



# **ESCUELA POLITÉCNICA NACIONAL**

## **FACULTAD DE CIENCIAS**

### **DEPARTAMENTO DE MATEMÁTICA**

#### **PARTICIONAMIENTO BALANCEADO DE HIPERGRAFOS EN UN NÚMERO FIJO DE COMPONENTES**

**TRABAJO DE TITULACIÓN PRESENTADO COMO REQUISITO PARA  
LA OBTENCIÓN DEL TÍTULO DE MAGISTER EN OPTIMIZACIÓN  
MATEMÁTICA**

**GUILLERMO ANDRÉS LANDÁZURI MEJÍA**

[guillermo.landazuri@epn.edu.ec](mailto:guillermo.landazuri@epn.edu.ec)

**DIRECTOR: DIEGO FERNANDO RECALDE CALAHORRANO**

[diego.recalde@epn.edu.ec](mailto:diego.recalde@epn.edu.ec)

**DMQ, SEPTIEMBRE 2023**

## **CERTIFICACIONES**

Yo, GUILLERMO ANDRÉS LANDÁZURI MEJÍA, declaro que el trabajo de titulación aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

---

Guillermo Andrés Landázuri Mejía

Certifico que el presente trabajo de integración curricular fue desarrollado por Guillermo Andrés Landázuri Mejía, bajo mi supervisión.

---

Diego Fernando Recalde Calahorrano  
**DIRECTOR**

## **DECLARACIÓN DE AUTORÍA**

A través de la presente declaración, afirmamos que el trabajo titulación aquí descrito, así como el(los) producto(s) resultante(s) del mismo, es(son) público(s) y estará(n) a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

Guillermo Andrés Landázuri Mejía

Diego Fernando Recalde Calahorrano

## **AGRADECIMIENTOS**

Agradezco a Diego Recalde, Ramiro Torres y Polo Vaca por toda su ayuda y conocimientos compartidos durante el desarrollo del proyecto. Así también, a todos los profesores y personal que forman parte de la maestría.

## **DEDICATORIA**

*A mis padres, por su ejemplo, cariño y apoyo incondicional.*

---

# Índice general

---

<b>1. Introducción</b>	<b>1</b>
<b>2. Definiciones y Notación</b>	<b>4</b>
2.0.1. Tipos de Hipergrafos . . . . .	5
2.0.2. Matrices de Hipergrafos . . . . .	6
2.0.3. Grafos asociados a un hipergrafo . . . . .	6
2.1. Métodos Exactos . . . . .	7
2.1.1. Programación Lineal Entera . . . . .	7
2.1.2. Planos Cortantes . . . . .	9
2.1.3. Branch & Bound . . . . .	10
2.2. Heurísticas . . . . .	14
2.2.1. Kernighan–Lin . . . . .	15
2.2.2. Fiduccia–Mattheyses . . . . .	15
2.2.3. Búsqueda Tabú . . . . .	16
2.3. Métodos de Solución . . . . .	16
<b>3. Planteamiento y Solución del Problema</b>	<b>18</b>
3.1. Particionamiento de Hipergrafos . . . . .	18
3.2. Variantes del Problema de Particionamiento de Hipergrafos .	18
3.3. Formulaciones . . . . .	19

3.3.1. Problema del k-Equiparticionamiento de Hipergrafos .	20
3.3.2. Problema de k-Particionamiento de Hipergrafos con Restricción de Tamaño Mínimo . . . . .	21
3.3.3. Problema de k-Particionamiento Balanceado de Hipergrafos . . . . .	21
3.3.4. Problema de k-Equiparticionamiento de Hipergrafos en Hiperárboles Lineales . . . . .	22
3.4. Planos Cortantes . . . . .	28
3.5. Heurística . . . . .	31
3.5.1. Heurística para Particionamiento Factible . . . . .	32
3.6. Aplicaciones . . . . .	34
<b>4. Pruebas Computacionales</b>	<b>36</b>
4.1. Generación de Instancias . . . . .	37
4.2. Análisis de Resultados . . . . .	37
4.2.1. Resultados formulación $\mathcal{F}_1$ . . . . .	38
4.2.2. Resultados formulación $\mathcal{F}_2$ . . . . .	41
4.2.3. Resultados formulación $\mathcal{F}_3$ . . . . .	44
4.2.4. Resultados formulación $\mathcal{F}_4$ y $\mathcal{F}_5$ . . . . .	47
4.2.5. Resultados Heurística . . . . .	50
<b>5. Conclusiones</b>	<b>52</b>
<b>Bibliografía</b>	<b>54</b>
<b>A. Tablas de Resultados</b>	<b>57</b>
A.1. Formulación $\mathcal{F}_1$ . . . . .	57
A.2. Formulación $\mathcal{F}_2$ . . . . .	63
A.3. Formulación $\mathcal{F}_3$ . . . . .	67
A.4. Formulación $\mathcal{F}_4$ y $\mathcal{F}_5$ . . . . .	72

---

## Índice de figuras

---

2.1. Ejemplo de hipergrafo. . . . .	4
2.2. Ejemplo de hiperárbol. . . . .	6
2.3. Hipergrafo y su grafo de intersección asociado. . . . .	7
2.4. Hipergrafo y su grafo de incidencia asociado. . . . .	7
2.5. Ejemplo de programa lineal entero. . . . .	8
2.6. Plano cortante. . . . .	10
2.7. Ejemplo Branch & Bound, problema original $S$ . . . . .	12
2.8. Ejemplo Branch & Bound, subproblemas $S_1$ y $S_2$ . . . . .	13
2.9. Ejemplo Branch & Bound, subproblemas $S_{11}$ y $S_{12}$ . . . . .	14
2.10. Árbol del método Branch & Bound. . . . .	14
3.1. Ejemplo de equiparticionamiento con $k = 2$ . . . . .	19
3.2. Construcción del grafo dirigido $D$ . . . . .	24
3.3. Grafo de incidencia asociado al hipergrafo. . . . .	24
3.4. Particionamiento en el modelo de flujo. . . . .	25
3.5. Ejemplo de particionamiento en componentes conexas. . . . .	27
3.6. Grafos del modelo de flujo de las soluciones factibles. . . . .	27
4.1. Resultados modelo $\mathcal{F}_1$ , $ e_{max}  = 2$ . . . . .	39
4.2. Resultados modelo $\mathcal{F}_1$ , $ e_{max}  = 5$ . . . . .	40



4.3. Resultados modelo $\mathcal{F}_1$ , $ e_{max}  = 10$ . . . . .	40
4.4. Resultados modelo $\mathcal{F}_1$ , $ e_{max}  = 20$ . . . . .	41
4.5. Resultados modelo $\mathcal{F}_2$ , $ e_{max}  = 2$ . . . . .	42
4.6. Resultados modelo $\mathcal{F}_2$ , $ e_{max}  = 5$ . . . . .	43
4.7. Resultados modelo $\mathcal{F}_2$ , $ e_{max}  = 10$ . . . . .	43
4.8. Resultados modelo $\mathcal{F}_2$ , $ e_{max}  = 20$ . . . . .	44
4.9. Resultados modelo $\mathcal{F}_3$ , $ e_{max}  = 2$ . . . . .	45
4.10 Resultados modelo $\mathcal{F}_3$ , $ e_{max}  = 5$ . . . . .	45
4.11 Resultados modelo $\mathcal{F}_3$ , $ e_{max}  = 10$ . . . . .	46
4.12 Resultados modelo $\mathcal{F}_3$ , $ e_{max}  = 20$ . . . . .	47
4.13 Resultados comparación modelos $\mathcal{F}_4$ y $\mathcal{F}_5$ , $ e_{max}  = 2$ . . . . .	48
4.14 Resultados comparación modelos $\mathcal{F}_4$ y $\mathcal{F}_5$ , $ e_{max}  = 5$ . . . . .	48
4.15 Resultados comparación modelos $\mathcal{F}_4$ y $\mathcal{F}_5$ , $ e_{max}  = 10$ . . . . .	49
4.16 Resultados comparación modelos $\mathcal{F}_4$ y $\mathcal{F}_5$ , $ e_{max}  = 20$ . . . . .	50

---

## Índice de tablas

---

4.1. Instancias para pruebas computacionales . . . . .	37
4.2. Resultados Heurística particionamiento factible . . . . .	50
A.1. Resultados computacionales modelo $\mathcal{F}_1$ . . . . .	57
A.2. Resultados computacionales modelo $\mathcal{F}_2$ . . . . .	63
A.3. Resultados computacionales modelo $\mathcal{F}_3$ . . . . .	68
A.4. Resultados computacionales modelo $\mathcal{F}_4$ y $\mathcal{F}_5$ . . . . .	72

---

## Índice de Algoritmos

---

1. Cubrimiento por hiperaristas. . . . .	32
2. Cubrimiento Factible-Biparticionamiento. . . . .	33
3. Generación de hipergrafos aleatorios . . . . .	38

## RESUMEN

En el presente trabajo se estudia el problema de particionamiento de hipergrafos en un número fijo de componentes. Los hipergrafos son la generalización de los grafos donde, en lugar de aristas, cada hiperarista puede conectar dos o más vértices. Dado un hipergrafo  $H = (V, E)$  se busca dividir su conjunto de vértices en  $k$  componentes disjuntas tal que cada vértice esté cubierto únicamente por hiperaristas contenidas completamente en alguna componente, a la vez que se minimiza el costo total de estas hiperaristas. Se proponen varias formulaciones de Programación Entera para los problemas de  $k$ -equiparticionamiento, particionamiento de tamaño mínimo, particionamiento balanceado y  $k$ -equiparticionamiento en hiperárboles lineales. Además, se presentan algunos tipos de desigualdades válidas a ser implementadas en las diferentes formulaciones. Finalmente, se discuten los resultados computacionales realizadas en las diferentes formulaciones para distintas instancias.

**Palabras clave:** particionamiento de hipergrafos, equiparticionamiento, desigualdades válidas, programación entera.

## ABSTRACT

In this paper the partitioning hypergraph problem in a fixed number of components is studied. Hypergraphs are the generalization of graphs where, unlike edges, each hyperedge can connect two or more vertices. Given a hypergraph  $H = (V, E)$  we seek to partition its vertex set into  $k$  disjoint components such that each vertex is covered by hyperedges completely contained in some component, while minimizing the cost of these hyperedges. Several Integer Programming formulations are proposed for the  $k$ -way equipartitioning, minimum size partitioning, balanced partitioning, and  $k$ -way equipartitioning in linear hyper-trees. Moreover, some types of valid inequalities are demonstrated and implemented for the different formulations. Finally, the computational results performed on the different formulations for different instances are discussed.

**Keywords:** hyperpgraph partitioning,  $k$ -way partitioning, valid inequalities, integer programming.

# Capítulo 1

---

## Introducción

---

Los hipergrafos son la generalización de los grafos en donde, a diferencia de las aristas, cada hiperarista del hipergrafo puede conectar dos o más vértices. El estudio de los hipergrafos se remonta a la década de los 60's [5], desde entonces, su estudio al igual que sus aplicaciones han tenido cada vez mayor relevancia en diferentes áreas. Uno de los motivos principales de este crecimiento se debe a la mayor capacidad de las hiperaristas de representar conectividad entre elementos y como consecuencia algunos problemas inviables en grafos resultan factibles en hipergrafos [3].

El problema de particionamiento de hipergrafos busca dividir al conjunto de vértices en subconjuntos disjuntos, normalmente sujetos a restricciones de balanceo. Este problema tiene un amplio número de aplicaciones en diferentes áreas. En diseño de circuitos integrados (VLSI) [14], el particionamiento ayuda a optimizar la distribución de los componentes para mejorar rendimientos y reducir costos de producción; en clasificación de datos [16] mejora el agrupamiento de datos (clustering) a través de algoritmos eficientes de clasificación; en computación científica [1] acelera operaciones matriciales a gran escala y en redes sociales [6] ayuda a balancear el manejo de datos. Adicionalmente, se puede generalizar el problema de particionamiento de grafos, el cual es un problema de la clase NP-hard [11] y decir que su extensión a hipergrafos es también un problema NP-hard.

El estudio del particionamiento de hipergrafos se remonta a la década de los 70's con la búsqueda de particionamientos de circuitos eléctricos que minimicen el número de conexiones entre elementos de diferentes subcircuitos. En [21], los autores explican que la estructura de grafos (conexión elemento–elemento) no representa adecuadamente las interconexiones en los circuitos eléctricos por lo que plantean un modelo que representa los circuitos a través de hiperaristas (descritas bajo el concepto de *nets*) y un algoritmo de particionamiento basado en el algoritmo de Kernighan-Linn. Alia and Maestrini [2] y Lawler [17] plantean la búsqueda de particionamientos óptimos a través de la búsqueda de subconjuntos de vértices que están más fuertemente relacionados entre ellos que con los vértices de su conjunto complementario, conocidos como conjuntos *LS* (Largest strong set).

