

UNIVERSIDAD DE CONCEPCIÓN - CHILE
FACULTAD DE INGENIERIA
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

***Propuesta y resolución de un modelo MIP
para la planificación táctica entre cosecha
forestal y aserraderos, usando reglas de
trozados dinámicos.***

por
Sebastián Fuentealba Rivas

Profesor Guía:
Dra. Lorena Pradenas Rojas
Profesor Co-Guía:
Dr. Rodrigo Linfati Medina

Concepción, Abril 2016

Tesis presentada a la
**DIRECCIÓN DE POSTGRADO
UNIVERSIDAD DE CONCEPCION**



Para optar al grado de

MAGISTER EN INGENIERÍA INDUSTRIAL

Dedicado a mis padres por su infinito amor y comprensión.
Al profesor Rodrigo Linfati por su apoyo constante y disposición a
entregar sus conocimientos.
Al Club Social y Deportivo Colo – Colo por entregarle alegría a la
mitad más uno del pueblo Chileno, en tiempos que la clase política
y económica solo quieren arrebatarla.

RESUMEN

Propuesta y resolución de un modelo MIP para la planificación táctica entre cosecha forestal y aserraderos, usando reglas de trozados dinámicos.

Sebastián Fuentealba Rivas

Abril 2016

PROFESOR GUIA: Dra. Lorena Pradenas Rojas.
PROFESOR CO-GUIA: Dr. Rodrigo Linfati Medina.
PROGRAMA: Magister en Ingeniería Industrial.

Este estudio propone y resuelve un modelo de programación matemática que permite optimizar la planificación táctica integrada entre cosecha forestal y aserraderos. Se usan dos estrategias, la primera es la generación de reglas de trozadas dinámicas para la cosecha forestal mediante, el método de generación de columnas y la segunda, es una integración entre las decisiones de cosecha forestal y aserraderos. Para obtener los resultados se genera un conjunto de instancias, las cuales presentan diversos escenarios con diferentes características reales, se utilizó el software CPLEX 12.6.1 de ILOG para implementar y resolver el modelo con las instancias propuestas. Se obtienen soluciones, que permiten inferir que el modelo es apropiado para la resolución del problema integrado.

Palabras claves: aserraderos, cosecha forestal, generación de columnas, MIP, reglas de trozado, planificación forestal.

ABSTRACT

Proposal and resolution of a MIP model for the tactical planning between forest harvesting and sawmills, with rules of dynamic bucking

Sebastián Fuentealba Rivas

April 2016

THESIS SUPERVISOR: Dra. Lorena Pradenas Rojas.
THESIS CO SUPERVISOR: Dr. Rodrigo Linfati Medina.
PROGRAM: Magister en Ingeniería Industrial.

In this study a mathematical programming model that allows minimizing an integral tactical planning between forest harvesting and sawmills is proposed and solved. In this case, two strategies are used: The first strategy is the generation of dynamic bucking rules for the forest harvest through the method of column generation, and the second strategy is the integration between the decisions of forest harvesting and sawmill. In order to obtain results, a set of instances is generated. These present several scenarios and real features. CPLEX 12.6.1 by ILOG software was used to implement and solve the method with the proposed characteristics. Solutions that allow inferring that the model is suitable for solving the integrated problem are obtained.

Keywords: sawmill, forest harvesting, column generation, MIP, bucking rules, forest planning.

ÍNDICE

Lista de Tablas	7
Lista de Figuras	8
CAPÍTULO 1. Introducción	10
1.1 Hipótesis de Investigación	11
1.2 Objetivo General	11
1.3 Objetivos Específicos.....	11
1.4 Metodología	12
CAPÍTULO 2. Antecedentes de los Procesos de la Cosecha Forestal.....	14
2.1 Industria Forestal.....	14
2.2 Industria Forestal en Chile	14
2.3 Ciclo de la Industria Forestal	15
2.4 Rendimiento de los Bosques	16
2.5 Transformación de Rollizos a Tablas.....	18
CAPÍTULO 3. Marco Teórico y Estado del Arte	20
3.1 Proceso Forestal	20
3.2 Planificación en la Cosecha Forestal	30
3.2.1 Planificación Estratégica Forestal.....	32
3.2.2 Planificación Táctica Forestal	33
3.2.3 Planificación Operativa Forestal.....	34
3.3 Revisión de la Literatura	35
CAPÍTULO 4. Descripción del Problema, Modelamiento y Propuesta de Solución ..	37
4.1 Descripción del problema	37
4.2 Modelo de Programación Matemática	42
4.3 Método de solución.....	52
CAPÍTULO 5. Resultados	55

CAPÍTULO 6. Conclusiones	64
Referencias.....	66
Anexo 1. Método Generación de Columnas	71
Anexo 2. Formulación Matemática Completa	80
A.2.1 Modelo de Programación Matemática	81
Anexo 3. Código del Modelo Matemático Implementado en Cplex 12.6.1	87

Lista de Tablas

Tabla 2.1 Cantidad de trozos por tipo de árbol utilizados por producto en m ³	15
Tabla 3.1 Ejemplo de regla de trozado	24
Tabla 3.2 Ejemplo de orden de exportación a Corea	26
Tabla 3.3 Niveles de planificación	32
Tabla 4.1 Características árbol y productos para generar la regla de trozado.....	53
Tabla 4.2 Resultados problema de ejemplo.....	53
Tabla 5.1 Dimensiones instancia original	55
Tabla 5.2 Características de los rodales en la instancia original	56
Tabla 5.3 Características de los productos en la instancia original	56
Tabla 5.4 Resultados instancia original	57
Tabla 5.5 Características de las reglas de trozados utilizadas en la instancia original	57
Tabla 5.6 Agrupación reglas de trozado por rodal	58
Tabla 5.7 Desglose de las reglas de trozado generadas instancia original	59
Tabla 5.8 Dimensiones utilizadas para el conjunto de instancias iniciales	61
Tabla 5.9 Resultados conjunto de instancias	62
Tabla A.1.1 Resumen de pedidos de clientes y características	72

Lista de Figuras

Figura 2.1 Esquema de creación de parcelas dentro de un rodal.....	17
Figura 2.2 Forcípula.	18
Figura 2.3 Medición con la forcípula de un árbol de sección no circular	18
Figura 3.1 Alturas y porcentajes de poda en esquema intensivo	21
Figura 3.2 Esquema general de torres de madereo	22
Figura 3.3 Maquinaria Skidder, encargada del método de madereo terrestre	22
Figura 3.4 Ejemplo de rollizos obtenidos de un árbol.	25
Figura 3.5 Relación entre 1 m^3 en madera rolliza y tablas aserradas en pt.	27
Figura 3.6 Transporte de productos a sus destinos.	28
Figura 3.7 Ejemplos de patrones de cortes en los aserraderos.	30
Figura 4.1 Representación Bosques, rodal y parcelas de cosecha	37
Figura 4.2 Representación largo, diámetro mínimo y diámetro máximo de un rollizo ...	38
Figura 4.3 Flujo de productos de los bosques hacia los clientes	39
Figura 4.4 Flujo de producto del aserradero hacia los clientes	40
Figura 4.5 Ejemplo de utilización de las expresiones (4.21) a la (4.24)	51
Figura 4.6 Representación problema de ejemplo regla de trozado	53
Figura 4.7 Esquematización resultados problema de ejemplo	54
Figura 5.1 Desglose del total de reglas de trozado	59
Figura 5.2 Desglose reglas de trozado	60
Figura 5.3 Comparación del valor de la función objetivo del conjunto de instancias ...	61
Figura 5.4 Comparación tiempos de resolución del conjunto de instancias	63
Figura A.1.1 Representación característica de productos solicitados	73

Figura A.2.1 Representación reemplazo de variables entre modelo completo y su descomposición	80
--	----

CAPÍTULO 1.

Introducción

La industria forestal ha adquirido un papel importante a nivel mundial y nacional, transformándose en un agente económico relevante. Esto se ve reflejado en los indicadores económicos donde las exportaciones han aumentado de US\$ 9,7 millones FOB (1962) hasta US\$ 3.495,4 millones FOB (2005). En el contexto de la economía nacional, los productos forestales han elevado su participación en exportaciones de 1,9 % (1962) a 12,7% (2006). (Carrasco. M., 2008)

En el proceso forestal se involucran diferentes actores en todas las etapas de la industria. El ciclo comienza con las empresas productoras (por ejemplo en el caso de Chile, CMPC, ARAUCO, entre otras) quienes solicitan servicios a terceras organizaciones denominadas empresas de servicios forestales (ESF), estas son las encargadas del establecimiento de los bosques (preparación de suelos, plantación, aplicación de químicos, entre otras tareas), manejo (poda, raleo) y cosecha forestal (extracción de madera desde los bosques). Esta etapa del ciclo forestal, es lo central del presente estudio. Posterior a la cosecha de productos en el bosque, se recurre a empresas de transportes quienes son los encargados de trasladar los rollizos a los diversos clientes, los más importantes son: exportación, aserraderos y celulosas.

Las empresas encargadas de la cosecha forestal planifican las acciones siguiendo el modelo tradicional piramidal: nivel estratégico (decisiones a largo plazo), táctico (decisiones a mediano plazo) y operativo (decisiones a corto plazo). La importancia de planificar radica en: organizar la empresa, prever dificultades, distribuir y potenciar los recursos y establecer objetivos.

Entre una las decisiones importantes que debe tomar una empresa forestal del área de explotación y cosecha de bosques se encuentra, definir con anterioridad la regla de trozado a aplicar en los árboles cosechados en otras palabras, en cuantas partes se dividirá un árbol trozado y la longitud de éstas, y así satisfacer la demanda de: exportación de maderas, aserraderos, papeleras, plantas de celulosa, entre otras, junto con obtener la mayor cantidad de beneficios.

En el lenguaje forestal, los bosques se dividen en rodales, los que se clasifican por tipo de árbol (*pinus radiata D.Don*, *eucalyptus nitens*, *eucalyptus globulus* en su mayoría) y edad. De los rodales, se obtienen rollizos, los cuales pueden ser almacenados, en canchas de acopio o

enviados a los diferentes clientes, donde los transforman en diversos productos para su comercialización.

La problemática de la planificación táctica, de la cosecha forestal, considera decisiones que tienen injerencia, en un plazo de 6 meses a dos años. En el presente estudio se trabajará la cosecha forestal como un problema de producción, considerando diversos factores; tales como: las canchas de acopio, producción de diversos productos, niveles de inventario, etc. Además, se integra a la planificación táctica las decisiones a tomar en los aserraderos, ya que estos poseen un vínculo más estrecho con la cosecha forestal porque, requieren una mayor cantidad de rollizos y el proceso de transformación de rollizos a tablas requiere de la aplicación de esquemas de corte similares al usado en el trozado de los rollizos.

Existen diversas herramientas para resolver este problema, uno es el uso del método de generación de columnas para obtener los mejores rendimientos en los bosques y así aplicar las mejores reglas de trozados, y patrones dinámicos dependiendo de los compromisos adquiridos por las empresas mandantes con los clientes. Esto entrega un enfoque más holístico al problema, permitiendo tomar mejores decisiones a nivel táctico; además de flexibilidad y una mayor gama de posibilidades al momento de programar la producción para satisfacer la demanda.

De acuerdo con la descripción anterior y el estado del arte realizado se planteó en este estudio la:

1.1 Hipótesis de Investigación

Es posible formular y resolver un modelo de programación matemática, que permita apoyar la toma de decisiones tácticas de mediano plazo (2 a 3 años), entre cosecha forestal y aserraderos, utilizando reglas de trozados dinámicas para los rodales obteniéndolas mediante el método de generación de columnas.

1.2 Objetivo General

Formular, resolver y evaluar un modelo de programación matemática, que permita apoyar la toma de decisiones en la planificación táctica entre cosecha forestal y aserraderos, utilizando reglas de trozado dinámicas a través del método de generación de columnas.

1.3 Objetivos Específicos

- Investigar sobre el estado del arte existente sobre: la cosecha forestal, trozados de árboles en los bosques, esquemas de corte en los aserraderos y optimización integrada.

- Diseñar modelo de programación matemática, que utilice patrones dinámicos en la cosecha forestal y que además, integre las decisiones tomadas en conjunto con los aserraderos.
- Implementar computacionalmente el modelo de programación matemática.
- Generar un conjunto de instancias estructuradas, de acuerdo a problemas reales que permita inferir diferentes conclusiones y aporten a la toma de decisiones de la planificación táctica.

1.4 Metodología

Para comenzar el estudio de trabajo se abordó la literatura existente, con el fin de encontrar metodologías e investigaciones anteriores sobre el problema de trozados de árboles en: el sector forestal, optimización integrada, localización de maquinarias entre otros temas asociados a esta Tesis de Grado. Se indagó en diversas fuentes de información tales como: Tesis, memorias de títulos, bases de datos como *SciendeDirect*, *ISI web of science*, *Oxford Journals*, etc., obteniendo así el estado del arte y la brecha de conocimiento sobre el tema.

El problema se abordó a través de la investigación de operaciones, área de las matemáticas nacida en la Segunda Guerra Mundial con fines bélicos, para asignar recursos escasos a las distintas maniobras militares y a las actividades que componían cada operación de la manera más eficaz (Hillier S., Frederick, 2010).

La investigación de operaciones es la herramienta matemática que permite formular y resolver un modelo matemático, considerando todas las restricciones de recurso escasos que existen para lograr maximizar los ingresos de la cosecha forestal y aserraderos, considerando un programa de optimización integrado.

Para aplicar reglas de trozado dinámicas a los bosques, se recurrió a la generación de columnas, método que consiste en resolver problemas de programación lineal donde las columnas (variables del problema) no son conocidas, o es difícil generarlas explícitamente y generalmente, usado en problemas que tienen un número exponencial de variables (Díaz Smith, Santiago, 2009). Considerando también, que a las reglas de trozado aplicadas a la cosecha del bosque, se le integro restricciones que reflejen el conjunto de decisiones a considerar, en la planificación táctica, tanto en la cosecha forestal como en los aserraderos.

Una vez planteado el modelo matemático, se implementó un programa computacional bajo la plataforma Cplex 12.6.1 de ILOG utilizando lenguaje de programación C++; generando un conjunto de instancias que simuló diferentes escenarios que permitieron generar conclusiones y extraer información para apoyar las decisiones tácticas del sector.

CAPÍTULO 2.

Antecedentes de los Procesos de la Cosecha Forestal

El presente capítulo se describe en detalle el problema de la cosecha forestal. Comenzando por destacar la importancia que ha adquirido la industria forestal por el aporte que realiza a la economía de diversos países alrededor del mundo y, sobre todo en Chile. Además se describe el ciclo forestal destacando las etapas que este posee, el rendimiento que se espera en los bosques y la transformación de rollizos a productos terminados.

2.1 Industria Forestal

La industria forestal es una importante fuente económica para muchos países, tales como Nueva Zelanda, Brasil, Canadá, Chile o Portugal. Esta industria, generadora de diversos productos derivados de la madera como: papel, cartón, celulosa, madera, entre otros; se ha ido internacionalizando e industrializando cada vez más, por lo que el mercado se torna cada vez más exigente. Si bien la demanda aumenta con la posibilidad de la exportación, también es más difícil la competencia; por lo que el perfeccionamiento en todo lo que abarca la industria forestal ha ido en aumento y sometida a constantes cambios, en comparación a décadas anteriores. (Zhang, Toppinen, & Uusivuori, 2014)

2.2 Industria Forestal en Chile

El sector forestal comprende el 1.8% del PIB nacional, lo que se traduce en casi dos mil millones de pesos. Es la segunda industria productora de divisas para el país, generando aproximadamente 130 mil empleos directos y más de 300 mil empleos indirectos; y en los últimos 15 años, representa el 13% del total anual de los retornos por exportaciones. (Corporación Nacional Forestal. CONAF., 2006)

La industria forestal se desarrolla desde la Quinta hasta la Décima Región. La Región del Bío-Bío es considerada el centro forestal más importante, debido a su gran superficie de patrimonio forestal. Esta región posee cerca del 44% de las plantaciones forestales del país, de las cuales 900 mil hectáreas son principalmente de *pinus radiata D.Don* y *eucaliptus Glóbulus*. Además, concentra el 76% de las exportaciones forestales a nivel nacional, y un 70% de la producción nacional de celulosa. También genera un 58% de la producción de madera aserrada y un 83% de la fabricación de tablero y chapas del país. (Instituto Forestal de Chile, INFOR., 2013)

Entre los productos generados a partir de los diversos trozos de árboles, se encuentra la pulpa, la madera aserrada, astillas, tableros y chapas, embalajes de madera, postes, polines y trozas de exportación. Cada uno de estos productos, abarca un porcentaje diferente en relación al total de trozos de madera producidos, dependiendo del tipo de árbol trozado.

En la tabla 2.1 se señala la cantidad de trozos en m³ utilizados por tipo de árbol en el año 2012.

Tabla 2.1 Cantidad de trozos por tipo de árbol utilizados por producto en m³.

Producto	Pino Radiata		Eucalipto y Otras Especies		Madera Nativa	
	Cantidad de trozas utilizadas en m ³ (x1000)	Porcentaje de trozas utilizadas	Cantidad de trozas utilizadas en m ³ (x1000)	Porcentaje de trozas utilizadas	Cantidad de trozas utilizadas en m ³ (x1000)	Porcentaje de trozas utilizadas
Pulpa	9320,1	35,28%	6233,9	50,60%	230,3	68,21%
Madera aserrada	13199,8	49,96%	395,6	3,21%	-	-
Astillas	466,3	1,76%	5265,6	42,74%	-	-
Tableros y chapas	3048,3	11,54%	362,6	2,94%	103,6	31%
Embalajes de madera	128,3	0,49%	13,1	0,11%	-	-
Madera hilada	-	-	37,5	0,30%	-	-
Postes y polines	254	0,96%	10,6	0,09%	-	-
Trozas de exportación	3,9	0,01%	1,5	0,01%	0,3	0,09%
Total	26420,8	100,00%	12320,4	100,00%	334,2	100,00%

Fuente: Instituto Forestal de Chile, INFOR., 2013.

2.3 Ciclo de la Industria Forestal

La industria forestal tiene como principal recurso la madera, cuya ventaja respecto a otros recursos naturales, es el ser renovable dentro de cierto período de tiempo. Por lo mismo, esta característica juega un rol fundamental al momento de planificar todo lo correspondiente a la industria de la madera.

La sustentabilidad se relaciona con la renovabilidad, esto se traduce en los múltiples factores que se deben considerar al plantar un bosque para su explotación comercial (excluyendo bosques nativos). No sólo se debe satisfacer la demanda, sino que también, mantener un trato “amable” con el medio ambiente, procurar la conservación del terreno para próximas plantaciones y cultivar un buen trato con las comunidades cercanas.

El ciclo de la madera que es destinada a su comercialización se puede dividir en 6 pasos; los cuales son; de gran importancia para obtener el mayor beneficio para la organización. Estos pasos son los siguientes:

1. **Habilitación y plantación del bosque:** Primera etapa del ciclo, la cual contempla la habilitación de terrenos para posterior plantación y las primeras aplicaciones de control químico. En esta etapa intervienen geólogos, genetistas, plantadores, ingenieros forestales, aplicadores de químicos, maquinaria de habilitación de terreno, entre otros.
2. **Mantenimiento del bosque:** Etapa en la cual, el bosque es tratado para obtener los máximos beneficios. Considera diferentes tipos de podas, correcciones de árboles y raleo.
3. **Cosecha del bosque:** Etapa que se centra en la extracción del bosque. Involucra a las faenas de cosecha forestal, las que a su vez, se encargan de la explotación de la totalidad del bosque, para su posterior conversión de los troncos en rollizos.
4. **Transporte de rollizos a diversos destinos:** En esta etapa los árboles cosechados son manipulados para luego ser transformados en rollizos. Dependiendo del tipo de árbol, algunos se descortezan y desraman antes de ser enviados a los clientes. Para transformarlos en rollizos, se les aplican diversas reglas de trozados con el fin de obtener el máximo beneficio del recurso y, además cumplir con los compromisos adquiridos con los clientes. Luego estos rollizos son enviados a una cancha de acopio para su transporte a los diferentes destinos.
5. **Transformación de rollizos a diversos productos:** En este paso es donde los rollizos son transformados en los diversos productos que se desean comercializar tales como pulpa, madera aserrada, astillas, tableros, chapas, entre otros. Una vez ya creados estos productos, se almacenan a la espera de su destino para su posterior comercialización.
6. **Comercialización de productos:** Esta es la última etapa del ciclo económico de la madera, donde los productos ya están terminados y son comercializados en diversas partes tanto a nivel nacional como internacional.

2.4 Rendimiento de los Bosques

Las decisiones a considerar en la planificación táctica de la cosecha forestal, están supeditadas a los compromisos adquiridos por las empresas forestales mandantes, estos son expresados en cantidad en metros cúbicos, (m^3 de rollizos). A su vez estos compromisos fueron

adquiridos por las empresas teniendo en conocimiento la cantidad de materia prima que poseen en los bosques.

Es importante para los productores de bosques que no poseen plantas industriales de manufactura y que son proveedores de trozos en el mercado, conocer anticipadamente: la cantidad, volumen, distribución y tamaño de las trozas posibles de obtener y, así determinar la mejor alternativa de venta de sus productos al momento de la cosecha y obtener el máximo retorno de sus bosques (Vidal, 2008).

En el sector forestal los bosques se dividen en rodales (división de los bosques por tipo y edad de los árboles) y las faenas de cosecha forestal dividen los rodales en parcelas. La Figura 2.1 representa la división de bosques, rodales y parcelas.

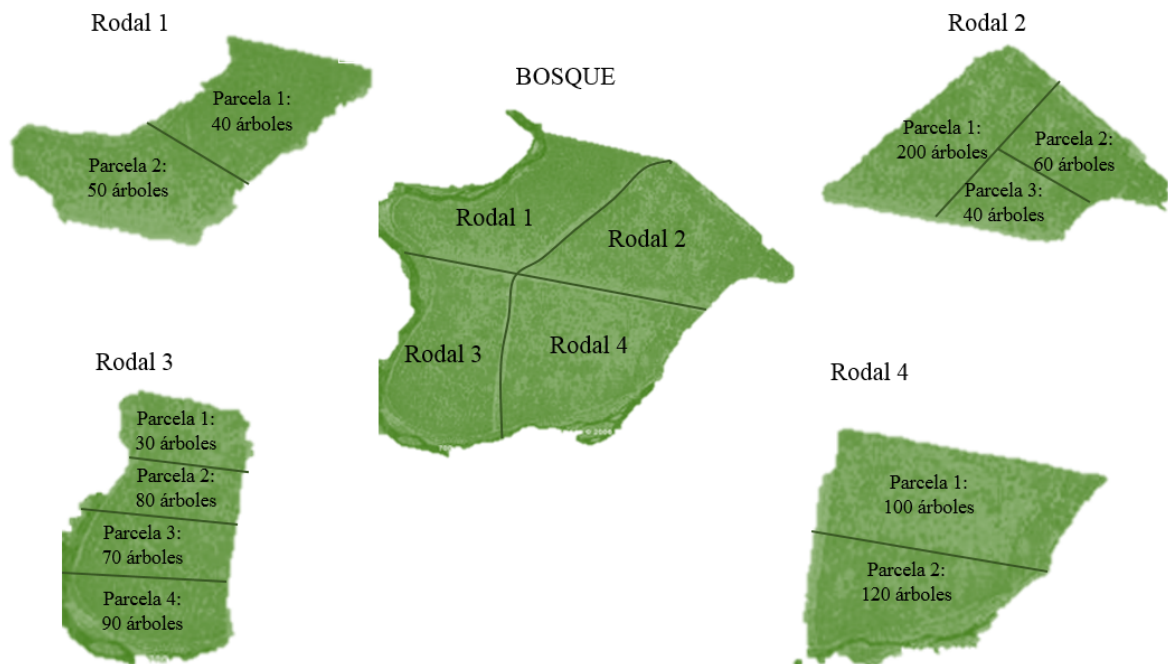


Figura 2.1: Esquema de creación de parcelas dentro de un rodal.

De cada árbol de las parcelas se obtienen los siguientes datos:

- Diámetro a altura de pecho, DAP: Es la medición de la corteza del árbol a la altura del pecho, 1,3 m. sobre el terreno. Para obtenerlo se utilizan dos instrumentos, una cinta diamétrica o una forcípula, Figura 2.2. En el caso de utilizar una forcípula se debe tomar, dos diámetros y obtener un promedio de estos, Figura 2.3.

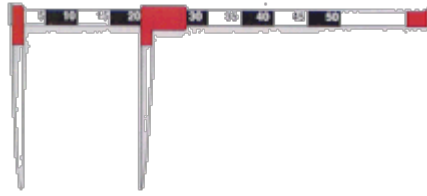


Figura 2.2: Forcípula.

Fuente: Departamento de Montes, 2004.

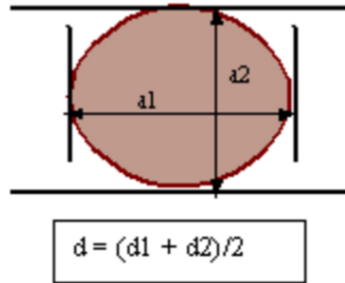


Figura 2.3: Medición con la forcípula de un árbol de sección no circular.

Fuente: Departamento de Montes, 2004.

- **Altura:** La medición de la altura de los árboles se puede desarrollar por varios métodos o instrumentos, tales como: tabla dendométrica, Blume-Leiss, Haga, el Relascopio Bitterlich, entre otros.

El DAP y la altura de los árboles además de permitir estimar la producción en m^3 que entregan los bosques, dan las características a los rollizos obtenidos (largo y diámetro;), las cuales permiten conocer qué tipo de productos se puede obtener de cada rollizo y buscar establecer que patrones de corte se asignaran, para satisfacer la demanda de los clientes con la menor cantidad de rollizos posibles.

2.5 Transformación de Rollizos a Tablas

Una vez que los bosques son cosechados y transformados en rollizos, son enviados a diversos destinos para ser procesados en: muebles, celulosa, pulpa, cartones, papeles, entre otros. Específicamente en la transformación de los rollizos a tablas, éstos son enviados a los aserraderos forestales que tienen como fin la actividad que implica obtener tablas de tamaños predeterminados de rollizos o troncos de madera (Hinostroza, Pradenas, & Parada, 2013).

