

Universidad de Chile  
Facultad de Ciencias Físicas y Matemáticas  
Departamento de Ingeniería Matemática

Restricciones de Volumen Elásticas para  
un Problema de Planificación Forestal con  
Restricciones de Área en Múltiples  
Periodos

Juan Pablo Vielma Centeno

Diciembre 2003

Universidad de Chile  
Facultad de Ciencias Físicas y Matemáticas  
Departamento de Ingeniería Matemática

**Restricciones de Volumen Elásticas para un  
Problema de Planificación Forestal con  
Restricciones de Área en Múltiples Periodos**

JUAN PABLO VIELMA CENTENO

<b>Comisión Examinadora</b>	<b>Nota(nº)</b>	<b>CALIFICACIONES</b> (Letras)	<b>Firma</b>
Profesor Guía Sr. Andrés Weintraub	_____	_____	_____
Profesores Integrantes Sr. Jorge Amaya	_____	_____	_____
Sr. Pablo Rey	_____	_____	_____
Nota Final Examen de Título	_____	_____	_____

**MEMORIA PARA OPTAR AL TÍTULO DE  
INGENIERO CIVIL MATEMÁTICO**

SANTIAGO DE CHILE  
Diciembre 2003

RESUMEN DEL INFORME FINAL  
PARA OPTAR AL TÍTULO DE  
INGENIERO CIVIL MATEMÁTICO  
POR: JUAN PABLO VIELMA  
PROF. GUÍA: SR. ANDRÉS WEINTRAUB

## **Restricciones de Volumen Elásticas para un Problema de Planificación Forestal con Restricciones de Área en Múltiples Periodos**

El objetivo central de este trabajo es desarrollar un método para resolver el problema de planificación forestal con restricciones de área y volumen en múltiples periodos. Este problema se preocupa de la generación de planes de cosecha que maximicen las utilidades manteniendo un balance en los flujos de volumen de madera. Con el fin de minimizar el impacto ambiental se espera además que estos planes de cosecha cumplan con reglamentos que limitan la extensión de zonas contiguas de cosecha. Este problema a sido modelado como un problema de programación lineal entera basado en un mapa forestal dividido en diferentes secciones o celdas. El modelo selecciona grupos de celdas contiguas a ser cosechadas siempre que estas cumplan con los límites de área impuestos. Sólo recientemente se ha desarrollado una formulación que permite resolver rápidamente este problema sin restricciones de volumen. Lamentablemente al incorporar restricciones estrictas sobre los flujos de volumen se pierde gran parte de la efectividad de esta formulación.

En este trabajo se estudió la posibilidad de utilizar restricciones de volumen elásticas para recuperar la efectividad de la formulación. Este tipo de restricciones permite violaciones a las restricciones originales, las que luego son penalizadas en la función objetivo. Los aspectos estudiados incluyeron la actualización dinámica de las penalizaciones, una heurística fijadora de variables y un tipo de ramificación por restricciones.

Se realizaron numerosas pruebas computacionales para determinar la efectividad del método basado en restricciones elásticas, las que mostraron que es posible mantener controladas las violaciones asociadas a las restricciones elásticas. También se comprobó que el método basado en restricciones elásticas permite obtener soluciones mucho más rápido que al utilizar directamente las restricciones de volumen estrictas. Si a esto se suma el hecho de que las restricciones elásticas modelan mejor los requerimientos de los planificadores forestales se concluye que su utilización es preferible a las versiones estrictas.

## AGRADECIMIENTOS

Quisiera agradecer a todas las personas que directa o indirectamente ayudaron a realizar este trabajo de título.

A mi profesor guía Andres Weintraub, quien con su entusiasmo y dedicación hizo que la realización de este proyecto fuera una experiencia extremadamente valiosa y reconfortante.

A Jorge Amaya y Pablo Rey por formar parte de la comisión examinadora y por sus acertados comentarios y sugerencias sobre esta memoria.

A Alan Murray y David Ryan por su activa participación y aportes a este trabajo.

A los departamentos de Ingeniería Matemática e Ingeniería Industrial, a la Facultad de Ciencias Físicas y Matemáticas y a la Universidad de Chile por la formación y oportunidades que me brindaron.

A Marcos Goycoolea por las fructíferas conversaciones respecto a esta memoria. A mis amigos y compañeros Miguel Carrasco, Angela Ganz, Luis Rademacher, Patricio Reyes y Álvaro Valdebenito con quienes compartí el duro pero edificante camino de la Ingeniería Matemática.

A mi familia, mis padres y mi esposa Johana por su apoyo y cariño incondicionales.

# Índice general

<b>Introducción</b>	<b>XIII</b>
<b>1. Descripción del Problema</b>	<b>1</b>
1.1. Formulación del Problema de Empaquetamiento de Clusters .	3
1.1.1. Representación de una Region Forestal . . . . .	3
1.1.2. Problema de Empaquetamiento de Clusters . . . . .	6
<b>2. Restricciones Elásticas</b>	<b>15</b>
2.1. Modelo con Restricciones Elásticas . . . . .	15
2.2. Eligiendo el nivel $\Delta_E$ . . . . .	18
<b>3. Métodos Basados en Modelo Elástico</b>	<b>21</b>
3.1. Penalizaciones Fijas . . . . .	21
3.2. Heurística <i>Integer Allocation</i> . . . . .	22
3.3. Ramificación por Restricciones . . . . .	24
3.3.1. Ramificación <i>Celdas</i> /Tiempo . . . . .	25
3.3.2. Ramificación Tipo Ryan-Foster sobre restricciones de Cliques Proyectados . . . . .	26
3.3.3. Ramificación sobre holgura de restricciones <i>Celda</i> /Tiempo . . . . .	27
3.4. Actualización Dinámica de las Penalizaciones . . . . .	28
3.4.1. Modificando la Función Objetivo . . . . .	30
3.4.2. Agregando Cortes . . . . .	30

3.5. Podando el Árbol de <i>Branch &amp; Bound</i> . . . . .	32
<b>4. Resultados Computacionales</b>	<b>34</b>
4.1. Instancias de Prueba . . . . .	34
4.2. Implementación . . . . .	35
4.3. Calculo de <i>gap</i> 's y objetivo de métodos elásticos . . . . .	36
4.4. <i>CPLEX</i> utilizado como <i>Branch &amp; Cut</i> . . . . .	37
4.4.1. Múltiples Periodos sin Restricciones de Volumen . . . . .	37
4.4.2. Múltiples Periodos con Restricciones de Volumen Exactas . . . . .	38
4.4.3. Múltiples Periodos con Restricciones de Volumen Elásticas y Penalizaciones Fijas . . . . .	39
4.5. <i>CPLEX</i> con llamadas auxiliares . . . . .	43
4.5.1. Ramificación por Restricciones sobre Modelo con Restricciones Estrictas . . . . .	43
4.5.2. Ramificación por Restricciones sobre Modelo con Restricciones Elásticas . . . . .	44
4.5.3. Ramificación por Restricciones con Actualización Dinámica de Penalizaciones sobre Modelo con Restricciones Elásticas . . . . .	46
4.5.4. Heurística y Ramificación por Restricciones sobre Modelo con Restricciones Elásticas . . . . .	47
4.5.5. Heurística y Ramificación por Restricciones con Actualización Dinámica de Penalizaciones sobre Modelo con Restricciones Elásticas . . . . .	48
4.6. <i>Branch &amp; Bound</i> Especializado . . . . .	50
4.6.1. Ramificación por Restricciones sobre Modelo Elástico utilizando $z_R$ . . . . .	51
4.6.2. Ramificación por Restricciones sobre Modelo Elástico utilizando $z_E$ . . . . .	52
4.7. Resumen de Resultados . . . . .	53
<b>5. Conclusiones</b>	<b>57</b>

5.1.	Resultados . . . . .	57
5.1.1.	Restricciones Generadoras de Fracciones . . . . .	58
5.1.2.	Uso de Restricciones Elásticas . . . . .	58
5.1.3.	Actualización Dinámica de las Penalizaciones . . . . .	58
5.1.4.	Ramificación por Restricciones . . . . .	59
5.1.5.	Heurística . . . . .	59
5.1.6.	Podado del Árbol de <i>Branch &amp; Bound</i> . . . . .	60
5.1.7.	Comentarios Finales . . . . .	60
5.2.	Temas por Investigar y Trabajo en Curso . . . . .	60
5.2.1.	Detalle de la Actualización Dinámica de Penalizaciones	61
5.2.2.	Podado del Árbol de <i>Branch &amp; Bound</i> . . . . .	61
5.2.3.	Permitir Violaciones de las Restricciones de Volumen .	61
5.2.4.	Ramificación por Restricciones . . . . .	62
5.2.5.	Estudiar Aplicabilidad del Método . . . . .	62
5.2.6.	Utilización del Método en Problemas Similares . . . . .	63
<b>A. Penalizaciones para Violaciones Nulas</b>		<b>66</b>
<b>B. Descripción de la Heurística</b>		<b>69</b>
B.1.	Primera Pasada de la Heurística . . . . .	69
B.1.1.	Fijado de Variables . . . . .	70
B.1.2.	Corrección de Violaciones de Restricciones de Volumen	71
B.2.	Segunda Pasada de la Heurística . . . . .	71
B.2.1.	Restricciones de Volumen Auxiliares . . . . .	72
<b>C. Problemas con Ramificación de <i>Cliques</i></b>		<b>74</b>
<b>D. Diagrama de Flujo del Método</b>		<b>78</b>
<b>E. Ganancias con Violaciones Reales</b>		<b>80</b>
<b>F. Primer Periodo Dominante</b>		<b>82</b>



# Lista de Formulaciones

1.1.	Problema de Empaquetamiento de Clusters (CPP)	6
1.2.	Problema de Empaquetamiento de Clusters con Restricciones de Cliques Proyectados (CPP-CLQ)	8
1.3.	CPP-CLQ para $\mathcal{T}$ Periodos (CPP-CLQ- $\mathcal{T}$ )	11
1.4.	CPP-CLQ para $\mathcal{T}$ Periodos con Restricciones Estrictas de Volumen No-Decreciente (CPP-CLQ- $\mathcal{T}$ - VND)	13
2.1.	(CPP-CLQ- $\mathcal{T}$ ) con Restricciones de Volumen Elásticas (CPP-CLQ- $\mathcal{T}$ - EIVND)	16
3.1.	(CPP-CLQ- $\mathcal{T}$ - EIVND) con Restricciones de Volumen en Forma de Cortes (CPP-CLQ- $\mathcal{T}$ - EIVND - CUT)	31
A.1.	Problema Genérico Original	66
A.2.	Dual del Problema Genérico Original	67
A.3.	Problema Genérico con Restricciones Elásticas	67
A.4.	Dual del Problema Genérico con Restricciones Elásticas	67

# Lista de Figuras

1.1. Mapa Forestal . . . . .	1
1.2. Grafo Forestal . . . . .	4
1.3. Clusters Compatibles . . . . .	5
1.4. Solución a la Relajación Lineal de CPP-CLQ . . . . .	10
C.1. Dos Clusters que Intersectan <i>Cliques</i> Idénticos . . . . .	75
D.1. Diagrama de Flujo del Método . . . . .	79

# Lista de Cuadros

4.1. Características de las Instancias de Prueba . . . . .	35
4.2. Valores LP de CPP-CLQ- $\mathcal{T}$ . . . . .	37
4.3. Resultados IP de CPP-CLQ- $\mathcal{T}$ . . . . .	38
4.4. Valores LP de CPP-CLQ- $\mathcal{T} - VND$ . . . . .	39
4.5. Resultados IP de CPP-CLQ- $\mathcal{T} - VND$ . . . . .	39
4.6. Violaciones para Formulación Elástica con Penalizaciones Fijas	41
4.7. Soluciones Válidas para Formulación Elástica con Penaliza- ciones Fijas . . . . .	42
4.8. Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T} - VND$ . . . . .	44
4.9. Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T} - EIVND$ . . . . .	45
4.10. Resultados Ramificación por Restricciones y Actualización Dinámica de Panalizaciones sobre CPP-CLQ- $\mathcal{T} - EIVND$ .	47
4.11. Resultados Ramificación por Restricciones y Heurística sobre CPP-CLQ- $\mathcal{T} - EIVND$ . . . . .	48
4.12. Resultados Ramificación por Restricciones, Actualiza- ción Dinámica de Penalizaciones y Heurística sobre CPP-CLQ- $\mathcal{T} - EIVND$ . . . . .	50
4.13. Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T} - EIVND$ utilizando $z_R$ . . . . .	51
4.14. Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T} - EIVND$ utilizando $z_E$ . . . . .	52
4.15. Tiempo primera solución bajo 1% <i>gap</i> para diferentes méto- dos de resolución . . . . .	54
4.16. Mejor <i>gap</i> obtenido luego de 4 horas para diferentes métodos de resolución . . . . .	55
4.17. Primera solución entera factible y válida para instancias de <i>El Dorado</i> . . . . .	55

4.18. Primera solución entera factible y válida para instancias de <i>12by12</i> . . . . .	56
E.1. Ganancias al permitir violaciones reales de las restricciones de volumen . . . . .	80
F.1. Resultados para CPP-CLQ- $\mathcal{T}$ -VND con primer periodo do- minante . . . . .	83
F.2. Resultados para CPP-CLQ- $\mathcal{T}$ -ECVND con primer periodo dominante . . . . .	83

# Introducción

La presente memoria se inscribe en el ámbito de la optimización, específicamente en el área de la optimización combinatorial y la programación entera.

El objetivo central de esta memoria es desarrollar un método para resolver el problema de planificación forestal con restricciones de área y volumen en múltiples periodos.

El problema de planificación forestal con restricciones de área consiste maximizar el beneficio neto al generar un plan de cosecha para un bosque. Este plan de cosecha debe cumplir además con ciertos requerimientos espaciales relacionados con reglamentaciones ecológicas y conservacionistas. Este problema ha sido extensamente estudiado en [GoyMurBarEpsWei03, Goy01] donde se propuso una formulación que permite resolver el problema eficientemente para pequeñas y medianas instancias. Lamentablemente, al agregarle a esta formulación restricciones sobre los volúmenes de madera cosechados el problema se hace mucho más difícil de resolver. En esta memoria estudiaremos la posibilidad de utilizar versiones elásticas de estas *restricciones de volumen* para recuperar las buenas propiedades de la formulación.

En el capítulo 1 describiremos en detalle el problema, presentaremos la formulación propuesta en [GoyMurBarEpsWei03, Goy01] y explicaremos por qué pierde su efectividad al agregar restricciones de volumen. En el capítulo 2 desarrollaremos las restricciones elásticas como una forma alternativa de modelar las restricciones de volumen y explicaremos cómo esto puede facilitar la resolución del problema. En el capítulo 3 analizaremos diversos aspectos necesarios para desarrollar un método que utilice las restricciones elásticas para resolver el problema de planificación forestal. Estos aspectos incluyen, la elección y actualización de las penalizaciones asociadas a las restricciones elásticas, un tipo de ramificación por restricciones adaptado especialmente al problema y una heurística basada en la relajación

lineal del problema. En el capítulo 4 presentaremos resultados numéricos para la formulación original y diversas implementaciones del método basado en restricciones elásticas. Finalmente en el capítulo 5 se desarrollaremos las conclusiones y futuros puntos de estudio.

En el apéndice A mostraremos como elegir penalizaciones que causen violaciones nulas y en el apéndice B se describiremos en detalle la heurística presentada en el capítulo 3. Luego, en el apéndice C discutiremos algunos problemas con una de las ramificaciones presentadas en el capítulo 3, mientras que en el apéndice D presentaremos un diagrama de flujo con todos los aspectos del método desarrollado en el capítulo 3. En los apéndices E y F presentaremos algunos resultados computacionales complementarios y finalmente en el apéndice G definiremos algunos conceptos utilizados a través de esta memoria.

# Capítulo 1

## Descripción del Problema

La planificación forestal consiste en seleccionar que partes de una región forestales serán cosechadas en diferentes periodos. Generalmente se utiliza como base un mapa forestal dividido en unidades o *celdas básicas* como el de la figura 1.1. Estos mapas son obtenidos normalmente utilizando *Sistemas de Información Geográfica* o *GIS*<sup>1</sup> y para cada *celda básica* incluyen información sobre su área, junto con el beneficio y volumen de madera obtenido si la *celda* es cosechada en cierto periodo.

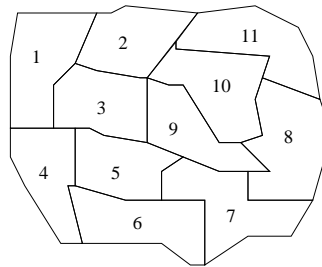


Figura 1.1: Mapa Forestal

El objetivo de la planificación es maximizar los beneficios obtenidos al seleccionar las *celdas básicas* que serán cosechadas en cada periodo. También se espera obtener cierta regularidad a través del tiempo de los flujos de volumen de madera.

---

<sup>1</sup>Del inglés Geographical Information Systems.

El modelamiento matemático ha sido frecuentemente utilizado para resolver este problema, lo que ha permitido la incorporación de diversos reglamentos y requerimientos al proceso de planificación forestal. Estos requerimientos son generalmente incorporados como restricciones en un problema de programación lineal entera y en muchas ocasiones hacen que el modelo sea más difícil de resolver.

Recientemente se han desarrollado diversas reglamentaciones que intentan minimizar el impacto ambiental de la cosecha forestal. Una de las reglamentaciones más conocidas consiste en limitar la extensión de zonas cosechadas contiguamente. La necesidad de utilizar este tipo de restricciones ha sido estudiada extensamente y ya se ha convertido en práctica estándar países como Estados Unidos entre otros (ver [ThoHalLyMil73, Mur99]).

Típicamente, el área contigua máxima cosechable es de 120 acres (40 hectáreas) pero puede variar dependiendo de la región estudiada (ver [BarGilDav98]). La implementación de este tipo de limitaciones sobre el área cosechada han llevado a restricciones típicamente conocidas como *Restricciones de Área Máxima*. Estas restricciones han sido modeladas principalmente de dos formas diferentes.

Si asumimos que las *celdas básicas* son suficientemente grandes (por ejemplo con áreas entre 25 y 49 ha.) tendremos que las *restricciones de área máxima* pueden implementarse de la siguiente manera. Si una celda es cosechada en un periodo  $t$  ninguna de sus vecinos podrá ser cosechado en el mismo periodo, por lo que bastará con prohibir que dos *celdas básicas* adyacentes sean cosechadas en un mismo periodo. Lamentablemente en la mayoría de los casos, la alta resolución de los mapas *GIS* tiene como resultado que las *celdas básicas* sean bastante más pequeñas que el área máxima cosechable, por lo que *si* es posible cosechar grupos de *celdas básicas* adyacentes. Una forma sencilla de tratar este caso es combinar *a priori* estas celdas pequeñas en conjuntos de celdas adyacentes con áreas conjuntas mayores (por ejemplo con áreas conjuntas entre 25 y 49 ha.) y tomar estas *super-celdas* como nuevas *celdas básicas*. Esta primera implementación es llamada *Modelo de Restricción de Unidades (U.R.M)*<sup>2</sup> en [Mur99].

La segunda implementación no combina las *celdas básicas* pequeñas *a priori*, si no que incluye esta decisión dentro del modelo en sí. Esta implementación es llamada *Modelo de Restricción de Área (A.R.M)*<sup>3</sup> en [Mur99].

En [MurWei02] se muestra que utilizar el *A.R.M.* en vez de el *U.R.M.*

---

<sup>2</sup>Del inglés Unit Restriction Model.

<sup>3</sup>Del inglés Area Restriction Model.

puede llevar a soluciones con valores objetivos considerablemente mayores. Desafortunadamente el *A.R.M.* a mostrado ser muy difícil de resolver. A pesar de que numerosas heurísticas han sido propuestas para resolver este modelo [BosBet01, RicGun00, ClaMelMacD00, BarGilDav98, LocMoo93, Hok83], solo recientemente han sido desarrollados métodos exactos capaces de resolver pequeñas y medianas instancias de este problema. En [Goy01] podemos encontrar una comparación detallada de las diferentes formulaciones del *A.R.M.* (junto con las formulaciones del *U.R.M.* y las diferencias entre ellas).

En lo que sigue trabajaremos sobre la formulación fortalecida del *A.R.M.* desarrollada en [GoyMurBarEpsWei03] y [Goy01], conocida como el *Problema de Empaquetamiento de Clusters (C.P.P.)*<sup>4</sup>, debido a que presenta los mejores resultados. Instancias sobre pocos periodos de esta formulación pueden ser resueltas fácilmente por programas comerciales de programación entera (como por ejemplo *CPLEX* [Cpx02]).

A continuación describiremos el *Problema de Empaquetamiento de Clusters* para uno y multiples periodos junto con las restricciones de volumen que destruyen las buenas propiedades de la formulación.

## 1.1. Formulación del Problema de Empaquetamiento de Clusters

El *Problema de Empaquetamiento de Clusters* modela el problema de planificación de cosecha forestal utilizando la siguiente representación de una región forestal.

### 1.1.1. Representación de una Region Forestal

Dado un mapa de una región forestal dividido en *celdas básicas*, como el de la figura 1.1, podemos considerar las siguientes relaciones de adyacencia entre pares de *celdas básicas*.

**Definición 1.1.1 (Adyacencia Fuerte).** Dos *celdas básicas* se dirán *fuertemente adyacentes* si comparten al menos parte de una arista del mapa forestal. Por ejemplo, en la figura 1.1 las *celdas básicas* 1 y 2 son *fuertemente adyacentes*.

---

<sup>4</sup>Del inglés Cluster Packing Problem.

**Definición 1.1.2 (Adyacencia Débil).** Dos *celdas básicas* se dirán *débilmente adyacentes* si comparten un vértice del mapa forestal o si son *fuertemente adyacentes*. Por ejemplo, en la figura 1.1 las *celdas básicas* 2 y 9 son *débilmente adyacentes* pero no *fuertemente adyacentes*.

