



Pontificia Universidad
JAVERIANA
Cali

**Optimización del Proceso de Corte Bidimensional Tipo Guillotina en una Empresa
Local de Servicios Metalmecánicos Mediante Metaheurística**

Programa de Maestría en Ingeniería

Presentado por:

Angela María Arango Ruiz

Dirigido Por:

Daniel Morillo Torres, Ph.D.

Pontificia Universidad Javeriana Cali
Facultad de Ingeniería y Ciencias
Diciembre 2025

TABLA DE CONTENIDO

INTRODUCCIÓN.....	1
2 DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN.....	6
2.1 Pregunta de investigación.....	9
2.2 Alcance del trabajo de grado	9
3 OBJETIVOS DEL PROYECTO	11
3.1 Objetivo general	11
3.2 Objetivos específicos	11
3.3 Resultados esperados	11
4 JUSTIFICACIÓN DEL TRABAJO DE GRADO	13
5 MARCO DE REFERENCIA.....	14
5.1 Marco conceptual.....	14
5.1.1. Conceptos Generales del Corte Bidimensional	14
5.1.2 Métodos Existentes para Resolver el Problema de Corte Bidimensional y Limitaciones Actuales	20
5.2 Revisión de literatura	28
6 METODOLOGÍA DE LA INVESTIGACIÓN	34
6.1 Formulación del Modelo Matemático Exacto	34
6.2 Diseño de la Solución Aproximada (Heurística)	40
7. RESULTADOS.....	51
7.1 Detalle de la implementación	51
7.2 Resultados Obtenidos	51
8 CONCLUSIONES.....	62
REFERENCIAS BIBLIOGRÁFICAS	64

INTRODUCCIÓN

Los problemas de corte y empaque constituyen una de las áreas clásicas de la optimización combinatoria y surgen en contextos donde se busca utilizar de manera eficiente recursos limitados para satisfacer una demanda específica. En términos generales, estos problemas consisten en disponer un conjunto de objetos (ítems) dentro de contenedores o materiales de capacidad finita, con el objetivo de minimizar desperdicios, reducir costos o aprovechar al máximo el espacio disponible. Su aplicación se extiende a múltiples industrias: desde el corte de rollos de papel, vidrio o metal en la manufactura, hasta la planificación de cargas en transporte y logística. Estos problemas son aún un gran desafío para empresas asociadas al área productiva, y su interés está vigente en la comunidad académica debido a su robustez del tipo *NP-Hard* (Dyckhoff 1990).

Particularmente, el problema de corte bidimensional relaciona características físicas y operativas; determinantes en su definición, modelado y resolución. Estas consisten principalmente en: la geometría, la ortogonalidad y el tipo de corte, el número de etapas, restricciones productivas, entre otras. Por su parte las geometrías (rectangulares, circulares e irregulares) definen problemas de una, dos y tres dimensiones. La ortogonalidad, determina la coincidencia paralela de los bordes de los ítems más pequeños respecto los bordes de la gran pieza (Oliveira et al. 2016). Permitiendo un giro de los ítems a 90° de manera exclusiva, es decir, desde una posición horizontal a una vertical, o viceversa.

De forma general, el problema de corte bidimensional, establece el corte de una gran pieza rectangular de ancho W y altura finita H , dentro de la cual se acomoda n ítems rectangulares con ancho w_i y altura h_i ($i = 1, \dots, n$), con el objetivo de minimizar el número de piezas grandes usadas, acomodando la mayor cantidad de ítems pequeños, de forma que se genere el mínimo desperdicio posible (Cao y Kotov 2011) (Mellouli y Dammak 2008).

Por otra parte, el corte guillotina consiste en un corte ortogonal de lado a lado de una pieza, generando un ítem o un conjunto de ítems u otros espacios de corte, a partir de los bordes de la pieza o de las particiones que se están generando por cortes previos (Toro, Garcés, y Ruiz 2008) (Oliveira et al. 2016). Puede determinarse en función del número de etapas de corte, a fin de cumplir con la demanda de los ítems solicitados. Se encuentran cortes de una, dos, tres y hasta n etapas, a partir de una serie de cortes con orientación inicial (vertical u horizontal) en cada etapa o incluso con rotación ortogonal, se continua la sucesión de cortes, hasta completar el número de etapas de corte. El objetivo del corte por etapas es dividir secciones grandes en pequeños ítems rectangulares, hasta separar el ítem del residuo (Imahori et al. 2005) (Y. P. Cui et al. 2015) (Iori et al. 2021).

La implementación práctica, propuesta por (Fleszar 2013), operan sobre un conjunto de árboles enraizados, donde cada árbol representa un patrón de corte guillotina en una lámina; que codifica, de forma jerárquica la secuencia de cortes y la disposición final de los ítems y de los rectángulos residuales. En términos de la estructura del árbol, se integran los siguientes conceptos:

- **Nodo raíz:** Representa la lámina o “gran pieza” inicial que será subdividida mediante cortes guillotina.
- **Nodos internos (nodos estructura):** Cada nodo interno simboliza una operación de corte guillotina aplicada al rectángulo asociado al nodo. Estos nodos se etiquetan según su orientación de corte: V (vertical) o H (horizontal), indicando la dirección en que se divide el rectángulo en dos rectángulos (los hijos del nodo). La orientación define la

interpretación geométrica de los hijos: el primero corresponde a la subregión izquierda o superior, y el segundo, a la derecha o inferior, dependiendo del tipo de corte.

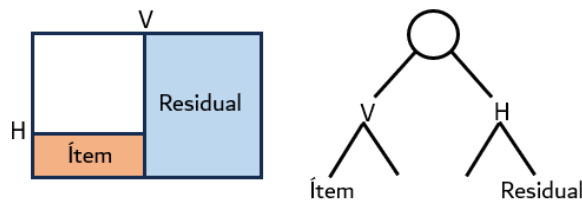
- Nodos hoja: Las hojas son rectángulos terminales y pueden ser de dos tipos:
 - Ítem (ítem): contiene una pieza demandada, con dimensiones y orientación (rotación) específicas.
 - Residual (leftover/waste): corresponde a una zona sobrante no asignada a ningún ítem, que puede considerarse desperdicio o sobrante reutilizable.

Las hojas garantizan que la partición final no presente solapamientos y que cada punto de la lámina pertenezca exactamente un rectángulo hoja (ítem o residual).

Cada nodo interno representa un corte que divide el rectángulo del nodo en dos rectángulos, asociados a sus nodos hijos. Extraer un ítem equivale a localizar una hoja de tipo ítem cuya posición y orientación satisfacen las dimensiones requeridas. Esta representación es canónica para patrones guillotina, ya que cualquier secuencia recursiva de cortes puede codificarse como una estructura binaria jerárquica.

La figura 1. ilustra la correspondencia entre la disposición geométrica del patrón y su representación mediante un árbol enraizado. En ella, los nodos internos (H o V) indican la orientación de los cortes, mientras que las hojas identifican los ítems insertados y las áreas residuales resultantes.

Figura 1. Representación de un Patrón Guillotina Mediante un Árbol Enraizado.



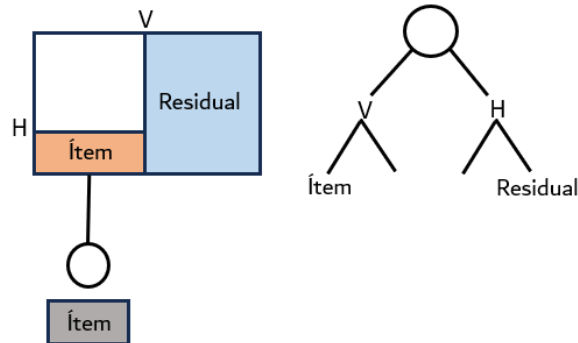
Adaptado de (Fleszar 2013)

A la izquierda se muestra la lámina original y los cortes aplicados (H: corte horizontal, V: corte vertical). A la derecha, el árbol binario asociado, donde los nodos internos representan cortes y las hojas corresponden a ítems o áreas residuales.

Por lo tanto, la evolución de la disposición espacial y del árbol binario asociado durante la inserción secuencial de ítems, se da en tres etapas: Etapa 1: lámina inicial abierta (nodo raíz). Etapa 2: inserción del primer ítem; aparece un nodo interno que indica la orientación del corte y las hojas correspondientes a ítem y residual. Etapa 3: inserción de un segundo ítem y actualización del árbol (creación de subárboles/residuos). Los nodos internos están etiquetados con la orientación del corte (V/H).

La Figura 2 muestra un ejemplo ilustrativo de la evolución de un patrón guillotina y su representación mediante un árbol enraizado. En cada etapa se inserta un ítem aplicando heurísticas constructivas *FFI* (*First Fit Insertion*), *BFI* (*Best Fit Insertion*) y *CFI* (*Critical Fit Insertion*), y se observa cómo el árbol se actualiza: aparecen nodos internos que registran la orientación del corte (V/H) y hojas que representan ítems o rectángulos residuales.

Figura 2. Evolución de un Patrón Guillotina y su Codificación Mediante un Árbol Enraizado (paso a paso).



Elaboración propia con base en (Fleszar 2013).

Por lo tanto, la evolución de la disposición espacial y del árbol binario asociado durante la inserción secuencial de ítems, se da en tres etapas: Etapa 1: lámina inicial abierta (nodo raíz). Etapa 2: inserción del primer ítem; aparece un nodo interno que indica la orientación del corte y las hojas correspondientes a ítem y residual. Etapa 3: inserción de un segundo ítem y actualización del árbol (creación de subárboles/residuos). Los nodos internos están etiquetados con la orientación del corte (V/H).

En la implementación práctica, las heurísticas propuestas por (Fleszar 2013), operan sobre un conjunto de árboles enraizados, donde cada árbol representa un patrón de corte en una lámina ya abierta. El objetivo es insertar nuevos ítems en alguna hoja residual disponible, siguiendo distintas estrategias, según las reglas:

- FFI (First Fit Insertion) o inserción de primer ajuste: Recorre las hojas residuales en un orden predefinido y coloca el ítem en la primera posición factible (considerando rotaciones permitidas).
- BFI (Best Fit Insertion), inserción de mejor ajuste: Evalúa todas las hojas candidatas y selecciona aquella que minimiza el desperdicio o genera el mejor ajuste según una métrica de eficiencia.
- CFI (Critical Fit Insertion) o inserción de ajuste crítico: Prioriza hojas o posiciones críticas, es decir, aquellas que permiten reducir la apertura de nuevas láminas o mejorar el aprovechamiento global del patrón.

Este tipo de representación no solo describe con precisión la secuencia operativa de los cortes guillotina, también constituye la base para controlar las restricciones asociadas al problema. En particular, las restricciones de cantidad o frecuencia de los ítems tienen un efecto directo sobre la estructura de los árboles y las estrategias de inserción, ya que determinan cuántos ítems de cada tipo pueden ser colocados en las láminas disponibles.

Por lo tanto, las restricciones desempeñan un papel determinante en la estructura y las técnicas de solución del problema de corte guillotina. En la literatura, el término *problema restringido* no hace referencia al hecho general de que existan restricciones en el modelo, sino a una condición específica relacionada con la cantidad de ítems que pueden obtenerse de la lámina original.

Un problema de corte no restringido asume que la demanda de ítems es abierta o infinita, y su objetivo es minimizar el desperdicio de material o el número de láminas utilizadas, sin imponer límites a la cantidad de piezas producidas (Russo et al. 2020). Por el contrario, un problema restringido, denominado también *CTDC* (*Constrained Two Dimensional Cutting Problem*), establece límites superiores en la frecuencia o cantidad de cada tipo de ítem, orientando la función objetivo hacia la maximización del valor total de la producción o del aprovechamiento del material disponible (Cui, Zhong, y Yao 2015).

Esta distinción tiene implicaciones prácticas y metodológicas: los problemas restringidos exigen estrategias de control sobre la inserción o asignación de ítems, mientras que los no restringidos se enfocan en la eficiencia del uso del material. En ambos casos, los enfoques heurísticos basados en representaciones arbóreas, como los propuestos por (Fleszar 2013), resultan especialmente adecuados, ya que permiten modelar de manera flexible tanto los límites de frecuencia como las condiciones geométricas de corte guillotina.

(Imahori et al. 2005) proponen un enfoque de optimización en el cual se incorpora una restricción asociada a cada patrón de corte, con el propósito de reducir el número total de aplicaciones de dichos patrones y, por tanto, simplificar la solución del problema. Su propuesta combina técnicas de programación lineal con estrategias de búsqueda local, limitando deliberadamente el tamaño del vecindario de soluciones para concentrar el proceso de búsqueda en regiones más prometedoras del espacio de soluciones. De esta manera, se logra un equilibrio entre la calidad de las soluciones obtenidas y la eficiencia computacional del método.

En este contexto, un *patrón de corte* es una configuración factible de subdivisión del material base de una lámina o placa rectangular en piezas más pequeñas de dimensiones requeridas. Cada patrón representa una disposición espacial específica en la que las piezas se distribuyen sin superposición, respetando los límites del material y las condiciones tecnológicas del proceso (como las restricciones de tipo guillotina). Matemáticamente, un patrón puede interpretarse como un conjunto de variables que indican cuántas piezas de cada tipo se obtienen al aplicar un corte sobre una lámina de dimensiones dadas. En consecuencia, la selección óptima y el número de veces que se aplica cada patrón determinan el plan de corte final que minimiza el desperdicio o el número de materiales base empleados.

Por consiguiente, el problema de corte bidimensional tipo guillotina con rotación ortogonal abarca una extensa variedad de aplicaciones (Furini, Malaguti, y Thomopoulos 2016). El objetivo de resolver este tipo de problemas de corte es proporcionar la mejor utilización de los recursos, la reducción de los costos y así beneficiar la gestión de operaciones (Peña, Orejuela, y Gil 2017).

En literatura más reciente, se han desarrollado enfoques híbridos que combinan modelos exactos y heurísticos para enfrentar la complejidad de instancias reales. Un ejemplo destacado es el trabajo de (Mostajabdaveh, Salman, y Tahmasbi 2022), quienes abordan el problema *Two Dimensional Guillotine Cutting Stock and Scheduling Problem* en la industria de impresión, integrando un modelo de programación entera no lineal con una metaheurística basada en algoritmos genéticos. Su investigación demuestra que, cuando las dimensiones del problema crecen, los métodos exactos resultan computacionalmente costosos, mientras que las heurísticas bien diseñadas logran un equilibrio entre precisión y eficiencia.

A partir de la revisión del estado del arte de los casos más relevante de solución, la presente investigación adopta los lineamientos metodológicos empleados en estudios recientes hacia el dominio del problema de corte bidimensional tipo guillotina en contextos industriales mediante

una estrategia dual combinando un enfoque exacto y un enfoque heurístico, con el propósito de equilibrar rigor matemático y eficiencia computacional. Por un lado, se formula un modelo matemático de programación lineal entera mixta *MILP* (*Mixed interger lineal programming*) el cual es resuelto mediante CPLEX y empleado como referencia comparativa, a fin de resolver los casos de tamaño moderado.

se desarrolla una estrategia heurística implementada en Python, diseñada para generar soluciones factibles de forma eficiente y evaluar su calidad respecto al modelo exacto. y para casos de prueba mayores se propone una metaheurística eficiente basada en la combinación de: un encoding o codificación por lista de permutaciones, que define cómo se representa una solución candidata mediante un orden específico de piezas y determina la secuencia de inserción durante la construcción del patrón; esta representación no constituye un método adicional, sino la base que permite aplicar sistemáticamente la regla de ubicación. Un decoding o decodificación que utiliza la regla Bottom Left (BL) como mecanismo de colocación para transformar la secuencia codificada en un patrón de corte factible bajo restricciones guillotina. BL determina la posición válida más cercana al origen para cada ítem, generando configuraciones rectilíneas de manera eficiente. Su velocidad y compatibilidad con estructuras guillotina la convierten en un procedimiento idóneo para materializar cada solución codificada. Y una metaheurística basada en *VNS* (*Variable Neighborhood Search*); que parte de una solución inicial y una estructura de búsqueda en vecindarios para seleccionar el mejor desempeño. Esta integración permite generar soluciones factibles, explorar múltiples secuencias de inserción y mejorar progresivamente la calidad del patrón de corte, para seleccionar la configuración que minimiza el desperdicio de material. El proceso preserva la simplicidad constructiva y, a la vez, incrementa la capacidad exploratoria a través de cambios estructurados en la representación.

Posteriormente, la propuesta se valida mediante un caso de estudio aplicado a una empresa metalmecánica, junto con instancias de prueba generadas para evaluar el desempeño de ambas estrategias (exacta y aproximada). Permitiendo contrastar ambos métodos en términos de calidad de solución, tiempo de cómputo y eficiencia operativa.

El desarrollo de esta tesis se organiza en una estructura coherente que permite avanzar desde la formulación conceptual del problema hasta su validación práctica en un entorno industrial real. En primer lugar, se presenta la definición del problema de investigación, junto con la pregunta orientadora, el alcance del estudio y los objetivos propuestos. A continuación, se expone la justificación del trabajo, destacando su relevancia en el ámbito académico y en el contexto operativo del sector metalmecánico. El marco de referencia integra los fundamentos teóricos del corte bidimensional tipo guillotina, sus variantes, las metodologías exactas, heurísticas y metaheurísticas, así como una revisión del estado del arte que contextualiza el aporte del presente trabajo frente a los avances existentes. Posteriormente, en la metodología se detalla el diseño general de la investigación, la formulación del modelo matemático exacto, y el desarrollo de la solución metaheurística propuesta, explicando tanto sus componentes como su estructura operativa. El capítulo de resultados presenta el proceso de implementación, la generación de instancias de prueba, los experimentos computacionales y el análisis comparativo entre el modelo exacto y la heurística en diferentes escenarios (los construidos y el real). Finalmente, se exponen las conclusiones derivadas del estudio, las implicaciones prácticas de los hallazgos.

2 DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN

Uno de los desafíos en la industria en general es lograr minimizar el desperdicio de material generado durante sus procesos productivos. Esto impacta directamente en los objetivos financieros a través del mejor aprovechamiento de las materias primas, la reducción de los costos de producción, los tiempos de procesamiento, y el mejoramiento de la gestión del medio ambiente a partir de la reducción y el procesamiento de los residuos.

En general, el sector metalmecánico se encuentra dentro de un rango desfavorable a causa de la cantidad y variedad de residuos que genera; incluso, la generación de chatarra metálica, cuando no es reciclada apropiadamente, conlleva un considerable efecto en lo que a generación de residuos y a contaminación de aguas se refiere. Según la Corporación Autónoma Regional del Valle del Cauca 2022, en el Valle del Cauca se generaron aproximadamente 16.700 toneladas de residuos peligrosos en 2020, de las cuales el 91 % correspondió a grandes generadores, dentro de los que se destaca el sector metalmecánico por su alta producción de chatarra y residuos metálicos.

En la empresa de servicios metalmecánicos en consideración, un importante flujo de los retales metálicos de chatarra generados está relacionado con el proceso de corte de láminas metálicas, el cual representa un 18,6% alrededor de 13,7 Ton, (cifra año 2024), del total de los residuos metálicos generados, lo que evidencia un nivel de pérdida significativo dentro del proceso productivo. Que se consolidan como una variedad de residuos generados de los procesos de mecanizado: retales de varillas, barras y tubería y otros residuos de procesos por arranque de viruta. Entre los cuales podrán considerarse aprovechables solo algunos retales, de acuerdo a sus dimensiones, al sumarse al inventario de materias primas clasificados como "saldos o retales disponibles", incluso sin certeza de uso, hasta que una nueva solicitud de un cliente u orden de compra lo justifique; situación que relaciona tiempos indefinidos de almacenamiento por la baja rotación.

La Tabla 1 presenta la discriminación de los desperdicios de lámina metálica generada por la empresa metalmecánica durante el año 2024, expresados en términos de cantidad y costo relativo. Se observa que el retal de lámina metálica constituye la mayor proporción del total, con un 48,7 %, equivalente a 2,93 toneladas anuales. Le siguen los retales de lámina de acero inoxidable (31,7 %) y de aluminio (14,1 %), mientras que los provenientes de otros materiales representan apenas un 5,5 %.

Tabla 1. Discriminación de los Desperdicios de Lámina en Empresa Metalmecánica (año 2024).

Tipo Desperdicio	Cantidad (Ton/año)	Cantidad (%)	Costo (%)
Retal de lámina metálica	2,93	48,7	8,73
Retal de lámina acero inoxidable	1,91	31,7	6,11
Retal de lámina aluminio	0,85	14,1	2,70
Retal lámina (otros materiales)	0,33	5,5	1,04
TOTAL	6,02	100	18,6

Fuente: Elaboración Propia.

La mayor incidencia del desperdicio en láminas metálicas sugiere oportunidades de mejora; así mismo, la proporción de acero inoxidable y aluminio, aunque menor en cantidad, implica un impacto relevante por su mayor valor unitario en el mercado.

En conjunto, los resultados permiten concluir que la generación de desperdicios no solo representa un problema ambiental, sino también un factor crítico de ineficiencia económica. Estos hallazgos justifican la implementación de estrategias de optimización orientadas al aprovechamiento del material, tales como modelos matemáticos de corte bidimensional, heurísticas de tipo guillotina o esquemas de reutilización de retales en la producción.

En cuanto la empresa escenario de análisis, es de carácter privado, legalmente constituida desde el año 1995, es decir con 30 años de experiencia en el sector metalmecánico. En la actualidad dedicados a la prestación de servicios especializados, mediante el diseño, la producción, el montaje e instalación de estructuras y piezas metálicas. Atendiendo los sectores: industrial (manufactura en general, mobiliario y accesorios, metalmecánico, automotriz y autopartista, alimentos, farmacéutico y agroindustrial), construcción, salud y comercial.

Entre sus líneas de servicios se encuentran:

- Línea de Seguridad Industrial: Fabricación de estructuras metálicas y estanterías con conceptos de normas internacionales: OSHA, AISI, etc.
- Línea Arquitectónica: Soluciones para espacios específicos con características estéticas y funcionales en aplicaciones urbanas y comerciales.
- Línea de Mecanizados Industriales: Mecanizado y fabricación de piezas de precisión, de repuesto y producción en serie y otros servicios.

En la empresa el proceso de corte inicia con una solicitud del cliente, con el propósito de satisfacer un servicio particular mediante orden de compra, donde el cliente especifica los requerimientos del servicio y las variables: cantidad y dimensión. O para fabricar un producto específico. Una vez ingresa la orden de compra o de producción, se genera un plan maestro de fabricación, en el cual se valida inventario de materias primas (existencias de láminas entera o de saldos), dado que la empresa tiene como política, en lo posible programar como prioridad el consumo de saldos o inventarios en reservas, resultantes de cortes anteriores.