Para iniciar la transformación, primero se realiza el descortezamiento del árbol (si es que éste no ha sido descortezado anteriormente en el bosque); luego los trozos de árboles son clasificados según su longitud y diámetro, posteriormente son cortados para producir tablas de diferentes tipos según sea el requerimiento del cliente (Pradenas, Garcés, Parada, & Ferland, 2013). La clasificación de la longitud y diámetro de los rollizos es primordial, ya que ésta permite crear un registro, para establecer el esquema de corte a crear para determinar los productos a generar, y también para establecer el rendimiento que éstos poseen, es decir, qué cantidad del rollizo se convierte en producto final, y qué cantidad del rollizo es desechada. (Maturana, Pizani, & Vera, 2010)

Una vez generada las tablas, se realizan diversos post-procesos dependiendo del producto final a obtener. Por lo general, se producen dos tipos de madera, seca y verde. Si las tablas se desean entregar secas, éstas deben pasar por hornos de secado. Por otro lado, si la madera se entrega en verde, se debe aplicar un baño anti manchas. Por último, se puede aplicar cepillado en las caras que se desee de la tabla, dependiendo del producto a obtener.

Las tablas generadas son almacenadas dependiendo del tipo de producto y de su calidad, característica que se mide por la cantidad de nudos, manchas, etc. Existen tablas, con cero defectos, que son comercializadas a un valor más elevado que el resto y que, en su mayoría, son destinadas a exportación. Por otro lado, las tablas que presentan defectos, su valor comercial es menor a las anteriores.

De acuerdo a lo anterior, la calidad de la tabla determina su valor comercial; por lo que los procesos anteriores al aserradero tales como: aplicación de químicos, poda, raleo, entre otros, son de vital importancia para obtener una mayor cantidad de madera de mejor calidad.

CAPÍTULO 3.

Marco Teórico y Estado del Arte

3.1 Proceso Forestal

El proceso de cosecha forestal ocurre cuando los diversos tipos de árboles (en su mayoría las diversas especies de pino y eucaliptus) son extraídos para ser transformados en rollizos y posteriormente en tablas para su comercialización.

Antes de la cosecha forestal existen varias etapas y procedimientos que son fundamentales para la obtención de la madera, éstas comienzan con la elección del terreno y del tipo de árbol a ser plantado, considerando aspectos: ecológicos, de suelo, ambientales, productivos, sociales, entre otros. (Baskent&Keles, 2005). Posteriormente, se prepara el terreno de manera manual o con maquinarias especializadas, con el fin de habilitarlo para la plantación de los árboles.

En los primeros años, se someten las plantaciones a controles químicos con el fin de fortalecer los árboles y contrarrestar posibles plagas. A medida que los árboles crecen, se aplican procedimientos relevantes para la comercialización, estos son el raleo y la poda.

- **Raleo:** Proceso que consiste en el retiro o extracción de ciertos árboles, ya sea porque no cumplen con las expectativas para su crecimiento o porque existen otros ejemplares que presentan mejores características. Se realiza para aprovechar mejor el suelo, los nutrientes y el oxígeno, entre otros factores, que aportan al desarrollo del árbol.

Existen dos tipos de raleo: el de desecho y el comercial.

- **Raleo de desecho:** Este raleo se realiza junto con la primera poda (4 a 6 años de edad), se le denomina de desecho ya que se eliminan los árboles que dentro de ciertos parámetros, no tienen ningún valor económico o comercial. (Davel M., Tejera L., & Honorato M., 2009)
- **Raleo Comercial:** Se denomina raleo comercial ya que los árboles a extraer pueden ser comercializados a pesar de su corta edad. (Sotomayor A., García E., & Helmke E., 2002). A su vez, se puede aplicar dos tipos de raleo comercial: el primero cuando el árbol tiene de 11 a 17 metros de altura (9 a 12 años de edad) y el segundo, u optativo, cuando el árbol alcanza una altura de 18 a 21 metros (13 a 16 años de edad).

- Poda: Este procedimiento se basa en la eliminación de las ramas de los árboles en su período de crecimiento, sobre todo cuando son jóvenes. La poda mejora la calidad de la madera porque elimina tallos secundarios y ramas pequeñas, también mejora el hábitat de crecimiento y aumenta la calidad y cantidad de la madera producida (Alvarez, Villagra, Villalba, & Debandi, 2013). Existen varios tipos de podas, denominadas por la altura del árbol (alta, media, baja) o en su defecto, determinada por el número de poda a la cual corresponde; las cuales eliminan cierta cantidad de ramas dependiendo de la altura del árbol tal como lo indica la Figura 3.1. (Sotomayor A. et al., 2002)

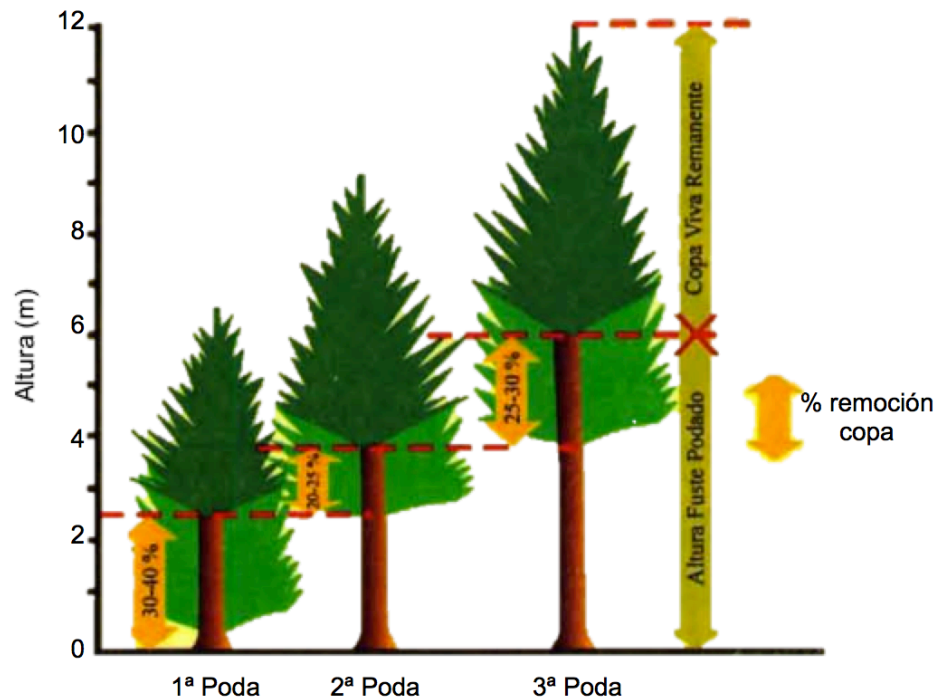


Figura 3.1 Alturas y porcentajes de poda en esquema intensivo.

Fuente: Sotomayor A. & García E., 2002.

Cuando los árboles alcanzan su madurez (alrededor de 22 a 28 años dependiendo de la especie y factores ambientales) (Weintraub & Romero, 2006), son cosechados en lugares predeterminados, denominados unidades de cosecha, las que son subdivisiones de los rodales y constituyen las áreas unitarias e indivisibles a cosechar.

El proceso de cosecha se realiza en un tiempo determinado, por empresas de servicios forestales (ESF). Dependiendo de las características del terreno, se utilizan métodos de cosecha terrestre (pendiente inferior o igual a un 30%), o aéreo (pendiente superior a 30%) (Lledó G., 2004). El método aéreo más utilizado es con torres de madereo, como lo indica la Figura 3.2

(Bossi P., 2007). Por su parte, la cosecha terrestre se realiza a través de máquinas llamadas *skidder*, maquinaria que se muestra en la Figura 3.3.

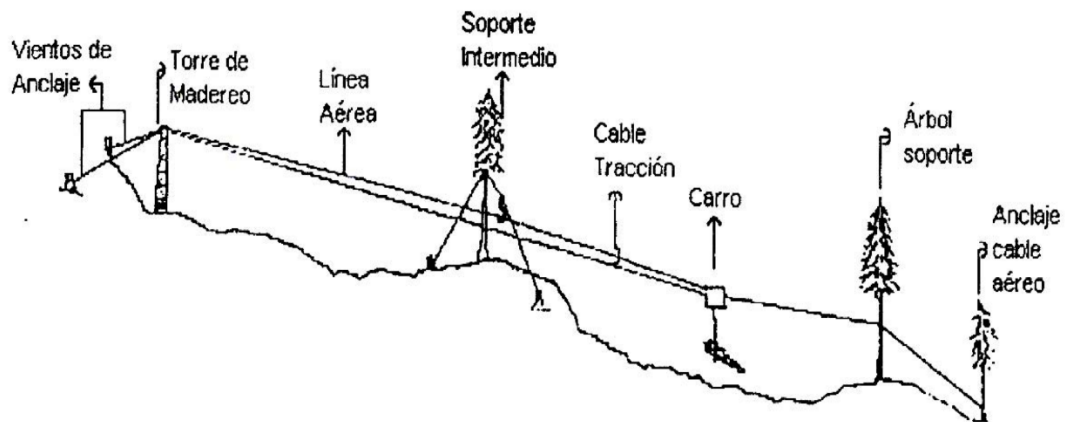


Figura 3.2 Esquema general de torres de madereo.

Fuente: Bossi P., 2007.

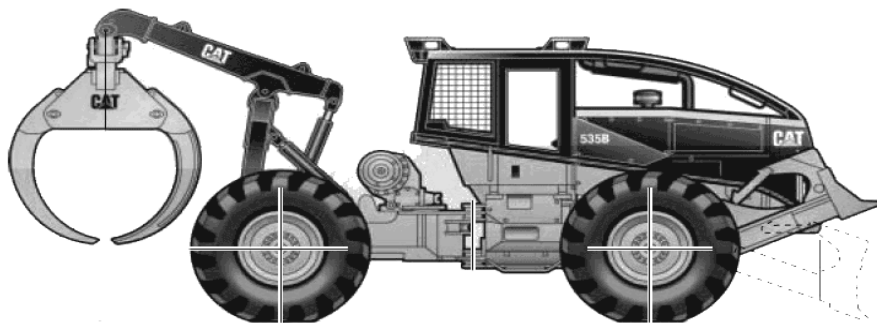


Figura 3.3 Maquinaria Skidder, encargada del método de madereo terrestre.

Fuente: Weintraub & Romero, 2006.

La cosecha consta de cuatro etapas: volteo, desrame, descortezado y madereo. El volteo es el corte del árbol en su parte inferior para ser extraído de la tierra. El desrame es la extracción de las ramas y conos de los árboles dejando sólo su fuste. El descortezado consiste en retirar la corteza del árbol; y por último el madereo consiste en la movilización de los rollizos a lugares predeterminados.

Existe un método de cosecha manual sin grandes maquinarias, como las torres de madereo ni *skidder*. Es realizado por motosierristas, encargados de voltear los árboles para luego acopiarlos y arrumarlos en una cancha especialmente diseñada para realizar el trozado de los fustes. Tanto el arrumado como el acopiado se realizan con animales (bueyes). Para el trozado

de los fustes se utilizan cuadrillas constituidas por: un motosierrista, encargado de trozar los productos, y dos ayudantes para desramar (hachero) y medir (huincher). La cuadrilla debe trozar siguiendo estrictamente una instrucción u orden de trozado conocida. (Lledó G., 2004)

Las maquinarias utilizadas por las ESF son variadas y frecuentemente son:

- Harvester: Maquina con un cabezal de tenaza compacta que: voltea el árbol, desrama, dimensiona, troza y apila los trozos dejándolos preparados para su transporte.
- Fordwarder: Transporta los rollizos y/o árboles hasta las canchas de acopio o a los camiones.
- FellerBuncher: Es similar a la de un Harvester pero, sólo tiene por función voltear y apilar árboles, no realiza las tareas de desrame ni trozado.
- Skidder: Posee una pala frontal, que empuja árboles, abre caminos a través de los bosques o limpia canchas. Ésta máquina puede transportar árboles enteros, fustes o trozos desde la zona de volteo hasta la cancha de trozado mediante la “garra trasera”.
- Trineumático: Sus funciones son variadas, todas asociadas al movimiento de trozos, árboles o fustes, los cuales ordena en las canchas de acopio hasta cargar los camiones que transporta la materia prima.
- Cargador Frontal: Maquinaria que efectúa el carguío mecanizado de los rollizos.
- Torre de madereo: Se utiliza en sectores montañosos para el transporte de los árboles hacia las canchas de trozados.

Una vez que los árboles son extraídos del suelo, son trasladados a las canchas de trozado, las cuales se ubican en lugares aledaños a la faenas de volteo, donde se realiza el trozado. Al poseer canchas de trozado se adquieren ventajas en el proceso, siendo las más considerables que éstas facilitan y ordenan las faenas en el bosque y, además reducen considerablemente el riesgo por accidente.

Cada fuste permite una gran cantidad de combinaciones de productos finales, por lo que la elección de la regla de trozado es fundamental. Las decisiones de ¿Cómo cortar el fuste?, se ven determinadas por las características que presenta un árbol. Algunas características relevantes de un árbol son: largo, diámetros, calidad y especie. Los productos que se pueden obtener de este árbol dependen de su largo y diámetro. (Epstein, Nieto, Weintraub, Chevalier, & Gabarró, 1999a)

Las decisiones tomadas en las canchas de trozados adquieren relevancia porque determina el producto a alcanzar en procesos posteriores, entonces deben estar en concordancia con la demandas existentes. Las decisiones de trozado no son reversibles, por lo que una mala decisión en la regla a aplicar, provoca la no generación de un producto final, lo que acarrearía consigo la insatisfacción de la demanda y la creación de un producto no comerciable.

La Tabla 3.1 presenta un ejemplo de regla de trozado:

Tabla 3.1 Ejemplo de regla de trozado

Pieza	Largo (m.)	Diámetro Mínimo (cm.)	Destino
Primera	12,1	26	Exportación
Segunda	4,1	24	Aserradero
Tercera	4,1	22	Aserradero
Cuarta	4,1	16	Aserradero
Quinta	2,44	8	Celulosa

Fuente: Epstein et al., 1999.

La Tabla 3.1 muestra un ejemplo de regla de trozado, en la cual se distinguen dos características para diferenciar los productos: el largo y el diámetro. Los productos están ordenados de acuerdo a la prioridad respecto de su valor comercial (primera, segunda, ... quinta), se desea obtener la mayor cantidad de piezas de primera, largo 12,1 metros y diámetros de 26 centímetros, ya que se obtienen mayores réditos. Sin embargo, obtener productos de primera no siempre es posible debido a las exigencias en su diámetro y largo (requisitos difíciles de cumplir cuando los parámetros son mayores). Los árboles son trozados respecto a las características y demandas a satisfacer.

La Figura 3.4 es un ejemplo de las piezas a obtener de un árbol, en este caso son dos piezas. La primera pieza, destino aserradero y la segunda pieza, más delgada, con destino celulosa.

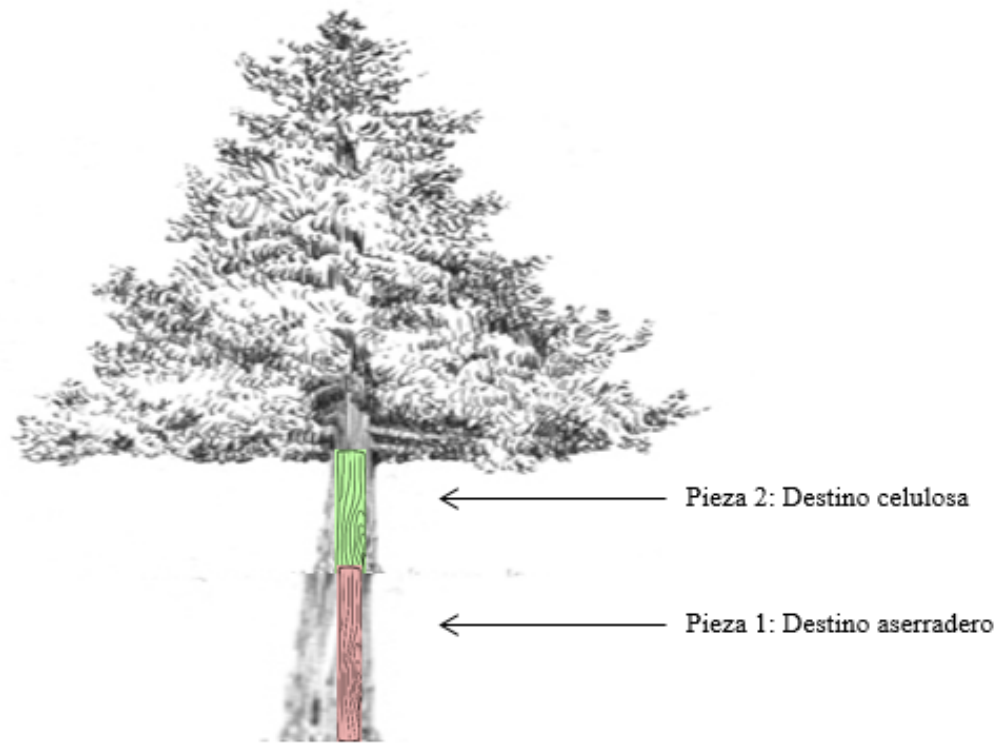


Figura 3.4 Ejemplo de rollizos obtenidos de un árbol.

Los clientes que Chile abastece, se pueden clasificar en tres tipos: (Epstein et al., 1999a)

- Exportación: Las piezas de mejor calidad son destinadas a diversos países como: Corea, Japón, Turquía, entre otros. Las características de los rollizos son más exigentes, por ejemplo un diámetro mayor de 30 cm, libre de nudos, largo mayor a 12 metros, etc.
- Aserraderos: Los productos en los aserraderos son de calidad menor, el diámetro de éstos corresponde a más de 16 cm. Este material es consumido por aserraderos locales para la producción de diversos productos como: paneles, molduras, maderas, entre otros.
- Celulosas: Estos clientes reciben los productos de menor calidad, debido a que demandan materia prima para la creación de productos que no necesitan mayores exigencias.

Un ejemplo de una solicitud de uno de los clientes de exportación, en este caso Corea, es mostrado en la Tabla 3.2:

Tabla 3.2 Ejemplo de orden de exportación a Corea.

Tipo de Pieza	Largo (m.)	Diámetro mínimo (cm.)	Diámetro promedio (cm.)	Volumen mínimo (m ³)	Volumen máximo (m ³)	Precio (\$/m ³)
Largo	11,1	20	26,7	18000	0	65
Mediano	7,4	20	26,7	0	0	65
Corto	5,5	20	26,7	0	2,4	65
Corto	3,7	20	26,7	0	3,6	65

Fuente: Epstein et al., 1999b.

Las solicitudes de los clientes son diversas y cada una posee información diferente (largo, diámetro, tipo de árbol, libre de nudos, entre otras). Estas solicitudes se deben satisfacer respecto de un volumen mínimo y un volumen máximo, por lo que los requerimientos deben ser en un valor que esté entre el rango determinados por estos volúmenes.

Junto con la clasificación se realiza el cálculo de la cantidad de madera a obtener. Este proceso se denomina cubicación de madera, es realizado antes de la cosecha forestal para obtener una estimación del bosque y posterior a la cosecha, para visualizar la real producción del bosque y medir los volúmenes de los productos a ser entregados a los clientes. Por lo general las solicitudes de los clientes son realizadas en dos unidades de medidas: metros cúbicos, m³, utilizada por el sistema de medición internacional y pies tablares, pt, utilizada mayoritariamente por Estados Unidos.

Una de las dificultades que se presenta en la cubicación, es conocer cuántos metros cúbicos representa un producto de los solicitados por los clientes, ya que las características de los productos solicitados no tienen medidas iguales, ni en diámetro ni en largo. (Ver Tabla 3.2)

Existen diversas fórmulas para medir el volumen. Dos de las más utilizadas son las fórmulas de Smalian y de Doyle. Smalian utiliza el sistema métrico internacional, es decir, el valor obtenido se entrega en m³, observar Fórmulas (3.1) a la (3.3).

$$V = \frac{\pi \times (D_x)^2 \times L}{4} \quad (3.1)$$

Donde:

V = Volumen en m³.

L = Largo en troza en metros.

D_x = Diámetro promedio de la troza en metros.

Siendo D_x

$$D_x = \frac{(D_a + D_b)}{2} \quad (3.2)$$

D_a = Diámetro mayor de la troza.

D_b = Diámetro menor de la troza.

La fórmula de Doyle entrega, los valores obtenidos en pt. (Fórmula 3.2).

$$V = \frac{(D_{menor} - 4)^2 \times L}{2} \quad (3.3)$$

V = Volumen en pies tablares

D_{menor} = Diámetro menor en pulgadas

L = Largo de troza en pies

Para la medición del diámetro menor de la troza, se disminuye el espesor de la corteza y además, la medición del largo debe tener el descuento por el despunte.

Junto con la evidente diferencia métrica, la diferencia más significativa de las dos fórmulas es la manera que consideran los diámetros. Por una parte, Smalian considera el diámetro promedio de la troza y, Doyle considera solo el diámetro menor. Por lo anterior, es necesario conocer la relación que existe entre pt y m^3 , cuando, solo se tiene el diámetro menor de una troza. (Figura 3.5)

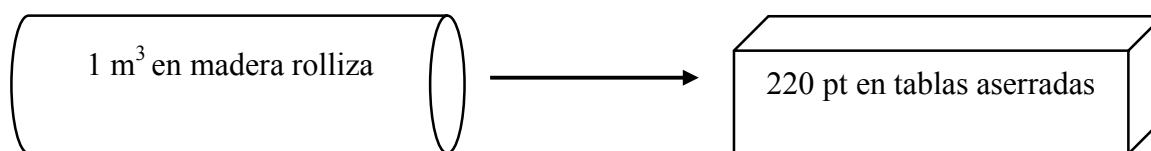


Figura 3.5 Relación entre 1 m^3 en madera rolliza y tablas aserradas en pt.

Fuente. Cuñachi Encinas, 2012

El fin último de las plantaciones forestales es alcanzar los objetivos de producción que permitan obtener la mejor rentabilidad para el productor, en concordancia con otros objetivos como: la conservación de los suelos; protección de cursos de agua; fauna; belleza escénica y otros. (Sotomayor A. et al., 2002). Este fin no ha sido modificado durante los años, se debe cumplir demandas que exigen diversas características siendo las más solicitadas: el diámetro, rollizos libre de nudos y largo. Esta decisión ha sido más complicada de tomar respecto a años

anteriores debido que, las faenas de establecimiento y manejo forestal, específicamente podas y raleos, permiten corregir los bosques y obtener mejores ejemplares, resultando así árboles de mejor calidad. (Vargas, L., Suárez, L., Robles. I., & Mercado, R., 2004).

La demanda existente, es conocida y se maneja dentro de un rango de volumen. Las empresas la distribuyen en diversos períodos de tiempos, como quinquenios (periodo de cinco años), anuales, o de la manera que más se acomode. Por su parte, la demanda correspondiente a la cosecha forestal no es balanceada respecto del período de tiempo utilizado por lo que en estos casos, es necesaria la existencia de inventarios para suplir la carencia, o almacenar el exceso. (Rönnqvist, 2003).

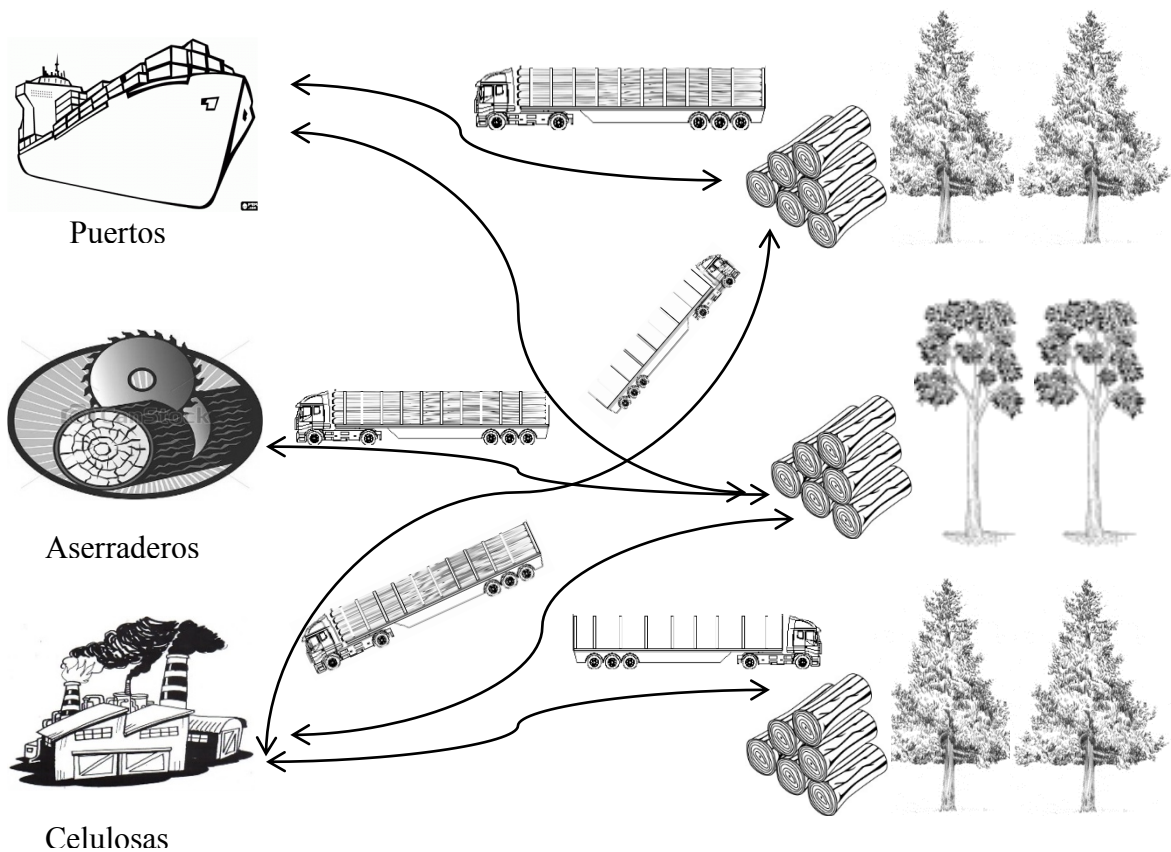


Figura 3.6 Transporte de productos a sus destinos.

Fuente: Weintraub & Romero, 2006

Una vez que los rollizos están disponibles para ser entregados a sus diversos clientes, aserraderos, exportación y celulosas, son transportados a sus destinos mediante camiones para ser usados, tal como muestra la Figura 3.6.