El desarrollo de técnicas heurísticas para el problema de particionamiento ha sido ampliamente tratado en la literatura, en [16] se reportan varios de estos desarrollos entre los que se encuentran el algoritmo de Fiduccia–Mattheyses [9], recocido simulado, búsqueda tabú, métodos multinivel, entre otros. Además, en la literatura se puede encontrar diversas variantes del problema de particionamiento de hipergrafos. En [13] se estudia el problema del particionamiento balanceado de hipergrafos donde se presenta una mejora basada en flujos al esquema KaHyPar-MF. Frank et al. [10] presentan el problema de descomposición de un hipergrafo en  $k$  subhipergrafos generadores, el cuál se demuestra que es un problema NP completo para enteros  $k \geq 2$ . Así también, técnicas espectrales han sido utilizadas en el particionamiento de hipergrafos, Saito and Herbster [20] proponen un esquema general para una clase abstracta de matrices Laplacianas de un hipergrafo ( $p$ -Laplacianas) y un novedoso algoritmo de particionamiento de hipergrafos para su matriz  $p$ -Laplaciana generalizada.

Çatalyürek et al. [6] realizaron una recopilación de los avances más recientes en el particionamiento de grafos e hipergrafos, donde se enfoca el desarrollo y mejora de las técnicas para su resolución como son métodos multinivel, metaheurísticas y métodos paralelos, con aplicaciones en redes neuronales, procesamiento en paralelo o segmentación de imágenes, tal como presenta Dai and Gao [8].

En el presente trabajo el objetivo del problema de particionamiento es dividir el conjunto de vértices en un número fijo de componentes, tal que cada vértice esté cubierto exclusivamente por hiperaristas cuyos nodos se encuentren en la misma componente, y que se minimice el costo total de las hiperaristas contenidas en cada componente. Se han desarrollado formulaciones enteras para los problemas de  $k$ -equiparticionamiento, particionamiento con restricción de tamaño mínimo, particionamiento balanceado y  $k$ -equiparticionamiento en hiperárboles lineales.

El documento está constituido de 4 partes principales: en el Capítulo 2 se incluyen las definiciones y notaciones sobre hipergrafos a ser utilizadas en el desarrollo del trabajo y se realiza una revisión bibliográfica de la metodología utilizada en la Programación Entera y Programación Entera Mixta, en el Capítulo 3 se plantean las variantes del problema de particionamiento de hipergrafos y se presentan las formulaciones de los diferentes problemas, en el Capítulo 4 se discuten los resultados de las pruebas computacionales de los modelos propuestos para diferentes instancias de hipergrafos y las diferentes implementaciones de mejora, por último en el Capítulo 5 se presentan las conclusiones del trabajo.



# Capítulo 2

---

## Definiciones y Notación

---

En este capítulo se presentan las definiciones básicas y la notación de hipergrafos que serán utilizadas en el presente trabajo. Para un número  $n \in \mathbb{N}$ , el conjunto  $\{1, 2, \dots, n\}$  se notará por  $[n]$ .

**Definición 2.1.** Un hipergrafo es un par  $H = (V, E)$  donde,  $V = \{1, 2, \dots, n\}$  es el conjunto de vértices y  $E = \{e_1, e_2, \dots, e_m\}$  el conjunto de hiperaristas tal que  $e_j \subset V$  para todo  $e_j \in E$ .

En la Figura 2.1 se presenta un ejemplo de hipergrafo.

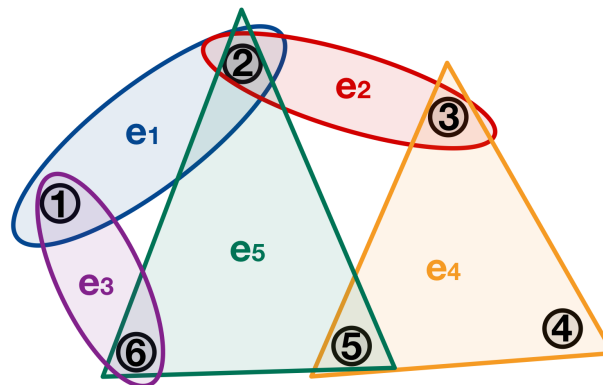


Figura 2.1: Ejemplo de hipergrafo.

**Definición 2.2.** Sea el vértice  $i \in V$ ,  $|E(i)|$  es la cardinalidad de  $i$ , donde  $E(i) := \{e \in E : i \in e\}$  es el conjunto de hiperaristas que contienen al vértice  $i$ . La cardinalidad de  $i$  también se conoce como el grado del nodo  $i$ .

**Definición 2.3.** Sea la hiperarista  $e \in E$  y  $V(e) := \{v \in V : v \in e\}$  el conjunto de vértices que pertenecen a la hiperarista  $e$ . Entonces,  $|V(e)|$  o simplemente  $|e|$  representa el tamaño de la hiperarista  $e$ . Además, para un subconjunto  $S \subset E$ , se define  $V(S) = \cup_{e_j \in S} V(e_j)$  como el conjunto de vértices contenidos en las hiperaristas de  $S$ .

**Definición 2.4.** Sean los nodos  $a, b \in V$ , un camino es una sucesión de hiperaristas  $(e_1, \dots, e_q)$  tal que  $a \in e_1$ ,  $b \in e_q$  y  $e_{\ell-1} \cap e_\ell \neq \emptyset$  para  $\ell = 2, \dots, q$ . Un ciclo es un camino con  $q \geq 2$  en donde  $a = b$ .

### 2.0.1. Tipos de Hipergrafos

Al hipergrafo  $H = (V, E)$ , se lo conoce como:

- Vacío, si  $V = \emptyset$  y  $E = \emptyset$ .
- Trivial, si  $V \neq \emptyset$  y  $E = \emptyset$ .
- Simple, si no contiene hiperaristas repetidas, es decir,  $e_i \subseteq e_j$  si y solo si  $i = j$ .
- Conexo, si para todo par de vértices  $i, j \in V$  existe un camino de  $i$  a  $j$ .
- Lineal, si es simple y  $|e_i \cap e_j| \leq 1$  para todo  $e_i, e_j \in E$  y  $i \neq j$ .
- $k$ -regular, cuando todos los vértices son de grado  $k$ .
- $r$ -uniforme, cuando todas las hiperaristas son de cardinalidad  $r$ .

**Definición 2.5.** Un vértice  $i \in V$  es una hoja si está contenido en una única hiperarista  $e \in E$ . Es decir,  $E(i) = \{e\}$  para algún  $e \in E$ .

**Definición 2.6.** Un hiperárbol es un hipergrafo conexo que no contiene ciclos.

En la Figura 2.2 se representa un hiperárbol. Además, las hiperaristas  $e_1, e_3$  son hojas del hiperárbol.

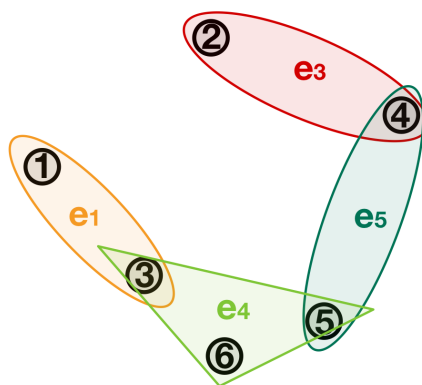


Figura 2.2: Ejemplo de hiperárbol.

### 2.0.2. Matrices de Hipergrafos

Para un hipergrafo  $H = (V, E)$ .

1. Matriz de incidencia: es la matriz  $A \in \mathbb{R}^{n \times m}$  tal que

$$a_{i,j} = \begin{cases} 1 & , \text{si } i \in e_j \\ 0 & , \text{caso contrario} \end{cases} , \forall i \in V, \forall e_j \in E$$

2. Matriz de adyacencia: es la matriz cuadrada  $B \in \mathbb{R}^{n \times n}$  tal que

$$b_{x,y} = |\{e \in E : x, y \in e\}|, \forall x, y \in V, x \neq y$$

y  $b_{x,x} = 0$ . Es decir, muestra el número de hiperaristas que contienen a ambos vértices  $x, y$ .

### 2.0.3. Grafos asociados a un hipergrafo

Al poder representar un hipergrafo a través de la estructura de grafos se abre la posibilidad de analizar y resolver problemas de hipergrafos con métodos utilizados en grafos. A continuación, se presentan dos tipos de grafos asociados a los hipergrafos.

**Definición 2.7.** El grafo de intersección asociado a un hipergrafo  $H = (V, E)$  es el grafo  $G = (V', E')$ , tal que  $V' = E$  y  $E' := \{i, j : e_i, e_j \in E, e_i \cap e_j \neq \emptyset\}$ .

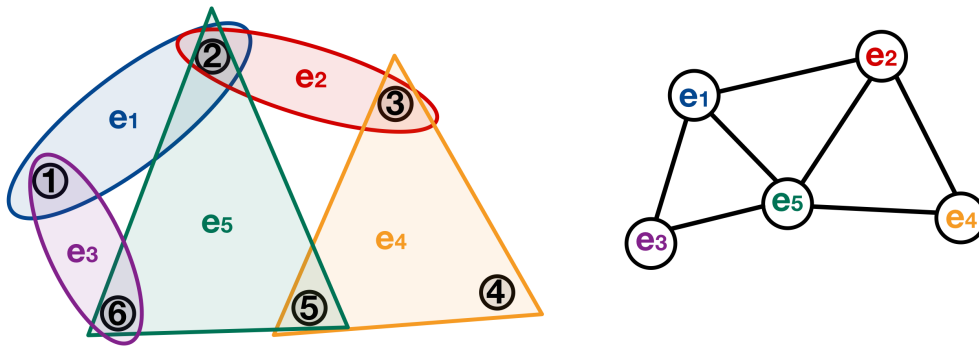


Figura 2.3: Hipergrafo y su grafo de intersección asociado.

**Definición 2.8.** El grafo de incidencia asociado a un hipergrafo  $H = (V, E)$  es el grafo bipartito  $G = (V \cup E, E')$ , tal que  $E' := \{i, j : i \in V, e_j \in E, i \in e_j\}$ .

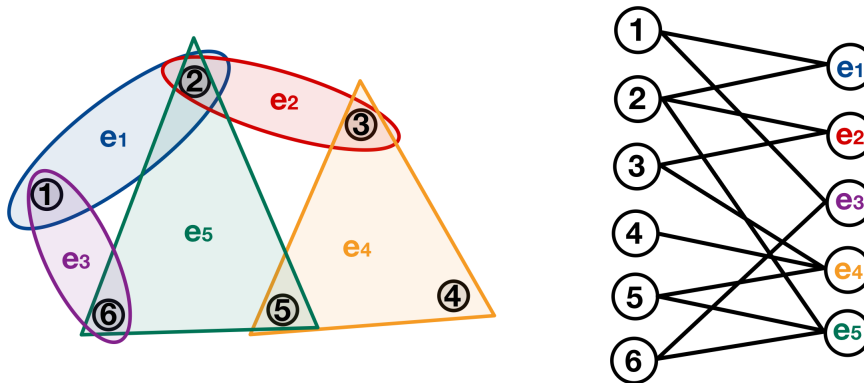


Figura 2.4: Hipergrafo y su grafo de incidencia asociado.

## 2.1. Métodos Exactos

### 2.1.1. Programación Lineal Entera

En un Programa Lineal (LP) se busca encontrar los valores de variables continuas  $(x_1, x_2, \dots, x_n)$  que minimicen o maximicen una función objetivo lineal  $z$ , a la vez que satisfacen un conjunto de restricciones lineales. Un Programa Lineal Entero (IP) se diferencia de un Programa Lineal en que los valores de sus variables deben ser enteros y se lo puede expresar de la siguiente forma:

$$\text{máx } z = c^T x$$

sujeto a:

$$Ax \leq b$$

$$x \in \mathbb{Z}^n$$

donde  $A$  es una matriz real de  $n \times m$ ,  $b \in \mathbb{R}^m$  y  $c \in \mathbb{R}^n$ . Además, dado que un problema de programación lineal entera es de la clase NP-hard [19], su dificultad es mucho mayor que la de un Programa Lineal.

Cuando un Programa Lineal Entero está conformado tanto por variables enteras como continuas, se lo conoce como un Programa Lineal Entero Mixto (MIP). En un Programa Lineal Entero, si no se considera la restricción de integralidad de sus variables obtenemos la *relajación lineal del programa entero*. De la resolución de esta relajación lineal (por ejemplo, con el método simplex) se obtiene información sobre cotas para el valor óptimo de la función objetivo y/o soluciones factibles, que puede ayudar en la resolución del IP.