A continuación, se procede a generar manualmente la distribución de las piezas requeridas a cortar dentro del espacio de la gran lámina, acomodando de manera que se ocupe en lo posible la totalidad de la lámina para efectos de reducir desperdicio y determinando el trazado de corte. Es posible encontrar dos tipos de distribuciones: Iguales, cuando se establece cortar piezas en serie. O diferentes, para cubrir solicitudes personalizadas. Esta actividad se realiza empíricamente y resulta ser una práctica dispendiosa, de acuerdo a las diferentes alternativas que se deben probar para cumplir la consideración.

La empresa tiene como política, en lo posible programar como prioridad el consumo de saldos o sobrantes y en caso de no ser posible, por diferencias dimensionales, habilita el consumo de láminas nuevas.

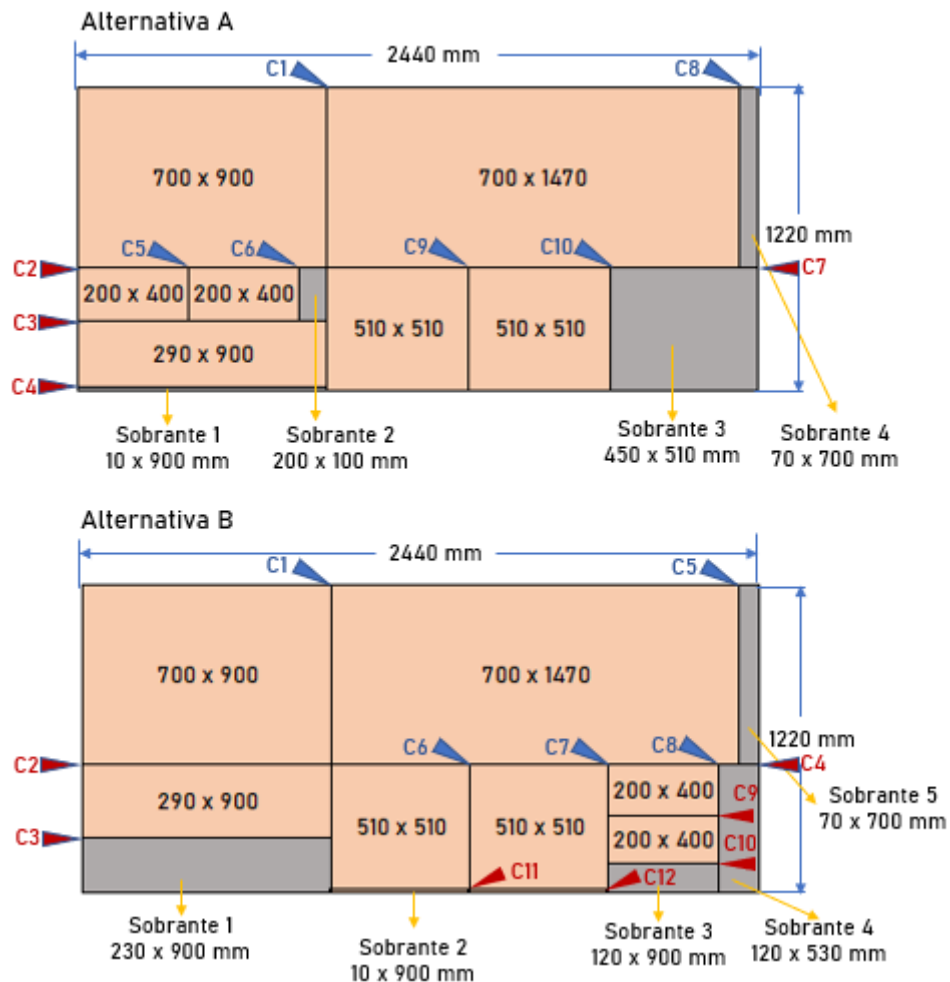
En la figura 3 presenta un ejemplo de distribución y direccionamiento del corte, en el cual se ha dimensionado sobre una lámina CR (Cold Rolled) de calibre 14, con formato inicial en milímetros (mm) de 1220 de Ancho por 2440 de longitud. Propiamente muestra dos alternativas para obtener la misma demanda requerida. La empresa considera la distribución con mejor aprovechamiento de lámina.

Las piezas requeridas se identifican con sus dimensiones, mientras que los sobrantes se presentan en color gris.

El corte es efectuado según lo indica la secuencia, el orden puede proceder por cortes verticales u horizontales distinguidos por letras y números en colores, azul (corte en orientación vertical) y rojo (corte en orientación horizontal); los números referencian la secuencia del corte. La lámina se somete a rotaciones, para efectos de realizar en la maquina el corte requerido.

Paulatinamente la longitud o el ancho de la lámina inicial se reduce por la generación de las nuevas secciones a través del corte.

Figura No. 3 Ejemplo de Dos Alternativas de Distribución Manual de las Piezas Requeridas para Corte en una Cizalla (guillotina), a Partir de una Lámina Entera de 1220 x 2440 mm.



Fuente: Elaboración Propia.

Definidos la distribución de piezas dentro de la lámina entera y la ruta de corte, se inicia el proceso productivo, el cual se lleva a cabo en una cizalla eléctrica (guillotina) modelo FH-9610, con las siguientes especificaciones:

- Capacidad máxima de espesor de lámina: 3.0 mm (láminas de acero Inoxidable y láminas acero laminado en frío (Cold Roll) y laminado en caliente (Hot Roll). Y 3.5 mm (láminas de Aluminio).

- Largo de corte: 3000 mm.
- Especificación del motor: 9 H.P. – 220 V – 3F

Para iniciar la operación dos técnicos son requeridos para realizar los ajustes o cuadros de máquina (la apertura de la sección de entrada de la máquina, según el espesor de la lámina); el acoplamiento del mecanismo de topes (para regular el ingreso de la lámina, conforme a las dimensiones a las que se efectuará el corte). Condicionamiento que resulta útil para el corte en serie (o corte de un lote de láminas de iguales dimensiones) y el accionamiento de la máquina, la movilización y rotación de la lámina.

La inserción de lámina se efectúa en la sección frontal de la maquina se realiza la inserción de la gran lámina, se ubica y alinea a la guía de entrada, se inicia el interruptor de encendido de la maquina y por último el de accionamiento de la máquina para dar inicio al corte y al fraccionamiento de piezas.

En términos del problema n láminas bidimensionales de ancho W_i y largo H_i , son introducidas paulatinamente en la cizalla según la secuencia y orientación de corte (vertical u horizontal), se establece rotación ortogonal para continuar el proceso de corte hasta obtener varias piezas bidimensionales de ancho w_i y largo h_i , según cantidad requerida. Los retales sobrantes son descartados como desperdicio y son almacenados en espera de ser empleados posteriormente.

Esta tesis aborda esta problemática como un problema de corte bidimensional de materiales *2DCSP (Two Dimensional Cut Stock Problem)*, ampliamente estudiado en la literatura de optimización debido a su relevancia práctica en industrias como la manufactura y la logística. Con la aplicación de las variantes tipo guillotina y rotación ortogonal, que consisten en determinar el tipo de corte y los giros necesarios a realizar para obtener las piezas de materiales rectangulares para satisfacer una demanda específica, minimizando el desperdicio generado.

2.1 Pregunta de investigación

De acuerdo con lo anterior, en este proyecto se pretende dar respuesta a la siguiente pregunta de investigación:

¿Cómo cortar de manera óptima lámina bidimensional y determinar eficientemente la secuencia de corte para minimizar el desperdicio de material y reducir el tiempo de proceso?

2.2 Alcance del trabajo de grado

El alcance del presente estudio, establece el análisis de diversas metodologías y trabajos previos como apoyo en la búsqueda de una propuesta de solución al problema de corte con restricciones de tipo guillotina y rotación ortogonal en láminas bidimensionales.

El presente trabajo de grado se enmarca en el estudio del problema de corte bidimensional *2DCSP*, con restricciones de corte tipo guillotina, cuyo objetivo consiste en determinar la disposición óptima de piezas rectangulares dentro de láminas o planchas de material, de modo que se minimice el desperdicio de material o el número de láminas utilizadas.

En este contexto, el desarrollo del proyecto comprende las siguientes actividades principales:

- Formulación de un modelo de exacto MILP que represente de manera exacta el problema de corte bidimensional con restricciones de guillotina, considerando láminas de dimensiones fijas y demandas conocidas para cada tipo de pieza.

- Diseño e implementación de una heurística que permita obtener soluciones de buena calidad en tiempos computacionales compatibles con la operación industrial, orientada a resolver instancias de mayor tamaño que resultan difíciles de abordar mediante métodos exactos.”
- Evaluación del desempeño tanto del MILP como de la heurística propuesta, a partir de la generación y análisis de instancias experimentales, comparando los resultados obtenidos en términos de calidad de solución y tiempo de cómputo.
- Análisis de los resultados, con el propósito de identificar las principales ventajas, limitaciones y posibles líneas de mejora en la aplicación de los métodos propuestos.

3 OBJETIVOS DEL PROYECTO

A continuación, se describen los objetivos a desarrollar con la presente propuesta.

3.1 Objetivo general

Desarrollar e implementar una solución eficiente que permita encontrar una solución aproximada y de calidad al problema de corte de láminas metálicas bidimensionales con restricción de corte guillotina y rotación ortogonal con aplicación a un caso real, en una empresa local del sector metalmeccánico, para minimizar desperdicio de material y reducir el tiempo de procesamiento.

3.2 Objetivos específicos

- Realizar una amplia revisión literaria para determinar los métodos de solución más relevantes en casos aplicados del 2DCSP.
- Caracterizar y cuantificar el sistema del problema de corte.
- Diseñar y desarrollar un algoritmo heurístico para la formulación del modelo que resolverá de manera aproximada el problema de corte.
- Implementar y probar el algoritmo heurístico con el caso de estudio del problema de corte de una empresa local de servicios metalmeccánicos.

3.3 Resultados esperados

Esta investigación tiene como enfoque de resultado encontrar una solución aproximada al problema de corte bidimensional de tipo guillotina con rotación ortogonal, a través del desarrollo de una combinación de algoritmos heurísticos que genere patrones de corte para el seccionamiento eficiente de lámina bidimensional, minimizando desperdicio de material y optimizando el tiempo de proceso. El resultado final integra los siguientes entregables:

- Revisión del estado del arte sobre el problema de corte bidimensional tipo guillotina con rotación ortogonal: el resultado esperado incluye una revisión de los conceptos generales del problema de corte bidimensional, con el fin de analizar los enfoques tradicionales y recientes, contextualizar el problema, comprender los enfoques previos e identificar las brechas y limitaciones de los métodos existentes en términos de eficiencia, calidad de las soluciones y aplicabilidad en el escenario industrial de interés. Para justificar la metodología propuesta que aborda el problema de corte bidimensional, incluyendo, sin limitarse a, los siguientes aspectos: Métodos exactos, técnicas heurísticas y metaheurísticas utilizadas en el pasado, algoritmos clásicos para el problema de corte guillotina y sus variantes, enfoques aplicados a la optimización de materiales y tiempos de producción; así como la comparación de resultados obtenidos con distintas metodologías.
- La formulación de un modelo matemático exacto de programación entera definiendo sus conjuntos, parámetros, variables, función objetivo y restricciones. Se determinará la solución de casos de prueba mediante solucionadores comerciales como CPLEX o Gurobi.
- El Diseño de codificación, que considere las restricciones del problema, capaz de generar cortes tipo guillotina y ubicar la mayor cantidad posible de ítems, permitiendo rotación ortogonal de estos. A partir de una heurística que integra una codificación mediante un

vector de permutaciones, un decoding mediante una heurística constructiva Bottom Left, y una heurística basada en VNS para búsqueda de múltiples vecindarios que permitan ampliar el espacio de soluciones y un criterio de selección para determinar el de mejor desempeño. La cual empleara como esquema de validación instancias de prueba construidas para generar configuraciones de ítems y demandas de manera automatizada. Este proceso permitirá obtener conjuntos de casos variados, factibles, controlados y representativos, adecuados para evaluar comparativamente el desempeño de la heurística propuesta respecto al modelo exacto como solución óptima.

- Evaluación experimental en un entorno industrial real. La propuesta será validada mediante un caso de estudio proveniente de una empresa local del sector de servicios metalmecánicos, utilizando datos reales (dimensiones de láminas y demandas requeridas). Esta fase permitirá analizar la aplicabilidad del modelo exacto y de la heurística en un escenario práctico, comparando su desempeño frente a condiciones reales de demanda, disponibilidad de material y restricciones operativas. El análisis incluirá mediciones de tiempo computacional, calidad de las soluciones y reducción de desperdicios, con el fin de determinar la utilidad de la metodología propuesta para apoyar la toma de decisiones en procesos de corte industrial.

En conjunto, estos resultados esperados proporcionan un marco integral que articula la fundamentación teórica, el desarrollo metodológico y la validación computacional de la propuesta.

La combinación del modelo exacto y la heurística permitirán no solo contrastar enfoques de naturaleza distinta, sino también demostrar la aplicabilidad y pertinencia de la metodología en escenarios industriales reales. Con ello, la investigación aspira ofrecer una contribución tanto conceptual como práctica al campo del corte bidimensional tipo guillotina, sentando bases sólidas para futuras mejoras en la eficiencia del uso de materiales y el desempeño operativo en contextos metalmecánicos.

4 JUSTIFICACIÓN DEL TRABAJO DE GRADO

La industria metalmecánica desempeña un papel crucial en el desarrollo económico e industrial, siendo clave para diversos sectores económicos como la construcción, la manufactura y la tecnología. Sin embargo, enfrenta retos significativos, relacionados con el problema de corte, que afectan la eficiencia de los procesos, el incremento de los costos y del desperdicio de materiales.

El presente estudio justifica su relevancia al abordar un problema complejo mediante el desarrollo de una solución eficiente que emplea dos técnicas de solución: un método exacto y una heurística. Herramientas que, si bien permiten tratar un problema con rigor académico, también adaptar las restricciones y condiciones reales de los entornos industriales, ofreciendo soluciones prácticas y aplicables.

La propuesta de optimización tiene como objetivo incrementar la eficiencia del proceso, la reducción del desperdicio de materiales y disminuir los tiempos de producción. Resultados que tienen un impacto directo en la competitividad de la empresa y en su capacidad para operar de manera más sostenible y rentable.

Este trabajo entrega una contribución relevante en el ámbito académico, por que combina enfoques teóricos avanzados con aplicaciones prácticas para la mejora de los procesos industriales; fomenta el avance del conocimiento en áreas como la optimización, la investigación operativa y la sostenibilidad industrial.

5 MARCO DE REFERENCIA

El 2DCSP es un problema clásico en el campo de la optimización combinatoria y la investigación operativa. Consiste en determinar la mejor manera de dividir materiales planos, como láminas metálicas, cartón, vidrio o textiles, en piezas más pequeñas que satisfagan requisitos específicos de tamaño y forma; siendo particularmente relevante en varios sectores industriales.

El objetivo principal del proceso de corte bidimensional es maximizar el aprovechamiento de material disponible. Este proceso resulta fundamental, dado que las piezas obtenidas se emplean en la fabricación de productos finales, y por tanto incide directamente en la eficiencia de los procesos productivos y en los costos operativos.

La reducción del desperdicio se logra mediante un diseño eficiente de *patrones de corte*, como forma específica de distribución y corte de piezas dentro de una lámina para cumplir la demanda y aprovechar al máximo el material y que permite generar piezas más pequeñas, de diferentes tamaños y formas, a partir de una sola lámina. De esta manera, se garantiza el cumplimiento de las especificaciones requeridas, se optimiza la productividad general de la operación y se mejoran los tiempos de producción, incrementando la flexibilidad ante los cambios en la demanda.

El problema de corte bidimensional tiene una amplia relevancia en el ámbito industrial. En la metalmecánica, este problema se presenta en la fabricación de componentes estructurales, piezas para maquinaria y herramientas metálicas, donde se busca optimizar el uso de láminas de acero, aluminio u otros metales. En la industria maderera, se aplica al corte de paneles y tableros destinados a la producción de mobiliario y elementos constructivos. En la del vidrio, contribuye a la elaboración de vidrios planos y paneles a medida, mientras que en la del cartón y los plásticos permite diseñar empaques y embalajes personalizados. En conjunto, estas aplicaciones industriales evidencian la importancia del problema de corte como herramienta para reducir los costos de material, mejorar los tiempos de producción y aumentar la capacidad de respuesta ante demandas personalizadas.

Dentro de estas aplicaciones, la industria metalmecánica ocupa un papel especialmente relevante. Este sector agrupa un amplio conjunto de empresas y fábricas dedicadas a la generación, procesamiento, distribución y comercialización de maquinaria, herramientas y bienes de consumo elaborados a partir de metales con diversas configuraciones físicas. En este contexto, la implementación de estrategias eficientes de corte bidimensional puede generar beneficios significativos en varios aspectos.

5.1 Marco conceptual

5.1.1. Conceptos Generales del Corte Bidimensional

El término *bidimensional* hace referencia a la naturaleza del material de entrada, que se representa en un plano con dos dimensiones principales: ancho y largo. Durante el proceso de corte, el grosor de la lámina se considera constante y no influye directamente en las decisiones de diseño u optimización.

El Corte bidimensional se caracteriza por:

- *Dimensionalidad*: Implica trabajar en un espacio de dos dimensiones (ancho y largo), donde el grosor del material se considera constante.

- *Geometría de las piezas:* Pueden ser rectangulares o tener formas más complejas, incluso irregulares, dependiendo de los requisitos del diseño.
- *Patrones de corte:* Configuración factible de subdivisión del material base (o lámina inicial), diseñados para maximizar el aprovechamiento del material y minimizar el desperdicio.
- *Restricciones del proceso:* Pueden incluir la orientación fija o rotacional de las piezas, la secuencia de los cortes; así como limitaciones físicas de las herramientas de corte.
- *Criterio de optimización:* Busca maximizar el uso del material disponible (minimización del desperdicio) o satisfacer una demanda específica con el menor costo.

Tipología:

La extensión de aplicaciones del problema de corte bidimensional abarca diversos escenarios industriales orientados a optimizar el uso de recursos espaciales, tales como el material disponible, el tiempo de procesamiento o la capacidad de contenedores. Desde el punto de vista computacional, este tipo de problemas pertenece a la clase de los problemas NP-Hard, lo que significa que su complejidad crece exponencialmente con el tamaño de la instancia y que no se conoce un algoritmo que garantice la obtención de la solución óptima en tiempo polinomial. Esta característica implica que, para instancias de gran escala, los métodos exactos resultan ineficientes o inviables computacionalmente, por lo que es común recurrir a enfoques heurísticos que permitan obtener soluciones de buena calidad en tiempos razonables.

Por lo tanto, el marco general del problema de corte y empaque *C&P* (*Cutting and Packing*) engloba una familia de problemas como *2DBPP*, *2DSPP*, *2DKP*, *2DCSP* y *TL*, que conforman un ecosistema de variantes diseñadas para resolver situaciones específicas bajo distintas objetivos y restricciones. Se describen a continuación:

- *2DBPP Problema de Embalaje en Contenedores Bidimensionales o Two Dimensional Bin Packing Problem.* Tiene como objetivo empaquetar ítems rectangulares en un número mínimo de contenedores (bins) de dimensiones fijas, sin que se solapen ni excedan las capacidades del contenedor. Una de las propiedades más relevantes del modelo es la minimización de la cantidad de contenedores utilizados. Entre sus características, pueden incluir restricciones como rotación libre o a 90° . Es muy frecuente en aplicaciones de diseño de empaques, logística de transporte y almacenamiento.
- *2DSPP Problema de Empaque en Tiras Bidimensionales o Two Dimensional Strip Packing Problem.* Busca empaquetar ítems en un contenedor de ancho fijo y altura infinita o lo suficientemente grande, minimizando la altura utilizada. Se enfoca en la organización vertical de los ítems dentro de un ancho limitado; muy similar al *2DBPP*, pero con contenedores "infinitos" en una dimensión. Común en aplicaciones como el corte de láminas largas en la industria textil, maderera y metalmecánica.
- *2DKP Problema de la Mochila Bidimensional o Two Dimensional Knapsack Problem.* Se centra en maximizar el valor total de los ítems empaquetados en un contenedor bidimensional de tamaño fijo e incluye restricciones de rotación y de posición. Cada ítem tiene un valor asociado, además de sus dimensiones. En términos generales no busca minimizar contenedores; sin embargo, comparte el uso eficiente del espacio, priorizando el valor agregado. Entre sus aplicaciones útil en la optimización de inventarios y selección de productos de alto valor.

- *2DCSP Problema de Corte de Material Bidimensional o Two Dimensional Cutting Stock Problem*. Tiene como fin cortar láminas grandes (stocks) en piezas más pequeñas, cumpliendo con una demanda específica y minimizando el desperdicio (o Trim Loss). Basado en patrones de corte específicos, puede incluir múltiples tipos de láminas y restricciones por ejemplo de guillotina. Tiene amplias aplicaciones en la industria metalmeccánica, fabricación de vidrio y papel y manufactura de muebles.
- *TL Pérdida por recorte o Trim Loss*: Se centra exclusivamente en la eficiencia del material desperdiciado al realizar cortes en un problema de optimización bidimensional. Al ser un subproblema del 2DCSP y formar parte del marco de Cutting and Packing como una métrica de eficiencia para ambos problemas de corte
- *C&P Problema Corte y Empaque*: Por sus siglas en inglés Cutting and Packing. Unifica todos los problemas al considerar tanto el corte como el empaque en distintos escenarios, dividiéndolos en dos grandes categorías: Problemas de corte (Cutting): 2DCSP y por otra parte, en problemas de empaque (Packing): 2DBPP, 2DSPP y 2DKP.

Es importante mencionar el Problema de Corte de Material Unidimensional *1DCSP*, (*One Dimensional Cutting Stock Problem*), porque constituye la base conceptual y metodológica de los problemas de corte en dimensiones superiores, incluido el bidimensional. El 1DCSP se clasifica también como un problema NP-Hard y presenta diversas aplicaciones cuyas estrategias de solución pueden extenderse al ámbito bidimensional. Este problema tiene como objetivo minimizar el desperdicio de material, es decir, el sobrante después del corte y, de forma alternativa, reducir el número de piezas base necesarias para satisfacer la demanda. En él los cortes deben planificarse de modo que la suma de las longitudes de las piezas cortadas no exceda la longitud del material base, garantizando que se cumpla la cantidad requerida para cada longitud solicitada. El material de partida suele consistir en piezas largas, como barras metálicas, rollos de papel o tabloncillos de madera, que deben fraccionarse en secciones más pequeñas con longitudes específicas. Tiene aplicaciones en diferentes sectores industriales para hacer efectivo el corte.