Tomando en cuenta el proceso realizado en los aserraderos, los rollizos son ordenados, almacenados y aserrados en tablas. Para lograr la transformación de rollizos a tablas se realiza el proceso de desramado y descortezado, (en el caso de que estos procesos no hubieran sido realizados en las faenas de cosecha), para luego entrar al proceso de aserrío. Las tablas, se obtienen desde los rollizos descortezados mediante cortes de guillotina en forma longitudinal, proceso que también es conocido como “cutting” donde los rollizos pasan por diversas sierras en las cuales se obtienen piezas laterales y centrales.

Del mismo modo que se ha logrado optimizar la calidad de los árboles obtenidos mediante el tratamiento en su crecimiento con trabajos que van desde la plantación del árbol, pasando por los diversos tipos de poda y raleo, los aserraderos han debido buscar la manera de obtener la mayor cantidad de productos posibles de los rollizos, generando la menor cantidad de desperdicios. Para lograr esto, se han enfocado en la mejora de la programación de la producción en los aserraderos, que consiste en la programación de las piezas de registro (rollizos que se registran y separan según, sus características al entrar al aserradero) para ser procesado, teniendo en cuenta el diámetro, longitud y grado de cada pieza de registro, por una parte, y los productos terminados que son necesarios para cumplir con los pedidos, por la otra. Al entrar la materia prima al proceso de aserrío, por lo general son todas piezas de registro de diferentes diámetros, longitudes y grados. La longitud de la pieza de registro y el patrón de corte, que es como se cortará el rollizo, determinan los productos que se generan y, también, el rendimiento, que es la cantidad de volumen del registro como productos terminados. Por otro lado el resto son residuos, los grados de troncos determinan la calidad de los productos que se pueden obtener a partir de ese registro. Por ejemplo, la madera para muebles debe provenir desde un registro con un muy buen grado. Madera para una caja, por otro lado, puede venir de un registro de bajo grado.

Con el fin de maximizar el rendimiento (minimizar los residuos), los planificadores de producción de los aserraderos tratan de utilizar un conjunto de patrones de corte que optimice el rendimiento para un diámetro de pieza de registro dado, y que estos sean lo más parecidos posible, lo que se llama patrones óptimos de corte. Estos patrones de corte se obtienen mediante la determinación del personal que asigna un patrón de corte para cada rango de diámetro, para así obtener el mejor rendimiento de los rollizos. Dado que cada patrón de corte determina un conjunto de productos que se generarán, como se muestra en la Figura 3.7, cualquier patrón de corte particular, generará un número de productos. Para complicar aún más las cosas, las

propiedades de la madera son diferentes a lo ancho del registro, por lo que algunos productos provienen del borde exterior, que tiene mejores propiedades que el centro del tronco (Maturana, Pizani, & Vera, 2010).

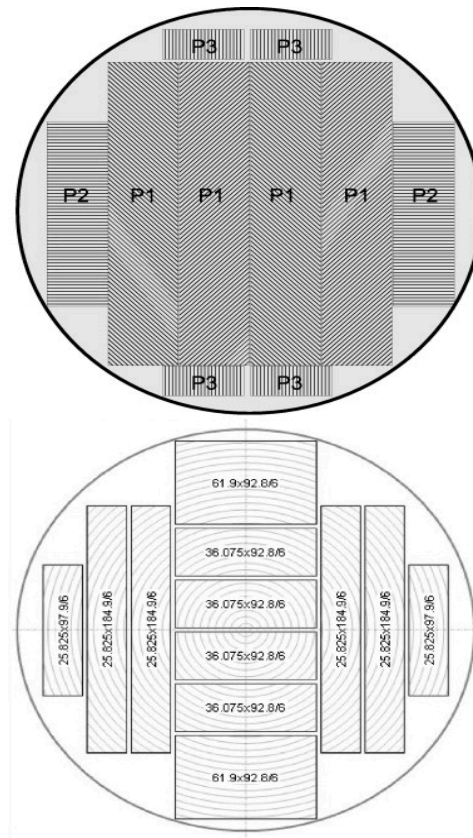


Figura 3.7: Ejemplos de patrones de cortes en los aserraderos.

Fuente: Maturana et al., 2010

3.2 Planificación en la Cosecha Forestal

El proceso de planificación de la cosecha y transporte en una temporada se basa en la aplicación de metodologías de cálculo y selección de tecnologías de producción, que responda a los cambios en las variables del terreno, bosque, maquinarias y parámetros económicos existentes (Carey, P., 1997).

Sapunar, P., Mansilla, B., & Fuentealba, G., (1999) señalan, que la planificación de la cosecha es la etapa previa, antes de ejecutar cualquier acción posterior. Ésta incluye la recopilación de información, su análisis, la toma de decisiones y la confección del programa. Además, mencionan que los objetivos dependen del planificador. En el caso de una empresa

forestal, pueden estar dirigidos a mejorar la productividad de la cosecha, las utilidades, la calidad de los productos, la seguridad en el trabajo y minimizar el impacto al medio ambiente.

La cosecha forestal, consiste de un conjunto de actividades, estas actividades pueden dividirse en las siguientes tres etapas:

- Administración forestal y cosecha: son todas las actividades que corresponden a las transformaciones y procesos a realizar en el bosque y lugares asociados.
- Transporte y ruteo: son las actividades que involucran el traslado de los rollizos hacia los clientes a través de camiones, creación de caminos con maquinaria pesada, selección de la mejor ruta para los camiones, etc.
- Producción: es la etapa que reúne a todas las actividades que involucran la transformación de los rollizos a productos finales para ser comercializados.

En todas las etapas, se toman decisiones, teniendo en cuenta que en cada una de éstas existen diversos tipos de planificaciones, siendo el esquema de planificación jerárquica el más utilizado. La planificación jerárquica considera por lo general tres planificaciones: la estratégica que es la encargada de tomar decisiones a largo plazo como de, índole financiera, análisis de proyectos, etc.; la planificación táctica o intermedia, que es la que recibe información de la planificación estratégica para guiar las actividades a realizar; y, por último, la planificación operativa o primera línea, que es la que lleva a cabo las actividades.

Las planificaciones se dividen para abordar de mejor manera los problemas. Existen herramientas que permiten ayudar a la toma de decisiones en todas las etapas, ya sea a nivel vertical (dentro de la misma planificación), u horizontal (involucrando a más de un tipo de planificación), siendo estas últimas soluciones las que más aportan al objetivo final. Sin embargo, son las más difíciles de llevar a cabo debido que requieren de un flujo de información expedito, entre las partes involucradas en las diversas planificaciones, además, de una coordinación difícil de encontrar debido a que se trata de partes que trabajan en aspectos y de manera diferente.

Rönnqvist, (2003) plantea, las tareas y decisiones que tienen que tomar los diversos niveles de planificación y además añade la planificación online. La Tabla 3.3 presenta las decisiones para la planificación.

Tabla 3.3 Niveles de planificación.

	Administración Forestal y Cosecha	Transporte y ruteamiento	Producción
Planificación estratégica	Plantación o siembra, cosecha de largo plazo	Construcción de caminos, mantención de caminos, Gestión de la flota	Planificación de inversiones.
Planificación táctica	Plan de cosecha anual	Mantención de caminos, utilización de equipos.	Plan de producción anual.
Planificación operativa	Programación de equipos de trabajo, secuencia de cosecha	Áreas de carga, planificación de retornos.	Tamaño de lote, programación.
Planificación online	Trozado	Despacho de camiones	Control de proceso, corte de rollizos.

Fuente: Rönnqvist, 2003.

Para dar una solución a las decisiones que deben tomar los diferentes tipos de planificaciones, se utiliza la programación matemática. Generalmente se acude a la programación lineal, entera, entera mixta o dinámica, la cual resuelve una formulación matemática que representa la problemática a tratar. También se requiere un método de solución apropiado que permita obtener soluciones, ya sean óptimas o buenas, en un tiempo acorde a la necesidad de la planificación. Las necesidades varían de acuerdo al tipo de planificación, así como la planificación estratégica, toma decisiones a largo plazo, puede esperar cierto tiempo a obtener soluciones. En cambio, las decisiones que toma la planificación operativa se deben tomar, en segundos ya que ellos se encuentran en la primera línea de las actividades.

Las planificaciones y sus modelos de solución, varían según la empresa, ya sea por diferentes recursos, costos, necesidades y otros factores que influyen en un modelo no sea usado por más de un empresa ya que cada una tiene un contexto diferente a otra.

3.2.1 Planificación Estratégica Forestal

La planificación estratégica forestal requiere de decisiones de tipo espacial y temporal. Estas decisiones son muy difíciles de tomar por la magnitud y variabilidad involucrada, pueden ser pocas hectáreas hasta miles como también, se puede tratar de diez a cientos de años. A lo anterior se agregan factores, como la producción de madera, la producción del agua, habitantes cercanos, estándares medio ambientales, entre otros (Baskent, Başkaya, & Terzioğlu, 2008).

La planificación de la producción forestal busca obtener, volúmenes constantes y sustentables en el tiempo que generen, la posibilidad de volver a utilizar la tierra, junto con maximizar el valor de la empresa teniendo en cuenta sus ventajas y limitaciones, conociendo sus recursos, demandas a satisfacer, entre otros múltiples factores. La planificación estratégica busca tomar decisiones a largo plazo se trabaja con cientos de miles de hectáreas y para 10 a 15 periodos de tiempo. Se usan los software FORPLAN en Estados Unidos y FOLPI en Nueva Zelanda.

La planificación estratégica en el ámbito del transporte y ruteo, se basa en la construcción y mantención de caminos para facilitar el acceso a los bosques. Para representar las decisiones de ruteo se utilizan las variables binarias 0 y 1. La complejidad de estos problemas lleva a que en la solución además, de la programación matemática se utilicen métodos heurísticos o meta heurísticos.

La planificación estratégica también, incluye las actividades de establecimiento y manejo forestal. Establecimiento, se preocupa de la plantación forestal, de la densidad de árboles que tendrá el terreno y de la decisión sobre si plantar uno o más rodales mientras, que el manejo forestal se preocupa del ¿Cuándo?, ¿Cómo? y el número de podas a realizar al bosque, así como también la manera de aplicar el raleo de desecho y en determinar si se aplica raleo comercial y de ¿Cuántas veces se debe realizar el raleo comercial? y ¿En qué edad del bosque?, etc. Estas decisiones estratégicas permiten obtener una mejor calidad de árbol porque elimina los nudos y los fortalece, dando paso a productos finales de mejor calidad.

3.2.2 Planificación Táctica Forestal

La planificación táctica se preocupa de las decisiones a mediano plazo, alrededor de 2 a 3 años, debe procurar generar decisiones sustentables que no afecten al entorno ni a los recursos utilizados, se requiere elegir las mejores formas de extraer la madera, las técnicas y evaluar todas las ventajas y desventajas de sus decisiones (Ghajar & Najafi, 2012). En las decisiones de la planificación táctica, ésta la forma de extraer la madera, decidiendo sobre los diversos métodos disponibles, eligiendo el más apropiado y que entregue el mayor beneficio a la empresa. También ve la generación de reglas de trozados con maquinarias para obtener la mayor cantidad de productos demandados con una menor cantidad de desperdicio de materia prima.

La planificación táctica, incide en las decisiones de transporte y en el mantenimiento de los caminos que permiten el acceso a los bosques sobre todo, en épocas en las cuales su acceso

se hace difícil como, lo son en temporada de invierno y primavera. Además, debe manejar de buena manera los recursos posibles, entendiendo como recursos toda la maquinaria necesaria para la extracción del bosque tales como: *skidder*, camiones, *forward*, etc.

Por último, los aserraderos son los encargados de transformar los rollizos a tablas. En el nivel táctico, se toman decisiones sobre: el nivel de producción a realizar: controla del nivel de inventario, manejo de los recursos existentes y satisfacer los compromisos de demanda que ha adquirido la institución.

3.2.3 Planificación Operativa Forestal

En la planificación operativa forestal, se realizan todas las actividades para: la extracción del bosque y toma de decisiones en el lugar de cosecha que involucran, la implementación de la faena diaria. Se definen también, implementos de trabajo, así como el análisis de la seguridad de las faenas, entre otras.

Para comenzar la planificación operativa, se debe tener claridad sobre los recursos a utilizar y que se encuentran disponibles, para definir si la faena, será totalmente mecanizada con el uso de *skidder*, *harvester*, torres de maderero, entre otras maquinarias o por el contrario, si se realizará de forma manual, algunas de las tareas, utilizando cuadrillas de cosecha y/o yuntas de bueyes.

Una vez extraído el bosque, se deben tomar las decisiones de trozados de los rollizos primero, decidiendo si se descortezará el árbol en terreno o se dejará esa labor para los aserraderos para luego tomar, las decisiones de trozado o *bucking*, cuyas labores se pueden realizar en el lugar de, cosecha, canchas aledañas o en su defecto en los aserraderos.

Cuando el corte de rollizos se realiza de manera automatizada, se escanea el rollizo y se identifica la mejor forma de trozado, la cual puede ser especificada de manera manual por el operario que utiliza la maquinaria. Estos sistemas computacionales, están basados en programación dinámica para determinar la forma de trozar el árbol dividiendo, el árbol en secciones de 1 a 10 cm. (Rönnqvist, 2003).

La planificación operativa, en las empresas de transportes juega un rol importante, ya que son los encargados de distribuir los rollizos desde los diversos puntos de cosecha hacia los clientes. Una de las decisiones que debe considerar, es el lugar de carga es decir, definir los puntos de cosecha con el objetivo de minimizar los tiempos y costos de transportes. Esta

decisión, es difícil de tomar por su complejidad pero sobre todo, por los costos que acarrearía una mala decisión. Este problema, a pesar de ser similar a un problema clásico de ruteo en donde existen nodos con ventanas de tiempo, es diferente ya que cada nodo se puede visitar más de una vez.

La transformación de rollizos a productos finales, varía según la empresa y el producto a obtener. En un aserradero, las decisiones son los esquemas de corte a adoptar, sobre los rollizos para obtener diferentes tipos de tablas considerando además, las demandas a satisfacer, como por ejemplo, una papelería tienen que decidir sobre los tipos de cartones y papeles a producir y los tipos de rollizos y cantidad que se asignarán a la producción.

3.3 Revisión de la Literatura

Las reglas de trozado históricamente se han establecido mediante la experiencia de las personas que se desempeñan en el rubro o mediante diversos programas computacionales los cuales solo consideran las características que presentan los árboles. Aquí es la primera diferencia del estudio con respecto a otros, ya que propone la utilización de reglas de trozado dinámicas, las cuales se generan considerando diversas características del problema tales como, demanda, costos de producción, costos de transporte, características de los rollizos, características de los rodales, entre otros. Al considerar múltiples factores, la programación matemática ayuda a tomar decisiones más certeras en la planificación táctica.

Dentro de la literatura se encuentran, diversos estudios relacionados con el tema de la cosecha forestal. Vega, R., (2000) propone, un algoritmo base para el trozado de árboles, en el cual solo considera una regla de trozados para cada rodal y sin tomar en cuenta, los costos asociados al transporte de rollizos a los diversos clientes. Rebolledo, R., (s. f.) también se centra en el trozado de árboles, si bien considera el traslado de los rollizos, no integra en las decisiones de la planificación, a los aserraderos.

Peñailillo, F., (2004) presenta un primer modelo matemático en el cual trata sobre la planificación agregada en los aserraderos, sin embargo excluye de esta, la cosecha forestal y el flujo de rollizos de está considerándolos solo, como un parámetro de entrada en el modelo. Baesler, et al., (2004) analizan, como optimizar el cuello de botella presentado en un aserradero junto con la forma de optimizar los esquemas de corte. No integran los rollizos ingresados desde el bosque, ni la capacidad de inventarios en los aserraderos y tampoco, la posibilidad de satisfacción de demanda con productos de mejor calidad. Otro estudio que considera la

producción en los aserraderos es el de Romero, et al., (2004), quienes proponen un modelo matemático para el cumplimiento de las demandas adquiridas por el aserradero, junto con la capacidad de inventario, dentro del aserradero no considera los esquemas de cortes, ni la materia prima que ingresa, el modelo solo abarca al requerimiento adquirido por el aserradero. Por su parte Garcés, J., (2007) expone un modelo matemático que optimiza los esquemas de corte en los aserraderos utilizando meta-heurísticas, solo se centra en los aserraderos y como poder minimizar el desperdicio en la transformación de los rollizos a tablas. Por ultimo en el estudio de Chavez, et al., (2008) proponen, un sistema integrado entre cosecha forestal y aserraderos utilizando teoría de juegos, consideran la cadena de suministros y los agentes que intervienen en estas sin embargo, no consideran las reglas de trozado ni los esquemas de corte , solo se concentran en la manera que fluye los productos a través de esta cadena, aunque destaca que integran las decisiones entre cosecha foresta, aserraderos y ventas.

CAPÍTULO 4.

Descripción del Problema, Modelamiento y Propuesta de Solución

En el presente capítulo, se presenta la descripción del problema, la forma de resolver el problema planteado considerando el modelo de programación matemática y las correspondientes estrategias de solución.

4.1 Descripción del problema

Para la realización del modelo matemático se considera una empresa de servicios forestales (ESF), la cual tiene derecho para la extracción de una cierta cantidad de bosques para realizar la faena forestal denominada *cosecha*. Dentro de los bosques existen los rodales (clasificación de los arboles por edad con respecto al año de plantación y tipo de árbol) y, dentro de los rodales se encuentran las parcelas (división de los rodales de uso exclusivo para las faenas de cosecha forestal). Una vez transformado en rollizos los árboles, se le aplican reglas de trozados. Posteriormente los rollizos son enviados a los clientes, en donde destacan los aserraderos, clientes que dentro de sus decisiones de planificación deben definir qué tipos de esquemas de corte se le aplicará a los rollizos para obtener los diversos tipos de tablas o productos.

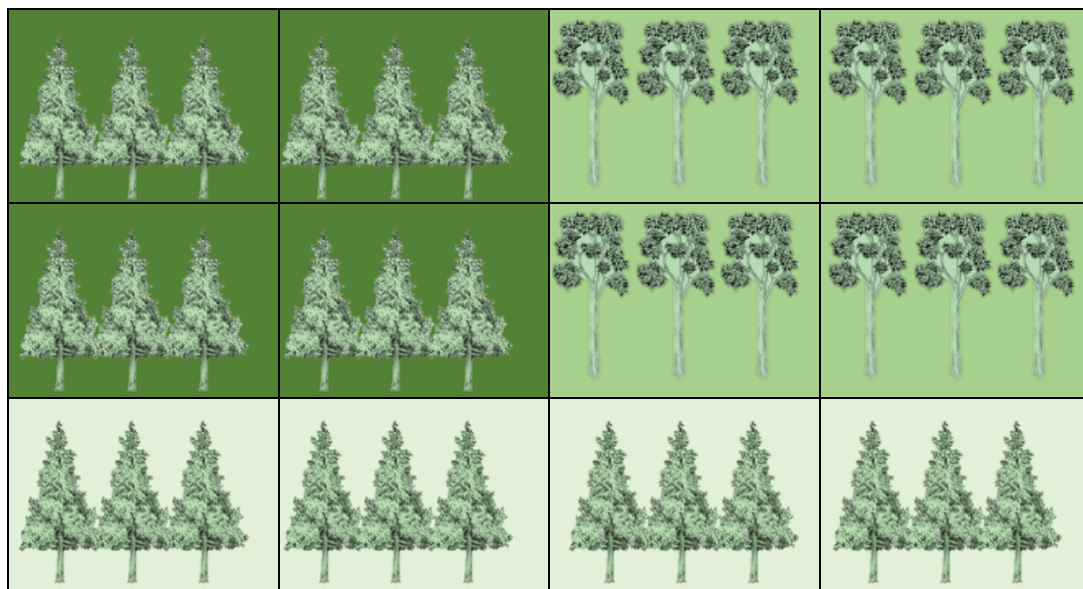


Figura 4.1 Representación Bosques, rodal y parcelas de cosecha.

En la Figura 4.1 se distingue un bosque con la existencia de dos especies forestales, *pinus radiata* y *eucaliptus nitens*. Dentro de este bosque se observan tres rodales, los cuales se

diferencian por el color del fondo. Por último, dentro de los rodales se observan cuatro parcelas de cosecha, siendo esta última, la división utilizada para planificar la faena, tal como se explica en capítulos previos.

A cada rollizo obtenido desde los bosques, se le asignan tres características: largo en metros; diámetro mínimo y máximo medido en centímetros, como se observa en la Figura 4.2. Por su parte los clientes solicitan que los productos posean, dos características diámetro mínimo medido en centímetros y largo medido en metros, tal como se mencionó en la Tabla 3.1 del Capítulo 3.

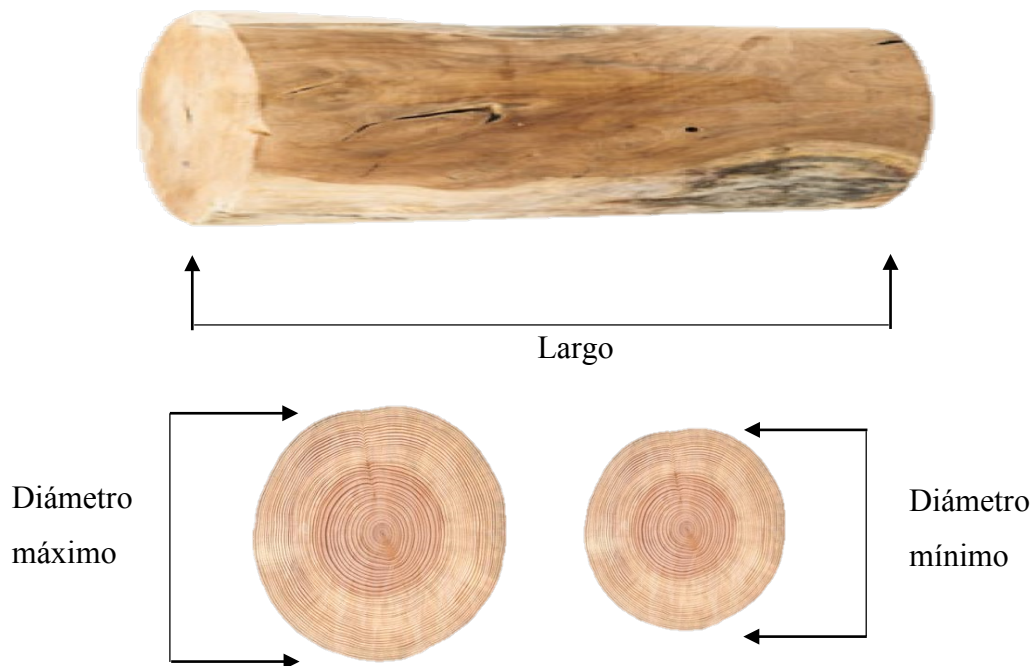


Figura 4.2 Representación largo, diámetro mínimo y diámetro máximo de un rollizo.

Para generar las reglas de trozados se consideran las siguientes decisiones:

- Maximizar la utilidad de los rollizos.
- Minimizar los desperdicios.
- Características solicitadas por los clientes (diámetro y largo).
- Demanda adquirida por la empresa con sus clientes.
- Cantidad de árboles disponibles en los bosques.

Los rodales abastecen diversas canchas de trozados. Éstas pueden ser abastecidas por uno o más rodales. Desde estas canchas de trozados se despachan los rollizos a los diferentes

clientes. También las canchas de trozados cumplen con la tarea de almacenamiento de rollizos para la satisfacción de demandas futuras.

La Figura 4.3 grafica el flujo de productos existentes desde el bosque hasta los diversos clientes. La primera parte hace alusión a los diversos rodales que abastecen las distintas canchas de trozado. Una vez que los productos están en la cancha de trozados y se ha aplicado el proceso del mismo nombre, los rollizos son enviados a los diversos clientes (exportación, aserraderos, celulosas, entre otras) destacando, que una cancha de trozado puede abastecer a más de un cliente.

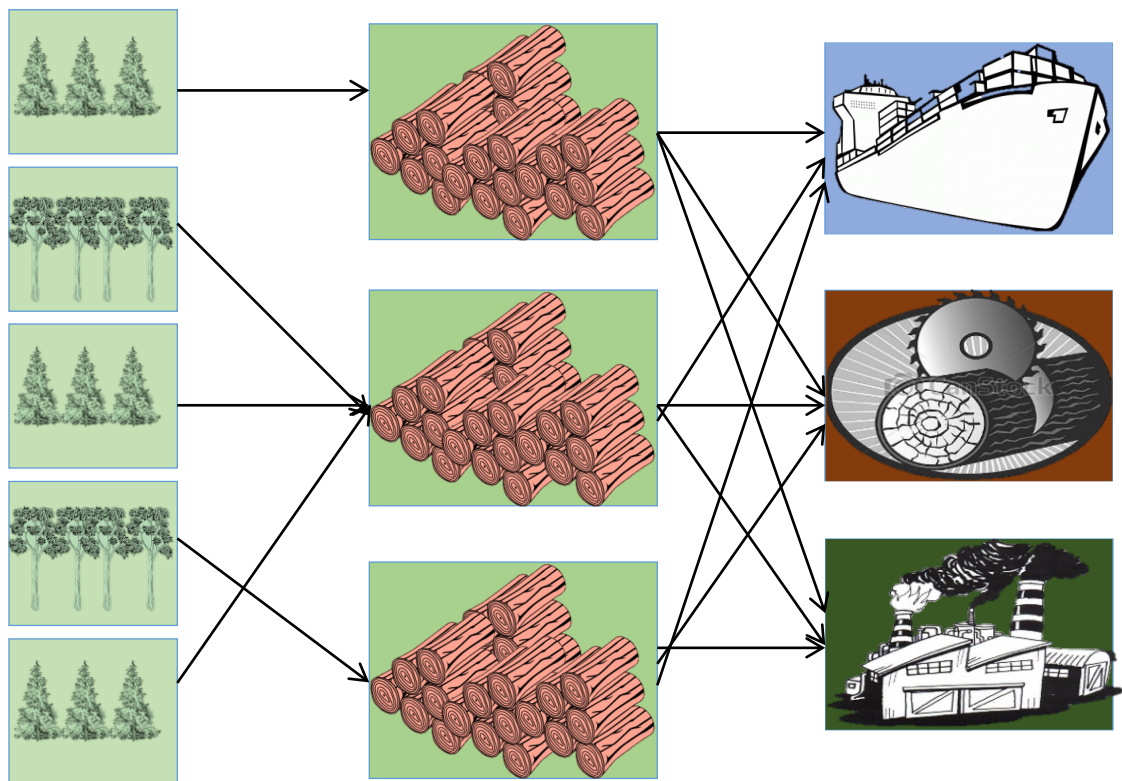


Figura 4.3 Flujo de productos de los bosques hacia los clientes.

