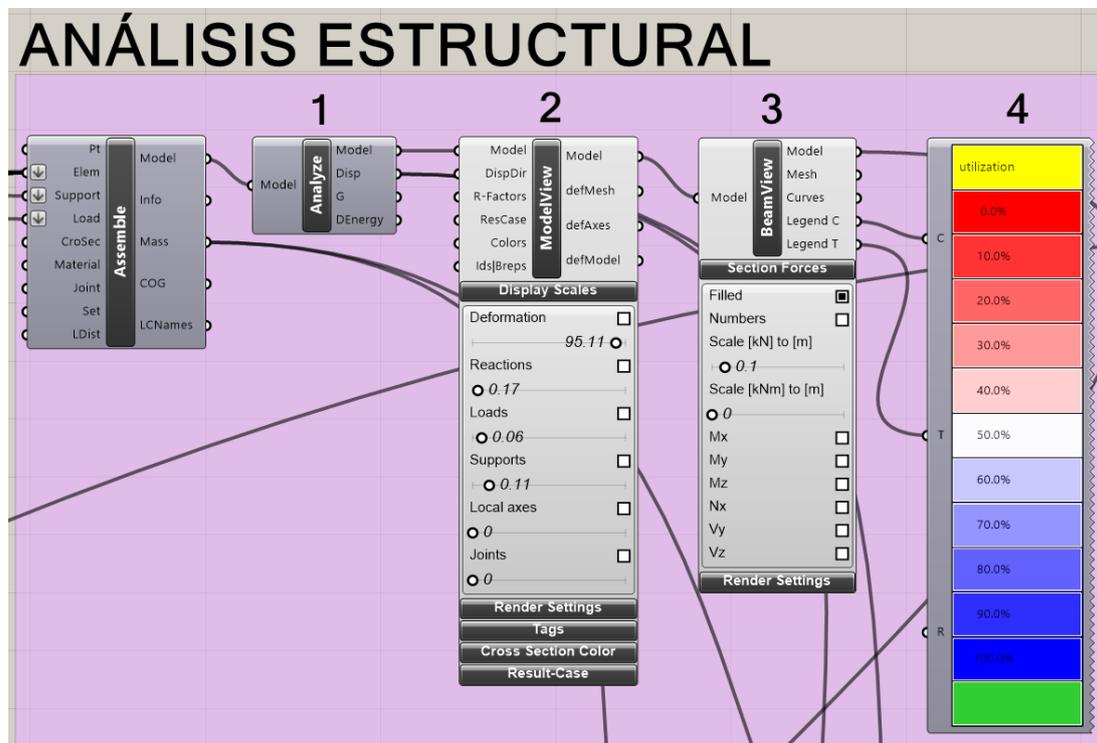


Figura 49 Bloque para realizar el análisis estructural del modelo

Una vez ensamblado el modelo, y con todos los datos necesarios para su análisis mediante el método de elementos finitos, se procesa a realizar el análisis estructural y la visualización del modelo utilizando las herramientas de “Karamba 3D”. A continuación, se detallan los componentes utilizados en este proceso:

1. **Analyze:** Este componente calcula los desplazamientos de la estructura utilizando la teoría de primer orden para deflexiones pequeñas. De este análisis es posible extraer el dato del máximo desplazamiento de la estructura, lo cual es crucial para evaluar su comportamiento bajo carga.
2. **ModelView:** Este componente permite visualizar el estado actual del modelo en la ventana de “Rhino”, mostrando los datos de deformaciones, reacciones, y cargas. Además, proporciona opciones para activar etiquetas que identifican propiedades específicas de cada elemento, así como la configuración de la renderización, lo que facilita una interpretación del estado actual del modelo.
3. **BeamView:** Con este componente se configuran las propiedades de visualización específicas para vigas y cerchas. Esta herramienta permite visualizar los diagramas de esfuerzo axial, cortante y momentos de cada elemento, proporcionando un análisis gráfico detallado de las fuerzas internas de la estructura.
4. **Legend:** Esta herramienta genera una leyenda con etiquetas y colores que se asocia visualmente al modelo en la ventana de “Rhino”. En este caso, la leyenda ayuda a interpretar los valores de esfuerzo axial, radio de utilización y los desplazamientos, facilitando una evaluación rápida y precisa de las áreas de mayor demanda estructural.

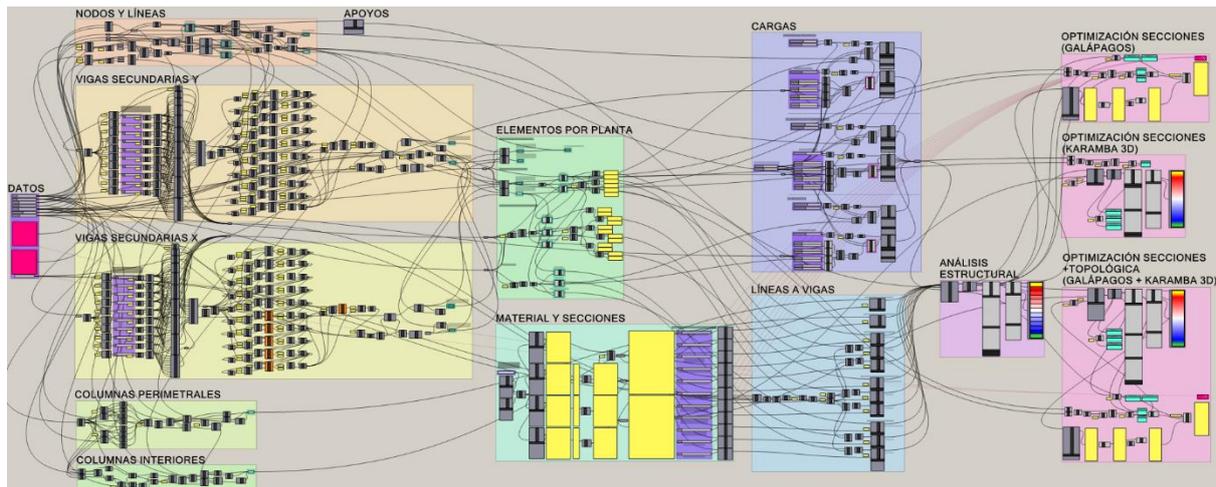


Figura 50 Vista general del algoritmo completo

3.4 Parámetros de modelación

Para demostrar el funcionamiento del algoritmo, se ha decidido aplicarlo a un caso de estudio no real que representa un edificio multifuncional pequeño, con distintos usos en cada planta. Se han colocado apoyos de empotramiento en las bases de las columnas para simular las uniones con la cimentación.

3.4.1 Descripción de los datos geométricos

- Área de implantación: 375 m²
- Longitud en “X”: 15 metros
- Longitud en “Y”: 25 metros
- Número de ejes en “X”: 3
- Número de ejes en “Y”: 5
- Altura del entrespacio de la planta baja: 3.5 metros
- Altura de los pisos superiores: 3.1 metros
- Número de plantas: 4
- Inicialmente se han dispuesto los ejes de manera simétrica. En la tabla 2 se muestra la posición de los ejes:

Tabla 2.

| ID | Ordenada en X (m) | ID | Ordenada en Y (m) |
|----|-------------------|----|-------------------|
| A | 0.0 | 1 | 0.0 |
| B | 7.5 | 2 | 6.3 |
| C | 15.0 | 3 | 12.5 |
| | | 4 | 18.7 |
| | | 5 | 25.0 |

Tabla 2 Posición de los ejes de la estructura

- En cada paño se han dispuesto, de manera inicial, 7 vigas secundarias en dirección “Y” con una separación de 0.94 metros.

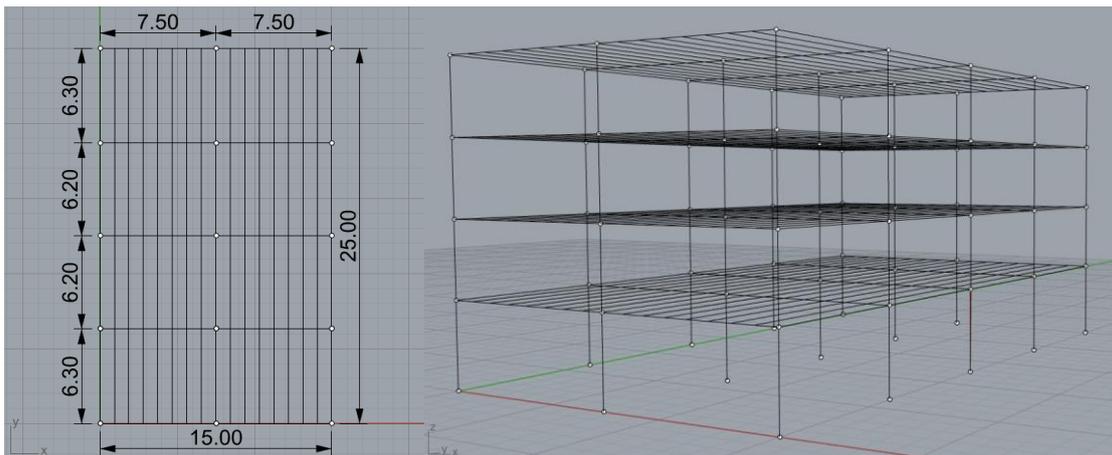


Figura 51 Visualización de la geometría de la estructura

3.4.2 Materiales

Para el diseño de los elementos estructurales en el caso de estudio, se ha utilizado acero A36 ya que es el material más usado en este tipo de proyectos.

3.4.3 Análisis de cargas

La aplicación de cargas a la estructura se ha basado en los valores de cargas muertas y vivas indicados en el capítulo 1 de la Norma Ecuatoriana de la Construcción (NEC-SE-CG).

3.4.3.1 Carga muerta

Los valores de la carga muerta aplicados corresponden a los comúnmente utilizados en el prediseño de estructuras de acero. En cada planta se ha considerado una carga muerta de 3.2 Kn/m², y para la cubierta, un valor de 2 Kn/m². Estos valores incluyen el peso aproximado de las vigas principales y secundarias, la losa, la tabiquería y los acabados.

3.4.3.2 Carga viva

La carga viva depende del uso de cada planta. En este caso, se ha considerado un uso residencial en todo el edificio, con una carga viva de 2 Kn/m² para los tres primeros pisos. Para la cubierta, que no es accesible, pero se ha considerado como área de paseo, se ha asignado una carga viva de 3 Kn/m².

3.4.3.3 Sobrecargas

Se implementarán sobrecargas en áreas específicas de cada planta, en respuesta a los diferentes usos dentro de cada planta:

- En la primera planta, una pequeña piscina de 1 metro de profundidad ocupa un área de 37.5 m², con una sobrecarga de 10 kN/m², considerando el peso del agua.
- En los pisos segundo y tercero, se ha asignado un área de 20 m² para estanterías, con una sobrecarga de 7.2 kN/m².
- En la cubierta, se dispone de un espacio de jardinería con un área irregular de 37.3 m² y una sobrecarga de 4.8 kN/m².

Para evitar la superposición de cargas con diferentes usos, a cada sobrecarga se le resta el valor de la carga viva correspondiente a la planta. Se ha aplicado una combinación de carga de servicio (D + L) para evaluar principalmente las deflexiones de la estructura.

A continuación, se muestra la disposición de las sobrecargas en cada planta:

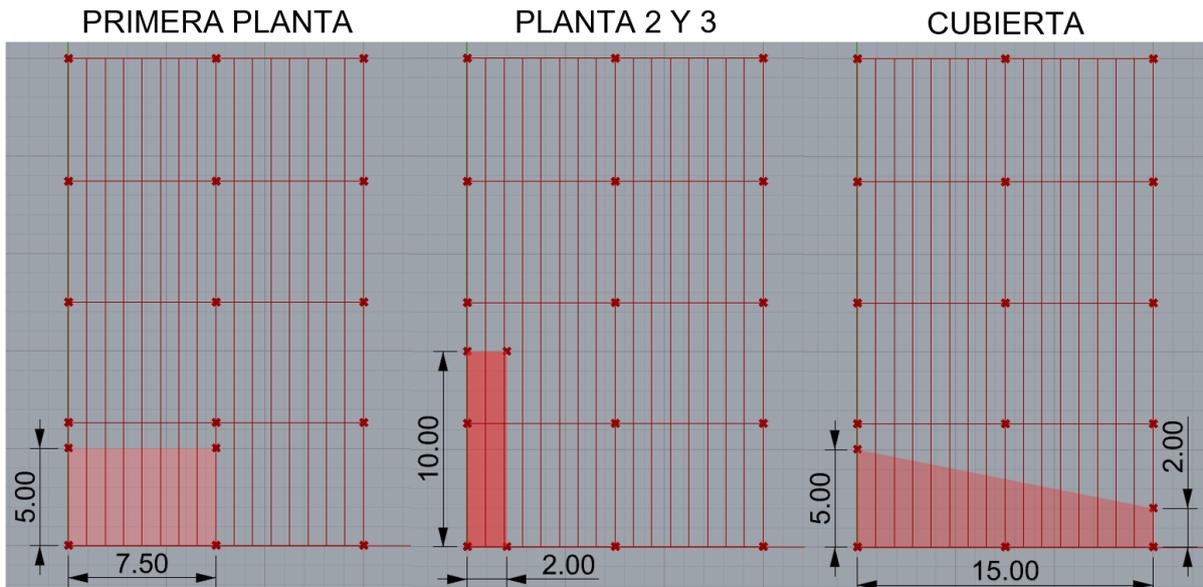


Figura 52 Disposición de las sobrecargas en las plantas de la estructura

A continuación, en la tabla 3, se muestra el resumen de las cargas implementadas en el caso de estudio.

