



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería Marítima y Ciencias del Mar

OPTIMIZACIÓN ESTRUCTURAL BASADA EN LAS NORMAS ISO
12215 DE UNA EMBARCACIÓN MENOR DE ALTA VELOCIDAD
CONSTRUIDA CON PRFV

PROYECTO INTEGRADOR

Previo a la obtención del Título de:

INGENIERO NAVAL

Presentado por:

Miguel Angel Onofre Valero

César Humberto Venegas Silverio

GUAYAQUIL - ECUADOR

Año: 2020

DEDICATORIA

El presente proyecto lo dedicamos a Dios y a todas las personas que nos apoyaron a lo largo de toda nuestra carrera universitaria, en especial a nuestros padres. Adolfo Onofre y María Valero; Marlene Silverio y César Venegas.

AGRADECIMIENTOS

Nuestro más sincero agradecimiento a nuestro Creador y a José R. Marín L. por su gran ayuda como tutor y guía como profesor para nuestro desarrollo profesional.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Miguel A. Onofre y César H. Venegas* damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

Miguel A. Onofre V.

Cesar H. Venegas S.

EVALUADORES

.....
Msc. Nadia Muñoz

PROFESOR DE LA MATERIA

.....
José R. Marín L. Ph. D.

PROFESOR TUTOR

RESUMEN

El transporte interislas de pasajeros en Galápagos se realiza en lanchas rápidas fabricadas usando Plástico Reforzado con Fibra de Vidrio (PRFV). El diseño de estas embarcaciones no sigue comúnmente la aplicación de estándares de construcción y por ello existe la posibilidad de que se esté sobredimensionado su estructura; además estas embarcaciones navegan a altas velocidades y por ello es recomendable investigar formas para reducir su peso. En este trabajo se optimizó la estructura de la sección central de una embarcación de 12 metros de eslora que navega a 25 nudos.

Para el proceso de optimización se empleó Programación No Lineal con Variables Mixtas (MINLP) puesto que en el diseño estructural existen variables enteras y reales. Estos problemas mixtos se suelen resolver aplicando el algoritmo Ramificación y Acotamiento. Debido a la combinación de variables, se utilizó la librería de optimización Gekko capaz de resolver problemas MINLP a través del lenguaje de programación Python. El proceso se dividió en dos fases, la primera optimiza el número de refuerzos y el laminado de los paneles; la segunda optimiza las dimensiones y laminado de los refuerzos. Las restricciones aplicadas corresponden a los requerimientos de la norma ISO 12215: resistencia flexural de paneles y refuerzos, resistencia inter-laminar al corte de los paneles, rigidez de los refuerzos, y, resistencia al pandeo del alma y ala de los refuerzos. Como resultado de la optimización estructural se redujo el peso estructural en 24%.

Finalmente se evaluaron los resultados del proceso de optimización usando el método de elementos finitos para analizar la sección actual y la optimizada. Se modeló la estructura con elementos planos y el material tipo laminado compuesto; las secciones transversales extremas se consideraron como empotradas. La carga se aplicó como presión uniforme tanto en el fondo y costado de la estructura, con valores diferentes en cada región. Se comprobó que en la sección actual existe sobredimensionamiento y que, en el caso de la sección optimizada, los niveles de esfuerzo no exceden los permisibles.

Dado a que el peso de los motores es elevado, se recomienda desarrollar una optimización para la región de popa de la estructura.

Palabras Clave: Optimización estructural, PRFV, MEF, MINLP, ISO 12215.

ABSTRACT

The inter-island transportation of passengers in Galapagos is carried out in high speed boats manufactured using Fiberglass Reinforced Plastic (GRP). The design of these boats does not commonly follow the application of construction standards and therefore there is the possibility that their structure is being oversized; In addition, these boats navigate at high speeds and therefore it is advisable to investigate ways to reduce their weight. In this work, the structure of the central section of a 12-meter-long craft sailing at 25 knots was optimized.

Mixed Integer Nonlinear Programming (MINLP) was used for the optimization process, since in the structural design there are integer and real variables. These mixed problems are usually solved by applying the Branching and Bounding algorithm. Due to the combination of variables, the Gekko optimization library, capable of solving MINLP problems through the Python programming language, was used. The process was divided into two phases, the first one optimizes the number of stiffeners and the laminate of the plates; the second one optimizes the dimensions and laminate of the stiffeners.

The constraints applied correspond to the requirements of the ISO 12215 standard: flexural strength of plates and stiffeners, inter-laminar resistance to shear of plates, stiffness of stiffeners, and resistance to buckling of the web and flange of reinforcements. As a result of the structural optimization, the structural weight was reduced by 24%.

Finally, the results of the optimization process were evaluated using the finite element method to analyze the current and the optimized section. The structure was modeled with flat elements and composite laminate type material; extreme cross sections were considered clamped. The load was applied as a uniform pressure both on the bottom and side of the structure, with different values in each region. It was verified that in the current section there is oversizing and that, in the case of the optimized section, the stress levels do not exceed the permissible ones.

Since the weight of the engines is high, it is recommended to develop an optimization for the aft region of the structure.

Keywords: *Structural Optimization, GRP, FEM, MINLP, ISO 12215.*

ÍNDICE GENERAL

| | |
|--|------|
| EVALUADORES..... | 5 |
| RESUMEN | I |
| <i>ABSTRACT</i> | II |
| ÍNDICE GENERAL | III |
| ABREVIATURAS..... | VI |
| SIMBOLOGÍA..... | VII |
| ÍNDICE DE FIGURAS | VIII |
| ÍNDICE DE TABLAS..... | X |
| CAPÍTULO 1 | 1 |
| 1. Introducción | 1 |
| 1.1 Descripción del problema | 3 |
| 1.2 Justificación del problema | 4 |
| 1.3 Objetivos | 4 |
| 1.3.1 Objetivo General | 4 |
| 1.3.2 Objetivos Específicos..... | 4 |
| 1.4 Marco teórico..... | 5 |
| 1.4.1 Descripción del problema de optimización | 5 |
| 1.4.2 Requerimientos estructurales de las Normas ISO 12215. | 9 |
| 1.4.3 Evaluación de esfuerzos por flexión y por corte en planchas y refuerzos | 12 |
| CAPÍTULO 2 | 20 |
| 2. OPTIMIZACIÓN ESTRUCTURAL DEL FONDO Y COSTADO | 20 |
| 2.1 Proceso de Optimización para el fondo y costado | 20 |
| 2.1.1 Simplificaciones del laminado | 20 |
| 2.1.2 Constantes y variables de diseño..... | 21 |

| | | |
|--------------------|---|----|
| 2.1.3 | Expresiones para las restricciones | 24 |
| 2.1.4 | Función Objetivo | 30 |
| 2.2 | Implementación del código de programación..... | 31 |
| 2.2.1 | Importación de módulos y librerías..... | 31 |
| 2.2.2 | Datos de entrada..... | 31 |
| 2.2.3 | Funciones desarrolladas | 32 |
| 2.2.4 | Definición de variables de diseño y restricciones en la librería Gekko... | 34 |
| 2.3 | Comprobación Numérica de la estructura optimizada..... | 37 |
| 2.3.1 | Entrada de datos de los materiales y la Geometría | 38 |
| 2.3.2 | Selecciones nombradas y mallado | 39 |
| 2.3.3 | <i>Definición del componente Setup en el módulo ACP (Pre)</i> | 41 |
| 2.3.4 | Análisis estático estructural | 42 |
| 2.3.5 | ACP (Post) | 43 |
| CAPÍTULO 3 | | 44 |
| 3. | Resultados Y ANÁLISIS | 44 |
| 3.1 | Resultados de análisis en la embarcación modelo | 44 |
| 3.2 | Solución del proceso de optimización..... | 48 |
| 3.3 | Comparación de resultados con el MEF actual y optimizado..... | 53 |
| 3.3.1 | Comparación de esfuerzos normales | 53 |
| 3.3.2 | Comparación de deformación en dirección vertical | 56 |
| CAPÍTULO 4 | | 59 |
| 4. | CONCLUSIONES Y RECOMENDACIONES | 59 |
| 4.1 | Conclusiones | 59 |
| 4.2 | Recomendaciones..... | 60 |
| BIBLIOGRAFÍA | | 61 |
| APÉNDICES..... | | 62 |

| | |
|------------------|-----|
| APÉNDICE A..... | 63 |
| APÉNDICE B..... | 64 |
| APÉNDICE C | 78 |
| APÉNDICE D | 87 |
| APÉNDICE E..... | 93 |
| APÉNDICE F..... | 99 |
| APÉNDICE G | 100 |

ABREVIATURAS

| | |
|-------|--|
| ESPOL | Escuela Superior Politécnica del Litoral |
| ANSYS | Analysis System |
| ACP | Ansys Composite PrepPost |
| NLP | Nonlinear Programming |
| MINLP | Mixed Integer Nonlinear Programming |
| SQP | Sequential Quadratic Programming |
| ASTM | American Society for Testing and Materials |
| ISO | International Organization for Standardization |
| PRFV | Plástico Reforzado con Fibra de Vidrio |
| MEF | Método de Elementos Finitos |
| CSM | Chopped Strand Mat |
| WR | Woven Roving |
| SS | Stacking Sequence |
| KKT | Karush-Kuhn-Tucker |

SIMBOLOGÍA

| | |
|-------------|---|
| kg | Kilogramo |
| m | Metro |
| cm | Centímetro |
| mm | Milímetro |
| N | Newton |
| Pa | Pascal |
| psi | Libras por pulgada cuadrada |
| kn | Nudo |
| E | Módulo de Young |
| EI | Rigidez Flexural |
| SM | Módulo Seccional |
| CF | Factor de Cumplimiento |
| P_{BMP} | Presión de fondo en lanchas a motor planeadoras |
| P_{SMP} | Presión de costado en lanchas a motor planeadoras |
| Fd | Fuerza Cortante |
| Md | Momento flector |
| ρ_M | Densidad superficial de tela CSM |
| ρ_W | Densidad superficial de tela WR |
| φ_M | Contenido de fibra en tela CSM |
| φ_W | Contenido de fibra en tela WR |
| σ_U | Esfuerzo normal último |
| τ_U | Esfuerzo cortante último |
| q | Flujo Cortante |

ÍNDICE DE FIGURAS

| | |
|---|-----------|
| Figura 1.1 Diagrama de flujo general para optimización | 6 |
| Figura 1.2 Representación gráfica de un NLP en un espacio x | 7 |
| Figura 1.3 Lancha con cuadernaje longitudinal | 10 |
| Figura 1.4 Lancha con cuadernaje transversal | 10 |
| Figura 1.5 Tela Mat (izquierda) y tela Woven Roving (derecha) | 12 |
| Figura 1.6 Cuaderna maestra - Lancha A..... | 12 |
| Figura 1.7 Refuerzo estructural tipo sombrero | 15 |
| Figura 2.1 Esquema de la mitad del laminado simétrico | 20 |
| Figura 2.2 Áreas de corte y dimensiones principales de refuerzos tipo sombrero. | 27 |
| Figura 2.3 Esquema de los módulos usados en el análisis de ANSYS | 37 |
| Figura 2.4 Discretización de la quilla, sección actual (izquierda) y sección optimizada (derecha)..... | 41 |
| Figura 3.1 Presiones de diseño en plancha y refuerzos de la sección actual ... | 44 |
| Figura 3.2 Momento Flector y Fuerza Cortante en refuerzos | 45 |
| Figura 3.3 Momento Flector y Fuerza Cortante en planchas..... | 45 |
| Figura 3.4 Número de capas óptimas función objetivo desde diferentes puntos iniciales para el planchaje del fondo..... | 49 |
| Figura 3.5 Número de refuerzos transversales y función objetivo para diferentes puntos iniciales | 50 |
| Figura 3.6 Restricciones no dimensionalizadas y función objetivo para diferentes puntos iniciales | 51 |
| Figura 3.7 Variación de la función objetivo en el fondo por cada iteración | 51 |
| Figura 3.8 Comparación de pesos de elementos estructurales y peso total entre la sección actual y optimizada | 53 |
| Figura 3.9 Comparación de modelos estructurales para análisis FEM | 53 |
| Figura 3.10 Distribución de esfuerzo en dirección vertical para la sección actual (izquierda), sección óptima (derecha)..... | 54 |

| | |
|--|-----------|
| Figura 3.11 Distribución de esfuerzos en dirección x (local) para la plancha del fondo actual | 55 |
| Figura 3.12 Distribución de esfuerzos en dirección x (local) para la plancha del fondo óptima..... | 55 |
| Figura 3.13 Distribución de esfuerzos en dirección x (local) para la plancha del fondo actual | 56 |
| Figura 3.14 Distribución de esfuerzos en dirección x (local) para la plancha del fondo óptima..... | 56 |
| Figura 3.15 Distribución de deflexiones en dirección vertical para la sección actual (izquierda), sección óptima (derecha). | 57 |
| Figura 3.16 Deflexión de la quilla, cuaderna y plancha del fondo de la sección modelo | 58 |
| Figura 3.17 Deflexión de la quilla, cuaderna y plancha del fondo de la sección optimizada..... | 58 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1.1 Características principales - lancha A | 11 |
| Tabla 1.2 Datos estructurales - Lancha A..... | 12 |
| Tabla 1.3 Presiones de diseño..... | 13 |
| Tabla 1.4 Momento flector (Md) y Fuerza cortante (Fd) | 14 |
| Tabla 1.5 Propiedades mecánicas del laminado | 15 |
| Tabla 1.6 Factor de cumplimiento en resistencia a la flexión y al corte en planchas y refuerzos estructurales | 16 |
| Tabla 1.7 Factor de cumplimiento en resistencia al impacto contra objetos flotantes | 17 |
| Tabla 1.8 Factor de cumplimiento en resistencia a la flexión de la quilla | 18 |
| Tabla 1.9 Factor de cumplimiento para el pandeo local de refuerzos estructurales | 18 |
| Tabla 1.10 Factor de cumplimiento de la rigidez flexural (EI) de los refuerzos estructurales..... | 19 |
| Tabla 2.1 Constantes directas e indirectas..... | 21 |
| Tabla 2.2 Propiedades mecánicas del Chanul (<i>Humiriastrum Procerum</i>)..... | 22 |
| Tabla 2.3 Variables de diseño para la primera fase | 22 |
| Tabla 2.4 Variables de diseño para la segunda fase | 23 |
| Tabla 2.5 Características de la embarcación..... | 31 |
| Tabla 2.6 Dimensiones de paneles del fondo y costado | 32 |
| Tabla 2.7 Propiedades mecánicas de los materiales de construcción | 32 |
| Tabla 2.8 Densidades y gramajes de materiales de construcción | 32 |
| Tabla 2.9 Funciones desarrolladas para el fondo, costado y quilla..... | 33 |
| Tabla 2.10 Valores y relaciones para construcción de objetos..... | 34 |
| Tabla 2.11 Asignación de variables para el uso en la librería Gekko | 35 |
| Tabla 2.12 Propiedades de la madera Chanul (asumiendo isotropía) (Organization, 2020)..... | 38 |
| Tabla 2.13 Propiedades de las telas CSM y Roving en "Engineering Data" | 38 |
| Tabla 2.14 Dimensiones de la sección analizada | 39 |
| Tabla 2.15 Nombre de selecciones para la embarcación modelo | 40 |
| Tabla 2.16 Tamaño de cada sizing | 41 |

| | | |
|-------------------|--|-----------|
| Tabla 2.17 | Espesor de cada tipo de tela | 42 |
| Tabla 3.1 | Análisis de falla en planchas | 45 |
| Tabla 3.2 | Análisis de falla en refuerzos estructurales | 46 |
| Tabla 3.3 | Peso por área del laminado de las planchas de la lancha A | 47 |
| Tabla 3.4 | Peso por área y dimensiones de refuerzos y quilla | 47 |
| Tabla 3.5 | Peso de elementos estructurales y peso total de la sección actual ... | 48 |
| Tabla 3.6 | Presión y cargas de diseño en elementos estructurales optimizados | 48 |
| Tabla 3.7 | Número total óptimos de capas en laminado de planchaje de fondo y costado | 52 |
| Tabla 3.8 | Peso de elementos estructurales optimizados del fondo | 52 |
| Tabla 3.9 | Peso de elementos estructurales optimizados del costado..... | 52 |

CAPÍTULO 1

1. INTRODUCCIÓN

Los requerimientos de diseño más importantes para una lancha dedicada al transporte de pasajeros son la cantidad de personas que pueden transportar y la velocidad de navegación. Además, los armadores o dueños de este tipo de lanchas esperan realizar las rutas programadas con el menor consumo de combustible con el fin de mejorar la utilidad de sus compañías. La cantidad de pasajeros dependen principalmente de tamaño de la embarcación mientras que velocidad navegación depende de la potencia propulsiva y resistencia al avance, la que a su vez depende de las formas y peso del casco. De manera que, si se desea mejorar la velocidad de navegación ó reducir el consumo de combustible es necesario disminuir la resistencia al avance o aumentar la potencia propulsiva. Una opción para reducir la resistencia al avance de una embarcación rápida es disminuir su peso, generalmente compuesto por: casco, combustible, motores y pasajeros. Por lo tanto, conviene mejorar el diseño estructural de estas lanchas minimizando el peso para obtener embarcaciones eficientes que puedan brindar servicio de transporte de pasajeros seguro en las Islas Galápagos.

Diseñar una embarcación con el menor peso posible y con resistencia estructural adecuada es una tarea compleja. Una opción para reducir peso sin perder resistencia y rigideces de la estructura es construir el casco utilizando plástico reforzado con fibra de vidrio (PRFV). Algunas de las características que definen la estructura del casco construido con dicho material son: separación entre refuerzos, laminado del material, espesor del laminado, módulo seccional de refuerzos, entre otras. Esto muestra que se debe considerar gran cantidad de variables que siguen relaciones no lineales, de manera que existe la necesidad de implementar un proceso de optimización no lineal de forma computacional para minimizar el peso estructural.

Debido a la complejidad del problema de optimización estructural de lanchas rápidas fabricadas con PRFV varios autores sugieren seguir un esquema de optimización de dos fases. La primera fase se encarga de optimizar el número de refuerzos y el laminado del forro, mientras que en la segunda fase se optimiza la geometría y recubrimiento de los refuerzos. Dentro del proceso de optimización es primordial reconocer las variables de diseño, el método para calcular la función objetivo y las restricciones que se debe

considerar en cada nivel. Los métodos de optimización estructural para embarcaciones basan sus restricciones usando el análisis estructural, conocido como “método directo” o el diseño basado en reglas de construcción o clasificación. El primer método es analíticamente complejo, mientras que la aplicación de estándares o reglas de diseño es recomendable para embarcaciones menores. De manera que, un proceso de optimización estructural en dos fases aplicando reglas para el diseño de embarcaciones parece adecuado para el caso de lanchas rápidas.

Se han desarrollado numerosos estudios sobre optimización estructural de embarcaciones con la finalidad de reducir el peso y costo de construcción. Entre los estudios más cercanos a este ámbito de trabajo está (Marín & Costaín, 1998) que muestra la optimización estructural minimizando el peso y costo de una embarcación de desplazamiento en acero usando las reglas de una sociedad de clasificación de buques; en ese trabajo los autores reconocen la limitación de tomar la separación entre refuerzos en lugar del número de ellos. En (Goubalt & Mayes, 1996) se presenta la comparación del peso y costo de la estructura primaria de una lancha patrullera construida con acero-aluminio frente a materiales compuestos; se logró reducir 65% el peso estructural y 95% el costo total. Trabajos más recientes como (Song & Oh, 2016) presenta el diseño estructural con menor peso para un yate construido con materiales compuesto; basado en los lineamientos de la ISO 12215 optimizó el contenido de fibra de vidrio para el laminado de paneles y refuerzos, como resultado logró la reducción del 10.01% del peso comparado con el diseño original.

Por otra parte, en un ambiente más académico se han elaborado trabajos usando métodos directos junto con teoría de fallas para materiales compuestos con el objetivo de optimizar estructuralmente embarcaciones construida con PRFV. El trabajo presentado por (Sobey, Blake, & Sheno, 2013) utiliza el método directo y las reglas de construcción para evaluar las restricciones en el proceso de optimización; empleando un algoritmo genético obtienen paneles óptimos en masa-costo. Concluyeron que utilizando el método directo para evaluar las restricciones se obtiene menor masa-costo en comparación con el uso de reglas de construcción. Las planchas laminadas de PRFV con refuerzos tipo sombrero son el enfoque del trabajo realizado en (Maneeapan, Sheno, & Blake, 2007), presenta distintos ejemplos de optimización estructural utilizando

algoritmo genético para planchas sin refuerzos, con refuerzos en una dirección y dos direcciones. Concluyen que los resultados obtenidos para cada ejemplo están acorde a los fundamentos teóricos en los que se basa el diseño estructural y los efectos de variación de parámetros en cada caso. En contraste con los trabajos anteriores en este trabajo se desarrollará la optimización estructural de la sección central de una lancha rápida con consideraciones prácticas para la implementación de su construcción. Se va a considerar formas típicas de la embarcación, materiales disponibles, tipos de reforzamientos, forma de construcción, entre otras.

1.1 Descripción del problema

Las embarcaciones dedicadas al servicio de transporte de pasajeros como cualquier buque deben estar diseñadas de manera eficiente con la finalidad de alcanzar la velocidad de diseño con mínima potencia propulsiva. Para esto existen diferentes metodologías a seguir, entre ellas el análisis detallado de la forma del casco y distribución de pesos para conseguir la menor resistencia al avance, utilizar métodos propulsivos más eficientes o la construcción del casco con materiales ligeros. Para diseñar una embarcación con la resistencia estructural y rigidez adecuadas es común seguir los lineamientos que proponen las sociedades clasificadoras de buques, llamados “reglas de clasificación de buques”. Para el caso de embarcaciones menores a 24 metros de eslora la Organización Internacional de Normalización (ISO) propone el conjunto de estándares ISO 12215 orientado al diseño de embarcaciones menores de altas velocidades.

De manera que al diseño estructural de una embarcación construida con materiales compuesto se lo puede acoplar con un proceso de optimización con el objetivo de obtener el mínimo peso estructural; se pueden considerar los requerimientos estructurales presentados en la norma ISO 12215 en la parte relativa al diseño estructural como las restricciones del proceso de optimización. Finalmente, para corroborar los resultados obtenidos en el proceso de optimización se puede utilizar el método de elementos finitos para evaluar los niveles de esfuerzos en la estructura optimizada.

1.2 Justificación del problema

La construcción local de embarcaciones fabricadas con plástico reforzado con fibra de vidrio (PRFV) es parte importante del mercado naval nacional, la mayoría de las lanchas dedicadas al transporte de pasajeros interislas en Galápagos están construidas con PRFV. Sin embargo, dado a que se trata de embarcaciones menores se construyen principalmente basándose en la experiencia y se puede esperar que en el proceso de diseño se está sobredimensionando la rigidez estructural de estas embarcaciones. Este problema puede ocurrir el desconocimiento de reglas o normas de construcción específicas para embarcaciones menores,

Con la finalidad de aumentar la velocidad de navegación y asegurar un dimensionamiento estructural adecuado se puede optar por aplicar un proceso de optimización al peso estructural siguiendo los requerimientos de estándares internacionales para el diseño de embarcaciones menores de alta velocidad. Lo que implicaría además de la reducción de peso y aumento de velocidad, un ahorro en el material de construcción y disminución de potencia de propulsión requerida.

1.3 Objetivos

1.3.1 Objetivo General

Minimizar el peso estructural de una embarcación menor de alta velocidad construida con PRFV implementando un proceso de optimización que considere los lineamientos de la norma ISO 12215.

1.3.2 Objetivos Específicos

1. Aplicar los lineamientos de las normas ISO 12215 parte 5 tomando las características de diseño de una embarcación interislas de Galápagos generando valores referenciales de espesor del forro y dimensiones de refuerzos estructurales.
2. Definir un proceso de optimización estructural adecuado para embarcaciones menores de alta velocidad aplicando los lineamientos de la norma ISO 12215 parte 5 como restricciones de diseño.
3. Implementar el proceso de optimización estructural usando el lenguaje de programación Python.

4. Evaluar estructuralmente el diseño optimizado y no optimizado empleando análisis de elementos finitos para materiales compuestos del software ANSYS.

1.4 Marco teórico

1.4.1 Descripción del problema de optimización

1.4.1.1 El problema de optimización no lineal y la región factible

En general los problemas de optimización son expresados como minimización (o maximización) de una o varias funciones sujetas a restricciones en forma de ecuaciones e inecuaciones que dependen de las variables de diseño. De forma general un problema de optimización se lo representa de la siguiente forma:

$$\begin{aligned} & \text{minimizar} && f(\mathbf{x}) \\ & \text{sujeta a} && g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m \\ & && h_j(\mathbf{x}) = 0 \quad j = 1, \dots, l \\ & && \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U \end{aligned} \tag{1.1}$$

donde $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ es un vector columna de n variables las cuales reciben el nombre de variables de diseño y pueden ser discretas o continuas, f es la función objetivo, g_i son las restricciones dadas en forma de inecuación, y h_j son las restricciones de igualdad. La notación generalmente usada es \mathbf{x}^0 para el punto inicial, \mathbf{x}^* para el óptimo, y \mathbf{x}^k para el punto actual en la k -ésima iteración. (Vanderplaats, 1999)

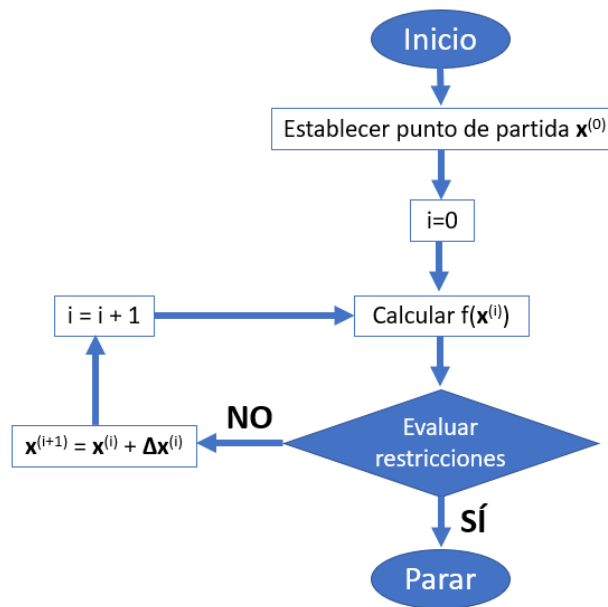


Figura 1.1 Diagrama de flujo general para optimización
[FUENTE: Optimization Methods Review.ppt por Mateusz Sztangret]

Cuando al menos una de las expresiones que definen las restricciones o función objetivo del proceso de optimización son no lineales, es decir de grado mayor o igual que dos, se debe recurrir a programación no lineal o NLP (Nonlinear Programming) para resolver el problema de optimización.

El problema descrito en la ecuación 1.1 puede expresarse como: minimizar $f(\mathbf{x})$, sujeta a \mathbf{x} que pertenece a Ω , donde Ω está definido en la ecuación 1.2.

$$\Omega = \{\mathbf{x} : \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U\} \quad (1.2)$$

Ω , un subconjunto de \mathbf{R}^n , es llamada la región factible. La representación gráfica del espacio de diseño (o espacio \mathbf{x}) para $n=2$ variables se muestra en la figura 1.2. Curvas de valor f constante o contornos de función objetivo se dibujan en ellas, y el óptimo es definido por la curva de contorno más alta pasando a través de Ω , que normalmente, pero no siempre, es un punto en la frontera Ω (Belegundu & Chandrupatla, 2011).

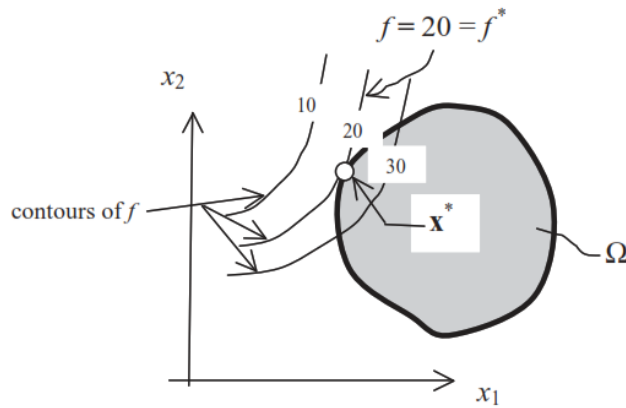


Figura 1.2 Representación gráfica de un NLP en un espacio x

[FUENTE: Optimization Concepts and Applications in Engineering, Ashok Belegundu & Tirupathi Chandrupatla]

Una restricción $g_i(x) \leq 0$ se llama activa en x si $g_i(x) = 0$, e inactiva en x si $g_i(x) < 0$. Las restricciones de igualdad son siempre activas. El conjunto activo en x está conformado por todas aquellas restricciones $g_i(x)$ que son activas en el punto actual. La solución de un problema de optimización multivariable con restricciones puede ser determinada usando el método Lagrangiano. El Lagrangiano aumentado que incluye restricciones de igualdad y desigualdad se define como:

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^m \lambda_i h_i(x) + \sum_{j=1}^l \mu_j g_j(x) \quad (1.3)$$

Cada λ se denomina multiplicador de Lagrange y cada μ se denomina multiplicador de KKT (Karush-Kuhn-Tucker) y constituyen variables adicionales en el problema. Hay cuatro condiciones KKT para variables del óptimo primal (x) y dual (λ, μ).

Se dice que el punto $x \in \mathbb{R}^n$, junto con los vectores de multiplicadores $\lambda \in \mathbb{R}^m$ y $\mu \in \mathbb{R}^l$, verifica las condiciones KKT para el NLP si:

$$\frac{\partial L(x, \lambda, \mu)}{\partial x} = \nabla f(x) + \sum_i \lambda_i \nabla h_i(x) + \sum_j \mu_j \nabla g_j(x) = 0 \quad (1.4)$$

$$\frac{\partial L(x, \lambda, \mu)}{\partial \lambda} = h(x) = 0 \quad (1.5)$$

$$\mu_j(x) \cdot g_j(x) = 0 \quad (1.6)$$

$$\mu_j(x) \geq 0 \quad , \quad g_j(x) \leq 0 \quad (1.7)$$

La primera condición, ecuación 1.4 es conocida como condición estacionaria, y la ecuación 1.5 como condición de igualdad. Por último, las ecuaciones 1.6 y 1.7 se conocen como condiciones de holgura complementaria. De la ecuación 1.5 se obtiene un sistema de ecuaciones de $n+m+l$ ecuaciones con igual número de $n+m+l$ variables \mathbf{x} , $\boldsymbol{\lambda}$ y $\boldsymbol{\mu}$. La condición necesaria de optimalidad es obtener un punto crítico (estacionario) mediante el teorema de KKT y la condición suficiente es que la matriz hessiana de la función de Lagrange aumentada sea definida positiva (Vanderplaats, 1999).

La estrategia del conjunto activo (Active-set strategy) ayuda a la solución del conjunto de ecuaciones KKT de la siguiente manera:

- 1) Definir el conjunto activo
- 2) Formular ecuaciones KKT
- 3) Si para toda \mathbf{j} se cumple la ecuación 1.7, entonces se ha obtenido la solución y si cualquier $g_i(\mathbf{x}) > 0$ y/o $\mu_j(\mathbf{x}) > 0$:
 - a) Eliminar del conjunto activo la restricción con el μ_i más negativo.
 - b) Añadir al conjunto activo todas las restricciones violadas $g_i(\mathbf{x}) > 0$ para hacerlas más activas.
 - c) Regresar a 2).

1.4.1.2 Método de ramificación y acotamiento (Branch and Bound)

El asumir que todas las variables de diseño son continuas delimita el proceso de optimización ya que no puede ser aplicado a una gran variedad de problemas prácticos debido a la incapacidad de los métodos de programación no lineal para manejar variables discretas. Una sugerencia para resolver este tipo de problemas usando programación no lineal, es resolver el problema como si fuese continuo y luego redondear las variables de diseño discretas al valor entero más cercano. Esta sugerencia no siempre es factible ya que la solución que se obtiene redondeando las variables puede ser no óptima o estar en la región no factible del problema.

Un método para resolver problemas con variables discretas y continuas es el de ramificación y acotamiento o BB por sus siglas en inglés (Branch and Bound), que comienza desarrollando una optimización de variable continua. Esto provee un punto de partida, así como el límite inferior en la solución discreta. Luego una sola variable es incrementada al su siguiente valor discreto y la optimización es desarrollada con respecto a las otras variables. Si el valor óptimo es mayor que el anterior, la variable se establece

a su siguiente valor discreto más bajo y el proceso se repite. Si una mejora es encontrada la búsqueda continúa en esa dirección hasta que no se encuentre ninguna mejora. Entonces se permite que la variable cambie, sujeta al límite impuesto actualmente, y el proceso se repite para la siguiente variable, hasta que se hayan examinado todas las variables discretas. (Vanderplaats, 1999).

1.4.1.3 Optimización MINLP

La programación mixta no lineal o MINLP por sus siglas en inglés (Mixed Integer Nonlinear Programming), combina funciones no lineales con el uso de variables mixtas. Muchos de los problemas de optimización en ciencia e ingeniería incluyen una mezcla de variables de diseño enteras y continuas junto con relaciones no lineales dentro de una región factible. La optimización del peso estructural de una embarcación construida con materiales compuestos busca reducir el peso manteniendo la rigidez adecuada mediante la selección óptima de las variables de diseño; las restricciones y función objetivo presentes en el problema son expresiones no lineales, además las variables de diseño son de tipo enteras y reales. Entonces este problema de optimización se clasifica como MINLP.

1.4.2 Requerimientos estructurales de las Normas ISO 12215.

1.4.2.1 Organización Internacional de Normalización (ISO)

ISO por sus siglas en inglés (International Organization for Standardization) es una federación mundial de organismos nacionales de estándares (organismos miembros ISO). El trabajo de preparar estándares internacionales se lleva a cabo normalmente a través de los comités técnicos de la ISO. Cada organismo miembro interesado en un tema por el cual un comité técnico ha sido establecido tiene el derecho de ser representado en ese comité (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008).

1.4.2.2 Uso de la norma ISO 12215

Debido a la baja rigidez flexural, en una embarcación el planchaje del casco, cubierta y caseta es reforzado con cualquier combinación de refuerzos convencionales longitudinales y transversales, y mamparos transversales. Las figuras 1.3 y 1.4 ilustran

los cuadernajes característicos de una embarcación menor construida con PRFV según las normas ISO. En el ejemplo de la figura 1.3 el forro es reforzado por refuerzos longitudinales secundarios soportados por los refuerzos primarios transversales, como bulárcamas, mamparos y varenga de fondo.