La Figura 4.4 esquematiza la continuación del flujo de productos de la Figura 4.3, centrado únicamente en el flujo en los aserraderos. Comienza por transformar los rollizos en tablas con diversos requerimientos, según el cliente de destino. Como por ejemplo, tablas para exportación, centros comerciales o fabricación de diversa mueblería.

Los diversos productos forestales se obtienen a partir de los rodales por medio de la cosecha forestal, que produce rollizos de madera para entregar como materia prima a los diversos clientes. Los diversos clientes del bosque que adquieren madera, son los encargados de transformar los rollizos en los productos finales para posteriormente comercializarlos. Dependiendo del tipo de producto a obtener (planchas, madera aserrada, tablas, chapas, pulpa, astillas, entre otros) se solicita, el tipo de rollizo por esto, la importancia de las reglas de trozados, para productos con un mayor valor comercial se utilizan rollizos de diámetro y largo mayor en cambio, para productos con menor valor comercial como la pulpa, se utilizan rollizos de menor diámetro y largo.

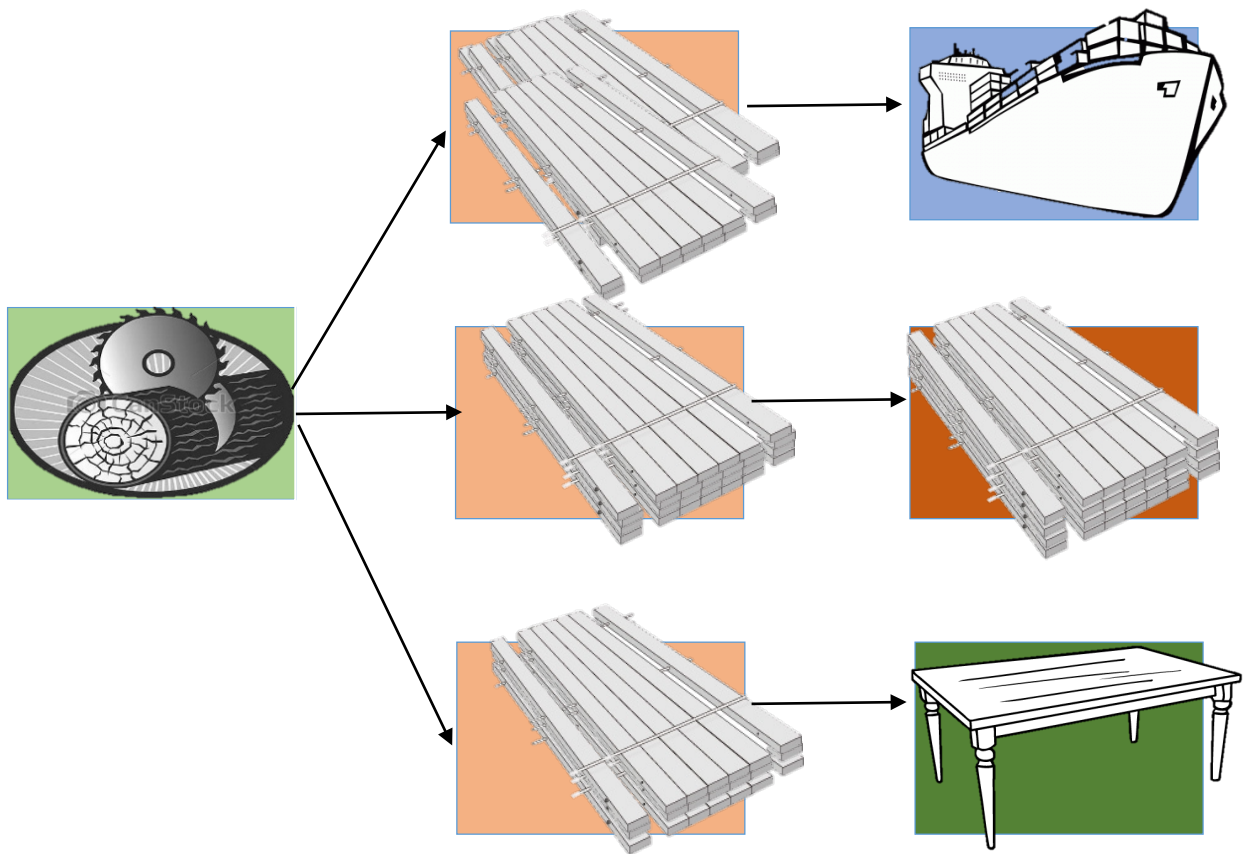


Figura 4.4. Flujo de producto del aserradero hacia los clientes.

Uno de los clientes mencionados son los aserraderos, encargados de generar diversos tipos de productos. Para desarrollar estos tipos de productos se toman decisiones sobre los esquemas de corte que se le aplicarán a los rollizos con el fin de obtener, la mayor cantidad de productos de mayor calidad posible, generando la menor cantidad de residuos. Una vez que las

rollizos son transformados en tablas éstas pasan por diversos procesos (secado, baño anti manchas, cepillado y clasificado) dependiendo del tipo de tabla.

En el problema de la cosecha forestal y el aserrío, se requiere seleccionar las reglas de trozados. Existe la posibilidad de seleccionar reglas que se han utilizado históricamente en la cosecha sin embargo, lo ideal es generar reglas que consideren los requerimientos actuales para generar mayor flexibilidad en la planificación y facilitar las decisiones de la planificación táctica. En el aserradero a su vez, se debe tomar la decisión del esquema de corte y los niveles de inventario a ocupar. La problemática es formulada, como un problema de optimización lineal entera mixta utilizando además, el método de generación de columnas para la obtención de las reglas de trozados. En la planificación táctica, se debe considerar las siguientes decisiones en los periodos correspondientes:

- ¿Qué rodales cosechar?
- ¿Cuántos árboles del rodal se deben cosechar?
- ¿Qué regla de trozados aplicar a cada rodal?
- Número de rollizos a producir.
- Los clientes que abastecerá total o parcialmente cada cancha de trozado.
- Niveles de inventario de cada cancha de trozado.
- Cantidad de rollizos enviados, desde el bosque hacia al aserradero para la producción de tablas.
- El esquema de corte a utilizar, para cada rollizo.
- El nivel de inventario de tablas, en el aserradero.

Además, al tomar las decisiones se deben considerar, las restricciones normales del problema, como las siguientes:

- Disponibilidad de materia prima (rodales para cosechar).
- Capacidad del inventario, en las canchas de trozado y aserraderos.
- Satisfacción de la demanda existente, en el periodo acordado.

Recalcar que la formulación matemática propuesta en este estudio, en comparación con otros estudios considera, la inclusión del problema de la cosecha forestal con el de los aserraderos en su nivel de planificación táctica y la generación de reglas de trozado, dinámicos mediante el método de generación de columnas.

4.2 Modelo de Programación Matemática

El modelo de programación matemática presentado a continuación, es una descomposición mediante el método de generación de columnas. La programación matemática completa se puede observar en el Anexo 2.

El modelo de programación matemática propuesto utiliza, la siguiente nomenclatura de índices, parámetros y variable.

Índices

- NI:** Número de rodales, $i = 1, 2, \dots, NI$
- NJ:** Número de canchas de trozados, $j = 1, 2, \dots, NJ$
- NK:** Número de clientes del bosque, $k = 1, 2, \dots, NK$
- NT:** Número de periodos de planificación, $t = 1, 2, \dots, NT$
- NP:** Número de tipo de rollizos, $p = 1, 2, \dots, NP$
- NR:** Número de reglas de trozado disponibles, $r = 1, 2, \dots, NR$ (*Índice de carácter dinámico, que varía con el número de columnas generadas. El índice inicial es de igual valor al número de productos, NP*)
- NE:** Número de esquemas de corte, $e = 1, 2, \dots, NE$
- NL:** Número de clientes para los aserraderos, $l = 1, 2, \dots, N$
- NC:** Número de tipo de tablas, $c = 1, 2, \dots, NC$
- S_j :** Conjunto de rodales i que abastecen (entregan producto) a la cancha de trozado j .
- Ab :** Conjunto de tipos de rollizos p desde, los cuales se puede obtener el tipo de tabla b .
- Ak :** Conjunto de destinos del bosque k , que corresponden a aserraderos a planificar.

Parámetros

- $Dmin_k^{pt}$:** Cantidad mínima en m^3 del tipo de rollizo p , que demanda el cliente k , en el periodo t .

$Dmax_k^{pt}$:	Cantidad máxima en m ³ del tipo de rollizo p , que demanda el cliente k , en el periodo t .
$I max_j$:	Cantidad máxima de rollizos en m ³ de inventario, que se puede almacenar en la cancha j .
D^p :	Precio duales del rollizo p .
E^{rp} :	Número de veces que el rollizo p , aparece en la regla de trozado r .
RL^p :	Largo del rollizo p .
$RDmin^p$:	Diámetro mínimo, del rollizo p .
ΔRO^p :	Delta permitido para las características, del rollizo p .
A_i :	Largo de los árboles, del rodal i .
$ADmin_i$:	Diámetro menor de los árboles, del rodal i .
$ADmax_i$:	Diámetro mayor de los árboles, del rodal i .
AD_i :	Cantidad de árboles disponibles para cosechar, en el rodal i .
RD_i^p :	Diámetro del árbol del rodal i , al momento de insertar el primer producto p .
\widehat{RD}_i^p :	Diámetro del árbol del rodal i , al momento de insertar el último producto p .
$RDnum_i^p$:	Diámetro del árbol del rodal i , cuando se corta la máxima cantidad permitida de productos p en el árbol, en el momento de insertar el primero producto p .
\widehat{RDnum}_i^p :	Diámetro del árbol del rodal i , cuando se corta la máxima cantidad permitida de productos p en el árbol, en el momento de insertar el último producto p .
Δ_i :	Pendiente del árbol i desde su diámetro mayor hacia su diámetro menor.
P_k^p :	Precio de venta de 1 m ³ , del tipo de rollizo p , al cliente k .
M_i^r :	Costo de cosechar con la regla de trozado r , en el rodal i .
T_{jk} :	Costo de transportar 1 m ³ de rollizos, desde la cancha de trozado j , al cliente k .
L :	Costo de almacenar 1 m ³ de rollizos, durante un periodo de tiempo.

- Q^{cep} :** Rendimiento en porcentaje, al aplicar el esquema de corte tipo e , al tipo de rollizo p , para producir el tipo de tabla c .
- $Umin_i^{ct}$:** Demanda mínima en m^3 , de tipos de tabla tipo c demandados, por el cliente l en el periodo t .
- $Umax_i^{ct}$:** Demanda máxima en m^3 , de tipos de tabla tipo c demandados, por el cliente l en el periodo t .
- $Smax$:** Cantidad máxima en m^3 de inventario de tablas, que se puede almacenar.
- C^{pe} :** Costo de utilizar los rollizos tipo p , bajo el esquema de corte e .
- V^c :** Costo de fabricar $1 m^3$, de tablas de tipo c .
- G_l^c :** Precio de venta de $1 m^3$ del tipo de tabla c , al cliente l .
- N :** Costo de almacenar $1 m^3$ de cualquier tipo de tabla, durante un periodo de tiempo.

Variables

- R_i^{rt} :** Arboles del rodal i , a los cuales se les aplica la regla de trozado r , en el periodo t .
- Op :** Número de veces que el rollizo p , se encuentra en la regla de trozado.
- $Onum^p$:** Distancia entre rollizos p , contiguos o adyacentes.
- $Obool^p$:** Variable binaria, que toma el valor de 1 si el rollizo p , se corta con la regla de trozado y 0 en caso contrario.
- I_j^{pt} :** Cantidad en m^3 del tipo de rollizo p , almacenado en la cancha de trozado j , en el periodo t .
- Y_{jk}^{pt} :** Cantidad en m^3 del tipo de rollizo p , transportado desde la cancha de trozado j hasta el cliente k , en el periodo t .
- X^{pet} :** Cantidad en m^3 de tipos de rollizos p , que se consumen utilizando el esquema de corte e en el periodo de tiempo t .
- S^{ct} :** Cantidad en m^3 , de tipos de tablas c , que se almacenan en el periodo de tiempo t .

B_l^{ct} : Cantidad en m^3 , de tipos de tablas c , enviadas al cliente l , en el periodo t .

Para resolver el problema del presente estudio se plantea una estrategia de descomposición la cual considera, la proposición de un problema maestro detallado a continuación y posteriormente el desarrollo de un sub-problema o problema esclavo. Ambos problemas interactúan para entregar la solución global.

En las expresiones (4.1) a las (4.14) se presenta el problema maestro parte de la estrategia de solución al problema planteado.

Problema Maestro

$$\begin{aligned}
 MAX \quad & \sum_{t=1}^{NT} \sum_{p=1}^{NP} \sum_{k=1}^{NK} \sum_{j=1}^{NJ} P_k^p Y_{jk}^{pt} - \sum_{t=1}^{NT} \sum_{i=1}^{NI} \sum_{r=1}^{NR} M_i^r R_i^{rt} - \sum_{t=1}^{NT} \sum_{p=1}^{NP} \sum_{k=1}^{NK} \sum_{j=1}^{NJ} T_{jk} Y_{jk}^{pt} \\
 & - \sum_{t=1}^{NT} \sum_{j=1}^{NJ} \sum_{j=1}^{NJ} L I_j^{pt} - \sum_{p=1}^{NP} \sum_{e=1}^{NE} \sum_{t=1}^{NT} C^{pe} X^{pet} + \sum_{t=1}^{NT} \sum_{l=1}^{NL} \sum_{c=1}^{NC} G_l^c B_l^{ct} \\
 & - \sum_{l=1}^{NL} \sum_{c=1}^{NC} \sum_{t=1}^{NT} V^c B_l^{ct} - \sum_{c=1}^{NC} \sum_{t=1}^{NT} N S^{ct}
 \end{aligned} \quad (4.1)$$

Sujeto a

$$\sum_{r=1}^{NR} R_i^{rt} \leq AD_i; \quad \forall i = 1, \dots, NI; \quad \forall t = 1, \dots, NT \quad (4.2)$$

$$\sum_{k=1}^{NK} Y_{jk}^{pt} = \sum_{i \in S_j} \sum_{r=1}^{NR} E^{rp} R_i^{rt} + I_j^{p,t-1} - I_j^{pt}; \quad \forall j = 1, \dots, NJ; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT. \quad (4.3)$$

$$Dmin_k^{pt} \leq \sum_{j=1}^{Nj} Y_{jk}^{pt} \leq Dmax_k^{pt}; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT; \quad \forall k = 1, \dots, NK \quad (4.4)$$

$$\sum_{p=1}^{NP} I_j^{pt} \leq Imax_j; \quad \forall j = 1, \dots, NJ; \quad \forall t = 1, \dots, NT. \quad (4.5)$$

$$I_j^{p0} = 0; \quad \forall p = 1 \dots NP; \quad \forall j = 1, \dots, NJ \quad (4.6)$$

$$\sum_{e=1}^{NE} X^{pet} \leq \sum_{k \in Ak} \sum_{j=1}^{Nj} Y_{jk}^{pt}; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT. \quad (4.7)$$

$$\sum_{l=1}^{NL} B_l^{ct} = \sum_{e=1}^{NE} Q^{cep} X^{cet} + S^{c,t-1} - S^{ct}; \quad \forall c = 1, \dots, NC; \quad \forall t = 1, \dots, NT \quad (4.8)$$

$$Umin_l^{ct} \leq B_l^{ct} \leq Umax_l^{ct}; \quad \forall c = 1, \dots, NC; \quad \forall t = 1, \dots, NT; \quad \forall l = 1, \dots, NL. \quad (4.9)$$

$$\sum_{c=1}^{NC} S^{ct} \leq Smax \quad ; \forall t = 1, \dots, NT \quad (4.10)$$

$$S^{c0} = 0 \quad ; \forall c = 1, \dots, NC \quad (4.11)$$

$$I_j^{pt}, Y_{jk}^{pt}, X^{pet}, B_l^{ct}, S^{ct}, R^{rt} \geq 0 \quad (4.12)$$

$$\forall i = 1, \dots, NI; \forall j = 1, \dots, NJ; \forall k = 1, \dots, NK; \forall p = 1, \dots, NP; \forall t = 1, \dots, NT; \forall e = 1, \dots, NE; \forall l = 1, \dots, NL; \forall c = 1, \dots, NC \quad (4.13)$$

A continuación entre las expresiones (4.15) y (4.30) se presenta el problema esclavo o sub-problema como la parte final de la estrategia de solución para resolver el problema planteado.

Problema Esclavo

$$\text{Maximizar } \sum_{p=1}^{NP} D^p O^p \quad (4.14)$$

Sujeto a:

$$\sum_{p=1}^{NP} RL^p O^p \leq A_i; \forall i = 1, \dots, NI \quad (4.15)$$

$$\Delta_i = \left(\frac{ADmax_i - ADmin_i}{A_i} \right); \forall i = 1, \dots, NI \quad (4.16)$$

$$RD_i^p = ADmin_i + \Delta_i Onum^p + \sum_{\bar{p}=0}^{\bar{p}<P} (\Delta_i RL^{\bar{p}} O^{\bar{p}} + \Delta_i Onum^{\bar{p}+1}); \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.17)$$

$$RDnum_i^p = ADmin_i + \sum_{\bar{p}=0}^{\bar{p}<P} \left(\Delta_i RL^{\bar{p}} \left[\frac{A_i}{RL^{\bar{p}}} \right] \right); \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.18)$$

$$\widehat{RD}_i^p = RD_i^p \Delta_i RL^p (O^p - 1); \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.19)$$

$$RDnum_i^p = RDnum_i^p \Delta_i RL^p \left(\frac{A_i}{RL^p} \right); \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.20)$$

$$\frac{RD_i^p}{\Delta RO^p} - RDnum_i^p Obool^p \leq RDmin^p; \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.21)$$

$$RDmin^p \leq RD_i^p \Delta RO^p + RDnum_i^p Obool^p; \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.22)$$

$$\frac{\widehat{RD}_i^p}{\Delta RO^p} - \widehat{RDnum}_i^p Obool^p \leq RDmin^p; \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.23)$$

$$RDmin^p \leq \widehat{RD}_i^p \Delta RO^p + \widehat{RDnum}_i^p Obool^p; \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.24)$$

$$O^p \leq \left(\frac{A_i}{RL^p} \right) O_{bool}^p; \forall i = 1, \dots, NI \quad \forall p = 1, \dots, NP \quad (4.25)$$

$$O^p \in \mathbb{Z} \quad (4.26)$$

$$O_{num}^p \geq 0 \quad (4.27)$$

$$O_{bool}^p \in [0,1] \quad (4.28)$$

$$\forall i = 1, \dots, NI; \forall p = 1, \dots, NP; \forall R = 1, \dots, NR \quad (4.29)$$

A continuación, se describirá cada una de las expresiones presentadas.

La función objetivo está expuesta en la expresión (4.1), la que plantea la maximización y para su comprensión es dividida en dos partes, una que representa lo que se realiza en el bosque, con los beneficios que se obtienen por la venta de rollizos descontando los costos operacionales, de transporte y de almacenamiento y la segunda, corresponde a lo realizado en los aserraderos, en donde se encuentran los ingresos por ventas de tablas (descontando los gastos operacionales y de almacenamiento correspondientes).

La expresión (4.2) es el conjunto de restricciones que señalan que se pueden cosechar igual o menor cantidad que los arboles disponibles que posee cada rodal. Esto ocurre para cada rodal disponible a cosechar.

La expresión (4.3) muestra el conjunto de restricciones que nos expone el balance de rollizos en el bosque. En un lado de la expresión se encuentra la cantidad de rollizos transportado desde las canchas de trozados hacia los clientes, la cual debe ser igual a los rollizos obtenidos en la cosecha forestal, añadiéndole la madera que es extraída desde el inventario, la cual es representada por la cantidad de inventario del período anterior menos la cantidad del período actual. La cantidad de rollizos obtenidos en la cosecha forestal es mostrada por la variable R_i^{rt} . Esto se repite para cada cancha de trozado, cada producto y cada período de tiempo. De este conjunto de restricciones se obtienen los precios duales que se utilizaran en el problema esclavo.

La expresión (4.4) nos indica el conjunto de restricciones que expresan la demanda de rollizos de cada cliente del bosque y en cada periodo de tiempo, que debe ser satisfecha. Esta demanda no es puntual, si no que un rango de demanda en el cual la variable Y_{jk}^{pt} , que representa a los rollizos entregados desde las canchas de trozados a los clientes del bosque, se debe encontrar. Esto se repite para cada período de tiempo.

La expresión (4.5) nos indica el conjunto de restricciones, que se preocupan que el inventario existente en cada cancha de trozado y en cada período de tiempo no exceda el inventario máximo. Esto ocurre para cada cancha de trozado y cada período de tiempo.

La expresión (4.6) establece que los niveles de inventario al momento de comenzar las operaciones, período 0, son nulos.

La expresión (4.7), es la que nos indica la unión del bosque con el aserradero mediante el conjunto de restricciones que exponen que la cantidad de rollizos que se pueden consumir para generar tablas bajo ciertos esquemas de corte en un período de tiempo, X^{pet} debe ser menor o igual a la cantidad de rollizos que fueron entregadas en el aserradero en el mismo período de tiempo, Y_{jk}^{pt} . Esto se comprueba para cada producto y cada período de tiempo.

La expresión (4.8), nos muestra el conjunto de restricciones que exponen el balance de materia entre rollizos y tablas. Por un lado se encuentran las tablas que son enviadas a diferentes destinos y clientes del aserradero, en un período de tiempo, las cuales deben ser iguales a las tablas obtenidas de los rollizos, añadiéndole las tablas que fueron sacadas desde inventario, las cuales son expresadas por la diferencia entre el inventario del período anterior con el del período actual.

La expresión (4.9), es el conjunto de restricciones que indican, la demanda mínima de tablas que exigen los clientes del aserradero y la demanda máxima, de tablas que estos clientes aceptan. El valor de las tablas enviadas, debe estar dentro de este rango. Este proceso se repite para cada período de tiempo

La expresión (4.10) es el conjunto de restricciones que establecen que el inventario de los aserraderos no exceda el inventario máximo permitido en cada período de tiempo.

La expresión (4.11) indica que el inventario al momento de comenzar las operaciones en los aserraderos, período 0, es nulo.

La expresión (4.12) indica las restricciones de no negatividad para las variables.

La expresión (4.13) indica los índices que participan en el problema maestro.

Como se explicó en el capítulo 3, la generación de columnas posee dos problemas, uno maestro y uno esclavo, siendo este último el encargado de generar la columna de este estudio la

regla de trozado. A continuación se describen las expresiones (4.14) a la (4.29) las que componen el problema esclavo.

La función objetivo del problema esclavo está señalada en la expresión (4.14), que señala que se desea maximizar la cantidad de rollizos que se obtienen de un árbol. Junto a la variable O^p , se encuentran las variables duales obtenidas, del problema maestro, D^p .

La expresión (4.15), es el conjunto de restricciones que muestran que la suma del largo de todos los productos presentes en una regla de trozado debe ser menor o igual al largo del árbol del rodal i y para cada rodal.

La expresión (4.16) señala como se obtiene Δ_i , que en conjunto, con el largo de cada rollizo o producto que se desea obtener, RL_i , permite calcular el diámetro resultante del rollizo cuando se aplica un corte.

Lo anterior, es un supuesto utilizado que considera, que el fuste del árbol decrece de forma regular desde su diámetro mayor hacia el menor, situación que en la realidad no ocurre así ya que los fustes de los árboles, decrecen de manera irregular. Para el cálculo del diámetro a diferente altura existen, diversas fórmulas y método, como se señaló en capítulos anteriores. La unión de estas fórmulas con el método de generación de columnas puede ser realizado en estudios posteriores.

La expresión (4.17), es la encargada de proporcionar el valor a RD_i^p , parámetro encargado de comparar que el árbol pueda generar el **primer** producto p considerando su diámetro mínimo. Esta expresión se repite para cada producto y cada rodal.

La expresión (4.18) proporciona valor a $RDnum_i^p$, parámetro utilizado relajar las expresiones (4.21) y (4.22), conjunto de restricciones encargadas de verificar la factibilidad del **primer** producto. $RDnum_i^p$ relaja los conjuntos de restricciones mencionados considerando, el mayor número de productos p pueden ser cortados en el árbol i . Esto ocurre para cada producto y cada rodal.

La expresión (4.19), es la encargada de proporcionar el valor a \widehat{RD}_i^p , parámetro encargado de comparar que el árbol pueda generar el **último** producto p considerando su diámetro mínimo. Esta expresión se repite para cada producto y cada rodal.

La expresión (4.20) proporciona valor a $RDnum_i^p$, parámetro utilizado relajar las expresiones (4.22) y (4.23), conjunto de restricciones encargadas de verificar la factibilidad del **último** producto. $RDnum_i^p$ relaja los conjuntos de restricciones mencionados considerando, el mayor número de productos p pueden ser cortados en el árbol i . Esto ocurre para cada producto y cada rodal.

Las expresión (4.21) es el conjunto de restricciones que verifica que el árbol cumpla con el requerimiento de diámetro mínimo del **primer** producto p . Este conjunto de restricciones es encargado de comprobar la factibilidad del diámetro mínimo del producto por su **extremo inferior**. Esto se repite para cada producto y para cada rodal.

Las expresión (4.22) es el conjunto de restricciones que verifica que el árbol cumpla con el requerimiento de diámetro mínimo del **primer** producto p . Este conjunto de restricciones es encargado de comprobar la factibilidad del diámetro mínimo del producto por su **extremo superior**. Esto se repite para cada producto y para cada rodal.

Las expresión (4.23) es el conjunto de restricciones que verifica que el árbol cumpla con el requerimiento de diámetro mínimo del **último** producto p . Este conjunto de restricciones es encargado de comprobar la factibilidad del diámetro mínimo del producto por su **extremo inferior**. Esto se repite para cada producto y para cada rodal.

Las expresión (4.24) es el conjunto de restricciones que verifica que el árbol cumpla con el requerimiento de diámetro mínimo del **último** producto p . Este conjunto de restricciones es encargado de comprobar la factibilidad del diámetro mínimo del producto por su **extremo superior**. Esto se repite para cada producto y para cada rodal.