Tabla 3.

| | Carga muerta (Kn/m ²) | Carga viva (Kn/m ²) | Sobrecarga (Kn/m ²) |
|-----------------|---|---|---|
| Planta 1 | 3.2 | 2.0 | 10 |
| Planta 2 | 3.2 | 2.0 | 7.2 |
| Planta 3 | 3.2 | 2.0 | 7.2 |
| Cubierta | 2.0 | 3.0 | 4.8 |

Tabla 3 Tabla resumen de cargas aplicadas

3.4.4 Secciones transversales

En los anexos 1 y 2 se presentan los catálogos que el algoritmo puede utilizar para optimizar la estructura. Inicialmente, se ha establecido que las columnas estarán formadas por perfiles SHS de dimensiones 300x300x8 mm, mientras que las vigas principales y secundarias estarán hechas con perfiles IPE de 600 mm.

3.5 Caso 1

Antes de realizar las optimizaciones, se registraron los datos de la estructura sin optimizar, utilizando las secciones más grandes. El valor de la masa se obtuvo del componente “Assemble Model”, y el valor del desplazamiento máximo inicial se determinó mediante el componente “Analyze” de Karamba3D.

Los valores iniciales obtenidos fueron los siguientes:

- Masa: 258.867 Ton
- Deflexión máxima: 0.695 cm



Figura 54 Estructura inicial sin optimizar

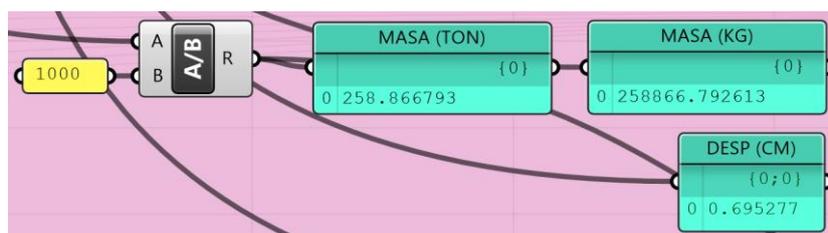


Figura 53 Masa y deflexión máxima obtenidas de la estructura sin optimizar

3.5.1 Optimización estructural de forma y tamaño con el algoritmo genético

Para la primera optimización, se utilizó el algoritmo genético “Galapagos”, cuyo objetivo, en este caso, es optimizar la forma de los perfiles, es decir, variar el tipo o familia de

perfil; y optimizar el tamaño, refiriéndonos a variar las dimensiones de la sección disponibles en cada familia.

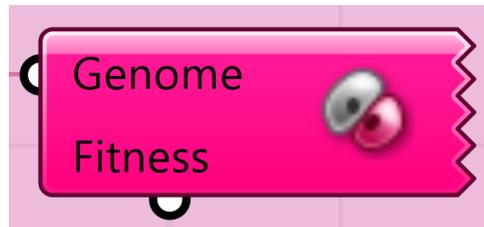


Figura 55 Componente "Galapagos"

El componente Galapagos cuenta con dos parámetros de entrada:

- **Genome:** Incluye todos aquellos parámetros que se modificarán para alcanzar el objetivo, actuando como los "genes" en los procesos evolutivos de la naturaleza. En este caso, se conectaron a "Genome" los "Number Sliders", que seleccionan las secciones transversales para cada tipo de elemento, resultando en un total de 8 genes.

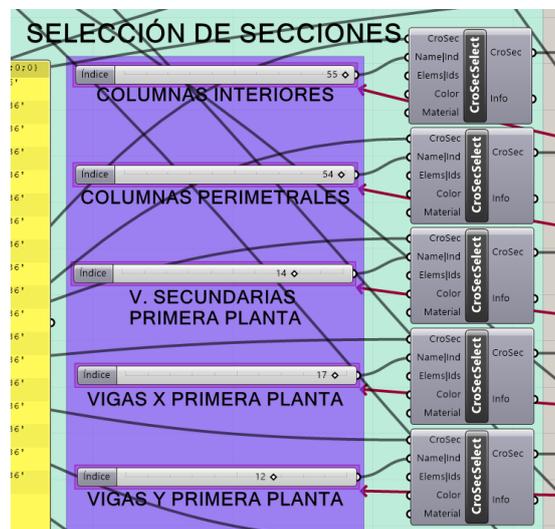


Figura 56 "Number Sliders" conectados al parámetro "Genome"

- **Fitness:** Se refiere al objetivo a alcanzar mediante la variación del genoma. En este caso, el objetivo es obtener la estructura más ligera posible, para reducir los costos, pero que los elementos cumplan con los criterios estructurales

establecidos, como la deflexión máxima aceptable y que no sobrepasen el 90% de utilización.

La función que responde al objetivo del caso de estudio se definió de la siguiente forma:

$$f = m + d + Abs(r - 1)$$

Donde:

m = Masa

d = Desplazamiento máximo

r = Radios de utilización

Esta función toma en cuenta la suma de la masa total de la estructura, el desplazamiento máximo y el radio de utilización de cada elemento. El algoritmo genético buscará minimizar este valor. Para implementar la función, se utilizó el componente "Evaluate", en el cual se ingresaron las variables necesarias de la función.

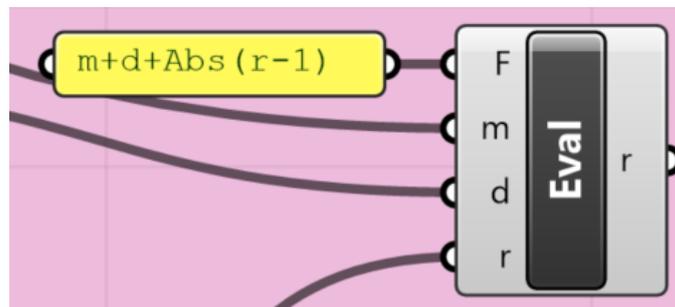


Figura 57 Componente "Evaluate" para crear la función objetivo

Al componente, se ingresó el valor de la masa en toneladas, tomado del componente "Assemble Model".

En el caso del desplazamiento, se debe garantizar que este se encuentre dentro del límite aceptable, el cual se ha definido con el criterio estructural que establece que el desplazamiento máximo debe ser menor a la longitud del elemento sobre 300.

$$\Delta < \frac{L}{300}$$

Para cumplir con este criterio, se empleó el componente de Visual Basic “VB Script” para generar un código que identifique aquellos desplazamientos que superen el límite aceptable para reemplazarlos con un valor de 10000, de modo que el valor del “Fitness” sea muy grande, y el algoritmo no considere esa solución como óptima.

Al componente de Visual Basic se ha ingresado los parámetros de desplazamiento máximo obtenido del componente “Analyze” de Karamba3D y la longitud de la viga más grande. El código utilizado para limitar el desplazamiento máximo está disponible en el Anexo 3.

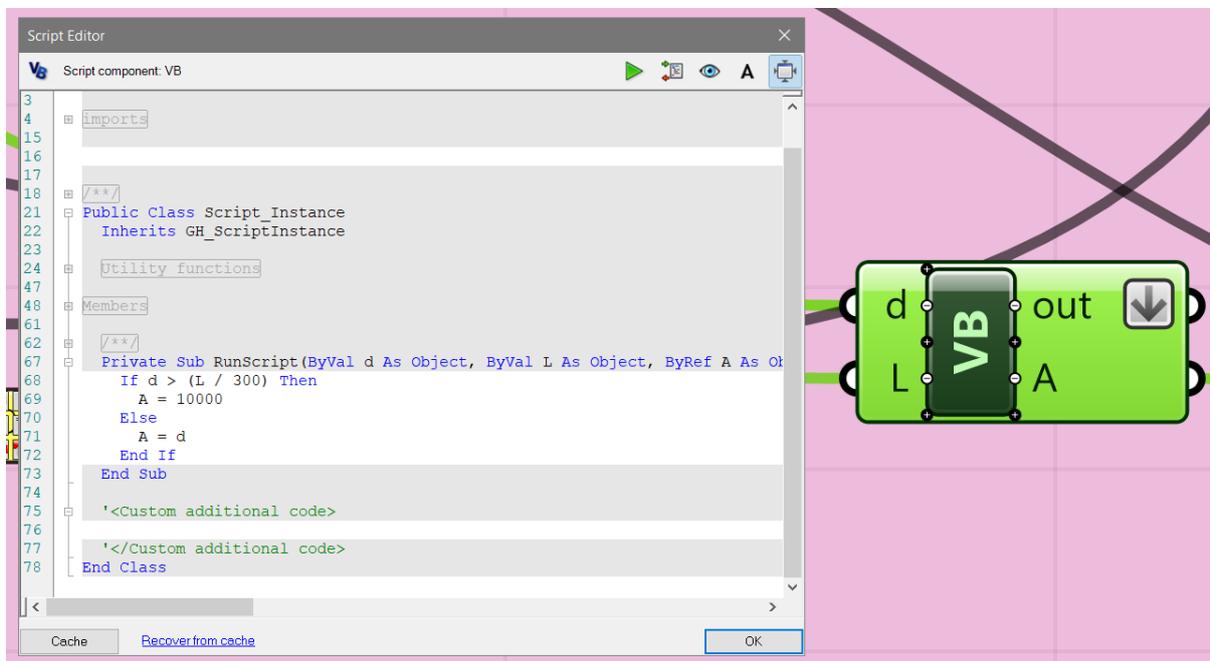


Figura 58 Componente "VB Script" con el código para filtrar el desplazamiento máximo

El último parámetro que se conectó a la función es el radio de optimización, para esto se utilizó el componente “Utilization” de Karamba3D, el cual entrega la lista del radio de utilización de cada elemento. Al igual que con el desplazamiento, es necesario realizar un proceso que restrinja, en este caso, aquellos valores que sean mayores a 0.9, reemplazándolos con un valor de 1000, de esta manera se asegura que los elementos no estén cerca del límite de

fallo. Como dato de entrada del componente, se ingresa la lista de los radios de utilización. El código de este proceso se encuentra en el anexo 4.

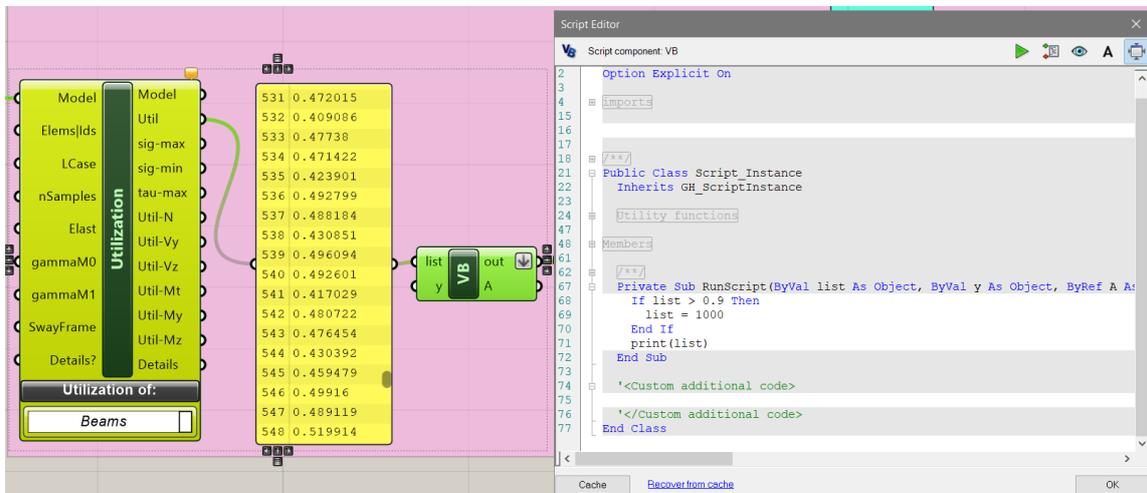


Figura 59 Componente "Utilization" y componente VB "Script" para filtrar los radios de optimización

El proceso continúa con otro componente de Visual Basic, en el cual se conectará como dato de entrada, la lista de radios de utilización modificada anteriormente y la longitud de dicha lista. El código identificará si existe algún valor igual a 1000, si es así, entonces convierte todos los valores de la lista en 10000. Esto se realiza con el objetivo de que el "fitness" de la función no identifique valores más pequeños a pesar de que algún elemento no cumple con la utilización, de lo contrario el algoritmo de optimización tomaría como valido soluciones que podrían fallar. El código de este proceso se encuentra en el anexo 5.