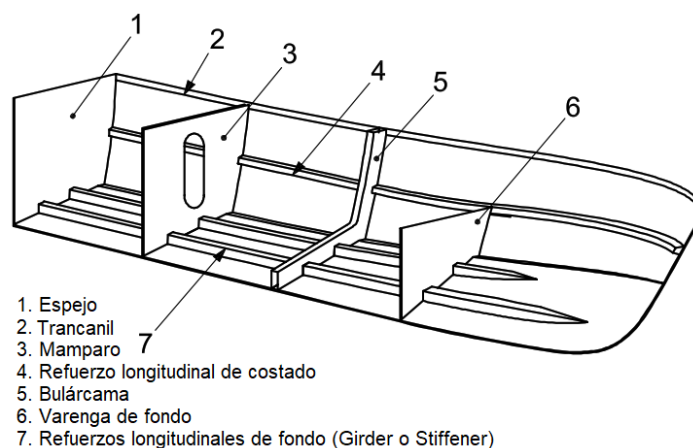


Figura 1.3 Lancha con cuadernaje longitudinal

[FUENTE: Structural Arrangements and Details, ISO 12215-6 (Standardization, Small Craft Hull Construction and Scantlings 12215-6, 2008)

Nota: 1, 3, 5 y 6 son refuerzos primarios; 2,4 y 7 son refuerzos secundarios.

En el ejemplo de la figura 1.4, el forro es reforzado por cuadernas (refuerzos secundarios) que son típicamente soportados en la línea de crujía, chinas o cambio de sección (fondo-costado) y en la conexión con la cubierta. En lanchas mayores, las vagras (refuerzos primarios) pueden ser instaladas, las cuales soportan estas cuadernas y también ayudan a transmitir las cargas a los mamparos transversales del casco.

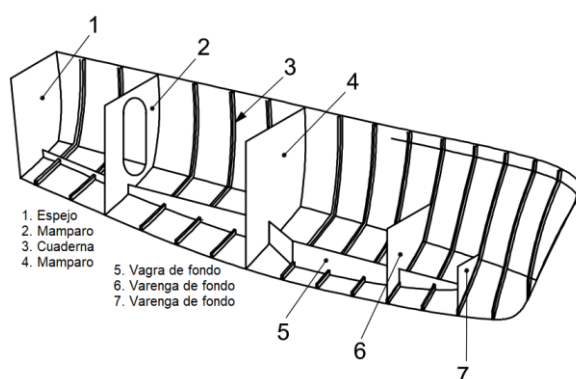


Figura 1.4 Lancha con cuadernaje transversal

[FUENTE: Structural Arrangements and Details, ISO 12215-6 (Standardization, Small Craft Hull Construction and Scantlings 12215-6, 2008)

Nota: 1, 2, 4, 5, 6 y 7 son refuerzos primarios; 3 es refuerzo secundario.

Antes de proceder a la optimización estructural, se va a analizar el nivel de esfuerzos y evaluarlos según los criterios de falla una embarcación rápida dedicada al transporte de pasajeros entre islas en Galápagos construida con plástico reforzado con fibra de vidrio PRFV. Para ello se aplican la parte 5: *Design pressure for monohulls, design stresses, scantlings determination* (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008), y parte 6: *Structural arrangements and details* (Standardization, Small Craft Hull Construction and Scantlings 12215-6, 2008) de las reglas ISO 12215 “Small Craft – Hull Construction and Scantlings Determination”. La lancha prototipo se denominada “lancha A” y sus características principales son:

Tabla 1.1 Características principales - lancha A

| | |
|----------------------|-------|
| Eslora [m] | 12.02 |
| Manga [m] | 3.45 |
| Puntal [m] | 1.51 |
| Calado de diseño [m] | 0.51 |
| Velocidad [nudos] | 25 |
| Desplazamiento [kg] | 8864 |

Para desarrollar la evaluación estructural de la lancha A, se estiman las presiones de diseño en el fondo y costado, y los esfuerzos permisibles en flexión y corte en planchas y refuerzos estructurales, También se comprueba la resistencia al pandeo local y los requerimientos mínimos para prevenir fallas debido a impacto con objetos flotantes.

Las dos telas de fibra de vidrio más usadas en construcción local de botes son Mat y Woven Roving. La tela Mat es un manto de fibra de vidrio en direcciones aleatorias de 300, 350, 450 y 600 gr/m². La tela Woven Roving es un manto tejido de fibra de vidrio bidireccional con densidades de 800 y 600 gr/m². Ambos tipos de telas se muestran en la figura 1.5 y las propiedades mecánicas de ambas telas se encuentran en la tabla 2.13.

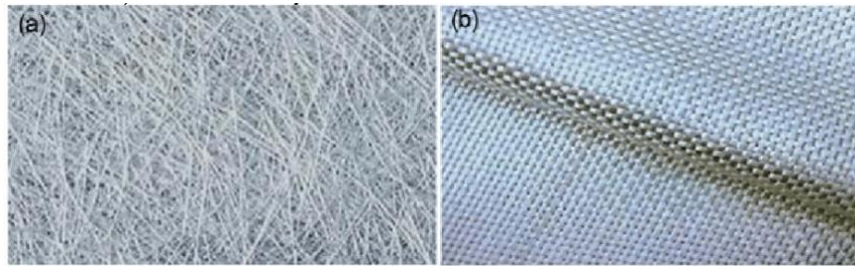


Figura 1.5 Tela Mat (izquierda) y tela Woven Roving (derecha)

[Fuente: Fibers and Polymers, A. M. T. Arifin, S. Abdullah, Md. Rafiquzzaman]

1.4.3 Evaluación de esfuerzos por flexión y por corte en planchas y refuerzos

Para proceder con el análisis de niveles de esfuerzos se utiliza la información estructural disponible de la lancha A:

Tabla 1.2 Datos estructurales - Lancha A

| | |
|---|----------------------|
| Espesor del forro [mm] | 12.5 |
| Laminado del forro | 8 CSM(450)+4 WR(800) |
| Laminado de refuerzos y quilla | 3 CSM (450) |
| Separación refuerzos longitudinales [mm] | 560 |
| Separación refuerzos transversales [mm] | 510 |
| Longitud no soportada refuerzos longitudinales [mm] | 4000 |
| Longitud no soportada refuerzos transversales [mm] | 900 |
| Dimensión refuerzo longitudinal fondo [cm] | 8 x 5 |
| Dimensión refuerzo transversal fondo [cm] | 5 x 5 |
| Dimensión refuerzo palmejar [cm] | 15 x 5 |
| Dimensión refuerzo transversal costado [cm] | 5 x 5 |
| Dimensión refuerzo quilla [cm] | 12 x 5 |

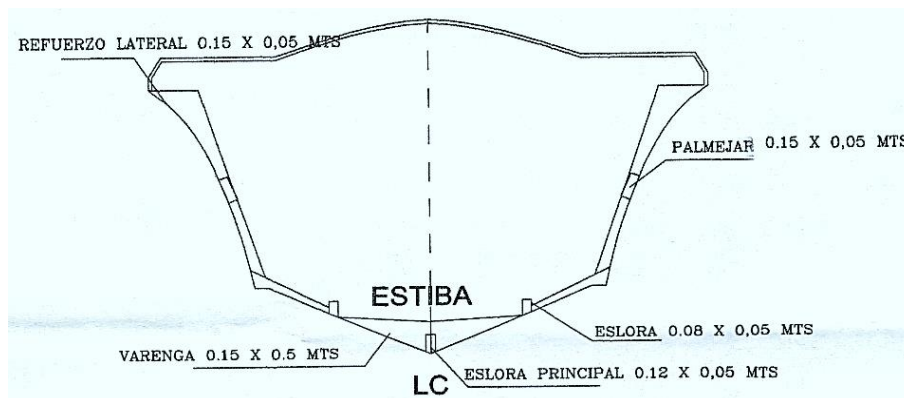


Figura 1.6 Cuaderna maestra - Lancha A

Debido a que las dimensiones de los refuerzos en dirección longitudinal y transversal de la Lancha A son similares, no se toman directamente dichas dimensiones para realizar el análisis. Se sigue la recomendación presentada en el estándar ISO 12215 parte 6 (Standardization, Small Craft Hull Construction and Scantlings 12215-6, 2008) para definir el espaciamiento entre refuerzos y la longitud no soportada de cada refuerzo cuando estos tienen dimensiones similares. Siguiendo los lineamientos propuestos en ISO 12215, se estima la presión de diseño ecuación 1.8, ecuación 1.9 y, ecuación 1.10, ecuación 1.11 para planchas y refuerzos de fondo y costado respectivamente.

$$P_{BMP} = P_{BMP\ BASE} * k_{AR} * k_L \quad kN/m^2 \quad (1.8)$$

$$P_{BM\ MIN} = 0.45m_{LDC}^{0.33} + (0.9 * L_{WL} * k_{DC}) \quad kN/m^2 \quad (1.9)$$

$$P_{SMP} = [P_{DM\ BASE} + k_Z * (0.25 * P_{BMP\ BASE} - P_{DM\ BASE})] * k_{AR} * k_{DC} * k_L \quad kN/m^2 \quad (1.10)$$

$$P_{SM\ MIN} = 0.9 * L_{WL} * k_{DC} \quad kN/m^2 \quad (1.11)$$

$$P_{DM\ BASE} = 0.35 * L_{WL} + 14.6 \quad kN/m^2 \quad (1.12)$$

La presión de diseño de fondo para lanchas a motor planeadoras es la máxima entre P_{BMP} y $P_{BM\ MIN}$, donde $P_{BM\ BASE}$ es la presión base de fondo para lanchas a motor en modo de planeo en kN/m^2 y los factores k 's son adimensionales. La presión de diseño de costado para lanchas a motor planeadoras es la máxima entre P_{SMP} y $P_{SM\ MIN}$

Tabla 1.3 Presiones de diseño

| | | |
|---------|--|-------|
| Costado | Plancha [kN/m ²] | 12.17 |
| | Palmejar [kN/m ²] | 7.76 |
| | Cuaderna [kN/m ²] | 10.72 |
| Fondo | Plancha [kN/m ²] | 47.24 |
| | Refuerzo longitudinal [kN/m ²] | 45.91 |
| | Refuerzo transversal [kN/m ²] | 54.88 |

Una vez estimadas las presiones de diseño, establecido el espaciamiento y longitud no soportada de cada refuerzo, el estándar ISO 12215 parte 5 (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008) recomienda seguir el método presentado en el Anexo H para estimar los niveles de esfuerzos actuando en cada capa del panel y sección del refuerzo. Primero se debe calcular la fuerza cortante en [N/mm] y momento flector en [N mm/mm] ecuación (1.13), ecuación (1.14) para

planchas, respectivamente, y, fuerza cortante en [N] y momento flector en [N.m] ecuación (1.15), ecuación (1.16) para refuerzos respectivamente. En estas ecuaciones, **P** es la presión actuando sobre el elemento estructural, **b** es la dimensión más corta del panel entre refuerzos, **s** y **lu** son el espaciamiento y longitud no soportada de los refuerzos estructurales, mientras que **kc**, **kshc**, **k2** y **kcs** son factores de reducción considerados por la norma ISO 12215.

$$F_d = \sqrt{k_C} * k_{SHC} * P * b * 10^{-3} \quad \text{F. Cortante – plancha} \quad \mathbf{(1.13)}$$

$$M_d = 83.33 * k_C^2 * 2k_2 * P * b^2 * 10^{-6} \quad \text{M. Flector – plancha} \quad \mathbf{(1.14)}$$

$$F_d = 5 * P * s * l_u * 10^{-4} \quad \text{F. Cortante – refuerzo} \quad \mathbf{(1.15)}$$

$$M_d = 83.33 * k_{CS} * P * s * l_u^2 * 10^{-9} \quad \text{M. Flector – refuerzo} \quad \mathbf{(1.16)}$$

Los valores resultantes de las fuerzas internas y momento flectores para planchas y refuerzos son:

Tabla 1.4 Momento flector (Md) y Fuerza cortante (Fd)

| Ubicación / Refuerzo | | M _d | Unidad | F _d | Unidad |
|----------------------|-----------------------|----------------|-----------|----------------|----------|
| Costado | Plancha | 2.67E+02 | [N mm/mm] | 2.67E+00 | [N / mm] |
| | Palmejar | 1,45E+03 | [N m] | 4,16E+03 | [N] |
| | Cuaderna | 5,60E+02 | [N m] | 5,09E+03 | [N] |
| Fondo | Plancha | 9.89E+02 | [N mm/mm] | 1.08E+01 | [N/mm] |
| | Refuerzo longitudinal | 1,14E+03 | [N m] | 8,96E+03 | [N] |
| | Refuerzo transversal | 6,80E+02 | [N m] | 7,56E+03 | [N] |

El Anexo C de ISO 12215 parte 5 presenta relaciones empíricas para estimar las propiedades mecánicas del laminado, como: esfuerzos permisibles y módulo de Young en función del contenido fibra de vidrio y forma de realizar el laminado. El contenido de fibra de vidrio depende de la relación entre el peso por área de tela CSM y el peso por área total del laminado. A continuación, se presentan las propiedades mecánicas del laminado del forro de lancha A según las relaciones presentadas en la norma ISO 12215 parte 5. En este cálculo se ca a asumir la secuencia del laminado del forro y además, que el laminado de los refuerzos es igual ya que no se cuenta con la información detallada. La secuencia de apilamiento SS (Stacking Sequence) de un panel laminado, lista las orientaciones de las fibras medidas desde un eje de referencia del laminado.

Cuando un grupo de capas se repite, entonces el número de repeticiones es usado como un subíndice para el grupo que se repite encerrado en un paréntesis, el subíndice “s” indica que el laminado es simétrico.

$$[\theta_1/(\theta_1/\theta_2/\theta_3)_3 \dots \theta_N]_s \quad (1.17)$$

Tabla 1.5 Propiedades mecánicas del laminado

| | |
|--|--|
| SS Laminado | [CSM ₂ /(0/90/CSM) ₂] _s |
| Contenido de fibra de vidrio | 0.364 |
| Módulo de Young [N/mm ²] | 8842.86 |
| Esfuerzo permisible en tensión [N/mm ²] | 114.02 |
| Esfuerzo permisible en compresión [N/mm ²] | 126.64 |
| Esfuerzo permisible en flexión [N/mm ²] | 173.62 |
| Esfuerzo permisible en corte [N/mm ²] | 67.14 |
| Esfuerzo permisible interlaminar [N/mm ²] | 16.13 |

La rigidez flexural de una viga (**EI**) es un estándar para medir su deformabilidad. Está basada en dos propiedades cruciales: El módulo de elasticidad del material **E** y el momento de inercia (función de geometría), **I**. El módulo seccional (**SM**) es una medida de la resistencia de una viga a la flexión. El flujo cortante (**q**) mide la fuerza cortante por unidad de longitud, y se calcula multiplicando el primer momento de área (**Q**) por la fuerza cortante (**Fd**) y dividiendo para la rigidez flexural. Los beneficios de usar refuerzos con sección tipo sombrero ver figura 1.7 son la alta rigidez flexural y resistencia torsional. Como su nombre lo dice, el refuerzo tiene una forma de sombrero con un ala, alma y un núcleo. La parte superior doblada entre el ala y el alma es reforzada aumentando más capas (preferiblemente sólo MAT).

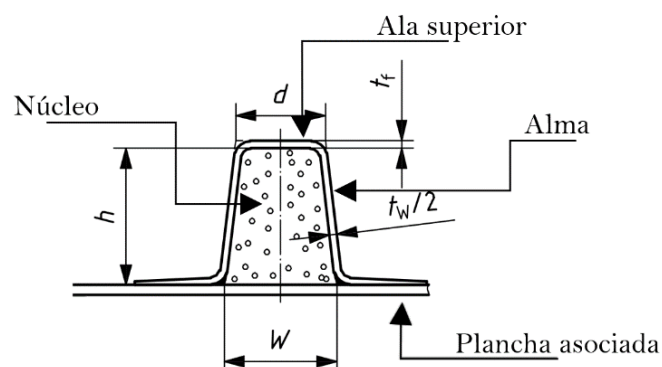


Figura 1.7 Refuerzo estructural tipo sombrero

Fuente: (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008)

El esfuerzo normal en flexión actuando en cada capa de la plancha y sección del refuerzo tipo sombrero es igual al momento flector correspondiente dividido para el módulo seccional respectivo; a su vez el módulo seccional depende de la rigidez flexural alrededor del eje neutro de la plancha o refuerzo dividido para el módulo de Young y la distancia desde el eje neutro al centro de cada capa o centroide de la sección del refuerzo. El esfuerzo debido al corte es igual al flujo cortante dividido para el ancho del área de corte, a su vez el flujo cortante depende de la fuerza cortante y primer momento de área dividido para la rigidez flexural de cada capa o sección del refuerzo. Para el análisis por corte se asume que la plancha es de 1 mm de ancho mientras que para los refuerzos se debe considerar el ancho de las almas y del núcleo efectivo. El estándar ISO 12215 requiere que para un diseño seguro los esfuerzos actuantes debido a las fuerzas y momento flectores no sobrepasen los esfuerzos de diseño en flexión y corte, siendo los esfuerzos de diseño la mitad de los esfuerzos permisibles. En la tabla 1.6 se presentan la relación entre el esfuerzo de diseño y esfuerzo actuante denomina factor de cumplimiento (**CF**) con menor valor para las plancha y refuerzos estructurales, además, de la posición donde ocurre.

Tabla 1.6 Factor de cumplimiento en resistencia a la flexión y al corte en planchas y refuerzos estructurales

| Análisis de planchas | | | | | |
|--|-----------------|---------------------------------------|---|-----------|-----------------|
| | Criterio | $\sigma_d T_d$ N/mm ² | $\sigma_{act} T_{act}$ N/mm ² | CF | Capa |
| Fondo | Flexión | 34,00 | 27,08 | 1,26 | 1 |
| | Corte | 5,64 | 1,28 | 4,40 | 5 |
| Costado | Flexión | 34,00 | 7,71 | 4,41 | 1 |
| | Corte | 5,64 | 0,35 | 15,97 | 5 |
| Análisis de refuerzos estructurales | | | | | |
| | Criterio | $\sigma_d T_d$ N/mm ² | $\sigma_{act} T_{act}$ N/mm ² | CF | Posición |
| Esloras Fondo | Flexión | 34,00 | 4,20 | 8,10 | Ala |
| | Corte | 6,68 | 2,68 | 1,48 | Núcleo |
| | Corte | 24,80 | 16,71 | 1,48 | Alma |
| Cuadernas Fondo | Flexión | 34,00 | 5,25 | 6,47 | Ala |
| | Corte | 6,68 | 1,99 | 1,04 | Núcleo |
| | Corte | 24,80 | 23,80 | 1,04 | Alma |
| Palmejar | Flexión | 34,00 | 4,99 | 6,81 | Ala |

| | | | | | |
|-------------------|---------|-------|-------|-------|--------|
| | Corte | 6,68 | 0,64 | 2,53 | Núcleo |
| | Corte | 24,80 | 9,79 | 2,53 | Alma |
| Cuadernas Costado | Flexión | 34,00 | 4,54 | 10,30 | Ala |
| | Corte | 6,68 | 2,27 | 1,61 | Núcleo |
| | Corte | 24,80 | 15,40 | 1,61 | Alma |

1.4.3.1 Evaluación de la resistencia al impacto contra objetos flotantes y resistencia de la quilla

Otro aspecto importante que se debe tener en cuenta según el estándar ISO 12215 es la capacidad de los paneles del forro de soportar impactos contra objetos flotantes al momento de navegar. Para esto el estándar dispone de la ecuación 1.18 para estimar la masa mínima por unidad de área que debe tener las planchas del fondo, costado, esta expresión depende de la velocidad, el desplazamiento y factores de reducción. Además, el estándar ISO 12215 parte 6 (Standardization, Small Craft Hull Construction and Scantlings 12215-6, 2008) dispone que la plancha asociada a la quilla debe tener al menos 2.2 el peso por área mínimo de la plancha del fondo.

$$w_{min} = 0.43 * k_5 * (A + k_7 * V + k_8 * m_{LDC}^{0.33}) \text{ kg/m}^2 \quad (1.18)$$

A continuación, se presentan los resultados de la masa mínima en kg/m² y su factor de cumplimiento para la lancha A.

Tabla 1.7 Factor de cumplimiento en resistencia al impacto contra objetos flotantes

| Resistencia al impacto | | Requerida | Actual | CF |
|--|---------|-----------|--------|------|
| Mínima fibra masa [kg/m ²] | Fondo | 2.26 | 6.80 | 3.01 |
| | Costado | 1.94 | 6.80 | 3.51 |
| | Quilla | 4.98 | 6.80 | 1.37 |

Otro aspecto que se debe analizar es el módulo seccional de la quilla que debe tener la resistencia suficiente para soportar el peso de la embarcación cuando entre a dique y/o sea remolcada. El estándar ISO 12215 parte 6 presenta una expresión para estimar el módulo seccional mínimo de la quilla, que depende de la eslora total, masa de la embarcación y el esfuerzo permisible en flexión del laminado de la quilla. Se asume que el laminado de la quilla es igual al laminado del fondo de la embarcación.

A continuación, se presenta el factor de cumplimiento de la quilla.

Tabla 1.8 Factor de cumplimiento en resistencia a la flexión de la quilla

| Min. SM [cm ³] | Criterio | Requerido | Actual | CF |
|----------------------------|----------|-----------|--------|------|
| Quilla | Flexión | 159,28 | 402,20 | 2,53 |

Para considerar la resistencia al pandeo local de los refuerzos estructurales (ver *figura 1.6* y *figura 1.7*), el estándar ISO 12215 parte 5 presenta las tablas 20 y 21. Estas tablas especifican las relaciones máximas entre el alto del alma sobre su espesor ($h/(t_w/2)$) y el ancho del ala superior sobre su espesor (d/t_f) cuando el esfuerzo actuando es igual (tabla 20) o menor (tabla 21) al esfuerzo diseño en compresión del laminado. Debido a que el laminado de los refuerzos de la lancha A es desarrollado con materiales disimilares, se realiza una corrección con respecto a las relaciones máximas presentadas en las tablas anteriores. Para ello se utiliza la expresión general para el pandeo de plancha simplemente soportada en sus bordes considerando que el laminado del refuerzo es especialmente ortotrópico, es decir cada lamina es orientada a 0 o 90 grados (Barbero, 2011). A continuación, se presentan los resultados obtenidos.

Tabla 1.9 Factor de cumplimiento para el pandeo local de refuerzos estructurales

| Ala superior (d/t_f) | Criterio | Max. | Actual. | CF | Refuerzo |
|--------------------------|------------------------------|-------|---------|------|--------------|
| Refuerzos del Fondo | Pandeo local en ala superior | 23,00 | 9,93 | 2,32 | Longitudinal |
| | | 23,00 | 9,93 | 2,32 | Transversal |
| Refuerzos de Costado | | 23,00 | 22,78 | 1,00 | Longitudinal |
| | | 23,00 | 9,93 | 2,32 | Transversal |
| Almas ($h / (t_w/2)$) | Criterio | Máx. | Actual. | CF | Refuerzo |
| Refuerzos del Fondo | Pandeo por corte en el alma | 31,00 | 25,36 | 1,22 | Longitudinal |
| | | 31,00 | 15,85 | 1,96 | Transversal |
| Refuerzos de Costado | | 31,00 | 15,85 | 1,96 | Longitudinal |
| | | 31,00 | 15,85 | 1,96 | Transversal |

Finalmente se analiza la resistencia a la rigidez esto es, oposición a la deflexión de los refuerzos estructurales del fondo y costado. El estándar ISO 12215 parte 5 propone un valor mínimo para la rigidez flexural (EI) de los refuerzos estructurales, ecuación 2.34, que depende de la separación entre refuerzos, longitud no soportada del refuerzo, presión de diseño y deflexión relativa permisible. A continuación, se presenta los resultados obtenidos en lancha A.

Tabla 1.10 Factor de cumplimiento de la rigidez flexural (EI) de los refuerzos estructurales

| | Criterio | Mín. | Actual | CF | Refuerzos |
|-----------------------|---------------------------------------|-------------|---------------|-----------|------------------|
| Refuerzos de Fondo | Rigidez Flexural [N mm ²] | 5,45E+08 | 9,78E+10 | 179,39 | Longitudinal |
| | | 2,29E+08 | 3,29E+10 | 143,72 | Transversal |
| Refuerzos del Costado | | 1,91E+09 | 6,22E+10 | 32,61 | Longitudinal |
| | | 2,31E+08 | 3,01E+10 | 130,28 | Transversal |

Como se puede observar en la tabla 1.10 el factor de cumplimiento que en este caso es la relación entre la rigidez flexural mínima requerida por las normas ISO y la rigidez flexural actual de los refuerzos de la lancha A; se obtiene valores elevados. Esto indica sobredimensionamiento de la estructura actual.

CAPÍTULO 2

2. OPTIMIZACIÓN ESTRUCTURAL DEL FONDO Y COSTADO

2.1 Proceso de Optimización para el fondo y costado

El proceso de optimización del módulo central de la estructura de la lancha A consta de dos fases, basándonos en los trabajos realizados por (Maneeapan, Sheno, & Blake, 2007) y (Sobey, Blake, & Sheno, 2013) : la primera optimiza el laminado de las planchas y número de refuerzos, mientras que, la segunda optimiza el laminado y geometría de los refuerzos. Como resultado de la primera fase, se tienen el número de refuerzos en dirección longitudinal y transversal, así como el laminado del forro. La segunda fase se encarga de optimizar el laminado de los refuerzos estructurales, así como las dimensiones de los mismo.

2.1.1 Simplificaciones del laminado

Para simplificar el proceso de optimización se consideraron las siguientes asunciones de carácter práctico, de acuerdo con la práctica común en el medio:

- El laminado de los paneles es simétrico.
- Los tipos fibra de vidrio empleados son: Chopped Strand Mat (CSM), Woven Roving (WR).
- La secuencia del laminado para planchas es $[CSM_2 / (WR/CSM)_n]_s$ donde n es el número de capas del sub-laminado WR/CSM.
- El laminado de los refuerzos está compuesto sólo de telas CSM.
- Los gramajes (peso por unidad de área) de CSM son: 300, 350, 450, 600 g/m².
- Los gramajes (peso por unidad de área) de WR son: 600 y 800 g/m².

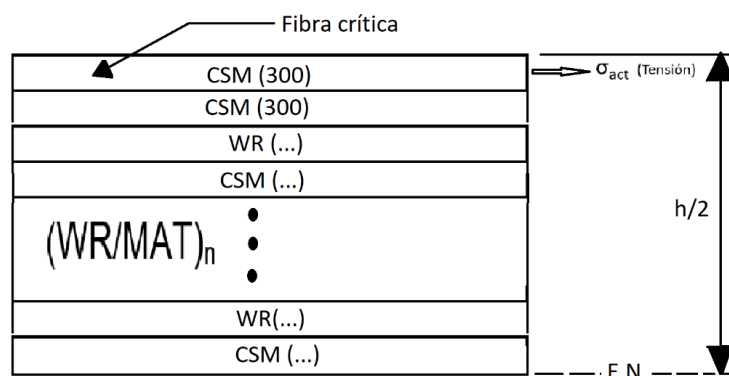


Figura 2.1 Esquema de la mitad del laminado simétrico

Dado que el laminado de las planchas es simétrico, el módulo seccional SM_{ANA} en la fibra más alejada por encima del eje neutro es igual al módulo seccional SM_{BNA} en la fibra más alejada por debajo del eje neutro. Esto, equivale a que la magnitud del esfuerzo debido al momento flector en tensión sea igual a la magnitud del esfuerzo en compresión. Además, según las propiedades mecánicas de los laminados que propone la Norma ISO 12215 el esfuerzo permisible en tensión (σ_{uT}) es menor que el esfuerzo permisible en compresión (σ_{uC}) es decir, se espera una falla por tensión antes que por compresión.

2.1.2 Constantes y variables de diseño

Las constantes que fueron tomadas en cuenta en el proceso de optimización son las propiedades mecánicas de los materiales usados: fibra, resina y madera. Dichas constantes fueron obtenidas con las expresiones del Anexo C de la ISO 12215 parte 5; para las propiedades de la madera se asumieron que corresponde a la madera tipo chanul (*Humiriastrum Procerum*), y sus propiedades fueron tomadas de ITTO (Organization, 2020)

Tabla 2.1 Constantes directas e indirectas

| | |
|----------------------------|--|
| ρ_M | Densidad de la tela CSM [g/m ³] |
| ρ_W | Densidad de la tela WR [g/m ³] |
| ρ_R | Densidad de la resina [g/m ³] |
| ρ_C | Densidad de la madera usada como núcleo [kg/m ³] |
| φ_M | Contenido de fibra de la tela CSM igual a 0.3 |
| φ_W | Contenido de fibra de la tela WR igual a 0.48 |
| E_C | Módulo de Young de la madera usada como núcleo [MPa] |
| E_r | Módulo de Young de la tela CSM [MPa] |
| σ_{UT} | Esfuerzo de tensión último [MPa] |
| T_M | Resistencia última al corte de la tela CSM [MPa] |
| T_W | Resistencia última al corte de la tela WR [MPa] |
| T_{iM} | Esfuerzo de corte interlaminar último de la tela CSM [MPa] |
| T_{iW} | Esfuerzo de corte interlaminar último de la tela WR [MPa] |
| $(t_i), \quad i = 1,2,3,4$ | Espesor de las capas CSM (300, 350, 450 y 600 g/m ²) respectivamente |
| $(t_i), \quad i = 5,6$ | Espesor de las capas WR (600 y 800 g/m ²) respectivamente |
| $(w_i), \quad i = 1,2,3,4$ | Gramaje de las capas CSM (300, 350, 450 y 600 g/m ²) respectivamente |
| $(w_i), \quad i = 5,6$ | Gramaje de las capas WR (600 y 800 g/m ²) respectivamente |

Los espesores y gramaje de cada capa también son constantes ya que vienen fijados por los fabricantes de telas de fibra de vidrio. La ecuación 2.1 muestra el cálculo del espesor de cada capa que depende de la resina, contenido de fibra y del gramaje de cada capa.

$$t = \frac{w}{\rho_F \rho_R} * \left(\frac{\rho_R}{\varphi} - (\rho_R - \rho_F) \right) \quad (2.1)$$

ρ_F : Densidad de la fibra

ρ_R : Densidad de la resina

w : Gramaje de la tela

φ : Contenido de fibra de la tela ($Mat = 0.3$ y $WR = 0.48$)

Las propiedades mecánicas de la madera utilizada en la lancha A se presentan a continuación:

Tabla 2.2 Propiedades mecánicas del Chanul (*Humiriastrum Procerum*)

| | |
|---|---------|
| E_c [MPa] | 1.795E4 |
| ρ_c [g/cm ³] | 0.87 |
| Resistencia a la flexión [MPa] | 144.35 |
| Compresión paralela a la fibra [Mpa] | 74.92 |
| Compresión perpendicular a la fibra [MPa] | 9.81 |

En la primera fase, las variables de diseño son el número de refuerzos en ambas direcciones; se considera que las zonas del fondo y costado tienen el mismo número de refuerzos transversales para proveer continuidad del anillo transversal estructural. También se establecieron como variables el número de capas por cada gramaje tanto de la tela CSM como de WR. Entonces se tiene un, de 6 variables enteras para el número de capas para fondo y 6 variables para las capas del costado.

Tabla 2.3 Variables de diseño para la primera fase

| | |
|-----------------------------|--|
| NT | Número de refuerzos transversales |
| NLB | Número de refuerzos longitudinales (fondo) |
| NLS | Número de refuerzos longitudinales (costado) |
| (nb_i) , $i = 1,2,3,4$ | Número de capas CSM en planchaje del fondo con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| (nb_i) , $i = 5,6$ | Número de las capas WR en planchaje de fondo con gramaje (600 y 800 g/m ²) respectivamente. |

| | |
|----------------------------|--|
| $(ns_i),$ $i = 1,2,3,4$ | Número de capas CSM en planchaje del costado con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| $(ns_i),$ $i = 5,6$ | Número de las capas WR en planchaje de costado con gramaje (600 y 800 g/m ²) respectivamente. |

Así como en la primera fase, la segunda fase tiene como variables de diseño el número de capas para el laminado, pero en este caso de los refuerzos en ambas direcciones y la quilla sólo considera capas de tipo CSM. También se consideraron como variables de diseño las dimensiones principales de los refuerzos y la quilla: la altura **h** y el ancho de la base del núcleo **b** del refuerzo tipo sombrero ver la figura 1.7. El espesor del laminado de los refuerzos y quilla depende del número de capas de tela CSM de cada gramaje.

Tabla 2.4 Variables de diseño para la segunda fase

| | |
|-----------------------------|---|
| $(nbl_i),$ $i = 1,2,3,4$ | Número de capas CSM en refuerzos longitudinales del fondo con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| $(nsl_i),$ $i = 1,2,3,4$ | Número de capas CSM en refuerzos longitudinales del costado con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| $(nt_i),$ $i = 1,2,3,4$ | Número de capas CSM en refuerzos transversales con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| $(n_k),$ $i = 1,2,3,4$ | Número de capas CSM en la quilla con gramaje de (300, 350, 450 y 600 g/m ²) respectivamente. |
| h_{lb} | Alto del alma de refuerzos longitudinales del fondo |
| b_{lb} | Ancho de la base del núcleo de refuerzos longitudinales del fondo |
| h_{ls} | Alto del alma de refuerzos longitudinales del costado |
| b_{ls} | Ancho de la base del núcleo de refuerzos longitudinales del costado |
| h_t | Alto del alma de refuerzos transversales |
| b_t | Ancho de la base del núcleo de refuerzos transversales |
| h_k | Alto del alma de la quilla |
| b_k | Ancho de la base del núcleo de la quilla |

2.1.3 Expresiones para las restricciones

Fase 1:

La primera restricción es para el sub-laminado [CSM,WR]_n, este define el laminado del forro, como se evidencia en la figura 2.1 el esquema que se consideró para el proceso de optimización es un laminado simétrico con combinación intercalada entre tela CSM y WR. Debido a la condición de simetría el número de capas de CSM dentro del sub-laminado deben ser igual al número de capas de WR.

$$h1: \quad \sum CSM - \sum WR = 0 \quad (2.2)$$

Otro aspecto importante que se consideró en el proceso de optimización es que según la experiencia de construcción local las dos primeras y últimas capas del laminado del forro son realizadas con tela CSM de menor gramaje posible, en este caso se utilizó CSM 300. Por consiguiente, la capa más alejada que soporta el esfuerzo de tensión es CSM-300 con la siguiente expresión para calcular el módulo seccional SM_{ANA} .

$$SM_{ANA} = \frac{EI_{NA}}{Z_{cri} * E_i} \quad (2.3)$$

EI_{NA} : Rigidez flexural de todo el laminado con respecto al eje neutro.