Su relación con los problemas bidimensionales radica en sus objetivos y características compartidas: optimizar recursos, minimizar desperdicios y mejorar la organización en espacios bidimensionales.

Variantes:

El problema 2DCSP presenta una amplia variedad de variantes derivadas de las condiciones geométricas, operativas y tecnológicas que caracterizan los procesos de manufactura. Estas diferencias han sido ampliamente revisadas en la literatura (Cheng, Feiring, y Cheng 1994) (R. W. Haessler y Sweeney 1991), donde se destacan aquellas relacionadas con el tipo de corte permitido, la posibilidad de rotación de las piezas y la repetitividad de los patrones de corte.

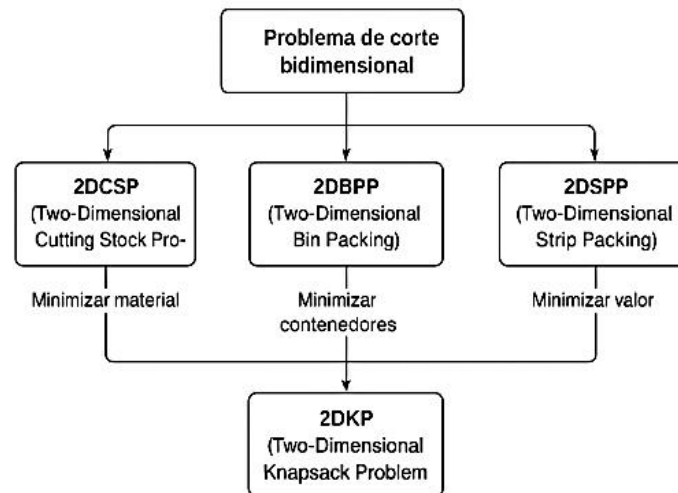
Estas variantes surgen de las restricciones impuestas por los equipos de corte, las propiedades de los materiales, las políticas de producción o las estrategias de optimización adoptadas. En particular, factores como el tipo de corte permitido, las posibilidades de rotación de las piezas, la repetitividad de los patrones, o la homogeneidad del material, determinan la estructura matemática del problema y su nivel de complejidad computacional.

Comprender estas variantes resulta esencial para seleccionar el modelo y el enfoque de solución más adecuado, así como para representar de manera realista las condiciones industriales que se buscan optimizar.

De manera general, las variantes más estudiadas son: 2DCSP, 2DBPP, 2DSPP y el 2DKP. Cada una de estas variantes comparte la misma estructura combinatoria (la disposición no superpuesta de rectángulos dentro de un área finita), pero difiere en su criterio de optimización y tipo de restricción principal (cantidad de láminas, longitud disponible o valor de los ítems). Estas diferencias inciden directamente en la complejidad computacional y en la estrategia de resolución elegida, lo que justifica su análisis independiente dentro del campo de los problemas de corte y empaquetamiento.

El diagrama jerárquico ilustra la relación entre las principales variantes del problema de corte bidimensional, que se subdivide en cuatro clases principales según su función objetivo y tipo de restricción.

Figura 4. Esquema Jerárquico de los Problemas de Corte Bidimensional



Fuente: Elaboración Propia.

Además de las variables clásicas, la literatura identifica otras como formulaciones extendidas asociadas al tipo de corte, a la orientación de las piezas o repetitividad de los patrones; que incorporan demandas variables, pérdida admisible o multiobjetivo (R. W. Haessler y Sweeney 1991), (Mingjie Hu 2023). Algunas de ellas integran el corte con la programación de operaciones (Mostajabdaveh, Salman, y Tahmasbi 2022), o consideran restricciones temporales y prioridades en la asignación de piezas.

Por otra parte, características del problema de corte bidimensional:

- *Corte tipo guillotina:* Se define como un corte recto ortogonal (vertical u horizontal) que atraviesa completamente la lámina, de extremo a extremo, dividiéndola en dos subpiezas rectangulares. Esta restricción, estudiada en profundidad por (Beasley 1985) y posteriormente retomada por (Aryanezhad et al. 2012), simplifica la operación de

corte y permite modelar el proceso mediante secuencias jerárquicas de subdivisiones rectangulares.

Este tipo de corte es característico de procesos industriales en los que las herramientas presentan restricciones físicas, como ocurre con las cizallas empleadas en la industria metalmeccánica. Su principal ventaja radica en la simplicidad operacional y en la posibilidad de automatizar el proceso de corte, aunque limita la flexibilidad en la disposición de las piezas.

- *Corte con rotación ortogonal:* Las piezas pueden rotarse únicamente en múltiplos de 90° (0° , 90° , 180° , 270°). Esta restricción es común en la literatura (Israni y Sanders 1982), (García et al. 2006), por que facilita el control dimensional y la manipulación física de las piezas. En contraste, permitir rotaciones libres o restringidas incrementa el espacio de búsqueda y la complejidad computacional del modelo. En entornos industriales, donde se busca reducir la complejidad de manipulación y mantener una orientación coherente con la geometría rectangular de las láminas, la rotación ortogonal permite cierto grado de optimización espacial sin comprometer la estabilidad del proceso.
- *Corte con rotación restringida:* En esta modalidad se permite la rotación de las piezas bajo un conjunto limitado de ángulos específicos, definidos previamente según las características del material o del proceso productivo. Este tipo de restricción busca equilibrar la flexibilidad geométrica con las limitaciones mecánicas o de sujeción de la maquinaria.
- *Corte sin rotación:* Las piezas deben mantenerse en su orientación original, sin posibilidad de giro. Esta condición simplifica el modelado y reduce el espacio de búsqueda computacional, aunque puede afectar el aprovechamiento del material al restringir las combinaciones posibles dentro de la lámina.
- *Corte sin restricciones de orientación:* En esta variante se permite rotar las piezas en cualquier ángulo, lo que incrementa las posibilidades de disposición y el potencial de aprovechamiento del material. Sin embargo, este enfoque conlleva una mayor complejidad computacional, ya que amplía significativamente el espacio de búsqueda y las variables geométricas a considerar en la optimización.
- *Corte con patrones únicos o repetitivos:* Define si los patrones de corte pueden repetirse o si deben ser exclusivos para cada sección del material. En los patrones únicos, cada corte se diseña de forma independiente, mientras que los patrones repetitivos permiten aplicar un mismo esquema de corte en distintas secciones de la lámina. Esta característica influye directamente en la eficiencia operativa, el tiempo de configuración de maquinaria y la consistencia del producto final.

Las variantes que contemplan el uso de patrones repetitivos o la presencia de materiales heterogéneos han sido abordadas en estudios recientes (Andrade, Birgin, y Morabito 2016), donde se considera la reutilización de restos o la asignación de piezas a láminas de distintos tamaños y propiedades

- *Corte heterogéneo:* Considera la existencia de múltiples tipos de materiales, tamaños de láminas o restricciones de procesamiento. En problemas homogéneos, todas las láminas comparten las mismas propiedades físicas y costos; en cambio, los problemas

heterogéneos incorporan variabilidad en los materiales, espesores o limitaciones técnicas, lo cual aumenta la complejidad del modelo y requiere estrategias adicionales para equilibrar el costo con el aprovechamiento de recursos.

Existen otras formulaciones del problema de corte bidimensional que incorporan restricciones adicionales o criterios de optimización más complejos. Estas extensiones amplían el alcance del modelo al considerar aspectos como la variabilidad de la demanda, la existencia de múltiples materiales, la inclusión de objetivos simultáneos o la adaptación dinámica del proceso de corte. En conjunto, estas variantes representan escenarios más realistas de aplicación industrial, en los que el problema de optimización debe equilibrar eficiencia, flexibilidad y costo computacional.

- *Problema de Corte y Empaque con Demanda Variable:* El problema incluye demandas específicas para cada pieza, lo que requiere cumplir con cantidades mínimas o máximas y se subclasifica en: Demanda exacta (Cada tipo de pieza debe ser producida en una cantidad fija), Demanda flexible (Se permite un margen de variación en la producción. Este tipo de problema tiene aplicación en sectores donde la personalización de productos es crucial, como la impresión digital y la carpintería industrial).
- *Problema de Corte con Pérdida Admisible (Skiving):* Se permite un cierto nivel de pérdida de material durante el proceso de corte. De este modo, en el caso de que se trate de un corte exacto, el objetivo es minimizar completamente el desperdicio, por otra parte, si es el caso de corte con pérdida tolerada se aceptan pequeñas pérdidas para simplificar el proceso o reducir los costos de computación. Es decir que su mayor énfasis de aplicación son situaciones donde el costo del desperdicio es menor que el de optimización extrema, como en materiales de bajo costo.
- *Corte Multiobjetivo:* Este tipo de corte considera múltiples objetivos simultáneamente, como minimizar el desperdicio y el tiempo de corte, o maximizar la productividad y la sostenibilidad. Asocia objetivos conflictivos es decir que el logro de un objetivo puede dificultar alcanzar otro; por ejemplo, menos desperdicio puede aumentar los tiempos. Para resolverse, se utilizan enfoques de optimización multiobjetivo, generalmente encontrando la frontera de Pareto (Pareto fronts).
- *Corte con Múltiples Materiales o Placas:* Se dispone de varias láminas de diferentes dimensiones con el objetivo de asignar piezas a cada una de las placas para maximizar el uso total del material.
- *Corte con Restricciones de Tiempo o Secuencia:* Además de optimizar el uso del material, se deben minimizar los tiempos de procesamiento o cumplir con un orden específico en los cortes.
- *Corte con Prioridades o Pesos Asignados:* Algunas piezas tienen mayor prioridad o valor, por lo que el objetivo es maximizar el valor total obtenido, no solo minimizar el desperdicio.
- *Corte en Tiempo Real:* Se busca generar patrones de corte dinámicos, adaptándose a nuevas piezas o cambios en la demanda en tiempo real.

En conjunto, las variantes del problema de corte bidimensional evidencian la evolución del campo hacia modelos más representativos de la realidad productiva. Desde los primeros

enfoques de corte tipo guillotina (Beasley 1985) hasta las formulaciones multiobjetivo y adaptativas recientes, la literatura ha demostrado que la flexibilidad y la precisión del modelo dependen directamente de la consideración de estas variantes (Cheng, Feiring, y Cheng 1994)(R. W. Haessler y Sweeney 1991). Por ello, su estudio resulta fundamental para seleccionar el enfoque metodológico más adecuado según el tipo de aplicación y los objetivos de optimización propuestos.

5.1.2 Métodos Existentes para Resolver el Problema de Corte Bidimensional y Limitaciones Actuales

El problema del corte bidimensional se aborda con métodos de optimización exactos (clásicos) y aproximados (heurísticos o metaheurísticos), según la complejidad y el tamaño del problema. Los métodos exactos buscan la solución óptima garantizada, mientras que los aproximados se centran en obtener soluciones de alta calidad con menor costo computacional (Dyckhoff 1990) (Wäscher, Haußner, y Schumann 2007).

Métodos Clásicos:

Los métodos clásicos se basan en técnicas matemáticas exactas que permiten modelar y resolver problemas de optimización. Estos enfoques buscan soluciones óptimas, pero su efectividad varía según la complejidad y el tamaño del problema. (Kantorovich 1960) (Gilmore y Gomory 1961) (Lodi, Martello, y Vigo 2002).

A continuación, se describen los principales métodos clásicos:

- *MILP Programación Lineal Entera Mixta:* Por sus siglas en inglés Mixed Integer Linear Programming, modela los problemas mediante una función objetivo, generalmente para minimizar desperdicio o maximizar el uso del material y un conjunto de restricciones que definen las dimensiones de la pieza, el número de cortes permitidos, las rotaciones y las capacidades del material base.

Este enfoque, introducido formalmente por (Kantorovich 1960) y extendido por (Gilmore y Gomory 1961) (Gilmore y Gomory 1963), ha sido ampliamente aplicado a los problemas de corte y empaquetamiento.

Las soluciones se obtienen mediante algoritmos como Simplex, Branch and Bound o Branch and Cut, implementados en herramientas de optimización como CPLEX, Gurobi o GLPK (Nemhauser y Wolsey 1988) .

Aunque garantiza soluciones óptimas, su complejidad computacional limita su aplicabilidad en instancias grandes o con múltiples restricciones operativas.

- *Branch and Bound (Ramificación y Poda):* Propuesto por (Land y Doig 1960), es una técnica exacta que divide el problema en subproblemas más pequeños para resolverlos de manera sistemática, no explora todas las posibles soluciones, pero garantiza encontrar la mejor.

Funciona explorando un árbol de decisiones que divide el problema original en subproblemas más simples, calcula límites superiores e inferiores (cotas) para determinar si una rama puede contener una mejor solución y elimina subproblemas que no mejoran la solución óptima (poda), (Martello 1990).

Encuentra soluciones óptimas sin explorar todas las combinaciones posibles y en problemas estructurados es más mucho más eficiente que una búsqueda exhaustiva, pero su eficiencia decae en problemas complejos. Pues, a medida que el problema crece, el árbol de decisiones puede volverse inmanejable, por lo que requiere tiempo computacional significativamente grande para problemas grandes.

- Método de Generación de Columnas (Column Generation CG): La técnica introducida por (Gilmore y Gomory 1961) (Gilmore y Gomory 1963), permite abordar problemas de gran escala formulados como MILP. Este método ha demostrado ser especialmente útil para problemas de corte, empaque, transporte y producción (Vance 1998) (Pisinger y Ropke 2007), de gran escala con muchas variables de decisión (columnas) donde las combinaciones posibles de soluciones son inmanejables para enfoques clásicos y reduce significativamente el tamaño del problema al evitar generar todas las variables posibles desde el inicio. Sin embargo, su principal limitación es el alto costo computacional asociado con la resolución iterativa de los problemas maestro y subproblema, lo cual exige conocimientos avanzados en optimización y modelado matemático.

Comienza con un conjunto básico de columnas o patrones factibles para el problema maestro. Por ejemplo, patrones de corte iniciales que cumplen con las restricciones del problema, de esta manera descompone el problema en dos partes principales: un problema maestro que representa el problema original, pero inicialmente solo con un subconjunto reducido de las posibles variables (o patrones de corte), resolviendo iterativamente para llegar a una solución factible con base en las columnas actuales. Por otro lado, un subproblema que identifica las nuevas variables o patrones (columnas) que podrían mejorar la solución del problema maestro y resuelve un problema auxiliar, generalmente por programación lineal evaluando el "costo reducido" o la contribución de agregar nuevas columnas.

Posteriormente en la actualización las columnas generadas en el subproblema se añaden al problema maestro y se repite el proceso iterativamente hasta que no se puedan generar nuevas columnas con costos reducidos negativos, indicando que la solución actual es óptima.

Este método encuentra soluciones óptimas, especialmente en problemas complejos con muchas restricciones y puede adaptarse a problemas específicos mediante ajustes en la formulación del maestro y el subproblema. Sin embargo, requiere resolver iterativamente dos problemas (maestro y subproblema), lo que incrementa el tiempo de procesamiento en instancias grandes por supuesto relacionando recursos computacionales significativos y conocimientos especializados.

Los métodos exactos, por una parte, garantizan soluciones óptimas y ofrecen un marco matemático sólido. Sin embargo, tienen lugar en función de las necesidades específicas del problema donde el tiempo de cómputo se incrementa de forma exponencial (Lodi, Martello, y Vigo 2002) (Wäscher, Haußner, y Schumann 2007).

La tabla 2. Ilustra una síntesis general de las características, ventajas y desventajas de los métodos clásicos.

Tabla 2. Comparación General de Métodos Exactos.

Métodos	Aplicaciones	Ventajas	Desventajas
MILP - Programación Lineal Entera Mixta	Optimización de patrones en manufactura.	Soluciones óptimas, incluye restricciones complejas.	No escala bien a problemas grandes.
Branch & Bound	Problemas pequeños a medianos.	Solución exacta, explora sistemáticamente.	Complejidad crece exponencialmente con el tamaño.
CG Generación Columnas	Problemas grandes y complejos	Alta precisión y soluciones óptimas	Costoso computacionalmente

Fuente: Elaboración Propia.

Métodos Metaheurísticos:

Son estrategias de optimización generales que se utilizan para resolver problemas complejos de optimización, especialmente en problemas NP-Hard, aquellos para los cuales no es práctico o posible encontrar una solución exacta debido a su complejidad computacional o al tamaño del espacio de búsqueda. Las metaheurísticas son técnicas aproximadas diseñadas para encontrar soluciones especialmente útiles para problemas de gran escala con múltiples restricciones dado que combinan elementos de búsqueda aleatoria y sistemática para encontrar soluciones de alta calidad en tiempos razonables, sacrificando la optimalidad exacta a cambio de eficiencia computacional (Blum y Roli 2003) (Crainic, Perboli, y Tadei 2009).

Las metaheurísticas se clasifican en dos grandes categorías según su enfoque de búsqueda:

Metaheurísticas de Búsqueda Local:

Exploran el espacio de soluciones moviéndose desde una solución inicial hacia soluciones vecinas, buscando mejorar iterativamente la calidad de la solución, centrandose su búsqueda en una solución inicial específica. Tienen estrategias de diversificación, para escapar de óptimos locales, a través de mecanismos de diversificación y aceptación probabilística de soluciones peores (Osman y Laporte 1996) (Talbi 2009).

En esta categoría se encuentran:

- *Recocido Simulado (Simulated Annealing SA)*: Es una metaheurística inspirada en el proceso físico de recocido de metales (Kirkpatrick, Gelatt, y Vecchi 1983), donde los materiales se enfrían lentamente para minimizar la energía y obtener una estructura estable, en la aplicación a partir de este comportamiento explora soluciones aceptando temporalmente las peores para evitar quedarse en mínimos locales. Es una herramienta sencilla de implementar y ajustable a diferentes problemas. Sin embargo, su rendimiento obedece a la correcta configuración de la tasa de enfriamiento y de la función de vecindad, por lo que puede requerir de un tiempo significativo si el proceso de enfriamiento es muy lento.

En la aplicación el método comienza con una solución inicial y una temperatura alta e iterativamente va generando una solución vecina; si la nueva solución es mejor,

se acepta, si por el contrario resulta ser peor igual se acepta con una probabilidad dependiente de la diferencia entre soluciones y la temperatura actual. Al final se espera que la temperatura disminuya gradualmente, reduciendo la probabilidad de aceptar soluciones peores.

- *Búsqueda Tabú (Tabu Search TS)*: Es una metaheurística de búsqueda local intensificada, propuesta por (Glover 1989), flexible que se adapta a diferentes problemas por lo que es diseñada para resolver problemas de optimización combinatoria.

Se basa en iterar sobre una solución inicial moviéndose hacia soluciones vecinas, empleando una estructura de memoria a corto plazo (lista tabú), para evitar soluciones repetidas (ciclos), favoreciendo la exploración de nuevas áreas dado que registra las ya visitadas. Este método combina exploración y explotación, permitiendo que se rompan restricciones temporales si se detecta una mejora significativa (criterio de aspiración). Su desempeño depende de parámetros como el tamaño de la lista tabú y los criterios de parada, lo que la hace poderosa pero sensible al ajuste de parámetros (Gendreau y Potvin 2010).

Su rendimiento depende de la correcta configuración de parámetros (el tamaño de la lista tabú y el criterio de parada. En caso de problemas con grandes vecindades puede ser computacionalmente costosa para problemas con vecindades grandes.

- *Búsqueda Vecindario Variable (Variable Neighborhood Search VNS)*: Es una metaheurística que explora dinámicamente diferentes vecindades de soluciones para diversificar la búsqueda escapando de óptimos locales y mejorar la exploración del espacio de soluciones (Mladenović y Hansen 1997). Presenta gran adaptabilidad al integrarse con otras metaheurísticas y es efectiva en aplicaciones de problemas con múltiples óptimos locales. –Sin embargo, requiere una definición adecuada de vecindades para cada problema y puede ser computacionalmente costosa si se usan vecindades grandes.

Funciona al definir un conjunto de vecindades, cada una representando diferentes tipos de cambios en la solución. Comienza en una solución inicial y busca mejorarla dentro de la primera vecindad; si no encuentra mejora, se cambia a la siguiente vecindad y el proceso se repite hasta cumplir un criterio de parada (número de iteraciones, tiempo o sin mejoras significativas).

La tabla 3. Ilustra una síntesis general de las características, ventajas y desventajas de las metaheurísticas de búsqueda local citadas.

Tabla 3. Comparación General de Metaheurísticas.

Metaheurística	Características	Ventajas	Desventajas
SA - Recocido Simulado	Evita ciclos mediante memoria a corto plazo.	Escapa eficientemente de óptimos locales.	Alta dependencia de parámetros.

TS - Búsqueda Tabú	Acepta soluciones peores con probabilidad decreciente.	Escapa de óptimos locales con simplicidad.	Sensible al ajuste de la temperatura.
VNS - Búsqueda Vecindario Variable	Cambia dinámicamente de vecindad para diversificar.	Explora diferentes regiones del espacio.	Requiere diseño adecuado de vecindades.

Fuente: Elaboración Propia.

Metaheurísticas Basadas en Población:

Son técnicas de optimización que operan sobre un conjunto de soluciones candidatas (una población) en cada iteración, en lugar de trabajar con una única solución como ocurre en las metaheurísticas de búsqueda local.

Se caracterizan por que cada solución representa un punto en el espacio de búsqueda, exploran el espacio de soluciones moviéndose desde una solución inicial hacia soluciones vecinas, buscando mejorar iterativamente la calidad de la solución. Por definición tienen estrategias para escapar de óptimos locales.

Son ampliamente aplicadas en el 2DCSP, problemas combinatorios y de optimización multiobjetivo (Eiben y Smith 2015) (Baeck, Fogel, y Michalewicz 1997), para abordar problemas complejos de optimización combinatoria, continua, multiobjetivo y en el procesamiento de imágenes y machine learning.

Muchos de estos métodos están inspirados en diferentes procesos. Por ejemplo, evolución biológica, comportamiento social entre otros fenómenos físicos.

Los inspirados en la evolución biológica:

- *Algoritmos Genéticos (Genetic Algorithm GA):* Propuestos por (Holland 1975), simulan la selección natural, utilizando operadores como cruce, mutación y selección. Los algoritmos evolutivos enfocados en conceptos generales de evolución, como evolución diferencial y los de programación evolutiva con características similares a los GA, enfatizan en la mutación en lugar del cruce.