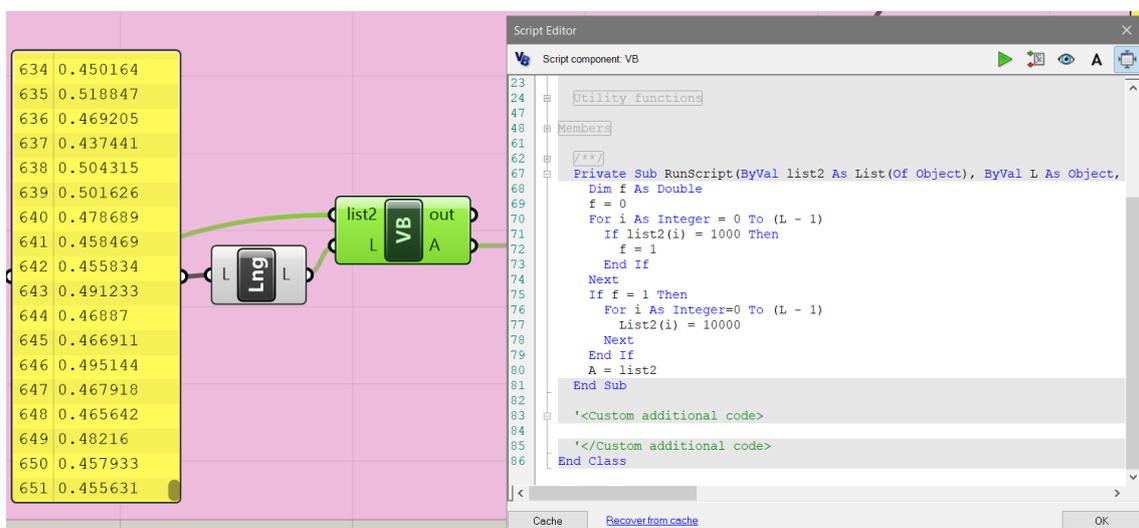


Figura 60 Componente "VB Script" para convertir todos los datos a 10000

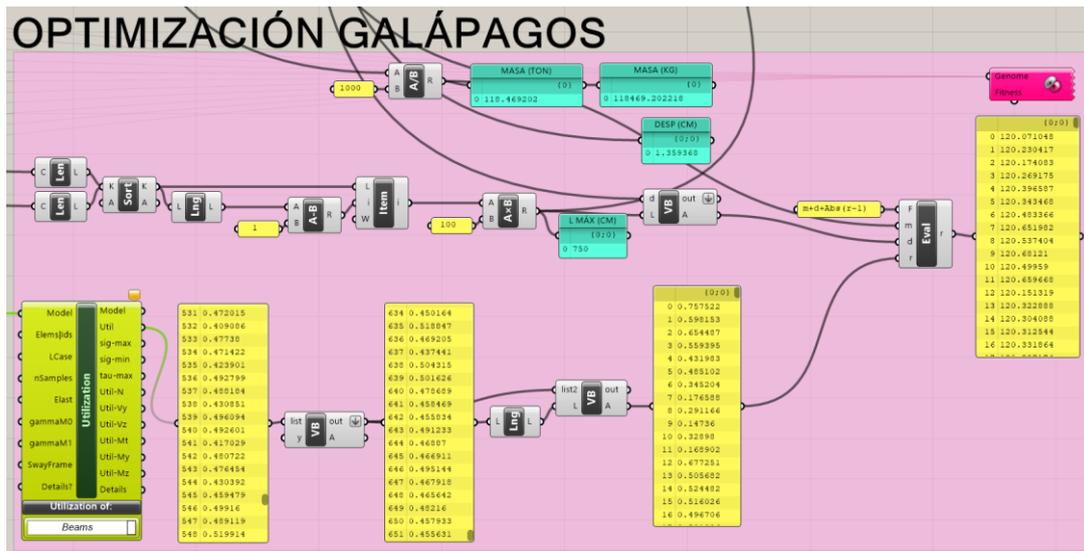


Figura 61 Bloque para realizar la optimización estructural con el algoritmo genético “Galapagos”

Con la función objetivo conectada al parámetro “Fitness” y las variables seleccionadas para el “Genome”, se inicia el proceso de optimización en Galapagos. La ventana de Galápagos cuenta con una sección de opciones que permite configurar los parámetros de optimización, a continuación, se describen los más importantes, y se detalla los valores establecidos para el caso de estudio:

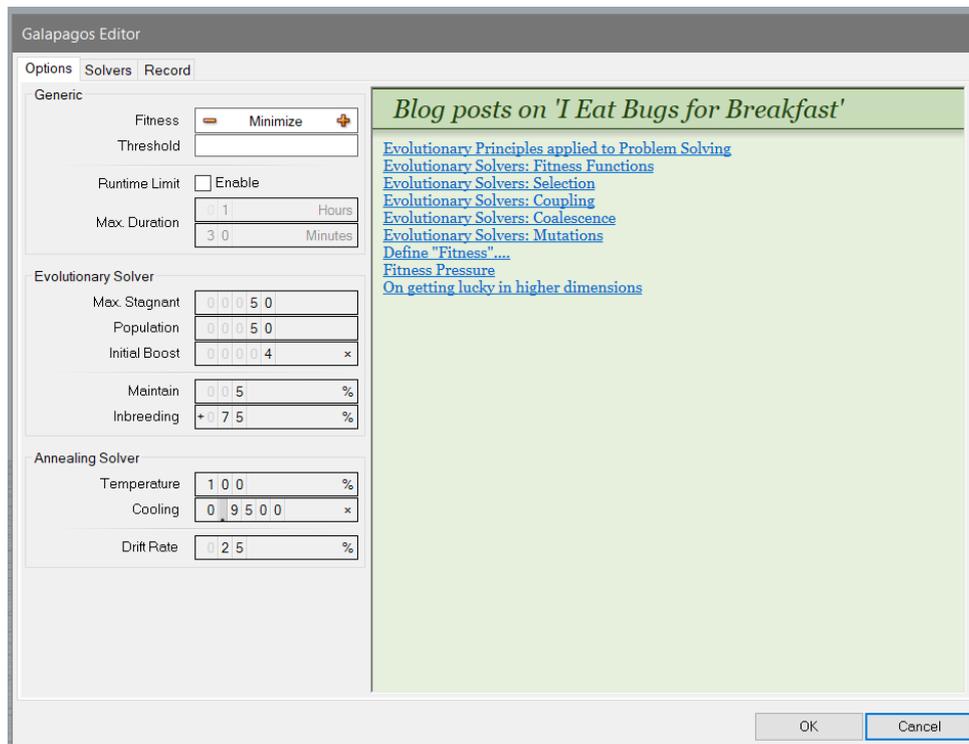


Figura 62 Ventana para configurar las opciones de Galapagos

- **Fitness:** Esta configuración es la más importante, es necesario especificar si Galapagos debe optimizar el Fitness establecido por el usuario hacia el máximo o hacia el mínimo. En este caso se seleccionó “Minimize”, ya que lo que se busca reducir el peso y los desplazamientos máximos de la estructura.
- **Treshold:** Se refiere al valor límite que determinará cuando el algoritmo debe parar. Si la mejora en la función objetivo entre iteraciones consecutivas es menor que el umbral definido, el algoritmo detiene su ejecución al asumir que se ha alcanzado una solución óptima. En este caso, no se utilizó esta opción, ya que esto podría impedir alcanzar una solución óptima.
- **Runtime Limit:** Define el tiempo máximo para que el algoritmo de optimización busque soluciones antes de detenerse. Esta opción es bastante útil para evitar que el programa consuma recursos indefinidamente, especialmente en problemas complejos, sin embargo, en este caso se deshabilitó esta opción, para poder analizar los tiempos de optimización
- **Máx Stagant:** Galapagos organiza las soluciones en etapas o generaciones. Este parámetro define el número máximo de generaciones consecutivas en las que no se observa una mejora significativa en el valor de la función objetivo antes de detener el proceso de optimización. Se dejó el valor predeterminado de 50, un valor suficientemente alto para evitar que el algoritmo se detenga prematuramente.
- **Population:** Define el número de resultados que se obtendrán en cada generación. En este caso, se estableció un valor de 50, el cual es el valor predeterminado.
- **Inicial Boost:** Este valor es el factor que multiplicará al parámetro “Population” cuyo producto definirá el número de resultados que tendrá la primera

generación. Es importante determinar la cantidad de resultados de la generación inicial ya que esto aumentará la posibilidad de encontrar soluciones óptimas de inicio, las cuales serán evolucionadas en las siguientes generaciones. En este caso, se cambió de 2 a 4 para aumentar la efectividad del algoritmo, lo que resulta en 200 resultados para la primera generación.

- **Maintain:** Este valor establece el porcentaje de resultados o individuos de una generación que se transferirán a la siguiente. En este caso, se ha asignado un 5%, por lo tanto, los 10 mejores resultados de los 200 iniciales serán transferidos a la siguiente generación para que puedan seguir evolucionando.
- **Inbreeding:** Este valor se suma al valor asignado a “Maintain”, cuyo resultado define el porcentaje de resultados que serán evoluciones o mejoras de los resultados de las generaciones anteriores. En este caso, se estableció en 75%, lo que significa que el algoritmo dedicará el 80% de los resultados a evolucionar las mejores soluciones, y el otro 20% lo dedicará a encontrar soluciones nuevas diferentes.

3.5.2 Resultados de la primera optimización

El solucionador Galapagos se ejecutó durante aproximadamente 16 horas, alcanzando 114 generaciones de resultados. La solución más óptima se logró en la generación 87. El tiempo de optimización fue extenso debido a la complejidad del algoritmo y del caso de estudio, además de depender de la capacidad de procesamiento de la computadora empleada.

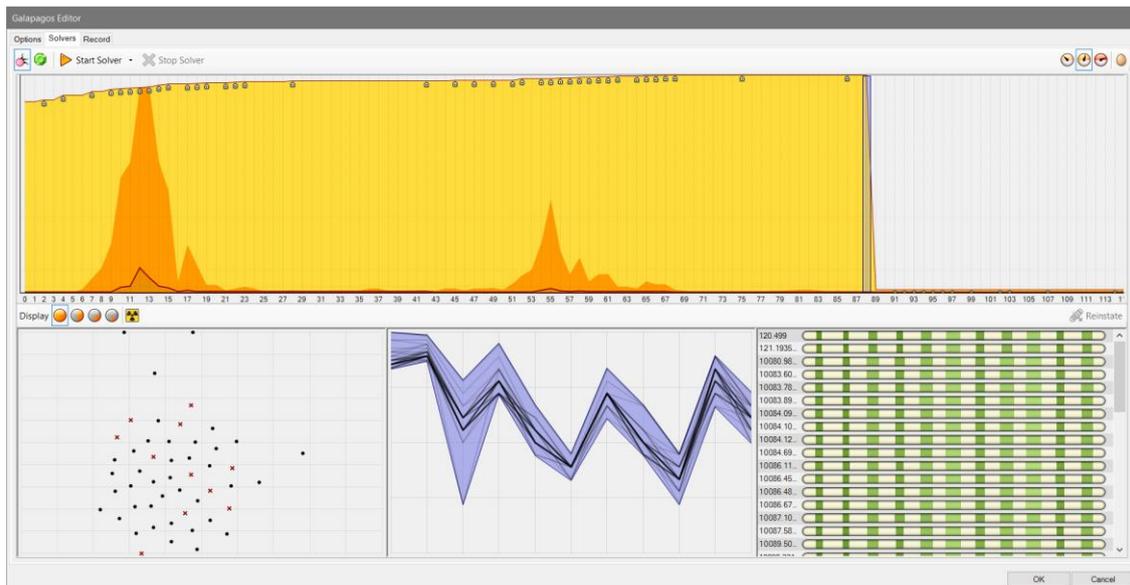


Figura 63 Ventana del solucionador de Galapagos para la primera optimización

Las secciones seleccionadas por el algoritmo de optimización fueron las siguientes:

- Columnas interiores: SHS 300x300x8
- Columnas perimetrales: SHS 300x300x6
- Vigas secundarias primera planta: IPE 450
- Vigas en X primera planta: IPE 600
- Vigas en Y primera planta: IPE 360
- Vigas secundarias plantas intermedias: IPE 240
- Vigas en X plantas intermedias: IPE 500
- Vigas en Y plantas intermedias: IPE 360
- Vigas secundarias cubierta: IPE 240
- Viga en X cubierta: IPE 550
- Vigas en Y cubierta: IPE 360

Se obtuvieron los siguientes resultados:

- Masa: 118.469 Ton
- Deflexión máxima: 1.359 cm



Figura 64 Estructura con la primera optimización

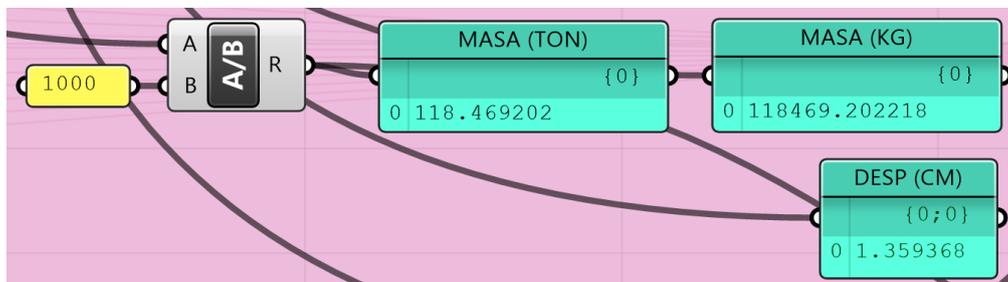


Figura 65 Masa y desplazamiento máximo obtenidos de la estructura con la primera optimización

3.6 Caso 2

3.6.1 Optimización estructural de forma y tamaño con el optimizador de Karamba 3D

Para el segundo caso, de igual manera se optimizará la forma y el tamaño de las secciones transversales de los elementos estructurales, pero a diferencia del primer caso, se utilizará el optimizador estructural de Karamba3D mediante el componente “Optimize Cross Section”. Este componente está diseñado para la selección automatizada de secciones transversales óptimas, considerando la capacidad de carga de las secciones transversales, y la deflexión máxima permitida de la estructura acorde a la normativa EC3 (EN 1993- 1- 1); para estructuras de acero, se aplica el procedimiento del anexo B de dicha normativa.

Como parámetro de entrada del componente, se ingresó el modelo ensamblado y la lista de secciones transversales disponibles para la optimización, mismo utilizado en el Caso 1.

La entrada “Max Util” tiene como valor predeterminado 1.0 pero para este caso se ha establecido un valor de 0.8 de radio de utilización. Inicialmente la idea para todos los casos es limitar la utilización a 0.9, pero en este caso fue necesario reducirlo, ya que el proceso iterativo de “Optimize Cross Section” no siempre garantiza la convergencia y a pesar de limitar este parámetro, algunos elementos suelen presentar sobreutilización. Se identificó que al colocar el valor de 0.8, algunos elementos presentaban un radio de utilización mayor a este, pero menor a 0.9, lo que fue conveniente para este caso.

Además, para que el sistema estructural se vuelva más estáticamente determinado, se cambió el número de iteraciones usadas para el estado límite de utilización, establecido predeterminadamente en 5, a un valor de 100, este parámetro se llama “Ultimate Limit State Iterations (ULSIter)” y se encuentra en la parte de configuraciones del componente.

El software ejecuta los siguientes pasos para el diseño de máxima utilización:

- Calcular las fuerzas seccionales a lo largo de todos los elementos, usando las secciones transversales iniciales (Las que se encuentren primero en la lista).
- Para cada elemento, el algoritmo busca y selecciona la primera sección que cumpla con el valor de la utilización máxima.
- Si no se requieren modificaciones o se ha alcanzado el número máximo de iteraciones, el algoritmo se detiene o se repite el paso 1 con las secciones transversales recién seleccionadas.

Por otro lado, se limitó la deflexión siguiendo el criterio de $L/300$, el mismo implementado en el caso 1. Es importante ingresar este parámetro “MaxDisp” ya que, de manera predeterminada, el componente no considera la deflexión máxima al optimizar.

El proceso iterativo que sigue el software para limitar las deflexiones comprende los siguientes pasos:

- Determinación de las deflexiones máximas: Abarcando los desplazamientos nodales y desplazamientos de los elementos en la posición inicial, intermedia y final.
- Adaptación mediante casos de cargas virtuales: Se genera y se calcula internamente un caso de carga virtual, el cual guía el cambio de tamaño de las secciones transversales de los elementos para cumplir con las restricciones definidas.
- Ajuste de las secciones transversales: El cambio de la sección transversal de los elementos se ejecuta dentro de un número predefinido de iteraciones especificado en “Displter”, en este caso, se ha dejado el valor en 5.

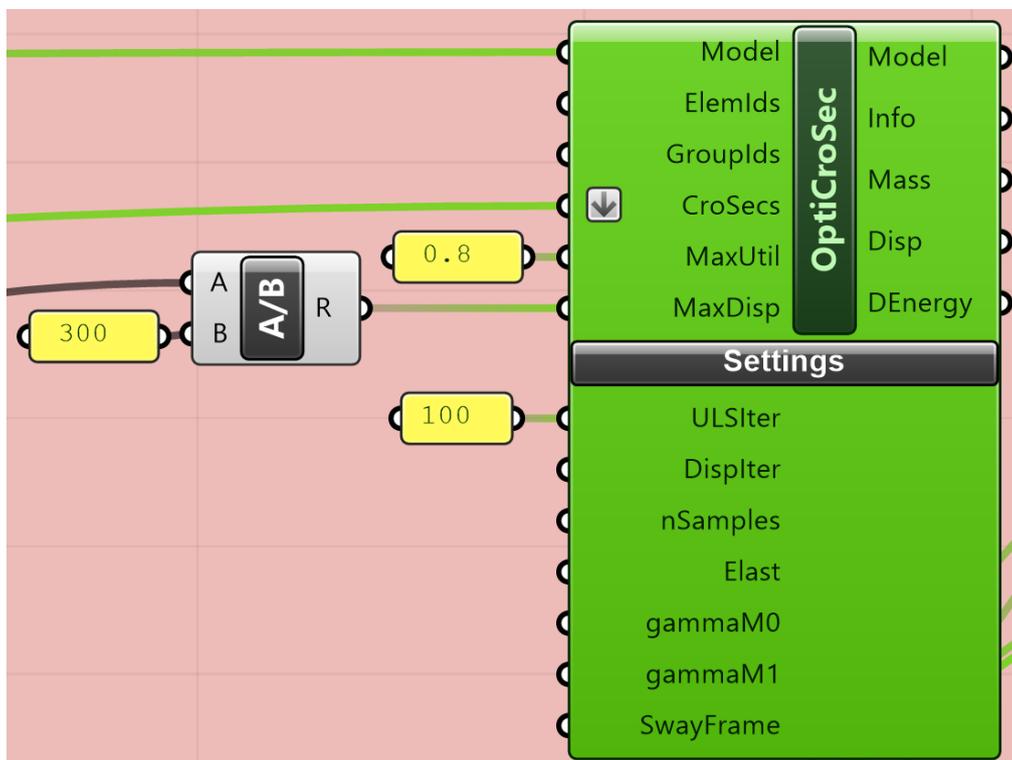


Figura 66 Componente “Optimize Cross Section”

Después del componente “Optimize Cross Section” se deben conectar los componentes de análisis estructural ya mencionados anteriormente para visualizar el modelo.

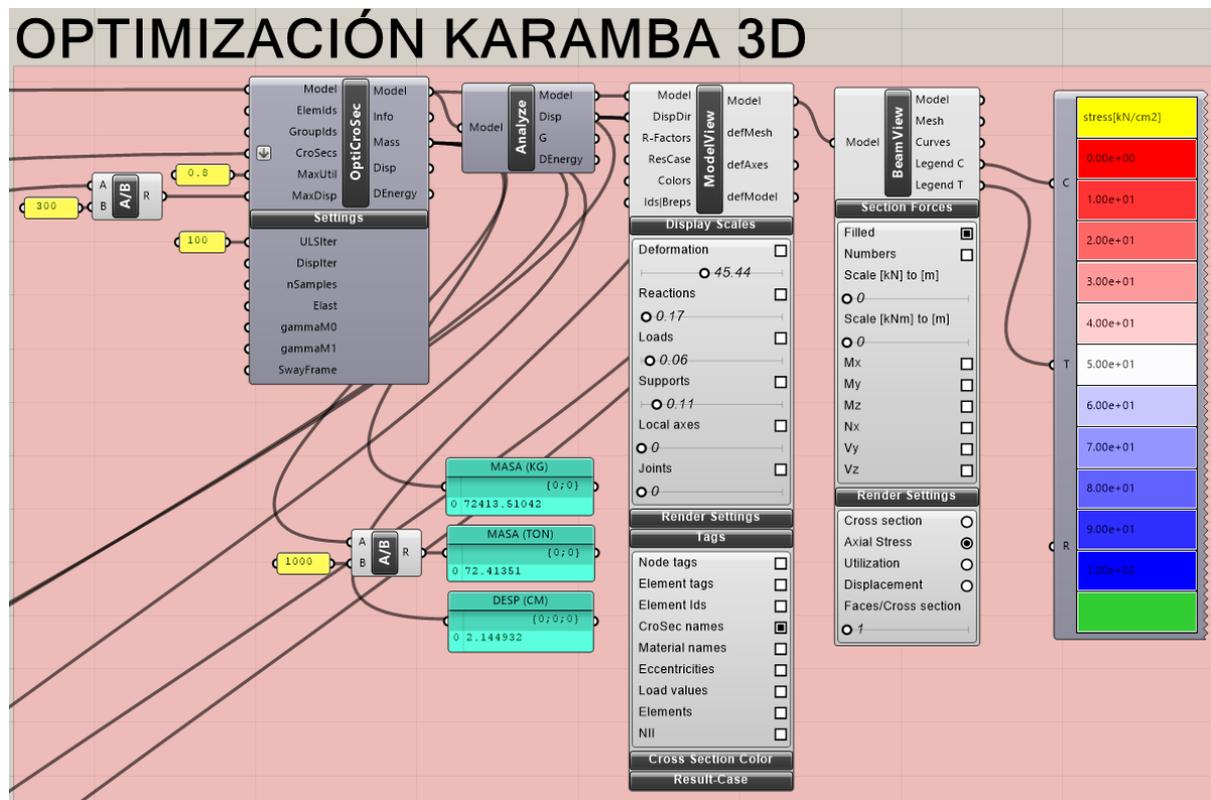


Figura 67 Bloque para la optimización en Karamba 3D

3.6.2 Resultados de la segunda optimización

La herramienta de optimización de Karamba 3D no permite distinguir entre diferentes tipos de elementos estructurales, por lo tanto, la solución es una estructura cuyos elementos individuales, tendrán todos diferentes secciones transversales a pesar de estar dentro del mismo tipo de elemento. Por ejemplo, cada viga secundaria dentro de un mismo paño tendrá secciones transversales diferentes. Incluso, cada viga principal en “X”, en este caso, tendrá una sección distinta en cada segmento dividido por las vigas secundarias. Este resultado no es práctico desde un punto de vista constructivo.

El optimizador de “Karamba 3D” tardó aproximadamente 5 segundos en optimizar la estructura.