En el siguiente ejemplo se muestra un Programa Lineal Entero en dos dimensiones y su relajación lineal:

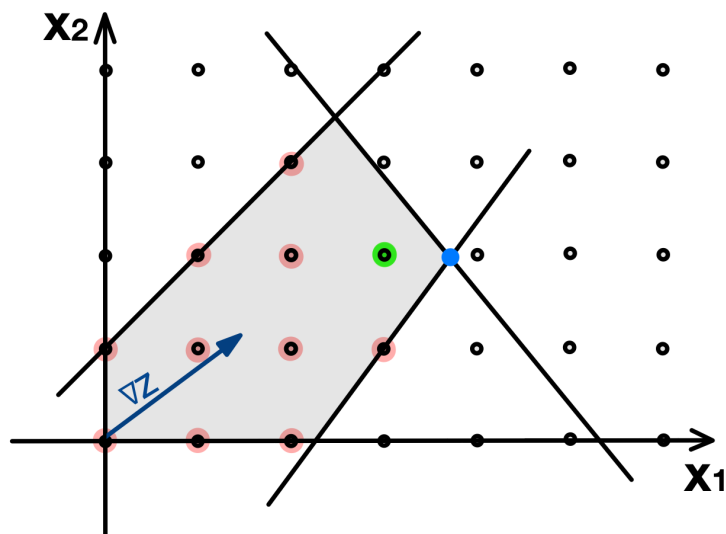


Figura 2.5: Ejemplo de programa lineal entero.

En la Figura 2.5 las soluciones factibles del programa entero se representan con los puntos en color rojo, la solución óptima del IP es el punto en color verde, el área sombreada representa la región de soluciones factibles de la relajación del IP y la solución óptima de la

relajación es el punto en color azul (la cuál difiere de la solución óptima del programa entero).

Para la resolución de los programas lineales enteros se cuenta con algunas técnicas, de las cuales la más utilizada es el método Branch & Cut. Este método, a su vez, combina las técnicas de planos cortantes y Branch & Bound.

### **2.1.2. Planos Cortantes**

La idea principal de este método es reducir el tamaño de la región factible de la relajación lineal de un programa entero a través de la inclusión de nuevas restricciones que eliminen parte de las soluciones factibles de la relajación lineal y sin excluir ninguna de las soluciones factibles del IP. El procedimiento de este método sigue los siguientes pasos principales: resolver la relajación del programa entero, si la solución óptima de la relajación lineal es una solución entera se resolvió el IP. Caso contrario, se añade una restricción lineal a la relajación lineal anterior y se repite el proceso.

Consideremos el programa lineal entero de la Figura 2.5, al añadir una nueva restricción lineal según lo descrito anteriormente, se reduce la región factible del modelo relajado y así también su solución óptima es más próxima a la solución óptima del programa entero. En la Figura 2.6, se ejemplifica la mejora que produce la adición de planos cortantes en el programa relajado del programa lineal. Así, si se continúa añadiendo adecuadamente más restricciones al programa relajado, se llega al óptimo del programa lineal entero.

### **Desigualdades Válidas**

Una desigualdad lineal es *válida* para un programa entero o entero mixto si se satisface para todas las soluciones factibles. Dada una solución  $x^*$  de la relajación lineal de un programa entero, lo que se busca es una familia de desigualdades válidas que separen a la solución  $x^*$  de la envolvente convexa del conjunto de soluciones factibles del programa entero. Dichas desigualdades válidas son conocidas como cortes y este

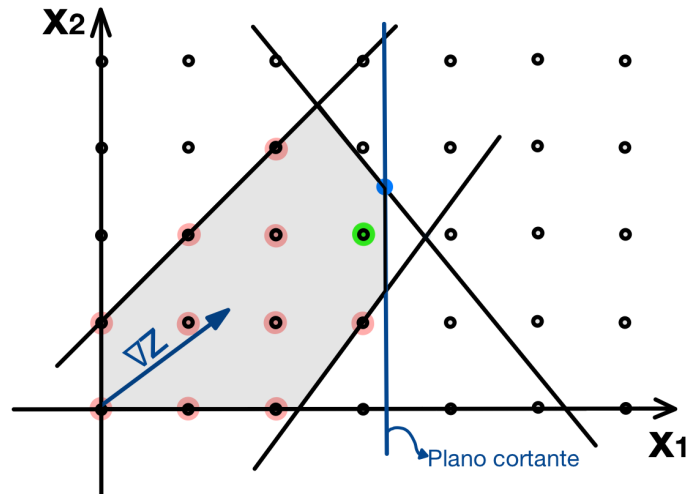


Figura 2.6: Plano cortante.

procedimiento es conocido como el problema de separación.

Una de las formas de clasificar los tipos de desigualdades válidas se basa en el tipo de problema MIP [7]:

- Tipo 1. Son desigualdades válidas generadas de programas enteros y enteros mixtos sin alguna estructura especial. Pueden ser aplicadas a todo IP o MIP pero los cortes derivados de ellas son débiles.
- Tipo 2. Proviene de problemas con alguna estructura local considerando una sola restricción. En este grupo se incluyen las desigualdades tipo knapsack.
- Tipo 3. Se generan de conjuntos de restricciones con estructuras especiales como restricciones del problema de flujo en redes con cargo fijo y problema de localización.

### 2.1.3. Branch & Bound

Este método consiste en dividir el problema entero en una serie de subproblemas más manejables para resolver y que dividen la región de soluciones factibles sin descartar ninguna de ellas. Este proceso tiene tres pasos principales:

1. Ramificación: se divide el problema en dos subproblemas lineales enteros, en cada uno de los subproblemas se añade una nueva restricción.
2. Acotamiento: al resolver los subproblemas generados se obtiene nuevas cotas para el valor óptimo de la función objetivo del problema entero.
3. Podado: se descarta un subproblema generado según alguno de los tres criterios por infactibilidad, optimalidad o acotamiento.

El proceso de resolución por el método de Branch & Bound se lo representa comúnmente a través de un árbol que se va ramificando de arriba hacia abajo. Se empieza en un nodo raíz resolviendo la relajación del problema entero ( $S$ ), si su solución óptima  $x^*$  es una solución factible del problema entero, este está resuelto, caso contrario, el valor de la función objetivo de la relajación lineal se convierte en una cota superior (en el caso de maximización) del valor óptimo de la función objetivo del problema entero. Se escoge alguna variable  $x_i^* \notin \mathbb{Z}$  y se ramifica el nodo en dos subproblemas (ramas). El primer subproblema ( $S_1$ ) se lo obtiene añadiendo la restricción  $x_i \leq \lfloor x_i^* \rfloor$  y el segundo añadiendo la restricción  $x_i \geq \lfloor x_i^* \rfloor + 1$ . Así se logra con cada uno de estos subproblemas reducir la región de soluciones factibles del problema relajado y dividir la región de soluciones factibles del problema entero sin eliminar ninguna ellas. De la solución de la relajación lineal en cada nodo se conocerá si dicho nodo necesita seguir ramificándose, ser podado o se ha obtenido una mejor cota superior de la función objetivo del problema original.

Existen tres casos por los cuales un nodo puede ser podado: la relajación de su subproblema no tiene solución factible, su subproblema tiene una solución (entera) del problema original y la cota superior de la relajación es menor o igual a la cota inferior del problema original. A estos casos se los conoce como poda por infactibilidad, optimalidad y acotamiento, respectivamente. Si un nodo es podado por optimalidad, su solución óptima sirve para aumentar el valor de la cota inferior del problema original.

Para un mejor entendimiento de este método se describe a continuación un ejemplo de su uso para la resolución de un problema



lineal entero: **Ejemplo 2.1**

$$S = \begin{cases} \text{máx} & 4x_1 + x_2 \\ \text{s.a. :} & x_1 + 5x_2 \leq 15 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1, x_2 \geq 0 \\ & x_1, x_2 \in \mathbb{Z} \end{cases}$$

Comenzamos resolviendo la relajación del problema original  $S$  para obtener una primera cota superior.

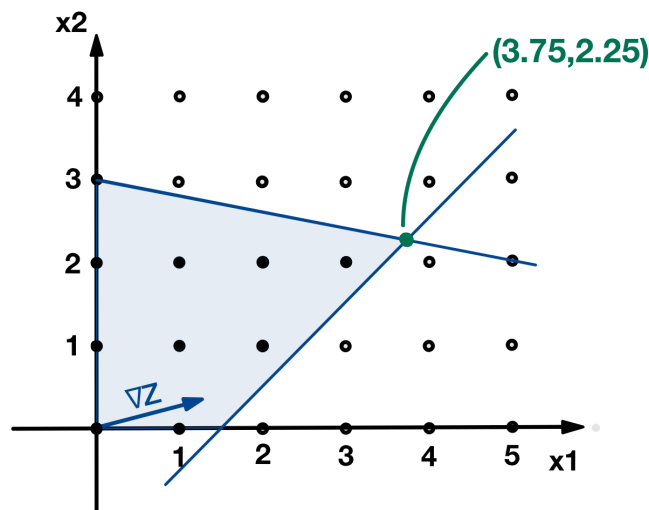


Figura 2.7: Ejemplo Branch & Bound, problema original  $S$ .

La solución óptima de la relajación lineal es  $x^* = (3.75, 2.25)$  con un valor de la función objetivo  $z = 17.25$ , por lo que la nueva cota superior del problema entero es 17.25. Como  $x^* \notin \mathbb{Z}^2$  se debe ramificar, se escoge arbitrariamente ramificar en  $x_1$  creando dos nuevos subproblemas:  $S_1$  añadiendo la restricción  $x_1 \leq 3$  y  $S_2$  añadiendo la restricción  $x_1 \geq 4$ .

$$S_1 = \begin{cases} \text{máx} & 4x_1 + x_2 \\ \text{s.a. :} & x_1 + 5x_2 \leq 15 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1 \leq 3 \\ & x_1, x_2 \geq 0 \end{cases} \quad S_2 = \begin{cases} \text{máx} & 4x_1 - x_2 \\ \text{s.a. :} & x_1 + 5x_2 \leq 15 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1 \geq 4 \\ & x_1, x_2 \geq 0 \end{cases}$$

Se puede ver en la Figura 2.8 que el subproblema  $S_2$  es infactible por

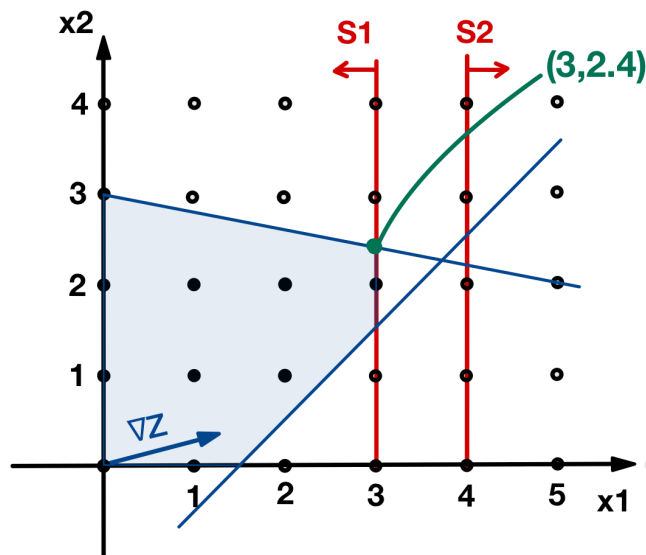


Figura 2.8: Ejemplo Branch & Bound, subproblemas  $S_1$  y  $S_2$ .

lo que se lo poda por infactibilidad y se procede a resolver la relajación lineal del problema  $S_1$ . La solución óptima de la relajación de  $S_1$  es  $x^* = (3, 2.4)$ , con un valor de la función objetivo  $z = 14.4$ . Entonces la nueva cota superior es 14.4. Ya que  $x_1^* \in \mathbb{Z}$  y  $x_2^* \notin \mathbb{Z}$ , se procede a ramificar en  $x_2$  al problema  $S_1$ .