Las expresiones (4.17) a la (4.24) son utilizadas para comprobar que el árbol pueda producir los productos solicitados considerando el diámetro mínimo exigido por el cliente. La Figura 4.5 ejemplifica el uso de de estas expresiones, señalando donde interviene cada una. Para el ejemplo utilizado en la Figura 4.5, se consideran cuatro productos que se quieren obtener, la Figura 4.5 indica que del producto 1 y producto 2 se obtienen tres piezas, mientras que del producto 3 se obtiene una y por ultimo del producto 4 se obtienen dos piezas. Las expresiones (4.21) y (4.22) intervienen en la primera pieza del producto 2, verificando que los diámetros de las piezas no superen el diámetro del árbol. De manera análoga las expresiones (4.23) y (4.24) verifican que los diámetros de la última pieza del producto 2, no superen el diámetro del árbol.

Cabe señalar que si la primera y última pieza están contenidas en el árbol, las que se encuentren dentro de ese rango están contenidas, tal como lo realiza la pieza número dos del producto 2. En la Figura 4.5. Este proceso se realiza para cada uno de los tres productos restantes. Se observa en la Figura 4.5 la variable O_{num}^p , la que contabiliza la distancia entre dos productos diferentes.

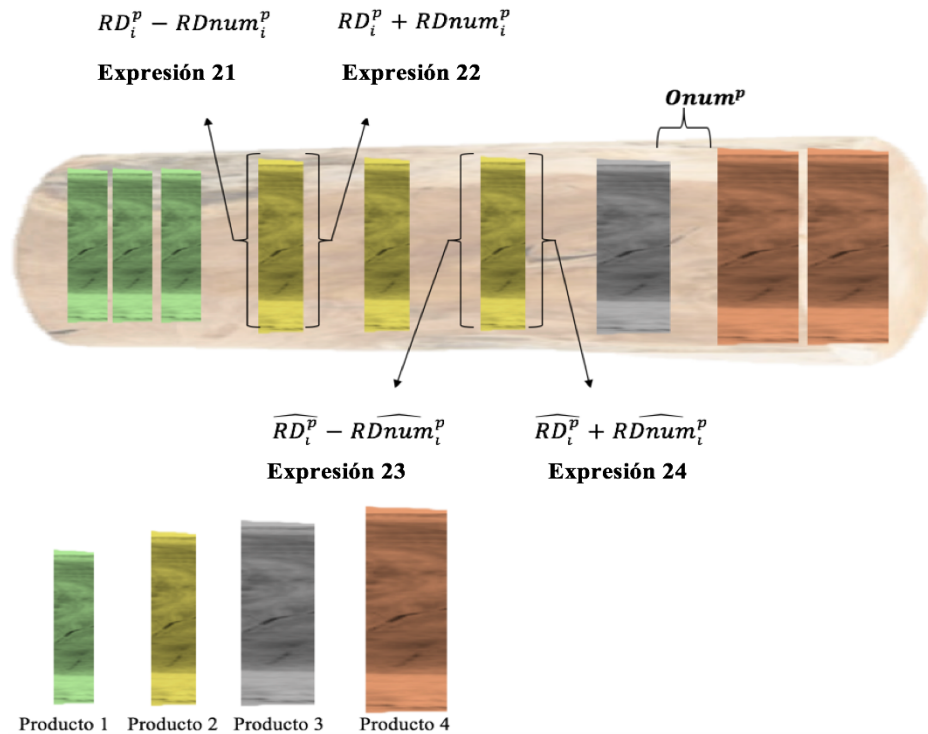


Figura 4.5. Ejemplo de utilización de las expresiones (4.21) a la (4.24)

La expresión (4.25) son los conjuntos de restricciones que unen las variables binarias O_{bool}^p con las variables entera O^p , considerando una relajación de la mayor cantidad de rollizos p que se pueden obtener del árbol del rodal i . Esto se repite para cada producto y cada rodal.

La expresión (4.26) indica las variables de tipo entera utilizadas en el problema esclavo.

La expresión (4.27) indica las restricciones de no negatividad para las variables continuas del problema esclavo.

La expresión (4.28) señala las variables de tipo binaria utilizadas en el problema esclavo.

La expresión (4.29) indica los índices que participan en el problema esclavo.

4.3 Método de solución.

La generación de columnas dispone de un problema maestro y un problema esclavo, tal como se detalló en la formulación anterior. En este caso, el problema maestro es el encargado de maximizar los beneficios de la organización a través de la satisfacción de la demanda de productos que solicitan sus clientes optimizando, los recursos tales como la cantidad de bosques disponibles. Mientras que el problema esclavo es el encargado de generar las reglas de trozado para la cosecha forestal, por lo que es relevante representar el problema esclavo gráficamente.

Se considera que el problema esclavo busca generar reglas de trozado considerando los siguientes supuestos:

- La suma de los largos de los rollizos presentes en la regla de trozados, no debe ser mayor al largo del árbol.
- Cada rollizo presente en la regla de trozado debe satisfacer los requerimientos del cliente en diámetro y largo.
- Se considera que el rollizo obtenido del árbol decrece de manera regular desde su diámetro mayor a su diámetro menor.
- Al generar las combinaciones de diversos rollizos en las reglas de trozados se desea, maximizar el uso del rollizo obtenido del árbol y minimizar sus desperdicios.

Para representar la solución, se utiliza el siguiente ejemplo:

Se tiene un rollizo obtenido de un árbol con las características que se señalan en la Tabla 4.1, en la columna árbol, y se desea generar reglas de trozado las cuales incluyen 3 rollizos de diferentes características y demandas, las que son señaladas en las columnas Producto 1, Producto 2 y Producto 3 de la misma Tabla 4.1.

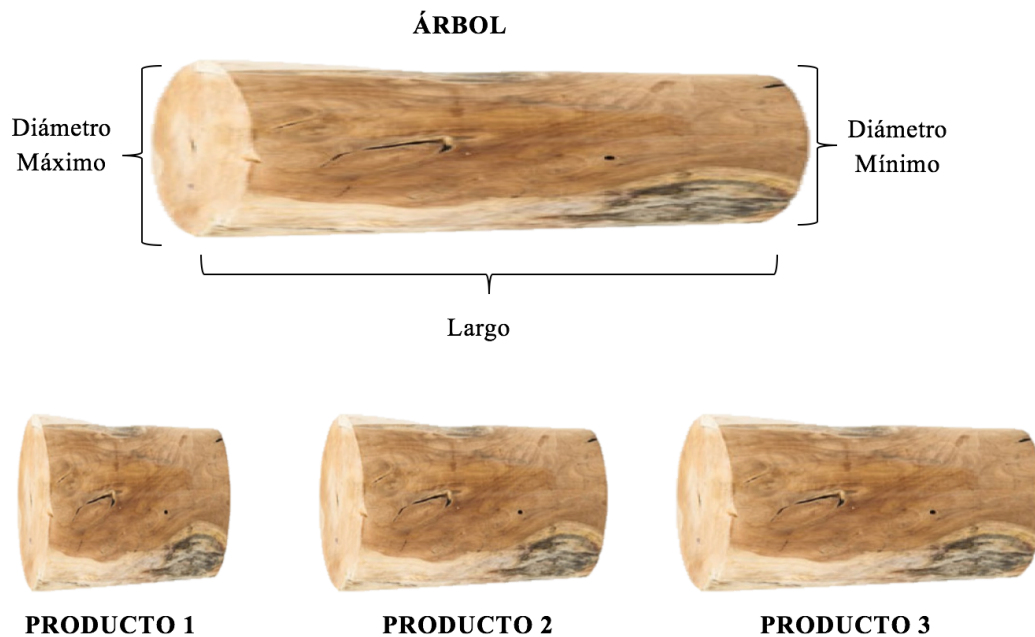


Figura 4.6. Representación problema de ejemplo regla de trozado

En resumen se posee un rollizo de longitud mayor en el cual se desea obtener la mejor regla de trozado considerando los rollizos que piden los clientes, como se observa en la Figura 4.6.

Tabla 4.1 Características árbol y productos para generar la regla de trozado.

	Árbol	Producto 1	Producto 2	Producto 3
Largo (m)	20	2	4	9
Diámetro Mínimo (cm)	25	8	22	24
Diámetro Máximo (cm)	45	-	-	-
Demanda (unidades)	-	16	8	4

Al resolver el problema planteado, se generan tres reglas de trozado y cada una de estas es utilizada 2 veces, lo que significa que para satisfacer la demanda señalada en la Tabla 4.1 necesitamos de 6 árboles. Observar Tabla 4.2 y la Figura 4.7.

Tabla 4.2 Resultados problema de ejemplo.

	Regla 1	Regla 2	Regla 3
Producto 1 (unidades)	10	-	1
Producto 2 (unidades)	-	5	-
Producto 3 (unidades)	-	-	2
Nº de veces utilizada la regla	2	2	2



Figura 4.7. Esquematización resultados problema de ejemplo.

CAPÍTULO 5.

Resultados

El modelo matemático presentado en el capítulo anterior fue implementado computacionalmente utilizando la tecnología *Concert* de *Ilog*, biblioteca que sirve para el desarrollo y despliegue de modelos de optimización utilizando programación lineal y cuadrática, entera mixta y con restricciones.

Las instancias que se presentarán en el presente capítulo se implementaron en lenguaje C++ utilizando la biblioteca mencionada anteriormente y para resolverlo se utilizó el programa *CPLEX 12.6.1* de *IBM Ilog*.

El desarrollador CPLEX fue implementado en un computador con un procesador *Intel Core i7*, 2.3 GHz de frecuencia, 16 Gb de memoria Ram en un sistema operativo *Windows 7 Professional*.

El código del modelo matemático en C++ implementado en CPLEX se encuentra en el anexo 1.

Se generó un problema de dimensiones medias para analizar la información detallada de las reglas de trozados obtenidas. A esta instancia la denominaremos instancia original y contiene las dimensiones señaladas en la Tabla 5.1.

Tabla 5.1 Dimensiones instancia original.

	Índice	
Nro. de rodales	NI	4
Nro. de canchas de trozados	NJ	3
Nro. de clientes del bosque	NK	4
Nro. de periodos de planificación	NT	4
Nro. de tipo de rollizos	NP	3
Nro. de reglas de trozados	NR	3*
Nro. de esquemas de corte	NE	3
Nro. de clientes para los aserraderos	NL	3
Nro. de tipo tablas	NC	3

* Índice de carácter dinámico, que varía con el número de columnas generadas. El índice inicial es de igual valor al número de productos, NP

La Tabla 5.2 presenta las características que posee un árbol promedio perteneciente a cada rodal. Los datos fueron estimados a partir de información recopilada a través de empresas del área forestal.

Tabla 5.2 Características de los rodales en la instancia original.

	Rodal 1	Rodal 2	Rodal 3	Rodal 4
Largo	10	15	20	25
Diámetro mínimo	20	25	30	35
Diámetro máximo	45	50	50	55
Arboles disponibles	2000	1000	1000	800

De igual modo las características de los rollizos o productos que se desean obtener, son señaladas en la Tabla 5.3. Las características fueron obtenidas a partir de la literatura (Epstein, et al., 1999c).

Tabla 5.3 Características de los productos en la instancia original.

	Producto 1	Producto 2	Producto 3
Largo	2	3	4
Diámetro mínimo	8	24	26
Tolerancia	20%	15%	5%
Demanda mínima*	9.000	7.000	7.000
Demanda máxima*	13.000	11.000	11.000

La función objetivo de la instancia original entrega ganancias por US\$176.030, tal como lo muestra la Tabla 5.4. Tabla que además señala que para llegar a dicho valor se generaron 83 reglas de trozados de las cuales fueron utilizadas solo 14 para dar solución a la instancia original, cuyas dimensiones se observan en la Tabla 5.1, considerando las características de los rodales y rollizos señalados en las Tablas 5.2 y 5.3 respectivamente.

La Tabla 5.5 entrega información más detallada respecto de las reglas de trozado generadas. Se observa que fueron 8 diferentes reglas de trozados generadas, replicándose varias de estas en diferentes periodos de tiempo. Al agrupar las reglas de trozado de manera distinta a la observada en la Tabla 5.5, se observa que los árboles de los rodales 2, 3 y 4, fueron ocupados totalmente, mientras que al rodal 1 le quedaron 525 árboles en pie, esto se justifica debido que las características de los rodales 2, 3 y 4 son superiores a las del rodal 1 (observar Tabla 5.2)

* Demanda total de productos considerando todos los periodos de tiempo

por lo que son más convenientes de cosechar. La Tabla 5.6 agrupa las reglas de trozados según se lo mencionado anteriormente.

Tabla 5.4 Resultados instancia original

Instancia original	
Número de variables problema maestro	347
Número de variables enteras problema maestro	83
Número de restricciones problema maestro	167
Número de variables Problema esclavo	9
Número de variables binarias problemas esclavos	3
Número de variables enteras problemas esclavos	3
Número de restricciones problemas esclavos	16
Reglas de trozado generadas	83
Reglas de trozado usadas	14
Tiempo (s)	11
Función Objetivo (US\$)	176.030

Tabla 5.5 Características de las reglas de trozados utilizadas en la instancia original

Nº regla de trozado	Rodal	Periodo de tiempo	Nº de veces de uso de la regla	Producto 1 (unidades)	Producto 2 (unidades)	Producto 3 (unidades)
12	3	1	106	10	-	-
13	3	2	139	10	-	-
14	3	3	235	10	-	-
15	3	4	371	10	-	-
20	1	1	1468	-	3	-
24	2	1	518	-	5	-
28	3	1	1	1	6	-
39	1	4	7	1	-	2
40	2	1	1	1	-	3
41	2	2	1	1	-	3
43	2	4	480	1	-	3
44	3	1	147	-	-	5
46	3	3	1	-	-	5
48	4	1	800	-	-	6

Tabla 5.6 Agrupación reglas de trozado por rodal.

Producto 1 (unidades)	Producto 2 (unidades)	Producto 3 (unidades)	Rodal	Utilizados en las reglas	Nº de veces utilizado	Nº de árboles usados	Nº de árboles disponibles
-	3	-	1	20	1468	1475	2000
1	-	2	1	39	7		
-	5	-	2	24	518	1000	1000
1	-	3	2	40; 41; 43	482		
10	-	-	3	12; 13; 14; 15	851		
1	6	-	3	28	1	1000	1000
-	-	5	3	44; 46	148		
-	-	6	4	48	800	800	800

La Tabla 5.7 muestra información que no puede ser inferida al analizar la Tabla de resultados 5.4. Esta información consta del desglose de las reglas de trozados generadas para resolver la instancia original, la cual ratifica lo detallado en la Tabla 5.5 con respecto a que las reglas efectivamente generadas son mucho menor a las totales generadas. Esto es debido que para comenzar el problema maestro se generan patrones de corte muy malos, para que sean remplazados fácilmente por las columnas generadas. El número de patrones iniciales es la cantidad de productos a satisfacer en la demanda, en este caso NI y para la instancia original la cantidad es 3. Además, los rodales poseen las mismas características para cada periodo de tiempo entonces, cada regla de trozado generada para un bosque, es repetida para cada periodo de tiempo. Adicionalmente, los patrones iniciales y generados por las repeticiones de periodos de tiempo existen otras reglas de trozados que se replican, estas son las utilizadas para detener la ejecución del modelo. Como se explicó en el capítulo 3, la generación de columnas se detiene cuando el problema esclavo no genera columnas que mejoren el problema maestro y esto se verifica con el valor de la función objetivo del problema esclavo; si el valor es mayor estricto a 1 si existen, columnas que puedan mejorar el problema maestro. Al contrario si el valor de la función objetivo es igual a 1, no existen columnas que mejoren el problema maestro. Sin embargo, existen soluciones del problema esclavo cuyos valores son muy cercanos a 1 y son los mejores que puede entregar dicho problema esclavo, pero con estos no se detiene la ejecución del modelo y continúan generando dicha columna reiteradamente sin establecer la solución del problema. Para evitar lo anterior, se generó un límite para establecer término a la generación de columnas por parte del problema esclavo y corresponde a la expresión 5.1.

$$2 * (NI * NT) \quad (5.1)$$

La expresión 5.1 indica que el límite del problema esclavo, es la multiplicación del número de bosques por el número de periodos de tiempo y esto multiplicado por dos para corroborar que la repetición de la columna generada no sea parte del proceso y si no parte del ciclo sin salida generado.

Tabla 5.7 Desglose de las reglas de trozado generadas instancia original

Reglas de trozado iniciales	3
Reglas de trozado efectivamente generadas	12
Repeticiones reglas de trozados periodos de tiempo	36
Repeticiones reglas de trozados periodos de tiempo	32
Total reglas de trozado generadas	83

En síntesis las reglas de trozados entregadas por el programa, se pueden agrupar en los cuatro grupo mencionados anteriormente, de los cuales dos de éstos corresponden a repeticiones de las reglas generadas. La Figura 5.1 permite visualizar lo mencionado.

Desglose del total de reglas de trozado

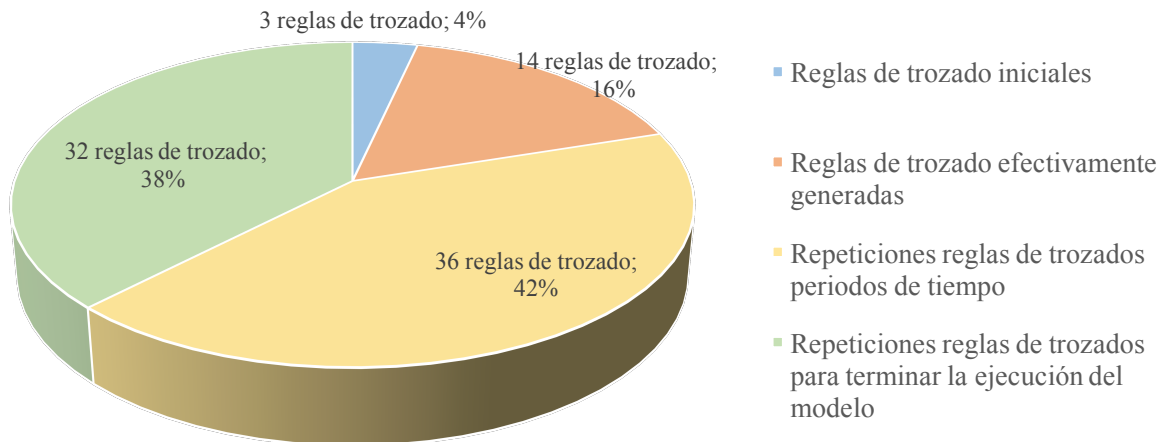


Figura 5.1 Desglose del total de reglas de trozado.

Considerando el total de reglas de trozados y las que realmente fueron utilizadas, se observa que estas últimas son menores en comparación a las primeras. Observar el Figura 5.2.

Desglose reglas de trozado

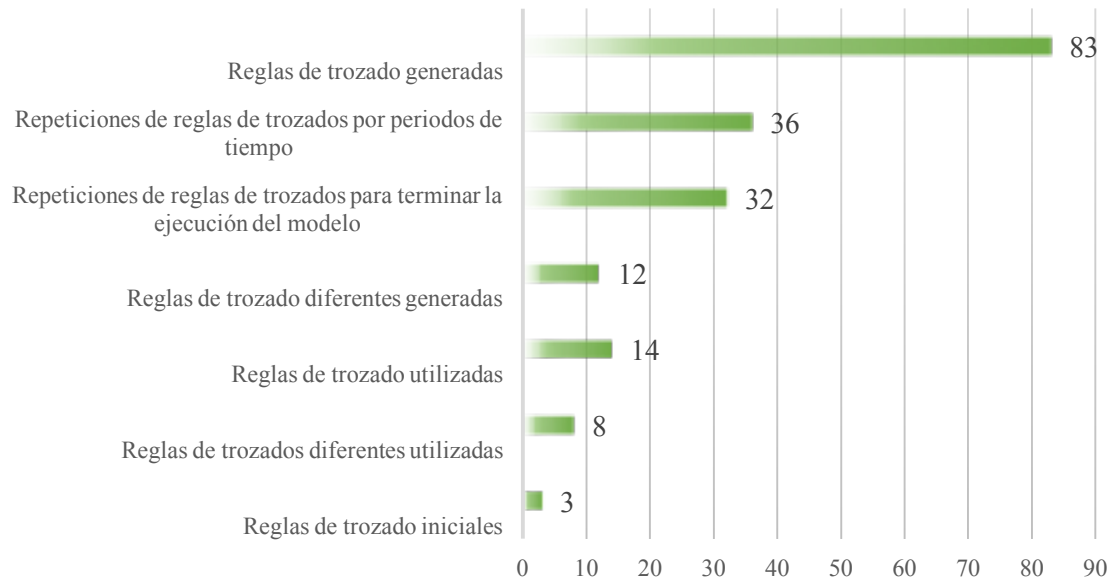


Figura 5.2. Desglose reglas de trozado

Como lo muestra la Figura 5.2, las reglas que realmente fueron generadas son menores a las que se generaron, por lo que queda en trabajos posteriores la utilización de métodos que eviten la repetición de columnas generadas, haciendo el modelo más eficiente, sin olvidar que las repeticiones de reglas de trozados fueron utilizadas para facilitar la obtención de resultados en la programación.

Además de la instancia original, se generó un conjunto de instancias cuyos valores de entrada de los parámetros fueron completados de acuerdo a información recopilada de la literatura forestal nacional existente e información facilitada por profesionales que trabajan en el rubro.

El conjunto de instancias generadas consta de 10 problemas de diferentes dimensiones, incluyendo la instancia original analizada anteriormente la cual corresponde a la instancia 4. La Tabla 5.8 muestra las dimensiones de estos 10 problemas.

Los resultados obtenidos del conjunto de instancias se muestran en la Tabla 5.9, donde se entrega información del problema maestro, problema esclavo y columnas generadas.

Tabla 5.8 Dimensiones utilizadas para el conjunto de instancias iniciales.

	Índice	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10
Nro. de rodales	NI	2	2	4	4	10	10	20	20	30	30
Nro. de canchas de trozados	NJ	2	3	2	3	3	4	3	4	3	4
Nro. de clientes del bosque	NK	2	3	3	4	4	5	4	5	4	5
Nro. de periodos de planificación	NT	1	2	3	4	4	6	4	6	4	6
Nro. de tipo de rollizos	NP	2	3	2	3	4	6	4	6	6	8
Nro. de reglas de trozados	NR	2*	3*	2*	3*	4*	6*	4*	6*	6*	8*
Nro. de esquemas de corte	NE	2	3	2	3	3	4	3	4	3	4
Nro. de clientes para los aserraderos	NL	2	3	2	3	3	5	3	5	3	5
Nro. de tipo tablas	NC	2	2	2	3	3	4	3	4	3	4

Comparación del valor de la función objetivo del conjunto de instancias

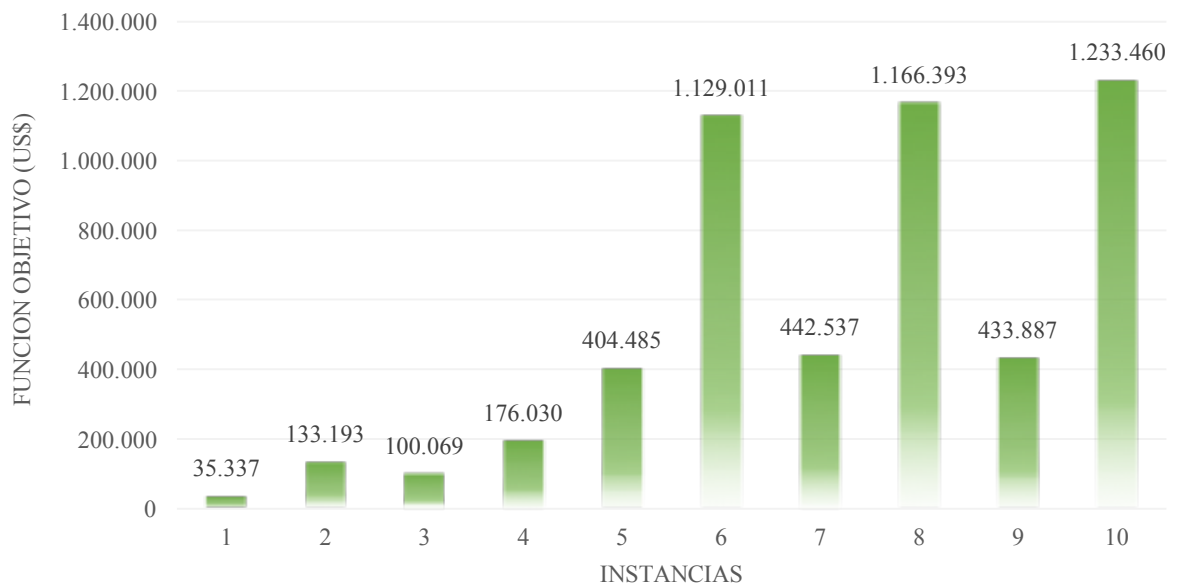


Figura 5.3. Comparación del valor de la función objetivo del conjunto de instancias.

* Índice de carácter dinámico, que varía con el número de columnas generadas. El índice inicial es de igual valor al número de productos, NP

Tabla 5.9 Resultados conjunto de instancias.

Instancia	Nro. de variables problema maestro	Nro. de variables enteras problema maestro	Nro. de restricciones problema maestro	Nro. de variables Problema esclavo	Nro. de variables binarias problemas esclavos	Nro. de variables enteras problemas esclavos	Nro. de restricciones problemas esclavos	Reglas de trozado generadas	Reglas de trozado usadas	Tiempo (s)	Función Objetivo (US\$)
1	32	10	23	6	2	2	11	10	3	24	35.337
2	129	23	71	9	3	3	16	23	5	38	133.193
3	128	50	69	6	2	2	11	50	6	46	100.069
4	327	63	167	9	3	3	16	83	14	11	176.030
5	476	140	206	12	4	4	21	140	7	16	404.485
6	1434	282	550	18	6	6	31	282	19	86	1.129.011
7	580	244	216	12	4	4	21	244	13	407	442.537
8	1326	174	560	18	6	6	31	174	14	97	1.166.393
9	846	366	292	18	6	6	31	366	14	181	433.887
10	2132	644	692	24	8	8	41	644	29	350	1.233.460

Los parámetros de entrada para costos o precios, fueron ingresados en dólares. La Figura 5.3 compara el resultado de la función objetivo de las 10 instancias. Las instancias 6, 8 y 10 presentan un valor mayor en su función objetivo debido al valor de sus dimensiones principalmente por el número de rodales, NI , y el número de periodos de planificación, NT .