En la figura 68 se ejemplifica como se optimizan las secciones transversales con el optimizador de “Karamba3D”:

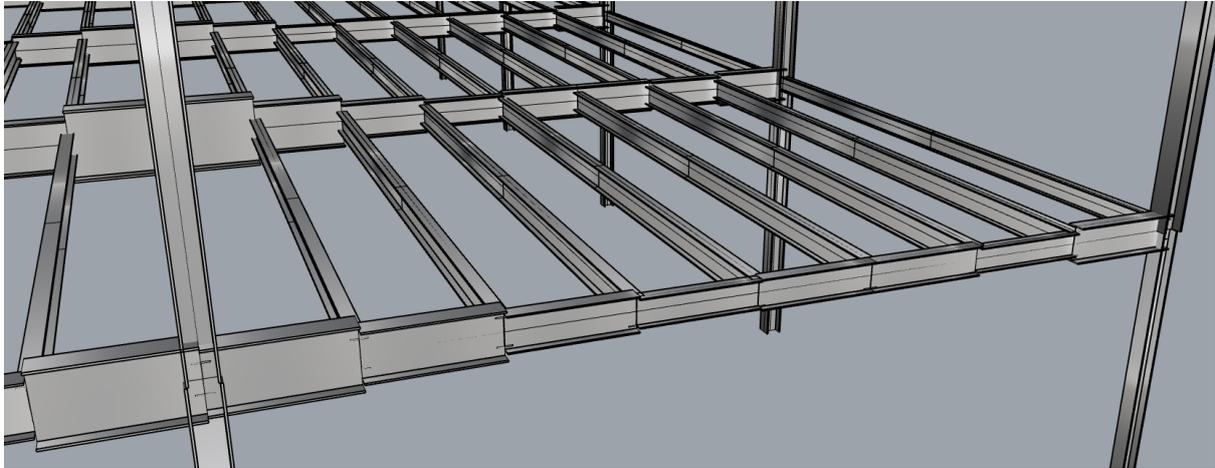


Figura 68 Ejemplo de elementos optimizados con Karamba3D

Se obtuvieron los siguientes resultados:

- Masa: 72.414 Ton
- Deflexión máxima: 2.149 cm



Figura 69 Estructura con la segunda optimización

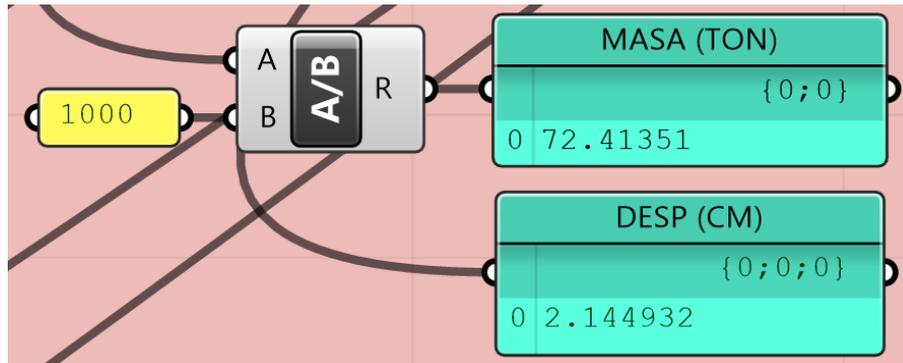


Figura 70 Masa y desplazamiento máximo obtenidos de la estructura con la segunda optimización

3.6.3 Racionalización del caso 2

El resultado obtenido en el caso 2 no es factible de construir debido a la alta cantidad de secciones transversales diferentes. Por ello, se debe racionalizar la solución, es decir, partir de la estructura optimizada y simplificarla manualmente para reducir la cantidad de secciones transversales. Se agruparán ciertos tipos de elementos, se tomará la sección más grande de estos y se la aplicará a todo este grupo. Este proceso sigue criterios constructivos y estructurales.

3.6.3.1 Resultados de la racionalización del caso 2

Se asignó una sola sección transversal para cada una de las 15 columnas, ya que anteriormente cada columna tenía diferentes secciones en cada segmento de entrepiso.

Para las vigas principales se unificaron las secciones transversales para las vigas ubicadas sobre un mismo eje, tanto en las direcciones "X" como "Y". Anteriormente, cada segmento dividido por vigas transversales o secundarias tenía una sección distinta. Se diferencian también las secciones entre plantas.

En cuanto a las vigas secundarias, se estableció para todas las vigas secundarias entre los ejes "A" y "B" una sola sección transversal y para las vigas secundarias entre los ejes "B" y "C" se establece otra sección transversal, diferenciadas igualmente en cada planta.

A continuación, se muestran las secciones transversales establecidas para cada elemento:

| Columnas | Sección | Vigas principales | Sección | Vigas secundarias | Sección |
|----------|---------|-------------------|---------|-------------------|---------|
| 1-A | HEB 200 | Planta 1 | | Planta 1 | |
| 1-B | HEB 220 | Eje 1 | IPE 500 | Paño A-B | IPE 270 |
| 1-C | HEB 180 | Eje 2 | IPE 600 | Paño B-C | IPE 220 |
| 2-A | HEB 240 | Eje 3 | IPE 550 | Planta 2 | |
| 2-B | HEB 280 | Eje 4 | IPE 600 | Paño A-B | IPE 240 |
| 2-C | HEB 200 | Eje 5 | IPE 400 | Paño B-C | IPE 220 |
| 3-A | HEB 200 | Eje A | IPE 200 | Planta 3 | |
| 3-B | HEB 220 | Eje B | IPE 220 | Paño A-B | IPE 240 |
| 3-C | HEB 200 | Eje C | IPE 200 | Paño B-C | IPE 220 |
| 4-A | HEB 200 | Planta 2 | | Cubierta | |
| 4-B | HEB 240 | Eje 1 | IPE 450 | Paño A-B | IPE 220 |
| 4-C | HEB 220 | Eje 2 | IPE 600 | Paño B-C | IPE 220 |
| 5-A | HEB 180 | Eje 3 | IPE 600 | | |
| 5-B | HEB 180 | Eje 4 | IPE 600 | | |
| 5-C | HEB 180 | Eje 5 | IPE 450 | | |
| | | Eje A | IPE 200 | | |
| | | Eje B | IPE 220 | | |
| | | Eje C | IPE 180 | | |
| | | Planta 3 | | | |
| | | Eje 1 | IPE 400 | | |
| | | Eje 2 | IPE 600 | | |
| | | Eje 3 | IPE 600 | | |
| | | Eje 4 | IPE 600 | | |
| | | Eje 5 | IPE 400 | | |
| | | Eje A | IPE 180 | | |
| | | Eje B | IPE 220 | | |
| | | Eje C | IPE 180 | | |
| | | Cubierta | | | |
| | | Eje 1 | IPE 450 | | |
| | | Eje 2 | IPE 600 | | |
| | | Eje 3 | IPE 600 | | |
| | | Eje 4 | IPE 600 | | |
| | | Eje 5 | IPE 450 | | |
| | | Eje A | IPE 180 | | |
| | | Eje B | IPE 220 | | |
| | | Eje C | IPE 180 | | |

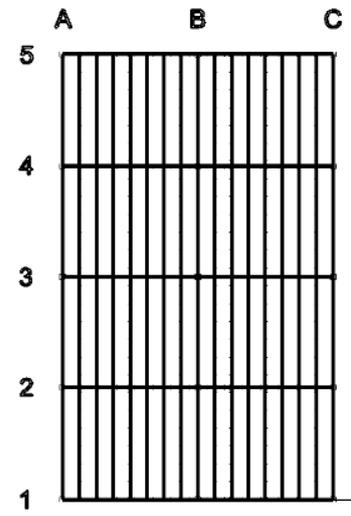


Figura 71 Secciones transversales establecidas para el caso 2 racionalizado

Se obtuvieron los siguientes resultados:

- Masa: 90.1 Ton
- Deflexión: 1.372 cm

El cálculo de la masa se realizó de manera manual y su desarrollo se evidencia en el anexo 6, 7 y 8.

Tabla 4.

| Elemento | Masa (Ton) |
|-------------------|-------------------|
| Columnas | 12.74 |
| Vigas principales | 37.37 |
| Vigas secundarias | 39.99 |
| | 90.10 |

Tabla 4 Masa de la estructura con la segunda optimización racionalizada

3.7 Caso 3

3.7.1 Optimización topológica

Para este caso se optimizaron tanto la forma como el tamaño de las secciones transversales de la estructura con la herramienta de Karamba 3D “Optimize Cross Section”. Adicionalmente, se realizó una optimización topológica mediante el algoritmo genético “Galapagos”, lo que realizará el algoritmo es modificar las listas de posiciones de los ejes en “X y en “Y” para encontrar la disposición óptima de las columnas cumpliendo con la función objetivo del algoritmo planteado en el caso 1.

En primer lugar, se debe utilizar el algoritmo de optimización de “Karamba 3D” mostrado en la figura 67, para posteriormente conectar los resultados de masa, deflexión máxima, y radios de optimización de este algoritmo, al algoritmo de “Galapagos” mostrado en la figura 61.

En el parámetro de entrada “Genome” se ingresó las dos listas de posiciones de los ejes, en “X” y en “Y” y al parámetro fitness se ingresó la misma función objetivo propuesta en el caso 1.

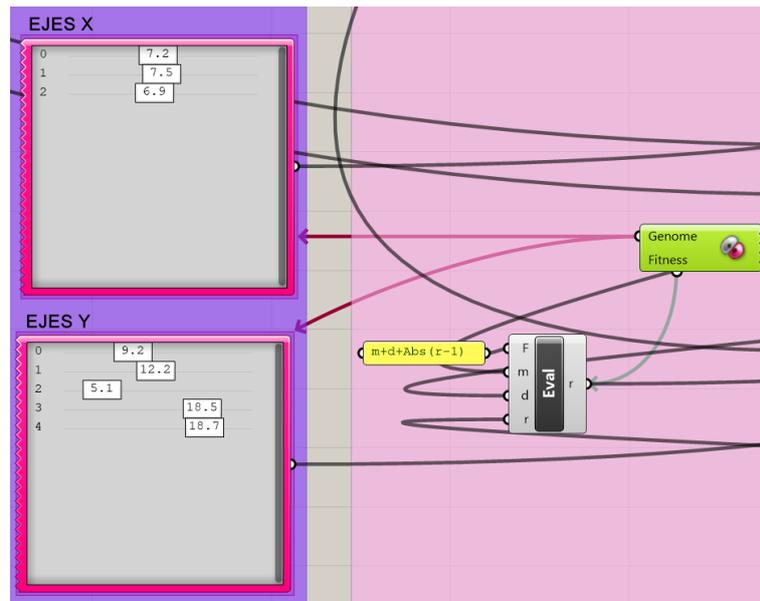


Figura 72 Parámetros conectados al componente "Galapagos"

3.7.2 Resultados de la tercera optimización

El solucionador de Galapagos se ejecutó durante aproximadamente 16 horas, obteniéndose 100 generaciones de resultados. La solución más óptima se alcanzó en la generación 81.

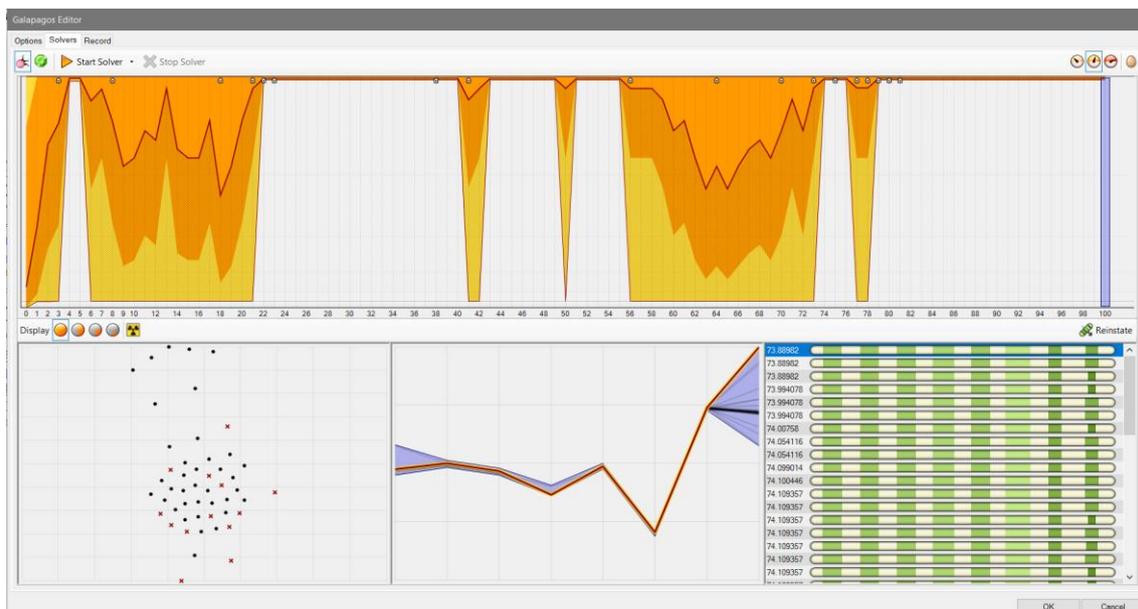


Figura 73 Ventana del solucionador de Galapagos para la tercera optimización

La optimización topológica dio como resultado los siguientes resultados en cuanto a la posición de los ejes y las columnas:

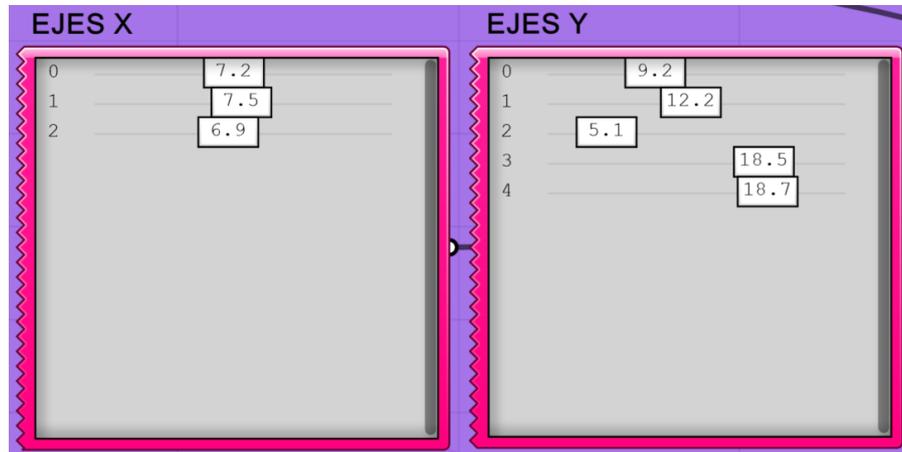


Figura 74 Lista de la posición de los ejes con la optimización de "Karamba 3D"

La figura 74 muestra el resultado en los componentes "Gene List" de "Grasshopper", sin embargo, los resultados se muestran de manera desordenada, en la tabla 5 se muestra la posición de los ejes de una manera más ordenada:

Tabla 5.

| ID | Ordenada en X (m) | ID | Ordenada en Y (m) |
|----|-------------------|----|-------------------|
| A | 0.0 | 1 | 0.0 |
| B | 7.5 | 2 | 5.1 |
| C | 15.0 | 3 | 12.2 |
| | | 4 | 18.5 |
| | | 5 | 25.0 |

Tabla 5 Posiciones de los ejes de la estructura resultantes de la optimización 3

Se puede observar que los ejes en "Y" no fueron variados por el algoritmo, pero los ejes en "X" si fueron modificados.

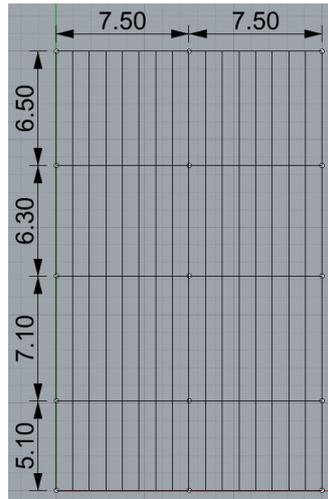


Figura 75 Esquema en planta de la posición de los ejes con la optimización topológica

Se obtuvieron los siguientes resultados de masa y deflexión máxima:

- Masa: 71.471 Ton
- Deflexión máxima: 2.118 cm



Figura 76 Estructura con la tercera optimización

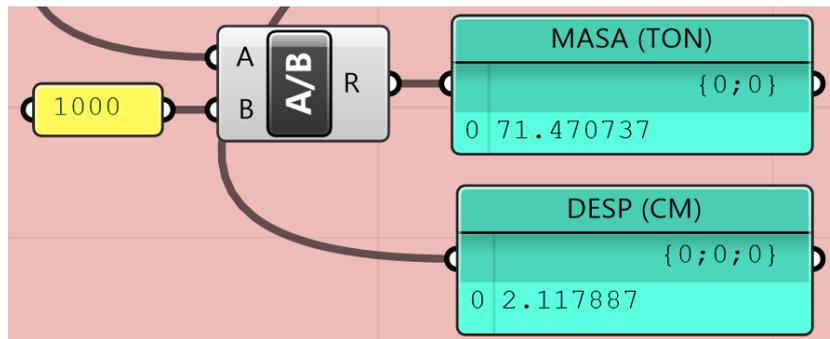


Figura 77 Masa y desplazamiento máximo obtenidos de la estructura con la tercera optimización

Al igual que en el caso 2, el optimizador de “Karamba 3D” no distingue entre tipos de elementos estructurales, por lo que es necesario racionalizar la solución.

3.7.3 Racionalización del caso 3

Para realizar la racionalización de este caso, se siguió los mismos principios que en el caso 2.

3.7.3.1 Resultados de la racionalización del caso 3

A continuación, se muestran las secciones transversales establecidas para cada elemento:

| Columnas | Sección | Vigas principales | Sección | Vigas secundarias | Sección |
|----------|---------|-------------------|---------|-------------------|---------|
| 1-A | HEB 180 | Planta 1 | | Planta 1 | |
| 1-B | HEB 200 | Eje 1 | IPE 450 | Paño A-B | IPE 270 |
| 1-C | HEB 160 | Eje 2 | IPE 600 | Paño B-C | IPE 240 |
| 2-A | HEB 220 | Eje 3 | IPE 600 | Planta 2 | |
| 2-B | HEB 260 | Eje 4 | IPE 600 | Paño A-B | IPE 240 |
| 2-C | HEB 220 | Eje 5 | IPE 400 | Paño B-C | IPE 220 |
| 3-A | HEB 220 | Eje A | IPE 180 | Planta 3 | |
| 3-B | HEB 240 | Eje B | IPE 220 | Paño A-B | IPE 240 |
| 3-C | HEB 220 | Eje C | IPE 180 | Paño B-C | IPE 240 |
| 4-A | HEB 220 | Planta 2 | | Cubierta | |
| 4-B | HEB 240 | Eje 1 | IPE 400 | Paño A-B | IPE 220 |
| 4-C | HEB 220 | Eje 2 | IPE 600 | Paño B-C | IPE 220 |
| 5-A | HEB 180 | Eje 3 | IPE 600 | | |
| 5-B | HEB 180 | Eje 4 | IPE 600 | | |
| 5-C | HEB 180 | Eje 5 | IPE 450 | | |
| | | Eje A | IPE 200 | | |
| | | Eje B | IPE 240 | | |
| | | Eje C | IPE 200 | | |
| | | Planta 3 | | | |
| | | Eje 1 | IPE 360 | | |
| | | Eje 2 | IPE 600 | | |
| | | Eje 3 | IPE 600 | | |
| | | Eje 4 | IPE 600 | | |
| | | Eje 5 | IPE 400 | | |
| | | Eje A | IPE 200 | | |
| | | Eje B | IPE 240 | | |
| | | Eje C | IPE 180 | | |
| | | Cubierta | | | |
| | | Eje 1 | IPE 400 | | |
| | | Eje 2 | IPE 600 | | |
| | | Eje 3 | IPE 600 | | |
| | | Eje 4 | IPE 600 | | |
| | | Eje 5 | IPE 450 | | |
| | | Eje A | IPE 200 | | |
| | | Eje B | IPE 240 | | |
| | | Eje C | IPE 200 | | |

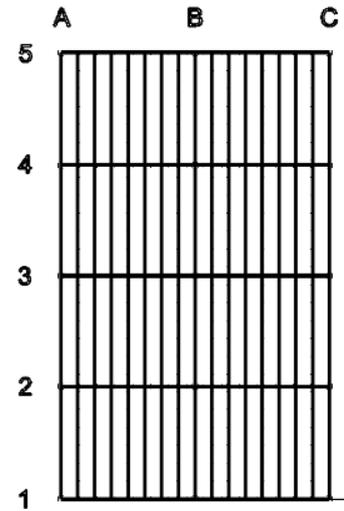


Figura 78 Secciones transversales establecidas para el caso 3 racionalizado

Se obtuvieron los siguientes resultados:

- Masa: 91.78 Ton
- Deflexión: 1.611 cm

El cálculo de la masa se realizó de manera manual y su desarrollo se evidencia en el anexo 9, 10 y 11.

Tabla 6.

| Elemento | Masa (Ton) |
|-------------------|-------------------|
| Columnas | 12.76 |
| Vigas principales | 37.45 |
| Vigas secundarias | 41.56 |
| | 91.78 |

Tabla 6 Masa de la estructura con la segunda optimización racionalizada

4. CAPÍTULO 4 ANÁLISIS DE RESULTADOS

El algoritmo de modelación paramétrica desarrollado en “Grasshopper” demostró una alta eficacia al modelar estructuras aporricadas de acero de manera versátil y fácil de usar. Los usuarios pueden modificar variables clave, como dimensiones y configuraciones geométricas, de manera intuitiva, adaptando rápidamente el modelo estructural a diferentes escenarios. Esta flexibilidad resalta la utilidad del algoritmo como herramienta de diseño, especialmente en proyectos que requieren iteraciones rápidas.

El análisis estructural, llevado a cabo con “Karamba 3D”, resultó en datos precisos y coherentes. Los datos de desplazamientos, esfuerzos y reacciones mostraron un comportamiento acorde con los parámetros establecidos, validando la capacidad del algoritmo para generar modelos precisos que responden a los requerimientos normativos y de diseño.

Para evaluar los diferentes casos de optimización estructural, se propuso un modelo geométrico cuyos elementos estructurales tengan las secciones más grandes del catálogo, de esta manera se podrá evidenciar el nivel de optimización que alcanzó cada caso, es decir, cuanto peso se redujo. Los valores iniciales obtenidos fueron los siguientes:

- Masa: 258.867 Ton
- Desplazamiento máximo: 0.695 cm

Caso 1: Optimización de tamaño y forma con Galapagos

El primer caso utilizó el algoritmo genético de Galapagos para optimizar el tamaño y la forma de las secciones transversales. Aunque los resultados fueron satisfactorios desde un punto de vista constructivo, el proceso requirió un tiempo significativo, alcanzando las soluciones óptimas después de 16 horas. Este tiempo prolongado se debe a la complejidad inherente del algoritmo genético y al número elevado de iteraciones necesarias para converger.

Los resultados mostraron una reducción del peso total de la estructura, aunque no fue la más eficiente en comparación con otros casos. Esto se atribuye a que ciertos elementos estructurales recibieron secciones mayores de lo necesario, un compromiso para mantener una solución más viable desde el punto de vista constructivo.

Resultados clave:

- Masa final: 118.469 Ton
- Deflexión máxima: 1.359 cm
- Masa reducida: 140.398 Ton

Caso 2: Optimización de tamaño y forma con Karamba 3D

Para el segundo caso se optimizó la forma y tamaño de las secciones transversales de los elementos con el programa “Karamba 3D”. El componente "Optimize Cross Section" de “Karamba 3D” demostró una notable eficiencia al generar resultados en apenas 5 segundos, significativamente más rápido que Galapagos. La optimización logró reducir considerablemente el peso de la estructura en comparación con el Caso 1. Sin embargo, los resultados presentaron una limitación práctica: los elementos estructurales tenían secciones transversales distintas incluso dentro del mismo tipo de elemento, esto se debe a que el optimizador de “Karamba 3D” no permite agrupar elementos o diferenciarlos en diferentes tipos. Esta diversidad excesiva complica la construcción y requiere un paso adicional de racionalización manual.

Resultados clave:

- Masa: 72.414 Ton
- Deflexión máxima: 2.149 cm
- Masa reducida: 186.453 Ton

En la racionalización, se asignaron secciones únicas a grupos de elementos, priorizando la factibilidad constructiva. Aunque el peso de la estructura aumentó después de este ajuste, seguía siendo menor que en el Caso 1.

Resultados clave:

- Masa después de la racionalización: 90.1 Ton
- Masa reducida: 168.767 Ton
- Deflexión máxima: 1.372 cm
- Masa aumentada con respecto al resultado sin racionalizar: 17.686 Ton

Caso 3: Optimización topológica y de tamaño con Galapagos y Karamba 3D

El tercer caso combinó la optimización topológica (modificación de la posición de columnas) con la optimización de tamaño y forma. El algoritmo genético de Galapagos funcionó correctamente, variando principalmente las posiciones de los ejes en dirección "X", mientras que los ejes en "Y" permanecieron sin cambios. Esto se alinea con la lógica de ubicar las columnas más cerca de las áreas con mayores sobrecargas. Sin embargo, la reducción del peso estructural fue mínima en comparación con el Caso 2, indicando que la optimización topológica no tuvo un impacto significativo.