E_i : Módulo de Young de la capa más alejada al eje neutro.

La distancia desde el eje neutro se calcula con la siguiente expresión:

$$Z_{cri} = \frac{1}{2}(h - t_1) \quad (2.4)$$

h : Espesor total del laminado. (ver fig. 2.1)

t_1 : Espesor de la capa más alejada del eje neutro.

Para estimar la rigidez flexural EI del laminado se obtuvo la media geométrica del módulo de Young del laminado y se lo multiplica por la inercia seccional del mismo. Recordando que el laminado de las planchas está compuesto solo de tela MAT y WR con módulo de Young 6400 y 13240 N/mm² respectivamente. La ecuación 2.7 que define el módulo seccional de la fibra más alejada se obtuvo reemplazando ecuación 2.4, 2.5 y 2.6 en ecuación 2.3.

$$E = \left[(6400^{(4+2*\sum_1^3 n_i)} * 13240^{(2*\sum_4^5 n_i)})^{\frac{1}{n_T}} \right] \quad (2.5)$$

$$I_{NA} = \frac{bh^3}{12} \quad (2.6)$$

$$SM_{ANA} = \frac{\left[(6400^{(4+2*\sum_1^3 n_i)} * 13240^{(2*\sum_4^5 n_i)})^{\frac{1}{n_T}} \right] * (2t_1 + \sum_1^5 n_i t_i)^3}{9600 * (t_1 + \sum_1^5 n_i t_i)} \quad (2.7)$$

Los criterios de falla que se tuvieron que cumplir para la fase 1 son la resistencia por flexión y por impacto. El criterio de falla por flexión (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008) establece que el esfuerzo normal actuando en la capa más alejada del eje neutro debido al momento flector M_d no supere al esfuerzo de diseño ($\sigma_d = 0.4 * \sigma_{uT}$) de aquella capa.

$$\sigma_{act} = \frac{M_d}{SM_{ANA}} \quad (2.8)$$

$$\frac{\sigma_d}{\sigma_{act}} - 1 \geq 0 \quad (2.9)$$

Reemplazando la ecuación 2.8 en ecuación 2.9, la primera restricción en forma de inecuación se expresa de la siguiente forma:

$$g_1: \frac{0.4 * \sigma_{uT} * \left[(6400^{(4+2*\sum_1^3 n_i)} * 13240^{(2*\sum_4^5 n_i)})^{\frac{1}{n_T}} \right] * (2t_1 + \sum_1^5 n_i t_i)^3}{9600 * (t_1 + \sum_1^5 n_i t_i) * M_d} - 1 \geq 0 \quad (2.10)$$

El criterio de falla por corte interlaminar no fue considerado ya que según la ISO 12215 no es crítico para un laminado simple a diferencia de laminados con núcleos debido a que los espesores son muy pequeños para el cálculo de flujo cortante. (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008). Para el criterio de falla por impacto, la expresión propuesta por ISO 12215, depende de la velocidad y el desplazamiento, que especifica el peso por área del panel suficiente para soportar las cargas por impactos de objetos flotantes es:

$$w_{min} = 0.0675 + 0.0135 * V + 0.0675 * m_{LDC}^{0.33} \quad (2.11)$$

Por lo tanto, la segunda restricción en forma de inecuación viene dada de la siguiente manera:

$$\frac{W_{disponible}}{W_{minimo}} - 1 \geq 0 \quad (2.12)$$

$$g_2: \frac{4 * w_1 + 2 * \sum_1^5 n_i w_i}{0.0675 + 0.0135 * V + 0.0675 * m_{LDC}^{0.33}} - 1 \geq 0 \quad (2.13)$$

Fase 2:

La fase 2 pretende la optimización del laminado y la geometría de los refuerzos, como primera restricción se establece que la altura **h** del refuerzo debe ser mayor o igual al ancho de la base del núcleo **b** para obtener la mayor rigidez ocupando menos espacio en la plancha.

$$g_3: \frac{h}{b} - 1 \geq 0 \quad (2.14)$$

Además, los criterios de falla comprendidos para optimizar los refuerzos son:

- I. Esfuerzo normal actuando en el ala debido al momento flector.
- II. Esfuerzo cortante actuando en el alma debido a la fuerza cortante.
- III. Esfuerzo cortante actuando en el núcleo debido a la fuerza cortante.
- IV. Mínima rigidez flexural de los refuerzos
- V. Las razones entre la altura del alma y su espesor (**h/t**), y, el ancho del ala y su espesor (**b/t**) requeridas para evitar pandeo local.

Para encontrar la expresión de la restricción relacionada al criterio de falla por flexión, fue necesario el cálculo del momento flector sobre los refuerzos con la ecuación 1.13. Dividiendo el momento flector ya calculado con el esfuerzo último de tensión de la tela CSM se obtiene el módulo seccional requerido en el ala:

$$SM_{req} = \frac{M_d}{0.4 \cdot \sigma_{ut}} \quad (2.15)$$

Utilizando la ecuación 2.3 para calcular el módulo seccional actual del ala (SM_{act}) y estableciendo que sea mayor al módulo seccional requerido, se obtiene la segunda restricción de la fase 2.

$$g_4: \frac{SM_{act}}{SM_{req}} - 1 \geq 0 \quad (2.16)$$

Para evaluar el criterio de falla por corte se comparó las áreas de corte del alma y del núcleo con aquellas requeridas usando el esfuerzo último de corte de la tela CSM y de la madera chanul respectivamente.

$$A_{Wreq} = \frac{F_d}{\tau_{uMAT}} \quad (2.17)$$

$$A_{Creq} = \frac{F_d}{\tau_{uCOR}} \quad (2.18)$$

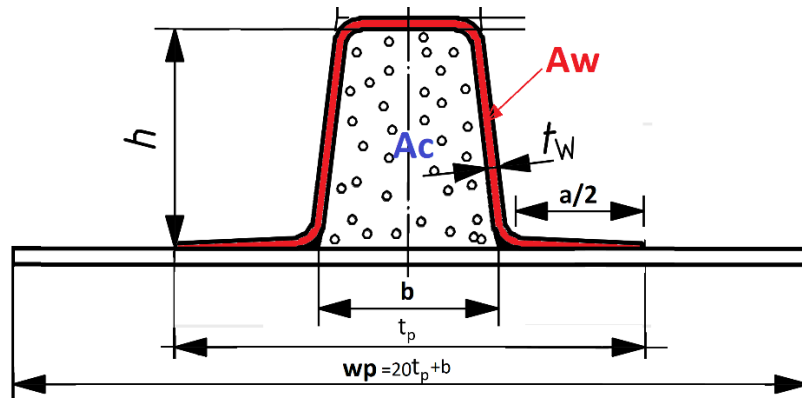


Figura 2.2 Áreas de corte y dimensiones principales de refuerzos tipo sombrero.

El refuerzo se une a la plancha con ayuda de las alas inferiores, estas se crean mediante el traslapado del laminado del refuerzo donde la primera capa tiene 25 mm de ancho y luego traslapado 15 mm por cada nueva capa del laminado del refuerzo. La expresión para el ancho $a/2$ ver figura 2.2 viene dada por la ecuación 2.19, cabe recalcar que esta expresión es solo para estimar el ancho de un solo lado del ala inferior del refuerzo, por lo que para considerar ambos lados en las relaciones posteriores se la multiplicó por 2.

$$\frac{a}{2} = 25 + (n_1 + n_2 + n_3 + n_4 - 1) \cdot 15 \quad (2.19)$$

Las ecuaciones de los primeros momentos de área del núcleo y alma son, respectivamente:

$$Q_C = 0.5 \cdot E_c \cdot b \cdot (h + t_p - NA)^2 + E_r [Z_{cr} t_w (b + 2t_w) + t_w (h + t_p - NA)^2] \quad (2.20)$$

$$Q_W = E_p (NA - 0.5t_p) \cdot wp \cdot t_p + E_r [(NA - t_p - 0.5t_w) \cdot a \cdot t_w] \quad (2.21)$$

Para obtener la rigidez flexural total del refuerzo (ecuación 2.28) y con respecto al eje neutro (ecuación 2.29) se aplican las siguientes expresiones en función de las variables de diseño:

$$\sum EA = E_P w_p t_p + E_C b h + E_R t (2a + 2h + b + 2t) \quad (2.22)$$

$$\begin{aligned} \sum EAZi = E_P w_p t_p \left(\frac{t_p}{2} \right) + E_C b h \left(\frac{h}{2} + t_p \right) \\ + E_R t \left(2a \left(t_p + \frac{t}{2} \right) + 2h \left(t_p + \frac{h}{2} \right) + (b + 2t) \left(t_p + h + \frac{t}{2} \right) \right) \end{aligned} \quad (2.23)$$

$$NA = \frac{\sum EAZi}{\sum EA} \quad (2.24)$$

$$Z_{cri}(top\ flange) = t_p + h - NA \quad (2.25)$$

$$\begin{aligned} \sum EAZi^2 = E_P w_p t_p \left(\frac{t_p}{2} \right)^2 \\ + E_C \left(b(NA - t_p) \left(\frac{NA + t_p}{2} \right)^2 + b(h - NA + t_p) \left(\frac{h + NA - t_p}{2} \right)^2 \right) \\ + E_R \left(2t(NA - t_p) \left(\frac{NA + t_p}{2} \right)^2 \right. \\ \left. + 2t(h - NA + t_p) \left(\frac{NA + h + t_p}{2} \right)^2 \right) \\ + E_R \left(2at \left(t_p + \frac{t}{2} \right)^2 + (b + 2t)t \left(t_p + h + \frac{t}{2} \right)^2 \right) \end{aligned} \quad (2.26)$$

$$\begin{aligned} \sum \frac{Ebh^3}{12} = \left(\frac{1}{12} \right) \left(E_P w_p t_p^3 + E_C b \left((h - NA + t_p)^3 + (NA - t_p)^3 \right) \right. \\ \left. + E_R 2t \left((h - NA + t_p)^3 + (NA - t_p)^3 \right) + E_R (2at^3 + (b + 2t)t^3) \right) \end{aligned} \quad (2.27)$$

$$\sum EI = \sum EAZi^2 + \sum \frac{Ebh^3}{12} \quad (2.28)$$

$$EI_{NA} = \sum EI - NA^2 \sum EA \quad (2.29)$$

Una vez definida la rigidez flexural con respecto al eje neutro, junto con el primer momento del área y espesor del alma y núcleo, se calcularon las áreas de corte respectivamente:

$$A_W = \frac{2EI_{NA} \cdot t_w}{Q_W} \quad (2.30)$$

$$A_c = \frac{EI_{NA} \cdot b}{Q_c} \quad (2.31)$$

Las expresiones de las restricciones para controlar el criterio de falla por corte usando las áreas del alma (**A_w**) y núcleo (**A_c**), son:

$$g_5: \quad \frac{A_w}{A_{wreq}} - 1 \geq 0 \quad (2.32)$$

$$g_6: \quad \frac{A_c}{A_{creq}} - 1 \geq 0 \quad (2.33)$$

La restricción que controla la deflexión se establece comparando la rigidez flexural calculada respecto al eje neutro con la rigidez mínima requerida por la norma ISO 12215.

$$\sum (E_{TC} * I) = \frac{26 * k_{CS}^{1.5} * P * s * l_u^3 * 10^{-11}}{k_{1s}} \quad (2.34)$$

$$g_7: \quad \frac{EI_{NA}}{EI_{MÍN}} - 1 \geq 0 \quad (2.35)$$

Las últimas dos restricciones están relacionadas con el criterio de falla por pandeo del panel del laminado que forma el ala y alama de los refuerzos, mediante razones de esbeltez máximos del alma y ala dadas en la tabla 20 de ISO 12215 parte 5 (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008)

$$g_8: \quad 1 - \frac{h}{30t_w} \geq 0 \quad (2.36)$$

$$g_9: \quad 1 - \frac{b}{21t_w} \geq 0 \quad (2.37)$$

Quilla:

Para el proceso de optimización de la quilla se consideró como restricción el valor mínimo del módulo seccional de la quilla para evitar falla durante el remolque de la embarcación fuera de agua. Para calcular este parámetro se siguió el mismo esquema empleado en la optimización de los refuerzos, utilizando como variables de diseño, la altura **h_k** y ancho de la base **b_k** del refuerzo tipo sombrero, y, el número de capas de laminado tipo CSM para cada gramaje (300, 350, 450 y 600 g/m²), es decir se consideraron 6 variables de diseño. Según ISO 12215 parte 6 el mínimo módulo seccional es:

$$SM_{KEEL} = 1.4 * 10^{-3} * \frac{130 * m_T * L_H}{\sigma_{fu}} \quad (2.38)$$

$$g_{11}: \quad \frac{SM_{KEEL}}{SM_{kmin}} - 1 \geq 0 \quad (2.39)$$

2.1.4 Función Objetivo

Para minimizar el peso total de los paneles del fondo, costado y la quilla, la función objetivo es la suma del peso del laminado y núcleo, de los refuerzos transversales, (\mathbf{W}_T), y longitudinales, (\mathbf{W}_L), de la plancha (\mathbf{W}_p) y la quilla, (\mathbf{W}_k), a su vez los pesos de los refuerzos dependen de su cantidad, dimensiones y gramaje total. El peso de la plancha y quilla dependen de sus dimensiones seccionales y del gramaje total:

$$W_L = \frac{\sum_{i=1}^4 n l_i w_i}{0.3} NL \cdot l_l \cdot (2h_l + b_l + a_l) + h_l \cdot b_l \cdot l_l \cdot NL \cdot \rho_c \quad (2.40)$$

$$W_T = \frac{\sum_{i=1}^4 n t_i w_i}{0.3} NT(NL + 1) \cdot l_t \cdot (2h_t + b_t + a_t) + h_t \cdot b_t \cdot l_t \cdot NT(NL + 1) \cdot \rho_c \quad (2.41)$$

$$W_K = l_p \left[2h_k + b_k + \left(25 + 15 \left(\sum_{i=1}^4 KMat_i - 1 \right) \right) + h_k b_k d_{keel} \right] \quad (2.42)$$

$$W_P = \frac{4w_1}{0.3} + \frac{2 \sum_{i=1}^4 n_i w_i}{0.3} + \frac{\sum_{i=5}^6 n_i w_i}{0.48} \quad (2.43)$$

$$W(l_p, b_p, W_P, W_T, W_L, W_k) = l_p b_p G_p + W_T + W_L + W_K \quad (2.44)$$

l_p : Largo de la plancha

b_p : Ancho de la plancha

l_l : Longitud de refuerzos longitudinales

l_t : Longitud de refuerzos transversales

Finalmente, el problema de optimización para minimizar el peso de un módulo de una lancha rápida de fibra de vidrio quedó esquematizado de la siguiente manera:

$$\begin{aligned} & \text{minimizar} && W(l_p, b_p, W_P, W_T, W_L, W_k) \\ & \text{sujeta a} && g_i(n_j, w_j, t_j, h_j, b_j, h_k, b_k) \leq 0 \quad i = 1, \dots, 11 \quad j = l, t \\ & && y \quad h(n_{csm}, n_{wr}) = 0 \\ & && y \quad n^L \leq n \leq n^U \end{aligned} \quad (2.45)$$

2.2 Implementación del código de programación

Se desarrolló el código en el lenguaje de programación **Python** a través de **Gekko**, que es un paquete para aprendizaje automático o (*Machine learning*) y optimización. **Gekko** es una librería orientada a objetos para facilitar la ejecución de **APMonitor** software de optimización para ecuaciones con mezcla de variables enteras y continuas y ecuaciones diferenciales algebraicas. Para la programación en lenguaje Python se requiere la instalación y/o importación de diferentes librerías cuando se resuelven problemas matemáticos, ingenieriles, estadísticos, etc.

2.2.1 Importación de módulos y librerías

Se utilizan los siguientes módulos y librerías.

- math:* Provee acceso a funciones matemáticas definidas por el estándar C.
- glob:* Encuentra todas las direcciones de archivos emparejando un patrón específico usado por Unix Shell.
- os:* Provee una manera portátil de usar la funcionalidad dependiente del sistema operativo.
- pandas:* (Python Data Analysis Library), toma datos (csv, tsv o base de datos sql) y crea un objeto py.
- numpy:* Constituye una biblioteca que agrega mayor soporte para operar con vectores y matrices
- matplotlib:* Crea visualizaciones estáticas, animadas e interactivas en Python.
- gekko:* Librería usada para el proceso de optimización; requiere la definición variable, restricciones y función objetivo.
- time:* Provee varias funciones relacionadas con el tiempo de ejecución en Python.

2.2.2 Datos de entrada

Los datos de entrada son aquellas características definidas de la embarcación, dimensiones de la plancha en el plano, propiedades mecánicas y físicas ingresadas por el usuario que no cambian durante la ejecución del programa.

Tabla 2.5 Características de la embarcación

| | |
|-----|------------------------------------|
| LH | Eslora total de la embarcación [m] |
| Lwl | Eslora en la línea de agua [m] |

| | |
|-------|--|
| BC | Manga al nivel de la china a 0.4 LH desde popa [m] |
| beta | Ángulo de astilla muerta a 0.4 LH desde popa [m] |
| V | Velocidad máxima [knots] |
| mLDC, | Desplazamiento de diseño [kg] |
| T | Calado [m] |
| D | Puntal [m] |

Tabla 2.6 Dimensiones de paneles del fondo y costado

| | |
|----|---|
| Lp | Dimensión de ambos paneles en dirección longitudinal [m] |
| B | Dimensión del panel del fondo en dirección longitudinal [m] |
| Bs | Dimensión del panel del costado en dirección longitudinal [m] |

Tabla 2.7 Propiedades mecánicas de los materiales de construcción

| | |
|----------|---|
| SigUT | Esfuerzo último de tensión de tela CSM [MPa] |
| Taoi_MAT | Esfuerzo último interlaminar de tela CSM [MPa] |
| Taoi_WR | Esfuerzo último interlaminar de tela WR [MPa] |
| Tao_MAT | Esfuerzo último de corte de tela CSM [MPa] |
| Tao_Core | Esfuerzo último de corte del núcleo [MPa] |
| Ec | Módulo de Young del núcleo en dirección de la fibra [MPa] |
| Er | Módulo de Young de la tela CSM en dirección de la fibra [MPa] |

Tabla 2.8 Densidades y gramajes de materiales de construcción

| | |
|-------|--|
| dres | Densidad de la resina poliéster [g/cm ³] |
| dmat | Densidad de las fibras CSM [g/cm ³] |
| dwr | Densidad de las fibras WR [g/cm ³] |
| Wmat | Lista que contiene los gramajes de tela CSM [g/m ²] |
| Wwr | Lista que contiene los gramajes de tela WR [g/m ²] |
| dkeel | Densidad de núcleo de madera de la quilla [kg/m ³] |
| dcore | Densidad de núcleo de madera en los refuerzos [kg/m ³] |

2.2.3 Funciones desarrolladas

Las funciones desarrolladas para el proceso de optimización ayudan a ordenar la secuencia de cálculos utilizando el formato propuesto por la librería Gekko para definir variables, constantes, parámetros, condicionales y expresiones intermedias; estas últimas son expresiones implícitas en función a las variables de diseño que el solver de

Gekko recalcula por cada iteración. El nombre de las funciones debe empezar en minúscula por nomenclatura de python.

Tabla 2.9 Funciones desarrolladas para el fondo, costado y quilla

| Región | Función / entrada | Salida |
|---------|---|--|
| Fondo | span_space_stiff (nt,nl) | st, lul, sl |
| | stiff_force (lul,sl,st) | pbt,pbl,fdt,fdl,mdt,mdl |
| | plate_force (bl,bt,lut,st) | pbp, fdp, mdp, l1, b1 |
| | bottom_optimize (lp,bp,nt,nl,h,b,hT,bT, st,lul,sl,pbt,pbl,fdt,fdl,mdt,mdl,mdp) | TOTAL_WEIGHT,PANEL_WEIGHT, STIFF_WEIGHT,C1L,C2L,C3L,EIL, C45,C46,C1T,C2T,C3T,EIT,C47,C48, Sact,C1P,C2P,C3P,L_Weight, T_Weight,B_T,Ep,twl,twc,B_T |
| Quilla | keel_optimize (KMat,Tmat,Wmat, b_t,Ep,Ec,dkeel,mdl,fdl,pbp,sl,lp) | SMkeel, SMkmin, Keel_weight, C1K,SML,SM_Req_Long,SML, C2K, C3K, EIK, C45K, C46K, kthick |
| Costado | span_space_stiff_s (nls,st) | sls,luls |
| | stiff_force_s (luls,sls,pb_l,d,t) | psl,fdsl,mdsl |
| | plate_force_s (bls,sls,b1ss,pb_p,d,t) | psp, fdps, mdps, l1s, b1s |
| | side_optimize (nt,twl,twc,nls,sls,hs,bs, psl,fdsl,mdsl,mdps,luls,b_s,bside,psp,st): | TOTAL_WEIGHTS,PANEL_WEIGHTS ,STIFF_WEIGHTS,Es,C1LS,C2LS, C3LS,EILS,C45S,C46S,C1PS,C2PS, C3PSSact,SL_weights,ST_weights,S_T,B_plies |

- 1) [span_space_stiff](#): Establece las longitudes y espaciamientos de los refuerzos en ambas direcciones en el panel del fondo, pero usando la notación de Gekko para las variables y ecuaciones que iteran dentro del sover APOPT.
- 2) [stiff_force](#): Calcula la fuerza cortante y momento flector sobre los refuerzos del fondo en ambas direcciones a partir de la presión de diseño ejercida sobre ellos.
- 3) [plate_force](#): Calcula las dimensiones locales del panel del fondo, y la fuerza cortante y momento flector sobre la plancha del fondo a partir de la presión de diseño ejercida sobre ella.
- 4) [bottom_optimize](#): Obtiene las dimensiones y pesos de la plancha y elementos estructurales optimizados del fondo cumpliendo las restricciones y fronteras definidas usando notación de la librería Gekko.

- 5) *keel_optimize*: Obtiene el peso y módulo seccional optimizados de la quilla cumpliendo las restricciones y fronteras definidas usando notación de la librería Gekko.
- 6) *span_space_stiff_s*: Repite el proceso de 1) en la plancha del costado, pero usando la notación de la librería Gekko para las variables y ecuaciones que iteran dentro del solver APOPT.
- 7) *stiff_force_s* : Calcula la fuerza cortante y momento flector sobre los refuerzos del costado en ambas direcciones a partir de la presión de diseño ejercida sobre ellos.
- 8) *plate_force_s*: Calcula las dimensiones locales del panel del costado, y la fuerza cortante y momento flector sobre ese panel a partir de la presión de diseño ejercida sobre ella.
- 9) *side_optimize*: Obtiene las dimensiones y pesos de la plancha y elementos estructurales optimizados del costado cumpliendo las restricciones y fronteras definidas usando la notación de la librería Gekko.

Panga de respeto nombres

2.2.4 Definición de variables de diseño y restricciones en la librería Gekko

La librería Gekko es una abstracción de alto nivel de problemas matemáticos de optimización. Los valores en los modelos están definidos por *Constantes*, *Parámetros*, y *Variables*. Estos valores están relacionados con otros por medio de *Intermedios* o *Ecuaciones*. Las funciones objetivo están definidas para maximizar o minimizar ciertos valores. Los objetos son colecciones integradas de valores (constantes, parámetros, y variables) y relaciones (intermedios, ecuaciones, y funciones objetivo). Los objetos se pueden construir a partir de otros objetos con relaciones orientadas a objetos. (Logan Beal, 2020). En la tabla 2.10 se presentan los tipos de objetos y sus estructuras con los que trabaja la librería Gekko.

Tabla 2.10 Valores y relaciones para construcción de objetos

| | | |
|----------|------------|--|
| Valor | Constante | m.Const(valor, [nombre]) |
| | Parámetro | m.Param([valor], [nombre]) |
| | Variable | m.Var([valor],[lb],[ub],[integer], [nombre]) |
| Relación | Intermedio | m.Intermediate([ecuación], [nombre]) |
| | Ecuación | m.Equation(ecuación) |

| | | |
|--|------------------|------------|
| | Función Objetivo | m.Obj(obj) |
|--|------------------|------------|

Tomando como variables el número de capas de cada gramaje, número de refuerzos como tipo variable entera, mientras que, la altura y ancho de refuerzos como variable real; la asignación de cada tipo de variable para trabajar en la librería Gekko se hace de la siguiente manera:

Tabla 2.11 Asignación de variables para el uso en la librería Gekko

| Asignar a: | Tipo | value= | lb= | ub= | integer= | name= |
|----------------------------|-------|---|-----|-----|----------|----------------------------------|
| ni,nsi | m.Var | na[i-1] ns[i-1] i=1,...,6 | 0 | 50 | True | 'ni,nsi' |
| NT,NL NLS | | value=R[i] i=0,1,2 | 1 | 10 | | 'NT,NL,NLS' |
| N_bli,N_bti N_sli,N_sti | | BL_Mat[i-1],BT_Mat[i-1] SL_Mat[i-1],ST_Mat[i-1] i=0,...,3 | 0 | 20 | | 'N_bli, N_bti' 'N_bli, N_bti' |
| hL,bLhT,bT hLs,bLs | | DIM[i], DIS[j] i=0,...,3 j=0,1 | 40 | 600 | False | 'hl,bl,ht,bt' 'hls,bls' |

Las restricciones se definen con relaciones a partir de las funciones que optimizan el fondo, costado y quilla; pueden ser de igualdad o en forma de inecuación, expresadas en forma adimensional y homogénea (en el miembro derecho de cada desigualdad aparezca el número 0). Las restricciones empleadas para la optimización del fondo en del código de programación desarrollado se muestran a continuación, desde la ecuación (2.46) hasta la (2.58).

$$\text{CON1} = \text{m.Equation}(\text{C1P} == 0) \quad (2.46)$$

$$\text{CON2} = \text{m.Equation}((0.4 * \text{SigUT} / \text{Sact}) - 1 \geq 0) \quad (2.47)$$

$$\text{CON3} = \text{m.Equation}((\text{C3P} / \text{C2P}) - 1 \geq 0) \quad (2.48)$$

$$\text{CONi} = \text{m.Equation}(\text{CjL} - 1 \geq 0) \quad , \quad i=4,5,6 \quad , \quad j=1,2,3 \quad (2.49)$$

$$\text{CON7} = \text{m.Equation}(\text{EIL} - 1 \geq 0) \quad (2.50)$$

$$\text{CONi} = \text{m.Equation}(\text{C1T} - 1 \geq 0) \quad , \quad i=8,9,10 \quad , \quad j=1,2,3 \quad (2.51)$$

$$\text{CON11} = \text{m.Equation}(\text{EIT} - 1 \geq 0) \quad (2.52)$$

$$\text{CON12} = \text{m.Equation}((\text{hL}/\text{bL}) - 1 \geq 0) \quad (2.53)$$

$$\text{CON13} = \text{m.Equation}((\text{hT}/\text{bT}) - 1 \geq 0) \quad (2.54)$$

$$\text{CON14} = \text{m.Equation}(1 - (\text{C45}/30) \geq 0) \quad (2.55)$$

$$\text{CON15} = \text{m.Equation}(1 - (\text{C46}/21) \geq 0) \quad (2.56)$$

$$\text{CON16} = \text{m.Equation}(1 - (\text{C47}/30) \geq 0) \quad (2.57)$$

$$\text{CON17} = \text{m.Equation}(1 - (\text{C48}/21) \geq 0) \quad (2.58)$$

Las restricciones empleadas para la optimización del costado en el código de programación se muestran desde la ecuación (2.59) hasta la (2.67).

$$\text{CON1} = \text{m.Equation}(\text{C1PS} == 0) \quad (2.59)$$

$$\text{CON2} = \text{m.Equation}((0.4 * \text{SigUT}/\text{Sacts}) - 1 \geq 0) \quad (2.60)$$

$$\text{CON3} = \text{m.Equation}((\text{C3PS}/\text{C2PS}) - 1 \geq 0) \quad (2.61)$$

$$\text{CON}_i = \text{m.Equation}(\text{C}_j\text{LS} - 1 \geq 0) \quad , \quad i=4,5,6 \quad , \quad j=1,2,3 \quad (2.62)$$

$$\text{CON7} = \text{m.Equation}(\text{EILS} - 1 \geq 0) \quad (2.63)$$

$$\text{CON12} = \text{m.Equation}((\text{hL}_s/\text{bL}_s) - 1 \geq 0) \quad (2.64)$$

$$\text{CON13} = \text{m.Equation}((\text{hT}/\text{bT}) - 1 \geq 0) \quad (2.65)$$

$$\text{CON14} = \text{m.Equation}(1 - (\text{C45S}/30) \geq 0) \quad (2.66)$$

$$\text{CON15} = \text{m.Equation}(1 - (\text{C46S}/21) \geq 0) \quad (2.67)$$

La función objetivo para el área del fondo, el costado y quilla se definen de la siguiente manera, respectivamente, para mayor entendimiento de las misma se sugiere al lector revisar los apéndices de este documento, en los cuales se ha detallado el código fuente del proceso de optimización desarrollado:

m.Obj(BOTTOM_WEIGHTS) (2.68)

m.Obj(SIDE_WEIGHTS) (2.69)

m.Obj(KEEL_WEIGHTS) (2.70)

2.3 Comprobación Numérica de la estructura optimizada.

Se usó 3 módulos del software **ANSYS** para el análisis con elementos finitos; el esquema de la forma de conexión entre ellos se muestra en la figura 2.3.

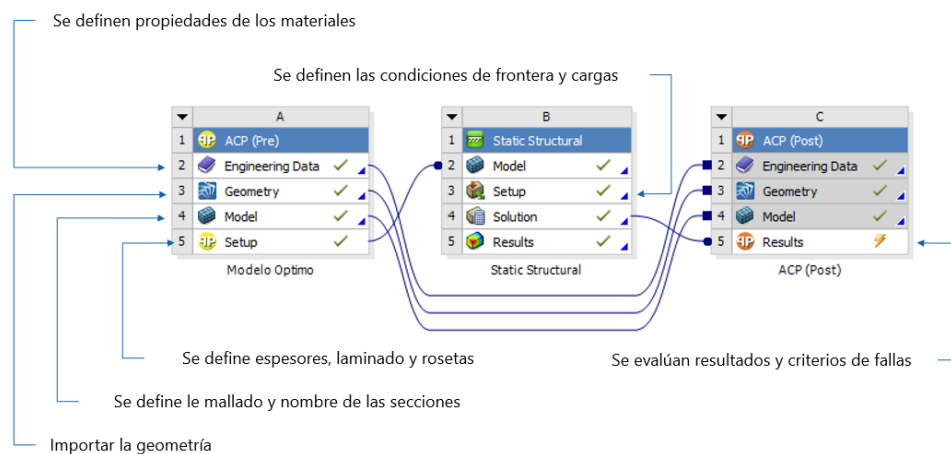


Figura 2.3 Esquema de los módulos usados en el análisis de ANSYS

ACP (Pre) : (Ansys Composite Prep) sirve para analizar estructuras fabricadas con materiales ortotrópicos y compuestos, definiendo las diferentes capas que componen el laminado y las direcciones locales de cada uno de los elementos de la estructura analizada.

Static Structural: Realiza un análisis estructural estático para determinar los desplazamientos, esfuerzos, deformaciones y fuerzas en estructuras o componentes causados por cargas que no inducen un cambio significativo en inercia y efectos de amortiguamiento, es decir se asumen cargas y condiciones estacionarias.

ACP (Post): (Ansys Composite Post) Utiliza los resultados del módulo Static Structural con las características de los laminados definidos en ACP Pre, para obtener la distribución de esfuerzos y deformaciones en cada una de las capas del laminado.

2.3.1 Entrada de datos de los materiales y la Geometría

Los materiales usados en las estructuras actual y optimizada son: madera chanul, fibra de vidrio E-glass CSM y E-glass Roving. Las propiedades de estos materiales fueron ingresadas en el componente **Engineering Data** del módulo **ACP (pre)**. Es común que cuando se realizan análisis estructurales con madera, estos sean asumidos como isotrópicos, ingresando como propiedades mecánicas las correspondientes a la dirección de las fibras de la madera.

Tabla 2.12 Propiedades de la madera Chanul (asumiendo isotropía) (Organization, 2020)

| Propiedad | Valor | Unidad |
|------------------------------------|----------|-------------------|
| Densidad | 870 | kg/m ³ |
| Elasticidad Isotrópica | | |
| Módulo de Young | 1.72E+10 | Pa |
| Razón de Poisson | 0.229 | |
| Módulo de Compresibilidad | 1.06E+10 | Pa |
| Módulo de Corte | 7.01E+09 | Pa |
| Límites Isotrópicos de Esfuerzos | | |
| Esfuerzo de cedencia en tensión | 1.55E+08 | Pa |
| Esfuerzo de cedencia en compresión | 8.06E+07 | Pa |

Para definir las propiedades de las fibras de vidrio se utilizó las tablas del anexo C de la norma ISO 12215 parte 5, en dichas tablas se muestran las propiedades mecánicas dependiendo del contenido estándar de fibra: 0.3 para CSM y 0.48 para WR, como porcentajes del compuesto.

Tabla 2.13 Propiedades de las telas CSM y Roving en "Engineering Data"

| Propiedad | Valor CSM | Valor WR | Unidad |
|-------------------------------|-----------|----------|-------------------|
| Densidad | 2.56 | 2.56 | g/cm ³ |
| Elasticidad Ortotrópica | | | |
| Módulo de Young (dirección X) | 6400 | 13240 | MPa |
| Módulo de Young (dirección Y) | | | |
| Módulo de Young (dirección Z) | | | |
| Razón de Poisson XY | 0.4 | 0.12 | |
| Razón de Poisson YZ | | | |
| Razón de Poisson XZ | | | |
| Módulo de Corte XY | 2750 | 3056 | MPa |

| | | | |
|-----------------------------------|------|--------|-----|
| Módulo de Corte YZ | | | |
| Módulo de Corte XZ | | | |
| Límites Ortrotópicos de Esfuerzos | | | |
| Tensión (Dirección X) | 85 | 183 | MPa |
| Tensión (Dirección y) | | | |
| Tensión (Dirección Z) | | | |
| Compresión (Dirección X) | -117 | -144 | MPa |
| Compresión (Dirección y) | | | |
| Compresión (Dirección Z) | | | |
| Corte XY | 62 | 76 | MPa |
| Corte YZ | | | |
| Corte XZ | | | |
| Tipo de fibra | | | |
| Tipo | MAT | Roving | |

Desde el componente “**Geometry**” se importaron los modelos 3D realizados con el software Rhino de diseño asistido por computadora (CAD). Se realizó el modelo estructural de la sección actual y de la optimizada con las siguientes dimensiones principales en común:

Tabla 2.14 Dimensiones de la sección analizada

| | |
|------------------------------|--------|
| Longitud de Sección [m] | 4 |
| Ancho del fondo a popa [m] | 1.336 |
| Ancho del fondo a proa [m] | 1.15 |
| Ancho de china a popa [m] | 0.1891 |
| Ancho de china a proa [m] | 0.1971 |
| Ancho del costado a popa [m] | 0.914 |
| Ancho del costado a proa [m] | 1.165 |

2.3.2 Selecciones nombradas y mallado

Al momento de realizar el mallado en el componente **Model** fue necesario agrupar ciertos elementos estructurales en función de su laminado, ya que en el siguiente componente **Setup** utiliza esta información para definir el laminado de cada sección.