Con estas definiciones representaremos una region forestal por un grafo  $G(V, E)$  definido de la siguiente manera. Cada *celda básica* del mapa forestal tendrá asociado un nodo  $u \in V$  y existirá una arista entre cada par de nodos asociados a *celdas básicas débilmente adyacentes* en dicho mapa.

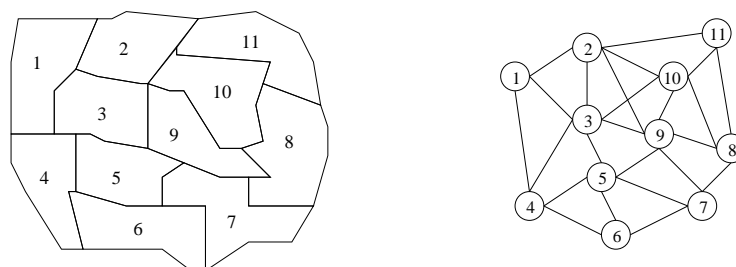


Figura 1.2: Grafo Forestal

Para cada  $u \in V$  denotaremos la información sobre el area, beneficio y volumen de la *celda básica* asociada por  $a_u, c_u$  y  $v_u$  respectivamente. Los valores  $a_u$  y  $v_u$  serán valores racionales positivos, mientras que los valores  $c_u$  podrán ser racionales positivos o negativos<sup>5</sup>. En lo que sigue nos referiremos indistintamente a las *celdas básicas* y al nodo asociado a ellas.

### Clusters Factibles y Clusters Incompatibles

Como ya hemos mencionado, en los modelos *A.R.M* es posible cosechar algunos conjuntos de *celdas* adyacentes sin violar las restricciones de area máxima. Llamaremos *clusters factibles* a todos estos conjuntos de *celdas* que además cumplan con ser *fuertemente adyacentes*. Más rigurosamente:

**Definición 1.1.3 (Clusters Factibles).** Dado  $E' \subset E$  un subconjunto de las aristas del grafo forestal definido como

$$E' = \{\{u, v\} / u \text{ y } v \text{ son fuertemente adyacentes en el mapa forestal}\}$$

<sup>5</sup>Debido a que parte de la planificación de cosecha forestal es asegurar flujos de volumen de madera regulares puede tener sentido cosechar *celdas básicas* que causen pérdidas en la función objetivo.

llamaremos *cluster factible* a los conjuntos de nodos  $S \subset V$  que cumplan:

1.  $S$  induce un subgrafo conexo en  $G(V, E')$
2.  $\sum_{u \in S} a_u \leq A_{max}$

donde  $A_{max}$  es el area contigua máxima cosechable

Denotaremos por  $\mathcal{C}$  al conjunto de todos los *clusters factibles*. Cada *cluster factible*  $S \in \mathcal{C}$  tendrá asociada información sobre area, beneficio y volumen ( $c_S, a_S$  y  $v_S$ ) definida como:

- $a_S := \sum_{u \in S} a_u$
- $c_S := \sum_{u \in S} c_u$
- $v_S := \sum_{u \in S} v_u$

**Definición 1.1.4 (Clusters Incompatibles).** Diremos que dos *clusters factibles* son incompatibles si comparten un nodo o si son *débilmente adyacentes*. Es decir,  $S, S' \in \mathcal{C}$  son incompatibles ssi:

$$S \cap S' \neq \emptyset \quad \vee \quad \exists u \in S, v \in S' \text{ tal que } \{u, v\} \in E$$

Supongamos que  $a_5 + a_6 \leq A_{max}$ ,  $a_1 \leq A_{max}$  y  $a_8 + a_{10} + a_{11} \leq A_{max}$  entonces la siguiente figura muestra tres *clusters factibles* ( $\{1\}, \{8, 10, 11\}$  y  $\{5, 6\}$ ). Además los tres son compatibles entre si.

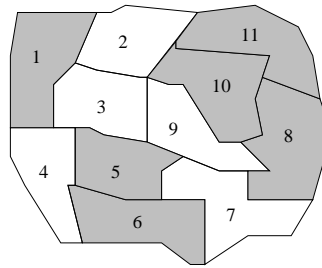


Figura 1.3: Clusters Compatibles

### 1.1.2. Problema de Empaquetamiento de Clusters

A continuación describiremos el *Problema de Empaquetamiento de Clusters*. Por simplicidad primero analizaremos el modelo para un solo periodo, luego lo extenderemos a multiples periodos y finalmente incorporaremos las restricciones de volumen.

#### Un Periodo

Como todos los *clusters factibles* cumplen con las *restricciones de área máxima* y los *clusters compatibles* no son adyacentes tendremos que conjuntos de *clusters factibles compatibles* también cumplirán con las *restricciones de área máxima*. Así, para cumplir con las *restricciones de área máxima* bastara con prohibir que dos *clusters incompatibles* sean cosechados al mismo tiempo. De esta manera el *Problema de Empaquetamiento de Clusters* se formula de la siguiente manera:

---

**Formulación 1.1** Problema de Empaquetamiento de Clusters (CPP)

---

$$\text{máx} \sum_{S \in \mathcal{C}} c_S x_S \quad (1.1a)$$

s.a.

$$x_S + x_{S'} \leq 1 \quad \forall S, S' \in \mathcal{C} \text{ incompatibles} \quad (1.1b)$$

$$x_S \in \{0, 1\} \quad \forall S \in \mathcal{C} \quad (1.1c)$$

---

donde

- $x_S = \begin{cases} 1 & \text{si el cluster } S \text{ es cosechado} \\ 0 & \text{si no.} \end{cases}$
- $c_S$  es el beneficio obtenido al cosechar el cluster  $S$ .
- $\mathcal{C}$  es el conjunto de todos los clusters factibles.

El objetivo es maximizar el beneficio neto al seleccionar que clusters serán cosechados.

Las restricciones (1.1b) prohíben cosechar dos clusters incompatibles al mismo tiempo y las restricciones (1.1c) definen las variables  $x_S$  como variables binarias.

El *Problema de Empaquetamiento de Clusters* es un problema tipo *Empaquetamiento de Conjuntos*<sup>6</sup> que puede verse como un problema de *Conjunto Independiente o Estable Máximo* sobre un grafo de *clusters*  $G_C(V_C, E_C)$  definido de la siguiente manera.

**Definición 1.1.5 (Grafo de *clusters*  $G_C(V_C, E_C)$ ).** Cada *cluster factible* tendrá asociado un nodo  $S \in V_C$  y existirá una arista entre cada par de *clusters incompatibles*.

Se conocen muchas desigualdades válidas para el poliedro asociado a los problemas de *Conjunto Independiente o Estable Máximo*. En particular, las restricciones basadas en *cliques maximales* del grafo  $G_C(V_C, E_C)$  son muy útiles pues definen facetas de este poliedro [Pad73]. Lamentablemente el gran número de nodos<sup>7</sup> y aristas<sup>8</sup> de  $G_C(V_C, E_C)$  hace que encontrar todos sus *cliques* sea imposible en la práctica. En [GoyMurBarEpsWei03] se propone una estrategia intermedia en la que se fortalece la formulación del *Problema de Empaquetamiento de Clusters* mediante restricciones conocidas como *Restricciones de Cliques Proyectados*. Estas restricciones están basadas en *cliques maximales* en el grafo forestal  $G(V, E)$  (en vez del grafo  $G_C(V_C, E_C)$ ) y básicamente prohíben cosechar simultáneamente *clusters factibles* que intersecten a un mismo *clique maximal* en  $G(V, E)$ . En [GoyMurBarEpsWei03] se muestra que este tipo de restricciones dominan a las restricciones de *clusters incompatibles* y que generalmente presenten propiedades de integralidad mucho mejores. Además el grafo  $G(V, E)$  si presenta un número computacionalmente manejable de *cliques maximales* en contraste con el grafo  $G_C(V_C, E_C)$  (ver [GoyMurBarEpsWei03], [Goy01] y la descripción de las instancias de prueba en la sección 4.1). El *Problema de Empaquetamiento de Clusters* con estas *restricciones de cliques proyectados* queda entonces definido como:

---

<sup>6</sup>Set Packing.

<sup>7</sup>Ver número de clusters en cuadro 4.1 en capítulo 4.

<sup>8</sup>En particular existe una arista para cada par de clusters que contengan una *celda básica* específica.

---

**Formulación 1.2** Problema de Empaquetamiento de Clusters con Restricciones de Cliques Projectados (CPP-CLQ)

---

$$\text{máx } \sum_{S \in \mathcal{C}} c_S x_S \quad (1.2a)$$

s.a.

$$\sum_{S \in \mathcal{C}(K)} x_S \leq 1 \quad \forall K \in \mathcal{K} \quad (1.2b)$$

$$x_S \in \{0, 1\} \quad \forall S \in \mathcal{C} \quad (1.2c)$$


---

donde

- $x_S = \begin{cases} 1 & \text{si el cluster } S \text{ es cosechado} \\ 0 & \text{si no.} \end{cases}$
- $c_S$  es el beneficio obtenido al cosechar el cluster  $S$ .
- $\mathcal{C}$  es el conjunto de todos los clusters factibles.
- $\mathcal{K}$  es el conjunto de todos los *cliques maximales* en el grafo forestal  $G(V, E)$ .
- $\mathcal{C}(K)$  es el conjunto de todos los clusters que intersectan al *clique maximal*  $K$ .

El objetivo es maximizar el beneficio neto al seleccionar que clusters serán cosechados.

Las restricciones (1.2b) prohíben cosechar al mismo tiempo dos clusters que intersecten a un mismo *clique*  $K$  y las restricciones (1.2c) definen las variables  $x_S$  como variables binarias.

En [GoyMurBarEpsWei03] se muestra que las *restricciones de cliques proyectados* no necesariamente definen facetas para el problema y se presenta un procedimiento simple para levantar<sup>9</sup> estas restricciones. A pesar de esto, en la práctica, la formulación con *cliques proyectados* presenta muy buenas propiedades de integralidad balanceadas con un número razonable de restricciones, lo que hace muy fácil resolverlo. No es claro aún si esto se debe

---

<sup>9</sup>Hacer *lifting* [NemWol98].

a propiedades polihedrales globales del problema<sup>10</sup>, pero en prácticamente todas las instancias estudiadas la relajación lineal del problema presenta muy pocas variables fraccionarias. Estas fracciones pueden generalmente ser eliminadas con sólo fijar una variable a integralidad o agregando un par de cortes de uso general<sup>11</sup>, lo que normalmente permite resolver el problema en el nodo raíz del árbol de *Branch & Bound*. Un hecho interesante es que en muchas ocasiones todas las fracciones de la relajación lineal del problema se ubican geográficamente contiguas. La siguiente figura muestra una de estas soluciones para la instancia de El Dorado<sup>12</sup>. Podemos ver que las celdas cosechadas parcialmente debido a clusters fraccionarios se encuentran sólo en una pequeña area.

---

<sup>10</sup>i.e. muy pocos puntos extremos fraccionarios en el polítopo asociado a la relajación lineal del problema.

<sup>11</sup>Como por ejemplo los agregados automáticamente por implementaciones comerciales de algoritmos tipo *Branch & Cut*.

<sup>12</sup>Esta instancia será descrita en detalle en el capítulo 4.

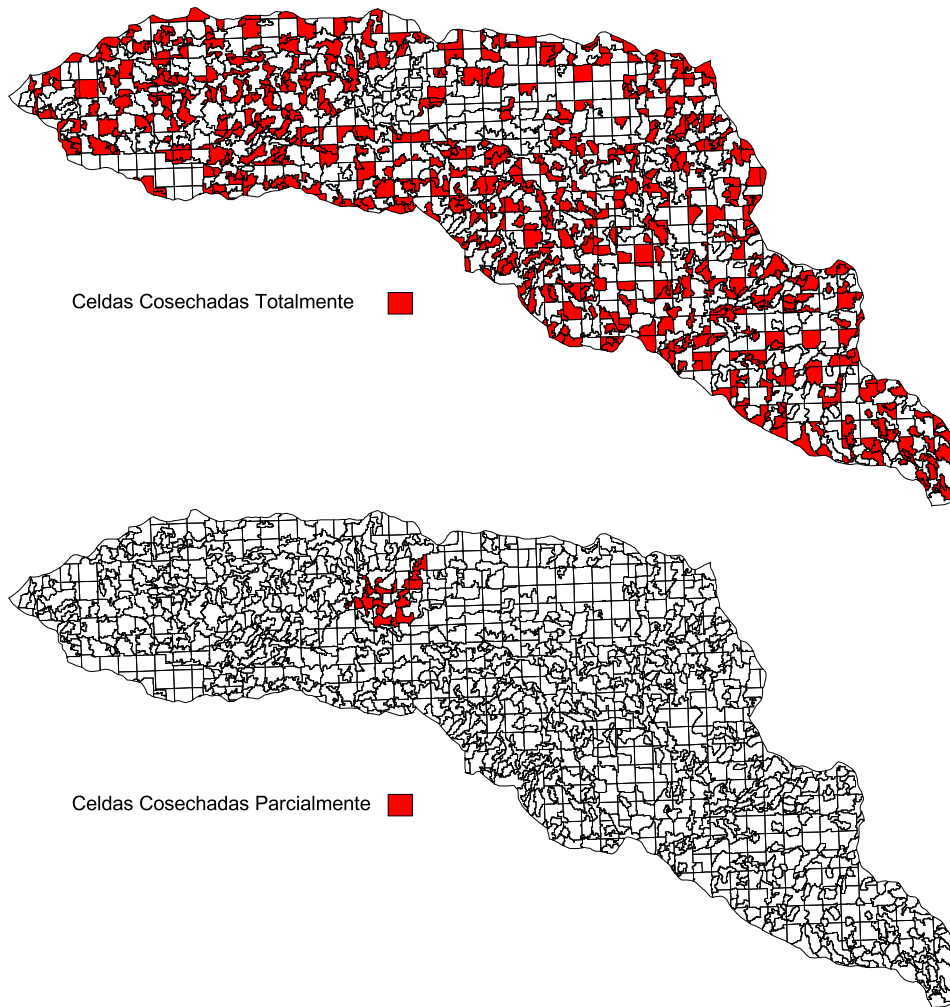


Figura 1.4: Celdas Cosechadas Total y Parcialmente en una Solución a la Relajación Lineal de CPP-CLQ para la Instancia de *El Dorado*

### Multiples Periodos sin Green-Up

El problema de empaquetamiento de clusters puede fácilmente extenderse a multiples periodos.

Supondremos que la velocidad de crecimiento del bosque será suficientemente rápida como para que una celda cosechada en un periodo  $t$  vuelva

a ser considerada terreno boscoso en el periodo  $t + 1$ . Por otro lado, consideraremos que la velocidad de crecimiento del bosque será suficientemente lenta como para que una celda cosechada no llegue nuevamente a una edad cosechable durante el horizonte de planificación. De esta manera, solo es necesario una copia del *problema de empaquetamiento de clusters* para cada periodo más una restricción que prohíba cosechar cada celda más de una vez.

Debido a que nuestro objetivo es maximizar el *valor presente neto* (*V.P.N.*) de los beneficios obtenidos, los valores  $c_{S,t}$  considerados en el modelo serán *valores presentes* en el periodo 0. La *tasa de descuento* utilizada dependerá de la instancia y será especificada en la sección 4.1.

Así, el *problema de empaquetamiento de clusters* con restricciones de cliques proyectados en múltiples periodos se formula como:

---

**Formulación 1.3** CPP-CLQ para  $T$  Periodos (CPP-CLQ- $T$ )

---

$$\text{máx} \sum_{t=1}^T \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t} \quad (1.3a)$$

s.a.

$$\sum_{S \in \mathcal{C}(K)} x_{S,t} \leq 1 \quad \forall K \in \mathcal{K}, \forall t \quad (1.3b)$$

$$\sum_{t=1}^T \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} \leq 1 \quad \forall u \in V \quad (1.3c)$$

$$x_{S,t} \in \{0, 1\} \quad \forall S \in \mathcal{C}, \forall t \quad (1.3d)$$


---

donde

- $x_{S,t} = \begin{cases} 1 & \text{si el cluster } S \text{ es cosechado en el periodo } t \\ 0 & \text{si no.} \end{cases}$
- $c_{S,t}$  es el *valor presente* del beneficio obtenido al cosechar el cluster  $S$  en el periodo  $t$ .
- $\mathcal{C}$  es el conjunto de todos los clusters factibles.

- $\mathcal{K}$  es el conjunto de todos los *cliques maximales* en el grafo forestal  $G(V, E)$ .
- $\mathcal{C}(K)$  es el conjunto de todos los clusters que intersectan al *clique maximal*  $K$ .
- $\mathcal{C}(\{u\})$  es el conjunto de clusters que contienen a la *celda/nodo*  $u$ .

El objetivo es maximizar el *valor presente neto* del beneficio seleccionando que clusters serán cosechados en cada periodo  $t$ .

Para cada periodo  $t$  las restricciones (1.3b) prohíben cosechar al mismo tiempo dos clusters que intersecten a un mismo *clique*  $K$ . El hecho de que cada celda sólo pueda ser cosechada una vez es impuesto por las restricciones (1.3c). Finalmente las restricciones (1.3d) definen las variables  $x_{S,t}$  como variables binarias.

Esta formulación preserva la mayor parte de la estructura de la formulación original. De hecho la única nueva fuente de soluciones fraccionarias viene de la posibilidad de cosechar una celda parcialmente en varios periodos de tiempo. Esto hace el problema más complicado pero en la practica esta formulación continua siendo relativamente fácil de resolver.

### Multi-periodo sin Green-Up con Restricción de Volumen

Es posible complementar el modelo en múltiples periodos con diversos tipos de restricciones sobre el volumen cosechado. Los planificadores forestales normalmente esperan cierto tipo de regularidad en los flujos de volumen de madera, como por ejemplo que el volumen cosechado en un periodo este dentro de un porcentaje fijo del periodo anterior. Esto puede obtenerse agregando una restricción del tipo:

$$(1 - \Delta) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} \leq \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} \leq (1 + \Delta) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1}$$

$\forall t \geq 2$ . Donde  $v_{S,t}$  es el volumen de madera cosechado si el cluster  $S$  es seleccionado para ser cosechado en el periodo  $t$  y  $\Delta \in (0, 1)$  es un valor fijo predeterminado (usualmente entre 0,1 y 0,15).

Al agregar este tipo de restricción de volumen obtenemos la siguiente formulación:

---

**Formulación 1.4** CPP-CLQ para  $\mathcal{T}$  Periodos con Restricciones Estrictas de Volumen No-Decreciente (CPP-CLQ- $\mathcal{T}$  – VND)

---

$$\text{máx} \quad \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t} \quad (1.4a)$$

s.a.

$$\sum_{S \in \mathcal{C}(K)} x_{S,t} \leq 1 \quad \forall K \in \mathcal{K}, \forall t \quad (1.4b)$$

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} \leq 1 \quad \forall u \in V \quad (1.4c)$$

$$(1 - \Delta) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} - \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} \leq 0 \quad \forall t \geq 2 \quad (1.4d)$$

$$\sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} - (1 + \Delta) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} \leq 0 \quad \forall t \geq 2 \quad (1.4e)$$

$$x_{S,t} \in \{0, 1\} \quad \forall S \in \mathcal{C}, \forall t \quad (1.4f)$$


---

Donde:

- $x_{S,t} = \begin{cases} 1 & \text{si el cluster } S \text{ es cosechado en el periodo } t \\ 0 & \text{si no.} \end{cases}$
- $c_{S,t}$  es el *valor presente* del beneficio obtenido al cosechar el cluster  $S$  en el periodo  $t$
- $\mathcal{C}$  es el conjunto de todos los clusters factibles
- $\mathcal{K}$  es el conjunto de todos los *cliques maximales* en el grafo forestal  $G(V, E)$
- $\mathcal{C}(K)$  es el conjunto de todos los clusters que intersectan al *clique maximal*  $K$
- $\mathcal{C}(\{u\})$  es el conjunto de clusters que contienen a la *celda/nodo*  $u$
- $v_{S,t}$  es el volumen de madera cosechado si el cluster  $S$  es seleccionado para ser cosechado en el periodo  $t$

El objetivo es maximizar el *valor presente neto* del beneficio seleccionando que clusters serán cosechados en cada periodo  $t$ .

Para cada periodo  $t$  las restricciones (1.4b) prohíben cosechar al mismo tiempo dos clusters que intersecten a un mismo *clique*  $K$ . El hecho de que cada celda sólo pueda ser cosechada una vez es impuesto por las restricciones (1.4c). Las restricciones (1.4d) aseguran que el volumen cosechado en el periodo  $t$  sea mayor que  $(1 - \Delta)$  del volumen cosechado en el periodo  $t - 1$  y las restricciones (1.4e) que sea menor que  $(1 + \Delta)$  del volumen cosechado en el periodo  $t - 1$ . Finalmente las restricciones (1.4f) definen las variables  $x_{S,t}$  como variables binarias.

Lamentablemente estas restricciones de volumen tienen un efecto adverso sobre la formulación. Usualmente, al menos uno de los grupos de restricciones de volumen ((1.4d) o (1.4e)) es activo. Es decir, las soluciones a la relajación lineal cumplen exactamente con el grupo de restricciones de volumen correspondiente. Desafortunadamente los valores  $v_{S,t}$  son generalmente fraccionarios, por lo que es muy poco probable que una solución entera cumpla con las restricciones de volumen de manera exacta. El efecto generador de fracciones de restricciones similares a sido estudiado en [EhrRya03] donde se discute como este tipo de restricciones puede actuar como un corte generador de puntos extremos fraccionarios sobre el polítopo de una formulación. Estos cortes afectan considerablemente el desempeño de procedimientos como *Branch & Bound* pues hacen muy difícil el obtener soluciones enteras a partir de soluciones a la relajación lineal del problema.