Resultados clave:

- Masa: 71.471 Ton
- Deflexión máxima: 2.118 cm
- Masa reducida: 187.369 Ton
- Diferencia de masa reducida con respecto al caso 2 sin racionalizar: 0.942 Ton

Tras la racionalización, los resultados reflejaron un peso ligeramente mayor que en el Caso 2, confirmando que este enfoque no es el más eficiente para reducir el peso estructural.

Resultados clave:

- Masa después de la racionalización: 91.78 Ton
- Masa reducida: 167.087 Ton
- Deflexión máxima: 1.611 cm
- Masa aumentada con respecto al caso 2 racionalizado: 1.68 Ton

Tabla 7.

| | Caso 1 | | Caso 2 | | Caso 3 | |
|--|----------------|----------------|--|------------------------------------|--|------------------------------------|
| | Modelo inicial | Optimización 1 | Optimización 2 | Racionalización | Optimización 3 | Racionalización |
| Masa (Ton) | 258.867 | 118.469 | 72.414 | 90.1 | 71.471 | 91.78 |
| Deflexión máxima (cm) | 0.695 | 1.359 | 2.149 | 1.372 | 2.118 | 1.611 |
| Columnas interiores | SHS 300x300x8 | SHS 300x300x8 | HEB 140, HEB 160, HEB 180, HEB 200, HEB 220, HEB 240, HEB 280 | HEB 220, HEB 240, HEB 240 | HEB 140, HEB 160, HEB 180, HEB 220, HEB 240, HEB 260 | HEB 240, HEB 260 |
| Columnas perimetrales | SHS 300x300x8 | SHS 300x300x6 | HEB 140, HEB 160, HEB 180, HEB 200, HEB 220, HEB 240 | HEB 180, HEB 200, HEB 220, HEB 240 | HEB 120, HEB 140, HEB 160, HEB 180, HEB 200, HEB 220 | HEB 160, HEB 180, HEB 200, HEB 220 |
| Vigas secundarias primera planta | IPE 600 | IPE 450 | IPE 200, IPE 220, IPE 240, IPE 270 | IPE 220, IPE 270 | IPE 200, IPE 220, IPE 240, IPE 270 | IPE 240, IPE 270 |
| Vigas en X primera planta | IPE 600 | IPE 600 | IPE 200, IPE 220, IPE 240, IPE 270, IPE 300, IPE 330, IPE 360, IPE 400, IPE 450, IPE 500, IPE 550, IPE 600 | IPE 400, IPE 500, IPE 550, IPE 600 | IPE 200, IPE 220, IPE 240, IPE 270, IPE 300, IPE 330, IPE 360, IPE 400, IPE 450, IPE 500, IPE 550, IPE 600 | IPE 400, IPE 450, IPE 600 |
| Vigas en Y primera planta | IPE 600 | IPE 360 | IPEB 160, IPE 180, IPE 200, IPE 220 | IPE 200, IPE 220 | IPE 160, IPE 180, IPE 200, IPE 220 | IPE 180, IPE 220 |
| Vigas secundarias plantas intermedias | IPE 600 | IPE 240 | IPE 220, IPE 240 | IPE 240, IPE 220 | IPE 200, IPE 220, IPE 240 | IPE 220, IPE 240 |
| Vigas en X plantas intermedias | IPE 600 | IPE 500 | IPE 200, IPE 220, IPE 240, IPE 270, IPE 300, IPE 330, IPE 360, IPE 400, IPE 450, IPE 500, IPE 550, IPE 600 | IPE 400, IPE 450, IPE 600 | IPE 200, IPE 220, IPE 240, IPE 270, IPE 300, IPE 330, IPE 360, IPE 400, IPE 450, IPE 500, IPE 600 | IPE 360, IPE 400, IPE 450, IPE 600 |
| Vigas en Y plantas intermedias | IPE 600 | IPE 360 | IPE 160, IPE 180, IPE 200, IPE 220 | IPE 180, IPE 200, IPE 220 | IPE 160, IPE 180, IPE 200, IPE 220, IPE 240 | IPE 180, IPE 200, IPE 240 |
| Vigas secundarias cubierta | IPE 600 | IPE 240 | IPE 180, IPE 220 | IPE 220 | IPE 140, IPE 200, IPE 220 | IPE 220 |
| Viga en X cubierta | IPE 600 | IPE 550 | IPE 240, IPE 270, IPE 300, IPE 330, IPE 360, IPE 400, IPE 450, IPE 500, IPE 600 | IPE 450, IPE 600 | IPE 220, IPE 240, IPE 270, IPE 300, IPE 360, IPE 400, IPE 450, IPE 500, IPE 600 | IPE 400, IPE 450, IPE 600 |
| Vigas en Y cubierta | IPE 600 | IPE 360 | IPE 180, IPE 220 | IPE 180, IPE 220 | IPE 140, IPE 160, IPE 180, IPE 200, IPE 220, IPE 240 | IPE 200, IPE 240 |
| Disminución del peso con respecto al modelo inicial | | 54.24% | 72.03% | 65.19% | 72.39% | 64.55% |

Tabla 7 Tabla comparativa de las optimizaciones realizadas

5. CAPÍTULO 5 (CONCLUSIONES Y RECOMENDACIONES)

El desarrollo e implementación del algoritmo demostró ser una solución efectiva para abordar el problema planteado en el capítulo 1 de esta tesis: la necesidad de integrar herramientas paramétricas, de análisis estructural y optimización para el diseño eficiente de estructuras. Se cumplió el objetivo de explorar métodos computacionales que permitan optimizar el diseño estructural, reduciendo el peso de la estructura mientras se garantiza el cumplimiento de criterios normativos y de seguridad.

El algoritmo desarrollado en “Grasshopper” probó ser una herramienta robusta para modelar paraméricamente estructuras aperticadas de manera eficiente. La capacidad de modificar variables geométricas y estructurales con facilidad refuerza la hipótesis inicial de que los enfoques paraméricos pueden simplificar procesos de diseño y análisis en proyectos estructurales. Su flexibilidad también abre la posibilidad de adaptarse a diferentes escenarios y requerimientos constructivos.

Las herramientas proporcionadas por “Karamba 3D” permitieron ensamblar el modelo estructural de manera eficiente, teniendo la posibilidad de generar diferentes tipos de cargas, apoyos, elementos, secciones transversales, etc. Y el análisis estructural hecho con esta herramienta permitió evaluar desplazamientos, esfuerzos y reacciones con precisión, facilitando la comparación entre diferentes casos de estudio y respaldando la toma de decisiones en el proceso de optimización.

En cuanto a los resultados de optimización obtenidos, se concluye para el caso 1 que, aunque el uso de Galapagos logró reducir el peso de la estructura, los tiempos prolongados para encontrar soluciones y la agrupación de elementos estructurales limitaron su efectividad en términos de costos y eficiencia. Sin embargo, desde un punto de vista constructivo, esta solución fue favorable al mantener una racionalización aceptable de las secciones transversales.

En el caso 2, el componente "Optimize Cross Section" de Karamba 3D mostró una gran ventaja en términos de tiempo de cálculo, pero los resultados no fueron directamente aplicables debido a la diversidad de secciones generadas. Aunque la racionalización manual permitió reducir el peso total, el proceso adicional implicó una inversión significativa de tiempo y esfuerzo.

En el caso 3, la optimización topológica no tuvo un impacto relevante en la reducción del peso estructural si lo comparamos con el caso anterior. Aunque se logró posicionar las columnas de forma coherente con las cargas aplicadas, la diferencia en el peso final fue marginal en comparación con los otros casos. Este resultado pone en duda la efectividad de este enfoque en estructuras similares y sugiere que la optimización topológica debería centrarse en otras variables en proyectos futuros.

Las herramientas computacionales utilizadas demostraron ser útiles para abordar problemas complejos de diseño estructural. Sin embargo, también presentaron limitaciones importantes:

El software "Rhino" y su extensión "Grasshopper" demostraron ser herramientas sumamente útiles para la creación de algoritmos de modelación paramétrica de cualquier nivel de complejidad. Dado el potencial que ofrece "Grasshopper", es recomendable seguir desarrollando extensiones y complementos que amplíen aún más sus capacidades y aplicaciones en diferentes áreas de la ingeniería y la arquitectura. Se concluye que este tipo de algoritmos puede aplicarse para modelar cualquier tipo de estructura, desde edificaciones simples hasta sistemas estructurales complejos, facilitando una integración más ágil entre el diseño y la ingeniería estructural.

No obstante, en el caso específico de las herramientas de optimización estructural, se identificaron ciertas limitaciones que afectaron la eficiencia y efectividad del proceso:

Karamba 3D: A pesar de ser una herramienta precisa y rápida, el componente “Optimize Cross Section” presentó dificultades para converger a una solución óptima en modelos complejos. Lo que resultó en la necesidad de ajustar manualmente ciertos parámetros, como el límite del radio de utilización, para garantizar que las soluciones propuestas cumplieran con los requisitos de seguridad estructural.

Galapagos: Aunque es una herramienta poderosa para la optimización basada en algoritmos genéticos, su principal limitación radica en el tiempo prolongado necesario para encontrar soluciones óptimas, especialmente en modelos estructurales complejos. Este factor puede ser un inconveniente significativo en proyectos con plazos ajustados o que requieren iteraciones rápidas.

Por estas razones, se recomienda aplicar estos algoritmos de optimización a modelos más simples o específicos. Un enfoque efectivo podría ser desarrollar algoritmos de optimización que respondan a un único tipo de sistema estructural. Esto permitiría obtener soluciones más rápidas y confiables, mientras se reduce la necesidad de ajustes manuales y racionalizaciones posteriores.

Es crucial continuar con el desarrollo de herramientas computacionales para modelación, análisis y optimización estructural. Se recomienda explorar nuevas metodologías de optimización topológica que puedan integrarse de manera más efectiva con los objetivos de diseño estructural.

El uso de herramientas paramétricas y de optimización tiene el potencial de transformar la industria de la construcción y representa una evolución necesaria. Aunque las herramientas utilizadas en este estudio tienen limitaciones, sus beneficios superan ampliamente sus desventajas, permitiendo diseñar estructuras y procesos más eficientes. Este enfoque marca un camino hacia el futuro del diseño estructural, donde la innovación tecnológica será clave para enfrentar desafíos más complejos y exigencias normativas más estrictas.

6. REFERENCIAS

- (MIDUVI), M. d. (2014). CARGAS (NO SÍSMICAS). En *NORMA ECUATORIANA DE LA CONSTRUCCIÓN* (págs. 25-30). Quito: Dirección de Comunicación Social, MIDUVI.
- Callejas, R. (2010). *Formulación y Evaluación de un Plan Negocio*. Quito, Ecuador: McGraw Hill. doi:978-9942-03-111-2
- Casado Bravo, F. (Febrero de 2022). *Optimización Estructural Mediante Algoritmos Computacionales Inspirados en la Naturaleza*. Obtenido de Archivo Digital UPM: <https://oa.upm.es/70277/>
- Casado, F. (2022). *OPTIMIZACIÓN ESTRUCTURAL MEDIANTE ALGORITMOS COMPUTACIONALES INSPIRADOS EN LA NATURALEZA*. Madrid: Universidad Politécnica de Madrid.
- Cervera, M., & Blanco, E. (2002). *Mecánica de estructuras Libro 2 Métodos de análisis*. Barcelona: Edicions UPC.
- Cormet, T., Leiserson, C., Rivest, R., & Stein, C. (2022). *Introduction to algorithms*. Londres: The MIT Press.
- Echeverri Montes, P. (27 de Septiembre de 2021). *Grasshopper para Rhino: ¿qué es y cómo se utiliza?* Obtenido de Echeverrimontes: <https://www.echeverrimontes.com/blog/grasshopper-para-rhino-que-es-y-como-se-utiliza>
- Erickson, J. (2019). *Algorithms*.
- García Alvarado, R., & Lyon Gottlieb, A. (2013). Diseño paramétrico en Arquitectura; método, técnicas y aplicaciones. *ARQUISUR REVISTA*, 18-27.