El problema  $S_1$  se divide en los subproblemas:  $S_{11}$  añadiendo la restricción  $x_2 \geq 3$  y  $S_{12}$  añadiendo la restricción  $x_2 \leq 2$ .

$$S_{11} = \begin{cases} \text{máx} & 4x_1 + x_2 \\ \text{s.a. :} & x_1 + 5x_2 \leq 15 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1 \leq 3 \\ & x_2 \geq 3 \\ & x_1, x_2 \geq 0 \end{cases} \quad S_{12} = \begin{cases} \text{máx} & 4x_1 + x_2 \\ \text{s.a. :} & x_1 + 5x_2 \leq 15 \\ & 2x_1 - 2x_2 \leq 3 \\ & x_1 \leq 3 \\ & x_2 \leq 2 \\ & x_1, x_2 \geq 0 \end{cases}$$

Se puede observar en la Figura 2.9 que el subproblema  $S_{11}$  es infactible por lo que se lo poda por infactibilidad y se procede a resolver la relajación lineal del problema  $S_{12}$ . La solución óptima de la relajación lineal es  $x^* = (3, 2)$ , con un valor de la función objetivo  $z = 14$ . Ya que la solución óptima de la relajación es una solución factible del problema entero, 14 se vuelve la nueva cota inferior del problema entero cuyo valor inicial era  $-\infty$  y

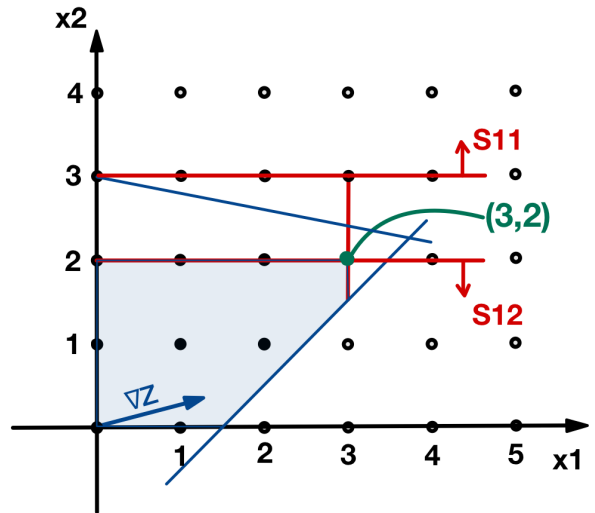


Figura 2.9: Ejemplo Branch & Bound, subproblemas  $S_{11}$  y  $S_{12}$ .

$(3, 2)$  se vuelve candidato a solución óptima del problema entero.

Se revisa el árbol generado con este método, representado en la Figura 2.10, al ya estar evaluados todos los nodos,  $(3, 2)$  es la solución del problema entero con un valor de la función objetivo  $z = 14$ .

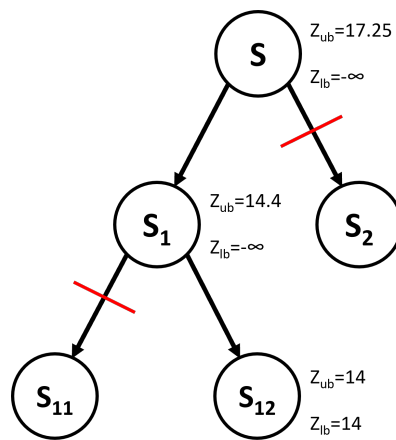


Figura 2.10: Árbol del método Branch & Bound.

## 2.2. Heurísticas

Una heurística es una técnica de resolución diseñada para encontrar una buena solución (no necesariamente óptima) a un problema lineal

entero o mixto, de manera fácil y rápida. Estas soluciones aproximadas pueden servir, por ejemplo, como puntos de partida en la resolución del problema entero a través del método de Branch & Bound reduciendo el tamaño de búsqueda en los nodos del árbol donde se tenga soluciones peores a la inicial. Existen tres enfoques de las heurísticas que han sido exitosos en la resolución de programas lineales mixtos [7]: búsqueda local, búsqueda tabú y algoritmos genéticos.

A continuación se presentan algunas de las heurísticas más desarrolladas y exitosas utilizadas en la resolución del problema de particionamiento de hipergrafos. Como ocurre en la mayoría de literatura de particionamiento, la función objetivo del problema es la minimización del número total de hiperaristas en el corte o del costo total de hiperaristas en el corte, por lo que el desarrollo de heurísticas tiene el mismo enfoque.

### 2.2.1. Kernighan–Lin

Es una heurística de búsqueda local donde se empieza con una solución factible dada. Kernighan y Linn [15] desarrollaron este algoritmo para el biparticionamiento de grafos basado en el intercambio de pares de vértices entre las componentes del particionamiento inicial cuya *ganancia* sea mayor. Definen una función de ganancia entre un par de vértices  $x, y$  como el número de hiperaristas en el corte (hiperaristas con vértices en más de una componente) que se reducirían al intercambiar los vértices  $x, y$  entre las componentes  $V_1$  y  $V_2$  del particionamiento. Esta función está dada por:

$$g(x, y) = \left( \sum_{i \notin V_1} a_{xi} - \sum_{j \in V_1} a_{xj} \right) + \left( \sum_{i \notin V_2} a_{yi} - \sum_{j \in V_2} a_{yj} \right) - 2a_{xy} \quad (2.1)$$

donde  $a_{ij}$  son las entradas de la matriz de adyacencia del hipergrafo.

### 2.2.2. Fiduccia–Mattheyses

Es una heurística con tiempo lineal por pase [9] utilizada para el biparticionamiento ( $k = 2$ ) de hipergrafos. A partir de un

biparticionamiento inicial se define la ganancia  $g(i)$  para cada vértice  $i$  del hipergrafo como el número de hiperaristas en el corte que se reducirían al mover el vértice  $i$  de una componente a otra. Una vez se haya movido un vértice a la otra componente, este se bloquea por el resto del pase y se recalcula la ganancia de los vértices en el vecindario de  $i$ . Este conjunto se forma por los vértices que comparten hiperaristas con  $i$  y que estas hiperaristas estén completamente contenidas en la componente inicial de  $i$ . Los vértices se moverán entre las componentes del biparticionamiento hasta que se bloqueen o se cumpla una condición de balanceo entre las componentes.

### **2.2.3. Búsqueda Tabú**

Es una heurística propuesta por Glover [12], se ha aplicado a problemas de programación entera, calendarización, particionamiento y coloración de grafos. Es un procedimiento iterativo que a partir de una solución factible inicial busca una mejor solución en la forma de un algoritmo de descenso más profundo. Se caracteriza por su habilidad de escapar de óptimos locales usando una memoria de corto plazo de soluciones recientes evitando que se mueva a soluciones que han sido visitadas recientemente (lista tabú).

Hay diferentes atributos del problema que se pueden usar para crear la memoria a corto plazo de las listas tabú. En el problema de particionamiento una posibilidad es identificar los atributos de un movimiento basado en el cambio de un vértice de un bloque al otro.

## **2.3. Métodos de Solución**

Para la resolución de los modelos planteados para los diferentes problemas se utilizará el método de *Branch & Cut*, el cual consiste en una combinación del método de Branch & Bound con la adición de planos cortantes en los nodos del árbol de búsqueda. La adición de estos planos ayudará a obtener mejores cotas del valor óptimo de la función objetivo en la resolución de los subproblemas antes de seguir ramificando o podando. Ya que en cada nodo se añaden desigualdades

válidas, el método puede resultar más costoso computacionalmente por lo que es recomendable determinar un número adecuado de planos añadidos en cada nodo según las características del modelo a resolver.

Adicionalmente, se utilizará una heurística desarrollada que permita encontrar una solución inicial factible del problema de particionamiento para  $k = 2$ . La solución factible de la heurística servirá dentro del método Branch & Cut para reducir el número de nodos explorados y así mejorar la eficiencia del método.

# Capítulo 3

---

## Planteamiento y Solución del Problema

---

### 3.1. Particionamiento de Hipergrafos

Sea  $V$  el conjunto de vértices del hipergrafo  $H$  y  $k \geq 2$  un número entero fijo, el particionamiento o  $k$ -particionamiento del hipergrafo  $H$  consiste en dividir su conjunto  $V$  en  $k$  componentes tal que  $V = V_1 \cup V_2 \cup \dots \cup V_k$  y que sus componentes sean conjuntos disjuntos, es decir,  $V_i \cap V_j = \emptyset$  para todo  $i, j \in [k]$ ,  $i \neq j$ . Cada vértice de  $V$  debe estar cubierto por al menos una hiperarista cuyos vértices se encuentren en la misma componente, estas hiperaristas son las que forman parte del particionamiento. Se define al corte como el conjunto de hiperaristas que contienen vértices en más de una componente del particionamiento.

### 3.2. Variantes del Problema de Particionamiento de Hipergrafos

En una  $k$ -equipartición, las  $k$  componentes poseen el mismo número de vértices, por lo que es condición necesaria que  $n = |V|$  sea divisible para  $k$ . En la Figura 3.1 se ejemplifica el  $k$ -equiparticionamiento de un hipergrafo:

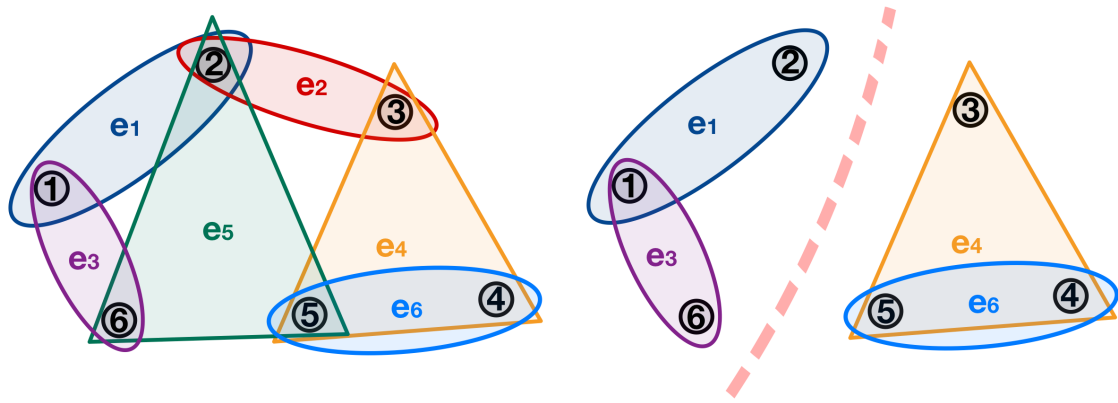


Figura 3.1: Ejemplo de equiparticionamiento con  $k = 2$ .

Se tiene al hipergrafo  $H = (V, E)$ , tal que  $V = \{1, 2, 3, 4, 5, 6\}$  y  $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ . Se propone un equiparticionamiento  $V = V_1 \cup V_2$  donde  $V_1 = \{1, 2, 6\}$  y  $V_2 = \{3, 4, 5\}$ , por lo tanto, las hiperaristas que forman parte de este particionamiento serían  $e_1, e_3 \in V_1$  y  $e_4, e_6 \in V_2$ , mientras que las hiperaristas  $e_2, e_5$  pertenecen al corte. Un  $k$ -equipartición en hiperárboles lineales es una  $k$ -equipartición donde cada componente es un hiperárbol lineal.

Una partición de tamaño mínimo es una partición donde se impone una cota inferior  $\alpha$  al tamaño de cada componente. Es decir,  $n/k \geq |V_i| \geq \alpha$  para todo  $i \in [k]$ , con  $\alpha$  un entero positivo.

Dada una función de costo sobre los vértices del hipergrafo, en una partición balanceada el costo total de los vértices de cada componente posee cotas inferior y superior.

### 3.3. Formulaciones

Para las diferentes formulaciones se considera un hipergrafo  $H = (V, E)$  con  $V = \{1, 2, \dots, n\}$  el conjunto de vértices,  $E = \{e_1, e_2, \dots, e_m\}$  el conjunto de hiperaristas y un particionamiento  $\{V_1, V_2, \dots, V_k\}$  para  $k \geq 2$  entero. Se define una función de costo sobre las hiperaristas  $w : E \mapsto \mathbb{R}^+$ . En el presente trabajo, la función objetivo del problema es minimizar el costo total de las hiperaristas contenidas completamente en alguna componente del particionamiento.



### 3.3.1. Problema del $k$ -Equiparticionamiento de Hipergrafos

Este problema consiste en encontrar una  $k$ -equipartición tal que se minimice el costo total de las hiperaristas que conforman cada componente del particionamiento. La formulación consta de dos conjuntos de variables binarias:  $x_i^c$  asociada a los vértices del hipergrafo y  $y_j^c$  asociada a las hiperaristas, ambos conjuntos de variables a su vez están asociados a las componentes del particionamiento. Sea  $x_i^c$  la variable que vale 1 si el vértice  $i \in [n]$  pertenece a la componente  $c \in [k]$  y cero caso contrario. De manera similar,  $y_j^c$  es la variable que vale 1 si la hiperarista  $j \in [m]$  pertenece completamente a la componente  $c \in [k]$  y cero caso contrario. Se cuenta como parámetro a la matriz de incidencia del hipergrafo  $A \in \mathbb{R}^{n \times m}$ , como se definió en la Sección 2.0.2. Entonces, el problema del  $k$ -Equiparticionamiento de Hipergrafos se formula como se presenta a continuación ( $\mathcal{F}_1$ ):

$$\text{mín} \sum_{c \in [k]} \sum_{e_j \in E} w_j y_j^c \quad (3.1)$$

s.a.

$$\sum_{c \in [k]} x_i^c = 1, \quad \forall i \in V, \quad (3.2)$$

$$\sum_{i \in V} x_i^c = \frac{n}{k}, \quad \forall c \in [k], \quad (3.3)$$

$$x_i^c \leq \sum_{e_j \in E} a_{ij} y_j^c, \quad \forall i \in V, \forall c \in [k], \quad (3.4)$$

$$\sum_{c \in [k]} y_j^c \leq 1, \quad \forall e_j \in E, \quad (3.5)$$

$$\sum_{i \in V} a_{ij} x_i^c \geq |e_j| y_j^c, \quad \forall e_j \in E, \forall c \in [k], \quad (3.6)$$

$$x_i^c \in \{0, 1\}, \quad \forall i \in V, \forall c \in [k], \quad (3.7)$$

$$y_j^c \in \{0, 1\}, \quad \forall e_j \in E, \forall c \in [k] \quad (3.8)$$

La función objetivo (3.1) busca minimizar el costo de las hiperaristas que pertenecen a las componentes del particionamiento. Las restricciones (3.2) aseguran que cada vértice pertenezca a una sola componente,

las restricciones (3.3) indican que cada componente debe contener exactamente  $n/k$  vértices. Las restricciones (3.4) nos indican que si el vértice  $i$  pertenece a la componente  $c$ , este debe estar contenido en al menos una hiperarista perteneciente a la componente  $c$ . Las restricciones (3.5) y (3.6) imponen que cada hiperarista puede formar parte de a lo más una componente del particionamiento y que todos los vértices que componen dicha hiperarista deben pertenecer a la misma componente, respectivamente.