## Capítulo 2

# Restricciones Elásticas

Las restricciones de volumen son realmente una guía y no un requerimiento estricto, por lo que violarlas levemente puede ser aceptable desde el punto de vista de la planificación forestal. Esto sugiere la utilización de versiones elásticas de estas restricciones, es decir permitir pequeñas violaciones penalizándolas en la función objetivo. Esta técnica ha sido utilizada con éxito en problemas de asignación de tripulación aérea [EhrRya03] y elimina gran parte del efecto generador de fracciones de las restricciones estrictas.

A menos que las penalizaciones sean demasiado grandes, las restricciones elásticas dejarán de ser activas y se recuperaran muchas de las propiedades de integralidad del modelo sin restricciones de volumen. Esto facilitará encontrar soluciones enteras a partir de soluciones a la relajación lineal, pero lograr que estas soluciones no violen demasiado las restricciones de volumen originales puede no ser tan sencillo.

La elección de las penalizaciones deberá realizarse cuidadosamente para mantener controladas la violaciones sin recuperar el efecto generador de fracciones. Si las penalizaciones son demasiado grandes las restricciones se comportarán como restricciones estrictas y se recuperarán las malas propiedades de estas últimas. Por otro lado, si las penalizaciones son demasiado pequeñas las violaciones pueden llegar a niveles inaceptables.

### 2.1. Modelo con Restricciones Elásticas

Para implementar el modelo con restricciones de volumen elásticas basta agregar variables continuas  $\underline{w}_t, \overline{w}_t$  que permitan violar las restricciones de

volumen

$$(1 - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} - \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} \leq \underline{w}_t$$

$$\sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} - (1 + \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} \leq \overline{w}_t$$

y luego agregar las siguientes penalizaciones a la función objetivo.

$$- \sum_{t=2}^{\mathcal{T}} \underline{p}_t \underline{w}_t - \sum_{t=2}^{\mathcal{T}} \overline{p}_t \overline{w}_t$$

Al reemplazar las restricciones de volumen estrictas por estas versiones elásticas obtenemos el siguiente modelo.

---

**Formulación 2.1** (CPP-CLQ- $\mathcal{T}$ ) con Restricciones de Volumen Elásticas (CPP-CLQ- $\mathcal{T}$  - EIVND)

---

$$\text{máx} \quad \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t} - \sum_{t=2}^{\mathcal{T}} \underline{p}_t \underline{w}_t - \sum_{t=2}^{\mathcal{T}} \overline{p}_t \overline{w}_t \quad (2.1a)$$

s.a.

$$\sum_{S \in \mathcal{C}(K)} x_{S,t} \leq 1 \quad \forall K \in \mathcal{K}, \forall t \quad (2.1b)$$

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} \leq 1 \quad \forall u \in V \quad (2.1c)$$

$$(1 - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} - \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} \leq \underline{w}_t \quad \forall t \geq 2 \quad (2.1d)$$

$$\sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} - (1 + \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} \leq \overline{w}_t \quad \forall t \geq 2 \quad (2.1e)$$

$$x_{S,t} \in \{0, 1\} \quad \forall S \in \mathcal{C}, \forall t \quad (2.1f)$$

$$\underline{w}_t, \overline{w}_t \geq 0 \quad \forall t \quad (2.1g)$$


---

donde

- $x_{S,t} = \begin{cases} 1 & \text{si el cluster } S \text{ es cosechado en el periodo } t \\ 0 & \text{si no.} \end{cases}$
- $c_{S,t}$  es el *valor presente* del beneficio obtenido al cosechar el cluster  $S$  en el periodo  $t$ .
- $\mathcal{C}$  es el conjunto de todos los clusters factibles.
- $\mathcal{K}$  es el conjunto de todos los *cliques maximales* en el grafo forestal  $G(V, E)$ .
- $\mathcal{C}(K)$  es el conjunto de indices de todos los clusters que intersectan al *clique maximal*  $K$ .
- $\mathcal{C}(\{u\})$  es el conjunto de clusters que contienen a la celda/nodo  $u$ .
- $v_{S,t}$  es el volumen de madera cosechado si el cluster  $S$  es seleccionado para ser cosechado en el periodo  $t$ .
- $\underline{w}_t, \overline{w}_t$  son las violaciones de las restricciones de volumen.
- $\underline{p}_t, \overline{p}_t$  son las penalizaciones por violar las restricciones de volumen inferiores y superiores respectivamente.

El objetivo es maximizar el *valor presente neto* del beneficio menos las penalizaciones asociadas a la violación de restricciones de volumen seleccionando que clusters serán cosechados en cada periodo  $t$ . El primer termino en la función objetivo (2.1a) es el beneficio real obtenido al cosechar los clusters seleccionados. El segundo termino es la penalización asociada a la violación de las restricciones que exigen que el volumen cosechado en el periodo  $t$  sea mayor que  $(1 - \Delta_E)$  del volumen cosechado en el periodo  $t - 1$ . El tercer termino es la penalización asociada a la violación de las restricciones que exige que el volumen cosechado en el periodo  $t$  sea menor que  $(1 + \Delta_E)$  del volumen cosechado en el periodo  $t - 1$ . Debemos notar que para controlar las violaciones de las restricciones de volumen lo importante no son los valores  $\underline{p}_t, \overline{p}_t$  en si, sino la relación entre las magnitudes:

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t}$$

y

$$-\sum_{t=2}^{\mathcal{T}} \underline{p}_t \underline{w}_t - \sum_{t=2}^{\mathcal{T}} \bar{p}_t \bar{w}_t$$

por este motivo, la elección de los valores  $\underline{p}_t, \bar{p}_t$  no sólo dependerá de la instancia estudiada (por ejemplo de los valores  $c_{S,t}$ ) sino también de los valores relativos de las variables  $x_{S,t}$  y  $\underline{w}_t, \bar{w}_t$ . Como veremos en las secciones 3.1 y 3.4 esto puede llevar a la necesidad de modificar los valores  $\underline{p}_t, \bar{p}_t$  durante la optimización para poder mantener controladas las violaciones.

Para cada periodo  $t$  las restricciones (2.1b) prohíben cosechar al mismo tiempo dos clusters que intersecten a un mismo *clique*  $K$ . El hecho de que cada celda sólo pueda ser cosechada una vez es impuesto por las restricciones (2.1c). Las restricciones (2.1d) y (2.1e) definen las violaciones de las restricciones que exigen que el volumen cosechado en el periodo  $t$  sea mayor que el  $(1 - \Delta_E)$  del volumen cosechado en el periodo  $t - 1$  y menor que el  $(1 + \Delta_E)$  del volumen cosechado en el periodo  $t - 1$  respectivamente. Notemos que hemos elegido un nivel  $\Delta_E$  diferente al nivel  $\Delta$  usado en el modelo con restricciones de volumen estrictas (formulación 1.4), discutiremos el porque de esto en la siguiente sección. Finalmente las restricciones (2.1g) definen las variables  $x_{S,t}$  como variables binarias.

## 2.2. Eligiendo el nivel $\Delta_E$

Debemos notar que las restricciones elásticas evitarán el efecto generador de fracciones de las restricciones estrictas sólo cuando permitan que las violaciones sean distintas de cero. Si elegimos penalizaciones que causen que las violaciones sean nulas, las soluciones a la relajación lineal caerán nuevamente sobre el corte generador de fracciones de las restricciones estrictas.<sup>1</sup> Por este motivo para utilizar el modelo con restricciones elásticas debemos aceptar *a priori* la validez de soluciones con violaciones pequeñas pero no nulas.

En el caso de que realmente necesitemos cumplir con las restricciones de volumen las restricciones elásticas no podrán ser usadas directamente.

---

<sup>1</sup>En realidad tener violaciones nulas sólo es un problema para las restricciones activas (i.e. con holgura nula). Si la restricción elástica tiene la forma  $\mathbf{v}\mathbf{x} - w \leq 0$  donde  $w$  es la violación, podemos reescribirla como  $\mathbf{v}\mathbf{x} - w + h = 0$  donde  $h$  es la holgura de la restricción. Como lo que queremos es alejarnos del corte  $\mathbf{v}\mathbf{x} = 0$  lo que realmente necesitamos es  $w > 0 \vee h > 0$ .

Afortunadamente una pequeña modificación del modelo permitirá recuperar su efectividad con un posible pero moderado sacrificio del valor objetivo.

Notemos que en la formulación con restricciones elásticas es posible elegir un nivel  $\Delta_E$  en las restricciones de volumen elásticas  $\{(2.1d),(2.1e)\}$  diferente del nivel  $\Delta$  en las restricciones de volumen estrictas originales  $\{(1.4d),(1.4e)\}$ . Si nuestro objetivo es cumplir con las restricciones de volumen estrictas de nivel  $\Delta$  podríamos utilizar restricciones elásticas de nivel  $\Delta_E < \Delta$ . De esta manera el cumplir con las restricciones estrictas no se traducirá a tener violaciones nulas sino que violaciones controladas como queríamos. Por ejemplo, si nuestro objetivo fuera resolver el problema con restricciones de volumen estrictas con nivel  $\Delta = 0,15$  podríamos utilizar el modelo con restricciones elásticas con  $\Delta_E = 0,14$ . Así sólo nos bastaría mantener las violaciones de las restricciones elásticas bajo 1% para cumplir con las restricciones originales<sup>2</sup>. Claramente, las penalizaciones asociadas a violaciones de las restricciones elásticas entre  $\Delta_E$  y  $\Delta$  pueden causar que buenas soluciones con volúmenes en ese rango no sean consideradas, con una consiguiente pérdida en el valor objetivo máximo. De hecho, las soluciones óptimas a la relajación lineal del problema cumplen exactamente con un grupo de restricciones estrictas (ya sea las cotas superiores o inferiores) y el efecto de *elastización* interna causado por utilizar  $\Delta_E < \Delta$  es precisamente alejarse de estas restricciones. Por este motivo, es muy probable cierta pérdida en el valor objetivo de la relajación lineal. Por suerte, el gran número de soluciones alternativas a la relajación lineal con valores objetivos similares ayudará a que esta pérdida en el valor objetivo óptimo sea relativamente pequeña si el rango  $[\Delta_E, \Delta]$  es pequeño.

Otro problema asociado con utilizar  $\Delta_E < \Delta$  esta asociado al podar nodos en el árbol de *Branch & Bound*, lo que analizaremos en detalle en la sección 3.5.

El utilizar  $\Delta_E < \Delta$  puede facilitar el cumplir con las restricciones de nivel  $\Delta$ , pues esto le dará cierta holgura a las violaciones. A pesar de esto, para asegurar el cumplimiento de las restricciones de nivel  $\Delta$  será necesario elegir penalizaciones suficientemente grandes que mantengan controladas las violaciones.

---

<sup>2</sup>La condición necesaria para cumplir con restricciones de volumen estrictas de  $\Delta = \Delta_E + 0,01$  al utilizar restricciones elásticas de  $\Delta_E$  no es *exactamente* tener violaciones bajo 1%. En lo que sigue de esta memoria, cuando mencionemos *mantener las violaciones bajo 1%* nos estaremos refiriendo a la condición sobre las violaciones que aseguran el cumplimiento de las restricciones estrictas de nivel  $\Delta = \Delta_E + 0,01$ .

En general un objetivo de utilizar restricciones elásticas es reconocer el hecho que las restricciones estrictas no modelan adecuadamente la realidad. A pesar de esto, en los resultados numéricos del capítulo 4 tomaremos como objetivo el cumplir con ciertas restricciones estrictas, debido a que facilita la comparación de valores objetivos entre soluciones al modelo estricto y elástico.

## Capítulo 3

# Métodos Basados en Modelo Elástico

A continuación estudiaremos diversos aspectos útiles para desarrollar un método que utilice los modelos con restricciones elásticas para resolver el modelo original con restricciones estrictas.

### 3.1. Penalizaciones Fijas

La primera alternativa es utilizar el modelo elástico (formulación 2.1) directamente y resolverlo con algún método exacto como *Branch & Bound* ( $B\&B$ ) o *Branch & Cut* ( $B\&C$ ). De esta manera las penalizaciones estarían fijas durante todo el proceso. Sólo sería necesario encontrar penalizaciones iniciales que mantuvieran controladas las violaciones.

En el apéndice A veremos que obtener penalizaciones que causen que las violaciones de volumen sean nulas es relativamente sencillo. Desafortunadamente encontrar penalizaciones que mantengan las violaciones controladas pero no nulas puede ser más complicado.

Por otro lado, las violaciones de las restricciones de volumen tienden a aumentar a medida que se desciende en el árbol de  $B\&B$ . Esto podría tener un efecto benéfico o perjudicial sobre métodos basados en restricciones elásticas.

Si comenzamos con penalizaciones que sean justo las necesarias para obtener violaciones nulas, la relajación lineal del modelo elástico en el nodo

raíz del árbol de  $B\mathcal{E}B$  tendrá una solución con valores prácticamente tan fraccionarios como una solución a la relajación lineal del modelo estricto. En principio esto puede parecer muy perjudicial para la generación de soluciones enteras, pero mientras este efecto no sea sostenido a través del árbol de  $B\mathcal{E}B$  el efecto sobre el método será mínimo. A medida que descendemos en el árbol de  $B\mathcal{E}B$  las violaciones irán aumentando y las soluciones óptimas a las relajaciones lineales se alejarán del corte generador de fracciones causado por las restricciones de volumen estrictas correspondientes<sup>1</sup>. Esto evitará el efecto generador de fracciones asociado a violaciones nulas donde es más perjudicial, en los nodos más profundos, cercanos a las hojas enteras del árbol de  $B\mathcal{E}B$ .

Lamentablemente, este aumento de las violaciones puede también ser perjudicial. Al avanzar en el árbol de  $B\mathcal{E}B$  las violaciones pueden aumentar sin control, llegando a niveles inaceptables. En este caso, el método podría generar soluciones que no puedan ser consideradas válidas. Por este motivo, sería interesante poder aumentar levemente las penalizaciones cada vez que las violaciones aumenten demasiado.

En principio puede ser posible encontrar penalizaciones fijas que mantengan las violaciones controladas pero no nulas (al menos no nulas en nodos suficientemente profundos), pero como veremos en los resultados computacionales de la sección 4.4.3, esto sólo sucederá en algunas instancias.

### 3.2. Heurística *Integer Allocation*

La heurística utilizada por *CPLEX* generalmente entregan soluciones con grandes violaciones de las restricciones de volumen originales. Probablemente se deba a que esta heurística no considera la necesidad de mantener las violaciones controladas y su principal meta es generar soluciones enteras aún cuando estas tengan valores objetivos muy pequeños<sup>2</sup>. Por este motivo decidimos desarrollar una heurística simple que si considerara el controlar las violaciones.

Esta heurística es del tipo conocido como *Integer Allocation* o *Dive & Fix*

---

<sup>1</sup>Específicamente la restricción estricta que aparece cuando la violación de una restricción elástica es necesariamente nula.

<sup>2</sup>En las pruebas numéricas descubrimos que la heurística de *CPLEX* estaba dispuesta incluso a generar soluciones con valores objetivos extremadamente negativos (debido a las penalizaciones causadas por grandes violaciones) con tal de encontrar al menos una solución entera.

[Wol98] y se basa en la formulación con restricciones de volumen elásticas (2.1). Consiste básicamente en tomar una solución a la relajación lineal de la formulación, fijar un número de variables fraccionarias y volver a resolver la relajación lineal, repitiendo el proceso hasta que todas las variables estén en valores enteros. Complementamos esta heurística simple permitiendo la corrección de grandes violaciones de las restricciones durante el proceso de fijado de variables.

Dependiendo de la relación entre la tasa de descuento y la razón de crecimiento de los árboles, el primer o último periodo serán los más convenientes para cosechar desde el punto de vista de los beneficios. En el caso de las instancias estudiadas en las pruebas computacionales el último periodo resultó ser el más conveniente y de hecho las restricciones de volumen activas en la formulación con restricciones de volumen estrictas (formulación 1.4) siempre fueron las restricciones de cotas superiores sobre el volumen cosechado (1.4e). Por este motivo la heurística comienza fijando variables en este último periodo. El hecho de que el último periodo sea el más conveniente puede ser discutible desde la perspectiva económica/forestal, pero debido a la *simetría* del problema con respecto al tiempo esto no es un gran problema desde el punto de vista de la optimización del problema<sup>3</sup>.

A continuación describiremos someramente la heurística para el caso en que el último periodo sea el más conveniente de cosechar. El apéndice B contiene una descripción más detallada de la heurística.

La heurística comienza eliminando fracciones en el último periodo, fijando variables con valores fraccionarios a 0 o 1. Luego vuelve a resolver la relajación lineal repitiendo el proceso hasta que no queden variables fraccionarias. La heurística eliminará todas las fracciones en un periodo antes de pasar al siguiente periodo a ser procesado (en el caso estudiando, el periodo anterior cronológicamente). Además las variables fijadas en periodos ya procesados se mantendrán fijas mientras se trabaja en los periodos anteriores para evitar la aparición de nuevas fracciones.

Antes de pasar al periodo anterior la heurística corregirá de una manera poco agresiva las violaciones de restricciones de volumen relacionadas con el último periodo procesado. Es decir la heurística intentará corregir las violaciones siempre que con eso no cree nuevas violaciones. Al terminar de procesar el primer periodo la heurística corregirá de manera agresiva todas las violaciones de restricciones de volumen que sean mayores al nivel acep-

---

<sup>3</sup>En el apéndice F discutiremos en detalle este punto y presentamos algunos resultados numéricos para una instancia en la que el periodo más conveniente es el primero.

table. Es decir, en esta corrección final la heurística corregirá las violaciones aún cuando la corrección cree nuevas violaciones, repitiendo el proceso iterativamente hasta que todas las violaciones se encuentren controladas.

En algunas instancias la corrección agresiva de las violaciones disminuye considerablemente el valor de la función objetivo por lo que es esos casos se realiza una segunda pasada de la heurística. Con fin de realizar esta segunda pasada, todas las variables que se hayan fijado en 0 se dejarán nuevamente libres, mientras que las variables fijadas en 1 mantendrán sus valores. Esta vez, la heurística comenzará procesando el primer periodo preocupándose de no crear nuevas violaciones. Para este efecto la heurística agrega cotas sobre las variables asociadas a las violaciones  $(\underline{w}_t, \overline{w}_t)$  con el objetivo de mantenerlas controladas y sólo fijará variables si esta acción no viola necesariamente las cotas. Las cotas evitan la necesidad de corregir violaciones muy grandes, pero es posible que estas cotas tengan un efecto generador de fracciones similar al de las restricciones de volumen estrictas. Afortunadamente, en la mayoría de los casos, se tiene que antes de llegar a la segunda pasada de la heurística un gran número de variables ya se hayan fijas en 1. Esto ayuda a que el efecto generador de fracciones de las cotas tenga una mínima incidencia sobre el desempeño de la heurística.

### 3.3. Ramificación por Restricciones

La ramificación por fijación de variables en el árbol de *Branch & Bound* ( $B\&B$ ) puede causar cambios muy grandes de un nodo a sus hijos en comparación con la ramificación por restricciones. Más específicamente, si se fija a 1 en cierto periodo una variable asociada a un *cluster factible* demasiado grande es posible que esto tenga implicaciones muy fuertes acerca del volumen de madera cosechado en ese periodo. Una consecuencia de esto es que las penalizaciones que mantienen controladas las violaciones de las restricciones de volumen en un nodo pueden no mantenerlas controladas en sus hijos. Por otro lado la ramificación por restricciones normalmente impone leves cambios de un nodo del árbol de  $B\&B$  a sus hijos, en particular, rara vez fija una variable a 1.<sup>4</sup> Debido a esto, la ramificación por restricciones parece ser más adecuada para problemas con restricciones penalizadas. A esto se suma el hecho conocido de que la ramificación por restricciones da como resulta-

---

<sup>4</sup>A pesar de que este no es su objetivo, en ocasiones se puede interpretar una ramificación por restricciones como el fijado de una variable individual. Este es el caso de ramificaciones sobre la *holgura* de restricciones como la descrita en la sección 3.3.3.

do árboles de  $B\mathcal{E}B$  mucho más balanceados [RyaFos81]. Además el modelo presenta una forma natural de implementar ramificación por restricciones. Por estos motivos complementaremos el método de  $B\mathcal{E}B$  con este tipo de ramificación.

A continuación describiremos tres tipos de ramificación por restricciones que en conjunto permiten resolver eficientemente todas las posibles fracciones del modelo con o sin restricciones de volumen en todas las instancias que hemos probado.

Cualquier solución fraccionaria debe tener al menos una *celda básica* cosechada *fraccionariamente* en alguno de los siguientes sentidos:

- La *celda básica* es cosechada parcialmente en varios periodos a la vez.
- La *celda básica* es cosechada por multiples clusters en un periodo en particular.
- La *celda básica* es cosechada parcialmente considerando todos los periodos en los que es cosechada.

Las siguientes ramificaciones se encargan de resolver cada una de estas posibles fracciones.

### 3.3.1. Ramificación *Celdas/Tiempo*

Esta ramificación se encarga de eliminar fracciones generadas por una *celda básica* que está siendo cosechada parcialmente en varios periodos a la vez. Sea  $\hat{\mathbf{x}}$  una solución fraccionaria para la relajación lineal de cualquiera de los tres modelos en múltiples periodos (formulaciones 1.3, 1.4 o 2.1) y sean  $\hat{x}_{S,t}$  los valores de las variables  $x_{S,t}$  correspondientes. Supongamos que una *celda básica*  $u \in V$  esta siendo cosechada parcialmente en más de un periodo. Es decir para algún  $n > 1$  existen  $\{t_1, \dots, t_n\} \subseteq \{1, \dots, \mathcal{T}\}$  tal que:

$$\sum_{S \in \mathcal{C}(\{u\})} \hat{x}_{S,t_i} > 0 \quad \forall i \in \{1, \dots, n\}$$

Entonces existe un periodo  $t_0 \in \{1, \dots, \mathcal{T}\}$  tal que :

$$\sum_{t=1}^{t_0} \sum_{S \in \mathcal{C}(\{u\})} \hat{x}_{S,t} > 0$$

y

$$\sum_{t=t_0+1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} \hat{x}_{S,t} > 0$$

Una ramificación que eliminaría esta solución fraccionaria sería imponer que la *celda*  $u$  sea cosechada sólo después o sólo antes del periodo  $t$ . Es decir, imponer:

$$\sum_{t=1}^{t_0} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} = 0$$

en la primera rama y

$$\sum_{t=t_0+1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} = 0$$

en la segunda.