Tabla 2.15 Nombre de selecciones para la embarcación modelo

| Nombre | Detalle | Cantidad | Tipo | |
|----------------|---|----------|-------------|-------------|
| fondo_bb | Plancha del fondo a babor | 2 | Superficies | |
| fondo_st | Plancha del fondo a estribor | | | |
| costa_bb | Panel del costado a babor | 1 | | |
| costa_st | Panel del costado a estribor | | | |
| quilla | Forro que recubre la quilla | 9 | | |
| trans_fondo_bb | Laminado que recubre los refuerzos transversales del fondo a babor | 108 | | |
| trans_fondo_st | Laminado que recubre los refuerzos transversales del fondo a estribor | | | |
| trans_costa_bb | Laminado que recubre los refuerzos transversales del costado a babor | 29 | | |
| trans_costa_st | Laminado que recubre los refuerzos transversales del costado a estribor | | | |
| palmejar_st | Laminado que recubre el palmejar a estribor | 9 | | |
| palmejar_bb | Laminado que recubre el palmejar a babor | | | |
| long_st | Laminado que recubre el refuerzo longitudinal del fondo a estribor | 9 | | |
| long_bb | Laminado que recubre el refuerzo longitudinal del fondo a babor | | | |
| cores | Núcleos de los refuerzos | 43 | | Sólidos |
| canal_st | Laminado que recubre la rudon para la conexión entre cubierta y costado en estribor | 3 | | Superficies |
| canal_bb | Laminado que recubre el rudon para la conexión entre cubierta y costado de babor | | | |

El tamaño de elementos empleados en la discretización se define especificando su dimensión máxima, usando la opción “sizings”. La malla desarrollada es uniforme en todas las secciones, con 508173 nodos y 309321 elementos para la sección de la embarcación modelo y de 414732 nodos y 272002 elementos para la optimizada. La disminución en la cantidad de nodos y elementos de la sección optimizada en comparación a la modelo se debe a que en la sección optimizada los refuerzos estructurales son más pequeños y por ende se necesitaron menor cantidad de elementos para definirlos.

Tabla 2.16 Tamaño de cada sizing

| Ubicación del mallado (sizing) | Nombre | Tamaño [mm] |
|--------------------------------|-------------|-------------|
| Núcleo de refuerzos | core_stiff | 20 |
| Núcleo de la quilla | core_quilla | 20 |
| Bordes empotrados | bordes | 10 |
| Laminados refuerzos | laminados | 10 |
| Planchas | Mesh | 30 |

Cabe recalcar que en ambos modelos tuvieron el mismo tamaño de elementos con el fin de que no varíen los resultados debido a diferencia entre discretización de elementos finitos. Por ejemplo, para discretizar el laminado del alma de la quilla y refuerzos transversales en la sección actual se empleó un promedio de 13 y 6 elementos, respectivamente, mientras que para la sección optimizada 25 y 5 elementos. Ver figura 2.3. y

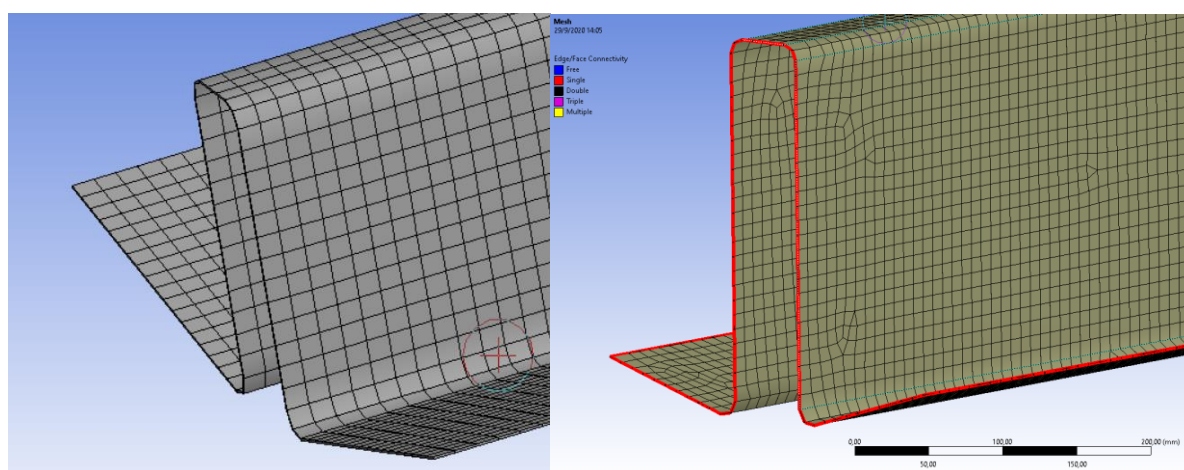


Figura 2.4 Discretización de la quilla, sección actual (izquierda) y sección optimizada (derecha)

2.3.3 Definición del componente Setup en el módulo ACP (Pre)

El componente **Setup** está compuesto de los siguientes campos:

Material Data : Este campo contiene las opciones *Materials*, *Fabrics*, *Stackups* y *Sub Laminate*. En *Materials* se muestran las propiedades mecánicas de cada material definidas en el componente **Engineering data**. En la opción *Fabrics* se definieron las distintas telas de fibra de vidrio

con sus respectivos espesores (CSM300, CSM350, CSM450, CSM600, WR600 y WR800) ver tabla 2.17. En la opción *Stackups* se definió la secuencia de laminado para las planchas del fondo, costado y refuerzos, ver figura 2.3. La opción *Sub Laminate* no fue necesaria para este análisis, dado que en la opción *Stackups* ya se creó el lamiando completo para los paneles de fondo y costado, así como el laminado de los refuerzos estructurales.

Element Sets : Este campo muestra todos los elementos nombrados en el componente **Model**; estos serán los elementos con los cuales se va a trabajar y aplicar los diferentes tipos de laminados.

Rosettes : En este campo se definen las rosetas o direcciones que sigue el laminado; contiene la roseta de origen y además se definieron rosetas locales en cada elemento estructural.

Oriented Selection Sets : Este campo contiene el punto de referencia, dirección y roseta usadas para cada elemento definido en *Element sets*.

Modeling Groups : El este campo se definieron los laminados de la plancha de fondo, plancha de costado y refuerzos según las direcciones y punto de referencia definidos en *oriented selection set*.

Tabla 2.17 Espesor de cada tipo de tela

| Tela (Fabric) | Espesor [mm] |
|---------------|--------------|
| CSM300 | 0.7005 |
| WR350 | 0.8173 |
| CSM450 | 1.0508 |
| CSM600 | 1.4010 |
| WR600 | 0.7760 |
| WR800 | 1.0347 |

2.3.4 Análisis estático estructural

En el módulo *Stactic Structural* se implementaron los soportes del sistema (condiciones de frontera), y presiones actuantes; además se definen los parámetros a presentarse como parte de la solución del proceso numérico: deformación en dirección vertical “z” y esfuerzo equivalente en los núcleos de los refuerzos. En este caso se modelaron las conexiones con los mamparos o extremos de la sección analizada como

empotrados. La presión actuante en el fondo tiene un valor de 47240 Pa (6.85 psi) y 52260 Pa (7.57 psi) para la sección de la embarcación modelo y optimizada respectivamente; estas fuerzas por unidad de área actúan en dirección normal a la plancha. La presión en los costados se modeló con un vector normal a la superficie de 12170 Pa (1.765 psi) y 7080 Pa(1.02 psi) para la embarcación modelo y optimizada respectivamente.

2.3.5 ACP (Post)

El módulo **ACP (post)** muestra los mismos campos que el módulo **ACP (pre)** con la diferencia que en este último módulo se muestran los resultados en el campo *Solutions*. En este proyecto se mostró los resultados como resultados: esfuerzos, deformaciones y los criterios de falla disponibles.

Esfuerzos : Se calculó el esfuerzo normal en la dirección más corta para las planchas del fondo en la capa inferior y superior, y, el esfuerzo normal en el ala superior de los refuerzos y quilla.

Deformaciones : Se calculó la deformación máxima en dirección vertical de la sección de la embarcación modelo y la embarcación optimizada.

Criterios de falla : Se aplicó el criterio del esfuerzo máximo (Barbero, 2011), lo que hace que el programa evalúe la relación entre el esfuerzo permisible del material y el actuante para cada elemento; como resultado presenta el factor de seguridad, la capa crítica y la forma de falla.

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

3.1 Resultados de análisis en la embarcación modelo

Como resultados del objetivo 1 de este proyecto se obtuvo las presiones de diseño en las planchas y refuerzos de fondo y costado de la sección analizada de la lancha modelo aplicando las normas ISO 12215 parte 5. Como es de esperarse las presiones son mucho mayores en los elementos estructurales del fondo, entre 40 y 60 kPa, mientras que en el costado la presión máxima está alrededor de 10 kPa.

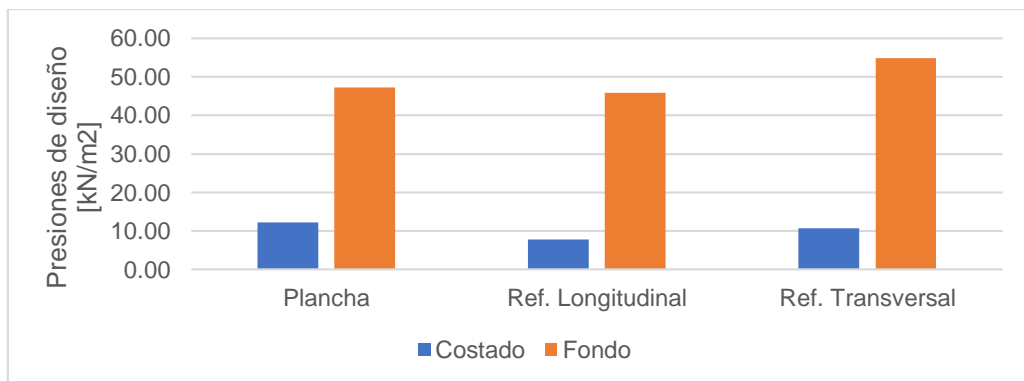


Figura 3.1 Presiones de diseño en plancha y refuerzos de la sección actual

Con las presiones de diseño se obtuvo la fuerza cortante y el momento flector a la que está sometido cada elemento estructural de la sección en su respectiva ubicación (fondo y costado). El máximo momento flector se presentó en los refuerzos longitudinales del costado con un valor de 1454.67 [N m] y la máxima fuerza cortante en los refuerzos longitudinales del fondo con 8955.12 [N]. Esto indica que los refuerzos longitudinales están sometidos a mayor corte y flexión que los transversales.

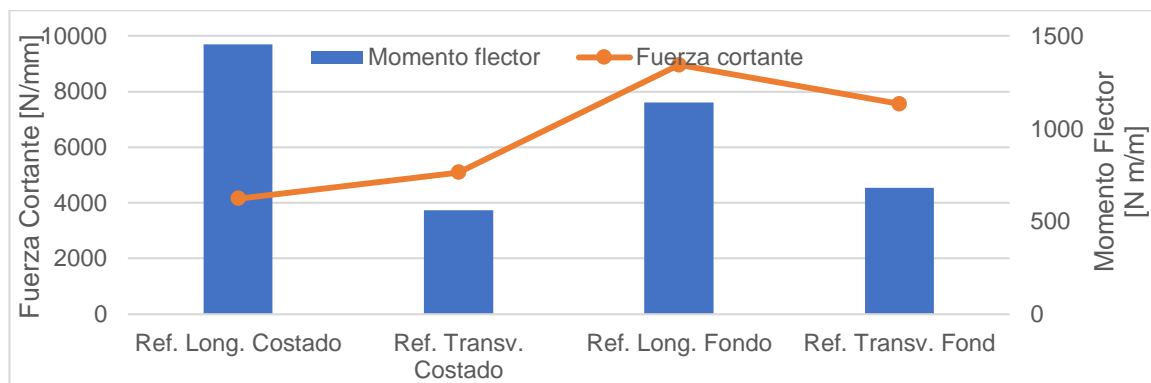


Figura 3.2 Momento Flector y Fuerza Cortante en refuerzos

En el caso de las planchas; la mayor fuerza cortante y momento flector se obtuvo para la plancha del fondo ya que está sometida a mayor presión que la plancha del costado. El momento flector y fuerza cortante de la plancha del costado son 27% de la del fondo.

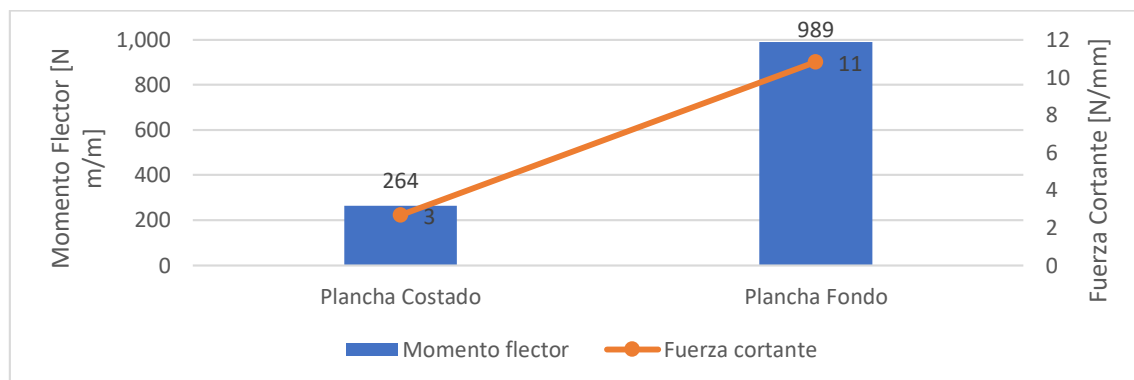


Figura 3.3 Momento Flector y Fuerza Cortante en planchas

Comparando los esfuerzos permisibles en cada plancha con los que actúan se obtuvieron los factores de cumplimiento. Los resultados muestran que existe sobredimensionamiento (factor de cumplimiento mayor que 1.0) de la plancha del costado en corte, los demás factores de cumplimiento en flexión, corte y mínimo peso seco de remolque indican un diseño respecto de los límites permisibles.

Tabla 3.1 Análisis de falla en planchas

| Esfuerzos en planchaje [N/mm ²] | Criterio de falla | Permisible | Actuando | CF | # Capa |
|---|-------------------|------------|----------|------|--------|
| Fondo | Flexión | 34.00 | 4.20 | 8.10 | 1 |
| | Corte | 5.64 | 1.28 | 4.40 | 5 |

| | | | | | |
|---|----------------|------------------|-------------------|-----------|-----------------|
| Costado | Flexión | 34.00 | 7.71 | 4.41 | 1 |
| | Corte | 5.64 | 0.35 | 15.97 | 5 |
| Mínima masa seca de planchaje [kg/m²] | | Requerido | Disponible | CF | Posición |
| Fondo | Min. Peso seco | 2.26 | 6.80 | 3.01 | - |
| Cubierta | Min. Peso seco | 1.94 | 6.80 | 3.51 | - |
| Plancha asociada a la quilla | Min. Peso seco | 4.98 | 9.00 | 1.37 | - |

En los refuerzos de fondo y costado se encontró valores de factor de cumplimiento mayores que 1.0 es decir sobredimensionamiento estructural al analizarlos por rigidez, véase la tabla 3.2. Se aprecia que el valor de CF para el refuerzo del costado en dirección longitudinal es menor que 1, esto se debe a que dicho refuerzo tiene la relación entre el ancho del ala superior y espesor de su laminado mayor que el permisible. El valor permisible (23) es para laminado con materiales similares, es decir con diferencia entre propiedades mecánicas menor al 25% (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008), dado a que se desconoce el laminado exacto del refuerzo no podemos asegurar que este falló o se encuentra en malas condiciones en la sección actual.

Tabla 3.2 Análisis de falla en refuerzos estructurales

| Esfuerzos en refuerzos [N/mm²] | Criterio de falla | Permisible | Actuando | CF | Posición |
|--|--------------------------|-------------------|-----------------|-----------|-----------------|
| Longitudinal del Fondo | Flexión | 34.00 | 4.20 | 8.10 | Ala |
| | Corte | 6.68 | 2.68 | 1.48 | Núcleo |
| Transversal del Fondo | Flexión | 34.00 | 5.25 | 6.47 | Ala |
| | Corte | 6.68 | 1.99 | 1.04 | Núcleo |
| Longitudinal del Costado (Palmejar) | Flexión | 34.00 | 4.99 | 6.81 | Ala |
| | Corte | 6.68 | 0.64 | 2.53 | Núcleo |
| Transversal del Costado | Flexión | 34.00 | 4.54 | 10.30 | Ala |
| | Corte | 6.68 | 2.27 | 1.61 | Núcleo |
| Pandeo Local en Ala superior (relación t/b) | Criterio de falla | Máximo | Actual | CF | Refuerzo |
| Refuerzos de Fondo | Pandeo | 23.00 | 9.93 | 2.32 | Longitudinal |
| | Ala | 23.00 | 9.93 | 2.32 | Transversal |
| Refuerzos del Costado | Pandeo | 23.00 | 25.78 | 0.89 | Longitudinal |
| | Ala | 23.00 | 9.93 | 2.32 | Transversal |

| Pandeo Local en Almas (relación h / (tw/2)) | Criterio de falla | Máximo | Actual | CF | Refuerzo |
|--|------------------------------|------------------|-------------------|-----------|-----------------|
| Refuerzos de Fondo | Pandeo | 31.00 | 25.36 | 1.22 | Longitudinal |
| | Alma (corte) | 31.00 | 15.85 | 1.96 | Transversal |
| Refuerzos del Costado | Pandeo | 31.00 | 15.85 | 1.96 | Longitudinal |
| | Alma (corte) | 31.00 | 15.85 | 1.96 | Transversal |
| Rigidez mínima [N mm²] | Criterio de falla | Requerido | Disponible | CF | Refuerzo |
| Refuerzos de Fondo | Rigidez | 5.45E+08 | 9.78E+10 | 179 | Longitudinal |
| | Rigidez | 2.29E+08 | 3.29E+10 | 144 | Transversal |
| Refuerzos del Costado | Rigidez | 1.91E+09 | 6.22E+10 | 32.6 | Longitudinal |
| | Rigidez | 2.31E+08 | 3.01E+10 | 130 | Transversal |
| Mínimo modulo seccional Quilla | Criterio de falla | Requerido | Disponible | CF | Posición |
| Quilla | Flexión | 159.28 | 402.20 | 2.53 | Quilla |

El peso total por área de las planchas del diseño de la estructura actual con un laminado asumido de 12 capas es de 18.67 kg/m², el espesor total es de 12.55 mm se presenta en la tabla 3.3.

Tabla 3.3 Peso por área del laminado de las planchas de la lancha A

| Laminado del Fondo / Costado | Núm. Capas | Espesor [mm] | Peso Fibra [kg/m ²] | Peso Resina [kg/m ²] | Peso/Área [kg/m ²] |
|---------------------------------|---------------|-----------------|------------------------------------|-------------------------------------|-----------------------------------|
| CSM 450 | 8.00 | 8.41 | 3.60 | 8.40 | 12.00 |
| WR 800 | 4.00 | 4.14 | 3.20 | 3.47 | 6.67 |
| Total | 12.00 | 12.55 | 6.80 | 11.87 | 18.67 |

Con un laminado asumido de tres capas CSM 450 para los refuerzos y quilla en referencia a construcciones locales se muestran las dimensiones (top-hat) y peso por área para longitudinales y transversales del fondo y costado en la figura 3.4.

Tabla 3.4 Peso por área y dimensiones de refuerzos y quilla

| Refuerzos Fondo | Núm. Ref. | Longitud no soportada [m] | Dimensiones | Peso Fibra [kg/m ²] | Peso Resina [kg/m ²] | Peso Área [kg/m ²] |
|-----------------|--------------|------------------------------|-------------|------------------------------------|-------------------------------------|-----------------------------------|
| Longitudinales | 1.00 | 4.00 | 0.08x0.05 | 1.35 | 3.15 | 4.50 |
| Transversales | 6.00 | 1.50 | 0.05x0.05 | 1.35 | 3.15 | 4.50 |

| | | | | | | |
|-------------------|-----------|---------------------------|-------------|---------------------------------|----------------------------------|--------------------------------|
| Quilla | 1.00 | 4.00 | 0.12x0.05 | 1.35 | 3.15 | 4.50 |
| Refuerzos Costado | Núm. Ref. | Longitud no soportada [m] | Dimensiones | Peso Fibra [kg/m ²] | Peso Resina [kg/m ²] | Peso Área [kg/m ²] |
| Longitudinales | 1.00 | 4.00 | 0.15x0.05 | 1.35 | 3.15 | 4.50 |
| Transversales | 3.00 | 1.00 | 0.05x0.05 | 1.35 | 3.15 | 4.50 |

Los pesos de los laminados de los refuerzos, núcleos, quilla y planchas se encuentran en la tabla 3.5 con un peso total de la sección actual de 642 kg.

Tabla 3.5 Peso de elementos estructurales y peso total de la sección actual

| | |
|---|--------|
| Peso total de laminado de planchas [kg] | 373.33 |
| Peso total del laminado de refuerzos [kg] | 107.01 |
| Peso total del núcleo de refuerzos [kg] | 132.24 |
| Peso total de la quilla [kg] | 29.52 |
| Peso total (Plancha + Refuerzos) [kg] | 642.10 |

3.2 Solución del proceso de optimización

Aplicando el proceso de optimización detallado en el capítulo 2 de este documento, se obtuvo las siguientes presiones de diseño, fuerzas cortantes y momentos flectores para los elementos estructurales de la sección optimizada. La máxima presión de diseño actúa en la plancha del fondo con un valor de 52 kPa; la plancha del costado está sometida a un 8% de la presión en el fondo, un 24% de la fuerza cortante que actúa en el fondo y un 41% del momento flector que actúa en el fondo. El momento flector que actúa en refuerzos del fondo es similar al de los costados, pero la fuerza cortante en el costado es un 43% de la que actúa en el fondo.

Tabla 3.6 Presión y cargas de diseño en elementos estructurales optimizados

| | | |
|--|--------|-------------------|
| Presión de diseño actuante en el Fondo | | |
| Refuerzo transversal (Cuaderna) | 40.75 | kN/m ² |
| Plancha | 52.26 | kN/m ² |
| Presión de diseño actuante en el Costado | | |
| Refuerzo transversal (Cuaderna) | 6.05 | kN/m ² |
| Plancha | 7.09 | kN/m ² |
| Fuerza cortante y Momento Flector actuante en la plancha fondo | | |
| Fuerza cortante | 9.41 | N / mm |
| Momento flector | 564.43 | N mm/mm |
| Fuerza cortante y Momento Flector actuante en la plancha costado | | |

| | | |
|---|----------|---------|
| Fuerza cortante | 2.22 | N / mm |
| Momento flector | 231.42 | N mm/mm |
| Fuerza cortante y Momento Flector actuante en los refuerzos fondo | | |
| Momento flector (Cuaderna) | 3056.36 | N m |
| Fuerza Cortante (Cuaderna) | 12225.92 | N m/m |
| Fuerza cortante y Momento Flector actuando en los refuerzos costado | | |
| Momento flector (Cuaderna) | 3540.43 | Nm |
| Fuerza Cortante (Cuaderna) | 5310.86 | N m/m |

Como resultado de la optimización empleando el método de Ramificación y Acotamiento (*Branch and Bound*) se obtuvo diferentes puntos óptimos que dependen de cada punto inicial, debido a la presencia de variables enteras y al método de optimización empleado. A continuación, en la figura 3.4, se presentan diez soluciones óptimas partiendo de diferentes puntos iniciales para las variables de diseño. La función objetivo óptima (peso estructural del fondo, un solo lado) oscila alrededor de 125 kg. Como cualquier punto inicial no asegura encontrar una solución óptima, en la figura 3.4 solo se presenta las variables de diseño que cambiaron por cada punto inicial que encontró una solución óptima, estas son: n1, n4 y n5 corresponden al número de capas de CSM 300, CSM 600 y WR 600, respectivamente. Se puede observar en la figura 3.5 que para los intentos 1,2,6,7 y 8 se obtuvo la misma solución desde diferentes puntos iniciales, pero esto no asegura que sea la solución más óptima, a su vez para el intento 3 y 5 se obtuvo la mejor solución con un valor de función objetivo de 120,55 kg. Como se mencionó anteriormente el algoritmo no es capaz de encontrar una solución óptima desde cualquier punto inicial, es por eso por lo que en los diez intentos presentados las demás variables se mantuvieron fijas para los puntos iniciales.

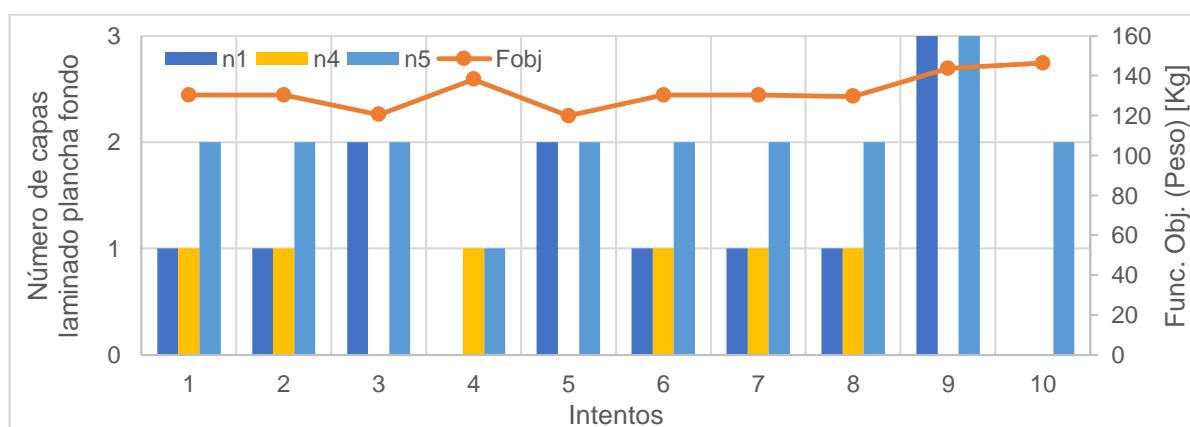


Figura 3.4 Número de capas óptimas función objetivo desde diferentes puntos iniciales para el planchaje del fondo.

Para la sección estructural optimizada el algoritmo concluye que no se requieren refuerzos longitudinales, mientras que el número de refuerzos transversales óptimo es 9. Cabe recalcar que para soluciones con número de refuerzos transversales de 10 y 7 presenta mayor valor de función objetivo según la figura 3.6.

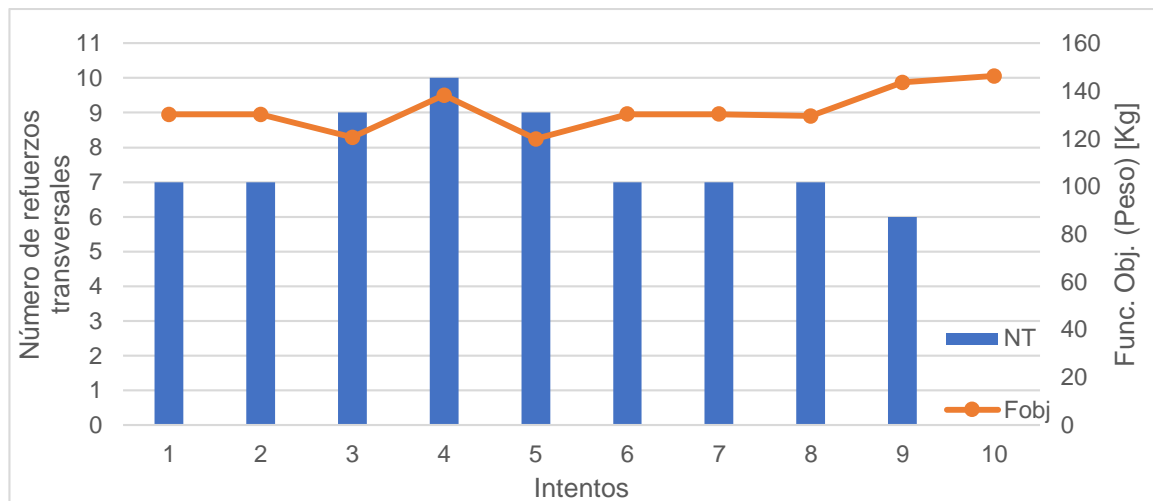


Figura 3.5 Número de refuerzos transversales y función objetivo para diferentes puntos iniciales

En todo proceso de optimización es necesaria la identificación de restricciones activas, es decir aquellas que estén en la frontera de su cumplimiento. En el presente trabajo se expresan las restricciones en forma adimensional en el rango de 0 a 1. Las restricciones activas en todos los puntos iniciales presentados son: la flexión en la plancha y el corte en el alma y núcleo de los refuerzos transversales. Por lo cual en el diseño y construcción de este tipo de embarcaciones se deben mantener presente la importancia de realizar un diseño estructural adecuado que soporte los esfuerzos normales en las planchas y suficiente área transversal en los refuerzos para soportar el esfuerzo cortante.

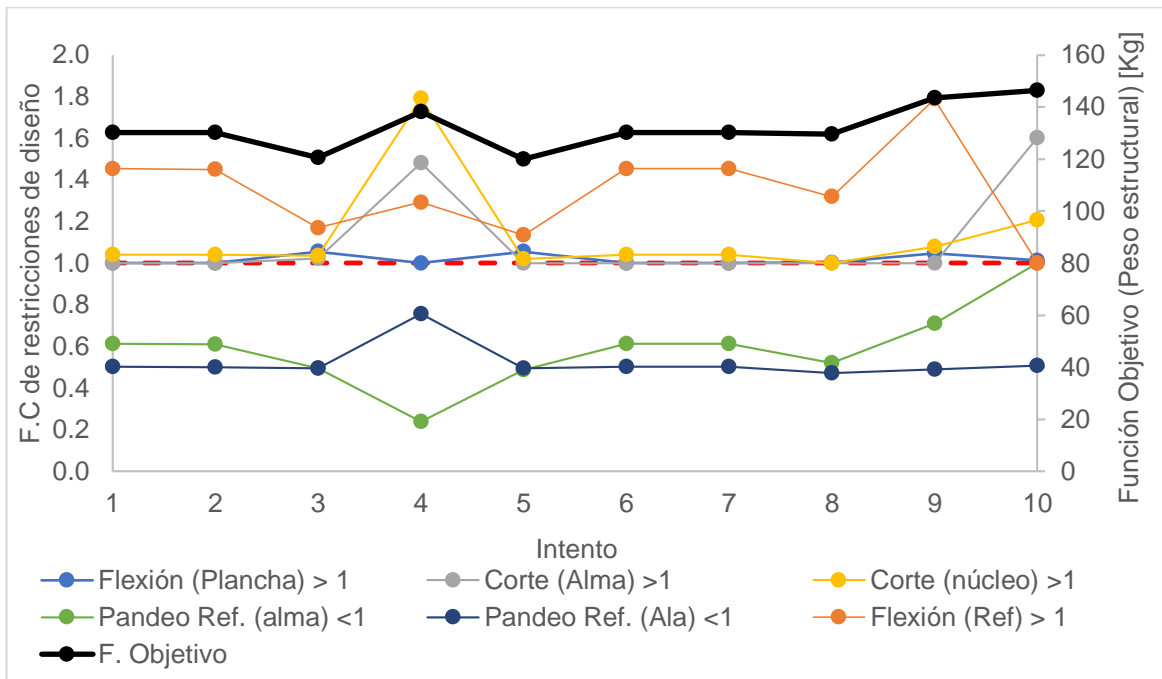


Figura 3.6 Restricciones no dimensionalizadas y función objetivo para diferentes puntos iniciales

Al aplicar el método de acotamiento y ramificación al proceso de optimización se obtuvo gráficas como la figura 3.7 donde la función objetivo presenta picos que llegan hasta los 300 kg a medida que se avanza en el proceso. Esto se debe a que, si la solución en cada rama no es entera, la rama se divide en dos nodos. En caso de que el nodo corresponda a una solución factible o el límite es el valor de una solución óptima, el valor de esta se compara con el titular y se mantiene la mejor solución y su valor. Si el límite no es mejor que el titular, el subproblema se descarta (o se profundiza), ya que ninguna solución factible del subproblema puede ser mejor que el titular.

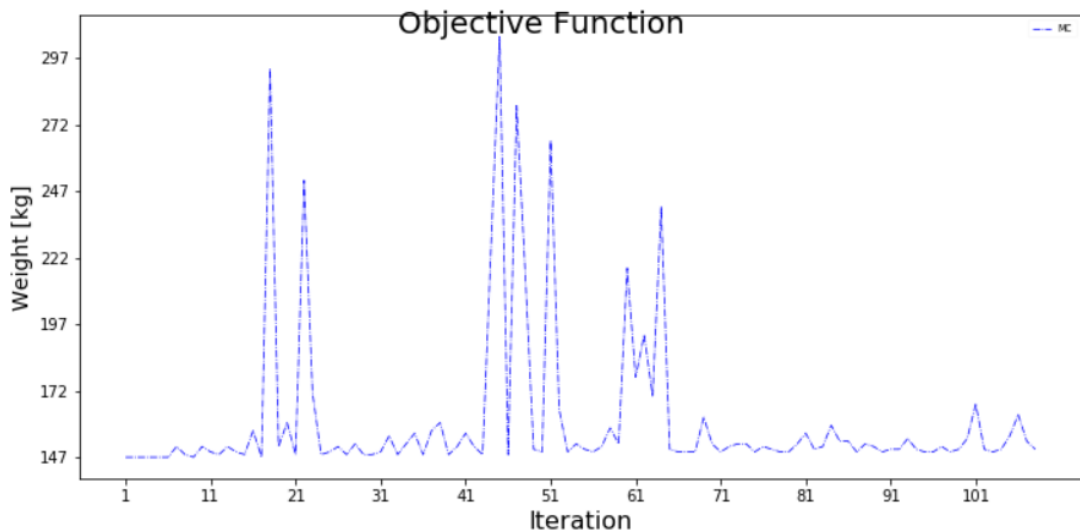


Figura 3.7 Variación de la función objetivo en el fondo por cada iteración

Tabla 3.7 Número total óptimos de capas en laminado de planchaje de fondo y costado

| MAT | | | | WR | | |
|-----------------------|-----|-----|-----|-----|-----|-----|
| Gramaje | 300 | 350 | 450 | 600 | 600 | 800 |
| Capas | 2 | 0 | 0 | 0 | 2 | 0 |
| Número de capas total | 12 | | | | | |

El panel del fondo abarca una mayor área por lo que su peso es el doble que el peso del panel del costado.