Es una metaheurística basada en la teoría de la evolución biológica de Darwin y utilizan una población de soluciones candidatas que evolucionan iterativamente para encontrar soluciones óptimas o cercanas a óptimas a través de operadores de selección, cruce y mutación.

El operador de selección, selecciona las soluciones más prometedoras (individuos) para reproducirse. El operador de cruce (o recombinación) combina dos soluciones (padres) para generar nuevas (hijos) y el operador de mutación, introduce pequeñas modificaciones aleatorias en las soluciones para mantener la diversidad y explorar nuevas áreas del espacio de búsqueda.

Este método tiene la capacidad para manejar espacios de búsqueda grandes y complejos, por lo tanto, cuenta con alta diversidad en las soluciones generadas. Sin

embargo, tiene lenta convergencia en comparación con otros métodos y alta sensibilidad a la parametrización (tamaño de población, tasas de cruce y mutación).

Es ampliamente usado en la solución de problemas combinatorios (problema del agente viajero o Travelling Salesman Problem TSP), diseño de rutas, corte y empaque; en problemas de optimización multiobjetivo (múltiples criterios: minimización de costos y maximización de calidad) y en problemas de ingeniería como el ajuste de parámetros y diseño de sistemas.

Los inspirados en el comportamiento Social:

- *Optimización por Enjambre de Partículas (Particle Swarm Optimization PSO)*: Introducida por (Kennedy y Eberhart 1995), se inspira en el comportamiento social de los enjambres; por ejemplo, como las bandadas de pájaros o cardúmenes de peces.

Cada partícula (solución) ajusta su posición en función de su experiencia y la del grupo. Es de rápida convergencia y fácil implementación, aunque puede quedar atrapada en óptimos locales si la diversidad poblacional disminuye (Clerc y Kennedy 2002).

Se caracteriza porque cada partícula tiene una posición (o solución candidata) que "vuela" por el espacio de búsqueda a una velocidad. Esta velocidad depende de la experiencia individual, es decir la mejor posición que alcanza la partícula o su influencia social a partir de la mejor posición alcanzada por el grupo. La posición de cada partícula se actualiza sumando su velocidad a su posición actual.

Entre sus ventajas es de fácil implementación, baja complejidad computacional y rápida convergencia en problemas continuos. Sin embargo, puede quedarse atrapada en óptimos locales si no se ajustan bien los parámetros y tiene menor diversidad en comparación con otras metaheurísticas basadas en población. Es aplicado en problemas de optimización continua, diseño de Ingeniería y optimización multiobjetivo.

- *Optimización Basada en Colonias de Hormigas (Ant Colony Optimization ACO)*: Propuesto por (Dorigo, Maniezzo, y Coloni 1996), el algoritmo se basa en el comportamiento de las hormigas al buscar comida; estas depositan feromonas en su camino, la concentración de las feromonas guía a otras hormigas e influyen en las decisiones de las siguientes hacia las rutas más prometedoras. Las feromonas disminuyen con el tiempo para evitar que el sistema se estanque en soluciones subóptimas.

Su ventaja radica en la capacidad de explorar múltiples soluciones simultáneamente, aunque la parametrización (intensidad y evaporación de feromonas) es crítica y su costo computacional puede ser alto (Dorigo y Stützle 2004).

Por supuesto cada hormiga elige su camino con base en una probabilidad que combina la intensidad de la feromona y el costo o calidad de la solución asociada a esa ruta. Este método tiene capacidad de explorar múltiples soluciones simultáneamente. Sin embargo, requiere ajuste cuidadoso de parámetros (intensidad

y evaporación de feromonas) y su complejidad computacional es elevada en problemas magnitud.

Se emplea en la solución de problemas de optimización de rutas (enrutamiento como TSP y el VRP o Vehicle Routing Problem). Problemas de flujo de redes (diseño de redes de telecomunicaciones y logística) y problemas Combinatorios (programación de tareas y diseño de circuitos).

Tabla 4. Comparación General de Metaheurísticas Basadas en Población.

Metaheurística	Aplicaciones	Ventajas	Desventajas
GA Algoritmos Genéticos	Problemas combinatorios y multiobjetivo	Alta diversidad, soluciones globales	Sensible a parámetros, lenta convergencia
PSO Optimización por Enjambre de Partículas	Optimización continua, diseño de sistemas	Fácil implementación, rápida en problemas continuos	Menor diversidad, atrapamiento en óptimos locales
ACO Colonia de Hormigas	Enrutamiento y Logística	Eficiente en problemas combinatorios	Complejidad computacional elevada, ajuste complejo

Fuente: Elaboración Propia.

Y los inspiradas en Fenómenos Físicos: como la Optimización Armónica, basada en el proceso de ajuste de tonos en una orquesta y la Optimización Gravitacional que modela la interacción gravitacional entre partículas.

Metaheurísticas Constructivas:

Son heurísticas que generan soluciones iniciales rápidas y viables utilizando reglas simples para construir patrones de corte. Estas soluciones no garantizan el óptimo y aunque pueden dejar espacios ineficientes o no explorar todo el espacio de soluciones, son fáciles de implementar y muy efectivas para problemas de gran escala donde los enfoques exactos son inviables; usualmente se emplean como puntos de partida para metaheurísticas más complejas. Aunque no garantizan la optimalidad, son rápidas y útiles como punto de partida para métodos más avanzados (Burke et al. 2013).

Algunos ejemplos comunes son:

- *Primer Ajuste (First Fit FF):* Asigna cada pieza al primer contenedor (bin o barra) disponible donde pueda encajar según el orden preestablecido sin intentar optimizar la colocación de la pieza más allá de esta regla básica. Aunque es simple y rápida generalmente no genera soluciones óptimas. Usual en aplicaciones de problemas de empaque y de corte como el 2DCSP.
- *Mejor Ajuste (Best Fit BF):* Ubica cada pieza en el contenedor que tenga el menor espacio sobrante; por lo tanto, busca minimizar el espacio desperdiciado al realizar cada asignación y requiere mayor tiempo de cálculo para evaluar cada contenedor antes de decidir. Por otro lado, reduce el espacio no utilizado en comparación con FF.

- *Siguiente Ajuste (Next Fit NF)*: Considera únicamente el contenedor actual y pasa al siguiente cuando no hay suficiente espacio. Pero puede ser menos eficiente y menos costoso computacionalmente.
- *Ajuste Crítico (Critical Fit CF)*: Asigna la pieza al contenedor donde su colocación genere un ajuste "crítico" o exacto, es decir, que busca contenedores en los que el espacio sobrante sea mínimo, maximizando la eficiencia en el uso del espacio disponible. Es común su aplicación en problemas donde las piezas tienen tamaños similares o restricciones estrictas de espacio.

Es una versión más refinada de BF priorizando configuraciones que minimizan el desperdicio en cada decisión.

La tabla 5 presenta una comparación entre los métodos metaheurísticos constructivos:

Tabla 5. Comparación General de Metaheurísticas Constructivas.

Metaheurística	Aplicaciones	Ventajas	Desventajas
FF Primer Ajuste	Asigna cada pieza al primer contenedor disponible donde encaje, sin reordenar ni evaluar otras opciones.	Muy rápida y fácil de implementar. Permite obtener soluciones iniciales viables con bajo costo computacional.	No garantiza soluciones óptimas; puede dejar espacios ineficientes debido a la falta de evaluación global.
BF Mejor Ajuste	Coloca cada pieza en el contenedor con el menor espacio sobrante disponible, evaluando todas las opciones antes de decidir.	Reduce el espacio desperdiciado respecto a FF. Mejora la utilización del material en la mayoría de los casos.	Requiere más tiempo de cálculo; puede generar soluciones localmente óptimas, pero no globalmente eficientes.
NF Siguiente Ajuste	Considera solo el contenedor actual y pasa al siguiente cuando no hay suficiente espacio. No reevalúa ubicaciones previas.	De muy baja complejidad y rápido procesamiento; adecuado para aplicaciones en tiempo real.	Menor eficiencia en el uso del espacio; puede incrementar el número de contenedores o láminas requeridas.
CF Ajuste Crítico	Busca el contenedor donde la pieza encaje de forma "crítica", es decir, minimizando al máximo el espacio sobrante.	Maximiza el aprovechamiento del espacio y reduce el desperdicio. Ideal cuando las piezas tienen tamaños similares.	Mayor tiempo de cómputo; puede volverse ineficiente si las piezas son de tamaños muy variados o si hay muchas opciones de ajuste.

Fuente: Elaboración Propia.

Los métodos metaheurísticos son eficientes para problemas complejos y de gran escala por su flexibilidad para adaptarse a diferentes restricciones. Pero no garantizan una solución óptima, sino aproximaciones, por lo que requiere de ajustes finos de parámetros para maximizar su rendimiento.

A pesar de los avances en técnicas de optimización se presentan varias limitaciones:

- Crecimiento exponencial de la complejidad: A medida que aumenta el tamaño del problema, el número de combinaciones posibles crece rápidamente, dificultando su resolución exacta.
- Restricciones prácticas: Factores como errores en la medición, tolerancias de corte y desgaste de herramientas no siempre son fáciles de modelar.
- Desperdicio residual: Aunque se minimiza el desperdicio, es difícil eliminarlo completamente debido a la geometría irregular de las piezas.
- Costo computacional: Las metaheurísticas requieren recursos significativos para ajustar parámetros y garantizar resultados aceptables.
- Adaptación a demandas cambiantes: En contextos dinámicos, como la personalización de productos, algunos métodos carecen de flexibilidad para adaptarse rápidamente.

En relación con el problema de corte 2DCSP, si bien los métodos clásicos ofrecen soluciones óptimas en problemas pequeños, las metaheurísticas destacan en instancias más grandes y complejas siendo altamente efectivos para resolver versiones complejas y de gran escala gracias a su flexibilidad, adaptabilidad y capacidad para aproximarse a soluciones óptimas en tiempos razonables.

Sin embargo, aún persisten desafíos relacionados con la escalabilidad, las restricciones prácticas, la integración de técnicas avanzadas en entornos industriales y el ajuste de parámetros, el costo computacional y la dificultad para incorporar restricciones industriales realistas (Crainic, Perboli, y Tadei 2009) (Eiben y Smith 2015). Estos aspectos subrayan la necesidad de seguir desarrollando enfoques innovadores y adaptativos que combinen las fortalezas de los métodos exactos y metaheurísticos y que permitan maximizar el aprovechamiento de los recursos y mejorar la sostenibilidad en los procesos productivos.

5.2 Revisión de literatura

En el sector metalmecánico, uno de los principales desafíos consiste en minimizar el desperdicio de material durante los procesos de producción. Este objetivo requiere considerar diversos aspectos, como las características y dimensiones del material, la demanda, los patrones de corte, el espacio de almacenamiento y las restricciones operativas a nivel industrial (Peña, Orejuela, y Gil 2017). En esta tesis, se aborda el problema de corte de láminas metálica, ampliamente conocido en la literatura y que ha despertado un notable interés en la comunidad académica e industrial, al ser clasificado como un problema de optimización tipo NP-Hard (Dyckhoff 1990)(Gilmore y Gomory 1963) (Kantorovich 1960) (Wäscher, Haußner, y Schumann 2007).

En la literatura, el *CSP (Cut Stock Problem)*, se define como el problema que generar un conjunto de patrones que determinen los cortes de ítems más pequeños, reduciendo pérdidas (Cui, Zhong, y Yao 2015), cumpliendo con una demanda determinada (Cheng, Feiring, y Cheng 1994).

(Gilmore y Gomory 1961) son los primeros en proponer un procedimiento de solución para el CSP unidimensional, donde determinan los posibles patrones de corte para elementos que diferían en anchura. Posteriormente ampliaron sus estudios hacia los problemas bidimensionales, abordados como un problema unidimensional de dos etapas. Se propone una formulación por programación entera en combinación de un algoritmo de generación de columnas, siendo cada

columna un posible patrón de corte (Gilmore y Gomory 1963). Al ser el problema de patrones de corte de gran dificultad, por sus características y variables, como por las técnicas de enfoque de solución, se permite la exploración de nuevas alternativas, el desarrollo de diversos modelos y métodos, (Cherri, Arenales, y Yanasse 2009). (Dyckhoff 1990) profundiza en la tipología de los problemas de corte y empaquetamiento para comprender las diferentes variantes y características del problema, proporcionando un marco de referencia amplio con un enfoque sistemático presentando una solución integral para la tipología.

Posteriormente se ampliaron estas taxonomías en los trabajos de (Wäscher, Haußner, y Schumann 2007) (Desrosiers y Lübbecke 2005), quienes documentaron generalizaciones más amplias y completas. Así mismo, (Yuen y Richardson 1995) extendieron su aplicación al *Strip Packing Problem* (SPP), considerado una tipología de dimensión abierta relacionada con el CSP.

Por su parte (Oliveira et al. 2016) a partir de la concepción de (Wäscher, Haußner, y Schumann 2007) discriminan y clasifican de manera amplia el problema de corte, especificándolo: por geometría (rectangular, irregular, redonda y esférica), por dimensión (unidimensional, bidimensional y tridimensional), por ortogonalidad (ortogonal, no ortogonal); por tipo de corte (guillotina, no guillotina), por rotación (orientado con o sin rotación) y una subclasificación según el embalaje de las tiras: (orientado guillotizable, orientado no guillotizable, sin rotación guillotizable y sin rotación no guillotizable).

La amplia caracterización y variantes permiten encontrar todo tipo de enfoques. Como por ejemplo enfoques restrictivos de modelamientos más comunes cuya pretensión está en limitar el número de patrones de corte diferentes. (Umetani, Yagiura, y Ibaraki 2003), por una parte, minimiza el número de los diferentes patrones de corte. (Imahori et al. 2005) restringiendo el uso de patrones diferentes al incluir un costo fijo por patrón en la función objetivo (Cui, Zhong, y Yao 2015); (R. Haessler 1975); (Mobasher y Ekici 2013).

El trabajo de (Furini, Malaguti, y Thomopulos 2016), concentra su atención en el análisis del problema bidimensional de la mochila 2DKSP, así extiende su aplicación al problema 2DCSP y 2DSPP, para minimizar el ancho de la tira empleada, mediante el modelamiento de restricciones en problemas de corte bidimensional, por programación entera.

Sin embargo, (Peña, Orejuela, y Gil 2017) en su compilación de enfoques en relación al CSP, aborda el significado de la complejidad de este tipo de problemas a nivel industrial, analiza investigaciones con diferentes planteamientos según el tipo de dimensión (unidimensional y bidimensional) y diversos enfoques desde métodos exactos, heurísticos y metaheurísticos, pero aún sin especificar para este tipo de problemas un método global de solución a este tipo de problemas por la complejidad ligada.

Por su parte (Morillo-Torres et al. 2021), utiliza programación lineal entera mixta para abordar el problema de corte en la industria del acero, se considera el contexto de la industria metalmecánica (Saldías et al. 2022). Mientras el trabajo clásico de (Kantorovich 1960), sienta las bases de la programación matemática aplicada a la planificación de la producción, fundamento teórico esencial para el modelamiento inicial y programación de la función objetivo y las restricciones del presente estudio.

Como un aspecto importante de la planificación de la producción en el contexto de CSP, es la secuenciación bajo dos objetivos similares. (Dyson y Gregory 1974) (Yuen y Richardson 1995) persiguen minimizar la cola de pedidos parcialmente cortados. Lo anterior, se debe a cada orden

completada de manera parcial, se generan discrepancia en la producción. Por lo tanto, su objetivo es completar un subconjunto de pedidos de igual dimensión iniciando el plan de producción para mantener minimizadas las solicitudes. Sin embargo, ninguno de ellos considera los tiempos de configuración dependientes de la secuencia de los trabajos.

Otras aplicaciones como (Jaramillo 2008) retoman la combinación de programación lineal y algoritmos genéticos para resolver problemas de corte, donde se minimiza los desperdicios, los cambios de patrones y la cantidad procesada. De esta manera, se valida que combinar estrategias presenta un desempeño superior respecto su uso individual, permitiendo sobrepasar la dificultad que asocia el aumento del tamaño del problema.

En el ámbito de las metaheurísticas, la investigación de (Kokten y Sel 2022) destaca por su enfoque en un problema de corte en la industria de productos de madera y su propuesta de dos etapas de la solución. Por otro lado, (Ülkü et al. 2022) abordan el problema de corte para la industria del acero utilizando algoritmos evolutivos.

La literatura también ha evidenciado problemas de corte desde perspectivas más específicas. Por ejemplo, el trabajo de (Scheithauer 1992) centra su atención en algoritmos para el problema de carga de contenedores *BPP (Bin Packing Problem)*, donde presenta técnicas de corte aplicadas en un contexto particular. Por otro lado, la investigación de (Arai y Haraguchi 2021) involucra formulaciones de Ising para reducir el costo de configuración en el problema de corte bidimensional.

Al revisar los modelos aproximados se encuentran los algoritmos basados en planos de corte multietapa. (Benjaoran y Bhokha 2014) desarrollan procedimientos de solución al generar patrones de corte con mínimo desperdicio de las existencias utilizadas. Estos procedimientos consisten en tres etapas: la generación de un conjunto de ajuste de los patrones de corte; la optimización del CSP unidimensional y el corte de los ítems de baja rotación, empleando el mejor ajuste a través de un algoritmo de reducción.

(Gracia 2010), presentó un enfoque híbrido basado en el uso de algoritmos genéticos para resolver el problema del corte de vigas estructurales que surgen en una empresa metalúrgica en España, dicha solución pertenece a la clase de CSP unidimensional para tamaños múltiples. En este orden de ideas, al tomar de referencia el análisis del algoritmo del plano secante o método de los planos del corte especificado de (Gilmore y Gomory 1961); el cual consiste en la introducción de una cantidad de posibles soluciones lineales del problema, de forma que todos sus puntos extremos lleguen a ser enteros. El método inicia con la resolución del problema de manera lineal al proponer un plano que corte la parte fraccional de la solución lineal, asegura que todos los puntos enteros factibles cumplan. Siendo el plano de corte una restricción válida deducida del conjunto de restricciones del problema. El algoritmo finaliza al determinar una solución al problema o en su defecto confirmando que tanto el problema no es factible y no es posible definir algún plano de corte.

Por su parte (Bouaine, Lebbar, y Ha 2018), analizan un caso de estudio, resuelto en base a dos etapas heurísticas. La primera etapa consiste la generación de patrones al resolver un problema de empaquetamiento bidimensional en contenedores tratando el problema *SPP (Strip Packing Problem)*, mediante un algoritmo que categoriza los ítems en orden descendente por niveles. La segunda etapa, este problema se convierte en un problema *BPP* unidimensional donde cada nivel es considerado como un nuevo ítem y mediante estrategia *FBF (Finite Best Fit)* se proceden a

ubicar los ítems en el nivel de menor espacio siempre desde la base inferior del contenedor conservando el orden de la primera etapa.

Otros trabajos como (Wuttke y Heese 2018), refieren desarrollos del 2DCSP, establecieron una heurística secuencial, con retroalimentación del lazo según enfoque de (Gilmore y Gomory 1963), formulando secuenciación con un programa enteros mixtos; desarrollando un límite inferior a través de una heurística de desempeño casi óptima.

Por otro lado, (Mellouli y Dammak 2008), establecieron una heurística de tres pasos: En el primer estadio se genera una enumeración de patrones factibles con un procedimiento de patrones, cuyo objetivo es construir la matriz de restricciones. Posteriormente, ejecutan una relajación de las restricciones del problema para obtener una formulación lineal. Finalmente, generan una solución del problema básico a través de la asociación del problema con las restricciones relacionadas.

Profundizando en las aplicaciones de (Kröger 1995) presenta un algoritmo genético secuencial, paralelo, capaz de generar esquemas de empaquetamiento casi óptimo con resultados superiores a enfoques de búsqueda aleatoria o recocido simulado. De acuerdo con la estructura empleada en la codificación permitiendo manipular soluciones existentes, satisfaciendo demandas. Estructura que retoma el árbol de corte como esquema de empaquetado guillotizable para *GBPP* (*Guillotinable Bin Packing Problem*).

(Tilane y Viaud 2018) describe un problema de corte, de láminas de vidrio cuyo objetivo es reducir al máximo las pérdidas de vidrio en el proceso de corte. Se define el concepto de representación de un patrón de corte a través de un árbol, debido a que un corte guillotina siempre divide un gran rectángulo inicial en dos más pequeños, por semejanza la raíz del árbol corresponde al rectángulo inicial, las hojas a los rectángulos más pequeños, o en su defecto al residual en caso de no llegar a emplearse. Entre tanto, (Toro, Garcés, y Ruiz 2008) relacionan una codificación de árbol binario completo con tres niveles, que delimita y reduce el número de árboles diferentes en el proceso de optimización, con el fin de minimizar desperdicio a partir de la obtención de los patrones de corte. (Alvarez 2010), quien fija la restricción de (Toro, Garcés, y Ruiz 2008) para los problemas de SPP, CSP y BPP en su estudio expone un tipo de codificación que asegura el cumplimiento de las restricciones asociadas al corte tipo guillotina que separa el árbol de cortes de (Toro, Garcés, y Ruiz 2008) en dos (un árbol de orientación y un árbol de distancias de cortes), y como el tamaño del espacio de soluciones del árbol de orientación de cortes es reducido respecto al de distancias, establecen una búsqueda exhaustiva en el primer árbol, mientras el segundo árbol representa el problema de optimización a resolver.

(Toro, Garcés, y Ruiz 2008) presenta un algoritmo de tres etapas (vecindad variable, recocido simulado y de técnicas constructivas), cada uno con propósitos individuales; siendo el primer algoritmo la inspiración principal de solución metaheurística del presente estudio y como algoritmo constructivo final los fundamentos de la propuesta de (Cao y Kotov 2011) cuya propuesta consiste en un algoritmo heurístico de mejor ajuste (Best Fit) para la tipología 2DBPP, bajo la estrategia de la esquina cóncava y el mejor valor de aptitud que es confrontado con otros algoritmos clásicos (métodos exactos, heurísticos y metaheurístico de búsqueda tabú); generando resultados confiables en poco tiempo para el enfoque propuesto.

La revisión de la literatura evidencia una evolución significativa en el estudio del problema de corte y empaquetamiento, tanto en el desarrollo de sus formulaciones teóricas como en las aplicaciones prácticas que han permitido abordar sus múltiples variantes. Los enfoques clásicos basados en modelos exactos, como el MILP, continúan siendo una herramienta fundamental para

la formulación del 2DCSP, ya que proporcionan soluciones óptimas en instancias de pequeña escala y sirven como referencia para evaluar el desempeño de métodos aproximados. Sin embargo, diversos autores coinciden en que la aplicabilidad de estos enfoques disminuye conforme aumenta la dimensión del problema, debido al crecimiento exponencial de la complejidad computacional.