Notemos que ambas ramas incluyen la posibilidad de que la *celda básica*  $u$  no sea cosechada en ningún periodo.

### 3.3.2. Ramificación Tipo Ryan-Foster sobre restricciones de Cliques proyectados

Esta ramificación se encarga de eliminar las fracciones generadas cuando una celda esta siendo cosechada por más de un cluster a la vez en el mismo período. En el apéndice C veremos que en algunos casos raros esta ramificación no es suficiente para resolver este tipo de fracciones, pero la elegimos debido a su eficiencia y a la justificación teórica detrás de su funcionamiento. Además el problema de esta ramificación siempre puede ser corregido por las otras dos ramificaciones descritas en esta sección (ver apéndice C). Este tipo de ramificación fue propuesta por Ryan y Foster en 1981 [RyaFos81] para problemas de *partición de conjuntos*<sup>5</sup> y se ha utilizado frecuentemente en problemas de generación de columnas [BarEllJohNemSavVan98]. Esta ramificación tiene una justificación interesante para el caso de problemas de *partición de conjuntos* [RyaFos81, BarEllJohNemSavVan98] la que puede extenderse directamente a problemas de *empaquetamiento de conjuntos*<sup>6</sup>.

Esta ramificación funciona de la siguiente manera. Sea  $\hat{\mathbf{x}}$  una solución fraccionaria para la relajación lineal de cualquiera de los tres modelos en

---

<sup>5</sup>*Set Partitioning.*

<sup>6</sup>*Set Packing.*

múltiples periodos (formulaciones 1.3, 1.4 o 2.1) y sean  $\hat{x}_{S,t}$  los valores de las variables  $x_{S,t}$  correspondientes. Supongamos que para cierto periodo  $\hat{t}$  existen dos *cluster factibles*  $S_1, S_2 \in \mathcal{C}$  cosechando una misma celda  $u \in V$ . Es decir:

$$\hat{x}_{S_1, \hat{t}}, \hat{x}_{S_2, \hat{t}} > 0 \wedge \exists u \in V \text{ tal que } u \in S_1, u \in S_2$$

Debido a las restricciones de *clusters incompatibles* (implementadas como restricciones de cliques proyectados) tendremos necesariamente que:

$$\hat{x}_{S_1, \hat{t}}, \hat{x}_{S_2, \hat{t}} < 1$$

y por lo tanto las variables  $x_{S_1, \hat{t}}$  y  $x_{S_2, \hat{t}}$  tendrán valores fraccionarios.

Debido a que  $u \in S_1 \cap S_2 \neq \emptyset$  debe existir un *clique maximal*  $K_1$  en el grafo forestal  $G(V, E)$  tal que  $K_1$  intersekte tanto a  $S_1$  como a  $S_2$ . Además, como  $S_1$  y  $S_2$  son dos clusters distintos debería existir otro *clique*  $K_2$  que intersekte sólo a uno de los dos clusters (En el apéndice C veremos que la ramificación falla cuando esta suposición, normalmente cierta, falla). Supongamos que  $K_2$  intersekte sólo a  $S_2$ . Así, una ramificación que eliminaría esta solución fraccionaria sería prohibir la utilización de clusters que intersektan ambos *cliques* en una rama y de clusters que intersektan exactamente a uno de los *cliques* en la otra. Es decir, imponer en la primera rama:

$$\sum_{S \in \mathcal{C}(K_1) \cap \mathcal{C}(K_2)} x_{S, \hat{t}} = 0$$

y en la segunda rama:

$$\sum_{S \in \mathcal{C}(K_1) \Delta \mathcal{C}(K_2)} x_{S, \hat{t}} = 0$$

donde  $\mathcal{C}(K)$  es el conjunto de todos los *clusters factibles* que intersektan al *clique*  $K$  y  $\Delta$  es la diferencia simétrica (i.e.  $A \Delta B := (A \cup B) \setminus (A \cap B)$ ).

### 3.3.3. Ramificación sobre holgura de restricciones Celda/Tiempo

Esta ramificación se encarga de eliminar fracciones generadas por una *celda básica* cosechada parcialmente considerando todos los periodos en los que es cosechada. Sea  $\hat{\mathbf{x}}$  una solución fraccionaria para la relajación lineal de

cualquiera de los tres modelos en múltiples periodos (formulaciones 1.3, 1.4 o 2.1) y sean  $\hat{x}_{S,t}$  los valores de las variables  $x_{S,t}$  correspondientes. Supongamos que una celda  $u \in V$  esta siendo cosechada parcialmente considerando todos los periodos en los que es cosechada. Es decir:

$$0 < \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} \hat{x}_{S,t} < 1$$

Una ramificación que eliminaría esta solución fraccionaria seria imponer:

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} = 1$$

en la primera rama y

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} = 0$$

en la segunda.

Claramente la posibilidad de que la celda no sea cosechada esta incluida en la segunda rama.

### 3.4. Actualización Dinámica de las Penalizaciones

Como mencionamos en la sección 3.1, el hecho de que las violaciones aumenten de manera impredecible a medida que se desciende en el árbol de *Branch & Bound* ( $B\&B$ ), o a medida que se fijan variables en la heurística, sugiere el ajustar dinámicamente las penalizaciones a través de todo el método.

Lo ideal es comenzar con las menores penalizaciones que mantengan las violaciones controladas o en su defecto nulas. Luego podemos permitir que las violaciones aumenten dentro de cierto rango a medida que avanzamos en el árbol de  $B\&B$ . Si alguna de las violaciones crece demasiado, normalmente bastará aumentar levemente la penalización asociada para mantenerla controlada. Es posible que un conjunto de ramificaciones cause una violación de volumen grande que no pueda ser corregida sin liberar variables fijadas por las ramificaciones. En este caso, el aumentar la penalización asociada incluso a niveles muy altos no logrará disminuir la violación. El sub-árbol enraizado en el nodo creado por esta sucesión de ramificaciones no contendrá ninguna

solución con violaciones controladas, por lo que podría ser podado. Esto no es un gran problema, pues el valor objetivo de la relajación lineal en ese nodo será muy bajo a causa de las penalizaciones, por lo que es muy probable que el sub-árbol sea podado automáticamente por el procedimiento de  $B\mathcal{E}B$ . Sólo es importante que el procedimiento de actualización de las penalizaciones considere esta posibilidad para no continuar inutilmente aumentando la penalizaciones.

Hemos elegido la actualización dinámica de las penalizaciones en cada nodo del árbol de  $B\mathcal{E}B$ , pero no dentro de la heurística. Es decir, si luego de resolver la relajación lineal del problema en un nodo del árbol de  $B\mathcal{E}B$  descubrimos que las violaciones de las restricciones de volumen son mayores a las permitidas, aumentaremos las penalizaciones correspondientes y resolveremos nuevamente la relajación lineal, repitiendo este ciclo hasta que las violaciones estén dentro de los límites aceptables o se detecte que el aumento de las penalizaciones no tiene ningún efecto<sup>7</sup>. Durante las pruebas iniciales notamos que la heurística funciona mejor si las violaciones iniciales están controladas. Además, si podemos garantizar que la relajación lineal de cada nodo del árbol de  $B\mathcal{E}B$  tiene las violaciones de las restricciones de volumen dentro de los límites aceptables, entonces las soluciones enteras encontradas en hojas del árbol de  $B\mathcal{E}B$  también las tendrán controladas. De esta manera las soluciones encontradas naturalmente por el procedimiento de  $B\mathcal{E}B$  podrán ser consideradas válidas. No actualizamos las penalizaciones dentro de la heurística debido a que sería recargar su implementación demasiado y el procedimiento de corrección de violaciones de la heurística hace las actualizaciones dentro de ella sean innecesarias.

Hay dos maneras en que podemos actualizar las penalizaciones. Una es simplemente modificando los coeficientes de la función objetivo y la otra es re-escribiendo la formulación para poder aumentar las penalizaciones mediante cortes. Con el segundo método sólo es posible aumentar las penalizaciones, a menos que sea posible eliminar cortes que ya se encuentren en la formulación. Esto no es un gran problema ya que en general sólo aumentaremos las penalizaciones.

---

<sup>7</sup>Por ejemplo, cuando hay violaciones causadas por las ramificaciones como se discutió anteriormente.

### 3.4.1. Modificando la Función Objetivo

La primera forma de actualizar las penalizaciones es simplemente modificando los coeficientes  $\underline{p}_t$  y  $\bar{p}_t$  en la función objetivo de la formulación con restricciones de volumen elásticas (formulación 2.1). Desafortunadamente, la mayoría de los softwares comerciales para resolver problemas de programación lineal entera (incluyendo *CPLEX*) no permiten este tipo de modificación durante el proceso de *Branch & Bound*.

### 3.4.2. Agregando Cortes

Una forma de agregar la actualización dinámica de penalizaciones a la resolución del problema mediante un software comercial de programación lineal entera es reescribir la formulación y actualizar las penalizaciones agregando cortes. Esta metodología nos permite también visualizar la actualización de las penalizaciones desde el punto de vista polihedral.

Supongamos que tenemos una restricción de volumen elástica de la forma:

$$\mathbf{v}^\top \mathbf{x} \leq w_t \quad (3.1)$$

y que el coeficiente de la función objetivo asociado a  $w_t$  es  $-p_t$  con  $p_t \in \mathbb{Q}_+$ .

Si multiplicamos la restricción (3.1) por  $p \in \mathbb{Q}_+ \setminus \{0\}$  obtendremos la restricción equivalente:

$$p \cdot \mathbf{v}^\top \mathbf{x} \leq p \cdot w_t \quad (3.2)$$

Si ahora reemplazamos el lado derecho  $p \cdot w_t$  de la ecuación (3.2) por una nueva variable  $pw_t$  obtendremos una ecuación sobre  $\mathbf{x}$  y la nueva variable  $pw_t$ :

$$p \cdot \mathbf{v}^\top \mathbf{x} \leq pw_t \quad (3.3)$$

Si el coeficiente de la función objetivo asociado a  $pw_t$  es igual a  $-1$  entonces, en particular, la ecuación (3.3) con  $p = p_t$  tendrá el mismo efecto sobre las variables  $\mathbf{x}$  que la restricción original (3.1).

Supongamos que agregamos desde un principio la variable positiva  $pw_t$  al modelo, con coeficiente asociado en la función objetivo igual a  $-1$ . Entonces,

en general, si queremos el mismo efecto sobre las variables  $\mathbf{x}$  de aumentar el coeficiente de la función objetivo asociado a  $w_t$  de  $-p_t$  a  $-q_t < -p_t$  bastará agregar al modelo el siguiente corte:

$$p_t \cdot \mathbf{v}^\top \mathbf{x} \leq pw_t \quad (3.4)$$

De esta manera la siguiente formulación (3.1) tendrá el mismo efecto sobre las variables  $x_{S,t}$  que la formulación original con restricciones de volumen elásticas (formulación 2.1).

---

**Formulación 3.1** (CPP-CLQ- $\mathcal{T}$  – EIVND) con Restricciones de Volumen en Forma de Cortes (CPP-CLQ- $\mathcal{T}$  – EIVND – CUT)

---

$$\text{máx} \quad \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t} - \sum_{t=2}^{\mathcal{T}} \underline{w}_t - \sum_{t=2}^{\mathcal{T}} \overline{w}_t \quad (3.1a)$$

s.a.

$$\sum_{S \in \mathcal{C}(K)} x_{S,t} \leq 1 \quad \forall K \in \mathcal{K}, \forall t \quad (3.1b)$$

$$\sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}(\{u\})} x_{S,t} \leq 1 \quad \forall u \in V \quad (3.1c)$$

$$\underline{p}_t \left( (1 - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} - \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} \right) \leq \underline{w}_t \quad \forall t \geq 2 \quad (3.1d)$$

$$\overline{p}_t \left( \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} - (1 + \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} \right) \leq \overline{w}_t \quad \forall t \geq 2 \quad (3.1e)$$

$$x_{S,t} \in \{0, 1\} \quad \forall S \in \mathcal{C}, \forall t \quad (3.1f)$$

$$\underline{w}_t, \overline{w}_t \geq 0 \quad \forall t \quad (3.1g)$$


---

Notemos que ahora las penalizaciones  $\underline{p}_t, \overline{p}_t$  no aparecen en la función objetivo (3.1a) sino en las restricciones de volumen (3.1d) y (3.1e).

Con esta formulacion el actualizar una penalización de  $\underline{p}_t$  o  $\overline{p}_t$  a  $\underline{q}_t > \underline{p}_t$

o  $\bar{q}_t > \bar{p}_t$  se reducirá a agregar el corte:

$$\underline{q}_t((1 - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1} - \sum_{S \in \mathcal{C}} v_{S,t} x_{S,t}) \leq \underline{w}_t \quad (3.5)$$

o

$$\bar{q}_t(\sum_{S \in \mathcal{C}} v_{S,t} x_{S,t} - (1 + \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1}) \leq \bar{w}_t \quad (3.6)$$

respectivamente.

La actualización de las penalizaciones mediante cortes sólo nos permite aumentarlas<sup>8</sup>. Esto no será un gran problema siempre que no realicemos aumentos grandes de las penalizaciones y por lo tanto las mantengamos a niveles bajos.

### 3.5. Podando el Árbol de *Branch & Bound*

El utilizar  $\Delta_E < \Delta$  para cumplir con las restricciones estrictas correspondientes presenta algunos inconvenientes a la hora de podar nodos en el árbol de *Branch & Bound* (*B&B*).

Supongamos que estamos utilizando la formulación con restricciones de volumen elásticas (formulación 2.1) con  $\Delta_E < \Delta$  y que de alguna forma podemos asegurar que las soluciones enteras factibles cumplirán con las restricciones de volumen estrictas de nivel  $\Delta$  (por ejemplo ajustando las penalizaciones o utilizando una heurística que respete estas restricciones). Entonces podemos definir los siguientes valores:

**Definición 3.5.1.** Dada una solución  $(x_{S,t}, \bar{w}_t, \underline{w}_t)$  (entera o no) para la formulación con restricciones de volumen elásticas (formulación 2.1) definiremos:

$$z_R(x_{S,t}, \bar{w}_t, \underline{w}_t) := \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t}$$

$$z_E(x_{S,t}, \bar{w}_t, \underline{w}_t) := \sum_{t=1}^{\mathcal{T}} \sum_{S \in \mathcal{C}} c_{S,t} x_{S,t} - \sum_{t=2}^{\mathcal{T}} \underline{p}_t \underline{w}_t - \sum_{t=2}^{\mathcal{T}} \bar{p}_t \bar{w}_t$$

es decir los valores objetivos de la formulación original con restricciones de volumen estrictas (formulación 1.4) y de la formulación con restricciones elásticas (formulación 2.1) respectivamente.

---

<sup>8</sup>A menos que el software de *B&B* nos permita eliminar cortes, lo que no es el caso de *CPLEX*.

Dada una solución  $\mathbf{x} = (x_{S,t})$  denotaremos  $z_R(\mathbf{x})$  y  $z_E(\mathbf{x})$  a estos mismos valores con los  $(\underline{w}_t, \bar{w}_t)$  asociados naturalmente a  $\mathbf{x}$ <sup>9</sup>.

Claramente  $z_E(\mathbf{x}) \leq z_R(\mathbf{x})$  y  $z_E(\mathbf{x}) < z_R(\mathbf{x})$  si  $\mathbf{x}$  no cumple con alguna de las restricciones de volumen estrictas de nivel  $\Delta_E$ .

Supongamos ahora que hemos encontrado una solución entera factible  $\hat{\mathbf{x}}$  que cumple con las restricciones de volumen estrictas (1.4d) y (1.4e) de nivel  $\Delta$ . Como  $\Delta_E < \Delta$  es muy probable que esta solución *no* cumpla con las restricciones estrictas de nivel  $\Delta_E$  y por lo tanto  $z_E(\hat{\mathbf{x}}) < z_R(\hat{\mathbf{x}})$ . Esto no es un problema mayor pues  $z_R(\hat{\mathbf{x}})$  puede ser calculado fácilmente.

Supongamos que luego, en cierto nodo  $N$  del árbol de  $B\mathcal{E}B$ , obtenemos una solución fraccionaria  $\tilde{\mathbf{x}}$  para la relajación lineal en ese nodo. Es decir que  $\tilde{\mathbf{x}}$  cumple:

$$\tilde{\mathbf{x}} \in \arg \max_{\mathbf{x} \in N} z_E(\mathbf{x}) \quad (3.7)$$

donde  $\mathbf{x} \in N$  quiere decir que  $\mathbf{x}$  cumple con las restricciones impuestas por las ramificaciones que dan origen al nodo  $N$ . Supongamos además que  $z_E(\tilde{\mathbf{x}}) < z_E(\hat{\mathbf{x}})$ . Si estamos usando un software comercial de  $B\mathcal{E}B$  el subárbol que tiene este nodo como raíz será podado, pues la función objetivo que ve el software es  $z_E(\cdot)$ . El problema es que como  $z_E(\hat{\mathbf{x}}) \leq z_R(\hat{\mathbf{x}})$  es posible que  $z_R(\tilde{\mathbf{x}}) > z_R(\hat{\mathbf{x}})$  y por lo tanto al podar el subárbol podríamos estar perdiendo una solución entera mejor. Una solución simple a este problema sería cambiar el criterio de podado de  $z_E(\tilde{\mathbf{x}}) < z_E(\hat{\mathbf{x}})$  a  $z_R(\tilde{\mathbf{x}}) < z_R(\hat{\mathbf{x}})$ . El problema es que:

$$z_R(\tilde{\mathbf{x}}) \leq \max_{\mathbf{x} \in N} z_R(\mathbf{x}) \quad (3.8)$$

por lo que al podar con este nuevo criterio aún podríamos estar perdiendo mejores soluciones enteras. Aún no hemos podido calcular *buenas* cotas para  $\max_{\mathbf{x} \in N} z_R(\mathbf{x})$  a partir de  $z_R(\tilde{\mathbf{x}})$ , por lo que por ahora utilizar el criterio de podado  $z_R(\tilde{\mathbf{x}}) < z_R(\hat{\mathbf{x}})$  parece ser una buena opción intermedia.

Desafortunadamente son pocos los softwares de  $B\mathcal{E}B$  que permiten modificar el criterio de podado<sup>10</sup>, por lo que para implementar el criterio de podado  $z_R(\tilde{\mathbf{x}}) < z_R(\hat{\mathbf{x}})$  probablemente sea necesario programar el método de  $B\mathcal{E}B$  desde 0.

---

<sup>9</sup>i.e. las violaciones de las restricciones de volumen de la solución  $(x_{S,t})$ .

<sup>10</sup>Por ejemplo, *CPLEX* no lo permite.

## Capítulo 4

# Resultados Computacionales

### 4.1. Instancias de Prueba

Se utilizaron dos mapas forestales para las pruebas computacionales. Un bosque real ubicado en el Norte de California llamado *El Dorado* y una grilla cuadrada de  $12 \times 12$  celdas generada aleatoriamente. Denotaremos estas instancias por *eldorado* y *12by12* respectivamente.

La grilla fue generada aleatoriamente con los siguientes parametros. A cada celda se le asignó aleatoriamente un area entre 20 y 40 (El area máxima para un *cluster factible* fue de  $A = 120$ ) y un volumen entre 100 y 1000. La tasa de crecimiento de los árboles y la tasa de descuento fueron fijadas en 7% y 5% respectivamente. El beneficio por volumen cosechado también se consideró fijo.

Para la instancia de *El Dorado* también se utilizó una tasa de descuento igual a 5% mientras que el crecimiento de los árboles fue variable y obtenido de datos reales.

Para ambos mapas se probaron instancias con 12 y 15 periodos. El siguiente cuadro muestra un resumen de las características de los mapas forestales usados en las instancias de prueba.

Mapa	# de celdas	# de <i>clusters factibles</i>	# de <i>cliques maximales</i>	Restricciones para 15 periodos	Variables para 15 periodos
eldorado	1363	21412	2105	32938	321180
12by12	144	2056	121	1959	30840

Cuadro 4.1: Características de las Instancias de Prueba

Las instancias serán identificadas como “nombre del mapa”+“número de periodos”. Por ejemplo, *eldoradot15* se referirá a la instancia de 15 periodos que ocupa el mapa de *El Dorado*.

## 4.2. Implementación

Se realizaron los siguiente tipos de pruebas computacionales basadas en el software *CPLEX 8.1* que serán descritas a continuación:

1. *CPLEX 8.1* usado como software de *Branch & Cut* sin modificaciones.
2. *CPLEX 8.1* usado como software de *Branch & Cut* con llamadas auxiliares para ramificación, heurísticas y cortes.
3. *CPLEX 8.1* usado como software *Simplex Dual* sobre un programa de *Branch & Bound (B&B)* propietario.

En el primer caso se utilizó el optimizador entero mixto de *CPLEX 8.1* directamente para resolver los problemas. No se modifico ninguno de los parámetros por defecto ni se utilizaron funciones auxiliares.

En el segundo caso se utilizó el optimizador entero mixto de *CPLEX 8.1* para resolver los problemas, pero se utilizo la función de llamadas auxiliares para agregar ramificaciones, heurísticas y cortes diseñados especialmente para el modelo de planificación de cosecha. Todas las funciones auxiliares fueron programadas en *C++*.

En el tercer caso solo se utilizó *CPLEX 8.1* para resolver relajaciones lineales del problema. En este caso se programó una version simple del algoritmo de *B&B* que utiliza la función de *Simplex Dual* de *CPLEX 8.1* para resolver las relajaciones lineales en cada uno de los nodos del árbol de *B&B*.

En los tres casos la generación de los problemas fue realizada por un programa en *C++* utilizando *ILOG Concert 1.2* [Cpx02].