Comparación tiempos de resolución del conjunto de instancias

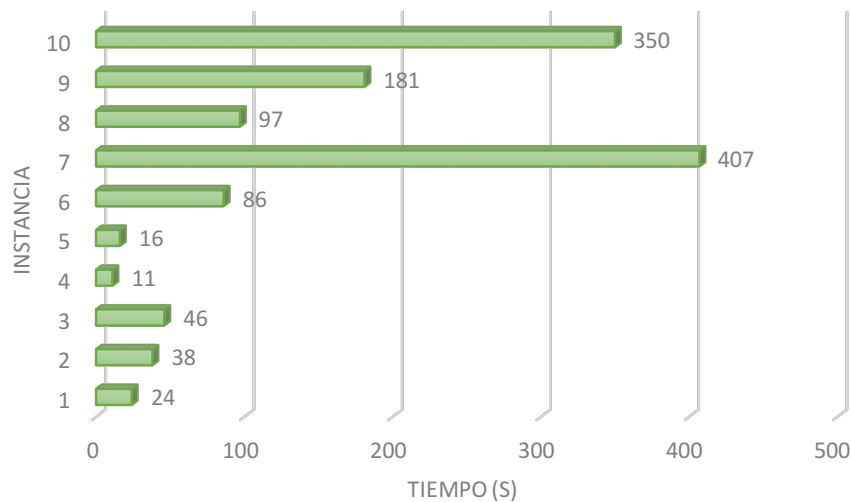


Figura 5.4. Comparación tiempos de resolución del conjunto de instancias.

El mismo análisis de comparación se puede realizar para el resto de los resultados entregados en la Tabla 5.9 obteniendo, la misma conclusión mencionada anteriormente las dimensiones del problema. Sin embargo dentro de los resultados resalta los tiempos de resolución los cuales, en su totalidad son relativamente bajos y la diferencia entre cada uno de los tiempos se debe a las dimensiones de cada instancia. No obstante existen otros factores que permiten que el tiempo de resolución sea bajo, uno de éstos es la escasa cantidad de variables binarias, las que sólo están presentes en los problemas esclavos.

CAPÍTULO 6.

Conclusiones

Se creó un modelo matemático el cual fue implementado computacionalmente para su resolución. La solución entregada por el modelo permite apoyar la toma de decisiones de la planificación táctica entre cosecha forestal y los aserraderos mediante el conocimiento de variables significativas destacando los reglas de trozados dinámicos los cuales se generan a partir de las condiciones del mercado, lo que trae a posterior mayores beneficios, no solamente a las empresas de cosecha del bosque, si no que lleva consigo también a los aserraderos.

El estado del arte de la industria forestal nacional entregó variada información acerca del sector, desde el comienzo en la plantación de los árboles pasando por el manejo que debe poseer el bosque y la importancia que esto tiene para la obtención de madera de una mejor calidad. En cuanto a la cosecha forestal y su bibliografía disponible, esta aportó información con respecto la planificación en todas sus etapas: estratégicas, tácticas y operativas, los métodos existentes para la cosecha forestal y cuáles son los que se aplican efectivamente en Chile. Dentro de la información obtenida en la cosecha forestal llama la atención la manera de obtener y aplicar los patrones de corte para la generación de los distintos rollizos, los que en su mayoría son creados a partir de datos históricos o en base a la experiencia del trabajador forestal. Con respecto a los aserraderos, hay disponible información sobre los clientes que se deben abastecer, los tipos de tablas que se pueden obtener de los rollizos y la manera de optimizar el esquema de corte de estos.

Comprender y esquematizar el proceso de la cosecha forestal y de cómo está fuertemente ligada a los aserraderos, sobre todo en la industria Chilena, permitió encontrar una propuesta de mejora a los modelos matemáticos existentes, la generación de patrones de corte o reglas de trozados dinámicos. Para abordar esta problemática se recurrió a la técnica de descomposición de generación de columnas, que se caracteriza por poseer un problema maestro quien se encarga de satisfacer las demandas y un sub problema o problema esclavo, encargado de optimizar el recurso disponible, en este caso maximizar el uso de los fustes. El modelo matemático logra plasmar los requerimientos exigidos por el cliente al solicitar los productos. Estas exigencias son: satisfacción de demanda, diámetro mínimo (cm) y largo (m).

El modelo maestro es quien busca maximizar las utilidades del problema, considera los costos operacionales, costos de transporte, inventario y producción de la cosecha y los

aserraderos. También es quien se encarga de la satisfacción de la demanda tanto de rollizos como de tablas. Mientras que el problema esclavo es quien busca maximizar el fuste del árbol generando la menor cantidad de desperdicio posible. Quien interactúa entre ambos problemas son las variables duales de la expresión 4.3 del capítulo 4, expresión que refleja el balance de rollizos en el bosque; rollizos entregados a los clientes son iguales a los rollizos cosechados en los bosques más el flujo de productos de inventario.

La instancia original permitió ver como se comportaban los patrones de corte generados, que productos poseía cada uno de los patrones, en que periodo de tiempo aplicar los patrones de corte y a la cantidad de arboles que se le debía aplicar. Información que permitió la exclusión de variables binarias en el problema maestro, variables que habían sido consideradas en estudios anteriores y que con su eliminación aportan a la rápida ejecución del modelo computacional.

Para todas las instancias del conjunto se encontró la solución óptima, la diferencia entre una y otra radica en las dimensiones de los escenarios. Llama la atención la generación de un gran número de columnas, sin embargo la mayoría de éstas se repiten varias veces, esto debido que las reglas se generan para los rodales y estas se repiten para cada periodo de tiempo y también por la utilización de un número máximo de columnas repetidas, generadas para terminar la ejecución del modelo matemático.

La Tesis de grado se destaca por poseer características que estudios similares no poseen, como la integración de la cosecha forestal y los aserraderos o la generación de patrones dinámicos mediante el método de generación de columnas, no obstante el estudio presenta varias posibilidades de mejora para estudios posteriores tal como: la consideración de los fustes de manera irregular, característica que transformaría al problema en uno no lineal; la integración de otras empresas del rubro como la generación de bio masa o la industria papelera; consideración de otros estimadores que pudiesen generar los precios duales; o la aplicación de generación de columnas en el aserradero quizás considerando coordenadas polares.

La utilización del modelo matemático de programación lineal entero mixto para la planificación táctica entre cosecha y aserraderos, utilizando patrones dinámicos facilita la labor de quienes toman las decisiones a nivel táctico, ofreciéndole mayor información, respuestas más certeras y flexibilidad para la toma de decisiones al obtener los patrones de acuerdo a las condiciones que presenta el mercado y posee la empresa.

Referencias

1. Alvarez, J. A., Villagra, P. E., Villalba, R., & Debandi, G. (2013). Effects of the pruning intensity and tree size on multi-stemmed *Prosopis flexuosa* trees in the Central Monte, Argentina. *Forest Ecology and Management*, 310, 857–864. <http://doi.org/10.1016/j.foreco.2013.09.033>
2. Baesler, F., Araya, E., Ramis, F., & Sepulveda, J. (2004). THE USE OF SIMULATION AND DESIGN OF EXPERIMENTS FOR PRODUCTIVITY IMPROVEMENT IN THE SAWMILL INDUSTRY. Presented at the Winter Simulation Conference.
3. Baskent, E. Z., Başkaya, Ş., & Terzioğlu, S. (2008). Developing and implementing participatory and ecosystem based multiple use forest management planning approach (ETÇAP): Yalnızçam case study. *Forest Ecology and Management*, 256(4), 798–807. <http://doi.org/10.1016/j.foreco.2008.05.039>
4. Baskent, E. Z., & Keles, S. (2005). Spatial forest planning: A review. *Ecological Modelling*, 188(2–4), 145–173. <http://doi.org/10.1016/j.ecolmodel.2005.01.059>
5. Bossi P. (2007). *Estudio de tiempo y rendimiento en torres de madereo en predio Ranchillo, Séptima región*. Universidad de Chile, Santiago, Chile.
6. Carey, P., U. A. de. (1997). *Bosque*. Universidad Austral de Chile.
7. Carrasco. M. (2008). *Caracterización de la Accidentalidad Ocupacional en Faenas de Silvicultura y Cosecha Forestal*. Universidad de Chile, Santiago, Chile.
8. Chavez, C., Cornejo, C., & Baesler, F. (2008). Optimization of the Production Planning in the Supply Chain (Market-Sawmill-Harvest) using a Game Theory Approach. Presented at the 51st International Convention of Society of Wood Science and Technology, Concepción, Chile.
9. Chvátal, V. (1946). *Linear Programming*. New York, Estados Unidos: W.H. Freeman and Company.

10. Corporación Nacional Forestal. CONAF. (2006). *Informe del sector Forestal en Chile*. Santiago, Chile.
11. Cuñachi Encinas, George. (2012). Manual Práctico de cubicación de madera rolliza - aserrada. CORPIAA - ATALAYA. Retrieved from http://www.itto.int/files/itto_project_db_input/3033/Technical/TFL-SPD-030-12-R1-M-Manual-Practico-CubicacionMaderaRollizayAserrada.pdf
12. Davel M., Tejera L., & Honorato M. (2009, April). Poda y raleo a desecho en plantaciones de pinos. Estación Experimental Agroforestal Esquel (Chubut). Retrieved from http://inta.gob.ar/documentos/poda-y-raleo-en-plantaciones-de-pinos/at_multi_download/file/INTA_forestal16_manejo_forestal.pdf
13. Departamento de Montes. (2004). *Inventario Forestal Nacional, Manual de Campo*. Guatemala: Organización de las Naciones Unidas para la Agricultura y la Alimentación. Retrieved from <ftp://ftp.fao.org/docrep/fao/008/ae578s/ae578s00.pdf>
14. Díaz Smith, Santiago. (2009, October 30). *Evaluación del método de generación de columnas para el problema de corte "Cutting Stock" usando heurísticas para la obtención de patrones de corte*. de las Américas Puebla, Cholula, Puebla.
15. Duarte Alleuy, Blas. (2009, Agosto). *Heurística basada en generación de columnas para el problema de ruteo de vehículos con ventanas de tiempo y flota homogénea. Caso: Cervecerías Chile S.A.* Universidad de Chile, Santiago, Chile. Retrieved from http://www.tesis.uchile.cl/tesis/uchile/2009/duarte_b/sources/duarte_b.pdf
16. Epstein, R., Nieto, E., Weintraub, A., Chevalier, P., & Gabarró, J. (1999a). A system for the design of short term harvesting strategy1. *European Journal of Operational Research*, 119(2), 427–439. [http://doi.org/10.1016/S0377-2217\(99\)00144-7](http://doi.org/10.1016/S0377-2217(99)00144-7)

17. Epstein, R., Nieto, E., Weintraub, A., Chevalier, P., & Gabarró, J. (1999b). A system for the design of short term harvesting strategy1. *European Journal of Operational Research*, 119(2), 427–439. [http://doi.org/10.1016/S0377-2217\(99\)00144-7](http://doi.org/10.1016/S0377-2217(99)00144-7)
18. Epstein, R., Nieto, E., Weintraub, A., Chevalier, P., & Gabarró, J. (1999c). A system for the design of short term harvesting strategy1. *European Journal of Operational Research*, 119(2), 427–439. [http://doi.org/10.1016/S0377-2217\(99\)00144-7](http://doi.org/10.1016/S0377-2217(99)00144-7)
19. Garcés, J. (2007). *Optimización de los esquemas de corte en aserraderos forestales mediante algoritmos metaheurísticos*. Universidad de Concepción, Concepción, Chile.
20. Ghajar, I., & Najafi, A. (2012). Evaluation of harvesting methods for Sustainable Forest Management (SFM) using the Analytical Network Process (ANP). *Forest Policy and Economics*, 21, 81–91. <http://doi.org/10.1016/j.forpol.2012.01.003>
21. Hillier S., Frederick, L., Gerald. (2010). *Introducción a la Investigación de Operaciones* (Novena edición). The Mc Graw Hill Companies, Inc.
22. Hinostroza, I., Pradenas, L., & Parada, V. (2013). Board cutting from logs: Optimal and heuristic approaches for the problem of packing rectangles in a circle. *International Journal of Production Economics*, 145(2), 541–546. <http://doi.org/10.1016/j.ijpe.2013.04.047>
23. Instituto Forestal de Chile, INFOR. (2013). *Anuario Forestal 2013* (No. Boletín Estadístico 140). Santiago, Chile.
24. Jakobs, S. (1996). On genetic algorithms for the packing of polygons. *European Journal of Operational Research*, 88(1), 165–181. [http://doi.org/10.1016/0377-2217\(94\)00166-9](http://doi.org/10.1016/0377-2217(94)00166-9)
25. Lledó G. (2004). *Evaluación del trozado para rodales de pino insigne en canchas de Forestal Bio Bio S.A.* (Tesis). Universidad de Chile, Santiago, Chile. Retrieved from http://www.tesis.uchile.cl/tesis/uchile/2004/lledo_g/sources/lledo_g.pdf

26. Lübbecke, M., & Desrosiers, J. (2005). Selected Topics in Column Generation, *53*(6), 1007–1023.
27. Maturana, S., Pizani, E., & Vera, J. (2010). Scheduling production for a sawmill: A comparison of a mathematical model versus a heuristic. *Computers & Industrial Engineering*, *59*(4), 667–674. <http://doi.org/10.1016/j.cie.2010.07.016>
28. Peñailillo, F. (2004). *Modelamiento y solución para un problema de Planificación Agregada en un sistema intermitente: aplicación en un aserradero* (Informe para optar al grado de Magíster de Ingeniería Industrial, Departamento de Ingeniería Industrial, Universidad de Concepción.). Universidad de Concepción, Concepción, Chile.
29. Pradenas, L., Garcés, J., Parada, V., & Ferland, J. (2013). Genotype–phenotype heuristic approaches for a cutting stock problem with circular patterns. *Engineering Applications of Artificial Intelligence*, *26*(10), 2349–2355. <http://doi.org/10.1016/j.engappai.2013.08.003>
30. Rebolledo, R. (n.d.). *Optimización del Trozado de Árboles y Transporte de Rollizos usando Simulated Annealing*. Universidad de Concepción, Concepción, Chile.
31. Romero, R., Poblete, M., & Baesler, F. (2004). MODELO DE PROGRAMACIÓN DE LA PRODUCCIÓN PARA LA INDUSTRIA DEL ASERRÍO. *Revista Ingeniería Industrial, Año 3*(N°1), 19–23.
32. Rönnqvist, M. (2003). Optimization in forestry. *Mathematical Programming*, *97*(1-2), 267–284. <http://doi.org/10.1007/s10107-003-0444-0>
33. Sapunar, P., Mansilla, B., & Fuentealba, G. (1999). Sistema Cartográfico de apoyo a la planificación y habilitación de la cosecha forestal. *XII SILVOTECNA. Concepción, Chile. 120 Pp.*
34. Sotomayor A., García E., & Helmke E. (2002). *Manejo y mantención de plantaciones forestales*. Santiago, Chile: Instituto forestal de Chile. INFOR.

35. Toro, E, Garcés, A, & Ruiz, H. (2008). Solución al problema de empaquetamiento bidimensional usando un algoritmo híbrido constructivo de búsqueda en vecindad variable y recocido simulado, *46*, 119–131.
36. Vargas, L., Suárez, L., Robles. I., & Mercado, R. (2004). Aplicación de la programación lineal en el sector forestal, (Volumen 3 (24)), 19–26.
37. Vega, R. (2000). *Bases generales para el diseño e implementación de un algoritmo eficiente en el problema de trozado de árboles*. Universidad de Concepción, Concepción, Chile.
38. Weintraub, A., & Romero, C. (2006). Operations Research Models and the Management of Agricultural and Forestry Resources: A Review and Comparison. *Interfaces*, *36*(5), 446–457. <http://doi.org/10.1287/inte.1060.0222>
39. Zhang, Y., Toppinen, A., & Uusivuori, J. (2014). Internationalization of the forest products industry: A synthesis of literature and implications for future research. *Forest Policy and Economics*, *38*, 8–16. <http://doi.org/10.1016/j.forpol.2013.06.017>

Anexo 1.

Método Generación de Columnas

Una de las herramientas de programación matemática usada en la planificación forestal es la generación de columnas, a continuación se proporcionan los detalles de su uso.

El método de Generación de Columnas puede ser considerado como uno de los más exitosos métodos para resolver problemas que consideran variables de naturaleza entera que son de gran tamaño. La idea general fue introducida por Gilmore y Gomory para obtener la solución óptima del problema *Cutting – Stock* o problemas de corte, donde el Sub-Problema tiene la estructura de un *Knapsack problem*. El problema consiste, en cortar un trozo de género en diferentes tamaños para satisfacer la demanda generada por trozos de tamaños específicos. Dado esto se desarrolló una metodología, que comienza con un conjunto factible de configuraciones de corte y la solución de este problema simplificado es utilizada para generar una nueva configuración potencial. Posteriormente, este procedimiento fue utilizado por gran cantidad de científicos en el sentido de modificar las estructuras de problemas combinatorios y para resolver problemas de distribución y problemas de configuración de equipos de trabajo. (Duarte Alleuy, Blas, 2009).

La técnica del método de generación de columnas consiste, en resolver problemas de programación lineal donde las columnas (variables del problema) no son conocidas o no es posible generarlas explícitamente. Para resolver estos problemas se empieza con un problema maestro, el cual es una relajación lineal del problema original y debe de tener una estructura relativamente simple (se genera solo un número pequeño de columnas, necesario para obtener una solución factible para el problema relajado). También, se dispone de un sub-problema (al que se le denomina sub-problema de *pricing*) que permite identificar columnas (variables) adicionales, que no han sido incluidas en el problema maestro y que mejoren el valor de la función objetivo. (Díaz Smith, Santiago, 2009).

Los problemas de corte y empaquetamiento pertenecen, a la categoría de problemas de optimización combinatoria denominados *Np-completos* debido a que el espacio de soluciones crece de forma exponencial de acuerdo al número de piezas a ser ubicadas. Si por ejemplo se tienen n piezas a ser ubicadas, entonces el espacio de soluciones estará dado por $2^n * n!$ (Jakobs, 1996). Este tipo de problema es fácil de definir, intuitivamente aunque no son fáciles, de modelar formalmente y presentan grados de dificultad que hacen difícil el manejo computacional. En

ellos se tiene un conjunto de piezas de diferentes tamaños y formas, que deben ser localizadas sobre un tablero de material de mayor tamaño sin superponerse unas piezas sobre otras. El objetivo de tal disposición, es maximizar el área utilizada de forma que se generen la menor cantidad de área desperdiciada. Existen diferentes criterios para clasificar los problemas de empaquetado, como son el tamaño y la cantidad de las piezas, las dimensiones del problema, etc. (Toro, E, Garcés, A, & Ruiz, H, 2008).

Actualmente, el método de generación de columnas es ampliamente usado en una gran cantidad de problemas, como por ejemplo: problemas de rutas de vehículos, algunos problemas del agente viajero, programación de horarios para equipos de trabajo, etc. (Lübbecke, M & Desrosiers, J, 2005).

Para explicar de mejor manera el método de generación de columnas, tomaremos el ejemplo expuesto en el libro de Chvátal, V, 1946, ilustrándolo, con el problema de corte de rollos de géneros, donde se debe satisfacer la demanda de clientes con productos de diferentes características.

La Tabla 1.1 muestra la cantidad de productos solicitados por los clientes, t_i , y las características de largo de estos, b_i . Además, el largo de los rollos para cortar es de 100 mm, W . (Figura 1.1).

Tabla A.1.1. Resumen de pedidos de clientes y características.

	b_i	t_i (mm)
1	97	45
2	610	36
3	395	31
4	211	15

Chvátal, V, 1946

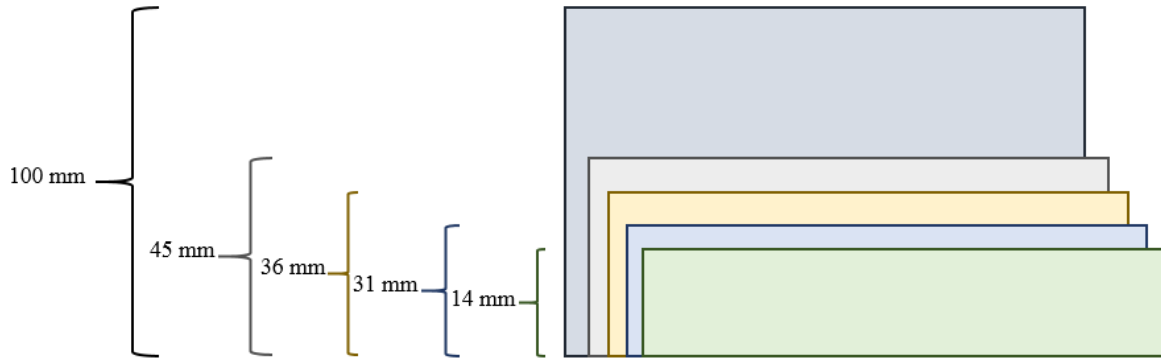


Figura A.1.1. Representación característica de productos solicitados.

El problema maestro lo que se busca es satisfacer los requerimientos de los clientes, utilizando la menor cantidad de rollos, x_j . Esto queda representado por el siguiente modelo, observar expresiones (A.1.2) a la (A.1.3):

$$\text{Minimizar } z = \sum_{j \in J} x_j \quad (\text{A.1.1})$$

Sujeto a:

$$\sum_{j \in J} a_{ij} x_j = b_i \quad \forall i \in I \quad (\text{A.1.2})$$

$$x_j \geq 0 \quad \forall j \in J \quad (\text{A.1.3})$$

En vez de resolver el problema entero, se resuelve la relajación lineal y así, obtener una cota inferior para la solución óptima del problema. Por lo tanto se relajan las restricciones de integridad. Para el ejemplo el método de generación de columnas puede ser iniciado por el problema maestro de la expresión (1.4):

$$\left[\begin{array}{cccc|c} 2 & 0 & 0 & 0 & 97 \\ 0 & 2 & 0 & 0 & 610 \\ 0 & 0 & 3 & 0 & 395 \\ 0 & 0 & 0 & 7 & 211 \end{array} \right] y \bar{b} = \left[\begin{array}{c} 48,5 \\ 305 \\ 131,66 \\ 30,14 \end{array} \right] \quad (\text{A.1.4})$$

Esto quiere decir que se generan 4 patrones iniciales. El patrón 1 en donde se obtienen 2 productos del tipo 1 y se repite 48,5 veces, el patrón 2 en donde, se obtienen 2 productos del tipo 2 y se repite 305 veces y así sucesivamente.

Se puede inicializar la generación de columnas con m patrones de corte en donde el i -ésimo patrón solo genere productos de ancho t_i .

Cuando tenemos el problema maestro definido, se empiezan las iteraciones.

Paso 1

Se resuelve el sistema $w = c_B B^{-1}$, para obtener las variables duales del problema (Chvátal, V, 1946). En el problema de corte o “cutting stock” c_B es igual a un vector cuyas componentes son todas 1. Para el ejemplo queda como la expresión (1.5).

$$w = [1 \quad 1 \quad 1 \quad 1] \begin{vmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/7 \end{vmatrix} = [1/2 \quad 1/2 \quad 1/3 \quad 1/7] \quad (\text{A.1.5})$$

Paso 2

Se busca números enteros no negativos, a_1, a_2, a_3, a_4 , que será la columna que entra a nuestra matriz básica, de tal manera que:

$$\sum_{i=1}^4 w_i a_i > 1 \quad (\text{A.1.6})$$

$$\sum_{i=1}^4 t_i a_i \leq W \quad (\text{A.1.7})$$

$$a_i \in \mathbb{Z} \forall i = 1, \dots, 4 \quad (\text{A.1.8})$$

Para verificar si existe un patrón de corte que mejore el valor de la función objetivo del problema maestro, se resuelve el siguiente problema de la mochila:

$$\text{Maximizar } z = \sum_{i=1}^4 w_i a_i \quad (\text{A.1.9})$$

Sujeto a:

$$\sum_{i=1}^4 t_i a_i \leq W \quad (\text{A.1.10})$$

$$a_i \in \mathbb{Z} \forall i = 1, \dots, 4 \quad (\text{A.1.11})$$

Si el valor de la función objetivo del problema esclavo (o sub problema) es mayor a 1, se continúa con el siguiente paso, de lo contrario se termina el algoritmo. El problema esclavo para el ejemplo planteado, es el siguiente:

$$\text{Maximizar } z = \frac{1}{2}a_1 + \frac{1}{2}a_2 + \frac{1}{3}a_3 + \frac{1}{7}a_4 \quad (\text{A.1.12})$$

Sujeto a:

$$45a_1 + 36a_2 + 31a_3 + 15a_4 \leq 100 \quad (\text{A.1.13})$$

$$a_1, a_2, a_3, a_4 \in \mathbb{Z} \quad (\text{A.1.14})$$

Resolviendo el problema esclavo se obtiene, la siguiente solución:

$$a_5 = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 2 \end{bmatrix} y z = 1,2857 \quad (\text{A.1.15})$$

Paso 3

Con el resultado del problema esclavo se resuelve el sistema: $y_5 = B^{-1}a_5$, expuesto a continuación:

$$y_5 = \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/7 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 2/7 \end{bmatrix} \quad (\text{A.1.16})$$

Paso 4

Para determinar la columna que sale de la base, para ser intercambiada por la columna a_5 , se debe calcular $\frac{\bar{b}_r}{y_{rk}}$ donde, el índice r se determina utilizando el criterio de razón mínima según la siguiente relación (Chvátal, V, 1946):

$$\frac{\bar{b}_r}{y_{rk}} = \text{mínimo}_{1 \leq k \leq m} \left\{ \frac{\bar{b}_r}{y_{rk}} : y_{rk} > 0 \right\} = x_k \quad (\text{A.1.17})$$

Y obteniendo los siguientes resultados,

$$\frac{\bar{b}_2}{y_{2k}} = \frac{305}{1} = 305 \quad (\text{A.1.18})$$

$$\frac{\bar{b}_4}{y_{4k}} = \frac{30,4}{\frac{2}{7}} = 105,5 \quad (\text{A.1.19})$$

Por lo tanto, como 105,5 es menor que 305, la columna que se intercambia por la columna a_5 es la columna a_4 .