En este contexto, y considerando la magnitud y características del problema real analizado en la empresa metalmecánica de referencia, se decidió emplear el modelo MILP únicamente para resolver instancias reducidas y como punto de comparación metodológica. Para instancias de mayor escala, se desarrolló una heurística específica, reconociendo que las heurísticas y metaheurísticas contemporáneas representan una alternativa más viable para problemas de corte bidimensional, al equilibrar precisión y eficiencia computacional.

La metaheurística propuesta se basa en un proceso integrado que combina Encoding, Decoding y una heurística de búsqueda basada en VNS. La codificación por ordenamiento secuencial mediante un vector de permutaciones; luego, el Decoding a partir una regla constructiva Bottom Left transforma el vector de permutaciones en un patrón de corte factible bajo restricciones guillotina. A partir de esta solución inicial, la heurística basada en VNS explora vecindarios mediante permutaciones controladas del orden de piezas, generando nuevas configuraciones que se evalúan y mejoran iterativamente. Esta integración permite equilibrar rapidez constructiva, capacidad exploratoria y calidad final en los patrones de corte.

En la Tabla 6 se sintetizan los aportes más representativos identificados en la revisión, organizados por tipología del problema 2DCSP, 2DSPP, 2DBPP, 2DKP, 2IDCSP, TL y C&P. Las variantes de corte: *GC* (*Guillotine Cuts*) corte guillotina, *OR* (*Orthogonal Rotation*) rotación ortogonal, *WOR* (*Without Orthogonal Rotation*) sin rotación ortogonal y *VS* (*Variable Sized Bins*) contenedor de tamaño variable. Y los principales enfoque o estrategia de solución a través de algunos métodos exactos en las columnas: MILP, *CG* (*Column Generation*) columna de generación, métodos heurísticos (constructivos y de mejora) y metaheurísticas: *PSO* (*Particle Swarm optimization*) u optimización cumulo de partículas, *SA* (*Simulated Annealing*) recocido simulado, *VNS*, *FBF* (*Finite Best Fit*) mejor ajuste finito, *BF* (*Best Fit*) mejor ajuste, *FF* (*First Fit*) primer ajuste, *CF* (*Critical Fit*) ajuste crítico, *LS* (*Local Search*) búsqueda local, *GA* (*Genetic Algorithm*) algoritmo genético.

Tabla 6. Artículos Representativos de la Revisión Bibliográfica del Problema de Corte Bidimensional.

Referencia	Tipología							Variantes			Enfoque de Solución											
	2DCSP	2DSPP	2DBPP	2DKP	2IDCSP	TL	C&P	GC	OR	VS	MILP	PSO	SA	VNS	FBF	BF	FF	CF	LS	GA	CG	
(Alvarez 2010)	x	x	x					x				x	x	x								
[(Bouaine, Lebar, y Ha 2018)	x		x					x		x						x						
(Cao y Kotov 2011)			x													x						
(Fleszar 2013)			x					x	x							x	x	x				
(Furini 2016)	x	x		x				x			x											
(Imahori et al. 2005)	x							x												x		
(Jaramillo 2008)		x			x	x	x	x													x	
(Mellouli y Dammak 2008)	x							x	x													x

Fuente: Elaboración Propia.

En síntesis, la revisión bibliográfica evidencia que el problema de corte y empaquetamiento ha evolucionado hacia una amplia diversidad de formulaciones y métodos de solución, ajustados a las condiciones geométricas, tecnológicas y computacionales de cada aplicación. Si bien los enfoques exactos continúan siendo una referencia teórica importante, su aplicabilidad práctica se ve restringida ante la creciente complejidad combinatoria de los problemas bidimensionales. En contraste, los métodos aproximados (heurísticas y metaheurísticas) al ofrecer soluciones de buena calidad con un costo computacional razonable, se consolidan como alternativas más eficientes y adaptables para entornos industriales reales.

Este análisis refuerza la pertinencia del enfoque adoptado en el presente estudio, centrado en el problema 2DCSP con cortes tipo guillotina, y su solución a través estrategias heurísticas permite equilibrar precisión, eficiencia y aprovechamiento del material en los procesos de manufactura metálica.

A partir de la base conceptual y del análisis comparativo entre los diferentes enfoques de solución, en el Capítulo 6 presenta el desarrollo la formulación matemática del problema, las estrategias de solución implementadas y las particularidades del contexto industrial que orientan su aplicación, estableciendo así el marco metodológico que sustenta la propuesta de investigación.

6 METODOLOGÍA DE LA INVESTIGACIÓN

La presente investigación se fundamenta en el diseño y evaluación de una estrategia de solución para el problema de corte bidimensional con restricciones de tipo guillotina. Un desafío ampliamente reconocido en el ámbito de la optimización combinatoria y de gran relevancia práctica en sectores industriales como el metalmecánico. Para abordar esta problemática, se propone una metodología que incluye enfoques exactos y aproximados, permitiendo no solo formular un modelo matemático riguroso, sino también explorar soluciones más eficientes y aplicables en contextos reales donde las limitaciones de tiempo y recursos hacen inviable recurrir únicamente a métodos exactos.

Primero, se formuló el modelo matemático del problema, detallando las variables, parámetros, restricciones y función objetivo que caracterizan el corte guillotina bidimensional. Dicho modelo fue implementado mediante programación entera y resuelto con el solver CPLEX, con el propósito de tener un resultado de referencia. Posteriormente, se desarrolló una propuesta de solución aproximada mediante técnicas metaheurísticas. Para ello, se diseñó una codificación, decodificación y la construcción de una heurística basada en VNS y posteriormente la validez práctica de este enfoque fue evaluada a través de un caso de estudio aplicado en una empresa local del sector metalmecánico, lo que aporta evidencia sobre su aplicabilidad en escenarios reales de producción. Finalmente, la metodología contempla un proceso de validación comparativa entre el enfoque exacto y el aproximado. Empleando instancias simuladas, para contrastar los resultados obtenidos en términos de eficiencia, calidad de solución y tiempo computacional.

De esta manera, la metodología propuesta consolida un marco integral de análisis y experimentación, que no solo busca dar solución a un problema de optimización específico, sino también aportar lineamientos prácticos para la toma de decisiones en la gestión eficiente de materiales dentro de la industria metalmecánica.

La investigación se enmarca dentro de un enfoque aplicado, cuantitativo y de carácter experimental, pues busca generar una solución práctica a un problema de optimización de relevancia industrial, al tiempo contrasta el desempeño de dos distintos métodos de solución. El estudio adopta una doble estrategia: la formulación matemática exacta del problema y con el diseño e implementación de un método aproximado basado en una metaheurística. Esta metodología permite no solo establecer un modelo de referencia que capture con precisión la estructura del problema de corte bidimensional con restricciones de guillotina, sino también explorar alternativas más flexibles y eficientes para instancias de gran escala, en las que los métodos exactos presentan limitaciones computacionales. El diseño metodológico articula tres fases complementarias:

- La definición y formulación del modelo exacto a través de programación entera.
- El desarrollo de un esquema de codificación, decodificación para incorporarlos en el diseño de una heurística para implementar en un caso real.
- La validación comparativa entre los dos enfoques, a partir de instancias de referencia y un análisis estadístico del desempeño.

6.1 Formulación del Modelo Matemático Exacto

La primera fase de la metodología se centra en la construcción de un modelo matemático que describe el problema de corte bidimensional bajo restricciones de guillotina. Este modelo

incorpora las variables de decisión necesarias para representar los cortes a realizar, los patrones de aprovechamiento y la asignación de materiales a las piezas demandadas. Asimismo, se definen los parámetros que representan las dimensiones de los materiales disponibles, las cantidades requeridas y los límites asociados al proceso productivo.

La función objetivo del modelo se orienta a maximizar el aprovechamiento del material disponible, equivalente a minimizar el desperdicio generado en el proceso de corte. Para ello, se incluyen restricciones que garantizan la viabilidad de las soluciones: cortes obligatoriamente en secuencia guillotina, satisfacción de la demanda, conservación de la capacidad de las placas e integridad de las variables de decisión.

A continuación, descripción del modelo matemático y su composición (conjuntos, parámetros, variables, función objetivo y restricciones).

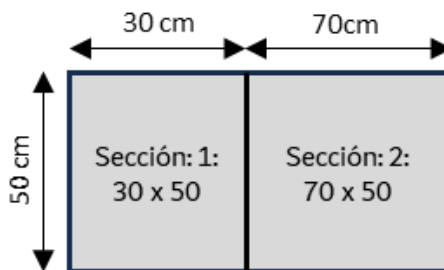
Conjuntos:

- J , Conjunto de secciones creadas a partir de los cortes realizados sobre una lámina. En un problema de corte bidimensional, cada vez que una lámina es dividida por un corte, se crean nuevas secciones. El conjunto J representa todas esas secciones que quedan disponibles para seguir realizando cortes. Cada sección es, por tanto, una región rectangular factible para insertar nuevos ítems.

Un ejemplo ilustrativo: Se tiene una lámina de $100\text{ cm} \times 50\text{ cm}$. Al realizar un vertical en $q = 30\text{ cm}$; se generan dos secciones: Sección 1: 30×50 y Sección 2: 70×50 . Ambos entran al conjunto J .

La figura 5 relaciona la representación gráfica del ejemplo en mención.

Figura 5 Generación de Secciones a Partir de Cortes Guillotina en el Modelo Bidimensional



Fuente: Elaboración Propia.

- JJ , Subconjunto de J que contiene los ítems que deben ser cortados para resolver el problema (piezas demandadas que el cliente requiere). $J \subset JJ$ Aunque todas las secciones pertenecen a J , solo algunas corresponden a ítems del problema.

Este subconjunto distingue los “espacios rectangulares útiles” del objetivo de producción.

Un ejemplo: se requiere producir secciones de dos tipos:

Tipo 1: 20×15

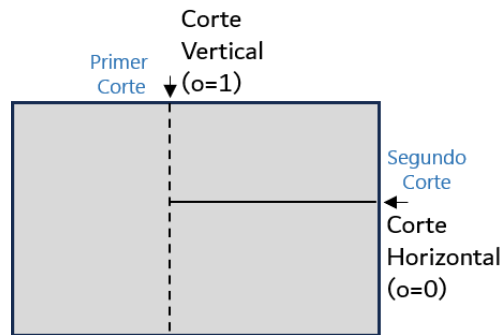
Tipo 2: 40×10

Esos ítems forman parte de J , aunque existan muchas otras secciones generadas durante los cortes.

- O , Conjunto binario que representa la orientación de los cortes que serán realizados; tomado el valor de 1 si el corte es vertical y 0 si el corte es horizontal. La orientación es crucial porque determina la geometría de las secciones resultantes.

La Figura 6 establece la representación de un corte vertical y uno horizontal.

Figura 6 Representación de la Orientación de los Cortes Vertical y Horizontal



Fuente: elaboración propia.

- Q_{jo} , Conjunto indexado de posiciones en las que la sección $j \in J$ puede ser cortada con orientación $o \in O$. Cada sección tiene posiciones continuas donde el corte podría realizarse, sin embargo, en la formulación matemática se discretiza para reducir la complejidad. Así, cada posición q representa una coordenada válida para cortar.

Ejemplo numérico: La sección j : 70×50 .

Si $o = 1$ (Vertical): Posiciones posibles.

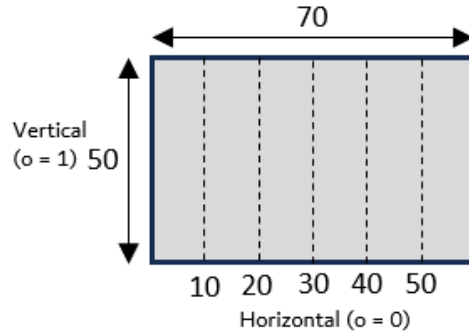
$$Q_{j1} = \{10,20,30,40,50,60\}.$$

Si $o = 0$ (Horizontal)

$$Q_{j0} = \{10,20,30,40\}$$

La figura 7 muestra la disposición grafica del ejemplo en mención.

Figura 7 Representación Gráfica del Conjunto Q_{jo} de Posiciones Discretas para la Aplicación de Cortes en una Sección j .



Fuente: Elaboración Propia

Parámetros:

- a_{kjq}^o Coeficiente o parámetro binario que representa la cantidad de ítems del tipo j (piezas) $j \in J$ generados mediante un corte en la posición $q \in Q$, con orientación $o \in O$ de un ítem (secciones) tipo $k \in K$. Este parámetro codifica la relación geométrica entre secciones e ítems posibles.

Si una pieza es 40×50 y el corte vertical en $q = 20$ genera una pieza de 20×50 que coincide con el tamaño del ítem k , entonces:

$$a_{kjq}^o = 1$$

- $PlatoOri$, Cantidad máxima de platos originales disponibles.
- Dem_j , Demanda requerida del ítem $j \in J$.
- $Area_j$, Área de cada ítem del tipo $j \in J$.

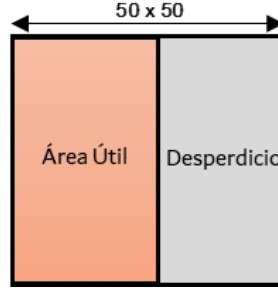
Un Ejemplo ilustrativo: Para un ítem de 20×15 su área es correspondiente a 300 cm^2 .

- $Desp_{jq}^o$, Área del desperdicio obtenido por el corte realizado en el ítem $j \in J$, con orientación $o \in O$ y posición $q \in Q$.

Por ejemplo: Un corte que divide una lámina de 50×50 puede generar dos piezas, uno de ellas se convierte en desperdicio cuando de él no es posible sacar algún ítem.

La Figura 8 presenta la generación de secciones y área desperdicio tras la ocurrencia de un corte guillotina.

Figura 8. Generación de Secciones y Área de Desperdicio Tras un Corte Guillotina.



Fuente: Elaboración Propia

Variables:

- $X_{qj}^o \in \mathbb{Z}^+ \cup 0$: Cantidad de veces que una sección $j \in J$, es cortado en la posición $q \in Q$ y orientación $o \in O$. Controla la aplicación de cortes sobre cada sección.
- $Inv_j \in \mathbb{Z}^+ \cup 0$: Cantidad de ítems $j \in JJ$ almacenados en inventario.

La incorporación del inventario en el modelo es fundamental para garantizar la continuidad operativa del sistema de corte y para preservar la coherencia entre la disponibilidad real de materia prima y las decisiones de producción. En términos operativos, la variable de inventario cumple el papel de componente auxiliar indispensable para representar los excedentes de piezas que resultan cuando los cortes generan una cantidad superior a la demanda requerida. Dado que, en el modelo, toda sección derivada de un corte debe clasificarse únicamente como ítem final o como sección utilizable para futuros cortes, se requiere un mecanismo adicional que permita contabilizar aquellos fragmentos que, aun siendo válidos geoméricamente, no pueden asignarse a la demanda. En este contexto, el inventario actúa como un contenedor formal para absorber dichos excedentes y garantizar la coherencia del balance de flujo dentro del modelo matemático.

- $Y_j \in \mathbb{Z} \geq 0$: cantidad total de ítems finales de tipo $j \in J$ producidos.

Función Objetivo:

Minimiza el desperdicio obtenido por los cortes y el área sobrante almacenada, cuantificada como $\sum_{j \in JJ} Inv_j * Area_j$, que representa las unidades no demandadas y almacenadas como ítems, también consideradas por el modelo como desperdicios.

$$\text{minimize } Z = \sum_{o \in O} \sum_{j \in J} \sum_{q \in Q_{jo}} X_{qj}^o * Desp_{jq}^o + \sum_{j \in JJ} Inv_j * Area_j \quad (1)$$

Restricciones:

El modelo se rige por un conjunto de restricciones que garantizan la coherencia en la generación, uso y demanda de piezas, bajo un principio de conservación de flujo.

- Restricción 1 (r1): Balance de ítems con demanda ($j \in JJ$)

Evalúa que la cantidad de ítems obtenidos al cortar una sección, sea equivalente a la cantidad de secciones que serán cortadas más adelante, más el total de ítems demandados que se generan.

$$\sum_{k \in J} \sum_{o \in O} \sum_{q \in Q_{jo}} a_{qkj}^o * X_{qk}^o = \sum_{q \in Q_{jo}} \sum_{o \in O} X_{qj}^o + Y_j, \quad \forall j \in JJ \quad (2)$$

El lado izquierdo cuantifica la generación total de piezas de tipo j a partir de cortes en secciones k . El lado derecho representa el consumo de esas piezas en cortes posteriores más las unidades destinadas a la producción final Y_j .

- Restricción 2 (r2): Balance de secciones intermedias ($j \in J \setminus (JJ \cup \{0\})$)

Respalda los cortes de los cuales se obtiene piezas intermedias para generar otros cortes. Verifica que la cantidad de ítems cortados corresponda a la cantidad de cortes realizados.

$$\sum_{k \in J} \sum_{o \in O} \sum_{q \in Q_{jo}} a_{qkj}^o * X_{qk}^o = \sum_{q \in Q_{jo}} \sum_{o \in O} X_{qj}^o, \quad \forall j \in J \setminus \{0\} \setminus JJ \quad (3)$$

En este caso, todas las secciones intermedias generados deben ser utilizados en cortes posteriores, debido a que no constituyen productos finales ni corresponden al plato inicial.

- Restricción 3 (r3): Relación entre producción, demanda e inventario ($j \in JJ$)

Cubre la demanda de placas proporcionada y suma el inventario excedente de los ítems finales.

$$Y_j = Dem_j + Inv_j, \quad \forall j \in JJ \quad (4)$$

De esta forma, la producción efectiva de cada ítem final Y_j debe satisfacer la demanda requerida Dem_j , mientras que cualquier excedente se contabiliza como inventario sobrante Inv_j

- Restricción 4 (r4): Límite de disponibilidad de platos originales *PlatoOri*

$$\sum_{o \in O} \sum_{q \in Q} X_{q0}^o \leq PlatoOri, \quad | PlatoOri = 1 \quad (5)$$

Esta restricción asegura que el número de cortes aplicados sobre el plato inicial $j = 0$ no exceda la disponibilidad de materia prima representada por el parámetro *PlatosOri*.

En particular, (r1) se aplica a los ítems con demanda, (r2) a las secciones intermedias; (r3) vincula la producción con la demanda y el inventario. En términos generales, las restricciones (r1) y (r3) se formulan bajo un principio de conservación de flujo de piezas: la cantidad generada en cortes debe ser equivalente a la suma de lo consumido en cortes subsecuentes y de lo producido como ítem final Y_j . Finalmente, la restricción (r4) acota el número de cortes aplicables al plato inicial, de acuerdo con la disponibilidad de materia prima (*platosOri*).

En cuanto a la implementación computacional del modelo exacto, se realizó mediante MILP, utilizando AMPL, Gurobi y CPLEX. Esta implementación permite resolver instancias simuladas de tamaño controlado con precisión, generando soluciones óptimas que servirán como referencia en la evaluación comparativa de metodologías.

6.2 Diseño de la Solución Aproximada (Heurística)

La presente investigación incorpora un enfoque heurístico diseñado para resolver de manera eficiente el problema de corte bidimensional con restricciones de tipo guillotina. La heurística combina tres componentes fundamentales: una codificación o solución inicial a partir de un vector de permutaciones, una decodificación a partir de una heurística constructiva de Bottom Left, integrados dentro de una estructura de búsqueda basada en VNS que actúa como optimizador externo. Esta arquitectura permite generar soluciones factibles de alta calidad para instancias donde el modelo exacto resulta computacionalmente infactible.

Justificación del enfoque heurístico:

La adopción de un enfoque heurístico se justifica por la naturaleza NP-Hard del problema de corte bidimensional guillotina, cuyo crecimiento combinatorio vuelve ineficientes los métodos exactos para instancias de gran escala. Aunque el modelo MILP garantiza optimalidad en casos pequeños y medianos, su desempeño se degrada significativamente cuando aumenta la cantidad de ítems y posibles posiciones de corte. Para extender la capacidad resolutoria del sistema, se incorpora una metaheurística que articula en un único proceso la generación, exploración y selección de patrones factibles, manteniendo tiempos computacionales reducidos y una calidad de solución competitiva frente al método exacto.

Esta combinación ofrece un equilibrio adecuado entre eficiencia computacional y capacidad de búsqueda, lo que resulta especialmente valioso en contextos industriales donde los tiempos de respuesta son reducidos y la disponibilidad de recursos es limitada.

A continuación, se van a introducir los componentes principales de la metaheurística: Encoding, Decoding, la función fitness de solución inicial y el algoritmo básico de búsqueda basada en VNS.

Encoding o Codificación

El Encoding corresponde al esquema mediante el cual se representa una solución candidata dentro de la metaheurística. La codificación se realiza mediante un vector de permutaciones, donde cada posición del vector indica el orden de inserción de las piezas durante el Decoding. Esta representación cumple con la característica fundamental de las permutaciones: no repetir elementos, garantizando así que cada ítem se considere exactamente una vez.

En consecuencia, el Encoding constituye la estructura lógica que define cómo se expresa una solución, mientras que la calidad final del patrón se determina posteriormente durante su decodificación y evaluación.

Un ejemplo de un vector de permutación, para una instancia con 5 tipos de ítems, identificados mediante sus índices {1,2,3,4,5}. Una solución codificada se representa mediante una permutación de estos índices, por ejemplo: $\pi = [3,1,5,2,4]$; esta permutación indica el orden exacto en el que los ítems serán procesados durante el Decoding. Por lo tanto, en este ejemplo, primero se ubica el ítem 3, luego el ítem 1, después el ítem 5; seguido del ítem 2 y finalmente el ítem 4. Cada número aparece una sola vez, lo que garantiza que el vector represente una secuencia válida sin redundancias ni repeticiones, característica fundamental del Encoding.

La Tabla 7 compila el pseudocódigo del Encoding.

Tabla 7. Pseudocódigo del Algoritmo *Encoding*.