Tabla 3.8 Peso de elementos estructurales optimizados del fondo

| | | |
|------------------|--------|------|
| Área | 6.00 | [m2] |
| Peso Planchas | 156.00 | [Kg] |
| Peso Refuerzos | 85.10 | [Kg] |
| Peso Quilla | 86.95 | [Kg] |
| Peso Total fondo | 328.05 | [Kg] |

Tabla 3.9 Peso de elementos estructurales optimizados del costado

| | | |
|--------------------|--------|------|
| Área | 4.00 | [m2] |
| Peso Planchas | 104.00 | [Kg] |
| Peso Refuerzos | 56.73 | [Kg] |
| Peso Total Costado | 160.73 | [Kg] |

Comparando la sección actual con la optimizada, se pudo lograr una reducción de peso en las planchas del -30.3% y de los refuerzos del -40.6% pero se observa un aumento en el peso de la quilla, +190%. En la figura 3.8 se presentan los valores totales de los pesos de la sección actual y optimizada con una reducción total del peso de la sección optimizada en comparación a la sección actual del 23.8%. El aumento en el peso de la quilla se debe a que en el proceso de optimización el requerimiento de la ISO 12215 parte 6 para el mínimo módulo seccional dio como resultado dimensiones no adecuadas para restringir la deflexión de esta; por esa razón el análisis con elementos mostró valores de deflexión altos. Debido a esto, se optó por tomar la misma rigidez flexural de la combinación de los refuerzos longitudinales de la sección actual, luego se le aplicó el proceso de optimización para tratar de reducir el peso y obtener menores deflexiones en el análisis de elementos finitos.

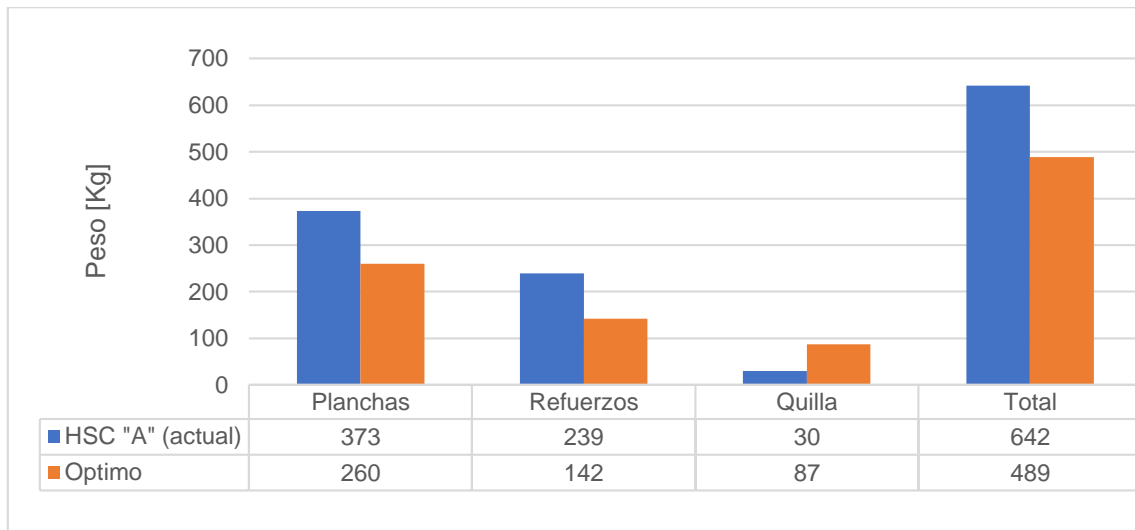


Figura 3.8 Comparación de pesos de elementos estructurales y peso total entre la sección actual y optimizada

3.3 Comparación de resultados con el MEF actual y optimizado.

En la figura 3.9 se muestran los modelos estructurales de la sección actual (izquierda) y sección optimizada (derecha).

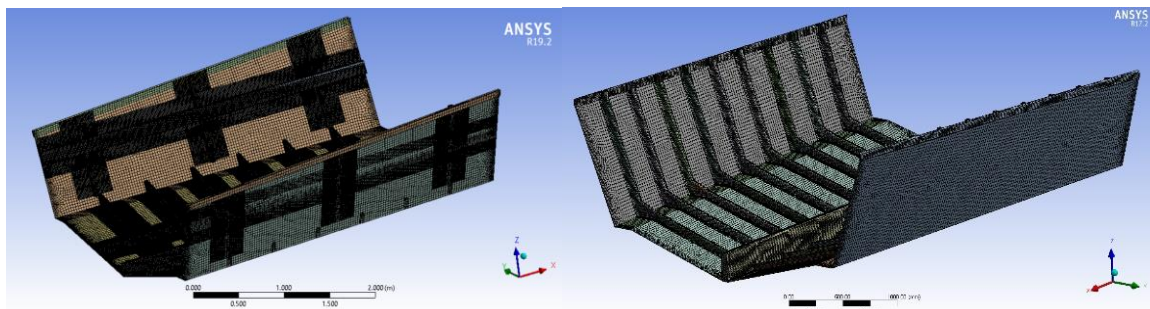


Figura 3.9 Comparación de modelos estructurales para análisis FEM

Como resultados del objetivo 4 de este trabajo, se analizó la sección actual y optimizada con el método de elementos finitos, se utilizó la herramienta **ANSYS** con sus módulos *ACP* y *Static Structural*.

3.3.1 Comparación de esfuerzos normales

Como resultado del análisis se obtuvo el nivel de esfuerzos actuando en las secciones actual y optimizada debido a la presión de diseño. En la figura 3.10 se muestra la distribución de esfuerzos normal en dirección longitudinal "x" para la capa externa e interna del laminado del fondo para la sección actual y optimizada. En ambas secciones

se aprecia que el esfuerzo actuante está por debajo de los esfuerzos permisibles para tensión y compresión en el panel del fondo, 34 y -47 N/mm² respectivamente; a excepción de los bordes considerados como empotrados. En los bordes empotrados de la sección actual el esfuerzo actuante en tensión es alrededor de 40 N/mm² y compresión es -18.5 N/mm² para la capa externa, mientras que para la capa interna el esfuerzo en tensión es 37 N/mm² y -52 N/mm² en compresión. En los bordes empotrados de la sección optimizada el esfuerzo actuante en tensión es alrededor de 44 N/mm² y compresión es -19.68 N/mm² para la capa externa, mientras que para la capa interna el esfuerzo en tensión es 49 N/mm² y -47 N/mm² en compresión. Esto se debe a que al considerar los bordes empotrados, aumenta el momento flector y con ello los esfuerzos actuantes.

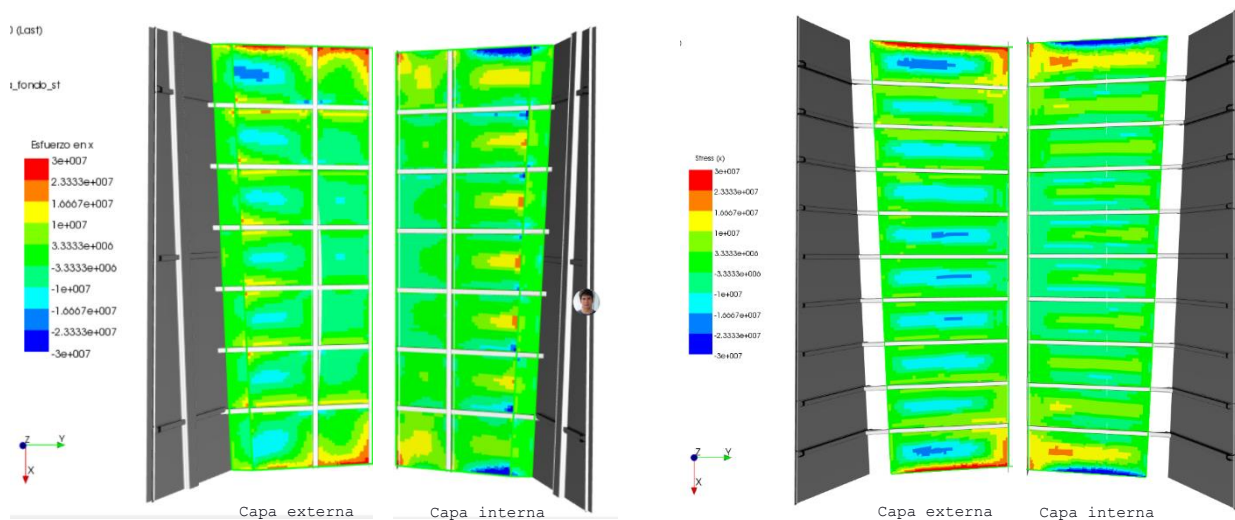


Figura 3.10 Distribución de esfuerzo en dirección vertical para la sección actual (izquierda), sección óptima (derecha).

Para resumir la información presentada en la figura 3.9 se muestran las figuras 3.10 y 3.11 donde se presenta la distribución del esfuerzo normal en dirección “x” (dirección más corta del panel) para la plancha del fondo en la capa externa (esfuerzo en compresión) y capa interna (esfuerzo en tensión) del laminado. Por facilidad de observación se presentó el valor absoluto del esfuerzo en compresión actuando en la capa externa junto con el esfuerzo en tensión actuante en la capa interior del forro.

Los resultados de la sección actual (ver figura 3.11) se mostraron que el esfuerzo máximo actuante en tensión es de 8.77 [MPa] el cual no supera al esfuerzo permisible

34 [MPa], de la misma manera el esfuerzo máximo actuando en compresión es de -10.2 [MPa] el cual no supera al permisible -47 [MPa].

Los resultados de la sección optimizada (ver figura 3.12) mostraron que el esfuerzo máximo actuante en tensión es de 16.55 [MPa], de la misma manera el esfuerzo máximo actuante en compresión es de -17.48 los cuales no superan el esfuerzo máximo permisible para los paneles del forro. La tendencia de las gráficas de esfuerzo normal en tensión para ambas secciones refleja los principios de teoría de planchas ya que el esfuerzo normal es mayor en los bordes empotrados y tiene a disminuir en el centro.

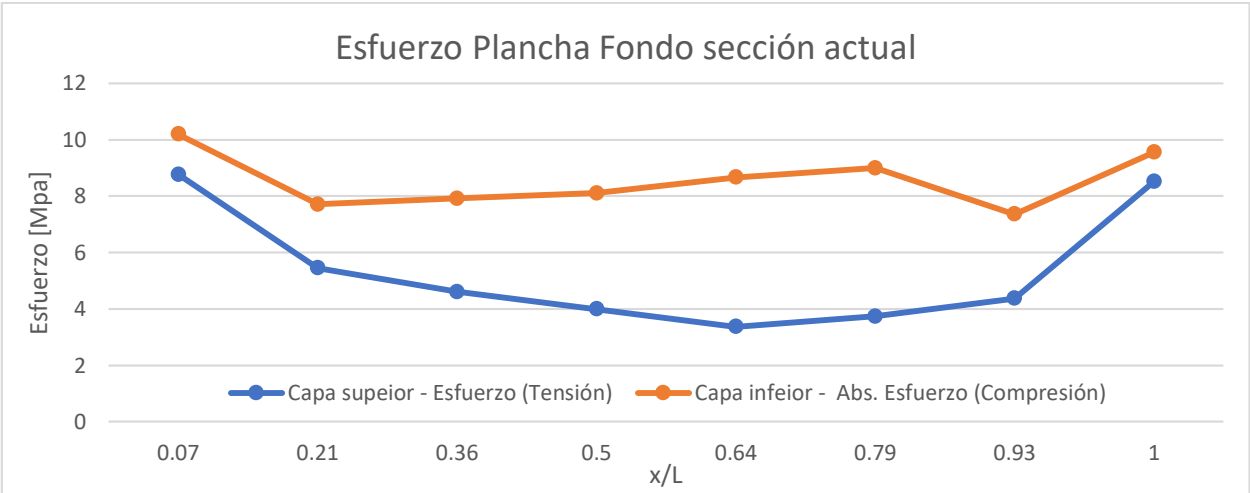


Figura 3.11 Distribución de esfuerzos en dirección x (local) para la plancha del fondo actual

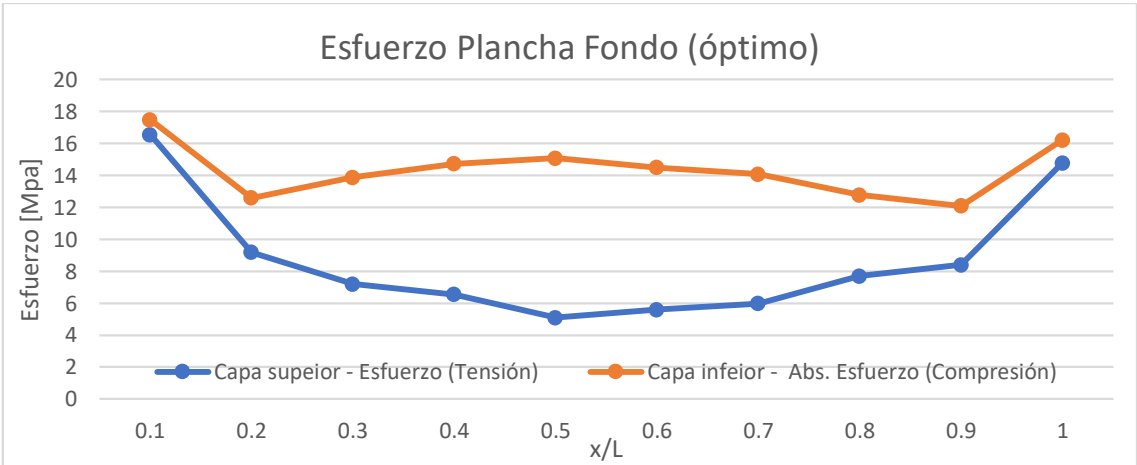


Figura 3.12 Distribución de esfuerzos en dirección x (local) para la plancha del fondo óptima

Otro aspecto importante es analizar el esfuerzo intralaminar entre capas, según la norma ISO este criterio de falla no es crítico cuando el laminado es simple (Standardization, Small Craft Hull Construction and Scantlings 12215-5, 2008). En la figura 3.13 y 3.14 se

muestra el esfuerzo intralaminar actuante en el panel del fondo tanto en la sección actual como en la óptima, con valor alrededor de 5 N/mm² se encuentra ubicado entre la combinación de tela MAT y WR más externa del laminado; el esfuerzo intralaminar permisible es de 6.9 N/mm² para tela MAT, entonces se puede asegurar que el esfuerzo intralaminar actuante no sobrepasa al esfuerzo permisible.

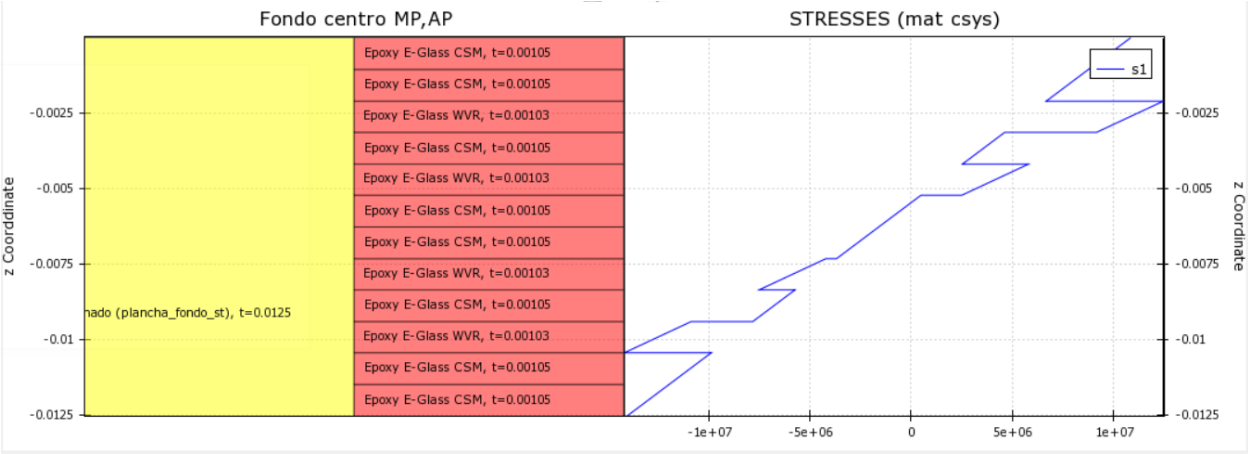


Figura 3.13 Distribución de esfuerzo interlaminar y normal en dirección “x” de un panel central de la sección actual.

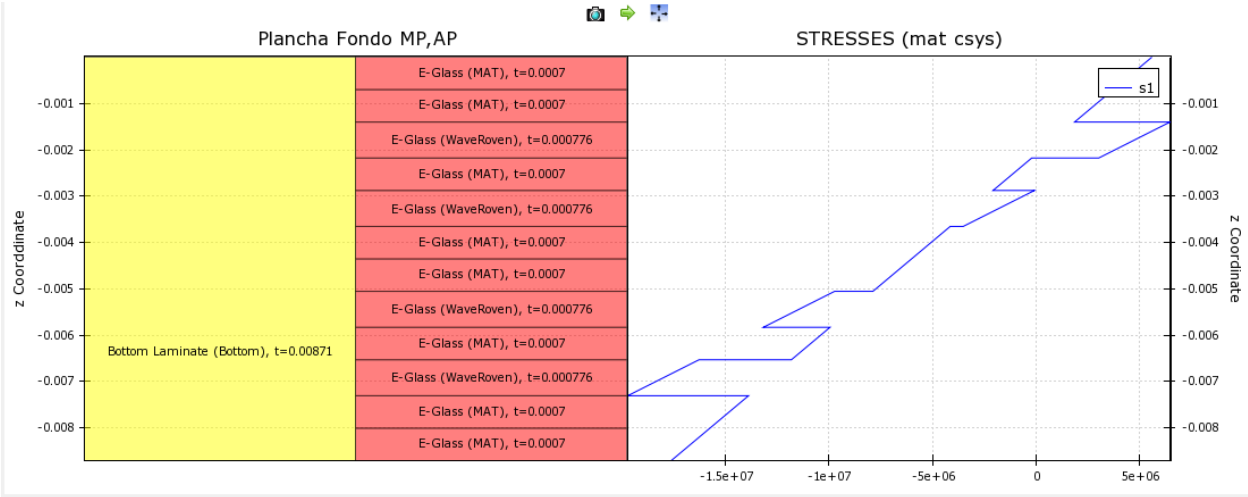


Figura 3.14 Distribución de esfuerzo interlaminar y normal en dirección “x” de un panel central de la sección optimizada.

3.3.2 Comparación de deformación en dirección vertical

La deflexión en dirección vertical en las secciones actual y óptima fueron comparadas mediante los resultados del análisis con elementos finitos. En la figura 3.13 se presenta la comparación de la deflexión vertical de ambas secciones como resultado del módulo ACP (post); se nota una deformación elevada en el modelo óptimo, pero se

encuentra por debajo del límite permisible de deflexión relativa (y/L) propuesto por la norma ISO 12215 parte 5 para refuerzos estructurales, que es igual a 0.05.

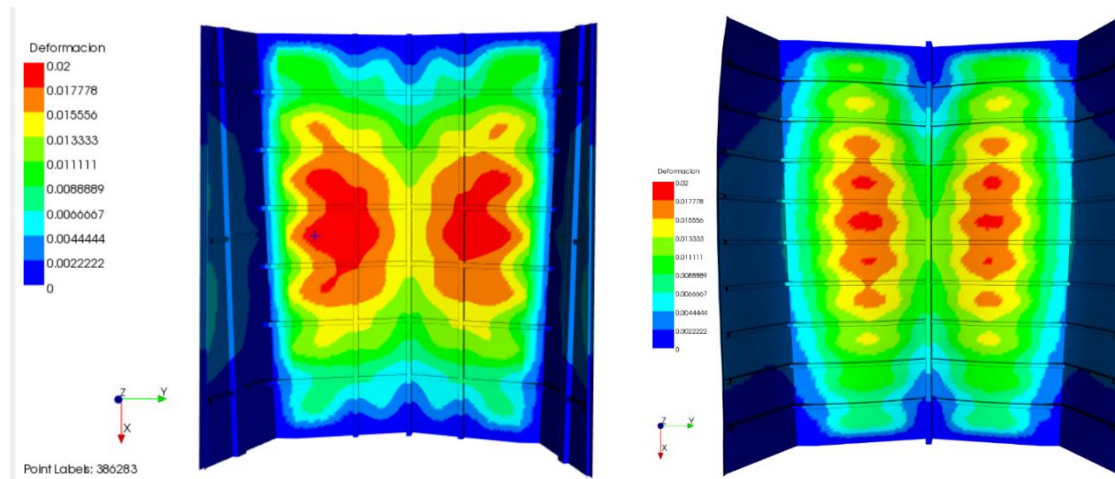


Figura 3.15 Distribución de deflexiones en dirección vertical para la sección actual (izquierda), sección óptima (derecha).

Se registró la deformación vertical para la quilla, plancha y refuerzos transversales, estos dos últimos a $\frac{1}{4}$ del ancho de la plancha del fondo para la sección actual y óptima. La figura 3.15 muestra las deflexiones de la sección actual, como se puede observar la máxima deflexión en dirección vertical es de alrededor 19.8 [mm] y se presenta aproximadamente en el panel ubicado entre los refuerzos transversales de sección media en la mitad del refuerzo longitudinal. La sección optimizada muestra que la máxima deflexión en dirección vertical es de alrededor de 18.2 [mm] y se localiza en el panel ubicado entre los refuerzos transversales de sección media. Este valor de deflexión elevado es debido a la restricción dentro proceso de optimización que controla la deflexión de los refuerzos estructurales, esta permite una deflexión relativa (y/L) de 0.05. La figura 3.16 y 3.17 muestran las deflexiones de cada elemento estructural para la sección actual y optimizada, respectivamente; como se puede observar la quilla tiene 4 metros de longitud y la restricción por deflexión según la norma ISO 12215-5 permite que se defleccione hasta 20 [mm]. Si se considera la diferencia entre el desplazamiento vertical de las cuadernas en el panel actual no se muestra una deflexión considerable. Mientras que para la sección óptima se obtiene una deflexión de 2 a 3 mm.

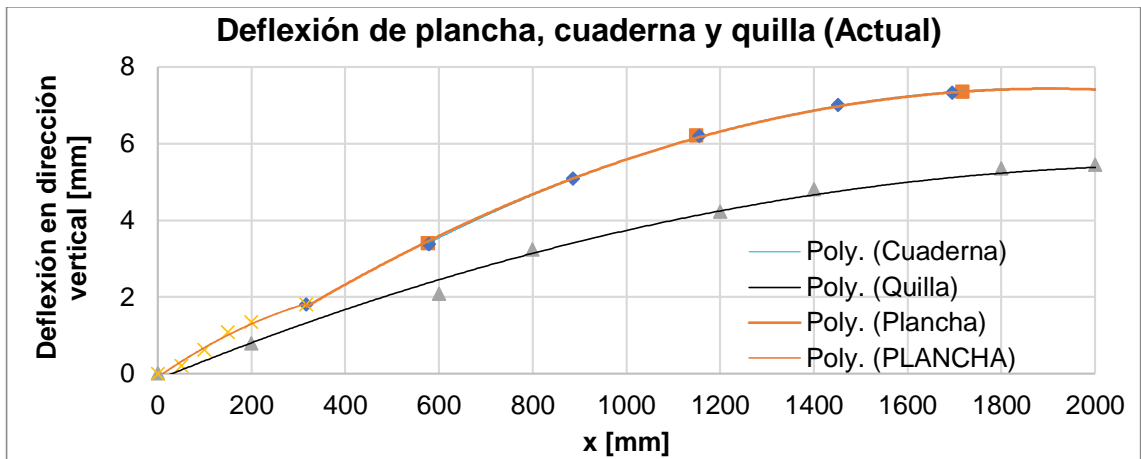


Figura 3.16 Deflexión de la quilla, cuaderna y plancha del fondo de la sección modelo

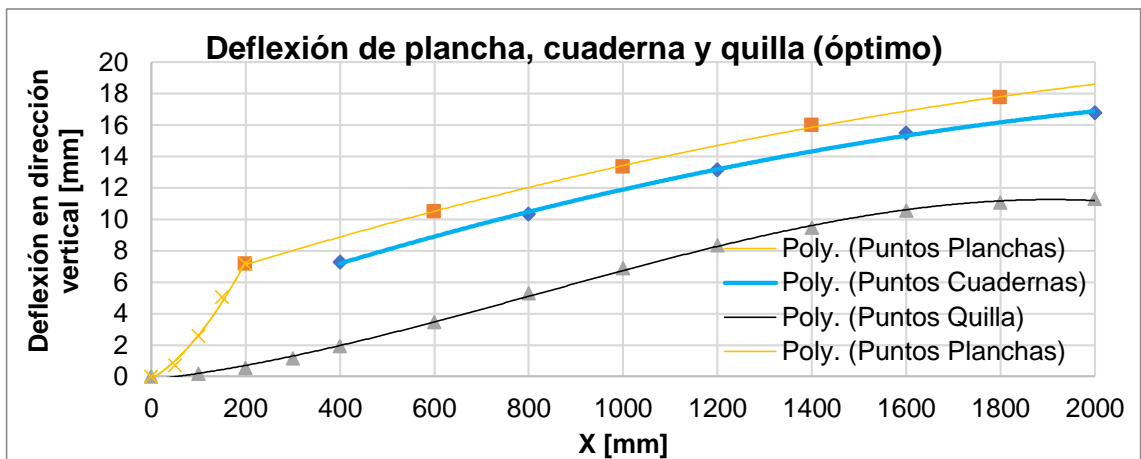


Figura 3.17 Deflexión de la quilla, cuaderna y plancha del fondo de la sección optimizada

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

En este trabajo se optimizó una sección con mismas formas y dimensiones mediante el lenguaje de programación orientado a objetos **Python**, se analizó el nivel de esfuerzos actuando en la estructura de una sección de la lancha que actualmente opera en Galápagos siguiendo los lineamientos de la norma ISO 12215. Se obtuvo una reducción total de peso estructural de 23.8% y se validaron estos resultados comparando los análisis en MEF para ambas secciones. A partir de la validación de estos resultados, se puede concluir:

4.1 Conclusiones

- El análisis de la estructura original muestra un nivel de factor de cumplimiento mayor a 1.0, sobre todo en el laminado de la plancha del fondo y costado donde se obtiene valores de CF entre 4 y 16. Considerando estos bajos niveles de esfuerzo calculados según la norma ISO 12215, existe sobredimensionamiento estructural en la embarcación modelo para transporte de pasajeros en las islas Galápagos.
- Las expresiones empleadas por la norma ISO 12215 consideran el modelado de las vigas y planchas con extremos empotrados. Comparando los resultados obtenidos aplicando el MEF, se concluye que dichas asunciones disminuyen la rigidez requerida para los elementos estructurales, y como resultado se obtienen deformaciones mayores a las esperadas.
- El proceso de optimización estructural MINLP incluyendo variables enteras y reales es capaz de obtener una estructura con menor peso. Además, los resultados con el análisis empleando el MEF comprueba que las restricciones son respetadas y satisface los requerimientos de la norma ISO 12215. Sin embargo, cabe destacar que las restricciones activas en el proceso son: la flexión en la plancha y el corte en el alma y núcleo de los refuerzos transversales, información que puede ser considerada por los diseñadores y constructores de este tipo de embarcaciones.

4.2 Recomendaciones

- Las embarcaciones interislas incluyen alrededor de tres motores de alta potencia y por lo tanto elevado peso, concentrado en el espejo. Por ello se recomienda desarrollar una optimización para la región de popa de la estructura. Además, se considera necesario comprobar la rigidez de la quilla en futuros trabajos que incluyan el análisis de mamparos, considerando que en el diseño óptimo no se incluyen refuerzos longitudinales. Finalmente se puede analizar la estructura optimizada completa al momento del impacto con las olas al navegar (*slamming*)

BIBLIOGRAFÍA

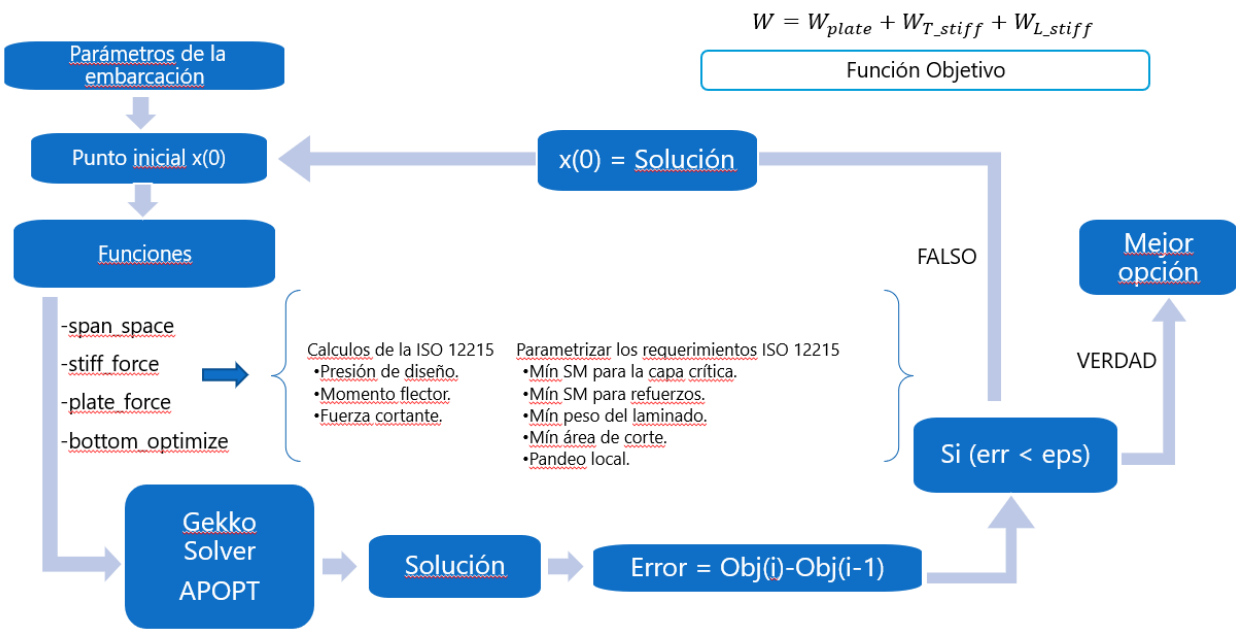
- Barbero, E. J. (2011). *Introduction to composite material design*. CRC Press.
- Belegundu, A. D., & Chandrupatla, T. R. (2011). *Optimization Concepts and Appli.* Cambridge: Cambridge University Press.
- Goubalt, P., & Mayes, S. (1996). Comparative Analysis of Metal and Composite Materials for the Primary Structures of Patrol Craft. *Naval Engineers Journal*, 11.
- Logan Beal, J. H. (2020). *Gekko Dynamic Optimization - Documentation* .
- Maneeapan, K., Sheno, R. A., & Blake, J. I. (2007). Genetic Algorithms (GAs) based optimisation of frp composite plated grillages in ship structure. *International Journal of Maritime Engineering*, 19.
- Marín, J. R., & Costáin, J. M. (1998). *Optimización estructural preliminar de una embarcación menor satisfaciendo las reglas de construccion ABS*. Revista Tecnologica.
- Organization, I. T. (2020). *ITTO Lesser Used Species*. Obtenido de ITTO Lesser Used Species: <http://www.tropicaltimber.info/specie/chanul-humiria-balsamifera/>
- Sobey, A. J., Blake, J., & Sheno, R. (2013). Optimization of composite boat hulls using first principles and design rules. *Ocean Engineering*, 9.
- Song, J.-H., & Oh, D. (2016). *Lightweight Structure Design for Composite Yacht with Optimun Fiber Mass Content*. South Korea.
- Standardization, I. O. (2008). *Small Craft Hull Construction and Scantlings 12215-5*. London.
- Standardization, I. O. (2008). *Small Craft Hull Construction and Scantlings 12215-6*. London.
- Vanderplaats, G. N. (1999). *Numerical Optimization Techniques for Engineering Design*. Colorado Springs: Vanderplaats Research & Development, Inc.

APÉNDICES

APÉNDICE A

ESQUEMA DEL PROCESO DE OPTIMIZACIÓN

A continuación, se presenta el esquema general que sigue el proceso de optimización, El proceso comienza definiendo las características de la embarcación, tales como: eslora tota, manga, calado, velocidad de navegación, entre otras. Luego es necesario dar un punto inicial a las variables de diseño, el proceso cuenta con 39 variables de diseño que son inicializada en diferentes momentos de la ejecución del programa. Luego las funciones, que son las encargadas de parametrizar todas las expresiones presentes en la norma ISO 12215 para el diseño de embarcaciones con PRFV en función a las variables de diseño. Finalmente, se llama al solver APOPT, este solver es el que dispone la librería Gekko para resolver problemas de optimización con variables mixtas. Luego se evalúa la diferencia entre las distintas soluciones optimas encontradas a partir de puntos iniciales hasta obtener la mejor solución.



APÉNDICE B

Se recuerda al lector que en el presente apéndice como en los apéndices posteriores se muestra el código fuente del proceso de optimización. Con el fin de que el lector entienda lo mejor posible el código fuente se muestra los comentarios (letras de color verde) para cada cálculo realizado y los valores de salida (letras color naranja) en cada función.