### 3.3.2. Problema de $k$ -Particionamiento de Hipergrafos con Restricción de Tamaño Mínimo

En este problema se busca un  $k$ -particionamiento en donde a cada componente se impone una restricción de tamaño mínimo. Se plantea la Formulación  $\mathcal{F}_2$  para el problema del  $k$ -Particionamiento de Hipergrafos con Restricción de tamaño mínimo en base a la Formulación  $\mathcal{F}_1$ , reemplazando las restricciones (3.3) por el siguiente conjunto de restricciones:

$$\sum_{i \in V} x_i^c \geq \alpha, \quad \forall c \in [k], \quad (3.9)$$

donde el número entero  $\alpha$  es el tamaño mínimo de las componentes del particionamiento, tal que  $n/k \geq \alpha > 0$ .

### 3.3.3. Problema de $k$ -Particionamiento Balanceado de Hipergrafos

Para este problema se define una función de costo sobre los vértices del hipergrafo  $r : V \mapsto \mathbb{R}^+$  y consiste en encontrar un  $k$ -particionamiento balanceado. La Formulación  $\mathcal{F}_3$  se basa en  $\mathcal{F}_1$  reemplazando las restricciones (3.3) por las siguientes:

$$L_c \leq \sum_{i \in V} r_i x_i^c \leq U_c, \quad \forall c \in [k], \quad (3.10)$$

donde  $L_c, U_c > 0$  son números enteros que imponen las cotas inferior y superior para los vértices en cada componente del particionamiento, respectivamente.

### 3.3.4. Problema de k-Equiparticionamiento de Hipergrafos en Hiperárboles Lineales

Como un primer acercamiento al problema del equiparticionamiento en hipergrafos cuando se exige conexidad en cada componente se plantea el  $k$ -particionamiento en *hiperárboles lineales*. Es decir, en este problema cada componente del particionamiento debe ser un hipergrafo conexo y para todo par de hiperaristas en la componente  $e_i, e_j$  se debe cumplir que  $|e_i \cap e_j| \leq 1$ , tal como se describe en la Sección 2.0.1 y en la Definición 2.6. Para este problema se han desarrollado dos formulaciones distintas.

#### Modelo Basado en la Formulación $\mathcal{F}_1$

Esta formulación se sustenta en los siguientes resultados:

**Teorema 3.1.** *Sea  $H = (V, E)$  un hipergrafo lineal.  $H$  es un hiperárbol lineal si y sólo si  $\sum_{e \in E} (|e| - 1) = |V| - 1$  y no contiene ciclos.*

*Demostración.* Supongamos que  $H$  es un hiperárbol lineal, por lo que no contiene ciclos. Por inducción sobre la cardinalidad de  $E$ , si  $|E| = 2$  el resultado es válido ya que existe un único nodo en la intersección. Se asume que es válido para  $|E| = m$  y que  $\sum_{e \in E} (|e| - 1) = |V| - 1$ . Entonces, si  $|E| = m + 1$ , la hiperarista  $e_{m+1}$  debe conectarse en exactamente un nodo con al menos una hiperarista del hiperárbol con conjunto de hiperaristas  $E \setminus \{e_{m+1}\}$ . Por lo tanto,

$$\begin{aligned} \sum_{e \in E \setminus \{e_{m+1}\}} (|e| - 1) + |e_{m+1}| - 1 &= |V(E \setminus \{e_{m+1}\})| - 1 + |e_{m+1}| - 1 \\ \iff \sum_{e \in E} (|e| - 1) &= (|V(E \setminus \{e_{m+1}\})| + |e_{m+1}| - 1) - 1 = |V(E)| - 1 \end{aligned}$$

Por otra parte, demostraremos por contraposición que si  $H$  no es un hiperárbol lineal, entonces este contiene ciclos o  $\sum_{e \in E} (|e| - 1) \neq |V| - 1$ . La

primera condición se sigue de la definición de hiperárbol, para la segunda condición supongamos que  $H$  está formado por  $k > 1$  sub-hiperárboles  $(V_1, E_1), (V_2, E_2), \dots, (V_k, E_k)$ , cada uno siendo un hiperárbol lineal. Así,

$$\begin{aligned} \sum_{e \in E} (|e| - 1) &= \sum_{e \in E_1} (|e| - 1) + \sum_{e \in E_2} (|e| - 1) + \dots + \sum_{e \in E_k} (|e| - 1) \\ &= |V_1| - 1 + |V_2| - 1 + \dots + |V_k| - 1 \\ &= |V| - k \neq |V| - 1 \end{aligned}$$

□

Se formula este problema como un programa entero  $\mathcal{F}_4$ , formado por las restricciones (3.1)-(3.8) y las siguientes restricciones:

$$\sum_{c \in [k]} y_j^c + y_l^c \leq 1, \quad \forall e_j, e_l \in E, |e_j \cap e_l| > 1, \quad (3.11)$$

$$\sum_{e_j \in E} (|e_j| - 1) y_j^c = \frac{n}{k} - 1, \quad \forall c \in [k], \quad (3.12)$$

$$\sum_{e_j \in S} y_j^c \leq |S| - 1, \quad \forall S \in \mathcal{C}(E), c \in [k], \quad (3.13)$$

donde,  $\mathcal{C}(E)$  está formado por todos los subconjuntos de hiperaristas de  $H$  que forman ciclos y de cardinalidad menor o igual a  $n/k$ . Las restricciones (3.11) imponen la condición de linealidad en las componentes del particionamiento y las restricciones (3.12)-(3.13), por el Teorema 3.1, garantizan la construcción de hiperárboles.

## Modelo de flujo

Para esta formulación se trabaja sobre una versión dirigida del grafo de incidencia asociado al hipergrafo:  $D = (W, A)$ , donde el conjunto de nodos está definido como  $W = V \cup V_E \cup \{s\}$  con  $V_E := \{v_e : e \in E\}$  y  $\{s\}$  el nodo fuente. Además, sea  $A = A_1 \cup A_2$ , donde  $A_1$  contiene a los arcos  $(s, i)$  con capacidades  $n/k$  para  $i \in V$  y  $A_2$  contiene a los arcos antiparalelos  $(i, j), (j, i)$  con capacidades  $n/k - 1$  para  $i \in V, j \in V_E$  si y sólo si el nodo  $i$  pertenece a la hiperarista  $e_j$ . Se introduce una función de demandas sobre los nodos  $d : W \mapsto \mathbb{Z}$ , donde  $d_s = -n, d_i = 1$ , para  $i \in V$  y  $d_j = 0$ , para  $j \in V_E$ . En la Figura 3.2 se representa la construcción del grafo dirigido

$D$ .

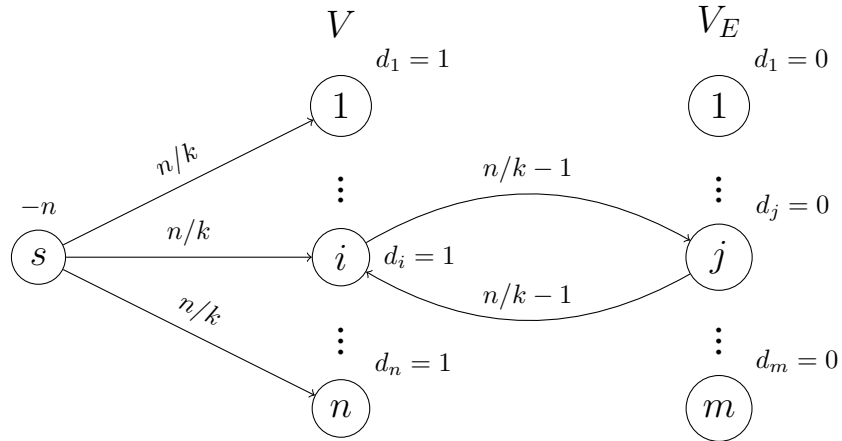


Figura 3.2: Construcción del grafo dirigido  $D$ .

En este modelo se debe encontrar un flujo que satisfaga las demandas en todos los nodos. A continuación, en la Figura 3.3 se presenta un ejemplo de particionamiento de un hipergrafo utilizando el modelo de flujo:

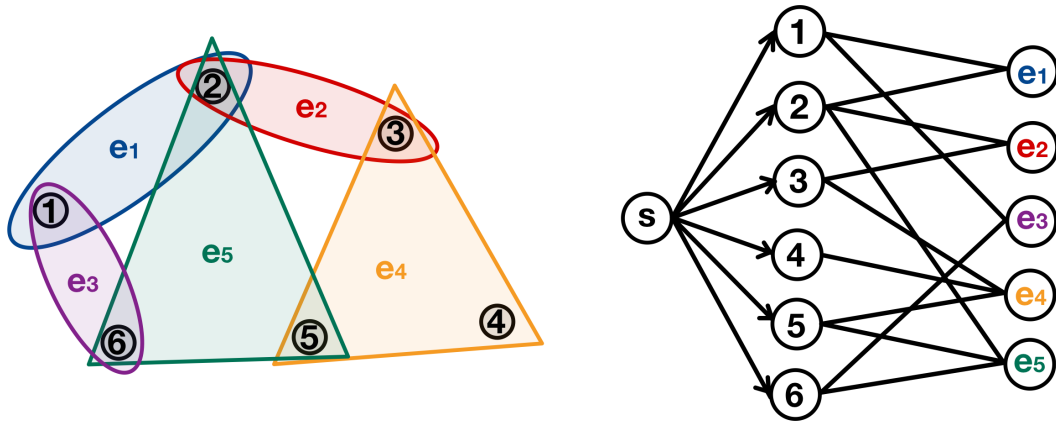


Figura 3.3: Grafo de incidencia asociado al hipergrafo.

Por simplicidad de la ilustración, cada par de arcos antiparalelos se ha representado por una sola arista.

En la Figura 3.4, al lado izquierdo se observa el hipergrafo con la solución al biparticionamiento ( $k = 2$ ) y al lado derecho la representación de dicho biparticionamiento a través del modelo de flujo en el grafo dirigido: el flujo sale de dos arcos del nodo fuente representando las dos componentes del particionamiento (color rojo y celeste), cada uno lleva

$n/k = 3$  unidades de flujo y satisfacen las demandas de 3 nodos.

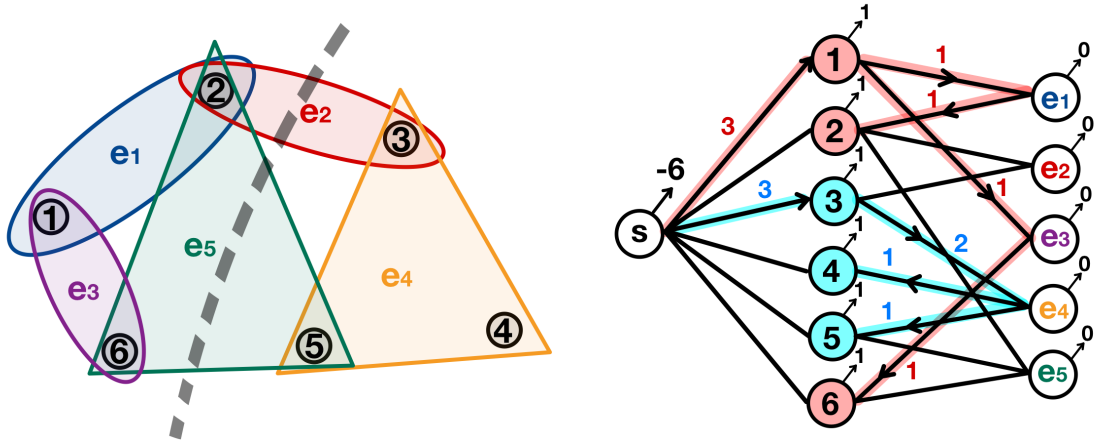


Figura 3.4: Particionamiento en el modelo de flujo.