El desarrollo del software fue realizado en un PC Pentium III, 800 mhz, 512 mb RAM con Mandrake Linux v9.0 y los resultados fueron obtenidos en un PC Pentium IV, 2.0 ghz, 2.0 gb RAM con Mandrake Linux v8.2. El compilador de *C++* utilizado fue gcc 3.0.

### 4.3. Cálculo de *gap*'s y objetivo de métodos elásticos

Debido a que queremos comparar el desempeño entre los métodos basados en restricciones elásticas y la resolución directa de la formulación con restricciones estrictas, en todos los resultados computacionales tomaremos como objetivo resolver algún problema con restricciones estrictas. En el caso de los resultados para la resolución del problema estricto en forma directa o con ramificación por restricciones (secciones 4.4.2 y 4.5.1) el problema objetivo será el mismo modelo resuelto. En el caso de los resultados para la formulación con restricciones elásticas se tomará como objetivo resolver el modelo con restricciones de volumen estrictas de un nivel  $\Delta$  especificado en cada sección. Para este objetivo se utilizarán restricciones elásticas de nivel  $\Delta_E < \Delta$  y sólo se aceptarán soluciones con violaciones controladas que *si* cumplan con las restricciones estrictas de nivel  $\Delta$  (como se discute en detalle en la sección 2.2). El nivel  $\Delta_E$  utilizado será especificado en cada sección.

Con el mismo objetivo de poder comparar resultados entre modelos con restricciones estrictas y elásticas todos los *gap*'s serán calculados con respecto al valor óptimo de la relajación lineal del modelo estricto con que se quiere cumplir. Además en el caso de los resultados para modelos elásticos la función considerada para calcular los *gap*'s será los beneficios reales sin penalizaciones. Por ejemplo, supongamos que nuestro objetivo es encontrar soluciones para el modelo con restricciones de volumen estrictas de nivel  $\Delta$  utilizando el modelo con restricciones elásticas de nivel  $\Delta_E < \Delta$ . Si  $\mathbf{x}$  es una solución que cumple con las restricciones de volumen estrictas de nivel  $\Delta$  entonces su *gap* será calculado como:

$$100 \cdot \frac{LP(\Delta) - z_R(\mathbf{x})}{z_R(\mathbf{x})}$$

donde  $z_R(\mathbf{x})$  es el beneficio real sin penalizaciones de  $\mathbf{x}$  (como fue definido en la sección 3.5) y  $LP(\Delta)$  es el valor óptimo de la relajación lineal de la formulación con restricciones de volumen estrictas de nivel  $\Delta$  (formulación 1.4) para la instancia correspondiente.

Claramente, en el caso en que se use una formulación estricta (secciones 4.4.2 y 4.5.1) este valor corresponderá sencillamente al *gap* entre el valor objetivo de la solución y el valor óptimo de la relajación lineal de la formulación resuelta.

#### 4.4. *CPLEX* utilizado como *Branch & Cut*

##### 4.4.1. Múltiples Periodos sin Restricciones de Volumen

Los siguientes cuadros muestran resultados para la formulación sin restricciones de volumen (formulación 1.3) al ser resuelta directamente por *CPLEX 8.1*.

El cuadro (4.2) muestra el valor óptimo de la relajación lineal del problema y el tiempo que tomó llegar a él. El cuadro (4.3) muestra información sobre el problema entero. Las primeras cuatro columnas de este último cuadro muestran la instancia resuelta, el valor de la solución óptima y el tiempo y nodos de *B&B* que tomó para declararla óptima. La quinta columna muestra el tiempo que tomó para encontrar la primera solución bajo un 1% de *gap*. Las últimas dos columnas muestran el tiempo en que fue encontrada la primera solución entera factible y el *gap* de esta.

<b>Instancia</b>	<b>LP</b>	<b>Tiempo LP [s]</b>
eldoradot12	6768816	115.00
eldoradot15	7114814	38.34
12by12t12	27913504	12.97
12by12t15	39585268	14.14

Cuadro 4.2: Valores LP de CPP-CLQ- $\mathcal{T}$

<b>Instancia</b>	<b>IP</b>	<b>Tiempo</b> [s]	<b>Nodos</b> <i>B&amp;B</i>	<b>1ra Sol.</b> <b>Bajo</b> <b>1 % <i>gap</i></b> [s]	<b>Tiempo</b> <b>1ra Sol.</b> <b>Factible</b> [s]	<b><i>gap</i></b> <b>1ra Sol.</b> <b>Factible</b> [%]
eldoradot12	6765360	720	30	448	173	6.58
eldoradot15	7114410	147	0	101	77	3.02
12by12t12	27845993	501	789	24	14	33.61
12by12t15	39490166	524	732	32	19	38.32

Cuadro 4.3: Resultados IP de CPP-CLQ- $\mathcal{T}$

Podemos ver que *CPLEX* pudo encontrar soluciones enteras factibles muy rápidamente y fue muy fácil declarar como óptima la mejor solución. Además la cercanía entre los valores óptimos del problema entero y la relajación lineal confirman la fortaleza de la formulación sin restricciones de volumen.

#### 4.4.2. Múltiples Periodos con Restricciones de Volumen Exactas

Los siguientes cuadros muestran resultados para la formulación con restricciones de volumen estrictas (formulación 1.4) al ser resuelta directamente por *CPLEX 8.1*. Para esta prueba se utilizó un límite de tiempo de 8 horas.

El cuadro (4.4) muestra el valor óptimo de la relajación lineal del problema y el tiempo que tomó llegar a él para dos valores distintos de  $\Delta$ . El cuadro (4.5) muestra información sobre el problema entero. Las primeras dos columnas de este último cuadro muestran información acerca de la instancia resuelta. La tercera columna muestra el número de nodos de *B&B* procesados en las 8 horas. Las siguientes dos columnas muestran el *gap* de la mejor solución encontrada y cuando fue encontrada. La columna 6 muestra el tiempo en que fue encontrada la primera solución bajo 1% de *gap* y las últimas dos columnas muestran el tiempo en que fue encontrada la primera solución entera factible y el *gap* de esta. Un guión (-) quiere decir que no se encontró ninguna solución entera factible para esa instancia.

Instancia	$\Delta$	LP	Tiempo LP [s]
eldoradot12	0.10	6292354	309
eldoradot15	0.10	6812231	407
eldoradot12	0.15	6413097	251
eldoradot15	0.15	6931024	355
12by12t12	0.10	17959328	193
12by12t15	0.10	23071854	91
12by12t12	0.15	18970483	25
12by12t15	0.15	25012357	751

Cuadro 4.4: Valores LP de CPP-CLQ- $T - VND$

Instancia	$\Delta$	Nodos <i>B&amp;B</i>	gap Mejor Sol. [%]	Tiempo Mejor Sol. [s]	1ra Sol. Bajo 1 % gap [s]	Tiempo 1ra Sol. Factible [s]	gap 1ra Sol. Factible [%]
eldoradot12	0.10	2133	1.47	18606	-	18606	1.47
eldoradot15	0.10	1575	0.83	18315	10839	10839	1.00
eldoradot12	0.15	2087	0.50	11211	10719	2323	1.51
eldoradot15	0.15	2067	0.59	20733	20274	20274	0.77
12by12t12	0.10	388	-	-	-	-	-
12by12t15	0.10	394	-	-	-	-	-
12by12t12	0.15	634	-	-	-	-	-
12by12t15	0.15	342	-	-	-	-	-

Cuadro 4.5: Resultados IP de CPP-CLQ- $T - VND$

Podemos ver que al agregar las restricciones de volumen el problema se vuelve extremadamente difícil. *CPLEX* tiene muchos problemas para encontrar soluciones enteras factibles. Para la instancia de *El Dorado* finalmente logra encontrar buenas soluciones, pero luego de un tiempo considerable. Para las grilla en cambio no logra encontrar ninguna solución factible en las 8 horas de procesamiento.

#### 4.4.3. Múltiples Periodos con Restricciones de Volumen Elásticas y Penalizaciones Fijas

Los cuadros 4.6 y 4.7 muestran resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta directamente por *CPLEX 8.1*. Para esta prueba se utilizó un límite de tiempo de 4 horas.

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original (formulación 1.4) con restricciones de volumen estrictas de nivel  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$ )

Para esta prueba se utilizaron penalizaciones fijas e idénticas para todas las restricciones. Es decir se utilizo  $\underline{p}_t = \bar{p}_t = P$  con  $P$  fijo durante la optimización. El cuadro muestra los resultados para diversos valores de la penalización  $P$ .

El cuadro 4.6 muestra información acerca de las violaciones a las restricciones de volumen y el numero de soluciones válidas. Las primeras dos columnas muestran información sobre la instancia resuelta y el nivel de penalización  $P$ . Las siguientes cuatro columnas muestran información sobre las soluciones enteras factibles encontradas. Primero tenemos el número total de soluciones enteras factibles encontradas, seguido por el número de soluciones enteras factibles válidas. Luego aparecen el número de soluciones en las que no se cosecho nada en algún periodo. Incluimos esta información debido a que para estos casos el porcentaje de violación de las restricciones de volumen no puede ser calculado. La sexta columna presenta el porcentaje de soluciones enteras factibles consideradas válidas. Finalmente la séptima columna muestra el promedio de las violaciones máximas de todas las soluciones enteras factibles sin considerar aquellas en las que no se cosecho nada en algún periodo.

El cuadro 4.7 muestra información acerca de cuando fueron encontradas soluciones válidas y sus *gap*'s. Sólo se presentan resultados para las instancias de *El Dorado* ya que no fue posible encontrar ninguna solución válida para las grillas. Las primeras dos columnas muestran información sobre la instancia resuelta y el nivel de penalización  $P$ . Las siguientes cuatro columnas muestran información sobre las soluciones enteras factibles válidas encontradas. Esta información incluye el tiempo en que fueron encontradas y el *gap* de la primera y la mejor soluciones enteras factibles válidas. Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$  por lo que pueden ser comparados con los *gap*'s del cuadro (4.5).

Instancia	P	# de sol.	# de sol. válidas	# de sol. vol. 0	% de sol. válidas	violación máxima promedio [%]
eldoradot15	1	410	0	34	0	865.10
eldoradot15	10	167	0	1	0	1883.71
eldoradot15	100	106	1	1	0.94	15.06
eldoradot15	1000	55	2	1	3.64	17.03
eldoradot15	10000	37	3	1	8.11	11.90
eldoradot15	100000	15	0	1	0	26.17
eldoradot15	1000000	15	0	1	0	27.11
eldoradot15	10000000	15	0	1	0	25.82
eldoradot15	100000000	111	0	2	0	10.20
12by12t15	1	7295	0	7262	0	5.00E+17
12by12t15	10	3701	0	3558	0	1.01E+20
12by12t15	100	133	0	1	0	40.95
12by12t15	1000	47	0	1	0	13.80
12by12t15	10000	15	0	1	0	19.03
12by12t15	100000	14	0	1	0	19.03
12by12t15	1000000	14	0	1	0	19.03
12by12t15	10000000	15	0	1	0	19.37
12by12t15	100000000	15	0	1	0	19.19
eldoradot12	1	147	0	50	0	795.47
eldoradot12	10	171	0	1	0	2253.55
eldoradot12	100	52	5	2	9.62	16.62
eldoradot12	1000	20	3	0	15	29.80
eldoradot12	10000	19	2	0	10.53	29.98
eldoradot12	100000	21	2	1	9.52	28.63
eldoradot12	1000000	19	4	0	21.05	29.85
eldoradot12	10000000	24	5	0	20.83	24.33
eldoradot12	100000000	20	4	1	20.00	30.04
12by12t12	1	7920	0	7841	0	1.57E+33
12by12t12	10	2463	0	2245	0	1.30E+19
12by12t12	100	142	0	1	0	18.47
12by12t12	1000	39	0	1	0	12.41
12by12t12	10000	12	0	1	0	22.05
12by12t12	100000	11	0	1	0	23.96
12by12t12	1000000	9	0	1	0	25.01
12by12t12	10000000	10	0	1	0	23.96
12by12t12	100000000	10	0	1	0	23.54

Cuadro 4.6: Violaciones para Formulación Elástica con Penalizaciones Fijas

<b>Instancia</b>	<b>P</b>	<b>Tiempo 1ra sol. válida [s]</b>	<b>gap 1ra sol. válida [%]</b>	<b>Tiempo mejor sol. válida [s]</b>	<b>gap mejor sol. válida [%]</b>
eldoradot15	1	-	-	-	-
eldoradot15	10	-	-	-	-
eldoradot15	100	13131	0.78	13131	0.78
eldoradot15	1000	6349	0.36	6349	0.36
eldoradot15	10000	9884	0.47	9884	0.47
eldoradot15	100000	-	-	-	-
eldoradot15	1000000	-	-	-	-
eldoradot15	10000000	-	-	-	-
eldoradot15	100000000	-	-	-	-
eldoradot12	1	-	-	-	-
eldoradot12	10	-	-	-	-
eldoradot12	100	2617	0.61	10894	0.36
eldoradot12	1000	715	0.35	715	0.35
eldoradot12	10000	713	0.35	713	0.35
eldoradot12	100000	711	0.35	711	0.35
eldoradot12	1000000	742	0.34	742	0.34
eldoradot12	10000000	731	0.34	731	0.34
eldoradot12	100000000	729	0.34	729	0.34

Cuadro 4.7: Soluciones Válidas para Formulación Elástica con Penalizaciones Fijas

Lo primero que podemos notar es que el encontrar soluciones enteras factibles es mucho más fácil con las restricciones elásticas que con las estrictas, por su puesto muchas de estas soluciones no son válidas a menos que se encuentre un nivel de penalización adecuado. Para la instancia de *El Dorado* con  $P = 1000$  es posible encontrar excelentes soluciones válidas muy rápido. De hecho soluciones bajo 1% de *gap* son encontradas 3 a 14 veces más rápido que utilizando *CPLEX* sobre la formulación original con restricciones de volumen estrictas (formulación 1.4) y las soluciones encontradas son aún mejores. Desgraciadamente para la grilla ni siquiera niveles muy grandes de  $P$  son suficientes para obtener soluciones enteras válidas, a pesar de que ya para el nivel  $P = 100$  la relajación lineal inicial presenta

violaciones inferiores al 1 %

## 4.5. *CPLEX* con llamadas auxiliares

### 4.5.1. Ramificación por Restricciones sobre Modelo con Restricciones Estrictas

El siguiente cuadro muestra resultados para la formulación con restricciones de volumen estrictas (formulación 1.4) al ser resuelta por *CPLEX* complementado con las ramificaciones por restricciones descritas en la sección 3.3.

La ramificación por restricciones fue implementada como una llamada auxiliar de *CPLEX* y se utilizó un criterio sencillo para la selección del tipo de ramificación y las variables que participan en ella. El orden de selección del tipo ramificación fue: primero buscar una ramificación tipo *Celdas/Tiempo* (sección 3.3.1), si no es posible encontrar ninguna, buscar una de tipo *Ryan-Foster sobre restricciones de Cliques Projectados* (sección 3.3.2) o finalmente una de tipo *Holgura de Restricciones Celda/Tiempo* (sección 3.3.3). Para la selección dentro de las múltiples alternativas de ramificaciones tipo *Celdas/Tiempo* se utilizó un criterio de menor fraccionalidad y menor número de periodos<sup>1</sup>. Para la ramificación tipo *Ryan-Foster sobre restricciones de Cliques Projectados* se seleccionó la primera ramificación encontrada y para la de tipo *Holgura de Restricciones Celda/Tiempo* se seleccionó la celda más cercana a ser cosechada completamente o no cosechada para nada.

Para esta prueba se utilizó un límite de tiempo de 8 horas. Se desactivaron las heurísticas de *CPLEX* para asegurar que todas las soluciones enteras fueran encontrados en nodos hojas enteros del árbol de *B&B*. Además se desactivaron los *cortes fraccionarios de Gomory*, pues estos generan problemas con los cortes actualizadores de penalizaciones (sección 3.4.2) y queremos comparar los resultados de esta sección con los resultados que utilizan estas actualizaciones (secciones 4.5.3 y 4.5.5).

El cuadro presenta los resultados en un formato similar al del cuadro 4.5 en la sección 4.4.2 e incluye además el número total de soluciones enteras factibles encontradas.

---

<sup>1</sup>Se eligió un período  $t_0$  y una celda  $u$  tal que  $\sum_{t=1}^{t_0} \sum_{S \in \mathcal{C}(\{u\})} \hat{x}_{S,t}$  fuera lo más cercano a 0 ó 1 y  $t_0$  fuera lo más pequeño posible.

<b>Instancia</b>	$\Delta$	<b>Nodos BBB</b>	<b>gap Mejor Sol. [%]</b>	<b>Tiempo Mejor Sol. [s]</b>	<b>Tiempo 1ra Sol. Factible [s]</b>	<b>gap 1ra Sol. Factible [%]</b>	<b># Sol. Factibles</b>
eldoradot12	0.15	9836	3.68	27975	9971	7.12	5
eldoradot15	0.15	7996	8.96	11640	11620	8.99	2
12by12t12	0.15	150662	9.19	17613	210	44.49	33
12by12t15	0.15	125241	27.37	27133	826	77.14	16

Cuadro 4.8: Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T}$  – VND

Para la instancia de *El Dorado* la velocidad con que se encuentran soluciones enteras es similar a utilizar *CPLEX* directamente, pero los valores objetivo de las soluciones son bastante menores. Por otro lado para el caso de los reticulados el desempeño al utilizar ramificación por restricciones es notablemente mejor. En alrededor de 10 minutos ya es posible encontrar soluciones enteras. Por supuesto los valores obtenidos objetivo obtenidos no son muy buenos, pero los resultados cobran valor si consideramos que al utilizar *CPLEX* directamente no fue posible encontrar ninguna solución entera luego de 8 horas.

No es sorprendente que la ramificación por restricciones haya logrado encontrar soluciones enteras para los reticulados tan rápidamente. Después de todo, la ramificación por restricciones toma decisiones acerca de cuando y por quien será cosechada cada celda. Como además solo hay 144 celdas en el reticulado, las restricciones que deben imponerse para ver como cosechar todas las celdas no son demasiadas.

#### 4.5.2. Ramificación por Restricciones sobre Modelo con Restricciones Elásticas

El siguiente cuadro muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por *CPLEX* complementado con las ramificaciones por restricciones descritas en la sección 3.3.

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de

volumen con  $\Delta = 0,15$ )

La ramificación por restricciones fue implementada como una llamada auxiliar de *CPLEX* y se utilizó el mismo criterio de selección utilizado en la sección 4.5.1.

Para las penalizaciones se utilizaron valores previamente calculados que dan como resultado violaciones nulas en la relajación lineal inicial.

Para esta prueba se utilizó un límite de tiempo de 8 horas. Se desactivaron las heurísticas de *CPLEX* para asegurar que todas las soluciones enteras fueran encontrados en nodos hojas enteros del árbol de *B&B*. Además se desactivaron los *cortes fraccionarios de Gomory*, pues estos generan problemas con los cortes actualizadores de penalizaciones (sección 3.4.2) y queremos comparar los resultados de esta sección con los resultados que utilizan estas actualizaciones (secciones 4.5.3 y 4.5.5).

El cuadro presenta los resultados en el mismo formato que los cuadros 4.6 y 4.7 en la sección 4.4.3.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original con restricciones de volumen estrictas (1.4) con  $\Delta = 0,15$  por lo que pueden ser comparados con los GAP's del cuadro (4.5).

Instancia	# de sol.	# de sol. válidas	% de sol. válidas	violación máxima promedio [%]	Tiempo 1ra sol. válida [s]	gap 1ra sol. válida [%]	Tiempo mejor sol. válida [s]	gap mejor sol. válida [%]
eldoradot12	244	0	0.00	5.09	-	-	-	-
eldoradot15	295	1	0.34	2.21	27433	3.15	27433	3.15
12by12t12	375	37	9.87	36.76	186	58.47	26353	12.68
12by12t15	3273	2	0.06	98.12	1512	52.54	6066	28.63

Cuadro 4.9: Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T}$  – *EIVND*

Como se esperaba, es posible encontrar más soluciones enteras factibles con la ramificación por restricciones sobre el modelo elástico que sobre el modelo estricto. Por supuesto, si se utilizan penalizaciones fijas la mayor parte de estas soluciones no son válidas debido a las altas violaciones.

El desempeño final al utilizar ramificación por restricciones sobre el modelo elástico con actualización dinámica de penalizaciones es similar al obtenido al utilizarla sobre el modelo estricto.

### 4.5.3. Ramificación por Restricciones con Actualización Dinámica de Penalizaciones sobre Modelo con Restricciones Elásticas

El cuadro 4.10 muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por *CPLEX* complementado con las ramificaciones por restricciones descritas en la sección 3.3 y los cortes actualizadores de penalizaciones descritos en la sección 3.4.2.

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$ )

La ramificación por restricciones fue implementada como una llamada auxiliar de *CPLEX* y se utilizó el mismo criterio de selección utilizado en la sección 4.5.1.

Los cortes actualizadores de penalizaciones fueron implementados como una llamada auxiliar de *CPLEX*. Cada vez que se encontró una violación de las restricciones de volumen elásticas (con  $\Delta_E = 0,14$ ) superior a 1% se incorporó a la formulación el corte que aumenta la penalización correspondiente. El aumento de la penalización fue igual al beneficio por volumen promedio de las celdas. Las penalizaciones iniciales fueron tomadas con un valor de 10.

Para esta prueba se utilizó un límite de tiempo de 8 horas. Se desactivaron las heurísticas de *CPLEX* para asegurar que todas las soluciones enteras fueran encontrados en nodos hojas enteros del árbol de *B&B*. Además se desactivaron los *cortes fraccionarios de Gomory*, pues estos generan problemas con los cortes actualizadores de penalizaciones.

El cuadro presenta los resultados en el mismo formato que los cuadros 4.6 y 4.7 en la sección 4.4.3.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$  por lo que pueden ser comparados con los *gap*'s del cuadro (4.5).