Paso 5

A continuación, se modifica la base B, y como se mencionó en el paso anterior, se intercambia la columna a_5 por la a_4 . Mientras que en \bar{b} , a los valores distintos de 0 en y_5 , se les resta el valor $\frac{\bar{b}_4}{y_{4k}}$ y en la posición \bar{b}_4 se reemplaza el valor por $\frac{\bar{b}_4}{y_{4k}}$:

$$B = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 2 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} y \bar{b} = \begin{bmatrix} 48,5 \\ 305 - \frac{\bar{b}_4}{y_{4k}} \\ 131,66 \\ \frac{\bar{b}_4}{y_{4k}} \end{bmatrix} = \begin{bmatrix} 48,5 \\ 199,5 \\ 131,66 \\ 105,5 \end{bmatrix} \quad (\text{A.1.20})$$

Una vez terminado el paso 5, se inicia una nueva iteración:

Paso 1

$$w = [1 \quad 1 \quad 1 \quad 1] \begin{vmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & -1/2 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/2 \end{vmatrix} = [1/2 \quad 1/2 \quad 1/3 \quad 0] \quad (\text{A.1.21})$$

Paso 2

$$\text{Maximizar } z = \frac{1}{2}a_1 + \frac{1}{2}a_2 + \frac{1}{3}a_3 + 0a_4 \quad (\text{A.1.22})$$

Sujeto a:

$$45a_1 + 36a_2 + 31a_3 + 15a_4 \leq 100 \quad (\text{A.1.23})$$

$$a_1, a_2, a_3, a_4 \in \mathbb{Z} \quad (\text{A.1.24})$$

$$a_6 = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 0 \end{bmatrix} y z = 1,166 \quad (\text{A.1.25})$$

Se observa que en la segunda iteración, el valor de la función objetivo del problema maestro es inferior a la iteración anterior.

Paso 3

$$y_6 = \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & -1/2 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2 \\ 2/3 \\ 0 \end{bmatrix} \quad (\text{A.1.26})$$

Paso 4

$$\frac{\bar{b}_2}{y_{2k}} = \frac{199,5}{\frac{1}{2}} = 399 \quad (\text{A.1.27})$$

$$\frac{\bar{b}_3}{y_{3k}} = \frac{131,66}{\frac{2}{3}} = 197,5 \quad (\text{A.1.28})$$

Como el valor de $\frac{\bar{b}_3}{y_{3k}}$ es menor, que el valor de $\frac{\bar{b}_2}{y_{2k}}$, la columna a intercambiar por la columna a_6 , es la columna a_3 .

Paso 5

$$B = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 1 & 2 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} y \bar{b} = \begin{bmatrix} 48,5 \\ 199,5 - \frac{1}{2} \frac{\bar{b}_3}{y_{3k}} \\ \frac{\bar{b}_3}{y_{3k}} \\ 105,5 \end{bmatrix} = \begin{bmatrix} 48,5 \\ 100,5 \\ 197,5 \\ 105,5 \end{bmatrix} \quad (\text{A.1.29})$$

A continuación, se procede la tercera iteración.

Paso 1

$$w = [1 \quad 1 \quad 1 \quad 1] \begin{vmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & -1/4 & -1/2 \\ 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/2 \end{vmatrix} = [1/2 \quad 1/2 \quad 1/4 \quad 0] \quad (\text{A.1.30})$$

Paso 2

$$\text{Maximizar } z = \frac{1}{2}a_1 + \frac{1}{2}a_2 + \frac{1}{4}a_3 + 0 a_4 \quad (\text{A.1.31})$$

Sujeto a:

$$45a_1 + 36a_2 + 31a_3 + 15a_4 \leq 100 \quad (\text{A.1.32})$$

$$a_1, a_2, a_3, a_4 \in \mathbb{Z} \quad (\text{A.1.33})$$

$$a_7 = \begin{vmatrix} 2 \\ 0 \\ 0 \\ 0 \end{vmatrix} y z = 1 \quad (\text{A.1.34})$$

Como el valor alcanzado, del problema esclavo es igual a 1, el proceso de generación de columnas se detiene y agregamos la restricción que las variables de decisión x_j para $j = 1, \dots, 6$ deben tomar valores enteros. Adicionalmente, la restricción mencionada se resuelve el siguiente problema de programación entera (Díaz Smith, Santiago, 2009):

$$\text{Minimizar } z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \quad (\text{A.1.35})$$

Sujeto a:

$$2x_1 + 0x_2 + 0x_3 + 0x_4 + 0x_5 + 0x_6 \geq 97 \quad (\text{A.1.36})$$

$$0x_1 + 2x_2 + 0x_3 + 0x_4 + 1x_5 + 2x_6 \geq 610 \quad (\text{A.1.37})$$

$$0x_1 + 0x_2 + 3x_3 + 0x_4 + 2x_5 + 0x_6 \geq 395 \quad (\text{A.1.38})$$

$$0x_1 + 0x_2 + 0x_3 + 7 + 0x_5 + 2x_6 \geq 211 \quad (\text{A.1.39})$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \quad (\text{A.1.40})$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \in \mathbb{Z} \quad (\text{A.1.41})$$

El problema anterior entrega como solución óptima, lo siguiente:

$$x = \begin{bmatrix} 49 \\ 100 \\ 0 \\ 0 \\ 198 \\ 106 \end{bmatrix} y z = 453 \quad (\text{A.1.42})$$

La que indica que el patrón, $1 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ se repite 49 veces, el patrón $2 = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \end{bmatrix}$ se repite 100 veces, el patrón, $5 = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 0 \end{bmatrix}$ se repite 198 veces y por último el patrón $6 = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 2 \end{bmatrix}$ se repite 106 veces.

Obteniendo como resultado 453 rollos, usados para satisfacer las demandas de los clientes.

Anexo 2.

Formulación Matemática Completa

La formulación matemática presentada en el Capítulo 4 es la descomposición del modelo de este anexo. La gran diferencia entre ambos modelos esta en la eliminación de la variable binaria H_{iz}^{rt} , variable que toma el valor de 1 cuando el sub rodal z , que es parte del rodal i , es cosechado en el periodo t , bajo la regla de trozado r . Además de la eliminación del parámetro O_i^{srt} , parámetro que nos señala la cantidad de rollizos s que puedo obtener cuando aplica la regla de trozado r , en el periodo t al sub rodal z , que pertenece al rodal i .

En conjunto la variable H_{iz}^{rt} y el parámetro O_i^{srt} , señalan los rendimientos o los rollizos que se puede obtener de los sub rodales al aplicar las diversas reglas de trozados, reglas que son fijas y estimadas a partir de la experiencia de las empresas. O_i^{srt} y H_{iz}^{rt} aparecen en conjunto en dos ocasiones, en la función objetivo señalada por la expresión (A.2.1) y en la expresión (A.2.4), en el modelo maestro de la generación de columnas, señalada en el Capítulo 4, en ambas expresiones mencionadas anteriormente la variable H_{iz}^{rt} y parámetro O_i^{srt} son reemplazadas por la variable R_i^{rt} , variable que indica la cantidad de arboles del rodal i a los cuales se le aplica la regla de trozado r en el periodo t . Observar Figura A.2.1. que representa el reemplazo mencionado.

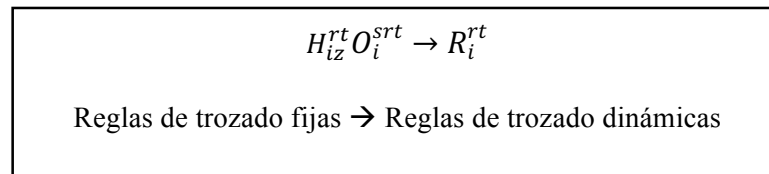


Figura A.2.1. Representación reemplazo de variables entre modelo completo y su descomposición.

El reemplazo señalado en la Figura A.2.1. permite generar reglas de trozado dinámicas, las que se generan de acuerdo a las condiciones del problema tales como la demanda, características de los productos, características de los arboles, entre otros, y permite mayor certeza en la toma de decisiones de la planificación táctica. La importancia del reemplazo también radica en que en la expresión (A.2.4) se obtienen los precios duales del problema maestro para utilizarlos en el problema esclavo.

A continuación se señala el modelo de programación matemática completo, el cual es bastante similar al problema maestro expuesto en el Capítulo 4.

A.2.1 Modelo de Programación Matemática

Índices

- NI:** Número de rodales, $i = 1, 2, \dots, NI$
- NZ:** Numero de sub rodales correspondientes al rodal i , $z = 1, 2, \dots, NZ$;
 $i = 1, 2, \dots, NI$
- NJ:** Número de canchas de trozados, $j = 1, 2, \dots, NJ$
- NK:** Número de clientes del bosque, $k = 1, 2, \dots, NK$
- NT:** Número de periodos de planificación, $t = 1, 2, \dots, NT$
- NP:** Número de tipo de rollizos, $p = 1, 2, \dots, NP$
- NR:** Número de reglas de trozado disponibles, $r = 1, 2, \dots, NR$
- NE:** Número de esquemas de corte, $e = 1, 2, \dots, NE$
- NL:** Número de clientes para los aserraderos, $l = 1, 2, \dots, N$
- NC:** Número de tipo de tablas, $c = 1, 2, \dots, NC$
- S_j:** Conjunto de rodales i que abastecen (entregan producto) a la cancha de trozado j .
- Ab:** Conjunto de tipos de rollizos p desde, los cuales se puede obtener el tipo de tabla b .
- Ak:** Conjunto de destinos del bosque k , que corresponden a aserraderos a planificar.

Parámetros

- O_{iz}^{srt}:** Cantidad en m³ del tipo del rollizo s que se puede obtener del sub rodal z , correspondiente al rodal i , al aplicar la regla de trozado r en el periodo t .
- Dmin_k^{pt}:** Cantidad mínima en m³ del tipo de rollizo p , que demanda el cliente k , en el periodo t .

- $Dmax_k^{pt}$:** Cantidad máxima en m^3 del tipo de rollizo p , que demanda el cliente k , en el periodo t .
- $I max_j$:** Cantidad máxima de rollizos en m^3 de inventario, que se puede almacenar en la cancha j .
- P_k^P :** Precio de venta de $1 m^3$, del tipo de rollizo p , al cliente k .
- M^S :** Costo de cosechar $1 m^3$ de rollizo s .
- T_{jk} :** Costo de transportar $1 m^3$ de rollizos, desde la cancha de trozado j , al cliente k .
- F_i :** Costo fijo de cosecha para el rodal i .
- A :** Costo de almacenar $1 m^3$ de rollizos, durante un periodo de tiempo.
- L :** Valor muy grande.
- Q^{cep} :** Rendimiento en porcentaje, al aplicar el esquema de corte tipo e , al tipo de rollizo p , para producir el tipo de tabla c .
- $Umin_l^{ct}$:** Demanda mínima en m^3 , de tipos de tabla tipo c demandados, por el cliente l en el periodo t .
- $Umax_l^{ct}$:** Demanda máxima en m^3 , de tipos de tabla tipo c demandados, por el cliente l en el periodo t .
- $Smax$:** Cantidad máxima en m^3 de inventario de tablas, que se puede almacenar.
- C^{pe} :** Costo de utilizar los rollizos tipo p , bajo el esquema de corte e .
- V^c :** Costo de fabricar $1 m^3$, de tablas de tipo c .
- E_l^c :** Precio de venta de $1 m^3$ del tipo de tabla c , al cliente l .
- N :** Costo de almacenar $1 m^3$ de cualquier tipo de tabla, durante un periodo de tiempo.

Variables

- R_i^t :** Variable binaria que toma el valor 1 si algún sub rodal del rodal i en el periodo t es cosechado y 0 en caso contrario.

- H_{iz}^{rt} : Variable binaria que toma el valor 1 si se utiliza la regla de trozado r en el sub rodal z del rodal i en el periodo t y 0 en caso contrario.
- I_j^{pt} : Cantidad en m^3 del tipo de rollizo p , almacenado en la cancha de trozado j , en el periodo t .
- Y_{jk}^{pt} : Cantidad en m^3 del tipo de rollizo p , transportado desde la cancha de trozado j hasta el cliente k , en el periodo t .
- X^{pet} : Cantidad en m^3 de tipos de rollizos p , que se consumen utilizando el esquema de corte e en el periodo de tiempo t .
- S^{ct} : Cantidad en m^3 , de tipos de tablas c , que se almacenan en el periodo de tiempo t .
- B_l^{ct} : Cantidad en m^3 , de tipos de tablas c , enviadas al cliente l , en el periodo t .

Modelo de Programación Matemática

$$\begin{aligned}
MAX \quad & \sum_{t=1}^{NT} \sum_{p=1}^{NP} \sum_{k=1}^{NK} \sum_{j=1}^{NJ} P_k^p Y_{jk}^{pt} - \sum_{t=1}^{NT} \sum_{i=1}^{NI} \sum_{z=1}^{NZ} \sum_{r=1}^{NR} M^s H_{iz}^{rt} O_i^{srt} - \sum_{t=1}^{NT} \sum_{p=1}^{NP} \sum_{k=1}^{NK} \sum_{j=1}^{NJ} T_{jk} Y_{jk}^{pt} \\
& - \sum_{t=1}^{NT} \sum_{i=1}^{NI} F_i R_i^t - \sum_{t=1}^{NT} \sum_{j=1}^{NP} \sum_{j=1}^{NJ} A I_j^{pt} - \sum_{p=1}^{NP} \sum_{e=1}^{NE} \sum_{t=1}^{NT} C^{pe} X^{pet} \\
& + \sum_{t=1}^{NT} \sum_{l=1}^{NL} \sum_{c=1}^{NC} E_l^c B_l^{ct} - \sum_{l=1}^{NL} \sum_{c=1}^{NC} \sum_{t=1}^{NT} V^c B_l^{ct} - \sum_{c=1}^{NC} \sum_{t=1}^{NT} N S^{ct}
\end{aligned} \tag{A.2.1}$$

Sujeto a

$$\sum_{t=1}^{NT} \sum_{r=1}^{NR} H_{iz}^{rt} \leq 1; \quad \forall i = 1, \dots, NI; \quad \forall z = 1, \dots, NZ. \tag{A.2.2}$$

$$\sum_{i=1}^{NI} L R_i^t \geq \sum_{i=1}^{NI} \sum_{z=1}^{NZ} \sum_{r=1}^{NR} H_{iz}^{rt}; \quad \forall t = 1, \dots, NT \tag{A.2.3}$$

$$\sum_{k=1}^{NK} Y_{jk}^{pt} = \sum_{i \in S_j} \sum_{r=1}^{NR} \sum_{z=1}^{NZ} H_{iz}^{rt} O_i^{srt} + I_j^{p,t-1} - I_j^{pt}; \quad \forall j = 1, \dots, NJ; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT. \tag{A.2.4}$$

$$Dmin_k^{pt} \leq \sum_{j=1}^{Nj} Y_{jk}^{pt} \leq Dmax_k^{pt}; \quad \forall p = 1, \dots, Np; \quad \forall t = 1, \dots, NT; \quad \forall k = 1, \dots, NK \tag{A.2.5}$$

$$\sum_{p=1}^{NP} I_j^{pt} \leq Imax_j; \quad \forall j = 1, \dots, NJ; \quad \forall t = 1, \dots, NT. \tag{A.2.6}$$

$$I_j^{p0} = 0; \quad \forall p = 1 \dots NP; \quad \forall j = 1, \dots, NJ \quad (\text{A.2.7})$$

$$\sum_{e=1}^{NE} X^{pet} \leq \sum_{k \in Ak} \sum_{j=1}^{Nj} Y_{jk}^{pt}; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT. \quad (\text{A.2.8})$$

$$\sum_{l=1}^{NL} B_l^{ct} = \sum_{e=1}^{NE} Q^{cep} X^{cet} + S^{c,t-1} - S^{ct}; \quad \forall c = 1, \dots, NC; \quad \forall t = 1, \dots, NT \quad (\text{A.2.9})$$

$$Umin_l^{ct} \leq B_l^{ct} \leq Umax_l^{ct}; \quad \forall c = 1, \dots, NC; \quad \forall t = 1, \dots, NT; \quad \forall l = 1, \dots, NL. \quad (\text{A.2.10})$$

$$\sum_{c=1}^{NC} S^{ct} \leq Smax; \quad \forall t = 1, \dots, NT \quad (\text{A.2.11})$$

$$S^{c0} = 0; \quad \forall c = 1, \dots, NC \quad (\text{A.2.12})$$

$$I_j^{pt}, Y_{jk}^{pt}, X^{pet}, B_l^{ct}, S^{ct}, R^{rt} \geq 0 \quad (\text{A.2.13})$$

$$\forall i = 1, \dots, NI; \quad \forall j = 1, \dots, NJ; \quad \forall k = 1, \dots, NK; \quad \forall p = 1, \dots, NP; \quad \forall t = 1, \dots, NT; \quad \forall e = 1, \dots, NE; \quad \forall l = 1, \dots, NL; \quad \forall c = 1, \dots, NC \quad (\text{A.2.14})$$

La función objetivo está expuesta en la expresión (A.2.1), la que plantea la maximización y para su comprensión es dividida en dos partes, una que representa lo que se realiza en el bosque, con los beneficios que se obtienen por la venta de rollizos descontando los costos operacionales, de transporte y de almacenamiento y la segunda, corresponde a lo realizado en los aserraderos, en donde se encuentran los ingresos por ventas de tablas (descontando los gastos operacionales y de almacenamiento correspondientes).

La expresión (A.2.2) nos indica el conjunto de restricciones que expresan que a cada sub rodal sólo se le puede asignar una regla de trozados y puede ser cosechado solo una vez en todos los períodos de tiempo. Esto ocurre para todos los rodales y sus sub rodales.

La expresión (A.2.3) es el conjunto de restricciones, el cual nos indica que la variable binaria R_i^t multiplicada por un número bastante grande será mayor o igual a la variable H_{iz}^{rt} . Este parámetro L es el que da un valor muy grande, en la realidad el valor más extremo que pudiese adoptar es la multiplicación entre la cantidad de sub rodales y el número de reglas utilizadas. Esta restricción se encarga de darle el valor de 1 al rodal siempre cuando al menos uno de sus sub rodales sea cosechado en el mismo período de tiempo. Esto ocurre para cada período de tiempo.

La expresión (A.2.4) muestra el conjunto de restricciones que nos expone el balance de rollizos en el bosque. En un lado de la expresión se encuentra la cantidad de rollizos transportado desde las canchas de trozados hacia los clientes, la cual debe ser igual a los rollizos obtenidos en la cosecha forestal, añadiéndole la madera que es extraída desde el inventario, la cual es representada por la cantidad de inventario del período anterior menos la cantidad del período actual. La cantidad de rollizos obtenidos en la cosecha forestal es mostrada por el parámetro de rendimiento O_{iz}^{srt} . Si el sub rodal se explota, se le aplica el rendimiento O_{iz}^{srt} al sub rodal y la regla correspondiente, esta decisión es expresada por la variable binaria H_{iz}^{rt} . Esto se repite para cada cancha de trozado, cada producto y período de tiempo.

La expresión (A.2.5) nos indica el conjunto de restricciones que expresan la demanda de rollizos de cada cliente del bosque y en cada periodo de tiempo, que debe ser satisfecha. Esta demanda no es puntual, si no que un rango de demanda en el cual la variable Y_{jk}^{pt} , que representa a los rollizos entregados desde las canchas de trozados a los clientes del bosque, se debe encontrar. Esto se repite para cada período de tiempo.

La expresión (A.2.6) nos indica el conjunto de restricciones, que se preocupan que el inventario existente en cada cancha de trozado y en cada período de tiempo no exceda el inventario máximo. Esto ocurre para cada cancha de trozado y cada período de tiempo.

La expresión (A.2.7) establece que los niveles de inventario al momento de comenzar las operaciones, período 0, son nulos.

La expresión (A.2.8), es la que nos indica la unión del bosque con el aserradero mediante el conjunto de restricciones que exponen que la cantidad de rollizos que se pueden consumir para generar tablas bajo ciertos esquemas de corte en un período de tiempo, X^{pet} debe ser menor o igual a la cantidad de rollizos que fueron entregadas en el aserradero en el mismo período de tiempo, Y_{jk}^{pt} . Esto se comprueba para cada producto y cada período de tiempo.

La expresión (A.2.9), nos muestra el conjunto de restricciones que exponen el balance de materia entre rollizos y tablas. Por un lado se encuentran las tablas que son enviadas a diferentes destinos y clientes del aserradero, en un período de tiempo, las cuales deben ser iguales a las tablas obtenidas de los rollizos, añadiéndole las tablas que fueron sacadas desde inventario, las cuales son expresadas por la diferencia entre el inventario del período anterior con el del período actual.

La expresión (A.2.10), es el conjunto de restricciones que indican, la demanda mínima de tablas que exigen los clientes del aserradero y la demanda máxima, de tablas que estos clientes aceptan. El valor de las tablas enviadas, debe estar dentro de este rango. Este proceso se repite para cada período de tiempo

La expresión (A.2.11) es el conjunto de restricciones que establecen que el inventario de los aserraderos no exceda el inventario máximo permitido en cada período de tiempo.

La expresión (A.2.12) indica que el inventario al momento de comenzar las operaciones en los aserraderos, período 0, es nulo.

La expresión (A.2.13) indica las restricciones de no negatividad para las variables.

La expresión (A.2.14) indica los índices que participan.

Anexo 3.

Código del Modelo Matemático Implementado en Cplex 12.6.1

```
#include <vector>
#include <string>
#include <sstream>
#include <ilcplex/ilocplex.h>

std::string makename(std::string _f, int a, int b, int c) {
    std::stringstream nn;
    nn << _f;

    if ( a == -1 ) return nn.str();
    nn << "_" << a;

    if ( b == -1 ) return nn.str();
    nn << "_" << b;

    if ( c == -1 ) return nn.str();
    nn << "_" << c;

    return nn.str();
}

ILOSTLBEGIN

int main(int argc, char *argv[]) {
    IloEnv env;

    try {

        std::cerr << "CortaArbol build " << __DATE__ << " " << __TIME__ <<
std::endl;
        std::cout.setf(std::ios_base::fixed);

        string instancename = "./";
        IloModel ModeloMaestro (env);

        IloInt NBosques;
        IloInt NProductos;
        IloInt NTiempo;
        IloInt NCanchas;
        IloInt NClientes;
        IloInt NEsquemas;
        IloInt NTablas;
        IloInt NDestino;

        ifstream Indices (( instancename + "Indices.txt").c_str() );
        if (! Indices){
            cout << "Error opening file Indices.txt" << endl;
            return EXIT_FAILURE;
        }
        Indices >> NBosques;
        Indices >> NProductos;
        Indices >> NTiempo;
        Indices >> NCanchas;
        Indices >> NClientes;
        Indices >> NEsquemas;
        Indices >> NTablas;
        Indices >> NDestino;
        Indices.close();
```

```

IloArray<IloBoolArray> Sj (env, NCanchas);
ifstream datos_Sj( ( instancename + "Sj.txt").c_str() );
if (! datos_Sj){
    cout << "Error opening file Sj.txt" << endl;
    return EXIT_FAILURE;
}
for (int i=0; i < NCanchas; ++i){
    Sj [i] = IloBoolArray (env, NBosques);
    for(int j = 0; j < NBosques; j++){
        datos_Sj >> Sj[i][j];
    }
}
datos_Sj.close();

IloBoolArray Ak (env, NClientes);
ifstream datos_Ak ( ( instancename + "Ak.txt").c_str ( ) );
if (! datos_Ak){
    cout << "Error opening file Ak.txt" << endl;
    return EXIT_FAILURE;
}
for(int i=0; i<NClientes; ++i){
    datos_Ak >> Ak[i];
}
datos_Ak.close();

IloArray<IloBoolArray> Ab (env, NProductos);
ifstream datos_Ab ( ( instancename + "Ab.txt").c_str ( ) );
if (! datos_Ab){
    cout << "Error opening file Ab.txt" << endl;
    return EXIT_FAILURE;
}
for (int i = 0; i < NProductos; ++i){
    Ab[i] = IloBoolArray (env, NTablas);
    for(int j = 0; j < NTablas; ++j){
        datos_Ab>>Ab[i][j];
    }
}
datos_Ab.close();

IloArray<IloNumArray> T(env, NCanchas);
ifstream datos_T ( ( instancename + "T.txt").c_str() );
if (! datos_T){
    cout << "Error opening file T.txt" << endl;
    return EXIT_FAILURE;
}
for( int i = 0; i < NCanchas; ++i){
    T[i]=IloNumArray(env, NClientes);
    for(int j = 0; j < NClientes; ++j){
        datos_T>>T[i][j];
    }
}
datos_T.close();

IloNum L;
ifstream datos_L ( ( instancename + "L.txt").c_str ( ) );
if (! datos_L){
    cout << "Error opening file L.txt" << endl;
    return EXIT_FAILURE;
}

```



```

datos_L >> L;

IloArray<IloNumArray> C(env, NProductos);
ifstream datos_C ( ( instancename + "C.txt").c_str ( ) );
if (! datos_C){
cout << "Error opening file C.txt" << endl;
return EXIT_FAILURE;
}
for( int i = 0; i < NProductos; ++i){
    C[i]= IloNumArray (env, NESquemas);
    for ( int j = 0; j < NESquemas; ++j){
        datos_C>>C[i][j];
    }
}
datos_C.close();

IloNumArray V (env, NTablas);
ifstream datos_V ( ( instancename + "V.txt").c_str ( ) );
if (! datos_V){
    cout << "Error opening file V.txt" << endl;
return EXIT_FAILURE;
}
for(int i = 0; i < NTablas ; ++i){
    datos_V>>V[i];
}
datos_V.close();

IloNum N;
ifstream datos_N ( ( instancename + "N.txt").c_str ( ) );
if (! datos_N){
cout << "Error opening file N.txt" << endl;
return EXIT_FAILURE;
}
datos_N>>N;
datos_N.close();

IloNumArray BosqueDiaMin(env, NBosques);
ifstream BDiaMin ((instancename + "BDiaMin.txt").c_str() );
if ( ! BDiaMin){
    cout << "Error opening file BDiamin.txt" << endl;
return EXIT_FAILURE;
}
for (int i = 0; i < NBosques; i++) {
    BDiaMin >> BosqueDiaMin[i];
}
BDiaMin.close();

IloNumArray BosqueDiaMax(env, NBosques);
ifstream BDiaMax ((instancename + "BDiaMax.txt").c_str() );
if ( ! BDiaMax){
    cout << "Error opening file BDiaMax.txt" << endl;
return EXIT_FAILURE;
}
for (int i = 0; i < NBosques; i++) {
    BDiaMax >> BosqueDiaMax[i];
}
BDiaMax.close();