Esta formulación consta de dos conjuntos de variables binarias y un conjunto de variables enteras positivas:  $h_j$  asociada a los nodos que representan a las hiperaristas,  $z_{i,j}$  asociada a los arcos del grafo dirigido y  $f_{i,j}$  asociada a los flujos sobre estos arcos. Sea  $h_j \in \{0, 1\}$  la variable que vale 1 si la hiperarista  $e_j \in E$  forma parte de la solución y cero caso contrario,  $f_{i,j} \in \mathbb{Z}^+$  el flujo sobre el arco  $(i, j) \in A$  y  $z_{i,j} \in \{0, 1\}$  la variable que vale 1 si  $f_{i,j} > 0$  y cero caso contrario. Entonces, el problema del  $k$ -Equiparticionamiento de Hipergrafos en Hiperárboles Lineales admite la siguiente formulación ( $\mathcal{F}_5$ ):

$$\text{mín } \sum_{j \in V_E} w_j h_j \quad (3.14)$$

s.a.

$$\sum_{(s,i) \in A_1} z_{si} = k, \quad (3.15)$$

$$f_{si} = \left(\frac{n}{k}\right) z_{si}, \quad \forall i \in V, \quad (3.16)$$

$$z_{ij} \leq f_{ij} \leq \left(\frac{n}{k} - 1\right) z_{ij}, \quad \forall (i, j) \in A, i \neq s, \quad (3.17)$$

$$\sum_{(i,j) \in A} f_{ij} - \sum_{(j,i) \in A} f_{ji} = d_i, \quad \forall i \in W, \quad (3.18)$$

$$\sum_{(j,i) \in A} z_{ji} = 1, \quad \forall i \in V, \quad (3.19)$$

$$\sum_{(i,j) \in A} z_{ij} = h_j, \quad \forall j \in V_E, \quad (3.20)$$

$$\sum_{(j,i) \in A} z_{ji} = (|e_j| - 1)h_j, \quad \forall j \in V_E, \quad (3.21)$$

$$z_{ij} + z_{ji} \leq 1, \quad \forall (i, j) \in A_2, \quad (3.22)$$

$$h_j \in \{0, 1\}, \quad \forall j \in V_E, \quad (3.23)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A, \quad (3.24)$$

$$f_{ij} \in \mathbb{Z}^+, \quad \forall (i, j) \in A. \quad (3.25)$$

La función objetivo (3.14) minimiza el costo de las hiperaristas que forman parte del particionamiento. Las restricciones (3.15) y (3.16) garantizan que el flujo salga a través de  $k$  arcos desde el nodo fuente y que cada flujo de estos arcos debe ser igual a su capacidad  $n/k$ . Las restricciones (3.17) imponen cotas positivas a los flujos de los arcos que han sido activados y de cero caso contrario. Las ecuaciones en (3.18) son las restricciones demanda sobre los nodos. Las restricciones (3.19) determinan que ingresa flujo por exactamente un arco a los nodos en  $V$ , además estas restricciones satisfacen las condiciones de linealidad ya que aseguran que la conexión de una hiperarista con otra sea a través de un único nodo de donde sale el flujo hacia la otra hiperarista. Las restricciones (3.20) y (3.21) aseguran que si una hiperarista  $e_j$  es parte de la solución, todos los vértices que la componen pertenezcan a la misma componente. Las restricciones (3.22) impiden que exista flujo en el arco anti-paralelo de un arco con flujo mayor a cero.

### Comprobación de la condición de linealidad del modelo $\mathcal{F}_5$

En la Figura 3.5 se plantea un ejemplo de particionamiento en componentes conexas sin considerar la condición de que las componentes sean hiperárboles lineales.

Sea el hipergrafo  $H = (V, E)$ , en donde  $V = \{1, 2, 3, 4, 5, 6, 7, 8\}$  y  $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ , con la condición de que  $w_{e_3} \gg w_{e_2}$ . Para el particionamiento planteado existen dos soluciones factibles: la primera solución (que no es un hiperárbol lineal) es considerando en la primera componente a las hiperaristas  $e_1, e_2$  y la segunda considerando a  $e_1, e_3$ , mientras que en ambas soluciones la segunda componente está conformada por las hiperaristas  $e_4, e_6$ . La solución óptima del

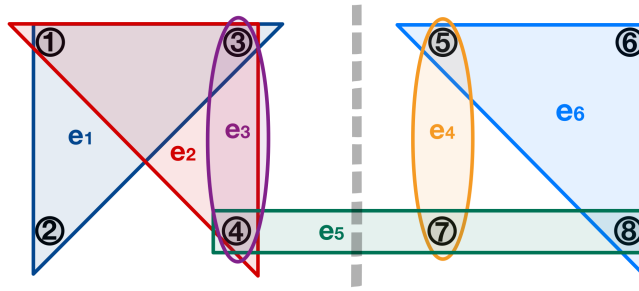


Figura 3.5: Ejemplo de particionamiento en componentes conexas.

particionamiento es considerando a  $e_1, e_2$  ya que el costo de  $e_2$  es mucho menor que el costo de  $e_3$ . A continuación se presentan las dos soluciones factibles del particionamiento mediante el grafo dirigido del modelo de flujo  $\mathcal{F}_5$ :

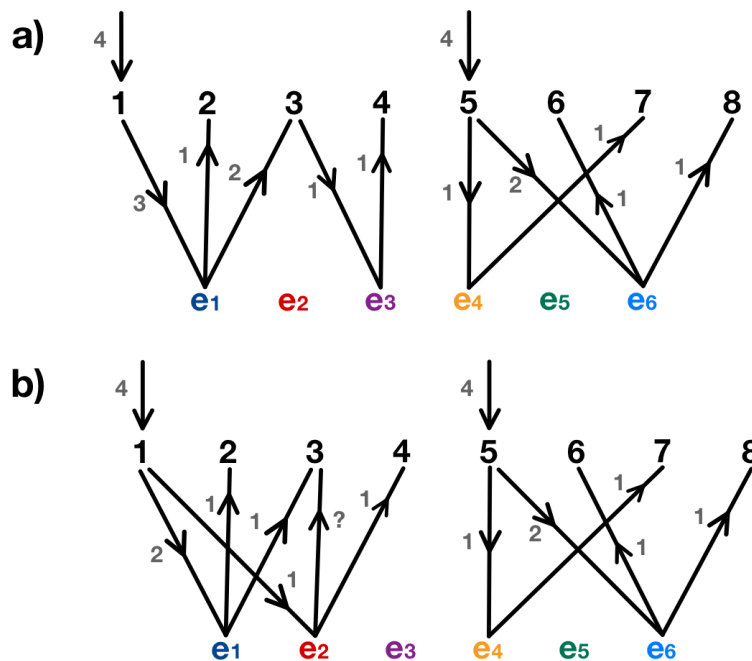


Figura 3.6: Grafos del modelo de flujo de las soluciones factibles.

En el grafo (a) de la Figura 3.6 se representa la solución factible considerando las hiperaristas  $e_1, e_3$  cuyo costo no es el mínimo (en el problema sin considerar la condición de linealidad). En el grafo (b) se muestra la solución óptima, esta solución en el grafo del modelo de flujo no es factible ya que a  $e_2$  sólo llega una unidad de flujo, lo cual no satisface la Restricción 3.21 que exige que salga flujo de  $e_2$  a través de 2 arcos ( $|e_2| - 1 = 2$ ) pero sólo llega una unidad de flujo. Por lo tanto,



el modelo de flujo planteado asegura que se cumple con la condición de conexión entre hiperaristas de una misma componente por medio de exactamente un vértice, lo cual da lugar a asegurar que las soluciones factibles del problema sean hiperárboles lineales.

### 3.4. Planos Cortantes

En esta sección se presentan varios conjuntos de desigualdades válidas para las diferentes formulaciones planteadas, así como otras desigualdades válidas específicas para cada formulación.

**Teorema 3.2** (Hiperarista tipo hoja). *Sea  $i \in V$  tal que  $E(i) = \{e_j\}$ . Entonces, las igualdades  $\sum_{c \in [k]} y_j^c = 1$  son válidas para  $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$  y  $\mathcal{F}_4$ , mientras que la igualdad  $h_j = 1$  es válida para  $\mathcal{F}_5$ .*

*Demostración.* Al ser  $e_j$  la única hiperarista que cubre al vértice  $i$ , es claro que esta hiperarista debe formar parte del particionamiento, por lo que debe asignarse a una componente del mismo.  $\square$

**Teorema 3.3** (Tamaño de hiperaristas mayor a  $n/k$ ). *Sea  $e_j \in E$  tal que  $|e_j| > n/k$ . Entonces, la igualdad  $\sum_{c=1}^k y_j^c = 0$  es válida para  $\mathcal{F}_1$  y  $\mathcal{F}_4$ , mientras que la igualdad  $h_j = 0$  es válida para  $\mathcal{F}_5$ .*

*Demostración.* Según se definió el problema de  $k$ -equiparticionamiento, toda hiperarista que forme parte de la solución debe estar contenida completamente en una componente (cuyo tamaño es  $n/k$ ). Por lo tanto, el tamaño máximo de las hiperaristas válidas para ser parte de la solución es  $n/k$ .  $\square$

**Teorema 3.4** (Eliminación de simetrías). *Sea un subconjunto de vértices  $L = \{1, 2, \dots, k\}$ . Entonces, las igualdades  $\sum_{c=i+1}^k x_i^c = 0$  para todo  $i \in L$ , son válidas para  $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$  y  $\mathcal{F}_4$ .*

*Demostración.* La validez de las desigualdades se deduce del hecho que si el nodo 1 es asignado a la componente 1 del particionamiento, entonces no podrá pertenecer a las componentes  $\{2, 3, \dots, k\}$ . Luego, si el nodo 2 es asignado a la componente 1 o componente 2, el resto de sus variables  $x_2^c, c \in \{3, 4, \dots, k\}$  debe ser cero. Siguiendo esta lógica, el nodo  $k - 1$  debe

pertenecer a alguna componente  $c \in \{1, 2, \dots, k-1\}$  y por lo tanto este nodo no puede pertenecer a la componente  $k$ .  $\square$

**Teorema 3.5** (Hiperaristas contenidas). *Sea  $T \subset E$  y  $e_j \in E \setminus T$ , tal que  $V(T) \subseteq e_j$ . Entonces, para todo  $e_\ell \in T$ , la desigualdad  $\sum_{c \in [k]} y_j^c + y_\ell^c \leq 1$ , es válida para  $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$  y  $\mathcal{F}_4$ , y la desigualdad  $h_j + h_\ell \leq 1$  es válida para  $\mathcal{F}_5$ .*

*Demostración.* Ya que  $e_j \cap e_\ell = e_\ell$  es válido para todo  $e_\ell \in T$ , si  $e_j$  pertenece a cualquier componente de la solución óptima, entonces  $e_\ell$  no puede pertenecer a la solución óptima debido a que los vértices en  $e_\ell$  ya se encuentran cubiertos por  $e_j$  y por lo tanto incluir a  $e_\ell$  incrementaría el valor de la función objetivo.  $\square$

**Definición 3.1.** *Sea  $F \subset E$ , se dice que  $F$  es un conjunto minimal si:*

1.  $|V(F)| > n/k$ , y
2.  $|V(F \setminus \{e\})| \leq n/k, \forall e \in F$

**Teorema 3.6.** *Sea el conjunto minimal  $S \subset E$ , entonces las desigualdades*

$$\sum_{e_\ell \in S} y_\ell^c \leq |S| - 1, \forall c \in [k],$$

*son válidas para  $\mathcal{F}_1$  y  $\mathcal{F}_4$ .*

*Demostración.* El resultado anterior se deduce directamente de la definición de conjunto minimal. Al eliminar cualquier hiperarista de  $S$  el tamaño del conjunto restante será menor o igual a  $n/k$ , por lo tanto todas las hiperaristas de  $S$  excepto una pueden formar parte de una componente del particionamiento.  $\square$

**Corolario 3.1.** *Sea  $S \subset E$  un conjunto minimal,  $(V(S), S)$  un hiperárbol lineal y  $T_S = \{e \in S : S \setminus \{e\} \text{ forma un hiperárbol lineal}\}$ . Entonces, las desigualdades  $\sum_{e_j \in T_S} y_j^c \leq |T_S| - 1$ , para todo  $c \in [k]$ , son válidas para  $\mathcal{F}_4$ .*