Instancia	# de sol.	# de sol. válidas	% de sol. válidas	violación máxima promedio [%]	Tiempo Ira sol. válida [s]	gap Ira sol. válida [%]	Tiempo mejor sol. válida [s]	gap mejor sol. válida [%]
eldoradot12	2	2	100.00	0.04	12947	18.31	13290	18.29
eldoradot15	2	2	100.00	0	16180	12.83	25751	6.41
12by12t12	10	7	70.00	5.78	485	42.55	6346	6.28
12by12t15	12	7	58.33	36.12	3335	43.22	7472	27.87

Cuadro 4.10: Resultados Ramificación por Restricciones y Actualización Dinámica de Penalizaciones sobre CPP-CLQ- $T$  – EIVND

Podemos ver que al utilizar la actualización dinámica de las penalizaciones el porcentaje de soluciones válidas aumenta considerablemente en comparación a cuando se utilizan penalizaciones fijas.

#### 4.5.4. Heurística y Ramificación por Restricciones sobre Modelo con Restricciones Elásticas

El cuadro 4.11 muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por *CPLEX* complementado con las ramificaciones por restricciones descritas en la sección 3.3 y la heurística descrita en la sección 3.2.

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y  $\Delta_E = 0,09$  con el objetivo de encontrar soluciones para la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$  y  $\Delta = 0,10$  respectivamente. Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% o 9% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$  y  $\Delta = 0,10$  respectivamente)

La ramificación por restricciones fue implementada como una llamada auxiliar de *CPLEX* y se utilizó el mismo criterio de selección utilizado en la sección 4.5.1.

Para las penalizaciones se utilizaron valores previamente calculados que dan como resultado violaciones nulas en la relajación lineal inicial.

Para esta prueba se utilizó un límite de tiempo de 4 horas. Se desactivaron las heurísticas de *CPLEX* para asegurar que todas las soluciones enteras fueran encontradas por la heurística de la sección 3.2. Además se desactivaron los *cortes fraccionarios de Gomory*, pues estos generan problemas con los cortes actualizadores de penalizaciones (sección 3.4.2) y queremos

comparar los resultados de esta sección con los resultados que utilizan estas actualizaciones (secciones 4.5.3 y 4.5.5).

El cuadro presenta los resultados en el mismo formato que el cuadro 4.5 en la sección 4.4.2.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original (formulación 1.4) con el  $\Delta$  correspondiente (0,15 o 0,10) por lo que pueden ser comparados con los *gap*'s del cuadro (4.5).

Todas las soluciones enteras factibles fueron generadas por la heurística, la que asegura que además sean válidas.

Instancia	$\Delta$	Nodos <i>B&amp;B</i>	<i>gap</i> Mejor Sol. [%]	Tiempo Mejor Sol. [s]	1ra Sol. Bajo 1% GAP [s]	Tiempo 1ra Sol. Factible [s]	<i>gap</i> 1ra Sol. Factible [%]
eldoradot12	0.10	21	0.43	14058	1834	1834	0.42
eldoradot15	0.10	7	0.43	14394	3015	3015	0.57
eldoradot12	0.15	24	0.22	14537	1758	1758	0.45
eldoradot15	0.15	12	0.26	5719	2513	2513	0.39
12by12t12	0.10	395	2.80	7996	-	627	14.29
12by12t15	0.10	26	11.52	10040	-	1880	38.42
12by12t12	0.15	88	2.78	7100	-	510	7.99
12by12t15	0.15	21	7.90	8345	-	786	25.14

Cuadro 4.11: Resultados Ramificación por Restricciones y Heurística sobre CPP-CLQ- $\mathcal{T}$  – *EIVND*

Al utilizar la heurística dentro del método de *B&B* podemos encontrar buenas soluciones muy rápidamente. Para el caso de *El Dorado* las soluciones encontradas son levemente mejores que al utilizar *CPLEX* sobre el modelo estricto. Además, las primeras soluciones enteras factibles siempre son encontradas más rápidamente (hasta 10 veces más rapido). Para el caso del reticulado los resultados son aún más alentadores. En solo cuatro horas es posible encontrar soluciones relativamente buenas y las primeras soluciones enteras ya aparecen antes de una hora.

#### 4.5.5. Heurística y Ramificación por Restricciones con Actualización Dinámica de Penalizaciones sobre Modelo con Restricciones Elásticas

El siguiente cuadro muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por *CPLEX*

complementado con las ramificaciones por restricciones descritas en la sección 3.3, los cortes actualizadores de penalizaciones descritos en la sección 3.4.2 y la heurística descrita en la sección 3.2.

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original (formulación 1.4) con  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$ )

La ramificación por restricciones fue implementada como una llamada auxiliar de *CPLEX* y se utilizó el mismo criterio de selección utilizado en la sección 4.5.1.

Los cortes actualizadores de penalizaciones fueron implementados como una llamada auxiliar de *CPLEX*. Cada vez que se encontró una violación de las restricciones de volumen elásticas (con  $\Delta_E = 0,14$ ) superior a 1% se incorporó a la formulación el corte que aumenta la penalización correspondiente. El aumento de la penalización fue igual al beneficio por volumen promedio de las celdas. Las penalizaciones iniciales fueron tomadas con un valor de 10.

Para esta prueba se utilizó un límite de tiempo de 4 horas. Se desactivaron las heurísticas de *CPLEX* para asegurar que todas las soluciones enteras fueran encontradas por la heurística de la sección 3.2. Además se desactivaron los *cortes fraccionarios de Gomory*, pues estos generan problemas con los cortes actualizadores de penalizaciones.

El cuadro presenta los resultados en el mismo formato que el cuadro 4.5 en la sección 4.4.2.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original (formulación 1.4) con  $\Delta = 0,15$  por lo que pueden ser comparados con los *gap*'s del cuadro (4.5).

Todas las soluciones enteras factibles fueron generadas por la heurística, la que asegura que además sean válidas.

<b>Instancia</b>	$\Delta$	<b>Nodos <i>B&amp;B</i></b>	<b><i>gap</i> Mejor Sol. [%]</b>	<b>Tiempo Mejor Sol. [s]</b>	<b>1ra Sol. Bajo 1 % GAP [s]</b>	<b>Tiempo 1ra Sol. Factible [s]</b>	<b><i>gap</i> 1ra Sol. Factible [%]</b>
eldoradot12	0.15	19	0.35	5603	3120.88	3120.88	0.36
eldoradot15	0.15	8	0.27	10537	4158	4158	0.67
12by12t12	0.15	214	2.78	11692	-	825	20.78
12by12t15	0.15	63	7.18	1647.73	-	1346	20.74

Cuadro 4.12: Resultados Ramificación por Restricciones, Actualización Dinámica de Penalizaciones y Heurística sobre CPP-CLQ- $\mathcal{T}$  – *EIVND*

El utilizar la actualización dinámica de las penalizaciones, en vez de utilizar penalizaciones fijas calculadas con anterioridad, no parece afectar demasiado el desempeño de la heurística dentro del método de *Branch & Bound*. De hecho en un par de casos las soluciones son un poco mejores o son encontradas levemente más rápido. Es comprensible el hecho de que las primeras soluciones factibles sean encontradas más lentamente, debido a que son necesarias unas cuantas iteraciones del proceso de actualización de penalizaciones para poder obtener una relajación inicial con violaciones controladas.

## 4.6. *Branch & Bound* Especializado

En esta sección estudiaremos de manera computacional el problema de podar nodos del árbol de *Branch & Bound* (*B&B*) al utilizar el modelo elástico que fue discutido en la sección 3.5. Para esto realizaremos pruebas en instancias pequeñas (3 y 5 periodos) comparando el desempeño al podar nodos utilizando el valor óptimo elástico ( $z_E$ ) y su parte sin penalizaciones ( $z_R$ ).

Ya que *CPLEX* solo puede podar y seleccionar nodos de acuerdo a la función objetivo elástica  $z_E$  (debido a que *CPLEX* solo tiene acceso a esta función) utilizaremos un método de *B&B* programado especialmente para este propósito. El programa es básicamente una implementación muy simple de *B&B* que permite podar con la función objetivo elástica o con la función objetivo real (i.e. sin las penalizaciones).

#### 4.6.1. Ramificación por Restricciones sobre Modelo Elástico utilizando $z_R$

El siguiente cuadro muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por un método de *B&B* propietario que utiliza *CPLEX* para resolver las relajaciones lineales y las ramificaciones por restricciones descritas en la sección 3.3. El podado de nodos se realizó con  $z_R$ .

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$ )

Para las penalizaciones se utilizaron valores previamente calculados que dan como resultado violaciones nulas en la relajación lineal inicial.

Para esta prueba se utilizó un límite de tiempo de 1 hora.

El cuadro presenta los resultados en el mismo formato que el cuadro 4.5 en la sección 4.4.2.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$ .

<b>Instancia</b>	$\Delta$	<b>gap Mejor Sol. [%]</b>	<b>Tiempo Mejor Sol. [s]</b>	<b>1ra Sol. Bajo 1% GAP [s]</b>	<b>Tiempo 1ra Sol. Factible [s]</b>	<b>gap 1ra Sol. Factible [%]</b>
eldoradot3	0.15	1.05	1325	-	1325	1.05
eldoradot5	0.15	1.68	3164	-	2060	1.69
12by12t3	0.15	0.42	3586	59	50	1.11
12by12t5	0.15	0.23	252	219	149	3.51

Cuadro 4.13: Resultados Ramificación por Restricciones sobre CPP-CLQ-T – EIVND utilizando  $z_R$

#### 4.6.2. Ramificación por Restricciones sobre Modelo Elástico utilizando $z_E$

El siguiente cuadro muestra resultados para la formulación con restricciones de volumen elásticas (formulación 2.1) al ser resuelta por un método de *B&B* propietario que utiliza *CPLEX* para resolver las relajaciones lineales y las ramificaciones por restricciones descritas en la sección 3.3. El podado de nodos se realizó con  $z_E$ .

Para estas pruebas se utilizó  $\Delta_E = 0,14$  y se tomo como objetivo el encontrar soluciones para la formulación original con restricciones de volumen estrictas (formulación 1.4) con  $\Delta = 0,15$ . Por este motivo solo se consideraron válidas soluciones que no violaran las restricciones de volumen de 14% por más de 1% (i.e. que cumplieran con las restricciones estrictas de volumen con  $\Delta = 0,15$ )

Para las penalizaciones se utilizaron valores previamente calculados que dan como resultado violaciones nulas en la relajación lineal inicial.

Para esta prueba se utilizó un límite de tiempo de 1 hora.

El cuadro presenta los resultados en el mismo formato que el cuadro 4.5 en la sección 4.4.2.

Estos *gap*'s son calculados con respecto a la relajación lineal de la formulación original con restricciones de volumen estrictas (1.4) con  $\Delta = 0,15$ .

<b>Instancia</b>	$\Delta$	<b>gap Mejor Sol. [%]</b>	<b>Tiempo Mejor Sol. [s]</b>	<b>1ra Sol. Bajo 1% GAP [s]</b>	<b>Tiempo 1ra Sol. Factible [s]</b>	<b>gap 1ra Sol. Factible [%]</b>
eldoradot3	0.15	1.05	1330	-	1330	1.05
eldoradot5	0.15	1.68	3154	-	2050	1.69
12by12t3	0.15	0.42	3573	60	51	1.11
12by12t5	0.15	0.17	1980	275	152	3.51

Cuadro 4.14: Resultados Ramificación por Restricciones sobre CPP-CLQ- $\mathcal{T}$  – *EIVND* utilizando  $z_E$

Por lo menos en instancias pequeñas el criterio de podado no parece tener gran incidencia en los valores objetivos de las soluciones encontrados. En algunos casos incluso es posible encontrar mejores soluciones utilizando la función objetivo elástica.

## 4.7. Resumen de Resultados

A continuación presentamos un resumen de los resultados obtenidos con los diferentes métodos, utilizando el siguiente formato para identificarlos.

**Sin Volumen** Formulación sin restricciones de volumen (formulación 1.3) resuelta directamente por *CPLEX*. Resultados de sección 4.4.1.

**Volumen Estricto** Formulación con restricciones de volumen estrictas (formulación 1.4) resuelta directamente por *CPLEX*. Resultados de sección 4.4.2.

**Elástico Fijo** Formulación con restricciones de volumen elásticas (formulación 2.1) con penalizaciones fijas resuelta directamente por *CPLEX*. Resultados de sección 4.4.3. Se consideró siempre el mejor resultado obtenido para los diferentes niveles de penalización  $P$ . Los resultados en las tablas de resumen no siempre corresponden al mismo  $P$ .

**RR Estricto** Formulación con restricciones de volumen estrictas (formulación 1.4) resuelta por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3. Resultados de sección 4.5.1.

**RR Elástico** Formulación con restricciones de volumen elásticas (formulación 2.1) resuelta por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3. Resultados de sección 4.5.2.

**RR Elástico + AD** Formulación con restricciones de volumen elásticas (formulación 2.1) resuelta por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3 y la actualización dinámica de penalizaciones de la sección 3.4. Resultados de sección 4.5.3.

**Heurística** Formulación con restricciones de volumen elásticas (formulación 2.1) resuelta por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3 y la heurística de la sección 3.2. Resultados de sección 4.5.4.

**Heurística + AD** Formulación con restricciones de volumen elásticas (formulación 2.1) resuelta por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3, la heurística de la sección 3.2 y la actualización dinámica de penalizaciones de la sección 3.4. Resultados de sección 4.5.5.

Todos los resultados con restricciones de volumen de esta sección corresponden a soluciones que cumplen con restricciones de volumen estrictas de nivel  $\Delta = 0,15$ .<sup>2</sup> Hemos incluido también los resultados sin restricciones de volumen solamente como referencia.

El siguiente cuadro muestra el tiempo que le tomo a cada método encontrar una solución entera válida bajo el 1 % de *gap*. Un guión (-) significa que no fue posible encontrar ninguna solución que cumpliera con este requisito.

Método	eldoradot12 tiempo [s]	eldoradot15 tiempo [s]	12by12t12 tiempo [s]	12by12t15 tiempo [s]
Sin Volumen	448	101	24	32
Volumen Estricto	11211	20274	-	-
Elástico Fijo	711	6349	-	-
RR Estricto	-	-	-	-
RR Elástico	-	-	-	-
RR Elástico + AD	-	-	-	-
Heurística	1758	2513	-	-
Heurística + AD	4158	3121	-	-

Cuadro 4.15: Tiempo primera solución bajo 1 % *gap* para diferentes métodos de resolución

Podemos ver que para las instancias de *El Dorado* algunos métodos basados en restricciones elásticas logran encontrar soluciones bajo 1 % de *gap* bastante más rápido que al utilizar restricciones estrictas. Alrededor de 10 veces más rápido dependiendo del método. Lamentablemente para las instancias de la grilla ningún método con restricciones de volumen es capaz de encontrar una solución bajo 1 % de *gap*.

El cuadro 4.16 presentado a continuación nos muestra el mejor *gap* obtenido luego de las primeras 4 horas de procesamiento. Un guión (-) significa que no fue posible encontrar ninguna solución entera válida en el límite de tiempo. *Optimal* significa que fue posible encontrar y declarar óptima la mejor solución encontrada.

<sup>2</sup>Para el caso de métodos basados en restricciones elásticas se utilizaron los criterios de solución *válida* descritos en la sección 4.3.

Método	eldoradot12 <i>gap</i> [%]	eldoradot15 <i>gap</i> [%]	12by12t12 <i>gap</i> [%]	12by12t15 <i>gap</i> [%]
Sin Volumen	optimal	optimal	optimal	optimal
Volumen Estricto	0.50	-	-	-
Elástico Fijo	0.34	0.36	-	-
RR Estricto	7.12	8.96	9.32	27.64
RR Elástico	-	-	58.47	52.54
RR Elástico + AD	18.29	-	6.28	27.87
Heurística	0.22	0.26	2.78	7.90
Heurística + AD	0.35	0.27	2.78	7.18

Cuadro 4.16: Mejor *gap* obtenido luego de 4 horas para diferentes métodos de resolución

Los dos métodos que utilizan la heurística presentan los mejores resultados en el sentido del mejor *gap* obtenido. Además parecen ser los más robustos pues son capaces de encontrar soluciones relativamente buenas para todas las instancias.

Finalmente los cuadros 4.17 y 4.18 presentan las características de la primera solución entera válida encontrada por los diferentes métodos.

Método	eldoradot12 <i>tiempo</i> [s]	eldoradot12 <i>gap</i> [%]	eldoradot15 <i>tiempo</i> [s]	eldoradot15 <i>gap</i> [%]
Sin Volumen	173	6.58	77	3.02
Volumen Estricto	2323	1.51	20274	0.77
Elástico Fijo	711	0.35	6349	0.36
RR Estricto	9971	7.12	11620	8.99
RR Elástico	-	-	27433	3.15
RR Elástico + AD	12947	18.31	16180	12.83
Heurística	1758	0.45	2513	0.39
Heurística + AD	3121	0.36	4158	0.67

Cuadro 4.17: Primera solución entera factible y válida para instancias de *El Dorado*

Método	12by12t12 <i>tiempo</i> [s]	12by12t12 <i>gap</i> [%]	12by12t15 <i>tiempo</i> [s]	12by12t15 <i>gap</i> [%]
Sin Volumen	14	33.61	19	38.32
Volumen Estricto	-	-	-	-
Elástico Fijo	-	-	-	-
RR Estricto	210	44.49	826	77.14
RR Elástico	186	58.47	1512	52.54
RR Elástico + AD	485	42.55	3335	43.22
Heurística	510	7.99	786	25.14
Heurística + AD	825	20.78	1346	20.74

Cuadro 4.18: Primera solución entera factible y válida para instancias de *12by12*

Podemos ver que algunos métodos elásticos tienden a encontrar la primera solución más rápido e incluso con mejores valores. De hecho comparado con *CPLEX* sobre el modelo con restricciones estrictas la heurística encuentra las primeras soluciones enteras hasta 100 veces más rápido <sup>3</sup>.

El análisis de estos resultados puede sugerir que el método con penalizaciones fijas es muy eficiente para la instancia de *El Dorado*, pero debemos tener en cuenta que para este resumen se seleccionó *a posteriori* la mejor solución para diferentes valores de  $P$ .

Finalmente, los métodos que incluyen la heurística parecen presentar el mejor balance entre eficiencia y robustez entre las instancias.

---

<sup>3</sup>Para la instancia del reticulado con 15 periodos con restricciones de volumen estrictas de  $\pm 15\%$  *CPLEX* sólo logró encontrar una solución factible luego de 22 horas.

# Capítulo 5

## Conclusiones

### 5.1. Resultados

La motivación principal de esta memoria ha sido resolver un problema de planificación forestal con restricciones de area y volumen en múltiples periodos. Con este objetivo hemos tomado como base el *Problema de Empaquetamiento de Clusters* fortalecido con restricciones de *cliques proyectados*. Esta formulación, desarrollada en [GoyMurBarEpsWei03, Goy01], permite resolver eficientemente el problema de planificación forestal con restricciones de area en múltiples periodos sin restricciones de volumen. Al agregar restricciones de volumen estrictas, la formulación pierde gran parte de sus buenas propiedades y sólo es posible encontrar soluciones para instancias con pocos periodos. Por este motivo, esta memoria se ha concentrado en estudiar la posibilidad de reemplazar las restricciones de volumen estrictas por otras menos perjudiciales. Dada que las restricciones de volumen no son requerimientos estrictos del problema de planificación forestal decidimos utilizar versiones elásticas de ellas con el fin de recuperar las buenas propiedades del *Problema de Empaquetamiento de Clusters* sin restricciones de volumen.

Durante esta memoria hemos podido confirmar computacionalmente que las restricciones de volumen estrictas son una de las principales causas de la dificultad del problema. Al incorporar versiones elásticas de las restricciones de volumen los resultados computacionales han sugerido que estas recuperan gran parte de las propiedades de integralidad del *Problema de Empaquetamiento de Clusters*, pero que este beneficio tiene un costo. El uso de restricciones elásticas genera la necesidad de estudiar formas de mantener controladas las violaciones de las restricciones. Esto nos ha llevado a

estudiar la implementación de un procedimiento de actualización dinámica de las penalizaciones, el desarrollo de un procedimiento de ramificación por restricciones y una heurística fijadora de variables. Finalmente hemos estudiado superficialmente algunos aspectos relacionados con la poda del árbol de *Branch & Bound* cuando se utilizan restricciones elásticas. A continuación presentamos en detalle los resultados asociados a cada uno de estos temas, seguido por algunos comentarios finales.

### 5.1.1. Restricciones Generadoras de Fracciones

Los resultados numérico presentados en las secciones 4.4.1 y 4.4.2 parecen confirmar la suposición de que gran parte de la dificultad para resolver el problema radica en el efecto generador de fracciones de las restricciones estrictas de volumen. Este hecho unido a la validez desde el punto de vista de la planificación forestal de violar levemente las restricciones de volumen han sugerido naturalmente la utilización de versiones elásticas de las restricciones de volumen.

### 5.1.2. Uso de Restricciones Elásticas

Los resultados computacionales de la sección 4.4.3 confirman el que las restricciones de volumen elásticas recuperan gran parte de las propiedades de integralidad del modelo sin restricciones de volumen desarrollado en [GoyMurBarEpsWei03]. También podemos ver que a veces es posible encontrar niveles de penalización que permitan encontrar soluciones con violaciones controladas, pero esto no es siempre el caso. Esto justifica la discusión en el capítulo 3 acerca de la elección y actualización dinámica de las penalizaciones, la ramificación por restricciones y el desarrollo de una heurística que tenga en cuenta el control de las violaciones.

### 5.1.3. Actualización Dinámica de las Penalizaciones

Los resultados computacionales de la sección 4.4.3 nos muestran que la utilización de penalizaciones fijas no siempre es práctico, por lo que estudiamos como implementar la actualización dinámica de las penalizaciones durante la optimización. En la sección 3.4.2 solucionamos el problema de incorporar la actualización dinámica de las penalizaciones a implementaciones comerciales de métodos tipo *Branch & Bound* o *Branch & Cut* utilizando

una reformulación de las restricciones elásticas. Numéricamente, en la sección 4.5.3 podemos ver que la incorporación de la actualización dinámica de las penalizaciones puede aumentar considerablemente el número de soluciones enteras con violaciones controladas, pero que también puede empeorar la calidad de las soluciones en algunas instancias.