```

```

IloNumArray BosqueDiaLargo(env, NBosques);
ifstream BLargo ((instancename + "BLargo.txt").c_str() );
if ( !BLargo){
    cout << "Error opening file BLargo.txt" << endl;
    return EXIT_FAILURE;
}
for (int i = 0; i < NBosques; i++) {
    BLargo >> BosqueDiaLargo[i];
}
BLargo.close();

IloNumArray BosqueDiaCantidad(env, NBosques);
ifstream BCantidad ((instancename + "BCantidad.txt").c_str() );
if( ! BCantidad){
    cout << "Error opening file BCantidad.txt" << endl;
    return EXIT_FAILURE;
}
for (int i = 0; i < NBosques; i++) {
    BCantidad >> BosqueDiaCantidad[i];
}
BCantidad.close();

IloNumArray BosqueCosto(env, NBosques);
ifstream BCosto ((instancename + "BCosto.txt").c_str() );
if ( !BCosto){
    cout << "Error opening file BCosto.txt" << endl;
    return EXIT_FAILURE;
}
for (int i = 0; i < NBosques; i++) {
    BCosto >> BosqueCosto[i];
}
BCosto.close();

IloNumArray PedidoDiametro(env, NProductos);
ifstream Diametro (( instancename + "PDMin.txt").c_str() );
if (! Diametro){
    cout << "Error opening file PDMin.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i = 0; i < NProductos; i++){
    Diametro >> PedidoDiametro[i];
}
Diametro.close();

IloNumArray PedidoLargo(env, NProductos);
ifstream Largo (( instancename + "PLargo.txt").c_str() );
if ( ! Largo){
    cout << "Error opening file PLargo.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i = 0; i < NProductos; i++){
    Largo >> PedidoLargo[i];
}
Largo.close();

IloNumArray PedidoTolerancia(env, NProductos);
ifstream Tol ((instancename + "PTOL.txt").c_str() );
if ( ! Tol){

```

```

        cout << "Error opening file PTOL.txt" << endl;
        return EXIT_FAILURE;
    }
    for (int i = 0; i < NProductos; i++){
        Tol >> PedidoTolerancia[i];
    }
    Tol.close();

IloNumArray PedidoDemMin(env, NProductos);
ifstream Min (( instancename + "PDemMin.txt").c_str() );
if ( ! Min){
    cout << "Error opening file PDemMin.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i = 0; i < NProductos; i++){
    Min >> PedidoDemMin[i];
}
Min.close();

IloNumArray PedidoDemMax(env, NProductos);
ifstream Max (( instancename + "PDemMax.txt").c_str() );
if ( !Max){
    cout << "Error opening file PDemMax.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i = 0; i < NProductos; i++){
    Max >> PedidoDemMax[i];
}
Max.close();

IloNumArray PedidoCosto(env, NProductos);
ifstream Precio ((instancename +"PCosto.txt").c_str() );
if ( !Precio){
    cout << "Error opening file PCosto.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i = 0; i < NProductos; i++){
    Precio >> PedidoCosto[i];
}
Precio.close();

IloArray<IloNumArray> P (env,NProductos);
ifstream datos_Pp (( instancename + "Pp.txt").c_str() );
if (! datos_Pp){
    cout << "Error opening file P.txt" << endl;
    return EXIT_FAILURE;
}
for(int i = 0; i < NProductos; ++i){
    P[i]= IloNumArray (env, NClientes);
    for(int j = 0; j < NClientes; ++j){
        datos_Pp>>P[i][j];
    }
}
datos_Pp.close();

IloArray<IloArray<IloNumArray> > Dmin(env, NTiempo);
ifstream datos_Dmin ( ( instancename + "Dmin.txt").c_str() );
if (! datos_Dmin){
    cout << "Error opening file Dmin.txt" << endl;
}

```

```

        return EXIT_FAILURE;
    }
    for(int i = 0; i < NTiempo; ++i){
        Dmin[i]=IloArray<IloNumArray>(env, NClientes);
        for (int j=0; j<NClientes; ++j){
            Dmin[i][j]=IloNumArray (env, NProductos);
            for (int k=0; k< NProductos; ++k){
                datos_Dmin >>Dmin[i][j][k];
            }
        }
    }
    datos_Dmin.close();

    IloArray<IloArray<IloNumArray> > Dmax(env, NTiempo);
    ifstream datos_Dmax ( ( instancename + "Dmax.txt").c_str() );
    if (! datos_Dmax){
        cout << "Error opening file Dmax.txt" << endl;
        return EXIT_FAILURE;
    }
    for(int i = 0; i < NTiempo; ++i){
        Dmax[i]=IloArray<IloNumArray>(env, NClientes);
        for (int j = 0; j<NClientes; ++j){
            Dmax[i][j] = IloNumArray (env, NProductos);
            for (int k = 0; k< NProductos; ++k){
                datos_Dmax >>Dmax[i][j][k];
            }
        }
    }
    datos_Dmax.close();

    IloNumArray Imax(env, NCanchas);
    ifstream datos_Imax ( ( instancename + "Imax.txt").c_str ( ) );
    if (! datos_Imax){
        cout << "Error opening file Imax.txt" << endl;
        return EXIT_FAILURE;
    }
    for( int i = 0; i < NCanchas; ++i){
        datos_Imax>>Imax[i];
    }
    datos_Imax.close();

    IloArray<IloArray<IloNumArray>> Umin (env, NTiempo);
    ifstream datos_Umin ( ( instancename+ "Umin.txt").c_str ( ) );
    if (! datos_Umin){
        cout << "Error opening file Umin.txt" << endl;
        return EXIT_FAILURE;
    }
    for( int i = 0; i < NTiempo; ++i){
        Umin [i] = IloArray<IloNumArray> (env, NDestino);
        for( int j = 0; j < NDestino; ++j){
            Umin[i][j] = IloNumArray (env, NTablas);
            for(int k = 0; k < NTablas; ++k){
                datos_Umin >> Umin[i][j][k];
            }
        }
    }
    datos_Umin.close();

    IloArray<IloArray<IloNumArray>> Umax (env, NTiempo);
    ifstream datos_Umax ( ( instancename+ "Umax.txt").c_str ( ) );

```

```

if (! datos_Umax){
    cout << "Error opening file Umax.txt" << endl;
    return EXIT_FAILURE;
}
for( int i = 0; i < NTiempo; ++i){
    Umax [i] = IloArray<IloNumArray> (env, NDestino);
    for( int j = 0; j < NDestino; ++j){
        Umax[i][j] = IloNumArray (env, NTablas);
        for(int k = 0; k < NTablas; ++k){
            datos_Umax >> Umax[i][j][k];
        }
    }
}
datos_Umax.close();

IloNum Smax;
ifstream datos_Smax ( ( instancename + "Smax.txt").c_str() );
if (! datos_Smax){
    cout << "Error opening file Smax.txt" << endl;
    return EXIT_FAILURE;
}
datos_Smax >> Smax;
datos_Smax.close();

IloNumArray F (env, NBosques);
ifstream datos_F ( ( instancename + "F.txt").c_str() );
if ( ! datos_F){
    cout << "Error opening file F.txt" << endl;
    return EXIT_FAILURE;
}
for ( int i =0; i < NBosques; i++){
    datos_F >> F[i];
}
datos_F.close();

IloArray<IloArray<IloNumArray>> Q(env, NEsquemas);
ifstream datos_Q ( ( instancename + "Q.txt").c_str ( ) );
if( ! datos_Q){
    cout << "Error opening file Q.txt" << endl;
    return EXIT_FAILURE;
}
for(int i = 0; i < NEsquemas; ++i){
    Q[i] = IloArray<IloNumArray> (env, NProductos);
    for(int j = 0; j < NProductos; ++j){
        Q[i][j] = IloNumArray (env, NTablas);
        for(int k = 0; k < NTablas; ++k){
            datos_Q>>Q[i][j][k];
        }
    }
}
datos_Q.close();

IloArray<IloNumArray> E (env, NTablas);
ifstream datos_E ( ( instancename + "E.txt").c_str ( ) );
if (! datos_E){
    cout << "Error opening file E.txt" << endl;
    return EXIT_FAILURE;
}
for(int i = 0; i < NTablas ; ++i){
    E[i]= IloNumArray (env, NDestino);
}

```

```

        for(int j = 0; j < NDestino; ++j){
            datos_E>>E[i][j];
        }
    }
    datos_E.close();

IloArray<IloArray<IloArray<IloNumVarArray> > > Y (env, NTiempo);
for ( int i = 0; i < NTiempo; i++){
    Y[i] = IloArray<IloArray<IloNumVarArray> > (env, NProductos);
    for ( int j = 0; j < NProductos; j++){
        Y[i][j]=IloArray<IloNumVarArray> (env, NCanchas);
        for ( int k = 0; k < NCanchas; k++){
            Y[i][j][k]=IloNumVarArray (env, NClientes, 0, IloInfinity);
            for ( int l = 0; l < NClientes; l++){
                std :: stringstream nn;
                nn << "Y_" << i+1 << "_" << j+1 << "_" << k+1 << "_" << l+1;
                Y[i][j][k][l].setName(nn.str().c_str());
            }
        }
    }
}

IloArray<IloArray<IloNumVarArray> > I (env, NTiempo);
for ( int i = 0; i < NTiempo; i++){
    I[i] = IloArray<IloNumVarArray> (env, NCanchas);
    for ( int j = 0; j < NCanchas; j++){
        I[i][j] = IloNumVarArray (env, NProductos, 0, IloInfinity);
        for (int k = 0; k < NProductos; k++){
            std :: stringstream nn;
            nn << "I_" << i+1 << "_" << j+1 << "_" << k+1;
            I[i][j][k].setName( nn.str().c_str());
        }
    }
}

IloArray<IloArray<IloNumVarArray>> X (env, NTiempo);
for ( int i = 0; i < NTiempo; i++){
    X[i] = IloArray<IloNumVarArray> (env, NProductos);
    for ( int j = 0; j < NProductos; j++){
        X[i][j] = IloNumVarArray (env, NEsquemas, 0, IloInfinity);

        for ( int k = 0; k < NEsquemas; k++){
            std::stringstream nn;
            nn << "X_" << i+1 << "_" << j+1 << "_" << k+1;
            X[i][j][k].setName(nn.str().c_str());
        }
    }
}

IloArray<IloArray<IloNumVarArray>> B (env, NTiempo);
for (int i = 0; i < NTiempo; ++i){
    B[i] = IloArray<IloNumVarArray> (env, NDestino);
    for (int j = 0; j < NDestino; ++j){
        B[i][j] = IloNumVarArray (env, NTablas, 0, IloInfinity);

        for (int k = 0; k < NTablas; k++){
            std::stringstream nn;
            nn << "B_" << i+1 << "_" << j+1 << "_" << k+1;
            B[i][j][k].setName( nn.str().c_str() );
        }
    }
}

```

```

    }
}

IloArray<IloNumVarArray> S (env, NTiempo);
for(int i = 0; i < NTiempo; ++i){
    S[i] = IloNumVarArray (env, NTablas, 0, IloInfinity);

    for(int j = 0; j < NTablas; j++){
        std::stringstream nn;
        nn << "S_" << i+1 << "_" << j+1;
        S[i][j].setName( nn.str().c_str() );
    }
}

IloArray<IloBoolVarArray> H (env, NTiempo);
for ( int i = 0; i < NTiempo; i++){
    H[i] = IloBoolVarArray (env, NBosques);

    for ( int j = 0; j < NBosques; j++){
        std :: stringstream nn;
        nn << "H_" << i+1 << "_" << j+1;
        H[i][j].setName( nn.str().c_str() );
    }
}

IloExpr temp1 (env);
IloExpr temp2 (env);

IloExpr ingresocos (env);
for ( int i = 0; i < NTiempo; i++){
    for ( int j = 0; j < NProductos; j++){
        for ( int k = 0; k < NClientes; k++){
            for ( int l = 0; l < NCanchas; l++){
                ingresocos += P[j][k]*Y[i][j][l][k];
            }
        }
    }
}

IloExpr ingresoase (env);
for( int i = 0; i < NDestino; i++){
    for ( int j = 0; j < NTablas; j++){
        for ( int k = 0; k < NTiempo; k++){
            ingresoase += E[j][i]*B[k][i][j];
        }
    }
}

IloExpr costot (env);
for ( int i = 0; i < NTiempo; i++){
    for ( int j = 0; j < NProductos; j++){
        for ( int k = 0; k < NCanchas; k++){
            for ( int l = 0; l < NClientes; l++){
                costot += T[k][l]*Y[i][j][k][l];
            }
        }
    }
}

```

```

}

IloExpr costol (env);
for ( int i = 0; i < NTiempo; i++){
    for (int j = 0; j <NProductos; j++){
        for ( int k = 0; k < NCanchas; k++){
            costol += L * I[i][k][j];
        }
    }
}

IloExpr costoc (env);
for (int i = 0; i < NProductos; i++){
    for ( int j = 0; j < NEsquemas; j++){
        for ( int k = 0; k < NTiempo; k++){
            costoc += C[i][j] * X[k][i][j];
        }
    }
}

IloExpr costov (env);
for( int i = 0; i < NDestino; i++){
    for ( int j = 0; j < NTablas; j++){
        for ( int k = 0; k < NTiempo; k++){
            costov += V[j]*B[k][i][j];
        }
    }
}

IloExpr coston (env);
for ( int i = 0; i < NTablas; i++){
    for ( int j = 0; j < NTiempo; j++){
        coston += N * S[j][i];
    }
}

IloExpr fo (env);
fo = costot + costol + costoc + costov + coston - ingresocos - ingresoase;

IloObjective  ArbolesUsados = IloAdd(ModeloMaestro, IloMinimize(env,fo));

fo.end();
costot.end();
costol.end();
costoc.end();
costov.end();
coston.end();

IloRangeArray Demandas = IloAdd(ModeloMaestro, IloRangeArray(env,
PedidoDemMin, PedidoDemMax));
IloRangeArray Oferta = IloAdd(ModeloMaestro, IloRangeArray (env, 0,
BosqueDiaCantidad));
IloNumVarArray PatronCorte(env);
IloNumArray ArbolPatronCorte(env);
IloNumArray TiempoPatronCorte(env);

for (int i = 0; i < PedidoDemMin.getSize(); ++i) {
    PatronCorte.add(IloNumVar(ArbolesUsados(10000000) + Demandas[i](1) ));
    PatronCorte[ PatronCorte.getSize()-1].setName( makeName("Patron",
PatronCorte.getSize(), 0, 0).c_str() );
    ArbolPatronCorte.add( -1 );
    TiempoPatronCorte.add(-1);
}

```



```

}

IloArray<IloArray<IloArray<IloRangeArray> > > Balance (env, NCanchas);
for (int i = 0; i < NCanchas; i++){
    Balance[i] = IloArray<IloArray<IloRangeArray>> (env, NTiempo);
    for (int j = 0; j < NTiempo; j++){
        Balance[i][j] = IloArray<IloRangeArray> (env, NProductos);
        for (int k = 0; k < NProductos; k++){
            Balance [i][j][k] = IloRangeArray (env);

            IloExpr temp1 (env);
            for (int l = 0; l < NClientes; l++){
                temp1 -= Y[j][k][i][l];
            }
            if (j) temp1 += I [j-1][i][k];
            temp1 -= I[j][i][k];

            Balance[i][j][k].add(IloRange(env, 0.0, temp1, 0.0,
makename("ba", i+1, j+1, k+1).c_str() ) );
            ModeloMaestro.add(Balance[i][j][k]);
        }
    }
}

for (int a = 0; a < NTiempo; ++a){
    for (int b=0; b < NCanchas; ++b){
        IloExpr temp2 (env);
        for (int i=0; i < NProductos; ++i){
            temp2 += I[a][b][i];
        }
        ModeloMaestro.add(temp2 <= Imax[b]);
    }
}

for (int i = 0; i < NProductos; i++){
    for ( int j = 0; j < NTiempo; j++){
        for ( int k = 0; k < NEsquemas; k++){
            temp1 += X[j][i][k];
        }
        for ( int l = 0; l < NClientes; l++){
            if (Ak[l]==1){
                for ( int m = 0; m < NCanchas; m++){
                    temp2 += Y[j][i][m][l];
                }
            }
        }
        ModeloMaestro.add(temp1<=temp2);
        temp1.end();
        temp1 = IloExpr(env);
        temp2.end();
        temp2 = IloExpr(env);
    }
}

for(int i = 0; i < NTablas; ++i){
    for(int j = 0; j < NTiempo; ++j){
        for(int l = 0; l < NDestino; ++l){
            temp1 += B[j][l][i];
        }
        for(int k=0; k<NProductos; ++k){

```

```

        if(Ab[k][i]==1){
            for(int e=0; e < NEsquemas; ++e){
                temp2 += Q[e][k][i]*X[j][k][e];
            }
        }
        if(j !=0){
            ModeloMaestro.add(temp1==temp2+S[j-1][i]-S[j][i]);
        }else{
            ModeloMaestro.add(temp1==temp2+S[j][i]);
        }
        temp1.end();
        temp1=IloExpr(env);
        temp2.end();
        temp2=IloExpr(env);
    }
}

for (int i = 0; i < NTiempo; ++i){
    for (int j = 0; j < NDestino; ++j){
        for (int l = 0; l < NTablas; ++l){
            ModeloMaestro.add (Umin[i][j][l]<=B[i][j][l]<=Umax[i][j][l]);
            temp1.end();
            temp1=IloExpr (env);
        }
    }
}

for(int i = 0; i < NTiempo; ++i){
    for(int j = 0; j < NTablas; ++j){
        temp1 += S[i][j];
    }
    ModeloMaestro.add(temp1<=Smax);
    temp1.end();
    temp1=IloExpr (env);
}

IloCplex CplexMaestro(ModeloMaestro);
cout << ModeloMaestro << endl;

IloArray <IloModel> ModelosEsclavos(env);
IloArray <IloObjective> ReducedCostos(env);
IloArray <IloCplex> CplexEsclavos(env);
IloNumVarArray ProductoInt (env, PedidoDemMin.getSize(), 0.0, IloIntMax,
ILOINT);
IloNumVarArray ProductoBool(env, PedidoDemMin.getSize(), 0.0, 1.0, ILOBOOL
);
IloNumVarArray ProductoNum (env, PedidoDemMin.getSize(), 0.0, IloInfinity);

for (int i = 0; i < PedidoDemMin.getSize(); ++i ) {
    ProductoInt[ i ].setName( makename("ProductoInt", i+1, -1, -1).c_str()
);
    ProductoBool[ i ].setName( makename("ProductoBool", i+1, -1, -1).c_str()
);
    ProductoNum[ i ].setName( makename("ProductoNum", i+1, -1, -1).c_str()
);
}

```

```

for (int mei = 0; mei < BosqueDiaLargo.getSize(); ++mei) {
    ModelosEsclavos.add( IloModel(env) );
    ReducedCostos.add(IloAdd(ModelosEsclavos[mei], IloMinimize(env, 1)));

    IloExpr R1(env);
    for (int i = 0; i < PedidoDemMin.getSize(); ++i) {
        R1 += ProductoNum[i] + PedidoLargo[i] * ProductoInt[i];
    }
    ModelosEsclavos[mei].add(R1 <= BosqueDiaLargo[mei]);

    for (int i = 0; i < PedidoDemMin.getSize(); ++i) {
        double delta = (BosqueDiaMax[mei] - BosqueDiaMin[mei]) /
BosqueDiaLargo[mei];

        IloExpr Rd(env);
        Rd += BosqueDiaMin[mei] + delta * ProductoNum[0];

        IloExpr Rdnum(env);
        Rdnum += BosqueDiaMin[mei];

        for (int j = 0; j < i; ++j) {
            Rd += delta * PedidoLargo[j] * ProductoInt[j] + delta *
ProductoNum[j+1];
            Rdnum += delta * PedidoLargo[j] * int(BosqueDiaLargo[mei] /
PedidoLargo[j]);
        }

        ModelosEsclavos[mei].add(Rd / PedidoTolerancia[i] - Rdnum *
ProductoBool[i] <= PedidoDiametro[i] );
        ModelosEsclavos[mei].add(
PedidoDiametro[i] <= Rd * PedidoTolerancia[i] + Rdnum * ProductoBool[i]);

        Rd += delta * PedidoLargo[i] * (ProductoInt[i] - 1);
        Rdnum += delta * PedidoLargo[i] * int(BosqueDiaLargo[mei] /
PedidoLargo[i]);

        ModelosEsclavos[mei].add(Rd / PedidoTolerancia[i] - Rdnum *
ProductoBool[i] <= PedidoDiametro[i] );
        ModelosEsclavos[mei].add(ProductoInt[i] <= int(BosqueDiaLargo[mei] /
PedidoLargo[i]) * ProductoBool[i]);
    }

    CplexEsclavos.add(IloCplex(ModelosEsclavos[mei]));
}

for (;;) {
    cout << "Nuevo Modelo Maestro " << endl << ModeloMaestro << endl;
    CplexMaestro.solve();

    IloNumArray PatronCorteVals (env);
    CplexMaestro.getValues (PatronCorte, PatronCorteVals);
    IloNum PatronCorteSum = 0;
    for ( int nxs = PatronCorte.getSize() - 2 * ( NTiempo * NBosques); nxs
< PatronCorte.getSize(); ++nxs){

        if (nxs < 0) continue ;
        PatronCorteSum += PatronCorteVals[nxs];
    }
}

```

```

    }
    if (PatronCorteSum < 0.01) break;

    cout << endl << "Costo Arboles usados " << CplexMaestro.getObjValue() <<
endl;
    for (int j = 0; j < PatronCorte.getSize(); j++) {
        cout << " Veces el patron " << j+1 << " = " <<
CplexMaestro.getValue(PatronCorte[j]) << "bosque " << endl;
    }
    cout << endl;

    for (int i = 0; i < Demandas.getSize(); i++) {
        cout << " Dual de la demanda " << i+1 << " = " <<
CplexMaestro.getDual(Demandas[i]) << endl;
    }
    cout << endl;

    IloNumArray price(env, PedidoDemMin.getSize());
    for (int i = 0; i < PedidoDemMin.getSize(); ++i) {
        price[i] = -CplexMaestro.getDual(Demandas[i]);
    }

    bool noexistencortes = true;

    for (int mei = 0; mei < BosqueDiaLargo.getSize(); ++mei) {
        ReducedCostos[mei].setLinearCoefs(ProductoInt, price);

        cout << "Nuevo Modelo esclavo " << mei << endl <<
ModelosEsclavos[mei];
        CplexEsclavos[mei].solve();

        cout << " Costo reducido " <<
CplexEsclavos[mei].getValue(ReducedCostos[mei]) << endl;
        if (CplexEsclavos[mei].getValue(ReducedCostos[mei]) <= -1.0e-6) {
            noexistencortes = false;
            for (int i = 0; i < ProductoInt.getSize(); i++) {
                cout << " Obtener pedido " << i+1 << ", " <<
CplexEsclavos[mei].getValue(ProductoInt[i]) << " veces " << endl;
            }

            cout << "Variables Binarias: " <<
CplexEsclavos[mei].getNbinVars() << endl;
            cout << "Variables Enteras: " <<
CplexEsclavos[mei].getNintVars() << endl;
            cout << "Filas - Restricciones: " <<
CplexEsclavos[mei].getNrows() << endl;
            cout << "Columnas - Variables: " <<
CplexEsclavos[mei].getNcols() << endl;

            IloNumArray Costo (env, PedidoDemMax.getSize());
            IloNum temp1 = 0;

            for (int i = 0; i < PedidoDemMax.getSize(); i++){
                Costo [i] = CplexEsclavos[mei].getValue(ProductoInt[i]) *
PedidoCosto[i];
                temp1 += Costo [i];
            }

            IloNumArray newPatt(env, PedidoDemMin.getSize());
            CplexEsclavos[mei].getValues(newPatt, ProductoInt);

            for ( int i = 0; i < NTiempo; i++){
                IloNumColumn newcol(env);

```

```

        newcol += ArbolesUsados(temp1 + BosqueCosto[mei]);

        for ( int j = 0 ; j < NCanchas; j++){
            for ( int k = 0; k < NProductos; k++){
                if (Sj[j][mei]==1){
                    newcol += Balance[j][i][k](newPatt);
                }
            }
        }

        newcol += Demandas(newPatt);
        newcol += Oferta[mei](1);

        if ( PatronCorte.getSize() == NProductos){
            for (int i = 0; i < NTiempo; i++){
                for ( int j = 0; j < NProductos; j++){
                    for ( int k = 0; k < NClientes; k ++){
                        IloExpr temp3 (env);
                        for ( int l = 0; l < NCanchas; l ++){
                            temp3 += Y[i][j][l][k];
                        }
                        ModeloMaestro.add (Dmin [i][k][j] <= temp3
<= Dmax [i][k][j]);
                    }
                }
            }

            PatronCorte.add( IloNumVar( newcol ));
            PatronCorte[ PatronCorte.getSize()-1 ].setName(
makename("Patron", PatronCorte.getSize(), mei+1, i+1).c_str() );
            ArbolPatronCorte.add ( mei );
            TiempoPatronCorte.add ( i );
        }
    }
    if (noexistencortes) break;
}

ModeloMaestro.add(IloConversion(env, PatronCorte, ILOINT));
cout << "modelo entero " << endl << ModeloMaestro << endl;
CplexMaestro.solve();

cout << "* SOLUCION ENTERA USA " << CplexMaestro.getObjValue() << " Dolares"
<< endl;
for (int j = 0; j < PatronCorte.getSize(); j++) {
    cout << " Veces el patron " << j+1 << " = " <<
CplexMaestro.getValue(PatronCorte[j]) << endl;
}

cout << "Variables Binarias: " << CplexMaestro.getNbinVars() << endl;
cout << "Variables Enteras: " << CplexMaestro.getNintVars() << endl;
cout << "Filas - Restricciones: " << CplexMaestro.getNrows() << endl;
cout << "Columnas - Variables: " << CplexMaestro.getNcols() << endl;
cout << CplexMaestro.getVersion()<< endl;
env.out()<<"Se demoro = " << env.getTime()<<endl;

} catch (IloException& ex) {
    std::cerr << "Error: " << ex << endl;
} catch (...) {

```

```
        std::cerr << "Error" << endl;
    }
    env.end();
    return EXIT_SUCCESS;
}
```