**Teorema 3.7.** *Sea  $T \subset E$  tal que  $|V(T)| < n/k, e_j \cap e_\ell = \emptyset$ , para todo  $e_j, e_\ell \in T$ , y  $r = n/k - |V(T)|$ . Además, sea  $\mathcal{P}(T) := \{e_p \in E \setminus T : |e_p| \geq r + 1, e_p \cap e_q = \emptyset, e_q \in E\}$ . Si  $\mathcal{P}(T) \neq \emptyset$ , entonces la desigualdad:*

$$\sum_{e_j \in T} |e_j| y_j^c + \sum_{e_p \in \mathcal{P}(T)} (|e_p| - r) y_p^c \leq |V(T)|, \quad \forall c \in [k]$$

*es válida para  $\mathcal{F}_1$ .*

*Demostración.* Se consideran dos casos:

- Si  $\sum_{e_p \in \mathcal{P}(T)} y_p^c = 0$ , entonces

$$\sum_{e_j \in T} |e_j| y_j^c + \sum_{e_p \in \mathcal{P}(T)} (|e_p| - r) y_p^c = \sum_{e_j \in T} |e_j| y_j^c \leq |V(T)|, \quad \forall c \in [k]$$

- Si  $\sum_{e_p \in \mathcal{P}(T)} y_p^c \geq 1$ , implica que

$$\begin{aligned} & \sum_{e_j \in T} |e_j| y_j^c + \sum_{e_p \in \mathcal{P}(T)} (|e_p| - r) y_p^c \\ &= \sum_{e_j \in T} |e_j| y_j^c + \sum_{e_p \in \mathcal{P}(T)} |e_p| y_p^c - \sum_{e_p \in \mathcal{P}(T)} r y_p^c \\ &\leq \sum_{e_j \in E} |e_j| y_j^c - r \sum_{e_p \in \mathcal{P}(T)} y_p^c \leq |V(T)| \end{aligned}$$

ya que  $\sum_{e_j \in E} |e_j| y_j^c \geq n/k$  para todo  $c \in [k]$ .

□

**Teorema 3.8.** Sean las hiperaristas  $e_i, e_j \in E$  tal que  $|e_i \cap e_j| > 1$ . Entonces, la desigualdad  $h_i + h_j \leq 1$  es válida para  $\mathcal{F}_5$ .

*Demostración.* Bajo la condición de que las componentes del particionamiento deben ser hiperárboles lineales, dos hiperaristas cuya intersección sea mayor a 1 no formarán parte de una misma componente en alguna solución factible. □

**Lema 3.1** (Número mínimo de hiperaristas). Sea  $H = (V, E)$  un hipergrafo tal que existe  $T \subset E$  y que  $(V, T)$  forma un hiperárbol lineal. Además, sea  $\tau_1 \geq \tau_2 \geq \dots \geq \tau_l$  enteros no-negativos donde  $\tau_1 = \max_{j \in [m]} \{|e_j|\}$ ,  $\tau_l = \min_{j \in [m]} \{|e_j|\}$ . Sea  $\alpha_j$ , el número mínimo de hiperaristas de tamaño  $\tau_j$  en el hiperárbol lineal, para  $j \in [l]$ . Entonces,  $|T| \geq \sum_{j=1}^l \alpha_j$ .

El Lema 3.1 proporciona una cota inferior al número de hiperaristas de un hiperárbol lineal contenido en un hipergrafo y su demostración se sigue de la definición de  $\alpha_j$ . Estos coeficientes se pueden obtener de manera iterativa al resolver el problema  $\min \{ \sum_{j=1}^l \alpha_j : \sum_{j=1}^l (\tau_j - 1) \alpha_j = |V| - 1, \alpha_j \in \mathbb{Z}^+, j \in [l] \}$ . Por lo tanto, aplicando el Lema 3.1 a las componentes del particionamiento (ya que cada una es un hiperárbol lineal), se obtiene las siguientes desigualdades válidas.

**Teorema 3.9.** *La desigualdad*

$$\sum_{c \in [k]} \sum_{j \in [m]} y_j^c \geq k \sum_{p=1}^l \alpha_p$$

es válida para  $\mathcal{F}_4$ , y

$$\sum_{j \in [m]} h_j \geq k \sum_{p=1}^l \alpha_p$$

es válida  $\mathcal{F}_5$ .

### 3.5. Heurística

Como se había mencionado previamente, las heurísticas más utilizadas en el particionamiento de hipergrafos se enfocan en la función objetivo del peso total de hiperaristas en el corte. Sin embargo, en la función objetivo de nuestro problema las hiperaristas en el corte no son tomadas en cuenta para cubrir los vértices del hipergrafo y por lo tanto en un particionamiento aleatorio (punto de partida para las heurísticas de particionamiento) pueden no existir hiperaristas contenidas completamente en las componentes del particionamiento que cubran a todos los vértices.

Por esta misma premisa dificulta aplicar, como primera aproximación, un algoritmo glotón ya que al construir una componente de un particionamiento se pueden estar usando o descartando hiperaristas que hacen posible cubrir los vértices de otra componente.

Se presenta el Algoritmo 1 para calcular las hiperaristas que cubren las diferentes componentes de un particionamiento y el número de vértices cubiertos en cada componente. En este algoritmo, dados los subconjuntos de vértices que forma un particionamiento, se encuentra todas las hiperaristas del hipergrafo que están completamente contenidas en cada partición y el número total de vértices que cubren estas hiperaristas.

Es fácil ver que si el número de vértices cubiertos en cada componente del particionamiento es  $n/k$ , el particionamiento es factible.

---

**Algoritmo 1:** Cubrimiento por hiperaristas.

---

**Input:** - Hipergrafo  $H = (V, E)$

- Particionamiento inicial aleatorio  $\{V_1, V_2, \dots, V_k\}$

**Output:** - Hiperaristas que forman parte del particionamiento

$\{E_1, E_2, \dots, E_k\}$

- Cantidad de vértices cubiertos en cada partición  $C \in \mathbb{R}^k$

```
1  $C \leftarrow \{0, \dots, 0\}$ 
2  $E_c \leftarrow \emptyset, \forall c \in [k]$ 
3 for  $c = 1, \dots, k$  do
4    $v_{cover} = \emptyset$ 
5   for  $j = 1, \dots, m$  do
6     if  $e_j \subseteq V_c$  then
7        $v_{cover} = v_{cover} \cup V(e_j)$ 
8        $E_c = E_c \cup \{e_j\}$ 
9    $C_k = |v_{cover}|$ 
10 return  $(E, C)$ 
```

---

### 3.5.1. Heurística para Particionamiento Factible

Basado en el algoritmo de Kernighan y Lin [15], la heurística que se desarrolló busca encontrar un particionamiento factible a partir de un particionamiento inicial aleatorio.

Definimos en base al Algoritmo 1 una función de ganancia  $g(x, y)$  entre el par de vértices  $x, y$  que pertenezcan a diferentes componentes del particionamiento como el número total de vértices cubiertos en las componentes a las que pertenecen los vértices  $x, y$  si son intercambiados.

Se presenta el Algoritmo 2 para encontrar un cubrimiento factible en el caso de un biparticionamiento:

---

**Algoritmo 2:** Cubrimiento Factible-Biparticionamiento.

---

**Input:** - Hipergrafo  $H = (V, E)$ ,  $|V| = n$ ,  $|E| = m$

- Particionamiento inicial aleatorio  $P_i = \{V_1, V_2\}$

- Número máximo de iteraciones  $it_{max}$

**Output:** - Particionamiento factible  $P_f$

```
1  $L_1, L_2 \leftarrow V_1, V_2$ 
2  $it \leftarrow 0$ 
3 while  $it < it_{max}$  do
4    $S_1, S_2 \leftarrow \emptyset$ 
5   while  $|S_1|, |S_2| \neq n/k$  do
6      $C \leftarrow \emptyset$ 
7     for  $i \in L_1 \setminus S_1$  do
8       for  $j \in L_2 \setminus S_2$  do
9          $C = C \cup \{g(i, j)\}$ 
10     $c_m = \max_{c \in C} \{c\}$ 
11     $x, y \leftarrow g(x, y) = c_m$ 
12     $L_1 = L_1 \setminus \{x\} \cup \{y\}$ 
13     $L_2 = L_2 \setminus \{y\} \cup \{x\}$ 
14    if  $c_m = n$  then
15       $P_f = \{L_1, L_2\}$ 
16      return (True,  $P_f$ )
17    else
18       $S_1 = S_1 \cup \{y\}$ 
19       $S_2 = S_2 \cup \{x\}$ 
20     $it = it + 1$ 
21 return (False,  $\emptyset$ )
```

---

Del particionamiento inicial  $P_i = \{V_1, V_2\}$  se extraen las listas de vértices  $L_1, L_2$  que están disponibles para intercambiar. Luego, se calcula la ganancia para cada par de vértices  $(i, j)$ ,  $i \in L_1$ ,  $j \in L_2$  y se escoge el par de vértices  $x \in L_1$ ,  $y \in L_2$  que tengan el valor de ganancia máximo, es decir, el par de vértices de cuyo intercambio entres las componentes del particionamiento resulte en la mayor cantidad de vértices cubiertos. Si la ganancia es igual a  $n$  se ha conseguido un particionamiento factible  $P_f$  con ese intercambio. Caso contrario, los vértices  $x, y$  son intercambiados

de componentes y eliminados de la lista de intercambios disponibles. Si luego de haber intercambiado todos los nodos disponibles no se ha conseguido un particionamiento factible, se reinicia la lista de nodos disponibles y se repite el proceso hasta conseguir una solución factible o llegar al número máximo de pases (cantidad de veces que se reinicia la lista de nodos disponibles).

Cada vez que se escoge la ganancia máxima se realizan  $n^2/4$  operaciones (número de intercambios entre las dos listas de nodos); estas operaciones a su vez se repiten un máximo de  $n/2$  veces, que son las veces que se actualiza la lista de nodos disponibles antes de quedar vacía. Por lo tanto, la complejidad de este algoritmo está acotada superiormente por  $O(pn^3)$ , donde  $n$  es el número de nodos y  $p$  es el número máximo de pases.

### 3.6. Aplicaciones

Al dar un enfoque distinto al particionamiento de hipergrafos con una función objetivo nueva, también se presenta la posibilidad de nuevas aplicaciones del mundo real para ser tratadas a través del particionamiento de hipergrafos.

En base a los diferentes conjuntos de datos de 'orden superior' (conectividad entre más de dos nodos) en [4], se plantean los siguientes usos del particionamiento de hipergrafos en:

- **Datos de coautoría:** donde los nodos representan autores y cada hiperarista es una publicación con varios autores.

Se podría encontrar un número dado de grupos de autores que hayan realizado publicaciones entre ellos a través del problema de particionamiento en hiperárboles lineales.

- **Redes de medicamentos del Directorio Nacional de Códigos de Medicamentos:** donde cada nodo es una sustancia y cada hiperaristas representa el conjunto de sustancias presentes en un medicamento.

Mediante un particionamiento de los datos en redes de medicamentos se agruparía las sustancias utilizadas en diferentes medicamentos.

Así, dando por ejemplo un costo relacionado a la producción de cada medicamento, el particionamiento nos ayudaría a identificar grupos de sustancias que producen ciertos medicamentos con el menor costo total de producción o también que que costos de las sustancias en cada grupo sea balanceado.

- **Contacto humano:** en donde cada nodo es una persona y cada hiperarista es el conjunto de personas que estuvieron en contacto próximo entre ellas.

Una posible aplicación de datos de contacto humano podría ser en redes sociales, en donde los nodos serían usuarios y cada hiperarista representaría un grupo o interés en común de los usuarios. Si se quisiera separar los datos de los usuarios en distintos servidores, se podría agrupar en conjuntos de usuarios con más intereses/grupos en común.

- **Colaboraciones musicales:** en donde cada nodo es un artista y cada hiperarista es el conjunto de artistas que colaboran en una canción.

Similar a los datos de coautoría, en colaboraciones musicales mediante el particionamiento se agruparía artistas que hayan hecho colaboraciones entre ellos. Y asignando un valor a cada artista, se puede plantear el problema del particionamiento balanceado.



# Capítulo 4

---

## Pruebas Computacionales

---

En este capítulo se presentan los resultados de las diversas pruebas computacionales realizadas al implementar las formulaciones  $\mathcal{F}_1$ ,  $\mathcal{F}_2$  y  $\mathcal{F}_3$ , así como la comparación de los modelos añadiendo los planos cortantes: Teorema 3.2 (TH), Teorema 3.4 (ES), Teorema 3.5 (CO), Teorema 3.6 (KS), como también diferentes combinaciones de estos planos cortantes. Luego se muestran los resultados de la comparación de las formulaciones  $\mathcal{F}_4$  y  $\mathcal{F}_5$ , así como los resultados de la implementación de la heurística para encontrar particionamientos factibles.