1	ALGORITMO Encoding
2	INICIO
3	// Inicialmente, el contenedor vacío contiene un único espacio disponible
4	// con dimensiones ancho x largo de la pieza base.
5	Espacios_Disponibles <- [Contenedor_Vacio]
6	Solucion <- Inicializar()
7	
8	PARA CADA Item EN Lista_Items HACER:
9	Ubicado <- FALSO
10	
11	// Buscar el primer espacio donde el ítem quepa
12	PARA CADA Espacio EN Espacios_Disponibles HACER:
13	SI (Ítem cabe en Espacio) O (Ítem_Rotado cabe en Espacio) ENTONCES:
14	Solucion.Agregar(Ítem, Coordenadas)
15	
16	// Actualizar espacios disponibles (Estrategia de corte)
17	Nuevos_Espacios <- GenerarRecortes(Espacio, Ítem)
18	Espacios_Disponibles.Actualizar(Nuevos_Espacios)

```

19
20         Ubicado <- VERDADERO
21         ROMPER (Ciclo de Espacios)
22         FIN SI
23         FIN PARA
24
25         SI NO Ubicado ENTONCES
26             Registrar como Demanda_Faltante
27         FIN PARA
28     FIN

```

Fuente: *Elaboración Propia.*

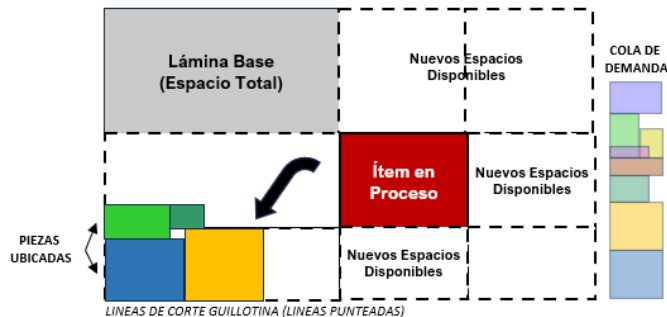
Decoding o Decodificación

El Decoding por otra parte corresponde al proceso mediante el cual la permutación codificada (Encoding) se transforma en un patrón de corte real. En esta etapa, el algoritmo interpreta el orden de los ítems definido en la lista y los ubica espacialmente en la lámina mediante la regla Bottom Left, que asigna cada pieza en la posición factible más cercana al origen preservando las restricciones guillotina, ubicando progresivamente cada ítem de la lista de demandas en la parte más hacia la izquierda y abajo posible en un espacio determinado.

De este modo, el Decoding convierte una representación abstracta del vector de permutación en una solución física y evaluable, generando los cortes, secciones residuales y el desperdicio asociado.

La figura 9, representa a grandes rasgos la estrategia de empaque del algoritmo Bottom Left.

Figura 9. Funcionamiento General del Algoritmo Bottom Left.



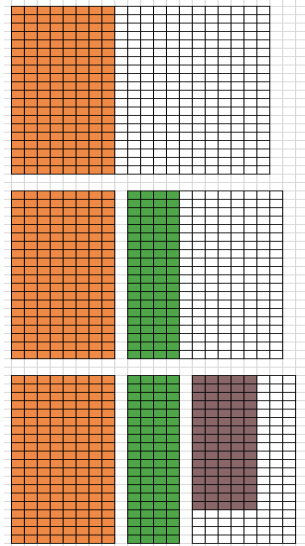
Fuente: *Elaboración Propia.*

Una vez se ha logrado ubicar un ítem utilizando la estrategia del Bottom Left, se procede con el corte guillotina sobre dicho espacio.

En el desarrollo de esta metaheurística, se decidió que el primer corte debe ser horizontal y posteriormente vertical. Cada corte a su vez va generando progresivamente nuevas secciones que se van almacenando. De esta forma, si un ítem no cabe en una sección x, el algoritmo intentará en otras secciones hasta lograr o no, ubicar finalmente el ítem.

En la figura 10, se puede ver representado un caso real de uno de los ejemplos utilizados en los resultados en Excel.

Figura 10. Representación de Cortes y Generación de Nuevas Secciones en Excel.



Fuente: Elaboración Propia

La lógica del Decoder consiste sencillamente en otorgar a un caso el desperdicio y el fitness encontrado con base en la lista de demandas procesadas por el Encoder.

El desperdicio se entiende como la suma de las secciones disponibles restante posterior al posicionamiento de los ítems, en las cuales no es posible ubicar alguno de los ítems de la demanda.

De otro lado, el fitness es la suma del desperdicio más una penalización de 2 por cada demanda no satisfecha. A continuación, se presenta la ecuación de la función fitness:

$$Fitness = Z + (2 * \sum Demanda Faltante) \quad (6)$$

La función fitness constituye el mecanismo mediante el cual el algoritmo evalúa la calidad de cada solución generada durante el proceso metaheurístico (encoding, decoding y heurística basada en VNS). Su propósito es incorporar simultáneamente dos criterios fundamentales del problema de corte:

- Cumplir completamente la demanda.
- Minimizar el desperdicio de material.

El diseño de la función fuerza un comportamiento jerárquico por lo que primero garantizar la demanda, luego optimizar la eficiencia del corte.

Componente 1: Desperdicio total Z:

$$Z = \sum \text{Areas vacias en todos los patrones de corte generados}$$

Este término corresponde al costo primario del proceso de corte. Y refleja la sección no utilizada dentro de cada matriz construida durante el decoding.

La función fitness en el algoritmo:

- Mejora iterativamente las permutaciones buscando minimizar su valor (fitness).
- Es el criterio que decide si una nueva permutación vecina es mejor que la actual, si una estructura de vecindario debe explorarse y si la solución global actual debe actualizarse.
- Durante el decoding, el algoritmo ubica ítems siguiendo la regla constructiva Bottom-Left. Cada vez que queda una sección residual que no puede ser aprovechada, su área se acumula.
- Finalmente, acumula el desperdicio total como la suma de todas las secciones vacías generadas.

Componente 2: Penalización por demanda faltante

$$(2 * \sum \text{Demanda Faltante})$$

Este término introduce una penalización explícita cuando una solución no logra producir todos los ítems requeridos. Pero ¿Por qué penalizar?:

En problemas de corte, una solución puede tener:

- Poco desperdicio,
- Fallo en cumplir la demanda.

Eso es inaceptable operativamente. Por lo tanto, la penalización asegura que el algoritmo descarte cualquier solución incompleta, incluso si presenta bajo desperdicio.

El factor 2 amplifica el impacto de la demanda faltante:

- El incumplimiento de demanda es un error estructural grave.
- Debe tener mayor peso que la eficiencia del patrón de corte.
- Sin penalización fuerte, la heurística podría privilegiar soluciones con buen empaquetamiento, pero sin cubrir toda la demanda.

Este multiplicador convierte la función fitness en una función jerárquica suavizada:

- Nivel 1 (prioritario): cero demanda faltante.
- Nivel 2: minimizar desperdicio entre todas las soluciones que cumplen la demanda.

La función fitness dirige la búsqueda hacia soluciones factibles y prácticamente relevantes, favoreciendo patrones de corte compactos únicamente cuando han satisfecho previamente la totalidad de la demanda. Lo cual asegura un comportamiento robusto y alineado con las necesidades industriales, en este caso, del entorno metalmecánico.

En la Tabla 8 se compila el pseudocódigo del algoritmo Decoder.

Tabla 8. Pseudocódigo del Algoritmo Decoder.

```

1  ALGORITMO Decoder(Solucion)
2  INICIO
3    // -----
4    // 1. Cálculo del Desperdicio (z)
5    // Se recorre la representación matricial de los contenedores
6    // contando los espacios vacíos (representados por 0).
7    // -----
8    Desperdicio_Total <- 0
9
10   PARA CADA Placa EN Solucion.Matrices_Disponibles HACER:
11     PARA CADA Fila EN Placa HACER:
12       PARA CADA Celda EN Fila HACER:
13         SI Celda = 0 ENTONCES:
14           Desperdicio_Total <- Desperdicio_Total + 1
15         FIN SI
16       FIN PARA
17     FIN PARA
18   FIN PARA
19
20   Solucion.Desperdicio <- Desperdicio_Total
21
22   // -----
23   // 2. Cálculo de Penalización por Demandas Faltantes
24   // Se suman todas las cantidades de items que no se pudieron ubicar.
25   // -----
26   Suma_Faltantes <- 0
27   PARA CADA Cantidad EN Solucion.Demandas_Faltantes.Valores() HACER:
28     Suma_Faltantes <- Suma_Faltantes + Cantidad
29   FIN PARA
30
31   Solucion.Fitness <- Desperdicio_Total + (2 * Suma_Faltantes)
32 FIN

```

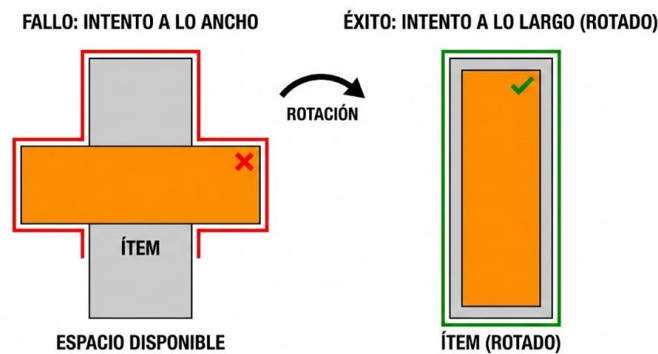
Por otra parte, la ubicación del ítem en una sección disponible depende tanto de las dimensiones del ítem como de las dimensiones de la sección disponible. Por lo que puede dar lugar a dos situaciones:

- Es un ítem cuadrado (ancho es igual a largo).
- Es un ítem rectangular (ancho no es igual a largo).

Si se trata de un ítem rectangular, se evalúa la posibilidad de ubicar el ítem inicialmente a lo ancho. Si no es posible ubicar el ítem a lo ancho, se evalúa a lo largo y si en ninguna de las disposiciones se logró ubicar el ítem, se descarta.

La figura 11, representa la acción del algoritmo al intentar ubicar un ítem rectangular en un espacio.

Figura 11. Ubicación de un Ítem Rectangular en un Espacio.



Fuente: Elaboración Propia

De este proceso surgen las siguientes aclaraciones lógicas:

- Si por algún motivo el ítem ocupa todo el ancho o todo el largo del espacio, se realiza un único corte horizontal o vertical según corresponda.
- Si el ítem ocupa todo el espacio, no se genera ningún corte.

De esta forma, el Encoding proporciona finalmente las demandas satisfechas e insatisfechas y los espacios restantes que se traducen en el desperdicio final para que posteriormente, en la lógica del Decoder, se calcule el fitness final.

Estructura General de la Heurística basada en VNS:

La lógica de la búsqueda de vecindad variable consiste en tres etapas:

1. Generación de solución inicial.
2. Exploración de vecindarios (Búsqueda Local).
3. Criterio de aceptación y parada.

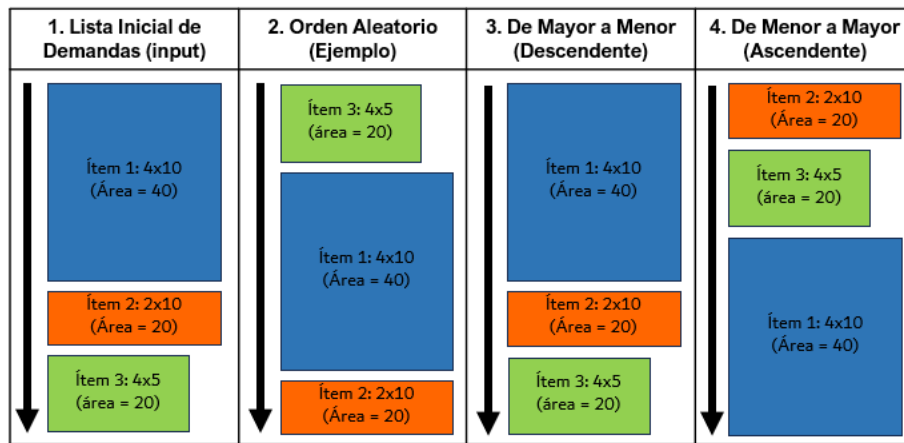
1. Generación de solución inicial:

Todo vecino/solución se genera con base en las dimensiones de la lámina grande y una lista de demandas. Para efectos de pruebas, cada caso fue sometido a tres diferentes ordenamientos de listas de demandas:

- Aleatorio.
- De mayor a menor área.
- De menor a mayor área.

La figura 12 representa un ejemplo de estos tres diferentes tipos de ordenamientos:

Figura 12. Ordenamientos de la Lista de Demandas Inicial.



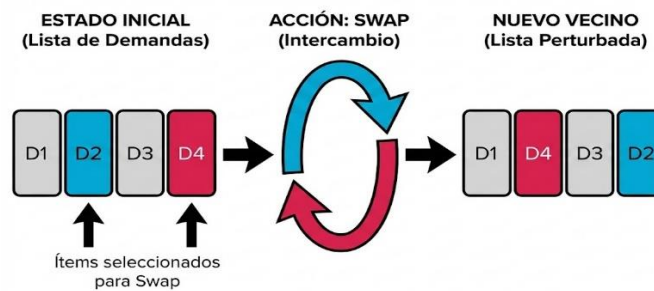
Fuente: Elaboración Propia.

2. Exploración de vecindarios (Búsqueda Local).

Se plantearon tres tipos de vecindarios fundamentados en la alteración de la lista de demandas, citados a continuación:

- Vecindario Swap: Explora el vecindario generado al intercambiar la posición de dos ítems en la lista de demandas. La figura 13 representa la generación de vecindarios swap.

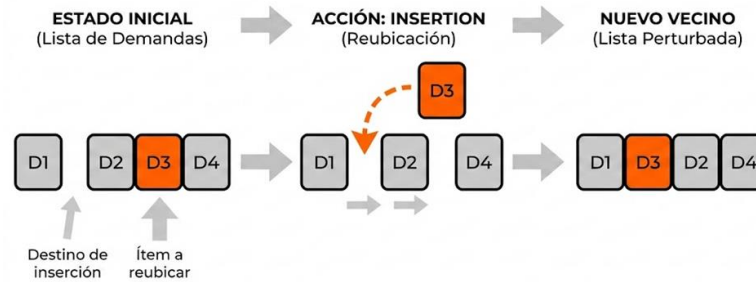
Figura 13. Representación de la Generación de Vecindarios Swap.



Fuente: Elaboración Propia.

- Vecindario Insertion: Explora el vecindario generado al tomar un ítem de la lista de demandas y reubicarlo en otra posición. La figura 14 representa la generación de vecindarios insertion.

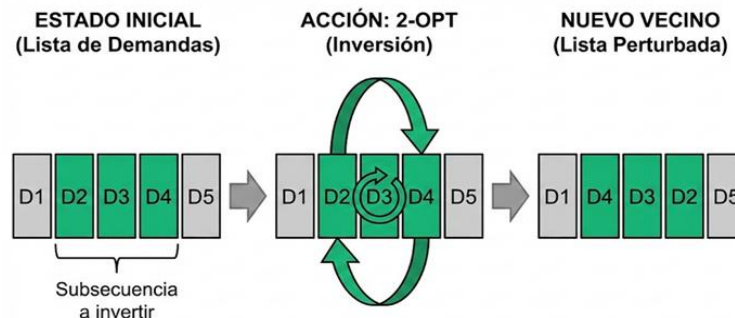
Figura 14. Representación de la Generación de Vecindarios Insertion.



Fuente: Elaboración Propia.

- Vecindario 2-Opt: Explora el vecindario generado, en el que se invierten subsecuencias de la lista de demandas. La figura 15 representa la generación de vecindarios 2-OPT.

Figura 15. Representación de la Generación de Vecindarios 2-OPT.



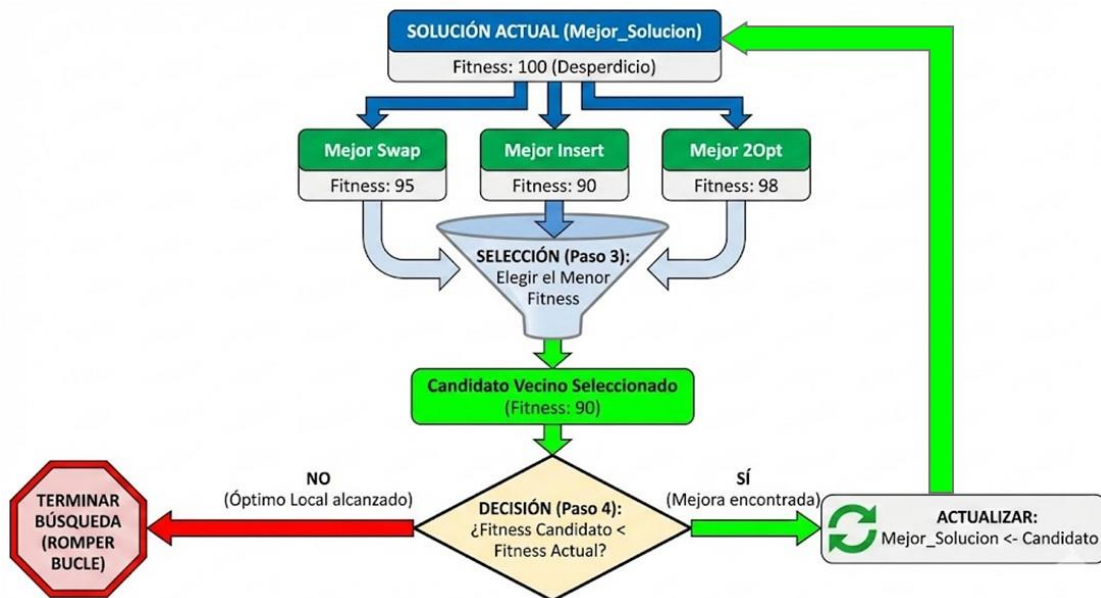
Fuente: Elaboración Propia.

El algoritmo identifica el vecino con menor fitness dentro de cada uno de estos tres vecindarios. Posteriormente, realiza una comparación global entre los tres ganadores (Mejor_Swap, Mejor_Insert, Mejor_2Opt) para seleccionar el mejor vecino global que representa la mejor transformación posible en la iteración actual.

3. Criterio de aceptación y parada.

El algoritmo verifica iterativamente si el mejor vecino obtenido tiene un fitness menor a la mejor solución. Si hay mejora, el candidato reemplaza a la solución actual y el ciclo se repite, utilizando esta nueva configuración como base para la siguiente exploración. Si no hay mejora, se determina que se ha alcanzado un Óptimo Local y se retorna esta última mejor solución encontrada. En la figura 16 se representa dicha búsqueda del óptimo local.

Figura 16. Representación de Búsqueda del Óptimo Local.



Fuente: Elaboración Propia.

La Tabla 9 compila el pseudocódigo del algoritmo VNS (Variable Neighborhood Search - Búsqueda de Vecindarios Variables).

Tabla 9. Pseudocódigo del Algoritmo VNS.

1	ALGORITMO VNS
2	INICIO
3	// 1. Inicialización
4	Solucion_Actual <- Generar_Solucion_Inicial(ancho, largo, items)
5	Mejor_Solucion <- Copiar(Solucion_Actual)
6	
7	// 2. Bucle principal de búsqueda
8	REPETIR SIEMPRE:
9	
10	// Exploración de vecindarios (Búsqueda Local)
11	// Se busca el mejor vecino dentro de cada tipo de movimiento
12	Mejor_Swap <- Explorar_Vecindario_Swap(Mejor_Solucion)
13	Mejor_Insert <- Explorar_Vecindario_Insertion(Mejor_Solucion)
14	Mejor_2Opt <- Explorar_Vecindario_2Opt(Mejor_Solucion)
15	
16	// 3. Selección del mejor movimiento global de esta iteración
17	Candidato_Vecino <- best(Mejor_Swap, Mejor_Insert, Mejor_2Opt)
18	
19	// 4. Criterio de Aceptación y Parada
20	// Verificamos si el mejor vecino encontrado mejora la solución global
21	SI (Candidato_Vecino.fitness < Mejor_Solucion.fitness) ENTONCES:

```

22      // Hubo mejora, actualizamos y continuamos el bucle
23      Mejor_Solucion <- Candidato_Vecino
24      SINO:
25      // No hubo mejora (Óptimo Local alcanzado), terminamos la búsqueda
26      ROMPER BUCLE (BREAK)
27      FIN SI
28
29      FIN REPETIR
30      RETORNAR Mejor_Solucion
31      FIN

```

Fuente: Elaboración Propia.

En este sentido, el modelo MILP desarrollado en capítulos anteriores constituye el benchmark que permite medir el grado de eficiencia, precisión y robustez del enfoque metaheurístico. La comparación entre ambas metodologías es esencial para determinar en qué escenarios la metaheurística logra soluciones cercanas al óptimo, cuánto se reduce el tiempo computacional y qué tan consistente es su desempeño ante variaciones en el número y tamaño de las piezas.

Por ello, el capítulo siguiente presenta el análisis de resultados obtenidos a partir de instancias simuladas y del caso de estudio real. Se examina la calidad de los patrones generados, el porcentaje de desperdicio alcanzado, la estabilidad de la heurística frente a cambios en las permutaciones y su relación con los valores óptimos entregados por el modelo exacto. Esta transición marca el paso natural desde el diseño metodológico hacia la validación experimental del método propuesto, permitiendo establecer sus fortalezas, limitaciones y potencial de aplicación en entornos industriales metalmeccánicos.

7. RESULTADOS

7.1 Detalle de la implementación

El algoritmo desarrollado se implementó en Python. Las pruebas computacionales se realizaron en una CPU AMD Ryzen 9 270 w/ Radeon 780M Graphics (4.00 GHz) - 32 GB de RAM y Sistema operativo Windows 11 de 64 bits.

7.2 Resultados Obtenidos

En este capítulo se presentan y analizan los resultados obtenidos a partir de las instancias generadas para la evaluación del modelo exacto y de la metaheurística propuesta, así como los casos con datos reales. El propósito es comparar el desempeño computacional y la calidad de las soluciones obtenidas mediante ambos enfoques, considerando diferentes tipos de instancias (ítems pequeños, medianos, grandes y reales). Además, se examina el comportamiento de las estrategias de ordenamiento utilizadas en la heurística y se discute su impacto en el uso del material y los tiempos de cómputo.

Generación de Instancias de prueba:

Con el propósito de evaluar comparativamente el desempeño del modelo exacto y de la heurística propuesta, se construyó un conjunto de instancias simuladas, además de las reales, que permiten analizar diferentes niveles de dificultad, características y escalabilidad del problema. Para ello, se generaron casos que combinan procedimientos de parametrización geométrica con ajustes controlados en las demandas, garantizando patrones de prueba representativos, reproducibles y comparables.