#### 5.1.4. Ramificación por Restricciones

El principal motivo para utilizar ramificaciones por restricciones fue intentar minimizar el efecto de la ramificación sobre las violaciones, pero en las secciones 4.5.1 y 4.5.2 vimos que la ramificación por restricciones puede ser útil tanto al utilizar la formulación con restricciones estrictas como elásticas. En el primer caso facilita el encontrar soluciones enteras, en especial cuando el número de celdas/nodos básicas en el grafo forestal es pequeño. En el segundo, la ramificación por restricciones puede ayudar un poco a encontrar soluciones con violaciones controladas.

#### 5.1.5. Heurística

Los mejores resultados desde el punto de vista de la calidad de las soluciones, la rapidez en que fueron encontradas y la efectividad en diversas instancias son obtenidas al utilizar la heurística descrita en la sección 3.2 dentro del proceso de *Branch & Bound* ( $B\&B$ ). Esto no es sorprendente debido a que parte de la facilidad para resolver problemas de tipo *empaquetamiento de conjuntos*<sup>1</sup> puro con formulaciones fortalecidas (Como la formulación de *Empaquetamiento de Clusters* sin restricciones de volumen) viene de la efectividad de heurísticas *fijadoras de variables*. Las restricciones elásticas recuperan la estructura de *empaquetamiento de conjuntos* pura del problema de *empaquetamiento de clusters* mientras que la posibilidad de corregir violaciones de volumen dentro de la heurística permite mantener controladas las violaciones. Esto permite que la heurística genere buenas soluciones muy rápidamente.

Por otro lado la heurística es suficientemente robusta como para ser incorporada directamente dentro del proceso de  $B\&B$ . De hecho, la ramificación por restricciones usada en el método de  $B\&B$  parece complementar muy bien la heurística, diversificando la naturaleza glotona de esta.

---

<sup>1</sup>Set Packing.

Finalmente los resultados de la sección 4.5.5 nos muestran que la heurística puede funcionar mejor si la relajación inicial tiene violaciones controladas, por lo que se verá beneficiada si se implemente junto con la actualización dinámica de las penalizaciones.

#### 5.1.6. Podado del Árbol de *Branch & Bound*

Con respecto a los problemas para podar nodos del árbol de *Branch & Bound* discutidos en la sección 3.5, los resultados numéricos preliminares de la sección 4.6 sugieren que desde el punto de vista del desempeño computacional este aspecto no sería muy influyente. De esta forma, el cómo podar nodos sólo tendría interés a la hora de probar la optimalidad de una solución.

#### 5.1.7. Comentarios Finales

El buen desempeño del método basado en restricciones elásticas apoya su utilización en vez de las versiones estrictas. A esto se suma el hecho de que las restricciones estrictas no modelan adecuadamente la realidad de la planificación forestal, puesto que pequeñas violaciones a las restricciones de volumen *si* son aceptables.

La posibilidad de violar las restricciones de volumen no esencial para justificar las restricciones elásticas. En los resultados computacionales de esta memoria hemos utilizado las restricciones elásticas como una herramienta para resolver un problema con restricciones estrictas<sup>2</sup> y aún así hemos obtenido buenos resultados. De todas maneras, la utilización de las restricciones elásticas como una herramienta para modelar la flexibilidad de los requerimientos sobre los flujos de volumen de madera probablemente darán aún mejores resultados. En las siguientes secciones discutiremos más este punto junto con otros temas por investigar relacionados con las restricciones elásticas.

### 5.2. Temas por Investigar y Trabajo en Curso

Diversos aspectos sobre la implementación de un método basado en restricciones de volumen elásticos no han quedado completamente estudiados

---

<sup>2</sup>ver sección 4.3.

en esta memoria. Algunos de ellos son el detalle de la actualización dinámica de penalizaciones, el como podar nodos del árbol de *Branch & Bound* para probar optimalidad, el modelar realísticamente la flexibilidad de los requerimientos sobre los flujos de volumen de madera, estudiar la aplicabilidad del método a problemas reales y la posibilidad de ajustarlo a otros problemas similares. A continuación discutiremos cada uno de estos puntos.

### 5.2.1. Detalle de la Actualización Dinámica de Penalizaciones

En la sección 3.4 hemos estudiado como agregar una actualización dinámica de las penalizaciones al método de *Branch & Bound* y a implementaciones comerciales de éste, pero no hemos analizado en detalle el como actualizar las penalizaciones. Falta aún estudiar la cantidad en la que deben aumentarse las penalizaciones cuando las violaciones son demasiado grandes. En esta memoria siempre se aumentaron las penalizaciones por un valor fijo igual a el beneficio por volumen promedio de las celdas, pero no es claro que esta sea una buena elección.

### 5.2.2. Podado del Árbol de *Branch & Bound*

Los resultados numéricos preliminares de la sección 4.6 sugieren que desde el punto de vista del desempeño computacional el utilizar la función objetivo con penalizaciones para podar y seleccionar nodos en el árbol de *Branch & Bound* (*B&B*) no es muy perjudicial. A pesar de esto, desde el punto de vista teórico sigue siendo interesante el estudiar el como podar adecuadamente el árbol de *B&B*. El lograr esto podría permitir probar la optimalidad de una solución, transformando el método basado en restricciones elásticas en un real método de optimización.

### 5.2.3. Permitir Violaciones de las Restricciones de Volumen

Por otro lado, en todos los resultados computacionales presentados se intenta cumplir con alguna restricción estricta de volumen,<sup>3</sup> lo que no es realmente necesario desde el punto de vista de la planificación forestal. Lo ideal sería tomar cierta restricción de volumen como base y realmente permitir violaciones a ella (i.e. utilizar  $\Delta_E = \Delta$ ) siempre que estén controladas

---

<sup>3</sup>ver sección 4.3.

con cierto criterio. Esto sugiere el desarrollo de criterios que se adecúen a los requerimientos de los planificadores forestales (por ejemplo, promedio de violaciones bajo cierto valor) y la incorporación de criterios difusos de selección de soluciones. Por ejemplo, si se toma como criterio aceptar sólo soluciones con violaciones bajo 1% desde el punto de vista práctico no tendría sentido rechazar una solución con una violación de 1,0001%. Otra opción sería utilizar algún tipo criterio *multiobjetivo* entre el valor objetivo y las violaciones o un método de selección interactivo para el usuario final.

#### 5.2.4. Ramificación por Restricciones

La ramificación por ramificaciones sólo fue estudiada superficialmente ya que fue principalmente usada para complementar la heurística incluida dentro de procedimiento de *Branch & Bound*. Aún así, tuvo un muy buen desempeño por si sola incluso sobre el modelo con restricciones de volumen estrictas. Esto motiva a estudiar en detalle su implementación, incluyendo criterios de selección más sofisticados y otros posibles tipos de ramificación.

#### 5.2.5. Estudiar Aplicabilidad del Método

Antes de saber si es práctico utilizar este método en problemas reales se deben estudiar los siguientes puntos.

En el apéndice F se discute que las *tasas de descuento* utilizadas en las pruebas computacionales pueden no ser realistas. También se comenta que el método basado en restricciones elásticas tendrá su mayor utilidad cuando las restricciones de volumen sean fuertemente activas.<sup>4</sup> Desafortunadamente es posible que para ciertas *tasas de descuento* todos los periodos sean igual de convenientes de cosechar y por lo tanto las restricciones de volumen estrictas no sean tan perjudiciales. Por este motivo, la utilidad del método basado en restricciones elásticas podría variar mucho de instancia a instancia.

Finalmente también debe estudiarse la validez de las suposiciones acerca de la velocidad de crecimiento descritas en la sección 1.1.2 o analizarse la efectividad del método bajo otras suposiciones más realistas.

---

<sup>4</sup>y por lo tanto el modelo con restricciones estrictas sea muy difícil de resolver.

### 5.2.6. Utilización del Método en Problemas Similares

La heurística y la ramificación por restricción están desarrolladas especialmente para el modelo de planificación forestal, por lo que probablemente no puedan ser adaptadas a otros problemas. Por otro lado, la implementación de la actualización dinámica de las penalizaciones puede aplicarse directamente a otros modelos, sólo es necesario identificar las restricciones generadoras de fracciones que necesiten ser elastizadas. Queda por determinar, eso si, la efectividad que las restricciones elásticas puedan tener en otros problemas.

# Bibliografía

- [BarEllJohNemSavVan98] C. Barnhart, Johnson E.L., Nemhauser G.L., Savelsberg M.W.P., Vance P.H. *Branch-and-Price: Column Generation for Solving Huge Integer Programs*, Operations Research, Vol 46, No. 3.
- [BarGilDav98] Barrett T., Gilles J. y L. Davis. 1998. *Economic and Fragmentation Effects of Clearcut Restrictions*. Forest Science Vol. 44, N4, 569-577.
- [BosBet01] Boston K. y Bettinger P. 2001. *The economic impact of green-up constraints in the southeastern United States*. Forest Ecology and Management 145: 191-202.
- [ClaMelMacD00] Clark M.M., R.D. Meller y T.P. McDonald. 2000. *A three-stage heuristic for harvest scheduling with access road network development*. Forest Science 46: 204-218.
- [EhrRya03] Ehrgott M. y Ryan D. 2003. *The Method of Elastic Constraints for Multiobjective Combinatorial Optimization and its Application in Airline Crew Scheduling*, Multi-Objective Programming and Goal Programming, Springer Verlag, Berlin.
- [FisDorSch89] Fichser S. Dornbusch R. y Schmalensee R. 1989. *Economía, Segunda Edición*, McGraw-Hill/Interamericana de España, España.
- [Goy01] Goycoolea M. 2001. *Un Problema de Planificación Forestal con Restricción de Área*, Memoria para optar al título de Ingeniero Civil Matemático.
- [GoyMurBarEpsWei03] Goycoolea M., Murray A., Barahona F., Epstein R., y Weintraub A., 2003. *Harvest scheduling subject to maximum area re-*

*restrictions: exploring exact approaches.*, Operations Research, por aparecer

- [Hok83] Hokans R.H. 1983. *Evaluating spatial feasibility of harvest schedules with simulated stand-selection decisions*. Journal of Forestry 81: 601-603,613.
- [Cpx02] Ilog. 2002. *ILOG CPLEX 8.0 Users Manual*.
- [LocMoo93] Lockwood C. y T. Moore. 1993. Harvest scheduling with spatial constraints: a simulated annealing approach. Can. J. For. Res. 23: 468-478.
- [Mur99] Murray A.T. 1999. *Spatial Restrictions in Harvest Scheduling*. Forest Science. Vol. 45, No. 1. 1-8.
- [MurWei02] Murray A. T. y Weintraub A. 2002. *Scale and unit specification influences in harvest scheduling with maximum area restrictions*. Forest Science, por aparecer.
- [NemWol98] Nemhauser G.L., Wolsey L.A. 1998. *Integer and Combinatorial Optimization*. John Wiley and Sons, Inc., New York.
- [Pad73] Padberg, M.W. 1973. *On the facial structure of set packing polyhedra*. Math. Prog. 5,199-215.
- [RicGun00] Richards E.W. y Gunn E.A. 2000. *A model and tabu search method to optimize stand harvest and road construction schedules*. Forest Science 46: 188-203.
- [RyaFos81] Ryan D. y Foster B. 1981. *An Integer Programming Approach to Scheduling*. Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling. A. Wren, (Ed.). North Holland, Amsterdam. 269-280.
- [ThoHalLyoMil73] Thompson E.F., Halterman B.G., Lyon T.S. y Miller R.L. 1973. *Integrating timber and wildlife management planning*. Forestry Chronicle 47: 247-250.
- [Wol98] Wolsey, L.A. 1998. *Integer Programming*, John Wiley & Sons, New York

## Apéndice A

# Penalizaciones para Garantizar Violaciones Nulas

Supongamos que queremos resolver el siguiente problema:

---

**Formulación A.1** Problema Genérico Original

---

$$\text{máx } \mathbf{c}^\top \bar{\mathbf{x}} \quad (\text{A.1a})$$

*s.a.*

$$A\bar{\mathbf{x}} \leq \mathbf{b} \quad (\text{A.1b})$$

$$B\bar{\mathbf{x}} \leq \mathbf{f} \quad (\text{A.1c})$$

$$\bar{\mathbf{x}} \geq \mathbf{0} \quad (\text{A.1d})$$

---

cuyo dual es:

---

**Formulación A.2** Dual del Problema Genérico Original

---

$$\text{mín} \quad \mathbf{b}^\top \bar{\boldsymbol{\mu}} + \mathbf{f}^\top \bar{\boldsymbol{\pi}} \quad (\text{A.2a})$$

*s.a.*

$$A^\top \bar{\boldsymbol{\mu}} + B^\top \bar{\boldsymbol{\pi}} \geq \mathbf{c} \quad (\text{A.2b})$$

$$\bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\pi}} \geq \mathbf{0} \quad (\text{A.2c})$$

---

y decidimos tratar las restricciones  $B\bar{\mathbf{x}} \leq \mathbf{f}$  como restricciones elásticas, obteniendo el siguiente problema con restricciones elásticas:

---

**Formulación A.3** Problema Genérico con Restricciones Elásticas

---

$$\text{máx} \quad \mathbf{c}^\top \underline{\mathbf{x}} - \mathbf{p}^\top \mathbf{s} \quad (\text{A.3a})$$

*s.a.*

$$A\underline{\mathbf{x}} \leq \mathbf{b} \quad (\text{A.3b})$$

$$B\underline{\mathbf{x}} - \mathbf{s} \leq \mathbf{f} \quad (\text{A.3c})$$

$$\underline{\mathbf{x}}, \mathbf{s} \geq \mathbf{0} \quad (\text{A.3d})$$

---

cuyo dual es:

---

**Formulación A.4** Dual del Problema Genérico con Restricciones Elásticas

---

$$\text{mín} \quad \mathbf{b}^\top \underline{\boldsymbol{\mu}} + \mathbf{f}^\top \underline{\boldsymbol{\pi}} \quad (\text{A.4a})$$

*s.a.*

$$A^\top \underline{\boldsymbol{\mu}} + B^\top \underline{\boldsymbol{\pi}} \geq \mathbf{c} \quad (\text{A.4b})$$

$$\underline{\boldsymbol{\pi}} \leq \mathbf{p} \quad (\text{A.4c})$$

$$\underline{\boldsymbol{\mu}}, \underline{\boldsymbol{\pi}} \geq \mathbf{0} \quad (\text{A.4d})$$

---

Las condiciones de holgura complementaria nos dicen que todo par primal/dual óptimo  $(\underline{\mathbf{x}}^*, \mathbf{s}^*), (\underline{\boldsymbol{\mu}}^*, \underline{\boldsymbol{\pi}}^*)$  de A.3 y A.4 deben cumplir:

$$(\mathbf{c} - A^t \underline{\mu}^* - B^t \underline{\pi}^*)_i \underline{\mathbf{x}}_i^* = 0 \quad (\text{A.5a})$$

$$(\underline{\pi}^* - p)_i \mathbf{s}_i^* = 0 \quad (\text{A.5b})$$

$$(\mathbf{f} - B \underline{\mathbf{x}}^* + \mathbf{s}^*)_i \underline{\pi}_i^* = 0 \quad (\text{A.5c})$$

$$(\mathbf{b} - A \underline{\mathbf{x}}^*)_i \mu_i^* = 0 \quad (\text{A.5d})$$

En particular la ecuación A.5c nos dice que si  $\mathbf{p}_i \neq \underline{\pi}_i^*$  para todo  $i$  entonces  $\mathbf{s}_i = 0$  para todo  $i$  y por lo tanto no hay violaciones de las restricciones  $B \underline{\mathbf{x}} \leq \mathbf{f}$ .

Notemos ahora que si  $\mathbf{p} \geq \bar{\pi}^*$ , donde  $\bar{\pi}^*$  es el valor de  $\bar{\pi}$  para *una* solución óptima de A.2, entonces los problemas A.2 y A.4 son idénticos y por lo tanto sus conjuntos de soluciones óptimas también lo son. En particular  $\bar{\pi}^*$  también será el valor de  $\underline{\pi}$  para *una* solución óptima de A.4. Por lo tanto si  $p > \bar{\pi}^*$  tendremos que las restricciones elastizadas no serán violadas. Es decir, basta elegir penalizaciones levemente mayores a los valores de las variables duales asociados a las restricciones correspondientes para garantizar que estas no serán violadas (Por lo menos en la relajación lineal en el caso de un problema entero).

## Apéndice B

# Descripción Detallada de la Heurística *Integer Allocation*

Asumiremos que la heurística está utilizando la formulación con restricciones elásticas (formulación 2.1) para resolver la formulación con restricciones estrictas (1.4) con  $\Delta_E \leq \Delta$  y posiblemente  $\Delta_E < \Delta$ . Por este motivo el objetivo de la heurística será generar soluciones que cumplan con las restricciones estrictas de volumen de nivel  $\Delta$ . Además asumiremos que el último periodo es el periodo más conveniente para cosechar desde el punto de vista del beneficio (ver sección 3.2 y apéndice F). En el caso en que el primer periodo sea el más conveniente la heurística será análoga.

### B.1. Primera Pasada de la Heurística

A continuación presentamos el pseudo-código de la primera pasada de la heurística. Luego describiremos el proceso de fijado de variables y los procesos de corrección de violaciones.

---

**ALGORITMO B.1 PRIMERA PASADA DE LA HEURÍSTICA**

---

**VARIABLES:**

$\hat{x}_{S,t}$  : valores de las variables  $x_{S,t}$  obtenidos en la última solución de la relajación lineal.

**begin**

[inicialización]

Resolver la relajación lineal y obtener valores  $\hat{x}_{S,t}$

**for**  $\hat{t} = \mathcal{T}$  **to** 1 **do****while**  $\exists \hat{S} \in \mathcal{C}, \hat{x}_{\hat{S},\hat{t}} \in (0, 1)$  **do**

Fijar  $\hat{x}_{S,\hat{t}}$  fraccionarios a 0 o 1

Resolver *LP* y obtener nuevos valores  $\hat{x}_{S,t}$

**end while**

Corregir conservadoramente violaciones relacionadas con periodo  $\hat{t}$

Fijar variables  $\hat{x}_{S,\hat{t}}$  en sus valores actuales

**end while**

Corregir agresivamente violaciones en todos los periodos

**end**

---

**B.1.1. Fijado de Variables**

El proceso de fijado de variables está pensado para no crear infactibilidad sobre las restricciones de la formulación sin restricciones de volumen (1.3). El proceso también trata de minimizar el número de violaciones a las restricciones de volumen que crea.

La dirección en que se fijen las variables fraccionarias dependerá del valor de:

$$M := \max_{S \in \mathcal{C}} \hat{x}_{S,\hat{t}}$$

y se realizará de la siguiente manera:

- **(Si  $M > 0,5$ )** Todas las variables  $x_{S,\hat{t}}$  cuyo valor  $\hat{x}_{S,\hat{t}}$  sea mayor que 0,5 y cercano a  $M^1$  se fijarán en 1.
- **(Si  $M = 0,5$ )** Sólo una variable con valor igual a 0,5 se fijara a 1

---

<sup>1</sup>i.e.  $\hat{x}_{S,\hat{t}} > M - \delta$  con  $\delta > 0$  un parámetro pequeño.

- (Si  $M < 0,5$ ) Fijaremos a 0 variables con valores bajo 0,5 mientras el volumen retirado del periodo  $\hat{t}$  no cause directamente una violación de las restricciones de volumen entre los periodos  $\hat{t}$  y  $\hat{t} + 1$ .

### B.1.2. Corrección de Violaciones de Restricciones de Volumen

Las correcciones actúan de la siguiente manera, si se tiene que para cierto  $\tilde{t}$ :

$$(1 - \Delta) \sum_{S \in \mathcal{C}} v_{S, \tilde{t}-1} \hat{x}_{S, \tilde{t}-1} - \sum_{S \in \mathcal{C}} v_{S, \tilde{t}} \hat{x}_{S, \tilde{t}} > 0$$

la corrección simplemente fijara variables  $x_{S, \tilde{t}-1}$  a 0 hasta eliminar la violación. En cambio si se tiene:

$$\sum_{S \in \mathcal{C}} v_{S, \tilde{t}} \hat{x}_{S, \tilde{t}} - (1 + \Delta) \sum_{S \in \mathcal{C}} v_{S, \tilde{t}-1} \hat{x}_{S, \tilde{t}-1} > 0$$

la corrección fijará variables  $x_{S, \tilde{t}}$  a 0 hasta eliminar la violación.

La diferencia entre las versiones conservadora y agresiva es simplemente que la primera sólo corregirá una violación si puede hacerlo sin crear otra. En cambio, la agresiva corregirá iterativamente las violaciones hasta eliminarlas todas. Claramente, esto puede terminar fijando en 0 un gran número de variables y por lo tanto disminuir considerablemente el valor objetivo de la solución.

## B.2. Segunda Pasada de la Heurística

Luego de finalizada la primera pasada de la heurística todas las variables fijadas en 0 son dejadas libres (i.e. se les permite nuevamente tener valores positivos) y se procede a la segunda pasada de la heurística. La segunda pasada de la heurística es igual a la primera, salvo que esta vez no se realizan correcciones de las violaciones sino que se previenen con restricciones de volumen auxiliares. A continuación presentamos el pseudo-código de la segunda pasada de la heurística. Luego describiremos las restricciones de volumen auxiliares utilizadas. El proceso de fijado de variables es idéntico al usado en la primera pasada.

**Variabes :**

$\hat{x}_{S,t}$  : valores de las variables  $x_{S,t}$  obtenidos en la ultima  
solucion de la relajacion lineal.