CÓDIGO PYTHON – OPTIMIZACIÓN DEL FONDO

```
#####IMPORT LIBRARIES#####
%matplotlib inline
#%matplotlib notebook
from math import*
import glob,os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import rcParams
from matplotlib.font_manager import FontProperties
import matplotlib
import matplotlib.gridspec as gridspec
import time
try:
    from gekko import GEKKO
except:
    !pip install gekko
    from gekko import GEKKO
    print('Gekko was installed!')
print('Optimization functions and libraries were loaded!')

#BOTTOM OPTIMIZATION
#BOTTOM FUNCTIONS

#THIS FUNCTION STIMATE THE SPAN AND SPACE OF STIFFENERS
def span_space_stiff(nt,nl):
    """
    INPUT
    nt,nl : numbers of transversal and longitudinals stiffeners
    lp    : [m] dimension of panel in "x" direction Aft/Fore
    b     : [m] dimension of panel in "y" direction Br/St
    OUTPUTS:
    lut   : [mm] Span of transversal stiffeners
    st    : [mm] Space between transversal stiffeners
    lul   : [mm] Span of longitudinals stiffeners
    sl    : [mm] Space of between longitudinals stiffener
    """

    sl = m.Intermediate(1000*b/(nl+1),name='sl')
    st = m.Intermediate(1000*lp/(nt+1),name='st')
    lul = m.Const(1000*lp,name='lul')

    return st, lul, sl
```



```

# THIS FUNCTION STIMATE THE PRESSURE, FORCE AND BENDING MOMENT ACTING
# ON STIFFENERS USING ISO 12215-PART5 EXPRESSIONS
def stiff_force(lul,sl,st):
    '''
    OUTPUTS:
    pbt : [kN/m2] Pressure of bottom transverlas stiffeners
    pbl : [kN/m2] Pressure of bottom longitudinals stiffeners
    fdt : [N] Design shear force for transversal stiffeners
    fdl : [N] Design shear force for longitudinal stiffeners
    mdt : [N m] Design bending moment for transversal stiffeners
    mdl : [N m] Design bending moment for longitudinals stiffeners
    '''

    # PRESSURE REDUCTION FACTORS ISO 12215 part-5
    # Fixed value for Design Category B
    kDC = m.Const(0.8,name='kdcr')

    # Dynamic Load Factor in "g" units
    nCG = (0.32*((lwl/(10*bc))+0.084)*(50-beta)*((v*bc)**2)/mldc))
    if nCG >= 3:
        nCG = 0.5*v/(mldc**0.17)
    if nCG > 7:
        nCG = 7
    nCG = m.Const(nCG,name='ncgr')

    #Area pressure reduction factor
    adt1 = m.Intermediate(0.33*(sl**2)*1e-6,name='adt1')
    adl1 = m.Intermediate(0.33*(lul**2)*1e-6,name='adl1')
    adt2 = m.Intermediate((sl*st)*1e-6,name='adt2')
    adl2 = m.Intermediate((lul*sl)*1e-6,name='adl2')
    ADT = m.Intermediate(m.max2(adt1,adt2),name = 'adt')
    ADL = m.Intermediate(m.max2(adl1,adl2),name = 'adl')

    karl1 = m.Intermediate((0.1*(mldc**0.15)/(ADL**0.3)),name='karl1')
    karl2 = m.Intermediate(m.max2(0.25,karl1),name='karl2')
    kARL = m.Intermediate(m.min2(1,karl2),name='karl')
    kART = m.Intermediate((0.1*(mldc**0.15)/(ADT**0.3)),name='kart')

    #DESIGN PRESSURE ISO 12215 part 5
    #Motor craft bottom pressure in planing mode in [kN/m2]
    pbt = m.Intermediate(((0.1*mldc*(1+(kDC**0.5)*nCG))/(lwl*bc))*kART
        ,name='pbt')
    pbl = m.Intermediate(((0.1*mldc*(1+(kDC**0.5)*nCG))/(lwl*bc))*kARL
        ,name='pbl')

    #STIMATE DESIGN FORCE [N] AND DESIGN BENDING MOMENT [N m]
    fdt = m.Intermediate(5*pbt*st*sl*1E-4,name='fdt') # [N]
    fdl = m.Intermediate(5*pbl*sl*lul*1E-4,name='fdl') # [N]
    mdt = m.Intermediate(83.33*pbt*st*(sl**2)*1E-9,name='mdt') # [N m]
    mdl = m.Intermediate(83.33*pbl*sl*(lul**2)*1E-9,name='mdl') # [N m]
    return pbt,pbl,fdt,fdl,mdt,mdl

```

```

#THIS FUNCTION STIMATE THE FORCE AND BENDING MOMENT ACTING ON THE PLATE
# USING ISO 12215 PART-5 EXPRESSIONS
def plate_force(b1,bt,sl,st):
    '''
    OUTPUTS:
    pbp : [kN/m2] Pressure on bottom plate
    fdp : [N] shear force on bottom plate
    mdp : [N m] Bending moment on bottom plate
    l1 : [mm] Large dimension of panel
    b1 : [mm] Short dimension of panel
    '''

    #Define the local panel dimensions
    # Long dimension of plate in milimeter
    l1 = m.max2(sl-b1,st-bt)
    # Short dimension of plate in milimeters
    b1 = m.min2(sl-b1,st-bt)

    # PRESSURE REDUCTION FACTORS ISO 12215 part 5
    # Fixed value for Design Category B
    kDC = m.Const(0.8,name='kdcp')

    # Dynamic Load Factor
    nCG = 0.32*((lwl/(10*bc))+0.084)*(50-beta)*((v*bc)**2)/mldc
    if nCG >= 3:
        nCG = 0.5*v/(mldc**0.17)
    if nCG > 7:
        nCG = 7
    nCG = m.Const(nCG,name='ncgp')

    #Area pressure reduction factor
    ad1 = m.Intermediate((l1*b1)*1E-6,name='ad1')
    ad2 = m.Intermediate(2.5*(b1**2)*1E-6,name='ad2')
    AD = m.Intermediate(m.min2(ad1,ad2),name='ad')
    kAR = m.Intermediate(0.1*(mldc**0.15)/(AD**0.3),name='kar')

    #DESIGN PRESSURE, BENDING MOMENT AND SHEAR FORCE ISO 12215 part 5
    # Planing mode bottom pressure PBP [kN/m2]
    pbp = m.Intermediate(((0.1*mldc*(1+(kDC**0.5)*nCG))/(lwl*bc))*kAR,
        name='pbp')

    # Shear Strength aspect ratio factor kSHC for l/b<2
    # the formulation bellow, for l/b>4 kSHC = 0.5
    kshc1 = m.Intermediate(0.035+0.394*(l1/b1)
        -0.09*((l1/b1)**2),name='kshc1')

    kSHC = m.Intermediate(m.if3(2-(l1/b1),0.5,kshc1),name='kshc')

    # Panel Aspec Ratio factor k2 for l/b>2 k= 0.5
    a1 = m.Intermediate(0.271*((l1/b1)**2)
        +0.910*(l1/b1)-0.554,name='a1')
    a2 = m.Intermediate(((l1/b1)**2)-0.313*(l1/b1)+1.351,name='a2')
    k2 = m.Intermediate(m.if3(2-(l1/b1),0.5,a1/a2),name='k2')

    #[N/mm] shear force in middle of "b1"
    fdp = m.Intermediate(1*kSHC*pbp*b1*(1E-3),name = 'fdp')
    #[N mm/mm] Bending moment in "b1" direction
    mdp = m.Intermediate(83.33*2*k2*pbp*(b1**2)*(1E-6),name = 'mdp')
    return pbp, fdp, mdp, l1, b1

```

```

# THIS FUNCTION PARAMETERIZE THE CONSTRAINST FOR BOTTOM PLATE AND STIFFENERS
def bottom_optimize(lp,bp,nt,nl,h,b,hT,bT
                  ,st,lul,sl,pbt,pbl,fdt,fdl,mdt,mdl,mdp):
    '''OUTPUTS:
TOTAL_WEIGHT      : [Kg] Total bottom weight (plate + stiffeners)
PANEL_WEIGHT      : [Kg] Weight of bottom panel
STIFF_WEIGHT      : [Kg] Weight of bottom stiffeners
C1L,C2L,C3L,EIL,C45,C46 : Bottom Longitudinals stiffener constraints
C1T,C2T,C3T,EIT,C47,C48 : Bottom Transversal stiffeners constraints
C1P,C2P,C3P       : Bottom plate constraints'''

#STIMATE THE WEIGHT PER AREA, THICKNESS ADN YOUNG'S MODULUS
# OF BOTTOM PLATE
Amat = m.Intermediate(n1*Tmat[0]+n2*Tmat[1]
                    +n3*Tmat[2]+n4*Tmat[3],name='amat')
Awr = m.Intermediate(n5*Twr[0]+n6*Twr[1],name='awr')
Cmat = m.Intermediate(n1*Wmat[0]+n2*Wmat[1]
                    +n3*Wmat[2]+n4*Wmat[3],name='cmat')
Cwr = m.Intermediate(n5*Wwr[0]+n6*Wwr[1],name='cwr')
# Young's Modulus of bottom plate laminate [N/mm2]
Ep = m.Intermediate((6400** (4+2*(n1+n2+n3+n4)) * (13240** (2*(n5+n6))))
                    ** (1/(4+2*(n1+n2+n3+n4+n5+n6))),name='ep')
# Total laminated thickness [mm]
B_T = m.Intermediate((4*min(Tmat)+2*(Amat+Awr)),name='b_t')

#STIFFENER OPTIMIZATION
# [cm3] Req. SM in flange of Long. Stiff
SM_Req_Long = m.Intermediate((mdl)/(SigUT),name='smrl')
# [cm3] Req. SM in flange of Trans. Stiff
SM_Req_Trans = m.Intermediate((mdt)/(SigUT),name='smrt')
# [cm2] Req. shear area on webs of Long. Stiff
Awebs_Req_Long = m.Intermediate((fdl)/(100*Tao_MAT),name='awrl')
# [cm2] Req. shear area on webs of Long. Stiff
Acore_Req_Long = m.Intermediate((fdl)/(100*Tao_Core),name='acrl')
# [cm2] Req. shear area on webs of Trans. Stiff
Awebs_Req_Trans = m.Intermediate((fdt)/(100*Tao_MAT),name='awrt')
# [cm2] Req. shear area on webs of Trans. Stiff
Acore_Req_Trans = m.Intermediate((fdt)/(100*Tao_Core),name='acrt')

#BT_LS : thickness of bottom longitudinal stiffeners.
BT_LS = m.Intermediate(N_bl1*Tmat[0]+N_bl2*Tmat[1]
                    +N_bl3*Tmat[2]+N_bl4*Tmat[3],name='bt_ls')
#L_W : weight of bottom longitudinal stiffeners.
L_W = m.Intermediate(N_bl1*Wmat[0]+N_bl2*Wmat[1]
                    +N_bl3*Wmat[2]+N_bl4*Wmat[3],name='l_w')
#BT_TS : thickness of bottom transversal stiffeners.
BT_TS = m.Intermediate(N_bt1*Tmat[0]+N_bt2*Tmat[1]
                    +N_bt3*Tmat[2]+N_bt4*Tmat[3],name='bt_ts')
#T_W : weight of bottom transversal stiffeners.
T_W = m.Intermediate(N_bt1*Wmat[0]+N_bt2*Wmat[1]
                    +N_bt3*Wmat[2]+N_bt4*Wmat[3],name='t_w')

# LONGITUDINALS STIFFENERS CALCULATIONS
a = m.Intermediate(10+(N_bl1+N_bl2+N_bl3+N_bl4)*15,name='a')
wp = m.Intermediate(20*B_T+b,name='wp')
EA = m.Intermediate(((Ep*wp*B_T)
                    +(Ec*b*h)
                    +(Er*(a*BT_LS+2*h*BT_LS+(b+2*BT_LS)*BT_LS))))
,name='EAL')
EAZ = m.Intermediate(((Ep*wp*B_T*(0.5*B_T))
                    +(Ec*b*h*(0.5*h+B_T))

```

```

+ (Er * ((a * BT_LS * (B_T + 0.5 * BT_LS)) + 2 * h * BT_LS * (B_T + 0.5 * h)
+ BT_LS * (b + 2 * BT_LS) * (B_T + h + 0.5 * BT_LS))))
, name = 'EAZL')
NA = m.Intermediate(EAZ/EA, name = 'NAL')
Zcri = m.Intermediate(B_T + h + 0.5 * BT_LS - NA, name = 'Zcri')
EAZ2 = m.Intermediate(((Ep * wp * B_T * 0.25 * (B_T ** 2)
+ (Ec * ((b * (NA - B_T) * 0.25 * (NA + B_T) ** 2) + b * (h - NA + B_T)
* (0.25 * (NA + h + B_T) ** 2))))
+ (Er * ((2 * BT_LS * (NA - B_T) * 0.25 * (NA + B_T) ** 2)
+ (2 * BT_LS * (h - NA + B_T) * 0.25 * (NA + h + B_T) ** 2))))
+ (Er * ((a * BT_LS * (B_T + 0.5 * BT_LS) ** 2)
+ (b + 2 * BT_LS) * BT_LS * ((B_T + h + 0.5 * BT_LS) ** 2))))
, name = 'EAZ2L')
Ebh3 = m.Intermediate((1/12) * ((Ep * wp * (B_T ** 3))
+ (Ec * b * ((h - NA + B_T) ** 3) + ((NA - B_T) ** 3)))
+ (Er * (a * (BT_LS ** 3) + (b + 2 * BT_LS) * (BT_LS ** 3))))
, name = 'Ebh3L')
EIna = m.Intermediate(EAZ2 + Ebh3 - (NA ** 2) * EA, name = 'EInaL')
Qc = m.Intermediate(((Ec * 0.5 * b * (h + B_T - NA) ** 2)
+ (Er * (Zcri * BT_LS * (b + 2 * BT_LS) + BT_LS * (h + B_T - NA) ** 2)))
, name = 'QcL')
Qw = m.Intermediate(((Ep * (NA - 0.5 * B_T) * wp * B_T)
+ (Er * ((NA - B_T - 0.5 * BT_LS) * a * BT_LS))) , name = 'QwL')
EImIn = m.Intermediate((26 * (1 ** 1.5) * (lul ** 2) * fdl * 0.001
/ (5 * 0.05)) , name = 'EImInL')

# Define longitudinal stiffeners constraints
# [cm3] actual section modulus for long. stiff
SML = m.Intermediate((0.001 * EIna / (Er * Zcri)) , name = 'sml')
C1L = m.Intermediate((SML / SM_Req_Long) , name = 'c1l')
# [cm2] actual web shear area for long. stiff
AWL = m.Intermediate((0.01 * EIna * 2 * BT_LS / Qw) , name = 'awl')
C2L = m.Intermediate((AWL / Awebs_Req_Long) , name = 'c2l')
# [cm2] actual core shear area for long. stiff
ACL = m.Intermediate((0.01 * EIna * b / Qc) , name = 'acl')
C3L = m.Intermediate((ACL / Acore_Req_Long) , name = 'c3l')
# [cm2] actual core shear area for long. stiff
EIL = m.Intermediate((EIna / EImIn) , name = 'eil')
# realtion for controle the local buckling in webs
C45 = m.Intermediate((h / BT_LS) , name = 'c45')
# realtion for controle the local buckling in flange
C46 = m.Intermediate((b / BT_LS) , name = 'c46')

# TRANSVERSAL STIFFENERS
aT = m.Intermediate((10 + (N_bt1 + N_bt2 + N_bt3 + N_bt4) * 15) , name = 'aT')
wpT = m.Intermediate(20 * B_T + bT , name = 'wpT')
EAT = m.Intermediate((Ep * (wpT * B_T)
+ (Ec * bT * hT)
+ (Er * (aT * BT_TS + 2 * hT * BT_TS + (bT + 2 * BT_TS) * BT_TS)))
, name = 'EAT')
EAZT = m.Intermediate(((Ep * wpT * B_T * (0.5 * B_T))
+ (Ec * bT * hT * (0.5 * hT + B_T))
+ (Er * ((aT * BT_TS * (B_T + 0.5 * BT_TS))
+ 2 * hT * BT_TS * (B_T + 0.5 * hT)
+ BT_TS * (bT + 2 * BT_TS) * (B_T + hT + 0.5 * BT_TS))))
, name = 'EAZT')
NAT = m.Intermediate(EAZT / EAT , name = 'NAT')
ZcriT = m.Intermediate(B_T + hT + 0.5 * BT_TS - NAT , name = 'ZcriT')
EAZ2T = m.Intermediate(((Ep * wpT * B_T * 0.25 * (B_T ** 2))
+ (Ec * (bT * (NAT - B_T) * 0.25 * ((NAT + B_T) ** 2) +
(bT * (hT - NAT + B_T) * (0.25 * (NAT + hT + B_T) ** 2))))

```

```

+ (Er * ((2 * BT_TS * (NAT - B_T) * 0.25 * (NAT + B_T) ** 2) +
          (2 * BT_TS * (hT - NAT + B_T) * 0.25 * (NAT + hT + B_T) ** 2)))
+ (Er * ((aT * BT_TS * (B_T + 0.5 * BT_TS) ** 2) +
          (bT + 2 * BT_TS) * BT_TS * ((B_T + hT + 0.5 * BT_TS) ** 2))))
,name='EAZ2T')

Ebh3T = m.Intermediate((1/12) * ((Ep * wpT * (B_T ** 3))
                                + (Ec * bT * ((hT - NAT + B_T) ** 3) + ((NAT - B_T) ** 3)))
                        + (Er * 2 * BT_TS * ((hT - NAT + B_T) ** 3) +
                          ((NAT - B_T) ** 3)))
                        + (Er * (aT * (BT_TS ** 3)
                                + (bT + 2 * BT_TS) * (BT_TS ** 3))))
,name='Ebh3T')

EInaT = m.Intermediate(EAZ2T + Ebh3T - (NAT ** 2) * EAT, name='EInaT')
Qc_T = m.Intermediate(((Ec * 0.5 * bT * (hT + B_T - NAT) ** 2)
                      + (Er * (ZcriT * BT_TS * (bT + 2 * BT_TS) +
                                BT_TS * (hT + B_T - NAT) ** 2)))
,name='QcT')

Qw_T = m.Intermediate(((Ep * (NAT - 0.5 * B_T) * wpT * B_T)
                      + (Er * ((NAT - B_T - 0.5 * BT_TS) * aT * BT_TS))), name='QwT')

EImint = m.Intermediate(((26 * (1 ** 1.5) * (sl ** 2) * fdt * 0.001)
                        / (5 * 0.05)), name='EImint')

#Define transversal stiffeners constraints
# [cm3] actual section modulus for trans stiff
SMT = m.Intermediate(0.001 * EInaT / (Er * ZcriT), name='smt')
C1T = m.Intermediate((SMT / SM_Req_Trans), name='c1t')
# [cm2] actual web shear area for trans stiff
AWT = m.Intermediate((0.01 * EInaT * 2 * BT_TS / Qw_T), name='awt')
C2T = m.Intermediate((AWT / Awebs_Req_Trans), name='c2t')
# [cm2] actual core shear area for trans stiff
ACT = m.Intermediate((0.01 * EInaT * bT / Qc_T), name='act')
C3T = m.Intermediate((ACT / Acore_Req_Trans), name='c3t')
# relation of stiffness for trans stiff
EIT = m.Intermediate((EInaT / EImint), name='eit')
# relation for controle the local buckling in webs
C47 = m.Intermediate((hT / BT_TS), name='c47')
# relation for controle the local buckling in flange
C48 = m.Intermediate((bT / BT_TS), name='c48')

#TOTAL STIFFENERS WEIGHT
L_Weight = m.Intermediate(((L_W * nl * lul / 300) * ((2 * h + b + a) / 1000) +
                          (h * b * lul * nl * dcore / (1000 ** 3))), name='LS_weight')

T_Weight = m.Intermediate(((T_W * nt * bp / 0.3) * ((2 * hT + bT + aT) / 1000)
                          + (hT * bT * nt * bp * dcore / 1000 ** 2))
,name='TS_weight')

#weight per length of mat in trans. stiff.
twl = m.Intermediate((T_W / 0.3) * ((2 * hT + bT + aT) / 1000), name='twlayer')
#weight per length of core in trans. stiff.
twc = m.Intermediate(hT * bT * dcore / 1000 ** 2, name='twcore')

#PLATE OPTIMIZATIONS
# Intermediates to compute the constraints
# Sectional Inercia
I = m.Intermediate((((4 * min(Tmat) + 2 * (Amat + Awr)) ** 3) / 12), name='I')
# Distance to critical fiber
Zcri = m.Intermediate((Amat + Awr) + 1.5 * min(Tmat), name='Zcrip')
# Acting tension stress
Sact = m.Intermediate((mdp / (Ep * I / (Zcri * 6400))), name='Sact')

```

```

# Total Panel dimension in squared meters
Panel_Dim = lp*bp
# Local panel dimension in squared meters
LPanel_Dim = m.Intermediate((sl-b)*(st-bT)/(1000**2),name='Lpanel')

C1P = m.Intermediate(n1+n2+n3+n4-n5-n6,name='c1p')
C2P = m.Const((0.0675+0.0135*v*0.0675*(mldc**0.33)),name='c2p')
C3P = m.Intermediate((4*min(Wmat)/0.3)
                    +2*((Cmat/0.3)+(Cwr/0.48)),name='c3p')

#DEFINE THE OBJECTIVE FUNCTION FOR BOTTOM PANEL (panel weight in Kg )
PANEL_WEIGHT = m.Intermediate(Panel_Dim*C3P,name='pw')
STIFF_WEIGHT = m.Intermediate((L_Weight+T_Weight),name='sw')
TOTAL_WEIGHT = PANEL_WEIGHT+STIFF_WEIGHT

return (TOTAL_WEIGHT,PANEL_WEIGHT,STIFF_WEIGHT,
        C1L,C2L,C3L,EIL,C45,C46,C1T,C2T,C3T,EIT,C47,C48,
        Sact,C1P,C2P,C3P,L_Weight,T_Weight,
        B_T,Ep,twl,twc,B_T)

```

```
#=====DESIGNVARIABLES=====
```

```
''' Comment
```

```
Design Variable FASE 1 (Plate Optimization)
```

```
NL : Number of stiffeners "z" direction1 (Integer) (bottom)
```

```
NT : Number of stiffeners "x" direction1 (Integer) (bottom and side)
```

```
NLS: Number of stiffeners "z" direction1 (Integer) (side)
```

```
bottom
```

```
n1 : Number of MAT 300 layer (Integer)
```

```
n2 : Number of MAT 350 layer (Integer)
```

```
n3 : Number of MAT 450 layer (Integer)
```

```
n4 : Number of MAT 600 layer (Integer)
```

```
n5 : Number of WR 600 layer (Integer)
```

```
n6 : Number of WR 800 layer (Integer)
```

```
side
```

```
ns1 : Number of MAT 300 layer (Integer)
```

```
ns2 : Number of MAT 350 layer (Integer)
```

```
ns3 : Number of MAT 450 layer (Integer)
```

```
ns4 : Number of MAT 600 layer (Integer)
```

```
ns5 : Number of WR 600 layer (Integer)
```

```
ns6 : Number of WR 800 layer (Integer)
```

```
Design Variables FASE 2 (Stiffeners optimization)
```

```
N_bl1 : Number of MAT 300 layer in bottom longitudinal stiffeners (Integer)
```

```
N_bl2 : Number of MAT 350 layer in bottom longitudinal stiffeners (Integer)
```

```
N_bl3 : Number of MAT 450 layer in bottom longitudinal stiffeners (Integer)
```

```
N_bl4 : Number of MAT 600 layer in bottom longitudinal stiffeners (Integer)
```

```
N_sl1 : Number of MAT 300 layer in side longitudinal stiffeners (Integer)
```

```
N_sl2 : Number of MAT 350 layer in side longitudinal stiffeners (Integer)
```

```
N_sl3 : Number of MAT 450 layer in side longitudinal stiffeners (Integer)
```

```
N_sl4 : Number of MAT 600 layer in side longitudinal stiffeners (Integer)
```

```
hl : height of longitudinal stiffeners [mm] (bottom) (Real)
```

```
bl : width of longitudinal stiffeners [mm] (bottom) (Real)
```

```
hls : height of longitudinal stiffeners [mm] (side) (Real)
```

```
bls : width of longitudinal stiffeners [mm] (side) (Real)
```

```
N_st1 : Number of MAT 300 layer in side transversal stiffeners (Integer)
```

```
N_st2 : Number of MAT 350 layer in side transversal stiffeners (Integer)
```

```
N_st3 : Number of MAT 450 layer in side transversal stiffeners (Integer)
```

```
N_st4 : Number of MAT 600 layer in side transversal stiffeners (Integer)
```

```
N_bt1 : Number of MAT 300 layer in bottom transversal stiffeners (Integer)
```

```
N_bt2 : Number of MAT 350 layer in bottom transversal stiffeners (Integer)
```

```
N_bt3 : Number of MAT 450 layer in bottom transversal stiffeners (Integer)
```

```
N_bt4 : Number of MAT 600 layer in bottom transversal stiffeners (Integer)
```

```
hT : height of transversal stiffeners [mm] (bottom) (Real)
```

```
bT : width of transversal stiffeners [mm] (bottom) (Real)
```

```
hTs : height of transversal stiffeners [mm] (side) (equal to hT) (Real)
```

```
bTs : width of transversal stiffeners [mm] (side) (equal to bT) (Real)
```

```
'''
```

MAIN PROGRAM (bottom)

```
#=====DEFINE SHIP PARAMETERS=====
LH = 12.0      # [m] Hull length
Lwl = 10.74    # [m] Legnth on design water line
BC = 2.93      # [m] Chine beam in meters
beta = 13      # [°] Deadrise angle
V = 25         # [kn] Maximun velocity
mLDC = 8864    # [kg] Design Displacement
T = 0.51       # [m] Draft
D = 1.5        # [m] Depth

#DEFINE FUNCTION PARAMETERS
Lp = 4         # 4x1.5 [m] Global Panel dimension in "x" direction (Aft/Fwr)
B = 1.5
Bs = 1.2

# it is necessary change the name of variable for functions calculations
lh,lwl,bc,v,mldc,lp,b,bside,d,t = LH,Lwl,BC,V,mLDC,Lp,B,Bs,D,T

#Permisibles stress, (Reverse Safe Factor = 0.4)
SigUT = 0.4*85      # N/mm2 ultimate tension stress MAT
Taoi_MAT = 0.4*17.25 # N/mm2 ultimate intralaminar stress MAT
Taoi_WR = 0.4*14.1  # N/mm2 ultimate intralaminar stress WR
Tao_MAT = 0.4*62    # N/mm2 ultimate shear strength stress MAT
Tao_Core = 0.4*16.7 # N/mm2 ultimate shear strength stress CORE

Ec = 17240          # N/mm2 young's modulus for Core material (CHANUL)
Er = 6400           # N/mm2 young's modulus for MAT fibers

#Material properties
dres = 1.2          # g/cm3 Polyester resine density
dmat = 2.55         # g/cm3 Mat ply density
dwr = 2.56         # g/cm3 Waven Roven ply density
Wmat = [0.3,0.35,0.45,0.6] # weigth per area (kg/m2)
                                # of Mat plies (300,350,450,600) grams.
Wwr = [0.6,0.8]    # weigth per area (kg/m2) of WR plies (600,800)
grams
dkeel = 870        # Kg/m3 Keel Core density (chanul)
dcore = 870        # Kg/m3 Stiffeners Core density (chanul)

#Thickness of each layer
Tmat = np.zeros(len(Wmat))
Twr = np.zeros(len(Wwr))
for i in range(len(Wmat)):
    Tmat[i] = (Wmat[i]/(dres*dmat))*((dmat/0.3)-(dmat-dres))
for i in range(len(Wwr)):
    Twr[i] = (Wwr[i]/(dres*dwr))*((dwr/0.48)-(dwr-dres))

#=====OPTIMIZATION OF BOTTOM PANELS=====
#=====INITIAL POINT FOR BOTTOM OPTIMIZATION=====
n1, n2, n3, n4, n5, n6 = 1,0,0,0,0,0
NL, NT =4,5
N_b11, N_b12, N_b13, N_b14 = 5,1,2,4
N_bt1, N_bt2, N_bt3, N_bt4 = 2,1,2,4
hL ,bL ,hT ,bT = 150,50,100,50

#=====VECTORS WITH INITIAL POINTS=====
Nmat = [n1,n2,n3,n4]
Nwr = [n5,n6]
BL_Mat = [N_b11, N_b12, N_b13, N_b14]
```



```

BT_Mat = [N_bt1, N_bt2, N_bt3, N_bt4]
DIM      = [hL ,bL ,hT ,bT]

na= [n1,n2,n3,n4,n5,n6] # Initial point for gekko na = [n1,n2,n3,n4,n5,n6]
NR = [NT,NL]           # Intial point for gekko NR = [NT,NL]

#===== GEKKO FIRST FASE INTERACTIONS =====
print(74*"=")
print(74*"=")
print(28*"=","START OPTIMIZATION",28*"=")
start_time = time.time()
print(74*"=")
from gekko import GEKKO
m = GEKKO(remote=False) # Initialize gekko
m.options.IMODE = 2      # solution mode 2 or 3
m.options.SOLVER= 1      # (1=APOPT, 3=IPOPT) APOPT is an MINLP solver
# optional solver settings with APOPT
m.solver_options = ['minlp_maximum_iterations 5000', \
                    # minlp iterations with integer solution
                    'minlp_max_iter_with_int_sol 5000', \
                    # treat minlp as nlp
                    'minlp_as_nlp 0', \
                    # nlp sub-problem max iterations
                    'nlp_maximum_iterations 1500', \
                    # 1 = depth first, 2 = breadth first , 3 , 4
                    'minlp_branch_method 3', \
                    #maximum value to be considered as an integer
                    'minlp_integer_max 2.0e9'\
                    # maximum deviation from whole number
                    'minlp_integer_tol 1e-5', \
                    # convergence tolerance
                    'minlp_gap_tol 0.001'\
                    #objective convergence tolerance 1.0e-6
                    'constraint_convergence_tolerance 1.0e-6'\
                    #convergence tolerance for the constraints
                    'objective_convergence_tolerance 1.0e-6']

#===== DESIGN VARIABLES.=====
print(8*".." ,"Initial point for bottom optimization",9*"..")
print("Number of Stiffeners [NT,NL] : ",NR)
print("")
print("Bottom Layers [n1,n2,n3,n4,n5,n6] : ",na)
print("")
print("Long. Stiffeners layers [bln1,bln2,bln3,bln4] : ",BL_Mat)
print("")
print("Trans. Stiffeners layer [btn1,btn2,btn3,btn4] : ",BT_Mat)
print("")
print("Stiffeners dimension [hL,bL,hT,bT] [mm] : ",DIM)
print(37*"..")
#=====first fase design variables.=====
n1 = m.Var(value=na[0],lb=0,ub=20,integer=True,name='n1')
n2 = m.Var(value=na[1],lb=0,ub=20,integer=True,name='n2')
n3 = m.Var(value=na[2],lb=0,ub=20,integer=True,name='n3')
n4 = m.Var(value=na[3],lb=0,ub=20,integer=True,name='n4')
n5 = m.Var(value=na[4],lb=0,ub=20,integer=True,name='n5')
n6 = m.Var(value=na[5],lb=0,ub=20,integer=True,name='n6')

NT = m.Var(value=NR[0],lb=0,ub=10,integer=True,name='NT')
NL = m.Var(value=NR[0],lb=0,ub=3 ,integer=True,name='NL')

#=====second fase design variables.=====

```

```

N_b11 = m.Var(value=BL_Mat[0],lb=0,ub=20,integer=True,name='N_b11')
N_b12 = m.Var(value=BL_Mat[1],lb=0,ub=20,integer=True,name='N_b12')
N_b13 = m.Var(value=BL_Mat[2],lb=0,ub=20,integer=True,name='N_b13')
N_b14 = m.Var(value=BL_Mat[3],lb=0,ub=20,integer=True,name='N_b14')
N_bt1 = m.Var(value=BT_Mat[0],lb=0,ub=20,integer=True,name='N_bt1')
N_bt2 = m.Var(value=BT_Mat[1],lb=0,ub=20,integer=True,name='N_bt2')
N_bt3 = m.Var(value=BT_Mat[2],lb=0,ub=20,integer=True,name='N_bt3')
N_bt4 = m.Var(value=BT_Mat[3],lb=0,ub=20,integer=True,name='N_bt4')

hL = m.Var(value=DIM[0],lb=40,ub=400,integer=False,name='hl')
bL = m.Var(value=DIM[1],lb=40,ub=400,integer=False,name='bl')
hT = m.Var(value=DIM[2],lb=40,ub=400,integer=False,name='ht')
bT = m.Var(value=DIM[3],lb=40,ub=400,integer=False,name='bt')

#=====GEKKO COMPUTATION.=====
#Stimulation of space and span stiffeners
sT, luL, sL = span_space_stiff(NT,NL)

#Stimulation of stiffeners forces
pb_t,pb_l,fd_t,fd_l,md_t,md_l = stiff_force(luL,sL,sT)

#Stimulation of plate bending moment and shear force
pb_p,fd_p,md_p,l_l,b_l = plate_force(bL,bT,sL,sT)

#Plate optimization.
(TOTAL_WEIGHT,PANEL_WEIGHT,STIFF_WEIGHT,
 C1L,C2L,C3L,EIL,C45,C46,C1T,C2T,C3T,EIT,
 C47,C48,Sact,C1P,C2P,C3P, L_Weight,T_Weight,
 B_T,EP,tw_l,tw_c,B_thick) = bottom_optimize(Lp,B,NT,NL,hL,bL,hT,bT,sT,luL,
 sL,pb_t,pb_l,fd_t,fd_l,md_t,
 md_l,md_p)

#=====EQUATIONS.=====
#=====first fase constraints.=====
#Same number of Mat layers and Wr layers into sub laminate
CON1 = m.Equation(C1P == 0)

#Tension Stress acting less than design tension stress
CON2 = m.Equation((SigUT/Sact) -1 >= 0)
c2 = m.Intermediate(SigUT/Sact)

#Minimun weight for impact loads
CON3 = m.Equation((C3P/C2P) -1 >= 0)
c3 = m.Intermediate(C3P/C2P)

#=====second fase constraints.=====
# Tension Stress acting on top flange of longitudinal stiffener
# must be less than design tension stress for fiber
CON4 = m.Equation(C1L -1 >= 0)
c4 = m.Intermediate(C1L)

# Shear stress acting longitudinal stiffeners webs must be
# less than design shear stress for fiber material
CON5 = m.Equation(C2L -1 >= 0)
c5 = m.Intermediate(C2L)

# Shear stress acting longitudinal stiffeners core must be
# less than design shear stress for core material
CON6 = m.Equation(C3L - 1 >= 0)
c6 = m.Intermediate(C3L)