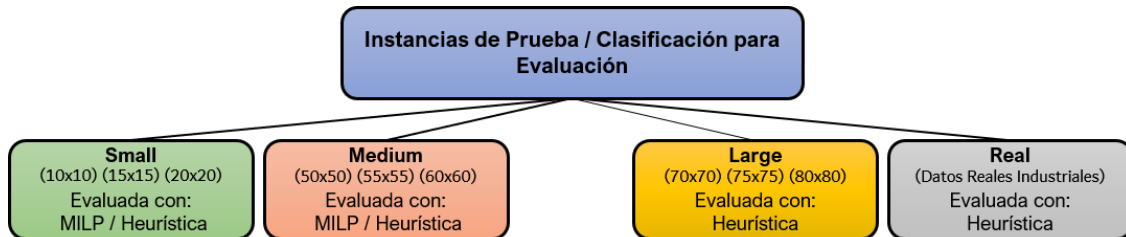
En una primera etapa, se generaron las dimensiones de los ítems (ancho y alto) y de las placas base considerando rangos típicos con base en datos reales. A partir de estas dimensiones iniciales se definieron demandas para cada ítem, representando configuraciones base de carga del material. Las instancias generadas tienen porcentajes de ocupación dentro de un rango entre 80 % y 90 %, incrementados de manera unitaria (80, 81, 82, ..., 90 %). La dificultad de resolución del 2DCSP depende principalmente de la relación entre el área total demandada y el área disponible en la lámina. Un mayor porcentaje de ocupación incrementa la restricción geométrica y, por tanto, el desafío combinatorio de encontrar patrones factibles. Se consolidó un conjunto de cuatro instancias:

- Small: ítems pequeños y de baja ocupación geométrica relativa con tamaños de pieza base: (10x10), (15x15) y (20x20).
- Medium: ítems de tamaño medio de: (50x50), (55x55) y (60x60); con mayor complejidad combinatoria.
- Large: ítems grandes, cercanos a las dimensiones del stock, lo que dificulta la generación de patrones factibles y tamaños entre: (70x70), (75x75) y (80x80).
- Real: conformada a partir de un conjunto de datos, de piezas reales, suministradas por la empresa del caso objeto de estudio.

Las instancias Small y Medium se utilizaron para evaluar tanto el MILP como la heurística, mientras que las clases Large y Real se emplearon exclusivamente para el enfoque heurístico debido a las limitaciones computacionales asociadas a los métodos exactos en problemas de gran escala.

La figura 17 presenta las cuatro categorías de instancias, sus tamaños y método de evaluación.

Figura 17. Categorización de Instancias de Prueba



Fuente: Elaboración Propia

Resultados Computacionales:

A continuación, se presenta los resultados derivados de la evaluación computacional del modelo exacto y de la heurística híbrida desarrollada para el problema de corte bidimensional con restricciones de guillotina. El análisis se estructura a partir del conjunto de instancias de prueba previamente generadas (Small, Medium, Large y Real).

Para el análisis comparativo se tuvieron en cuenta cuatro criterios:

- Factibilidad.
- Calidad de la solución.
- Eficiencia computacional, y
- Escalabilidad frente al tamaño de las instancias.

Las tablas presentan los resultados obtenidos para cada una de las categorías de instancias. Para los casos Small y Medium se muestran los resultados derivados del modelo exacto y los alcanzados por la metaheurística propuesta. Posteriormente, para los casos Large y los Reales únicamente los resultados entregados por la metaheurística.

En cuanto la nomenclatura de las tablas de resultados la primera columna detalla dimensiones de la pieza base (ancho x alto), la demanda en porcentaje de ocupación. Las seis columnas siguientes están dedicadas al reporte del modelo exacto: “AMPL Elapse time” es tiempo de lectura, compilación y ejecución total de AMPL; “Total solve elapse time” relaciona el tiempo exclusivo que toma AMPL en resolver el problema MILP; “Peso (data.dat)” registra el peso en megabytes del archivo de datos; “Tiempo de creación” en segundos equivalente al tiempo que toma ejecutar el scrip para generar los datos; “Status” toma valor de 0 a 99 cuando AMPL resuelve de forma óptima el problema, un valor distinto

puede indicar no factibilidad o límite de tiempo; “z” corresponde al valor de la función objetivo del MILP como resultado al correr AMPL.

En cuanto a la heurística la tabla reúne los resultados para tres tipos de ordenamiento inicial: Aleatorio (random), Mayor a Menor y Menor a Mayor. Generando para cada uno resultados la columna “Tiempo de solución” correspondiente al tiempo que toma solucionar la metaheurística para cada caso; “Desperdicio” que corresponde al área libre que queda en la lámina una vez ubicados los ítems, donde no es factible insertar piezas adicionales por restricciones geométricas o de corte; “Fitness” como resultado de decoding y la evaluación de la calidad de la solución a través de la función fitness en la ecuación (6); y “GAP” computado como se muestra en la ecuación (8). Para los casos Large y Real, el GAP no se calcula dado que no existe referencia exacta.

$$GAP = \left[\frac{Z_{heuristic} - Z_{exacto}}{Z_{exacto}} \right] * 100 \quad (8)$$

Donde, donde Z_{exacto} corresponde al valor óptimo proporcionado por el solver y $Z_{heuristic}$ representa el valor alcanzado por la heurística para la misma instancia. El numerador es la diferencia numérica directa entre el desempeño de la heurística y el valor óptimo. mientras que la normalización de la diferencia absoluta respecto al valor óptimo para poder comparar instancias de distinta escala.

La Tabla 10 resume los resultados para la categoría de instancia de casos pequeños (Small) para el modelo exacto (AMPL) y la heurística.

Tabla 10. Resultados Instancias Casos de Prueba Small

Caso		AMPL						Heurística									GAP
Pieza base (Ancho x Alto)	Ocupación (%)	AMPL Elapsed Time	Total Solve Elapsed Time	Peso data.dat KB	Tiempo de creación (s)	Status	Z	Random			Mayor a menor			Menor a mayor			
								Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness		
10x10	80%	0.016000	0.047000	27	0.015	0	20	0.014	20	20	0.002	20	20	0.002	20	20	0
10x10	81%	0.032000	0.031000	27	0.012	0	19	0.016	19	19	0.037	19	19	0.036	19	19	0
10x10	82%	0.047000	0.031000	27	0.012	0	18	0.010	18	18	0.002	18	18	0.001	18	18	0
10x10	83%	0.015000	0.047000	27	0.012	0	17	0.009	17	17	0.002	17	17	0.001	17	17	0
10x10	84%	0.063000	0.031000	27	0.012	0	16	0.028	16	16	0.002	16	16	0.002	16	16	0
10x10	85%	0.031000	0.031000	27	0.012	0	15	0.011	15	15	0.001	15	15	0.001	15	15	0
10x10	86%	0.032000	0.046000	27	0.013	0	14	0.014	14	14	0.004	14	14	0.006	14	14	0
10x10	87%	0.047000	0.031000	27	0.012	0	13	0.008	13	13	0.001	13	13	0.002	13	13	0
10x10	88%	0.015000	0.047000	27	0.013	0	12	0.074	12	12	0.030	12	12	0.066	12	12	0
10x10	89%	0.016000	0.047000	27	0.012	0	11	0.027	11	11	0.039	11	11	0.020	11	11	0
10x10	90%	0.032000	0.031000	27	0.013	0	10	0.010	10	10	0.003	10	10	0.002	40	42	0
15x15	80%	0.078000	0.078000	88	0.027	0	45	0.016	45	45	0.004	45	45	0.008	45	45	0
15x15	81%	0.062000	0.063000	88	0.028	0	43	0.011	43	43	0.003	43	43	0.006	43	43	0
15x15	82%	0.063000	0.094000	88	0.031	0	41	0.019	41	41	0.012	41	41	0.018	41	41	0
15x15	83%	0.094000	0.031000	88	0.030	0	38	0.012	38	38	0.001	38	38	0.002	38	38	0
15x15	84%	0.047000	0.125000	88	0.029	0	36	0.020	36	36	0.023	36	36	0.022	36	36	0
15x15	85%	0.095000	0.109000	88	0.028	0	34	0.020	34	34	0.013	34	34	0.005	34	34	0
15x15	86%	0.047000	0.109000	88	0.028	0	31	0.024	31	31	0.023	31	31	0.023	31	31	0
15x15	87%	0.078000	0.078000	88	0.029	0	30	0.079	30	30	0.068	30	30	0.069	30	30	0
15x15	88%	0.063000	0.078000	88	0.028	0	27	0.020	27	27	0.005	27	27	0.011	27	27	0
15x15	89%	0.110000	0.078000	88	0.029	0	25	0.073	25	25	0.033	25	25	0.066	25	25	0

15x15	90%	0.047000	0.062000	88	0.028	0	23	0.012	23	23	0.003	23	23	0.002	23	23	0
20x20	80%	0.249000	0.110000	211	0.075	0	80	0.010	80	80	0.002	80	80	0.003	80	80	0
20x20	81%	0.235000	0.203000	211	0.074	0	76	0.028	76	76	0.019	76	76	0.039	76	76	0
20x20	82%	0.250000	0.172000	211	0.076	0	72	0.015	152	154	0.009	72	72	0.005	152	154	0
20x20	83%	0.249000	0.094000	211	0.075	0	68	0.402	68	68	0.183	68	68	0.442	68	68	0
20x20	84%	0.235000	0.203000	211	0.076	0	64	0.020	64	64	0.042	64	64	0.026	64	64	0
20x20	85%	0.266000	0.140000	211	0.075	0	60	0.023	60	60	0.016	60	60	0.032	60	60	0
20x20	86%	0.296000	0.219000	211	0.075	0	56	0.022	56	56	0.015	56	56	0.030	56	56	0
20x20	87%	0.281000	0.188000	211	0.075	0	52	0.024	52	52	0.018	52	52	0.026	52	52	0
20x20	88%	0.282000	0.156000	211	0.076	0	48	0.282	48	48	0.094	48	48	0.185	48	48	0
20x20	89%	0.266000	0.203000	211	0.075	0	44	0.042	44	44	0.018	64	66	0.034	64	66	0
20x20	90%	0.249000	0.172000	211	0.075	0	40	0.024	40	40	0.017	40	40	0.026	40	40	0

La Tabla 11 vincula los resultados para la categoría de instancia de casos medianos (Medium) para el modelo exacto (AMPL) y la heurística.

Tabla 11. Resultados Instancias Casos de Prueba Medium

Caso		AMPL						Heurística									GAP
Pieza base (Ancho x Alto)	Ocupación (%)	AMPL Elapsed Time	Total Solve Elapsed Time	Peso data.dat KB	Tiempo de creación (s)	Status	Z	Random		Mayor a menor			Menor a mayor				
								Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness	
50x50	80%	44,1570	3,20300	3.408	4.684	0	500	0.317	500	500	0.152	500	500	0.328	500	500	0
50x50	81%	34,3910	5,68700	3.408	4.662	0	475	0.259	475	475	0.151	475	475	0.243	475	475	0
50x50	82%	35,7510	2,54600	3.408	4.684	0	450	0.448	450	450	0.143	450	450	0.289	450	450	0
50x50	83%	34,6560	3,51600	3.408	4.719	0	425	0.370	425	425	0.311	425	425	0.730	425	425	0
50x50	84%	35,6710	5,11000	3.408	4.744	0	400	0.188	400	400	0.360	400	400	0.410	400	400	0
50x50	85%	35,2800	7,01600	3.408	4.764	0	375	2.165	375	375	1.362	375	375	2.945	375	375	0
50x50	86%	35,2500	4,28100	3.408	4.734	0	350	0.494	350	350	0.527	350	350	0.875	350	350	0
50x50	87%	35,2350	5,15600	3.408	4.729	0	325	0.408	325	325	0.117	325	325	0.395	325	325	0
50x50	88%	36,3590	3,73400	3.408	4.796	0	300	0.393	300	300	0.185	410	412	0.508	300	300	0
50x50	89%	35,5470	6,59400	3.408	4.744	0	275	0.691	275	275	0.618	275	275	0.956	275	275	0
50x50	90%	35,6560	7,89100	3.408	4.799	0	250	0.625	400	402	0.581	250	250	0.915	250	250	0
55x55	80%	55,6260	8,03100	4.583	7.614	0	605	0.586	605	605	0.283	605	605	1.005	605	605	0
55x55	81%	52,5150	5,09400	4.583	7.617	0	575	0.741	575	575	0.850	575	575	1.245	575	575	0
55x55	82%	53,8280	19,25000	4.583	7.682	0	545	0.649	545	545	0.423	545	545	1.116	545	545	0
55x55	83%	54,0940	4,25000	4.583	7.628	0	514	9.561	731	733	2.947	514	514	10.621	514	514	0
55x55	84%	54,6880	10,42200	4.583	7.600	0	484	0.546	484	484	0.221	484	484	0.766	484	484	0
55x55	85%	53,1240	5,81300	4.583	7.602	0	453	0.655	653	655	0.577	453	453	1.052	453	453	0
55x55	86%	53,2500	11,14100	4.583	7.664	0	423	1.315	423	423	0.691	423	423	1.025	683	685	0
55x55	87%	53,7500	12,29700	4.583	7.682	0	393	2.775	393	393	2.737	463	465	4.532	393	393	0
55x55	88%	54,0000	4,20300	4.583	7.659	0	363	0.397	363	363	0.816	363	363	1.141	363	363	0
55x55	89%	53,3750	14,50000	4.583	7.633	0	333	12.245	333	333	8.058	333	333	23.992	333	333	0
55x55	90%	53,6720	12,75000	4.583	7.705	0	302	3.925	302	302	3.946	302	302	6.699	302	302	0

60x60	80%	88,7030	14,43800	5.995	11.738	0	720	1.849	720	720	2.996	720	720	1.852	720	720	0
60x60	81%	85,9690	13,03100	5.995	11.870	0	684	0.593	684	684	0.635	684	684	0.510	684	684	0
60x60	82%	86,9840	3,51600	5.995	11.921	0	648	0.033	648	648	0.036	648	648	0.071	648	648	0
60x60	83%	89,2190	11,25000	5.995	11.893	0	612	1.196	612	612	0.858	612	612	0.742	612	612	0
60x60	84%	86,8750	12,50000	5.995	11.885	0	576	0.576	576	576	0.524	576	576	0.528	576	576	0
60x60	85%	86,8130	6,26500	5.995	11.819	0	540	0.735	540	540	0.217	540	540	0.481	795	797	0
60x60	86%	89,4060	5,00000	5.995	11.866	0	504	1.717	504	504	1.175	504	504	1.828	504	504	0
60x60	87%	87,6100	7,90600	5.995	11.869	0	468	1.532	468	468	2.180	468	468	3.013	468	468	0
60x60	88%	89,0470	7,12500	5.995	11.828	0	432	0.739	432	432	0.488	608	610	0.724	432	432	0
60x60	89%	87,9060	20,56200	5.995	11.823	0	396	8.051	396	396	12.930	396	396	12.321	396	396	0
60x60	90%	87,9850	15,31200	5.995	11.839	0	360	1.264	360	360	0.848	360	360	1.150	360	360	0

Para los casos Large y Reales guardan el mismo formato y nomenclatura para la evaluación de la heurística. La Tabla 12 relaciona los resultados para la categoría de instancia de casos grandes (Large) para la heurística.

Tabla 12. Resultados Instancias Casos de Prueba Large

Caso		Heurística								
Pieza base (Ancho x Alto)	Ocupación (%)	Random			Mayor a menor			Menor a mayor		
		Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness
70x70	80%	0.550	980	980	0.286	980	980	0.545	980	980
70x70	81%	1.818	931	931	0.881	931	931	1.860	931	931
70x70	82%	2.184	882	882	1.988	882	882	2.354	882	882
70x70	83%	2.292	833	833	2.788	833	833	3.025	833	833
70x70	84%	0.596	784	784	0.335	784	784	0.652	784	784
70x70	85%	2.058	735	735	0.651	735	735	0.942	735	735
70x70	86%	0.821	686	686	0.397	686	686	1.417	686	686
70x70	87%	5.202	637	637	2.574	637	637	5.793	637	637
70x70	88%	3.431	966	970	3.936	588	588	5.060	966	970
70x70	89%	9.037	540	540	4.537	540	540	9.316	540	540
70x70	90%	9.290	490	490	3.278	490	490	7.073	490	490
75x75	80%	4.322	1125	1125	1.174	1125	1125	2.818	1125	1125
75x75	81%	5.199	1069	1069	4.329	1069	1069	4.213	1069	1069
75x75	82%	10.707	1013	1013	2.715	1013	1013	6.819	1013	1013
75x75	83%	5.732	957	957	3.202	957	957	3.749	957	957
75x75	84%	0.967	1758	1762	0.632	1758	1762	0.706	1758	1762
75x75	85%	0.147	2435	2437	0.038	2435	2437	0.044	2435	2437
75x75	86%	0.024	788	788	0.015	788	788	0.016	788	788
75x75	87%	0.025	732	732	0.015	732	732	0.016	732	732
75x75	88%	0.093	2322	2324	0.053	2322	2324	0.062	2322	2324
75x75	89%	5.990	619	619	1.894	619	619	4.368	619	619
75x75	90%	18.203	563	563	5.751	563	563	8.377	563	563
80x80	80%	16.046	1280	1280	4.017	1280	1280	9.445	1280	1280
80x80	81%	20.042	1216	1216	9.322	1216	1216	12.110	1216	1216
80x80	82%	2.915	1152	1152	1.620	1152	1152	2.007	1152	1152
80x80	83%	2.647	1088	1088	2.538	1088	1088	1.451	1088	1088
80x80	84%	16.668	1024	1024	3.423	1024	1024	6.181	1024	1024
80x80	85%	10.566	960	960	6.460	960	960	7.553	960	960
80x80	86%	1.189	896	896	1.441	896	896	1.839	896	896
80x80	87%	1.355	832	832	0.676	832	832	0.749	832	832
80x80	88%	7.115	768	768	2.786	768	768	2.927	768	768
80x80	89%	22.041	704	704	14.945	904	908	8.522	704	704

80x80	90%	23.793	640	640	9.529	640	640	15.552	640	640
-------	-----	--------	-----	-----	-------	-----	-----	--------	-----	-----

La Tabla 13 relaciona los resultados para la categoría de instancia de casos Reales (caso de empresa industria metalmeccánica) para la heurística.

Tabla 13. Resultados Instancias Casos de Prueba Reales.

Caso		Heurística								
Pieza base (Ancho x Alto)	Ocupación (%)	Random			Mayor a menor			Menor a mayor		
		Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness	Tiempo de solución (s)	Desperdicio	Fitness
183x275	96.4	20.832	7777	7781	29.503	6017	6021	29.450	7096	7100
185x274	96.17	36.635	6030	6034	72.237	3630	3632	57.030	3630	3632
204x300	87.6	28.362	7589	7589	36.116	7589	7589	37.985	7589	7589
220x310	97.35	465.015	7020	7026	638.392	4300	4304	510.237	4360	4364
244x122	87.35	1.320	3766	3766	0.850	3766	3766	1.073	3766	3766
244x488	79.84	54.272	24002	24002	48.079	24002	24002	83.671	24002	24002
250x200	72.0	3.649	14000	14000	3.194	14000	14000	2.153	14000	14000
275x138	93.46	48.653	5683	5685	91.575	2483	2483	75.486	2483	2483
275x184	77.79	70.180	17045	17051	84.221	11237	11237	76.340	11237	11237
275x273	96.52	162.397	10312	10316	129.041	4824	4826	227.507	5385	5389
280x207	67.48	52.825	18846	18846	37.074	18846	18846	45.799	18846	18846

Discusión de Resultados:

Los resultados compilados en las tablas para los cuatro casos de instancias permiten identificar diferencias claras entre el modelo exacto y la metaheurística basada en VNS, especialmente en tiempo computacional y escalabilidad. Para las instancias Small y Medium, el MILP alcanza consistentemente soluciones óptimas con tiempos de resolución bajos, aunque el tiempo de carga y construcción de datos crece con la dimensión de la lámina base.

En contraste, la metaheurística mantiene tiempos computacionales significativamente menores en todos los tamaños y reproduce la calidad del modelo exacto en las instancias donde existe referencia óptima (Small y Medium), con GAP = 0 en todos los casos. Esto se explica porque la metaheurística no opera mediante un método independiente de inserción, sino mediante un mecanismo de decodificación basado en Bottom Left, cuya efectividad depende directamente del ordenamiento de los ítems que se exploran durante las permutaciones generadas por la metaheurística basada en VNS.

El análisis de las cuatro clases de instancias evidencia un comportamiento coherente y progresivo en términos de tiempos de cómputo, calidad de solución y sensibilidad del algoritmo frente al incremento en escala y heterogeneidad geométrica.

En las instancias Small y Medium, el MILP resolvió todos los casos con optimalidad garantizada (status = 0), con tiempos de solución bajos para Small (<0.20 s) y crecientes para Medium (hasta ~90 s). Este incremento confirma la pérdida natural de escalabilidad de los métodos exactos ante aumentos moderados en el tamaño del dominio de corte y de la densidad de demanda.

En contraste, la metaheurística mantuvo tiempos significativamente menores en ambas clases, con rangos de milisegundos a pocos segundos, y reprodujo exactamente los valores óptimos del MILP en todos los casos evaluados (GAP = 0 %). La variación entre ordenamientos fue limitada y tuvo un efecto marginal en la calidad final, destacándose la secuencia mayor a menor por su estabilidad temporal. Estos resultados validan la capacidad de la metaheurística para capturar soluciones óptimas en problemas de escala pequeña y media, con una eficiencia computacional notablemente superior.

Para las instancias Large, la metaheurística mostró un incremento esperado en los tiempos de ejecución con variabilidad de milisegundos hasta valores entre 18 y 24 s en configuraciones aleatorias atribuible a la mayor complejidad espacial y a la diversidad de patrones evaluados durante el proceso de decodificación. No obstante, la calidad de solución se mantuvo altamente estable: los valores de desperdicio y fitness fueron equivalentes para los tres ordenamientos en todos los tamaños.

Finalmente, en las instancias Reales caracterizadas por geometrías heterogéneas y demandas elevadas (67 % – 97 %); la metaheurística mostró la mayor dispersión temporal del conjunto experimental, con tiempos comprendidos entre menos de 1 ms y aproximadamente 0.6 s. Esta variabilidad responde a diferencias en tamaño de placa, distribución de demanda y estructura de huecos generada durante la construcción del patrón. A pesar de ello, el desperdicio y el fitness fueron prácticamente idénticos entre los ordenamientos en 9 de las 11 instancias. El ordenamiento mayor a menor mantuvo los tiempos más bajos en la mayoría de los casos, coherente con su efecto de reducción temprana de fragmentación.

En conjunto, los resultados demuestran que la metaheurística basada en VNS ofrece altamente estabilidad y calidad consistente en todos los rangos experimentales, igualando el óptimo exacto en Small y Medium, y proporcionando soluciones compactas y computacionalmente viables en Large y Reales. La comparación global evidencia que la metaheurística no solo es escalable y eficiente, sino que mantiene patrones de corte de alta densidad independientemente de la secuencia inicial, lo que sustenta su aplicabilidad industrial como alternativa confiable frente a métodos exactos en escenarios reales de gran complejidad.