**begin**

[inicialización]

Agregar restricciones de volumen auxiliares

Resolver la relajación lineal y obtener valores  $\hat{x}_{S,t}$

**for**  $\hat{t} = 1$  **to**  $\mathcal{T}$  **do**

**while**  $\exists \hat{S} \in \mathcal{C}, \hat{x}_{\hat{S},\hat{t}} \in (0, 1)$  **do**

Fijar  $\hat{x}_{S,\hat{t}}$  fraccionarios a 0 o 1

Resolver *LP* y obtener nuevos valores  $\hat{x}_{S,t}$

Actualizar restricciones de volumen auxiliares

**end while**

Fijar variable  $\hat{x}_{S,\hat{t}}$  en sus valores actuales

**end while**

**end**

---

### B.2.1. Restricciones de Volumen Auxiliares

Debido a que el la segunda pasada de la heurística no queremos crear violaciones a las restricciones de volumen de nivel  $\Delta$  nos gustaría imponer las siguientes restricciones sobre  $\underline{w}_t$  y  $\bar{w}_t$ :

$$\underline{w}_t \leq (\Delta - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1}$$

$$\bar{w}_t \leq (\Delta - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} x_{S,t-1}$$

Esto es equivalente a agregar las restricciones de volumen estrictas de nivel  $\Delta$ . Una aproximación a esto sería agregar las siguientes cotas superiores sobre  $\underline{w}_t$  y  $\bar{w}_t$ :

$$\underline{w}_t \leq (\Delta - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} \hat{x}_{S,t-1}$$

$$\bar{w}_t \leq (\Delta - \Delta_E) \sum_{S \in \mathcal{C}} v_{S,t-1} \hat{x}_{S,t-1}$$

donde  $\hat{x}_{S,t-1}$  son los valores de  $x_{S,t-1}$  obtenidos en la última solución a la relajación lineal.

En la segunda pasada de la heurística estas cotas son agregadas y actualizadas cada vez que se resuelve la relajación lineal. Estas restricciones tendrán prácticamente el mismo efecto generador de fracciones que las restricciones de volumen estrictas originales, pero debido a que en la segunda pasada de la heurística la mayor parte de las variables ya están fijas esto no debería ser un gran problema.

## Apéndice C

# Problemas con Ramificación de *Cliques Proyectados*

Las tres ramificaciones por restricciones presentadas en la sección 3.3 deberían poder resolver todas las fracciones del problema debido a que eliminan todos los tipos de fracciones que pueden aparecer. Desafortunadamente, en casos muy especiales, las *Ramificaciones de Cliques Proyectados* descritas en la sub-sección 3.3.2 no pueden eliminar las fracciones generadas por dos clusters cosechando una misma celda. A continuación describiremos el por qué de esta situación y mostraremos como las otras dos ramificaciones son capaces de solucionar este problema.

El hecho de que dos *clusters factibles* intersecten diferentes *cliques maximales* es esencial para que las *Ramificaciones de Cliques Proyectados* funcionen, pero en ciertos casos especiales puede suceder que dos clusters diferentes intersecten exactamente los mismos *cliques*. Esto sucede frecuentemente en las esquinas de las grillas. La siguiente figura muestra el grafo forestal de una grilla con un ejemplo de dos *clusters* que intersectan exactamente a los mismos *cliques*.

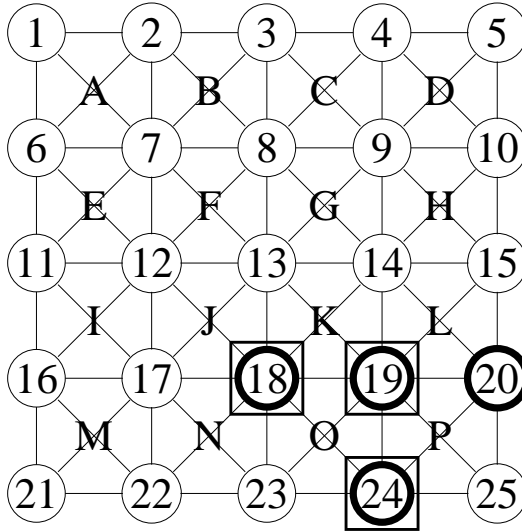


Figura C.1: Dos Clusters que Intersectan *Cliques* Idénticos

En la figura C.1 podemos ver dos *clusters* ( $S_1 = \{18, 19, 24\}$  y  $S_2 = \{18, 19, 20, 24\}$ ) que intersectan exactamente a los mismos *cliques* ( $\{J, K, L, N, O, P\}$ ). Si estos *clusters* estuvieran siendo cosechados fraccionalmente al mismo tiempo la *Ramificaciones de Cliques Projectados* no podría eliminar esa fracción.

Desde el punto de vista de la formulación las columnas asociadas estos *clusters* tendrán exactamente los mismos 1's en las filas asociadas a las restricciones de *cliques maximales*. Esto podría ser un problema para la formulación en un sólo periodo, pero no para la formulación para multiples periodos, ya que en este caso las filas asociadas a las restricciones de *celda/tiempo* distinguirían a las columnas.

En todos los casos en los que la *Ramificaciones de Cliques Projectados* tiene este problema es posible resolver la fracción utilizando una *Ramificación sobre holgura de restricciones Celda/Tiempo* definida en la sub-sección 3.3.3 o una *Ramificación Celdas/Tiempo* definida en la sub-sección 3.3.1. En el ejemplo de la figura C.1, es claro que si ambos clusters tienen asociados valores positivos en cierto periodo  $\hat{t}$  entonces estos valores serán fraccionarios. Por lo tanto, la *celda* 20 estará siendo cosechada fraccionalmente en el periodo  $\hat{t}$ . Si la *celda* 20 esta siendo cosechada también en otros periodos una *Ramificación Celdas/Tiempo* sobre ella podría eliminar

la fracción. En caso contrario bastaría una *Ramificación sobre holgura de restricciones Celda/Tiempo* sobre la *celda 20*.

Para demostrar que las ramificaciones de *Celdas/Tiempo* y de *holgura de restricciones Celda/Tiempo* siempre solucionarán los problemas de la ramificación *Cliques Projectados* nos bastará probar que cuándo las restricciones de *Cliques Projectados* fallan en un periodo  $t$  siempre podremos encontrar una *celda básica* que esta siendo cosechada parcialmente en ese periodo. A continuación haremos esa demostración.

Sea  $\hat{\mathbf{x}}$  una solución fraccionaria para la relajación lineal de cualquiera de los tres modelos en múltiples periodos (formulaciones 1.3, 1.4 o 2.1) y sean  $\hat{x}_{S,t}$  los valores de las variables  $x_{S,t}$  correspondientes. Supongamos que para cierto periodo  $\hat{t}$  existen dos *cluster factibles*  $S_1, S_2 \in \mathcal{C}$  cosechando una misma *celda*  $u \in V$ . Es decir:

$$\hat{x}_{S_1, \hat{t}}, \hat{x}_{S_2, \hat{t}} > 0 \wedge \exists u \in V \text{ tal que } u \in S_1, u \in S_2 \quad (\text{C.1})$$

Debido a las restricciones de *clusters incompatibles* (implementadas como restricciones de cliques proyectados) tendremos necesariamente que:

$$\hat{x}_{S_1, \hat{t}}, \hat{x}_{S_2, \hat{t}} < 1 \quad (\text{C.2})$$

y por lo tanto las variables  $x_{S_1, \hat{t}}$  y  $x_{S_2, \hat{t}}$  tendrán valores fraccionarios.

Supongamos además que los *clusters factibles*  $S_1$  y  $S_2$  intersectan exactamente a los mismos *cliques maximales*. Veamos ahora que necesariamente debe haber una *celda básica* que esta siendo cosechada parcialmente en el periodo  $\hat{t}$  por la solución  $\hat{\mathbf{x}}$ .

$S_1$  y  $S_2$  son *clusters* diferentes, por lo tanto, sin pérdida de generalidad podemos suponer que existe una *celda*  $v \in S_1$  tal que  $v \notin S_2$ . Si  $v$  solo esta siendo cosechada por el *cluster*  $S_1$  entonces  $v$  estará siendo cosechada parcialmente en el periodo  $t$ , pues:

$$\hat{x}_{S_1, \hat{t}} < 1 \quad (\text{C.3})$$

La única posibilidad de que  $v$  no este siendo cosechada parcialmente es que existan otros *clusters* que cosechen el *resto* de  $v$  no cosechado por  $S_1$ . Supongamos que estos *clusters*  $R_i$  existen, es decir:

$$\exists \{R_i\}_{i=1}^n \subseteq \mathcal{C} \text{ tal que } \hat{x}_{R_i, \hat{t}} > 0 \wedge v \in R_i \quad (\text{C.4})$$

Supongamos además que los *clusters*  $R_i$ 's y  $S_1$  son todos los que cumplen con las condiciones (C.4). De esta manera la cantidad total cosechada de la *celda*  $v$  en el periodo  $t$  sera:

$$\hat{x}_{S_1, \hat{t}} + \sum_{i=1}^n \hat{x}_{R_i, \hat{t}} \quad (\text{C.5})$$

Tenemos que  $v$  pertenece a algún *clique*  $K_v$  en el grafo forestal. Entonces todos los *clusters* que contengan a  $v$  intersectarán a  $K_v$ , en particular,  $S_1$  y los  $R_i$ 's intersectarán a  $K_v$ . Pero  $S_2$  intersecta exactamente a los mismos *cliques* que  $S_1$ , por lo tanto  $S_2$  también intersectará a  $K_v$ . Por lo tanto, la restricción de *cliques proyectados* asociada a  $K_v$  impondrá:

$$\hat{x}_{S_1, \hat{t}} + \sum_{i=1}^n \hat{x}_{R_i, \hat{t}} + \hat{x}_{S_2, \hat{t}} \leq 1 \quad (\text{C.6})$$

y como  $\hat{x}_{S_2, \hat{t}} > 0$  tendremos que la suma en C.6 es menor que 1:

$$\hat{x}_{S_1, \hat{t}} + \sum_{i=1}^n \hat{x}_{R_i, \hat{t}} < 1 \quad (\text{C.7})$$

es decir la  $v$  estará siendo cosechada parcialmente en el periodo  $t$ .

## Apéndice D

# Diagrama de Flujo de Método Basado en Restricciones Elásticas

La figura D.1 muestra un diagrama de flujo del procesamiento de nodos del árbol de *Branch & Bound* por el método basado en restricciones elásticas. El diagrama muestra cuándo se realizan los procedimientos de actualización de penalizaciones, ramificación por restricciones y la heurística *Integer Allocation*.

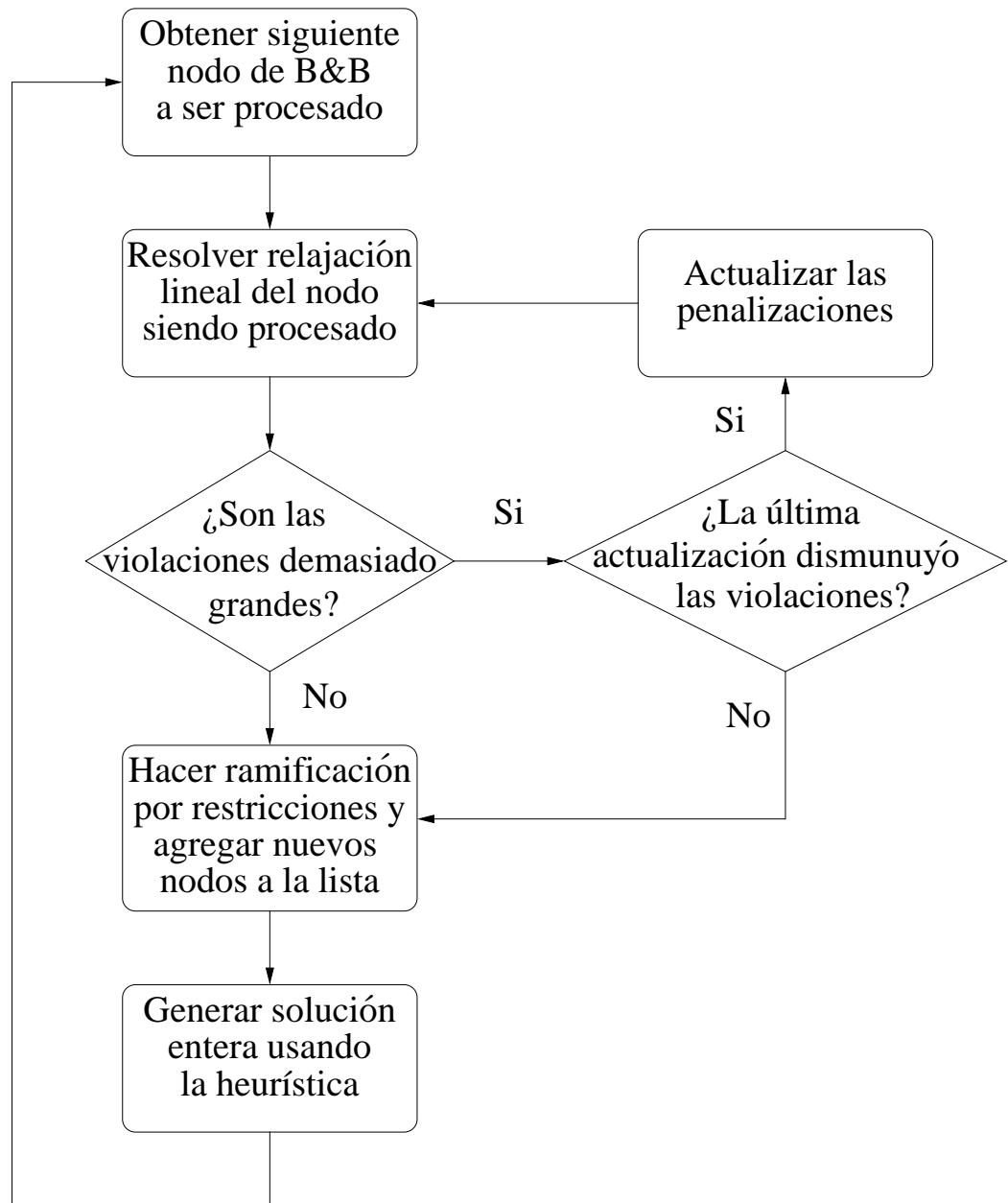


Figura D.1: Diagrama de Flujo del Método

## Apéndice E

# Ganancias Debido a Aceptar Violaciones Reales

Los siguientes resultados tienen como objetivo estudiar la ganancia obtenida al aceptar reales violaciones a las restricciones de volumen. Para esto utilizamos el modelo con restricciones elásticas con  $\Delta_E = \Delta$  y *sí* aceptamos soluciones con violaciones a las restricciones de volumen estrictas de nivel  $\Delta$ , siempre y cuándo estas sean menores que 1%.<sup>1</sup> Los *gap*'s presentados son calculados con respecto a el valor óptimo de la relajación lineal del modelo con restricciones de volumen estrictas de nivel  $\Delta$ .

El siguiente cuadro muestra resultados para las instancias de *El Dorado* con  $\Delta_E = \Delta = 0,15$ . El modelo elástico fue resuelto por *CPLEX* complementado con la ramificación por restricciones de la sección 3.3, la heurística de la sección 3.2 y la actualización dinámica de penalizaciones de la sección 3.4, de manera similar a los resultados de la sección 4.5.5.

Instancia	$\Delta$	Nodos <i>B&amp;B</i>	<i>gap</i> Mejor Sol. [%]	Tiempo Mejor Sol. [s]	1ra Sol. Bajo 1 % GAP [s]	Tiempo 1ra Sol. Factible [s]	<i>gap</i> 1ra Sol. Factible [%]
eldoradot12	0.15	14	0.00	7822	3750	3750	0.02
eldoradot15	0.15	12	-0.01	16142	4189	4189	0.02

Cuadro E.1: Ganancias al permitir violaciones reales de las restricciones de volumen

---

<sup>1</sup>Es decir, que las soluciones cumplan con las restricciones de volumen estrictas de nivel  $\Delta + 0,01$ .

Como era de esperar, al permitir verdaderas violaciones a las restricciones de volumen es posible encontrar soluciones con mejores valores objetivo. Sin embargo, las ganancias son moderadas y la velocidad en que son encontradas las soluciones es similar a cuándo se utiliza  $\Delta_E < \Delta$ . Notemos que el hecho de tener *gap*'s negativos es válido debido a que las soluciones no cumplen necesariamente con las restricciones de volumen estrictas de nivel  $\Delta$  por lo que pueden tener valores mayores que el óptimo de la relajación lineal con respecto a la cual se calculan los *gap*'s.

## Apéndice F

# Resultados Computacionales para Instancia con Primer Periodo Dominante

Como mencionamos previamente, dependiendo de la relación entre la *tasa de descuento* y la tasa de crecimiento de los árboles el primer o último periodo puede ser el más conveniente de cosechar dependiendo de los resultados. Todas las instancias estudiadas en esta memoria fueron consideradas con *tasas de descuento* que hacen que el último periodo sea el más conveniente de cosechar. Dependiendo de la región forestal estudiada, es posible que este supuesto no sea realista. En este apéndice estudiaremos el desempeño del método basado en restricciones elásticas sobre dos instancias modificadas del mapa *El Dorado*.

Se utilizó el mapa de *El Dorado* como base, pero esta vez se tomó una *tasa de descuento* que causara que el primer periodo fuera el más conveniente de cosechar.

El método basado en restricciones elásticas es *verdaderamente* útil cuando las restricciones de volumen son muy activas.<sup>1</sup> Por este motivo utilizamos una tasa de descuento exagerada de 50 % para asegurarnos que las restriccio-

---

<sup>1</sup>Decimos que las restricciones son *poco activas* cuando los valores duales asociados a ellas son pequeños y no existe una fuerte tendencia a cumplirlas con igualdad, por ejemplo cuando todos los periodos son relativamente igual de convenientes de cosechar. En este caso el efecto generador de fracciones de las restricciones será mínimo y es muy probable que el problema pueda ser resuelto directamente, es decir utilizando las restricciones estrictas originales.

nes de volumen fueran suficientemente activas como para justificar el uso de restricciones elásticas. Posiblemente *tasas de descuento* más realistas también den restricciones de volumen suficientemente activas, pero esto queda por ser estudiado (ver discusión acerca de aplicabilidad del modelo en la sección 5.2).

A continuación presentamos los resultados computacionales para dos instancias modificadas de *El Dorado* (con 12 y 15 periodos). La primera tabla presenta los resultados obtenidos al resolver la formulación con restricciones de volumen estrictas (formulación 1.4) directamente por *CPLEX*. La segunda tabla presenta los resultados obtenidos al resolver la formulación con restricciones de volumen elásticas (formulación 2.1) utilizando el método de *Branch & Bound* complementado con la heurística, la ramificación por restricciones y la actualización dinámica de las penalizaciones (De manera similar a los resultados de la sección 4.5.5).

El límite de tiempo utilizado fue de 4 horas.

Instancia	$\Delta$	Nodos <i>B&amp;B</i>	gap Mejor Sol. [%]	Tiempo Mejor Sol. [s]	1ra Sol. Bajo 1 % GAP [s]	Tiempo 1ra Sol. Factible [s]	gap 1ra Sol. Factible [%]
eldoradot12	0.15	903	1.38	11488	-	7221	1.85
eldoradot15	0.15	720	-	-	-	-	-

Cuadro F.1: Resultados para CPP-CLQ- $\mathcal{T}$ -VND con primer periodo dominante

Instancia	$\Delta$	Nodos <i>B&amp;B</i>	gap Mejor Sol. [%]	Tiempo Mejor Sol. [s]	1ra Sol. Bajo 1 % GAP [s]	Tiempo 1ra Sol. Factible [s]	gap 1ra Sol. Factible [%]
eldorado t12	0.15	25	0.94	4307	4307	2107	1.04
eldorado t15	0.15	10	9.85	7132	-	7132	9.85

Cuadro F.2: Resultados para CPP-CLQ- $\mathcal{T}$ -ECVND con primer periodo dominante

Podemos ver que nuevamente el método basado en restricciones elásticas tiene un mejor desempeño que *CPLEX* sobre el modelo estricto, incluso encontrando soluciones enteras factibles cuándo *CPLEX* no puede generar ninguna. Debemos notar que probablemente la dificultad de *CPLEX* para encontrar soluciones enteras se deba a la gran *tasa de descuento* utilizada

y que esto puede ser poco realista. Por lo tanto serán necesarias más pruebas computacionales antes de llegar a conclusiones generales acerca de la aplicabilidad del método a problemas reales.

## Apéndice G

# Definiciones

**Branch & Bound** Algoritmo utilizado para resolver problemas de programación entera basado en la resolución de relajaciones lineales del problema. [NemWol98]

**Branch & Cut** Extensión del método de *Branch & Bound* que incorpora la generación de *cortes fraccionarios*. [Cpx02, NemWol98]

**Clique Maximal** Dado un grafo  $G(V, E)$  un conjunto  $C \subseteq V$  es llamado *clique* si cada par de nodos en  $V$  esta unido por un arco en  $E$ . Un *clique* se dirá *maximal* si no esta incluido en ningún otro *clique*. [NemWol98]

**Cortes fraccionarios de Gomory** Tipo de *corte fraccionario* utilizado en la implementación de *Branch & Cut* de *CPLEX*. [Cpx02, NemWol98]

**gap** Medida de la calidad de una solución a un problema de optimización. Generalmente es una medida de la diferencia entre el valor objetivo de la solución y una cota superior del valor óptimo del problema (en el caso de la maximización).

**Simplex Dual** Algoritmo utilizado para resolver problemas de programación lineal. [NemWol98]

**Valor Presente Neto, Valor Presente y Tasa de Descuento** El valor presente o "valor actual de un pago futuro es la cantidad que habría que invertir hoy para producir exactamente ese pago en la fecha en que ha de realizarse." [FisDorSch89]. La tasa de descuento es la mejor tasa de interés disponible en caso de hacer dicha inversión.

$\mathbb{Q}$  y  $\mathbb{Q}_+$  Conjunto de números *racionales* y *racionales positivos* respectivamente.