- Gonzales, R. (19 de Febrero de 2021). *¿Qué es Rhinoceros 3D y por qué deberías usar esta herramienta en tus diseños?* Obtenido de Crehana:
<https://www.crehana.com/blog/estilo-vida/rhinoceros-3d/#que-es-rhinoceros-3d>
- IONOS. (09 de 10 de 2020). *Programación visual: la entrada más sencilla al mundo digital*. Obtenido de IONOS Digital Guide: <https://www.ionos.com/es-us/digitalguide/paginas-web/desarrollo-web/programacion-visual/>
- Landeta, B. (2021). *Optimización de estructuras paramétricas de acero mediante algoritmos genéticos evolutivos*. Quito: Universidad Internacional SEK.
- McNeel, R. (2024). *Rhino-Characterísticas*. Obtenido de Rhinoceros.
- Molinare, A. (09 de Noviembre de 2011). *¿Qué es el diseño paramétrico?* Obtenido de ArchDaily en Español: <https://www.archdaily.cl/cl/02-118243/%25c2%25bfque-es-el-diseno-parametrico>
- Nápoles, E., Gonzáles, R., & Olivares, E. (2015). *UNA INTRODUCCIONAL AL ANÁLISIS POR ELEMENTOS FINITOS: APLICACIONES Y EJEMPLOS*. Editorial Universitaria.
- Navarrete, S. (2014). Diseño paramétrico. El gran desafío del siglo XXI. En S. Navarrete, *Cuaderno 38. Cuadernos del Centro de Estudios en Diseño y Comunicación* (págs. 63-72). Buenos Aires: Universidad de Palermo.
- Romero, A., Galvín, P., & Solís, M. (2020). *Apuntes para el Análisis de Estructuras*. Sevilla: Escuela Técnica Superior de Ingeniería.
- Sánchez, S. (2012). *OPTIMIZACION ESTRUCTURAL Y TOPOLOGICA DE ESTRUCTURAS MORFOLÓGICAMENTE NO DEFINIDAS MEDIANTE ALGORITMOS GENÉTICOS*. Valencia: Universitat Politècnica de Valencia.
- Seguí, P. (Diciembre de 2023). *Arquitectura paramétrica: El diseño paramétrico transforma la construcción y el urbanismo*. Obtenido de OVACEN: <https://ovacen.com/disen-parametrico-arquitectura/>

7. ANEXOS

7.1 Anexo 1. Catálogo de secciones transversales disponibles para columnas del caso de estudio

| Familia | Nombre |
|---------|--------|
| HEB | HEB100 |
| | HEB120 |
| | HEB140 |
| | HEB160 |
| | HEB180 |
| | HEB200 |
| | HEB220 |
| | HEB240 |
| | HEB260 |
| | HEB280 |
| | HEB300 |
| | HEB320 |
| | HEB340 |
| | HEB360 |
| | HEB400 |
| | HEB450 |
| | HEB500 |
| | HEB550 |
| HEB600 | |

| Familia | Nombre |
|---------------|---------------|
| QRO(EN10219) | SHS 40x40x4 |
| | SHS 40x40x5 |
| | SHS 50x50x4 |
| | SHS 50x50x5 |
| | SHS 50x50x6 |
| | SHS 60x60x4 |
| | SHS 60x60x5 |
| | SHS 60x60x6 |
| | SHS 70x70x4 |
| | SHS 70x70x5 |
| | SHS 70x70x6 |
| | SHS 80x80x4 |
| | SHS 80x80x5 |
| | SHS 80x80x6 |
| | SHS 90x90x4 |
| | SHS 90x90x5 |
| | SHS 90x90x6 |
| | SHS 100x100x4 |
| | SHS 100x100x5 |
| | SHS 100x100x6 |
| | SHS 125x125x4 |
| | SHS 125x125x5 |
| | SHS 125x125x6 |
| | SHS 150x150x4 |
| | SHS 150x150x5 |
| | SHS 150x150x6 |
| | SHS 175x175x4 |
| | SHS 175x175x5 |
| | SHS 175x175x6 |
| | SHS 200x200x4 |
| | SHS 200x200x5 |
| | SHS 200x200x6 |
| SHS 250x250x4 | |
| SHS 250x250x5 | |
| SHS 250x250x6 | |
| SHS 300x300x6 | |
| SHS 300x300x8 | |

7.2 Anexo 2. Catálogo de secciones transversales disponibles para vigas principales y secundarias del caso de estudio

| Familia | Nombre |
|----------------|---------------|
| IPE | IPE80 |
| | IPE100 |
| | IPE120 |
| | IPE140 |
| | IPE160 |
| | IPE180 |
| | IPE200 |
| | IPE220 |
| | IPE240 |
| | IPE270 |
| | IPE300 |
| | IPE330 |
| | IPE360 |
| | IPE400 |
| | IPE450 |
| | IPE500 |
| | IPE550 |
| IPE600 | |
| IPE 750x185 | |

7.3 Anexo 3. Código en Visual Basic para limitar el desplazamiento máximo del modelo

```
Private Sub RunScript(ByVal d As Object, ByVal L As Object, ByRef A As Object)
```

```
    If d > (L / 300) Then
```

```
        A = 10000
```

```
    Else
```

```
        A = d
```

```
    End If
```

```
End Sub
```

7.4 Anexo 4. Código en Visual Basic para limitar los radios de utilización

```
Private Sub RunScript(ByVal list As Object, ByVal y As Object, ByRef A As Object)
```

```
    If list > 0.9 Then
```

```
        list = 1000
```

```
    End If
```

```
    print(list)
```

```
End Sub
```

7.5 Anexo 5. Código en Visual Basic para convertir a 10000 todos los valores de la lista si algún radio de optimización excede el máximo aceptable

```
Private Sub RunScript(ByVal list2 As List(Of Object), ByVal L As Object,
ByRef A As Object)

    Dim f As Double

    f = 0

    For i As Integer = 0 To (L - 1)

        If list2(i) = 1000 Then

            f = 1

        End If

    Next

    If f = 1 Then

        For i As Integer=0 To (L - 1)

            List2(i) = 10000

        Next

    End If

    A = list2

End Sub
```

**7.6 Anexo 6. Cálculo de la masa de las columnas de la estructura del caso 2
racionalizado**

| Columnas | Sección | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|----------|---------|--------------|--------------------|---------------------|
| 1-A | HEB 200 | 12.8 | 61.3 | 0.785 |
| 1-B | HEB 220 | 12.8 | 71.5 | 0.915 |
| 1-C | HEB 180 | 12.8 | 51.2 | 0.655 |
| 2-A | HEB 240 | 12.8 | 83.2 | 1.065 |
| 2-B | HEB 280 | 12.8 | 103 | 1.318 |
| 2-C | HEB 200 | 12.8 | 61.3 | 0.785 |
| 3-A | HEB 200 | 12.8 | 61.3 | 0.785 |
| 3-B | HEB 220 | 12.8 | 71.5 | 0.915 |
| 3-C | HEB 200 | 12.8 | 61.3 | 0.785 |
| 4-A | HEB 200 | 12.8 | 61.3 | 0.785 |
| 4-B | HEB 240 | 12.8 | 83.2 | 1.065 |
| 4-C | HEB 220 | 12.8 | 71.5 | 0.915 |
| 5-A | HEB 180 | 12.8 | 51.2 | 0.655 |
| 5-B | HEB 180 | 12.8 | 51.2 | 0.655 |
| 5-C | HEB 180 | 12.8 | 51.2 | 0.655 |
| | | | | 12.739 |

7.7 Anexo 7. Cálculo de la masa de las vigas principales de la estructura del caso 2 racionalizado

| Vigas principales | Sección | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|--------------------------|----------------|---------------------|------------------------|-------------------------|
| Planta 1 | | | | |
| Eje 1 | IPE 500 | 15 | 90.7 | 1.361 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 550 | 15 | 106 | 1.590 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje A | IPE 200 | 25 | 22.4 | 0.560 |
| Eje B | IPE 220 | 25 | 26.2 | 0.655 |
| Eje C | IPE 200 | 25 | 22.4 | 0.560 |
| Planta 2 | | | | |
| Eje 1 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje A | IPE 200 | 25 | 22.4 | 0.560 |
| Eje B | IPE 220 | 25 | 26.2 | 0.655 |
| Eje C | IPE 180 | 25 | 18.8 | 0.470 |
| Planta 3 | | | | |
| Eje 1 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje A | IPE 180 | 25 | 18.8 | 0.470 |
| Eje B | IPE 220 | 25 | 26.2 | 0.655 |
| Eje C | IPE 180 | 25 | 18.8 | 0.470 |
| Cubierta | | | | |
| Eje 1 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje A | IPE 180 | 25 | 18.8 | 0.470 |
| Eje B | IPE 220 | 25 | 26.2 | 0.655 |
| Eje C | IPE 180 | 25 | 18.8 | 0.470 |
| | | | | 37.370 |

7.8 Anexo 8. Cálculo de la masa de las vigas secundarias de la estructura del caso 2 racionalizado

| Vigas secundarias | Sección | Cantidad | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|-------------------|---------|----------|--------------|-----------------|------------------|
| Planta 1 | | | | | |
| Paño A-B | IPE 270 | 7 | 25 | 36.1 | 6.3175 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Planta 2 | | | | | |
| Paño A-B | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Planta 3 | | | | | |
| Paño A-B | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Cubierta | | | | | |
| Paño A-B | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| | | | | | 39.9875 |

**7.9 Anexo 9. Cálculo de la masa de las columnas de la estructura del caso 3
racionalizado**

| Columnas | Sección | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|-----------------|----------------|---------------------|----------------------------|-----------------------------|
| 1-A | HEB 180 | 12.8 | 51.2 | 0.655 |
| 1-B | HEB 200 | 12.8 | 61.3 | 0.785 |
| 1-C | HEB 160 | 12.8 | 42.6 | 0.545 |
| 2-A | HEB 220 | 12.8 | 71.5 | 0.915 |
| 2-B | HEB 260 | 12.8 | 93 | 1.190 |
| 2-C | HEB 220 | 12.8 | 71.5 | 0.915 |
| 3-A | HEB 220 | 12.8 | 71.5 | 0.915 |
| 3-B | HEB 240 | 12.8 | 83.2 | 1.065 |
| 3-C | HEB 220 | 12.8 | 71.5 | 0.915 |
| 4-A | HEB 220 | 12.8 | 71.5 | 0.915 |
| 4-B | HEB 240 | 12.8 | 83.2 | 1.065 |
| 4-C | HEB 220 | 12.8 | 71.5 | 0.915 |
| 5-A | HEB 180 | 12.8 | 51.2 | 0.655 |
| 5-B | HEB 180 | 12.8 | 51.2 | 0.655 |
| 5-C | HEB 180 | 12.8 | 51.2 | 0.655 |
| | | | | 12.763 |

7.10 Anexo 10. Cálculo de la masa de las vigas principales de la estructura del caso 3 racionalizado

| Vigas principales | Sección | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|-------------------|---------|--------------|-----------------|------------------|
| Planta 1 | | | | |
| Eje 1 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje A | IPE 180 | 25 | 18.8 | 0.470 |
| Eje B | IPE 220 | 25 | 26.2 | 0.655 |
| Eje C | IPE 180 | 25 | 18.8 | 0.470 |
| Planta 2 | | | | |
| Eje 1 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje A | IPE 200 | 25 | 22.4 | 0.560 |
| Eje B | IPE 240 | 25 | 30.7 | 0.768 |
| Eje C | IPE 200 | 25 | 22.4 | 0.560 |
| Planta 3 | | | | |
| Eje 1 | IPE 360 | 15 | 57.1 | 0.857 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje A | IPE 200 | 25 | 22.4 | 0.560 |
| Eje B | IPE 240 | 25 | 30.7 | 0.768 |
| Eje C | IPE 180 | 25 | 18.8 | 0.470 |
| Cubierta | | | | |
| Eje 1 | IPE 400 | 15 | 66.3 | 0.995 |
| Eje 2 | IPE 600 | 15 | 122 | 1.830 |
| Eje 3 | IPE 600 | 15 | 122 | 1.830 |
| Eje 4 | IPE 600 | 15 | 122 | 1.830 |
| Eje 5 | IPE 450 | 15 | 77.6 | 1.164 |
| Eje A | IPE 200 | 25 | 22.4 | 0.560 |
| Eje B | IPE 240 | 25 | 30.7 | 0.768 |
| Eje C | IPE 200 | 25 | 22.4 | 0.560 |
| | | | | 37.454 |

7.11 Anexo 11. Cálculo de la masa de las vigas secundarias de la estructura del caso 3 racionalizado

| Vigas secundarias | Sección | Cantidad | Longitud (m) | Masa / L (Kg/m) | Masa total (Ton) |
|-------------------|---------|----------|--------------|-----------------|------------------|
| Planta 1 | | | | | |
| Paño A-B | IPE 270 | 7 | 25 | 36.1 | 6.3175 |
| Paño B-C | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Planta 2 | | | | | |
| Paño A-B | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Planta 3 | | | | | |
| Paño A-B | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Paño B-C | IPE 240 | 7 | 25 | 30.7 | 5.3725 |
| Cubierta | | | | | |
| Paño A-B | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| Paño B-C | IPE 220 | 7 | 25 | 26.2 | 4.585 |
| | | | | | 41.5625 |