La Tabla 14 presenta una síntesis comparativa que integra los resultados obtenidos en las cuatro clases de instancias. Así como el comportamiento del método exacto y de la metaheurística frente a variaciones en tamaño, demanda, complejidad geométrica y estabilidad de solución. Su propósito es ofrecer una visión global del rendimiento de ambos enfoques y destacar la evolución del desempeño metaheurístico a medida que el problema incrementa en escala y heterogeneidad.

Tabla 14 Comparación Global de Desempeño por Caso de Instancias

Concepto	Small	Medium	Large	Real
Tamaño Placa Base	10x10 15x15 20x20	50x50 55x55 60x60	70x70 75x75 80x80	Heterogéneas (11 Tamaños Distintos)
Porcentaje Ocupación	80 – 90%	80 – 90%	80 – 90%	67 – 97 %
Tiempo MILP	Viable y óptimo. Tiempo < 0.20 s	Viable, más lento. Tiempo hasta ≈ 90 s	-	-
Status MILP	0 (óptimo en todos)	0 (óptimo en todos)	-	-
Tiempos VNS	0.001–0.40 s	0.001–13 s	0.015–24 s	0.001–0.60 s (alta variabilidad)
Calidad de solución (desperdicio / fitness)	Igual al óptimo (GAP = 0%)	Igual al óptimo (GAP = 0%)	Alta estabilidad entre ordenamientos	Alta estabilidad entre ordenamientos
Comportamiento del Ordenamiento	Mínimas Variaciones	Leve Variación Mejor: Mayor a Menor	Igualdad en desperdicio y fitness, diferencia sólo en tiempo	Igualdad en desperdicio y fitness, diferencia sólo en tiempo
Robustez Metaheurística	Muy Alta	Muy Alta	Muy Alta	Muy Alta
Escalabilidad	Excelente	Excelente	Alta (única alternativa viable)	Alta (única alternativa viable)

Fuente: Elaboración Propia.

A partir de los hallazgos expuestos en este capítulo, se consolidan elementos suficientes para valorar la efectividad del enfoque propuesto y su potencial aplicación en entornos reales de la industria metalmecánica. Los resultados cuantitativos y comparativos obtenidos no solo permiten validar la capacidad de la heurística para reproducir soluciones de alta calidad, sino que también destacan su fortaleza frente a las limitaciones computacionales inherentes al método exacto. En consecuencia, el siguiente capítulo presenta las conclusiones generales de la investigación, integrando las contribuciones metodológicas, los aprendizajes derivados del análisis experimental y las implicaciones prácticas para la gestión eficiente de materiales en procesos de corte bidimensional.

8 CONCLUSIONES

La presente investigación abordó un problema real de la industria metalmecánica mediante la abstracción de un problema ampliamente estudiado en la literatura, que se denomina 2DCSP con restricciones del tipo guillotina; evaluando dos enfoques complementarios: un modelo exacto formulado mediante programación entera mixta MILP y una metaheurística basada en VNS diseñada para mejorar la escalabilidad y reducir los tiempos de cómputo en instancias de mayor complejidad.

En primer lugar, la formulación exacta permitió establecer un marco de referencia sólido para la validación de la metaheurística. El MILP demostró ser altamente efectivo para instancias de pequeña y mediana escala (Small y Medium), resolviendo todos los casos con estado óptimo (status = 0) y tiempos de solución inferiores a un segundo. Estos resultados confirman que en configuraciones moderadas en número de ítems y niveles de ocupación entre el 80 % y 90 % la optimización exacta, constituye una herramienta confiable y rigurosa para encontrar soluciones óptimas en contextos controlados.

Sin embargo, el desempeño del MILP se volvió restrictivo a medida que se incrementó la dimensión de las piezas base y el número de ítems por instancia. Lo anterior se evidenció en casos de mayor tamaño (Large) y en las instancias reales provenientes del caso de estudio, debido al crecimiento exponencial del espacio de decisiones y a la dificultad de representar cortes guillotina en estructuras de orden superior. Esto confirma una limitación ampliamente reportada en la literatura sobre los enfoques exactos que, aunque precisos, carecen de escalabilidad cuando la complejidad geométrica y combinatoria del problema aumenta.

En contraste, la metaheurística propuesta mostró un comportamiento robusto, eficiente y consistente en todos los tipos de instancias. Su estructura secuencial permitió generar soluciones factibles con fitness igualando al óptimo para las clases Small y Medium, con un GAP de 0 % en la totalidad de los casos evaluados. Este resultado confirma la capacidad del enfoque para reproducir el comportamiento exacto sin incurrir en los costos computacionales asociados al MILP.

Asimismo, en las instancias Large y Reales, donde el MILP no es aplicable, la metaheurística mantuvo tiempos de ejecución manejables y generó soluciones con niveles de compactación y aprovechamiento altamente competitivos. Aunque en estos escenarios se observaron variaciones en los tiempos de ejecución debido al tipo de ordenamiento utilizado, los resultados finales, en términos de desperdicio y fitness, permanecieron estables. Este comportamiento sugiere que la metaheurística no sólo es rápida, sino también resiliente frente a cambios en la configuración de entrada.

Un aporte relevante de esta investigación es haber demostrado que la integración de enfoques constructivos y selectivos dentro de un mismo esquema metaheurístico constituye una alternativa viable para problemas industriales reales en los que el tamaño de las instancias supera la capacidad operativa de los métodos exactos. Además, el diseño modular de la metaheurística abre la posibilidad de incorporar mecanismos de mejora adicionales (búsqueda local, aleatorización controlada o metaheurísticas basadas en construcción de vecindades para fortalecer aún más la calidad de las soluciones).

Finalmente, el caso de estudio de la empresa metalmecánica validó empíricamente la aplicabilidad del enfoque propuesto. La metaheurística reprodujo patrones de corte consistentes

con las necesidades operativas, generó tiempos de cómputo compatibles con flujos productivos reales y demostró capacidad para reducir el desperdicio frente a prácticas empíricas o reglas manuales de corte. Estos resultados permiten concluir que la propuesta constituye una herramienta potencialmente implementable y con impacto tangible en la eficiencia del uso de material.

REFERENCIAS BIBLIOGRÁFICAS

- Alvarez, David. 2010. “Solución del problema de empaquetamiento optimo bidimensional en una sola placa, en placas y rollos infinitos con o sin rotación de piezas usando técnicas metaheurísticas de optimización”. Universidad Tecnológica de Pereira. <https://core.ac.uk/download/pdf/487138487.pdf> (el 6 de septiembre de 2025).
- Andrade, R., E.G. Birgin, y R. Morabito. 2016. “Two-stage two-dimensional guillotine cutting stock problems with usable leftover”. *International Transactions in Operational Research* 23(1–2): 121–45. doi:10.1111/itor.12077.
- Arai, Hiroshi, y Harumi Haraguchi. 2021. “A Study of Ising Formulations for Minimizing Setup Cost in the Two-Dimensional Cutting Stock Problem”. *Algorithms* 14(6): 1–9. doi:10.3390/a14060182.
- Aryanezhad, Mir-Bahador, Nima Fakhim Hashemi, Ahmad Makui, y Hasan Javanshir. 2012. “A simple approach to the two-dimensional guillotine cutting stock problem”. *Journal of Industrial Engineering International* 8(1): 21. doi:10.1186/2251-712X-8-21.
- Baack, Thomas, D.B Fogel, y Z Michalewicz. 1997. *Handbook of Evolutionary Computation*. CRC Press. doi:10.1201/9780367802486.
- Beasley, J. E. 1985. “Algorithms for Unconstrained Two-Dimensional Guillotine Cutting”. *The Journal of the Operational Research Society* 36(4): 297. doi:10.2307/2582416.
- Benjaoran, Vacharapoom, y Sdhabhon Bhokha. 2014. “Three-Step Solutions for Cutting Stock Problem of Construction Steel Bars”. *KSCE Journal of Civil Engineering* 18(5): 1239–47. doi:10.1007/s12205-014-0238-3.
- Blum, Christian, y Andrea Roli. 2003. “Metaheuristics in combinatorial optimization”. *ACM Computing Surveys* 35(3): 268–308. doi:10.1145/937503.937505.
- Bouaine, Amine, Maria Lebbar, y Mohamed Ait Ha. 2018. “Minimization of the Wood Wastes for an Industry of Furnishing: A Two Dimensional Cutting Stock Problem”. *Management and Production Engineering Review* 9(2): 42–51. doi:10.24425/119524.
- Burke, Edmund, Graham Kendall, Jim Newall, Emma Hart, Peter Ross, y Sonia Schulenburg. 2013. “Hyper-Heuristics: An Emerging Direction in Modern Search Technology”. En *Handbook of Metaheuristics*, Boston: Kluwer Academic Publishers, 457–74. doi:10.1007/0-306-48056-5_16.
- Cao, Dayong, y V. M. Kotov. 2011. “A Best-Fit Heuristic Algorithm for Two-Dimensional Bin Packing Problem”. En *Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology*, IEEE, 3789–91. doi:10.1109/EMEIT.2011.6023883.

- Cheng, C. H., B. R. Feiring, y T. C.E. Cheng. 1994. "The Cutting Stock Problem — a Survey". *International Journal of Production Economics* 36(3): 291–305. doi:10.1016/0925-5273(94)00045-X.
- Cherri, Adriana Cristina, Marcos Nereu Arenales, y Horacio Hideki Yanasse. 2009. "The One-Dimensional Cutting Stock Problem with Usable Leftover - A Heuristic Approach". *European Journal of Operational Research* 196(3): 897–908. doi:10.1016/j.ejor.2008.04.039.
- Clerc, M., y J. Kennedy. 2002. "The particle swarm - explosion, stability, and convergence in a multidimensional complex space". *IEEE Transactions on Evolutionary Computation* 6(1): 58–73. doi:10.1109/4235.985692.
- Corporación Autónoma Regional del Valle del Cauca (CVC). 2022. Informe Regional del Reporte de Registro de Generadores de Residuos Peligrosos (RESPEL) en la jurisdicción del Valle del Cauca — Vigencia 2020.
- Crainic, Teodor Gabriel, Guido Perboli, y Roberto Tadei. 2009. "TS2PACK: A two-level tabu search for the three-dimensional bin packing problem". *European Journal of Operational Research* 195(3): 744–60. doi:10.1016/j.ejor.2007.06.063.
- Cui, Yaodong, Cheng Zhong, y Yi Yao. 2015. "Pattern-set generation algorithm for the one-dimensional cutting stock problem with setup cost". *European Journal of Operational Research* 243(2): 540–46. doi:10.1016/j.ejor.2014.12.015.
- Desrosiers, Jacques, y Marco E. Lübbecke. 2005. "A Primer in Column Generation". En *Column Generation*, Springer, Boston, MA, 1–32. doi:10.1007/0-387-25486-2_1.
- Dorigo, M., V. Maniezzo, y A. Coloni. 1996. "Ant system: optimization by a colony of cooperating agents". *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 26(1): 29–41. doi:10.1109/3477.484436.
- Dorigo, Marco, y Thomas Stützle. 2004. *Ant Colony Optimization*. The MIT Press. doi:10.7551/mitpress/1290.001.0001.
- Dyckhoff, Harald. 1990. "A typology of cutting and packing problems". *European Journal of Operational Research* 44(2): 145–59. doi:10.1016/0377-2217(90)90350-K.
- Dyson, R. G., y A. S. Gregory. 1974. "The Cutting Stock Problem in the Flat Glass Industry". *Journal of the Operational Research Society* 25(1): 41–53. doi:10.1057/jors.1974.5.
- Eiben, A.E., y J.E. Smith. 2015. *Introduction to Evolutionary Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-662-44874-8.
- Fleszar, Krzysztof. 2013. "Three insertion heuristics and a justification improvement heuristic for two-dimensional bin packing with guillotine cuts". *Computers and Operations Research* 40(1): 463–74. doi:10.1016/j.cor.2012.07.016.

- Furini, Fabio, Enrico Malaguti, y Dimitri Thomopulos. 2016. "Modeling Two-Dimensional Guillotine Cutting Problems via Integer Programming". *INFORMS Journal on Computing* 28(4): 736–51. doi:10.1287/ijoc.2016.0710.
- García, Luis, Coromoto León, Gara Miranda, y Casiano Rodríguez. 2006. "A Parallel Algorithm for the Two-Dimensional Cutting Stock Problem". En , 821–30. doi:10.1007/11823285_86.
- Gendreau, Michel, y Jean-Yves Potvin. 2010. 146 *Handbook of Metaheuristics*. Boston, MA: Springer US. doi:10.1007/978-1-4419-1665-5.
- Gilmore, P. C., y R. E. Gomory. 1961. "A Linear Programming Approach to the Cutting-Stock Problem". *Operations Research* 9(6): 849–59. doi:10.1287/opre.9.6.849.
- Gilmore, P. C., y R. E. Gomory. 1963. "A Linear Programming Approach to the Cutting Stock Problem—Part II". *Operations Research* 11(6): 863–88. doi:10.1287/opre.11.6.863.
- Glover, Fred. 1989. "Tabu Search—Part I". *ORSA Journal on Computing* 1(3): 190–206. doi:10.1287/ijoc.1.3.190.
- Gracia, Carlos. 2010. "Métodos y algoritmos para resolver problemas de corte unidimensional en entornos realistas: aplicación a una empresa del sector siderúrgico". Universitat Politècnica de València.
<https://dialnet.unirioja.es/servlet/tesis?codigo=240195&info=resumen&idioma=SPA> (el 6 de septiembre de 2025).
- Haessler, Robert. 1975. "Controlling Cutting Pattern Changes in One-Dimensional Trim Problems". *OPERATIONS RESEARCH* 23: 483–93. <http://www.jstor.org/stable/169698> (el 6 de septiembre de 2025).
- Haessler, Robert W., y Paul E. Sweeney. 1991. "Cutting stock problems and solution procedures". *European Journal of Operational Research* 54(2): 141–50. doi:10.1016/0377-2217(91)90293-5.
- Holland, J.H. 1975. *Adaption in Natural and Artificial Systems*. ed. University of Michigan Press.
- Imahori, Shinji, Mutsunori Yagiura, Shunji Umetani, Shinya Adachi, y Toshihide Ibaraki. 2005. "Local search algorithms for the two-dimensional cutting stock problem with a given number of different patterns". *Operations Research/ Computer Science Interfaces Series* 32: 181–202. doi:10.1007/0-387-25383-1_8.
- Iori, Manuel, Vinícius L. de Lima, Silvano Martello, Flávio K. Miyazawa, y Michele Monaci. 2021. "Exact Solution Techniques for Two-Dimensional Cutting and Packing". *European Journal of Operational Research* 289(2): 399–415. doi:10.1016/j.ejor.2020.06.050.
- Israni, Sharat, y Jerry Sanders. 1982. "Two-dimensional cutting stock problem research: A review and a new rectangular layout algorithm". *Journal of Manufacturing Systems* 1(2): 169–82. doi:10.1016/S0278-6125(82)80027-7.

- Jaramillo, Juan. 2008. "Programación Lineal y Algoritmos Genéticos Para la Solución de un Problema de Corte". Universidad Eafit. <http://hdl.handle.net/10784/2545> (el 6 de septiembre de 2025).
- Kantorovich, L. V. 1960. "Mathematical Methods of Organizing and Planning Production". *Management Science* 6(4): 366–422. doi:10.1287/mnsc.6.4.366.
- Kennedy, J., y R. Eberhart. 1995. "Particle swarm optimization". En *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, 1942–48. doi:10.1109/ICNN.1995.488968.
- Kirkpatrick, S., C. D. Gelatt, y M. P. Vecchi. 1983. "Optimization by Simulated Annealing". *Science* 220(4598): 671–80. doi:10.1126/science.220.4598.671.
- Kokten, Erkan Sami, y Çağrı Sel. 2022. "A cutting stock problem in the wood products industry: a two-stage solution approach". *International Transactions in Operational Research* 29(2): 879–907. doi:10.1111/itor.12802.
- Kröger, Berthold. 1995. "Guillotineable Bin Packing: A Genetic Approach". *European Journal of Operational Research* 84(3): 645–61. doi:10.1016/0377-2217(95)00029-P.
- Land, A. H., y A. G. Doig. 1960. "An Automatic Method of Solving Discrete Programming Problems". *Econometrica* 28(3): 497. doi:10.2307/1910129.
- Lodi, Andrea, Silvano Martello, y Daniele Vigo. 2002. "Heuristic and Metaheuristic Approaches for a Class of Two-Dimensional Bin Packing Problems". *INFORMS Journal on Computing* 11(4): 345–57. doi:10.1287/ijoc.11.4.345.
- Martello, Silvano y Toth, Paolo. 1990. *Knapsack Problems: Algorithms and Computer Implementations*. ed. John Wiley & Sons.
- Mellouli, Ahmed, y Abdelaziz Dammak. 2008. "An Algorithm for the Two-Dimensional Cutting-Stock Problem Based on a Pattern Generation Procedure". *Information and Management Sciences* 19(2): 201–18.
- Mingjie Hu, Jie Yan, Liting Chen, Qingwei Lin. 2023. "The Ensemble Approach of Column Generation for Solving Cutting Stock Problems". doi:10.48550/arXiv.2305.14055.
- Mladenović, N., y P. Hansen. 1997. "Variable neighborhood search". *Computers & Operations Research* 24(11): 1097–1100. doi:10.1016/S0305-0548(97)00031-2.
- Mobasher, Azadeh, y Ali Ekici. 2013. "Solution approaches for the cutting stock problem with setup cost". *Computers and Operations Research* 40(1): 225–35. doi:10.1016/j.cor.2012.06.007.
- Morillo-Torres, Daniel, Mauricio Torres Baena, John Wilmer Escobar, Alfonso R. Romero-Conrado, Jairo R. Coronado-Hernández, y Gustavo Gatica. 2021. "A Mixed-Integer Linear Programming Model for the Cutting Stock Problem in the Steel Industry". *Applied Computer Sciences in*

- Engineering. WEA 2021. *Communications in Computer and Information Science* 1431: 315–26. doi:10.1007/978-3-030-86702-7_27.
- Mostajabdaveh, Mahdi, F. Sibel Salman, y Nadia Tahmasbi. 2022. “Two dimensional guillotine cutting stock and scheduling problem in printing industry”. *Computers & Operations Research* 148: 106014. doi:10.1016/j.cor.2022.106014.
- Nemhauser, G.L., y Wolsey, L.A. 1988. *Integer and Combinatorial Optimization*. ed. Wiley-Interscience. New York.
- Oliveira, José Fernando, Alvaro Neuenfeldt, Elsa Silva, y Maria Antónia Carravilla. 2016. “A survey on heuristics for the two-dimensional rectangular strip packing problem”. *Pesquisa Operacional* 36(2): 197–226. doi:10.1590/0101-7438.2016.036.02.0197.
- Osman, Ibrahim H., y Gilbert Laporte. 1996. “Metaheuristics: A bibliography”. *Annals of Operations Research* 63(5): 511–623. doi:10.1007/BF02125421.
- Peña, Diana, Juan Orejuela, y Cristian Gil. 2017. “El Problema de patrones de corte, clasificación y enfoques/Cutting stock problem, classification and approaches”. *Prospectiva* 15(1): 112–25. doi:10.15665/rp.v15i1.718.
- Pisinger, David, y Stefan Ropke. 2007. “A general heuristic for vehicle routing problems”. *Computers & Operations Research* 34(8): 2403–35. doi:10.1016/J.COR.2005.09.012.
- Russo, Mauro, Maurizio Boccia, Antonio Sforza, y Claudio Sterle. 2020. “Constrained two-dimensional guillotine cutting problem: upper-bound review and categorization”. *International Transactions in Operational Research* 27(2): 794–834. doi:10.1111/itor.12687.
- Saldías, Oscar, Carla Guerrero, Rodrigo Olivares, Daniel Morillo-Torres, y Gustavo Gatica. 2022. “A Strategic-Operational Contribution to the One-Dimensional Multi-period Cutting Stock Problem for Sawmills”. *Communications in Computer and Information Science* 1676 CCIS: 30–41. doi:10.1007/978-3-031-20316-9_3.
- Scheithauer, Guntram. 1992. “Algorithms for the Container Loading Problem”. En *Operations Research Proceedings 1991*, Springer Berlin Heidelberg, 445–52. doi:10.1007/978-3-642-46773-8_112.
- Talbi, El-Ghazali. 2009. *Metaheuristics: From Design to Implementation*. ed. John Wiley & Sons.
- Tlilane, Lydia, y Quentin Viaud. 2018. *Cutting Optimization Problem Description*.
- Toro, Eliana, Alejandro Garcés, y Hugo Ruiz. 2008. “Solución al problema de empaquetamiento bidimensional usando un algoritmo híbrido constructivo de búsqueda en vecindad variable y recocido simulado”. *Revista Facultad de Ingeniería Universidad de Antioquia* 46: 119–31. <https://www.redalyc.org/articulo.oa?id=43004613> (el 6 de septiembre de 2025).

- Ülkü, İlayda, Ugur Tekeoğlu, Müge Özler, Nida Erdal, y Yağız Tolonay. 2022. "Solving the Cutting Stock Problem for a Steel Industry". *Lecture Notes in Mechanical Engineering*: 510–18. doi:10.1007/978-3-030-90421-0_43.
- Umetani, Shunji, Mutsunori Yagiura, y Toshihide Ibaraki. 2003. "One-dimensional cutting stock problem to minimize the number of different patterns". *European Journal of Operational Research* 146(2): 388–402. doi:10.1016/S0377-2217(02)00239-4.
- Vance, Pamela H. 1998. "Branch-and-Price Algorithms for the One-Dimensional Cutting Stock Problem". *Computational Optimization and Applications* 9(3): 211–28. doi:10.1023/A:1018346107246.
- Wäscher, Gerhard, Heike Haußner, y Holger Schumann. 2007. "An improved typology of cutting and packing problems". *European Journal of Operational Research* 183(3): 1109–30. doi:10.1016/j.ejor.2005.12.047.
- Wuttke, David A., y H. Sebastian Heese. 2018. "Two-dimensional cutting stock problem with sequence dependent setup times". *European Journal of Operational Research* 265(1): 303–15. doi:10.1016/j.ejor.2017.07.036.
- Yuen, Boon J., y Ken V. Richardson. 1995. "Establishing the optimality of sequencing heuristics for cutting stock problems". *European Journal of Operational Research* 84(3): 590–98. doi:10.1016/0377-2217(95)00025-L.