```

```

# Longitudinal stiffener deflection must be less
# than permissible deflection
CON7 = m.Equation(EIL - 1 >= 0)
c7 = m.Intermediate(EIL)

# Local buckling for webs and flange of longitudinal stiffeners
CON14 = m.Equation(1 - (C45/30) >= 0)
CON15 = m.Equation(1 - (C46/21) >= 0)
c14 = m.Intermediate(C45/30)
c15 = m.Intermediate(C46/21)

# Tension Stress acting on top flange of transversal stiffener
# must be less than design tension stress for fiber
CON8 = m.Equation(C1T - 1 >= 0)
c8 = m.Intermediate(C1T)

# Shear stress acting transversal stiffeners webs must be less
# than design shear stress for fiber material
CON9 = m.Equation(C2T - 1 >= 0)
c9 = m.Intermediate(C2T)

# Shear stress acting transversal stiffeners core must
# be less than design shear stress for core material
CON10 = m.Equation(C3T - 1 >= 0)
c10 = m.Intermediate(C3T)

# Transversal Stiffener deflection must be less
# than permissible deflection
CON11 = m.Equation(EIT - 1 >= 0)
c11 = m.Intermediate(EIT)

# Local buckling for webs and flange of transversal stiffeners:
CON16 = m.Equation(1 - (C47/30) >= 0)
CON17 = m.Equation(1 - (C48/21) >= 0)
c16 = m.Intermediate(C47/30)
c17 = m.Intermediate(C48/21)
#====Objective function.=====
#Objective function for first fase
Objective = m.Obj(TOTAL_WEIGHT) # Objective function, panel weight
minimization
#====CALL SOLVER.=====
#Solve and print results
output = m.solve(dis= True) # Solver
#====New Values.=====
panel = PANEL_WEIGHT.value[0]
stiff = STIFF_WEIGHT.value[0]
OBJ1_act = m.options.objfcnval
OBJ_old = OBJ1_act
na = [n1.value[0],n2.value[0],n3.value[0],n4.value[0],
      ,n5.value[0],n6.value[0]]
Nmat = [n1.value[0],n2.value[0],n3.value[0],n4.value[0]]
Nwr = [n5.value[0],n6.value[0]]
NR = [NT.value[0],NL.value[0]]
BL_Mat = [N_bl1.value[0], N_bl2.value[0], N_bl3.value[0], N_bl4.value[0]]
BT_Mat = [N_bt1.value[0], N_bt2.value[0], N_bt3.value[0], N_bt4.value[0]]
DIM = [hL.value[0], bL.value[0], hT.value[0], bT.value[0]]
print(50*" ")
print(50*" ")
if NT.value[0]*NL.value[0] > 0:
    print(50*" ")
    print(18*" ", "Eval. Const.", 18*" ")

```

```

print("Const. Flex. Plate      >= 1 :", round(c2.value[0],3))
print("Const. Num. Layer      == 0 :", round(C1P.value[0],3))
print("Const. Impact Loads    >= 1 :", round(c3.value[0],3))
print("Const. Tension Long     >= 1 :", round(c4.value[0],3))
print("Const. Shear Webs Long  >= 1 :", round(c5.value[0],3))
print("Const. Shear Core Long  >= 1 :", round(c6.value[0],3))
print("Const. Stiffness Long   >= 1 :", round(c7.value[0],3))
print("Const. Buck. Webs Long  <= 1 :", round(c14.value[0],3))
print("Const. Buck. Core Long  <= 1 :", round(c15.value[0],3))
print("Const. Tension Trans    >= 1 :", round(c8.value[0],3))
print("Const. Shear Webs Trans >= 1 :", round(c9.value[0],3))
print("Const. Shear Core Trans >= 1 :", round(c10.value[0],3))
print("Const. Stiffness Trans  >= 1 :", round(c11.value[0],3))
print("Const. Buck. Webs Trans  <= 1 :", round(c16.value[0],3))
print("Const. Buck. Core Trans  <= 1 :", round(c17.value[0],3))
print(50*"")
print(50*"")
elif NT.value[0]==0 and NL.value[0] != 0:
print(50*"")
print(18*"", "Eval. Const.", 18*"")
print("Const. Flex. Plate      >= 1 :", round(c2.value[0],3))
print("Const. Num. Layer      == 0 :", round(C1P.value[0],3))
print("Const. Impact Loads    >= 1 :", round(c3.value[0],3))
print("Const. Tension Long     >= 1 :", round(c4.value[0],3))
print("Const. Shear Webs Long  >= 1 :", round(c5.value[0],3))
print("Const. Shear Core Long  >= 1 :", round(c6.value[0],3))
print("Const. Stiffness Long   >= 1 :", round(c7.value[0],3))
print("Const. Buck. Webs Long  <= 1 :", round(c14.value[0],3))
print("Const. Buck. Core Long  <= 1 :", round(c15.value[0],3))
print(50*"")
print(50*"")
elif NL.value[0] == 0 and NT.value[0] != 0:
print(50*"")
print(18*"", "Eval. Const.", 18*"")
print("Const. Flex. Plate      >= 1 :", round(c2.value[0],3))
print("Const. Num. Layer      == 0 :", round(C1P.value[0],3))
print("Const. Impact Loads    >= 1 :", round(c3.value[0],3))
print("Const. Tension Trans    >= 1 :", round(c8.value[0],3))
print("Const. Shear Webs Trans >= 1 :", round(c9.value[0],3))
print("Const. Shear Core Trans >= 1 :", round(c10.value[0],3))
print("Const. Stiffness Trans  >= 1 :", round(c11.value[0],3))
print("Const. Buck. Webs Trans  <= 1 :", round(c16.value[0],3))
print("Const. Buck. Core Trans  <= 1 :", round(c17.value[0],3))
print(50*"")
print(50*"")
print(50*"")

#=====SHOW THE BEST SOLUTION.=====
nplies = 4+2*sum(na) #Number of plies
print(50*"")
print("CONGRATULATIONS YOU GET THE SOLUTION")
print(50*"")
print("Number of Stiffeners          [NT,NL] : ",NR)
if NT.value[0]*NL.value[0] > 0:
print("Stiffeners dimensions      [hL,bL,hT,bT] [mm] : ",
      [round(DIM[0],2), round(DIM[1],2)
       , round(DIM[2],2), round(DIM[3],2)])
elif NT.value[0]==0 and NL.value[0] != 0:
print("Stiffeners dimensions      [hL,bL] [mm] : ",
      [round(DIM[0],2), round(DIM[1],2)])
elif NL.value[0] == 0 and NT.value[0] != 0:

```

```

        print("Stiffeners dimensions [hT,bT] [mm] : ",
              [round(DIM[2],2),round(DIM[3],2)])
print(" ")
print("Bottom Layers [n1,n2,n3,n4,n5,n6] : ",na)
print("Total Plies : ",[nplies])
print("Total Thickness [mm] : ",
      [round(B_thick.value[0],3)])
print(" ")
if NT.value[0]*NL.value[0] > 0:
    print("Long. Stiffeners layers [bln1,bln2,bln3,bln4] : ",BL_Mat)
    print("Trans. Stiffeners layer [btn1,btn2,btn3,btn4] : ",BT_Mat)
elif NT.value[0]==0 and NL.value[0] != 0:
    print("Long. Stiffeners layers [bln1,bln2,bln3,bln4] : ",BL_Mat)
elif NL.value[0] == 0 and NT.value[0] != 0:
    print("Trans. Stiffeners layer [btn1,btn2,btn3,btn4] : ",BT_Mat)
print(" ")
print("Global Panel weight [Kg] : ",[round(panel,3)])
print("Global Stiffeners weight [Kg] : ",[round(stiff,3)])
print(" ")
print('Total Weight [Kg] : '
      + str([round(m.options.objfcnval,3)]))
# print(" ")
print("Time: --- %s seconds ---" % round((time.time() - start_time),4))

# GRÁFICA DE FUNCIÓN OBJETIVO
# Leer datos

file_names=glob.glob(m._path+'\\*.out')
file=file_names[0]
data=pd.read_csv(file,header=1,skiprows=2,delimiter= '\s+',decimal='.')

#Quitar columnas de strings
df = pd.DataFrame(data=data)
df = df[df['Iter:'].str.contains('Iter:')]
indexNames = df[ df['I:'] == 'I:-11' ].index
df.drop(indexNames , inplace=True)
df.drop(df.columns[np.arange(0,15,2)], axis=1,inplace=True)

#Tabla final
df=df.T.reset_index().T
df.columns=['Iter','I','Tm','NLPi','Dpth','Lvs','Obj','Gap']
df.index=np.arange(0,len(df.index))

plt.figure(figsize=(10,5),facecolor='#ffffff')
#plt.scatter(df['Iter'], df['Obj'])
df['Gap'] = df['Gap'].fillna('0.6')
df['Gap'][0]=0.6
plt.plot(df['Iter'].astype(float), df['Gap'].astype(float),
         c='r',lw=0.75,label=r'MC',ls='-')
plt.suptitle('Gap ', fontsize=25)
plt.xlabel('Iteration', fontsize=18)
plt.ylabel('Gap', fontsize=16)
#plt.grid(ls=':',c='#4f4f4f')
plt.legend(loc=1,facecolor='#ffffff',fontsize=6.1,framealpha=0.1)
plt.xticks(np.arange(df['Iter'].astype(float).min(),
                    df['Iter'].astype(float).max()+1, 50))
plt.yticks(np.arange(df['Gap'].astype(float).min(),
                    df['Gap'].astype(float).max()+1, 0.1))

plt.tight_layout()
plt.show()

```

APÉNDICE C

CÓDIGO PYTHON- OPTIMIZACIÓN DEL COSTADO

```
#SIDE OPTIMIZATION
# SIDE FUNCTIONS
#THIS FUNCTION STIMATE THE SPAN AND SPACE OF STIFFENERS
def span_space_stiff_s(nls,bs):
    """
    INPUT
    nsl  : numbers of longitudinals stiffeners
    bs   : [m] dimension of side panel in "z" direction deck-chine
    OUTPUTS:
    luls : [mm] Span of longitudinals stiffeners
    sls  : [mm] Space of between longitudinals stiffener
    """
    luls = m.Const(1000*lp,name='luls')
    sls  = m.Intermediate(1000*bs/(nls+1),name='sls')
    return sls,luls

# THIS FUNCTION STIMATE THE PRESSURE, FORCE AND BENDING MOMENT ACTING
# ON STIFFENERS USING ISO 12215-PART5 EXPRESSIONS
def stiff_force_s(luls,sls,pb_l,d,t):
    """
    OUTPUT
    pst  : [kN/m2] Pressure of side transversal stiffeners
    psl  : [kN/m2] Pressure of side longitudinals stiffeners
    fdst : [N]      Design shear force for transversal stiffeners
    fdsl : [N]      Design shear force for longitudinal stiffeners
    mdst : [N m]   Design bending moment for transversal stiffeners
    mds1 : [N m]   Design bending moment for longitudinal stiffeners
    """
    # PRESSURE REDUCTION FACTORS ISO 12215 part 5
    # Fixed value for Design Category B
    kDC = m.Const(0.8,name='kdcr')

    # Dynamic Load Factor in "g" units
    nCG = (0.32*((lwl/(10*bc))+0.084)*(50-beta)*(((v*bc)**2)/mldc))
    if nCG >= 3:
        nCG = 0.5*v/(mldc**0.17)
    if nCG > 7:
        nCG = 7
    nCG = m.Const(nCG,name='ncgr')

    #Area pressure reduction factor
    adl1 = m.Intermediate(0.33*(luls**2)*1e-6,name='adl1')
    adl2 = m.Intermediate((luls*sls)*1e-6,name='adl2')
    ADL = m.Intermediate(m.max2(adl1,adl2),name = 'adl')
    karl1 = m.Intermediate((0.1*(mldc**0.15)/(ADL**0.3)),name='karl1')
    KARL = m.Intermediate(m.max2(0.25,karl1),name='karl')

    #DESIGN PRESSURE ISO 12215 part 5
    #Minimun design pressure in kN/m2
    #Motor craft side pressure in planing mode in kN/m2
    z =m.Const(d-t)
    h =m.Intermediate(z/2,name='hs')
    kz =m.Intermediate((z-h)/z,name='kzs')
```

```

psl = m.Intermediate(((0.35*lw1+14.6+kz*(0.25*(2.24*(mldc**0.33)+20)
                    -(0.35*lw1+14.6))))
                    *kARL*kDC,name='psl') #[kN/m2]

#STIMATE DESIGN FORCE [N] AND DESIGN BENDING MOMENT [Nm]e
fds1 = m.Intermediate(5*psl*sls*luls*1E-4,name='fds1') # [N]
mds1 = m.Intermediate(83.33*psl*sls*(luls**2)*1E-9,name='mds1') # [N m]

return psl,fds1,mds1

#THIS FUNCTION STIMATE THE FORCE AND BENDING MOMENT ACTING ON THE PLATE
# USING ISO 12215 PART-5 EXPRESSIONS
def plate_force_s(bls,sls,blss,pb_p,d,t):
'''
OUTPUT
pbp : [kN/m2] Pressure on bottom plate
psp : [kN/m2] Pressure on side plate
fdp : [N/mm] shear force on bottom plate
mdp : [N mm/mm] Bending moment on bottom plate
fdsp : [N/mm] shear force on side plate
mdsp : [N mm/mm] Bending moment on side plate
'''
#Define the local panel dimensions
# Long dimension of plate in milimeterter
lls = m.Intermediate(m.max2(sls-bls,blss),name='lls')
# Short dimension of plate in milimeters
bls = m.Intermediate(m.min2(sls-bls,blss),name='bls')

# PRESSURE REDUCTION FACTORS ISO 12215 part 5
# Fixed value for Design Category B
kDC = m.Const(0.8,name='kdcp')

# Dynamic Load Factor
nCG = 0.32*((lw1/(10*bc))+0.084)*(50-beta)*(((v*bc)**2)/mldc)
if nCG >= 3:
    nCG = 0.5*v/(mldc**0.17)
if nCG > 7:
    nCG = 7
nCG = m.Const(nCG,name='ncgp')

#Pressure area reduction factor
ad1 = m.Intermediate((lls*bls)*1E-6,name='ad1')
ad2 = m.Intermediate(2.5*(bls**2)*1E-6,name='ad2')
AD = m.Intermediate(m.min2(ad1,ad2),name='ad')
kar1 = m.Intermediate(0.1*(mldc**0.15)/(AD**0.3),name='kar1')
kAR = m.Intermediate(m.max2(0.25,kar1),name='kar')

#DESIGN PRESSURE, BENDING MOMENT AND SHEAR FORCE ISO 12215 part 5
# planing mode side pressure PBP
z=(d-t)
h=(z/2)
kz = ((z-h)/z)
psp = m.Intermediate((((0.35*lw1+14.6)+kz*
                    (0.25*(2.24*(mldc**0.33)+20)
                    -(0.35*lw1+14.6))))*kAR*kDC,name='psp')

# Shear Strength aspect ratio factor kSHC for l/b<2
# the formulation bellow, for l/b>4 kSHC = 0.5
kshc1 = m.Intermediate(0.035+0.394*(lls/bls)

```

```

        -0.09*((l1s/b1s)**2),name='kshc1')
kSHC = 0.5 # m.Intermediate(m.if3(2-(l1s/b1s),0.5,kshc1),name='kshc')

# Panel Aspect Ratio factor k2 for l/b>2 k= 0.5
a1s= m.Intermediate(0.271*((l1s/b1s)**2)
                    +0.910*(l1s/b1s)-    .554,name='a1s')
a2s= m.Intermediate(((l1s/b1s)**2)-0.313*(l1s/b1s)+1.351,name='a2s')
k2s = 2 #m.Intermediate(m.if3(2-(l1s/b1s),0.5,a1s/a2s),name='k2')

#[N/mm] shear force in middle of "b1s"
fdps= m.Intermediate(1*kSHC*psp*b1s*(1E-3),name = 'fdps')
#[N mm/mm] Bending moment in "b1s" direction
mdps= m.Intermediate(83.33*2*k2s*psp*(b1s**2)*(1E-6),name = 'mdps')

return psp, fdps, mdps, l1s, b1s

# THIS FUNCTION PARAMETERIZE THE CONSTRAINST FOR BOTTOM PLATE AND STIFFENERS
def side_optimize(nt,twl,twc,nls,sls,hs,bs,
                 psl,fdsl,mdsl,mdps,luls,b_s,bside,psp,st):
    ...
    INPUT
    nmat : Number of each MAT layer (Vector with Design Variables)
    nwr  : Number of each WR layer (Vector with Design Variables)
    nmat =[n1 : Number of MAT 300 layers,
           n2 : Number of MAT 350 layers,
           n3 : Number of MAT 450 layers,
           n4 : Number of MAT 600 layers]
    nwr  =[n5 : Number of WR 600 layers,
           n6 : Number of WR 800 layers]
    nsmat=[ns1: Number of MAT 300 layers,
           ns2: Number of MAT 350 layers,
           ns3: Number of MAT 450 layers,
           ns4: Number of MAT 600 layers]
    nswr =[ns5: Number of WR 600 layers,
           ns6: Number of WR 800 layers]
    OUTPUT
    ...
    # STIMATE THE WEIGHT PER AREA, THICKNESS ADN YOUNG'S MODULUS
    # OF BOTTOM PLATE
    Amts = m.Intermediate(ns1*Tmat[0]+ns2*Tmat[1]
                          +ns3*Tmat[2]+ns4*Tmat[3],name='amts')
    Awrs= m.Intermediate(ns5*Twr[0]+ns6*Twr[1],name='awrs')
    Cmts = m.Intermediate(ns1*Wmat[0]+ns2*Wmat[1]
                          +ns3*Wmat[2]+ns4*Wmat[3],name='cmts')
    Cwrs = m.Intermediate(ns5*Wwr[0]+ns6*Wwr[1],name='cwrs')
    Es   = m.Intermediate((6400** (4+2*(ns1+ns2+ns3+ns4)) *
                          (13240** (2*(ns5+ns6))))
                          ** (1/(2*(ns1+ns2+ns3+ns4+ns5+ns6)+4)),name='eps')
    S_T  = m.Intermediate((4*min(Tmat)+2*(Amts+Awrs)),name='s_t')

    #STIFFENER OPTIMIZATION
    # SigUT      # ultimates stresses [N/mm2]
    # Tao_MAT    [N/mm2]
    # Tao_Core   [N/mm2]
    # [cm3] Req. SM in flange of Long. Stiff
    SM_Req_Lgs  = m.Intermediate((mdsl)/(SigUT),name='smrsl')
    # [cm2] Req. shear area on webs of Long. Stiff
    Awebs_Req_Lgs = m.Intermediate((fdsl)/(100*Tao_MAT),name='awrsl')
    # [cm2] Req. shear area on webs of Long. Stiff

```



```

Acore_Req_Lgs = m.Intermediate((fdsl)/(100*Tao_Core),name='acrsl')

#ST_LS : thickness of side longitudinal stiff.
#L_WS : weight of side longitudinal stiff.
ST_LS = m.Intermediate(N_sl1*Tmat[0]+N_sl2*Tmat[1]
+N_sl3*Tmat[2]+N_sl4*Tmat[3],name='st_ls')
L_WS = m.Intermediate(N_sl1*Wmat[0]+N_sl2*Wmat[1]
+N_sl3*Wmat[2]+N_sl4*Wmat[3],name='l_ws')

# LONGITUDINALS STIFFENERS CALCULATIONS
asd= m.Intermediate(10+(N_sl1+N_sl2+N_sl3+N_sl4)*15,name='asd')
wsp= m.Intermediate(20*S_T+bs,name='wsp')
EAS= m.Intermediate((Es*wsp*S_T
+(Ec*bs*hs)
+(Er*(asd*ST_LS+2*hs*ST_LS+(bs+2*ST_LS)*ST_LS)))
,name='EALS')

EAZS= m.Intermediate(((Es*wsp*S_T*(0.5*S_T))
+(Ec*bs*hs*(0.5*hs+S_T))
+(Er*((asd*ST_LS*(S_T+0.5*ST_LS))
+2*hs*ST_LS*(S_T+0.5*hs)
+ST_LS*(bs+2*ST_LS)*(S_T+hs+0.5*ST_LS))))
,name='EAZLS')
NAS = m.Intermediate(EAZS/EAS,name='NALS')
Zcrs = m.Intermediate(S_T+hs+0.5*ST_LS-NAS,name='Zcrs')
EAZ2S= m.Intermediate(((Es*wsp*S_T*0.25*(S_T)**2)
+(Ec*(bs*(NAS-S_T)*0.25*(NAS+S_T)**2)
+bs*(hs-NAS+S_T)*(0.25*(NAS+hs+S_T)**2)))
+(Er*((2*ST_LS*(NAS-S_T)*0.25*(NAS+S_T)**2)
+(2*ST_LS*(hs-NAS+S_T)*0.25*(NAS+hs+S_T)**2)))
+(Er*((asd*ST_LS*(S_T+0.5*ST_LS)**2)
+(bs+2*ST_LS)*ST_LS*((S_T+hs+0.5*ST_LS)**2))))
,name='EAZ2LS')
Esh3 = m.Intermediate((1/12)*(Es*wsp*(S_T**3))
+(Ec*bs*((hs-NAS+S_T)**3)+((NAS-S_T)**3))
+(Er*2*ST_LS*((hs-NAS+S_T)**3)+
((NAS-S_T)**3))
+(Er*(asd*(ST_LS**3)+
(bs+2*ST_LS)*(ST_LS**3))))
,name='Esh3L')
EInas= m.Intermediate(EAZ2S+Esh3-(NAS**2)*EAS,name='EInaLs')
Qcs= m.Intermediate((Ec*0.5*bs*(hs+S_T-NAS)**2)
+(Er*(Zcrs*ST_LS*(bs+2*ST_LS)+
ST_LS*(hs+S_T-NAS)**2)))
,name='QcLs')
Qws= m.Intermediate((Es*(NAS-0.5*S_T)*wsp*S_T)
+(Er*((NAS-S_T-0.5*ST_LS)*asd*ST_LS)))
,name='QwLs')

EImins= m.Intermediate((26*(1**1.5)*psp*st*((1000*bside)**3)*(1E-7)/0.05)
,name='EIminsL')

#Define longitudinal stiffeners constraints
# [cm3] actual section modulus for longitudinal stiff
SMLS= m.Intermediate((0.001*EInas/(Er*Zcrs)),name='smls')
C1LS= m.Intermediate((SMLS/SM_Req_Lgs),name='c1ls')
# [cm2] actual web shear area for long. stiffside
AWLS= m.Intermediate((0.01*EInas*2*ST_LS/Qws),name='awls')
C2LS= m.Intermediate((AWLS/Awebs_Req_Lgs),name='c2ls')
# [cm2] actual core shear area for long. stiff
ACLS= m.Intermediate((0.01*EInas*bs/Qcs),name='acls')
C3LS= m.Intermediate((ACLS/Acore_Req_Lgs),name='c3ls')
# realtion of stiffness for long stiff side
EILS= m.Intermediate((EInas/EImins),name='eils')

```

```

# realtion for controle the local buckling in webs (side)
C45S= m.Intermediate((hs/ST_LS),name='c45s')
# realtion for controle the local buckling in flange (side)
C46S= m.Intermediate((bs/ST_LS),name='c46s')

#TOTAL STIFFENERS WEIGHT
SL_weights = m.Intermediate(((L_WS*nls*luls/300)*((2*hs+bs+asd)/1000))
                             +(hs*bs*luls*nls*dcore/(1000**3)),
                             name='sl_weights')

ST_weights = m.Const(bside*nt*(tw_l.value[0]
                             +tw_c.value[0]),name='st_weights')

#SIDE PLATE OPTIMIZATIONS
# Intermediates to compute the constraints
# Sectional Inercia
I = m.Intermediate((((4*min(Tmat)+2*(Amts+Awrs))**3)/12),name='I')
# Distance to critical fiber
Zcri = m.Intermediate((Amts+Awrs)+1.5*min(Tmat),name='Zcrip')
# Acting tension stress
Sact = m.Intermediate((mdps/(Es*I/(Zcri*6400))),name='Sact')

Panel_Dim = Lp*bside # Total Panel dimension in squared meters
# Local panel dimension in squared meters
LPanel_Dim = m.Intermediate((sls-bs)*b_s/(1000**2),name='Lpanel')
# Total numbers of plies
B_plies =
m.Intermediate((2*(ns1+ns2+ns3+ns4+ns5+ns6)+4),name='bplies')

C1PS = m.Intermediate(ns1+ns2+ns3+ns4-ns5-ns6,name='c1ps')
C2PS = m.Const((0.0675+0.0135*v*0.0675*(mldc**0.33)),name='c2ps')
C3PS = m.Intermediate((4*min(Wmat)/0.3)+
                       2*((Cmts/0.3)+(Cwrs/0.48)),name='c3ps')

#Define the objective function (panel weight in Kg )
STIFF_WEIGHTS = m.Intermediate((SL_weights+ST_weights),name='sws')
PANEL_WEIGHTS = m.Intermediate(Panel_Dim*C3PS,name='pws')
TOTAL_WEIGHTS = PANEL_WEIGHTS+STIFF_WEIGHTS

return (TOTAL_WEIGHTS,PANEL_WEIGHTS,STIFF_WEIGHTS,Es
        ,C1LS,C2LS,C3LS,EILS,C45S,C46S,C1PS,C2PS,C3PS
        ,Sact,SL_weights,ST_weights,S_T,B_plies)

```

```

MAIN PROGRAM (SIDE)
#=====OPTIMIZATION OF BOTTOM PANELS=====
#=====INITIAL POINT FOR SIDE OPTIMIZATION=====
ns1, ns2, ns3, ns4, ns5, ns6 = 0,0,1,0,0,2
NLS = 3
N_sl1, N_sl2, N_sl3, N_sl4 = 0,0,0,1
hLs,bLs= 150,100
#=====STARTING VECTOR=====
Nms      = [ns1,ns2,ns3,ns4]
Nws      = [ns5,ns6]
SL_Mat   = [N_sl1, N_sl2, N_sl3, N_sl4]
DIS      = [hLs ,bLs]
#===== GEKKO FIRST FASE INTERACTIONS =====
ns= [ns1,ns2,ns3,ns4,ns5,ns6] # Initial point x0 =
[n1,n2,n3,n4,n5,n6,EIG,EIS]
NRS = [NR[0],NLS]           # Intial point R = [EIG,EIS,NT,NL]
print(74*"=")
print(74*"=")
print(28*"=","START OPTIMIZATION",28*"=")
start_time = time.time()
print(74*"=")
from gekko import GEKKO
m = GEKKO(remote=False) # Initialize gekko
m.options.IMODE = 3      # solution mode 2 or 3
m.options.SOLVER= 1      # (1=APOPT, 3=IPOPT) APOPT is an MINLP solver
# options1 solver settings with APOPT
m.solver_options = ['minlp_maximum_iterations 5000', \
                    # minlp iterations with integer solution
                    'minlp_max_iter_with_int_sol 26000', \
                    # treat minlp as nlp
                    'minlp_as_nlp 0', \
                    # nlp sub-problem max iterations
                    'nlp_maximum_iterations 1500', \
                    # 1 = depth first, 2 = breadth first , 3 , 4
                    'minlp_branch_method 3', \
                    #maximum value to be considered as an integer
                    'minlp_integer_max 2.0e9'\
                    # maximum deviation from whole number
                    'minlp_integer_tol 0.0001', \
                    # convergence tolerance
                    'minlp_gap_tol 0.0001'\
                    #objective convergence tolerance 1.0e-6
                    'constraint_convergence_tolerance 1.0e-6'\
                    #convergence tolerance for the constraints
                    'objective_convergence_tolerance 1.0e-6']
#===== DESIGN VARIABLES.=====
print(9*".." ,"Initial point for side optimization",9*"..")
print("Number of Longitudinal Stiffeners          [NT,NLS] : ",NRS)
print("")
print("Side Layers          [ns1,ns2,ns3,ns4,ns5,ns6] : ",ns)
print("")
print("Long. Stiffeners layers [bsln1,bsln2,bsln3,bsln4] : ",SL_Mat)
print("")
print("Stiffeners dimension          [hLs,bLs,hTs,bTs][mm] : "
      ,DIS,round(DIM[2],2),round(DIM[3],2))
print(37*"..")
#=====first fase design variables.=====
ns1 = m.Var(value=ns[0],lb=0,ub=50,integer=True,name='ns1')
ns2 = m.Var(value=ns[1],lb=0,ub=50,integer=True,name='ns2')
ns3 = m.Var(value=ns[2],lb=0,ub=50,integer=True,name='ns3')
ns4 = m.Var(value=ns[3],lb=0,ub=50,integer=True,name='ns4')

```

```

ns5 = m.Var(value=ns[4],lb=0,ub=50,integer=True,name='ns5')
ns6 = m.Var(value=ns[5],lb=0,ub=50,integer=True,name='ns6')

NLS = m.Var(value=NRS[1],lb=0,ub=2,integer=True,name='NLS')

#=====second fase design variables.=====
N_sl1 = m.Var(value=SL_Mat[0],lb=0,ub=50,integer=True,name='N_sl1')
N_sl2 = m.Var(value=SL_Mat[1],lb=0,ub=50,integer=True,name='N_sl2')
N_sl3 = m.Var(value=SL_Mat[2],lb=0,ub=50,integer=True,name='N_sl3')
N_sl4 = m.Var(value=SL_Mat[3],lb=0,ub=50,integer=True,name='N_sl4')

hLs = m.Var(value=DIS[0],lb=40,ub=400,integer=False,name='hls')
bLs = m.Var(value=DIS[1],lb=40,ub=400,integer=False,name='bls')

#=====GEKKO COMPUTATION.=====
#Stimulation of space and span stiffeners
sL_s, luL_s = span_space_stiff_s(NLS,bside)

#Stimulation of stiffeners forces
ps_l, fds_l, mds_l = stiff_force_s(luL_s,sL_s,pb_l.value[0],d,t)

#Stimulation of plate bending moment and shear force
ps_p, fds_p, mds_p, ls_l, bs_l = plate_force_s(bLs,sL_s,b_l.value[0],
                                              pb_p.value[0],d,t)

#Plate optimization.
(TOTAL_WEIGHTS,PANEL_WEIGHTS,STIFF_WEIGHTS,Es,
 C1LS,C2LS,C3LS,EILS,C45S,C46S,C1PS,C2PS,C3PS,Sacts,SL_W,ST_W,
 S_thick,S_plies) = side_optimize(NR[0],tw_l,tw_c,NLS
                                ,sL_s,hLs,bLs,ps_l,fds_l,mds_l,mds_p,
                                luL_s,b_l.value[0],bside,ps_p,sT.value[0])
#=====EQUATIONS.=====
# m.Equation((hLs/bLs)-1 >=0)
#=====first fase constraints.=====
#Same number of Mat layers and Wr layers into sub laminste
CON1s = m.Equation(C1PS == 0)

#Tension Stress acting less than design tension stress
CON2s = m.Equation((SigUT/Sacts) -1 >= 0)
c2s = m.Intermediate(SigUT/Sacts)

#Minimum weight for impact loads
CON3s = m.Equation((C3PS/C2PS) -1 >= 0)
c3s = m.Intermediate(C3PS/C2PS)

#=====second fase constraints.=====
# Tension Stress acting on top flange of longitudinal stiffener
# must be less than design tension stress for fiber
CON4s = m.Equation(C1LS -1 >= 0)
c4s = m.Intermediate(C1LS)

# Shear stress acting longitudinal stiffeners webs must be
# less than design shear stress for fiber material
CON5s = m.Equation(C2LS -1 >= 0)
c5s = m.Intermediate(C2LS)

# Shear stress acting longitudinal stiffeners core must be
# less than design shear stress for core material
CON6s = m.Equation(C3LS - 1 >= 0)
c6s = m.Intermediate(C3LS)

```

```

# Longitudinal stiffener deflection must be less
# than permissible deflection
CON7d = m.Equation(EILS - 1 >= 0 )
c7s = m.Intermediate(EILS)

# Local buckling for webs and flange of longitudinal stiffeners
CON8s = m.Equation(1 - (C45S/30) >= 0)
CON9s = m.Equation(1 - (C46S/21) >= 0)
c8s = m.Intermediate(C45S/30)
c9s = m.Intermediate(C46S/21)
#=====Objective function.=====
#Objective function for side
m.Obj(TOTAL_WEIGHTS) # Objective function, panel weight minimization
#=====SOLVER.=====
#Solve and print results
output = m.solve(disp=False) # Solver
#=====New Values.=====
panels = PANEL_WEIGHTS.value[0]
stiffs = STIFF_WEIGHTS.value[0]
ns = [ns1.value[0],ns2.value[0],ns3.value[0],ns4.value[0]
      ,ns5.value[0],ns6.value[0]]
NRS[1] = NLS.value[0]
SL_Mat = [N_sl1.value[0], N_sl2.value[0], N_sl3.value[0], N_sl4.value[0]]
DIS = [hLs.value[0] ,bLs.value[0]]
print(50*" -")
print(50*" -")
print(50*" *")
print(18*" *", "Eval. Const.",18*" *")
print("Const. Flex. Plate      >= 1 :",round(c2s.value[0],3))
print("Const. Num. Layer      == 0 :",round(C1PS.value[0],3))
print("Const. Impact Loads    >= 1 :",round(c3s.value[0],3))
print("Const. Tension Long     >= 1 :",round(c4s.value[0],3))
print("Const. Shear Webs Long  >= 1 :",round(c5s.value[0],3))
print("Const. Shear Core Long  >= 1 :",round(c6s.value[0],3))
print("Const. Stiffness Long   >= 1 :",round(c7s.value[0],3))
print("Const. Buck. Webs Long  <= 1 :",round(c8s.value[0],3))
print("Const. Buck. Core Long  <= 1 :",round(c9s.value[0],3))
print(50*" *")
print(50*" -")
#=====SHOW THE RESULTS.=====
print(50*" =")
print("CONGRATULATIONS YOU GET THE SOLUTION")
print(50*" =")
print("Number of Stiffeners          [NTS,NLS] : ",NRS)
print(" ")
print("Side Panel Layers    [ns1,ns2,ns3,ns4,ns5,ns6] : ",ns)
print("Total Side Plies          : ",[S_plies.value[0]])
print("Total Side thickness    [mm] : ",[S_thick.value[0]])
print(" ")
if NRS[1] > 0:
    print("Side Stiffeners dimensions    [hLs,bLs,hTs,bTs] [mm] : "
          , [round(DIS[0],2) ,round(DIS[1],2)
            ,round(hT.value[0],2) ,round(bT.value[0],2)])
    print(" ")
    print("Side longitudinal stiff layer [sln1,sln2,sln3,sln4] : ",SL_Mat)
    print(" ")
    print("Side transversal stiff layer  [stn1,stn2,stn3,stn4] : ",BT_Mat)
elif NRS[1] == 0 :
    print("Side Stiffeners dimensions    [hTs,bTs] [mm] : "
          , round(hT.value[0],2) ,round(bT.value[0],2))

```

```

        print(" ")
        print("Side transversal stiff layer [stn1,stn2,stn3,stn4] : ",BT_Mat)
print(" ")
print("Global Panel weight [Kg] : ",[round(panels,3)])
# print(" ")
print("Global Stiffeners weight [Kg] : ",[round(stiffs,3)])
print(" ")
print('Total Side Weight [Kg] : '
      + str([round(m.options.objfcnval,3)]))
# print(" ")
print("Time: --- %s seconds ---" % round((time.time() - start_time),4))

# m.open_folder()
print("Panels,Long_weight, Trans_weight[Kg]:
",panels,SL_W.value[0],ST_W.value)
print("sLs, luLs,ls_1,bs_1 [mm] : ",sL_s.value[0], luL_s.value,
      ls_1.value[0],bs_1.value[0])
print("PS_P, PS_L [kN/m2] : ",ps_p.value[0],ps_l.value[0])
print("Mds_P, Fds_P [N mm/mm] [N/mm] : ",mds_p.value[0],fds_p.value[0])
print("Mds_L, Fds_L, [Nm] [N] : ",mds_l.value[0],fds_l.value[0])
print("Sigma act [N/mm2] : ",Sacts.value[0])
print("")

# GRÁFICA DE FUNCIÓN OBJETIVO
# Leer datos
file_names=glob.glob(m._path+'\\*.out')
file=file_names[0]
data=pd.read_csv(file,header=1,skiprows=2,delimiter= '\s+',decimal='.')

#Quitar columnas de strings
df = pd.DataFrame(data=data)
df = df[df['Iter:'].str.contains('Iter:')]
indexNames = df[ df['I:'] == 'I:-11' ].index
df.drop(indexNames , inplace=True)
df.drop(df.columns[np.arange(0,15,2)], axis=1,inplace=True)

#Tabla final
df=df.T.reset_index().T
df.columns=['Iter','I','Tm','NLPi','Dpth','Lvs','Obj','Gap']
df.index=np.arange(0,len(df.index))
plt.figure(figsize=(10,5),facecolor='#ffffff')
plt.scatter(df['Iter'], df['Obj'])
df['Gap'] = df['Gap'].fillna('0.6')
df['Gap'][0]=0.6
plt.plot(df['Iter'].astype(float), df['Gap'].astype(float),
         c='r',lw=0.75,label=r'MC',ls='-.')
plt.suptitle('Gap ', fontsize=25)
plt.xlabel('Iteration', fontsize=18)
plt.ylabel('Gap', fontsize=16)
plt.grid(ls=':',c='#4f4f4f')
plt.legend(loc=1,facecolor='#ffffff',fontsize=6.1,framealpha=0.1)
plt.xticks(np.arange(df['Iter'].astype(float).min(),
                    df['Iter'].astype(float).max()+1, 50))
plt.yticks(np.arange(df['Gap'].astype(float).min(),
                    df['Gap'].astype(float).max()+1, 0.1))

plt.tight_layout()
plt.show()

```

APÉNDICE D

CÓDIGO PYTHON- OPTIMIZACIÓN DE LA QUILLA

```
#KEEL OPTIMIZATION
```

```
#KEEL FUNCTIONS
```

```
def keel_optimize(KMat,Tmat,Wmat,b_t,Ep,Ec,dkeel,mdl,fdl,ppb,sl,lp):  
    # Design variables  
    # hk : keel height (real)  
    # bk : keel width (real)  
    # n1k : layers of MAT 300  
    # n2k : layers of MAT 350  
    # n3k : layers of MAT 450  
    # n4k : layers of MAT 600  
    # KMat = [n1k,n2k,n3k,n4k]  
  
    # THICKNESS AND WEIGHT OF LAMINATED  
    KT = 0 # [mm] thickness of keel laminate  
    KW = 0 # [kg/m2] weight of keel laminate  
    for i in range(len(KMat)):  
        KT += KMat[i]*Tmat[i]  
        KW += KMat[i]*Wmat[i]  
    #laminate thickness  
    kthick = m.Intermediate(KT,name='kthick')  
  
    #KEEL OPTIMIZATION  
    # [cm3] Req. SM in flange of keel  
    SM_Req_Long = m.Intermediate((mdl)/(SigUT),name='smrl') #23831.27  
    # [cm2] Req. shear area on webs of keel  
    Awebs_Req_Long = m.Intermediate((fdl)/(100*Tao_MAT),name='awrl')  
    # [cm2] Req. shear area on webs of keel  
    Acore_Req_Long = m.Intermediate((fdl)/(100*Tao_Core),name='acrl')  
  
    # KEEL SECTIONAL MODULUS  
    a =m.Intermediate((25+(sum(KMat)-1)*15),name='ak')  
    wp =m.Intermediate(20*b_t+bk,name='wpk')  
    EA =m.Intermediate((Ep*(wp*b_t)  
        +(Ec*bk*hk)  
        +(Er*(a*KT+2*hk*KT+(bk+2*KT)*KT))),name='EAK')  
    EAZ =m.Intermediate(((Ep*wp*b_t*(0.5*b_t)  
        +(Ec*bk*hk*(0.5*hk+b_t))  
        +(Er*((a*KT*(b_t+0.5*KT))+2*hk*KT*(b_t+0.5*hk)  
        +KT*(bk+2*KT)*(b_t+hk+0.5*KT))))),name='EAZk')  
    NA =m.Intermediate(EAZ/EA,name='NAk')  
    Zcri =m.Intermediate(b_t+hk+0.5*KT-NA,name='Zcrik')  
    EAZ2 =m.Intermediate(((Ep*wp*b_t*0.25*(b_t)**2)  
        +(Ec*((bk*(NA-b_t)*0.25*(NA+b_t)**2)  
        +bk*(hk-NA+b_t)*(0.25*(NA+hk+b_t)**2)))  
        +(Er*((2*KT*(NA-b_t)*0.25*(NA+b_t)**2)  
        +(2*KT*(hk-NA+b_t)*0.25*(NA+hk+b_t)**2)))  
        +(Er*(a*KT*(b_t+0.5*KT)**2)  
        +(bk+2*KT)*KT*((b_t+hk+0.5*KT)**2))))),  
        name='EAZ2k')  
    Ebhk3 =m.Intermediate((1/12)*((Ep*wp*(b_t**3))  
        +(Ec*bk*((hk-NA+b_t)**3)+((NA-b_t)**3)))  
        +(Er*2*KT*((hk-NA+b_t)**3)+((NA-b_t)**3)))  
        +(Er*(a*(KT**3)+(bk+2*KT)*(KT**3))))  
        ,name='Ebhk3')
```

```

EInak =m.Intermediate(EAZ2+Ebhk3-(NA**2)*EA,name='EInakk')

SMkeel =m.Intermediate(EInak/(Zcri*Er),name='SMkeelk')

SMkmin =m.Intermediate(1.4*(10**-3)*130*LH*mLDC/SigFlex,name='SMmink')

Qc = m.Intermediate(((Ec*0.5*bk*(hk+b_t-NA)**2)
+ (Er*(Zcri*KT*(bk+2*KT)+
KT*(hk+b_t-NA)**2)))
,name='QcL')
Qw = m.Intermediate(((Ep*(NA-0.5*b_t)*wp*b_t)
+ (Er*((NA-b_t-0.5*KT)*a*KT))),name='QwL')

EImin = m.Intermediate((26*(1**1.5)*pbb*sl*((1000*lp)**3)*(1E-7)/0.05)
,name='EIminL')

#Define longitudinal stiffeners constraints
# [cm3] actual section modulus for long. stiff
SML = m.Intermediate(0.001*SMkeel,name='msk')
C1K = (SML/SM_Req_Long)
# [cm2] actual web shear area for long. stiff
AWL = m.Intermediate((0.01*EInak*2*KT/Qw),name='awk')
C2K = (AWL/Awebs_Req_Long)
# [cm2] actual core shear area for long. stiff
ACL = m.Intermediate((0.01*EInak*bk/Qc),name='ack')
C3K = (ACL/Acore_Req_Long)
# [cm2] actual core shear area for long. stiff
EIK = m.Intermediate((EInak/EImin),name='eik')
# realtion for controle the local buckling in webs
C45K = m.Intermediate((hk/KT),name='c45k')
# realtion for controle the local buckling in flange
C46K = m.Intermediate((bk/KT),name='c46k')

Keel_weight = (lp*((KW/0.3)*(2*hk+bk+(25+(sum(KMat)-1)*15))/1000)+
(hk*bk/(1000**2))*dkeel)
return (SMkeel, SMkmin, Keel_weight, C1K,SML,SM_Req_Long,SML,
C2K, C3K, EIK, C45K, C46K, kthick)

```



```

MAIN (KEEL)
#=====OPTIMIZATION OF BOTTOM PANELS=====
#=====INITIAL POINT FOR SIDE OPTIMIZATION=====
KMat = [0,0,2,5]
DIM = [150,150]
# KMat = [1,0,2,0]
# DIM = [100,100]
SigFlex = 0.8*152.18 # [N/mm2] Ultimate Flexural Stress
#=====GEKKO KEEL OPTIMIZATION=====
start_time = time.time()
m = GEKKO(remote=False) # Initialize gekko
m.options.IMODE = 2 # solution mode 2 or 3
m.options.SOLVER= 1 # (1=APOPT,3=IPOPT) APOPT is an MINLP solver
# optional solver settings with APOPT
m.solver_options = ['minlp_maximum_iterations 1000', \
# minlp iterations with integer solution
'minlp_max_iter_with_int_sol 1500', \
# treat minlp as nlp
'minlp_as_nlp 0', \
# nlp sub-problem max iterations
'nlp_maximum_iterations 1500', \
# 1 = depth first, 2 = breadth first , 3 , 4
'minlp_branch_method 3', \
#maximum value to be considered as an integer
'minlp_integer_max 2.0e9'\
# maximum deviation from whole number
'minlp_integer_tol 0.0001', \
# convergence tolerance
'minlp_gap_tol 0.0001'\
#objective_convergence_tolerance 1.0e-6
'constraint_convergence_tolerance 1.0e-6'\
#convergence tolerance for the constraints
'objective_convergence_tolerance 1.0e-6']
#=====define gekko variables.=====
n1k = m.Var(value=KMat[0],lb=0,ub=10,integer=True,name='N_bk1')
n2k = m.Var(value=KMat[1],lb=0,ub=10,integer=True,name='N_bk2')
n3k = m.Var(value=KMat[2],lb=0,ub=10,integer=True,name='N_bk3')
n4k = m.Var(value=KMat[3],lb=0,ub=10,integer=True,name='N_bk4')

hk = m.Var(value=DIM[0],lb=40,ub=400,integer=False,name='hk')
bk = m.Var(value=DIM[1],lb=40,ub=400,integer=False,name='bk')
KMat = [n1k,n2k,n3k,n4k]
DIM = [hk ,bk]
#=====gekko FASE1.=====
#Optimization of dimension and stacking sequence for long. and trans.
stiffeners
(SMkeel, SMkmin, Keel_weight, C1K, SML, SM_Req_Long,sml,
ak, wpk, EAK, EAZk, NAK, Zcrik, EAZ2k, Ebh3k, EInak, Qck, Qwk, EIMIN,
C2K, C3K, EIK, C45K, C46K, KT) = keel_optimize(KMat,Tmat,Wmat,B_T.value[0],
EP.value[0],Ec,dkeel
,md_l.value[0],fd_l.value[0],
pb_p.value[0],sL.value[0],lp)
#=====EQUATIONS.=====
m.Equation(hk/bk -1 >= 0)

EIangy = 2.09E+12 # [N/mm2] Rigidez Flexural angie
SMangy = 527 # [cm3] Modulo Seccional de la quilla an

# Analisis con teoria de vigas, tomada como empotradas lu = 4 m
EIreq2 = 2.633E+11 # [N/mm2] rigidez minima y/lu = 0.05

```

```

#Minimum Sectional Modulus for the Keel
#SM and EI actual for Ang
# CON1k = m.Equation(SMkeel/(1000*SMangy) - 1 >= 0)
c1k = m.Intermediate(SMkeel/(1000*SMangy))
# CON8k = m.Equation((EInak/EIangy) - 1 >= 0)
c8k = m.Intermediate(EInak/EIangy)

# CON9k = m.Equation((EInak/EIreq2) - 1 >= 0)
c9k = m.Intermediate(EInak/EIreq2)

#SM min for ISO part-6
CON1k = m.Equation(SMkeel/(1000*SMkmin) - 1 >= 0)
c1k = m.Intermediate(SMkeel/(1000*SMkmin))

# Tension Stress acting on top flange of longitudinal stiffener
# must be less than design tension stress for fiber
# CON2k = m.Equation(C1K -1 >= 0)
c2k = m.Intermediate(C1K)

# Shear stress acting longitudinal stiffeners webs must be
# less than design shear stress for fiber material
# CON3k = m.Equation(C2K -1 >= 0)
c3k = m.Intermediate(C2K)

# Shear stress acting longitudinal stiffeners core must be
# less than design shear stress for core material
# CON4k = m.Equation(C3K - 1 >= 0)
c4k = m.Intermediate(C3K)

# Longitudinal stiffener deflection must be less
# than permissible deflection
# CON5k = m.Equation(EIK - 1 >=0 )
c5k = m.Intermediate(EIK)

# Local buckling for webs and flange of longitudinals stiffeners
CON6k = m.Equation(1 - (C45K/30) >= 0)
CON7k = m.Equation(1 - (C46K/21) >= 0)
c6k = m.Intermediate(C45K/30)
c7k = m.Intermediate(C46K/21)

#Objective function for second fase (panel weight + stiffeners weight)
KeelObj =m.Obj(Keel_weight)

#=====SOLVER.=====
#Solve and print results
outerk = m.solve(disp=True) # Solver

#=====New Values.=====
npliesk = n1k.value[0]+n2k.value[0]+n3k.value[0]+n4k.value[0]
KMat = [n1k.value[0],n2k.value[0],n3k.value[0],n4k.value[0]]
DIM = [hk.value[0] ,bk.value[0]]
print(50*"")
print(18*"", "Eval. Const.",18*"")
print("Const. Min Sectional Modulus (ISO part6) >= 1
:",round(c1k.value[0],3))
print("Const. Min Sectional Modulus (ISO part5) >= 1
:",round(c2k.value[0],3))
print("Const. Shear Stress on Webs Keel >= 1
:",round(c3k.value[0],3))
print("Const. Shear Stress on Core Keel >= 1
:",round(c4k.value[0],3))

```

```

print("Const. Keel Stiffness                                     >= 1
      :",round(c5k.value[0],3))
print("Const. Buckling of Keel Webs                           <= 1
      :",round(c6k.value[0],3))
print("Const. Buckling of Keel Core                           <= 1
      :",round(c7k.value[0],3))
print(50*"")
#=====Values for next fase.=====
print(50*"")
print("CONGRATULATIONS YOU GET THE SOLUTION")
print(50*"")
print(" ")
print("Keel layout          [n1k,n2k,n3k,n4k] : ",KMat)
#   print(" ")
print("Total Plies          : ",npliesk)
print("Total keel layers thickness [mm] : ",KT.value)
print(" ")
print("Keel dimensions      hk,bk [mm] : ",DIM)
print(" ")
print('Total Keel Weight [Kg] : '
      + str([round(m.options.objfcnval,3)]))
#   print(" ")
print("Time: --- %s seconds ---" % round((time.time() - start_time),4))

print(" ")
print("SMKeel          : ",SMkeel.value[0]/(10**3))
print("SM req (part6)  : ",SMkmin.value[0])
print("SM req (part5)  : ",SM_Req_Long.value[0])
print("SM ratio part5/SMkeel :", (md_l.value[0]*1000/SMkeel.value[0]))
print(" ")
print("a          ",ak.value)
print("wp.value   ",wpk.value)
print("EA.value,  ",EAK.value,)
print("EAZ.value  ",EAZk.value)
print("NA.value   ",NAk.value)
print("Zcri.value",Zcrik.value)
print("EAZ2.value",EAZ2k.value)
print("Ebh3.value",Ebh3k.value)
print("EIna.value",EInak.value)
print("Qw.value   ",Qwk.value)
print("Qc.value   ",Qck.value)
print("EInak/EIangy Angy : ",c8k.value[0])
print("EInak/EIangy L/20 : ",c9k.value[0])

```

```

#GRAFICAR LA FUNCION OBJETIVO
# Leer datos

file_names=glob.glob(m._path+'\\*.out')
file=file_names[0]
data=pd.read_csv(file,header=1,skiprows=2,delimiter= '\s+',decimal='.')

#Quitar columnas de strings
df = pd.DataFrame(data=data)
df = df[df['Iter:'].str.contains('Iter:')]
indexNames = df[ df['I:'] == 'I:-11' ].index
df.drop(indexNames , inplace=True)
df.drop(df.columns[np.arange(0,15,2)], axis=1,inplace=True)

#Tabla final
df=df.T.reset_index().T
df.columns=['Iter','I','Tm','NLPi','Dpth','Lvs','Obj','Gap']
df.index=np.arange(0,len(df.index))

fig=plt.figure(figsize=(10,5),facecolor='#ffffff')
#plt.scatter(df['Iter'], df['Obj'])
plt.plot(df['Iter'].astype(float), df['Obj'].astype(float),
         c='b',lw=0.75,label=r'MC',ls='-.')
plt.suptitle('Objective Function ', fontsize=20)
plt.xlabel('Iteration', fontsize=16)
plt.ylabel('Weight [kg]', fontsize=14)
plt.yticks(np.arange(df['Obj'].astype(float).min(),
                    df['Obj'].astype(float).max(),10))
plt.xticks(np.arange(1, float(len(df['Iter']))+1,1))

#plt.grid(ls=':',c='#4f4f4f')
plt.legend(loc=1,facecolor='#ffffff',fontsize=6.1,framealpha=0.1)
plt.tight_layout()
plt.show()
# if x_labels:
#     plt.xticks(x_values, x_labels, rotation='vertical')
# if y_labels:
#     plt.yticks([-1, 0, 1], y_labels, rotation='horizontal')

plt.figure(figsize=(10,5),facecolor='#ffffff')
#plt.scatter(df['Iter'], df['Obj'])
df['Gap'] = df['Gap'].fillna('0.6')
df['Gap'][0]=0.6
plt.plot(df['Iter'].astype(float), df['Gap'].astype(float),
         c='r',lw=0.75,label=r'MC',ls='-.')
plt.suptitle('Gap ', fontsize=25)
plt.xlabel('Iteration', fontsize=18)
plt.ylabel('Gap', fontsize=16)
#plt.grid(ls=':',c='#4f4f4f')
plt.legend(loc=1,facecolor='#ffffff',fontsize=6.1,framealpha=0.1)
plt.xticks(np.arange(df['Iter'].astype(float).min(),
                    df['Iter'].astype(float).max()+1, 50))
plt.yticks(np.arange(df['Gap'].astype(float).min(),
                    df['Gap'].astype(float).max()+1, 0.1))

plt.tight_layout()
plt.show()

```

APÉNDICE E

SALIDA – OPTIMIZACIÓN DEL FONDO

```
=====
=====
===== START OPTIMIZATION =====
=====
..... Initial point for bottom optimization .....
Number of Stiffeners          [NT,NL] : [5, 4]

Bottom Layers                 [n1,n2,n3,n4,n5,n6] : [1, 0, 0, 0, 0, 0]

Long. Stiffeners layers      [bln1,bln2,bln3,bln4] : [5, 1, 2, 4]

Trans. Stiffeners layer     [btn1,btn2,btn3,btn4] : [2, 1, 2, 4]

Stiffeners dimension        [hL,bL,hT,bT] [mm] : [150, 50, 100, 50]
.....
-----
APMonitor, Version 0.9.2
APMonitor Optimization Suite
-----

----- APM Model Size -----
Each time step contains
  Objects      : 7
  Constants    : 6
  Variables    : 63
  Intermediates: 114
  Connections  : 21
  Equations    : 150
  Residuals    : 36

Number of state variables: 77
Number of total equations: - 49
Number of slack variables: - 18
-----

Degrees of freedom      : 10

-----
Model Parameter Estimation with APOPT Solver
-----
Iter:      1 I: 0 Tm: 0.05 NLPi: 37 Dpth: 0 Lvs: 3 Obj: 1.47
E+02 Gap:      NaN
Iter:      2 I: 0 Tm: 0.01 NLPi: 9 Dpth: 1 Lvs: 5 Obj: 1.54
E+02 Gap:      NaN
Iter:      3 I: 0 Tm: 0.01 NLPi: 7 Dpth: 1 Lvs: 7 Obj: 1.47
E+02 Gap:      NaN
Iter:      4 I: 0 Tm: 0.01 NLPi: 7 Dpth: 1 Lvs: 9 Obj: 1.47
E+02 Gap:      NaN
Iter:      5 I: 0 Tm: 0.01 NLPi: 9 Dpth: 2 Lvs: 10 Obj: 1.53
E+02 Gap:      NaN
Iter:      6 I: 0 Tm: 0.00 NLPi: 5 Dpth: 2 Lvs: 12 Obj: 1.47
E+02 Gap:      NaN
```



```

Iter:    65 I:  0 Tm:    0.01 NLPi:    9 Dpth:   11 Lvs:  116 Obj:  1.48
E+02 Gap:      NaN
Iter:    66 I:  0 Tm:    0.00 NLPi:    4 Dpth:   11 Lvs:  117 Obj:  1.48
E+02 Gap:      NaN
Iter:    67 I:  0 Tm:    0.00 NLPi:    4 Dpth:   12 Lvs:  119 Obj:  1.49
E+02 Gap:      NaN
Iter:    68 I:  0 Tm:    0.01 NLPi:   11 Dpth:   12 Lvs:  121 Obj:  1.43
E+02 Gap:      NaN
--Integer Solution:  1.46E+02 Lowest Leaf:  1.43E+02 Gap:  1.84E-02
Iter:    69 I:  0 Tm:    0.01 NLPi:    9 Dpth:   13 Lvs:  120 Obj:  1.46
E+02 Gap:  1.84E-02
Iter:    70 I:  0 Tm:    0.01 NLPi:    7 Dpth:   13 Lvs:  122 Obj:  1.43
E+02 Gap:  1.84E-02
--Integer Solution:  1.46E+02 Lowest Leaf:  1.43E+02 Gap:  1.84E-02
Iter:    71 I:  0 Tm:    0.01 NLPi:    8 Dpth:   14 Lvs:  121 Obj:  1.46
E+02 Gap:  1.84E-02
Iter:    72 I:  0 Tm:    0.01 NLPi:    7 Dpth:   14 Lvs:  123 Obj:  1.43
E+02 Gap:  1.84E-02
Iter:    73 I:  0 Tm:    0.01 NLPi:    7 Dpth:   14 Lvs:  125 Obj:  1.43
E+02 Gap:  1.84E-02
Iter:    74 I:  0 Tm:    0.00 NLPi:    4 Dpth:   15 Lvs:  124 Obj:  1.46
E+02 Gap:  1.84E-02
Iter:    75 I:  0 Tm:    0.01 NLPi:    7 Dpth:   15 Lvs:  126 Obj:  1.43
E+02 Gap:  1.84E-02
Iter:    76 I:  0 Tm:    0.00 NLPi:    4 Dpth:   16 Lvs:  125 Obj:  1.46
E+02 Gap:  1.84E-02
Iter:    77 I:  0 Tm:    0.02 NLPi:   16 Dpth:   16 Lvs:  126 Obj:  1.19
E+02 Gap:  1.84E-02
Iter:    78 I:  0 Tm:    0.00 NLPi:    4 Dpth:   17 Lvs:  127 Obj:  1.20
E+02 Gap:  1.84E-02
Iter:    79 I:  0 Tm:    0.01 NLPi:    6 Dpth:   17 Lvs:  129 Obj:  1.23
E+02 Gap:  1.84E-02
Iter:    80 I:  0 Tm:    0.00 NLPi:    3 Dpth:   18 Lvs:  131 Obj:  1.20
E+02 Gap:  1.84E-02
Iter:    81 I:  0 Tm:    0.00 NLPi:    4 Dpth:   19 Lvs:  133 Obj:  1.17
E+02 Gap:  1.84E-02
Iter:    82 I:  0 Tm:    0.00 NLPi:    5 Dpth:   20 Lvs:  135 Obj:  1.19
E+02 Gap:  1.84E-02
Iter:    83 I:  0 Tm:    0.00 NLPi:    4 Dpth:   20 Lvs:  137 Obj:  1.17
E+02 Gap:  1.84E-02
Iter:    84 I:  0 Tm:    0.01 NLPi:    4 Dpth:   20 Lvs:  139 Obj:  1.17
E+02 Gap:  1.84E-02
Iter:    85 I:  0 Tm:    0.00 NLPi:    5 Dpth:   21 Lvs:  140 Obj:  1.19
E+02 Gap:  1.84E-02
Iter:    86 I:  0 Tm:    0.01 NLPi:    5 Dpth:   21 Lvs:  142 Obj:  1.19
E+02 Gap:  1.84E-02
Iter:    87 I: -9 Tm:    4.95 NLPi: 1501 Dpth:   21 Lvs:  141 Obj:  1.17
E+02 Gap:  1.84E-02
Iter:    88 I:  0 Tm:    0.00 NLPi:    4 Dpth:   21 Lvs:  142 Obj:  1.19
E+02 Gap:  1.84E-02
Iter:    89 I:  0 Tm:    0.00 NLPi:    4 Dpth:   21 Lvs:  143 Obj:  1.19
E+02 Gap:  1.84E-02
--Integer Solution:  1.46E+02 Lowest Leaf:  1.19E+02 Gap:  2.01E-01
Iter:    90 I:  0 Tm:    0.01 NLPi:   14 Dpth:   21 Lvs:  142 Obj:  1.02
E+08 Gap:  2.01E-01
--Integer Solution:  1.21E+02 Lowest Leaf:  1.19E+02 Gap:  1.35E-02
Iter:    91 I:  0 Tm:    0.00 NLPi:    3 Dpth:   22 Lvs:  141 Obj:  1.21
E+02 Gap:  1.35E-02

```



```

--Integer Solution: 1.21E+02 Lowest Leaf: 1.19E+02 Gap: 1.35E-02
Iter: 92 I: 0 Tm: 0.00 NLPi: 3 Dpth: 22 Lvs: 140 Obj: 1.21
E+02 Gap: 1.35E-02
Iter: 93 I: 0 Tm: 0.00 NLPi: 4 Dpth: 22 Lvs: 139 Obj: 1.33
E+02 Gap: 1.35E-02
--Integer Solution: 1.21E+02 Lowest Leaf: 1.19E+02 Gap: 1.35E-02
Iter: 94 I: 0 Tm: 0.00 NLPi: 3 Dpth: 22 Lvs: 138 Obj: 1.21
E+02 Gap: 1.35E-02
Iter: 95 I: 0 Tm: 0.01 NLPi: 4 Dpth: 22 Lvs: 137 Obj: 1.23
E+02 Gap: 1.35E-02
--Integer Solution: 1.21E+02 Lowest Leaf: 1.19E+02 Gap: 1.35E-02
Iter: 96 I: 0 Tm: 0.00 NLPi: 3 Dpth: 21 Lvs: 136 Obj: 1.21
E+02 Gap: 1.35E-02
--Integer Solution: 1.21E+02 Lowest Leaf: 1.19E+02 Gap: 1.35E-02
Iter: 97 I: 0 Tm: 0.00 NLPi: 3 Dpth: 21 Lvs: 135 Obj: 1.21
E+02 Gap: 1.35E-02
Iter: 98 I: 0 Tm: 0.00 NLPi: 5 Dpth: 21 Lvs: 134 Obj: 1.23
E+02 Gap: 1.35E-02
--Integer Solution: 1.21E+02 Lowest Leaf: 1.19E+02 Gap: 9.48E-03
Iter: 99 I: 0 Tm: 0.00 NLPi: 3 Dpth: 22 Lvs: 134 Obj: 1.21
E+02 Gap: 9.48E-03
Successful solution

```

```

-----
Solver          : APOPT (v1.0)
Solution time   : 5.6563 sec
Objective       : 120.55279384422414
Successful solution
-----

```

```

-----
*****
***** Eval. Const. *****
Const. Flex. Plate    >= 1 : 1.056
Const. Num. Layer     == 0 : 0.0
Const. Impact Loads  >= 1 : 24.767
Const. Tension Trans  >= 1 : 1.17
Const. Shear Webs Trans >= 1 : 1.024
Const. Shear Core Trans >= 1 : 1.036
Const. Stiffness Trans >= 1 : 9.178
Const. Buck. Webs Trans <= 1 : 0.496
Const. Buck. Core Trans <= 1 : 0.494
*****

```

```

-----
=====
CONGRATULATIONS YOU GET THE SOLUTION
=====

```

```

Number of Stiffeners          [NT,NL] : [9.0, 0.0]
Stiffeners dimensions        [hT,bT] [mm] : [57.41, 40.0]

Bottom Layers                 [n1,n2,n3,n4,n5,n6] : [2.0, 0.0, 0.0, 0.0, 2.0
, 0.0]
Total Plies                   : [12.0]
Total Thickness                [mm] : [8.712]

```

Trans. Stiffeners layer [btn1,btn2,btn3,btn4] : [0.0, 0.0, 1.0, 2.0]

Global Panel weight [Kg] : [78.0]

Global Stiffeners weight [Kg] : [42.553]

Total Weight [Kg] : [120.553]

Time: --- 6.0783 seconds ---

APÉNDICE F

SALIDA – OPTIMIZACIÓN DEL COSTADO

```
=====
=====
===== START OPTIMIZATION =====
==
=====
..... Initial point for side optimization .....
Number of Longitudinal Stiffeners      [NT,NLS] : [9.0, 3]

Side Layers          [ns1,ns2,ns3,ns4,ns5,ns6] : [0, 0, 1, 0, 0, 2]

Long. Stiffeners layers [bsln1,bsln2,bsln3,bsln4] : [0, 0, 0, 1]

Stiffeners dimension   [hLs,bLs,hTs,bTs] [mm] : [150, 100] 57.41 40.
0
.....
-----
*****
***** Eval. Const. *****
Const. Flex. Plate      >= 1 : 1.818
Const. Num. Layer       == 0 : 0.0
Const. Impact Loads    >= 1 : 24.767
Const. Tension Long    >= 1 : 1.0
Const. Shear Webs Long >= 1 : 2.155
Const. Shear Core Long >= 1 : 2.177
Const. Stiffness Long  >= 1 : 163.269
Const. Buck. Webs Long <= 1 : 0.528
Const. Buck. Core Long <= 1 : 0.571
*****
-----
=====
CONGRATULATIONS YOU GET THE SOLUTION
=====
Number of Stiffeners      [NTS,NLS] : [9.0, 0.0]

Side Panel Layers        [ns1,ns2,ns3,ns4,ns5,ns6] : [2.0, 0.0, 0.0, 0.0, 2.0,
0.0]
Total Side Plies         : [12.0]
Total Side thickness     [mm] : [8.7120098039]

Side Stiffeners dimensions [hTs,bTs] [mm] : 57.41 40.0

Side transversal stiff layer [stn1,stn2,stn3,stn4] : [0.0, 0.0, 1.0, 2.0
]

Global Panel weight      [Kg] : [62.4]
Global Stiffeners weight [Kg] : [34.042]

Total Side Weight        [Kg] : [96.442]
Time: --- 0.3002 seconds ---
```

APÉNDICE G

SALIDA – OPTIMIZACIÓN DE LA QUILLA

APMonitor, Version 0.9.2
APMonitor Optimization Suite

----- APM Model Size -----

Each time step contains

Objects : 0
Constants : 0
Variables : 10
Intermediates: 34
Connections : 0
Equations : 39
Residuals : 5

Number of state variables: 10
Number of total equations: - 4
Number of slack variables: - 4

Degrees of freedom : 2

Model Parameter Estimation with APOPT Solver

Iter: 1 I: 0 Tm: 0.01 NLPi: 13 Dpth: 0 Lvs: 2 Obj: 1.37
E+01 Gap: NaN
--Integer Solution: 1.41E+01 Lowest Leaf: 1.37E+01 Gap: 2.73E-02
Iter: 2 I: 0 Tm: 0.00 NLPi: 4 Dpth: 1 Lvs: 1 Obj: 1.41
E+01 Gap: 2.73E-02
Iter: 3 I: 0 Tm: 0.00 NLPi: 8 Dpth: 1 Lvs: 2 Obj: 1.38
E+01 Gap: 2.73E-02
Iter: 4 I: 0 Tm: 0.00 NLPi: 6 Dpth: 2 Lvs: 3 Obj: 1.38
E+01 Gap: 2.73E-02
Iter: 5 I: 0 Tm: 0.01 NLPi: 5 Dpth: 2 Lvs: 4 Obj: 1.38
E+01 Gap: 2.73E-02
Iter: 6 I: 0 Tm: 0.00 NLPi: 7 Dpth: 3 Lvs: 5 Obj: 1.38
E+01 Gap: 2.73E-02
Iter: 7 I: 0 Tm: 0.00 NLPi: 4 Dpth: 3 Lvs: 6 Obj: 1.38
E+01 Gap: 2.73E-02
Iter: 8 I: 0 Tm: 0.00 NLPi: 7 Dpth: 4 Lvs: 7 Obj: 1.38
E+01 Gap: 2.73E-02
--Integer Solution: 1.39E+01 Lowest Leaf: 1.38E+01 Gap: 9.00E-03
Iter: 9 I: 0 Tm: 0.00 NLPi: 4 Dpth: 4 Lvs: 7 Obj: 1.39
E+01 Gap: 9.00E-03
Successful solution

Solver : APOPT (v1.0)
Solution time : 0.0489 sec
Objective : 13.925459454536181
Successful solution

***** Eval. Const. *****
Const. Min Sectional Modulus (ISO part6) >= 1 : 1.0
Const. Min Sectional Modulus (ISO part5) >= 1 : 0.113
Const. Shear Stress on Webs Keel >= 1 : 0.157
Const. Shear Stress on Core Keel >= 1 : 0.229
Const. Keel Stiffness >= 1 : 0.182
Const. Buckling of Keel Webs <= 1 : 1.0
Const. Buckling of Keel Core <= 1 : 0.812

=====
CONGRATULATIONS YOU GET THE SOLUTION
=====

Keel layout [n1k,n2k,n3k,n4k] : [0.0, 0.0, 1.0, 1.0]
Total Plies : 2.0
Total keel layers thickness [mm] : [2.4534313725]

Keel dimensions hk,bk [mm] : [73.602941176, 41.847378951]

Total Keel Weight [Kg] : [13.925]
Time: --- 0.2575 seconds ---