Todas las pruebas se realizaron en una laptop con procesador Intel i7 de 2.20 GHz, 16 GB de memoria RAM, sistema operativo Windows 11 Home y con el solver de optimización Gurobi 9.5.0 y el lenguaje de programación Python. Se configuró el tiempo máximo de resolución de las instancias en 1800 segundos y se desactivó el parámetro para la generación predeterminada de cortes.

Los códigos de implementación de las cinco formulaciones se pueden encontrar en el siguiente repositorio: [https://github.com/gullolan/Hypergraph\\_partitioning](https://github.com/gullolan/Hypergraph_partitioning).

## 4.1. Generación de Instancias

Las instancias comprenden hipergrafos aleatorios con número de vértices ( $n$ ) entre 45 a 180, número de hiperaristas ( $m$ ) entre 100 y 2000, y para cada instancia se consideran tamaños máximos de hiperaristas ( $|e_{max}|$ ) desde 2 hasta 20, dependiendo del tamaño de cada instancia. En la Tabla 4.1 se muestran las instancias generadas para las pruebas con sus diferentes tamaños máximos de hiperaristas:

Tabla 4.1: Instancias para pruebas computacionales.

Instancias ( $n, m$ )	$ e_{max} $
(45,100)	
(45,200)	2,5,10
(45,500)	
(90,150)	
(90,500)	2,5,10,20
(90,1000)	
(150,300)	
(150,800)	2,5,10,20
(150,1500)	
(180,400)	
(180,1000)	2,5,10,20
(180,2000)	

Para la generación de las instancias se desarrolló un algoritmo basado en Lee et al. [18] (Algoritmo 1):

Con este algoritmo se obtiene la matriz de incidencia del hipergrafo dado el número de vértices  $n$ , el número de hiperaristas  $m$  y la cardinalidad máxima de las hiperaristas  $c$ .

## 4.2. Análisis de Resultados

Para todos los modelos e instancias se consideró un particionamiento con  $k = 3$ ; si una instancia resulta infactible, se generará un nuevo hipergrafo aleatorio hasta que se obtenga una instancia factible o se hayan generado diez hipergrafos diferentes. Los resultados se presentan

---

**Algoritmo 3:** Generación de hipergrafos aleatorios.

---

**Input:** - Número de nodos  $n$   
- Número de hiperaristas  $m$   
- Tamaño máximo de hiperarista  $c$

**Output:** Matriz de incidencia  $A \in \mathbb{R}^{n \times m}$

```
1  $|e_j| \leftarrow$  tamaño de hiperarista aleatorio entre 2 y  $c$ ,  $\forall j \in [m]$ 
2  $a_{ij} \leftarrow 0$ ,  $\forall i \in [n], j \in [m]$ 
3 for  $j = 1, \dots, m$  do
4    $t = 0$ 
5   while  $t < |e_j|$  do
6      $s \leftarrow$  número aleatorio entre 1 y  $n$ .
7     if  $a_{sj} = 0$  then
8        $a_{sj} = 1$ 
9        $t = t + 1$ 
10 return  $A$ 
```

---

en las diferentes tablas con el siguiente formato: la instancia de cada prueba (número de vértices, número de hiperaristas), tamaño promedio de hiperarista, tamaño máximo de hiperarista, tipo de plano cortante, valor obtenido de la función objetivo, gap de optimalidad en porcentaje entre las cotas superior y menor finales, el número de nodos explorados en el esquema Branch&Cut, el número de planos cortantes añadidos (mediante funciones Callback y/o en el nodo raíz) y el tiempo de ejecución en segundos. Así también, para instancias con  $|e_{max}| = 2$  no se implementan las desigualdades CO y en las instancias donde los hipergrafos son menos densos (menor número de hiperaristas respecto al número de vértices) y  $|e_{max}| = 2$ , son las únicas donde se consideran las desigualdades TH.

#### 4.2.1. Resultados formulación $\mathcal{F}1$

Los resultados completos se presentan en la Tabla A.1 del Anexo A. A continuación se presentan los resultados de las instancias y planos cortantes, se han dividido las instancias por el tamaño máximo de hiperarista:

En la Figura 4.1 se presentan las instancias de grafos ( $|e_{max}| = 2$ ), las cuales se resuelven hasta la optimalidad en tiempos menores a 1 segundo. No se evidencia una mejora considerable con la adición de los

diferentes planos cortantes, incluso en algunas instancias los planos aumentan el tiempo de ejecución. Adicionalmente, la instancia [150, 300] no es factible.

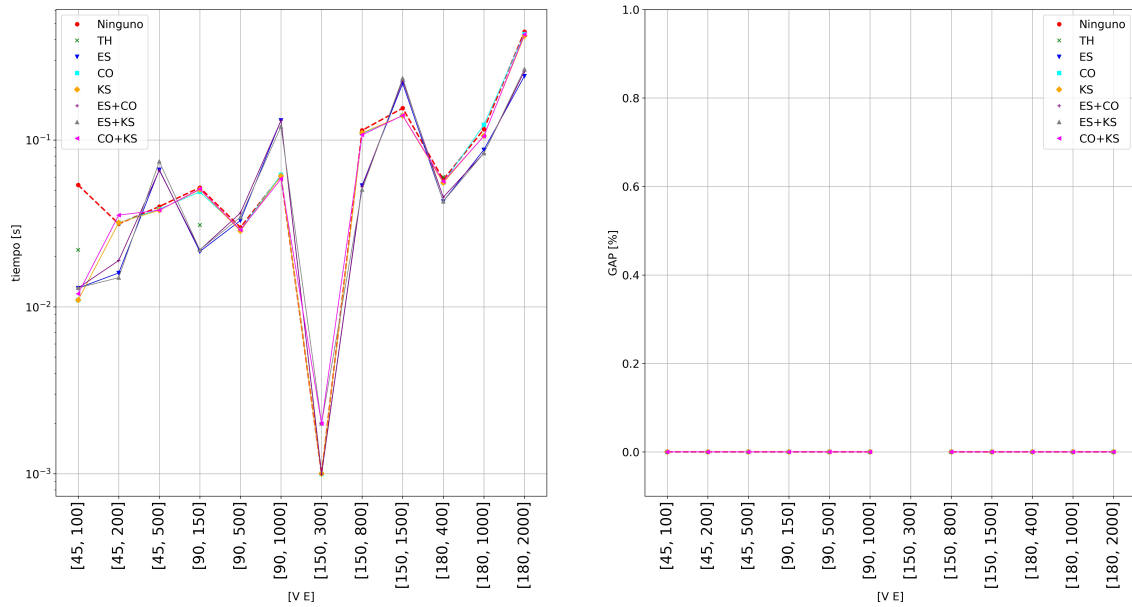


Figura 4.1: Resultados modelo  $\mathcal{F}_1$ ,  $|e_{max}| = 2$ .

Todas las instancias con  $|e_{max}| = 5$  se resuelven hasta la optimalidad en un tiempo máximo aproximado de 100 segundos. De los planos cortantes implementados, el que presenta mejores tiempos en la mayoría de instancias es el de eliminación de simetrías **ES**. Además, con los planos **CO** se reduce el número de nodos explorados, como se puede observar en la Tabla A.1. Los resultados se muestran en la Figura 4.2.

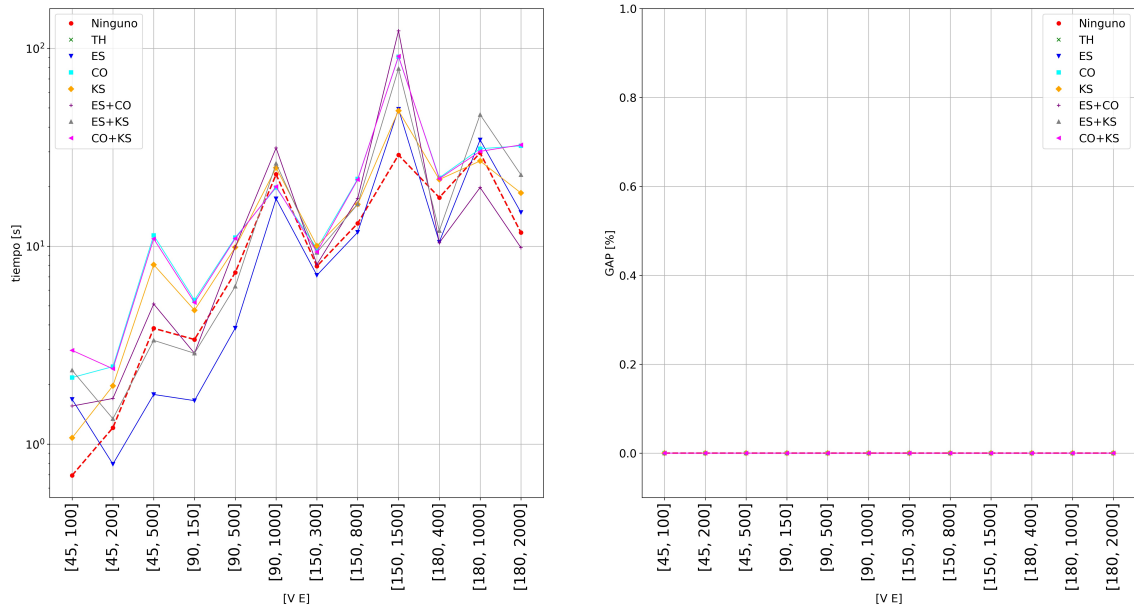


Figura 4.2: Resultados modelo  $\mathcal{F}_1$ ,  $|e_{max}| = 5$ .

En las instancias con  $|e_{max}| = 10$ , hasta la instancia  $[90, 150]$  se resuelven hasta la optimalidad. Desde la instancia  $[90, 500]$ , todas llegan al tiempo límite en donde los planos cortantes **ES**, **CO** y su combinación presentan mejoras en el GAP de optimalidad y/o menor número de nodos explorados, como se señala en la Tabla A.1. En la Figura 4.3 se muestra los resultados obtenidos.

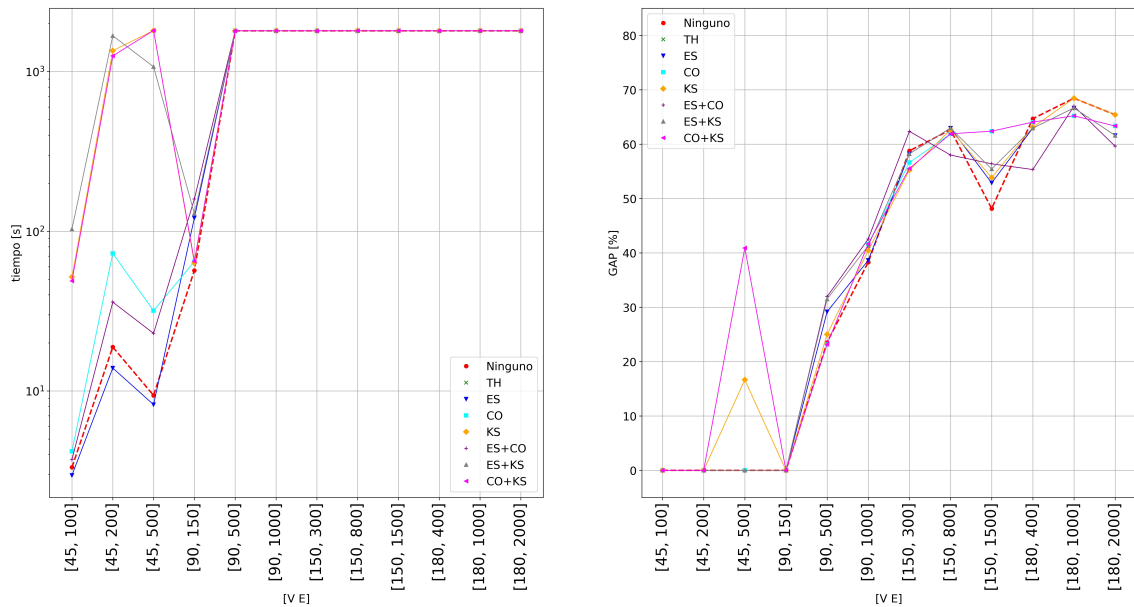


Figura 4.3: Resultados modelo  $\mathcal{F}_1$ ,  $|e_{max}| = 10$ .

Para las instancias con  $|e_{max}| = 20$ , la instancia  $[90, 150]$  fue infactible, mientras que en las instancias  $[150, 300]$  y  $[180, 400]$  no se obtuvo una solución factible en el tiempo máximo de ejecución. El resto de instancias llegan al tiempo límite y en donde los planos **CO** y **ES** son los que más mejoran el valor de la función objetivo y exploran un menor número de nodos, como se señala en la Tabla A.1. En la Figura 4.4 se muestra los resultados obtenidos.

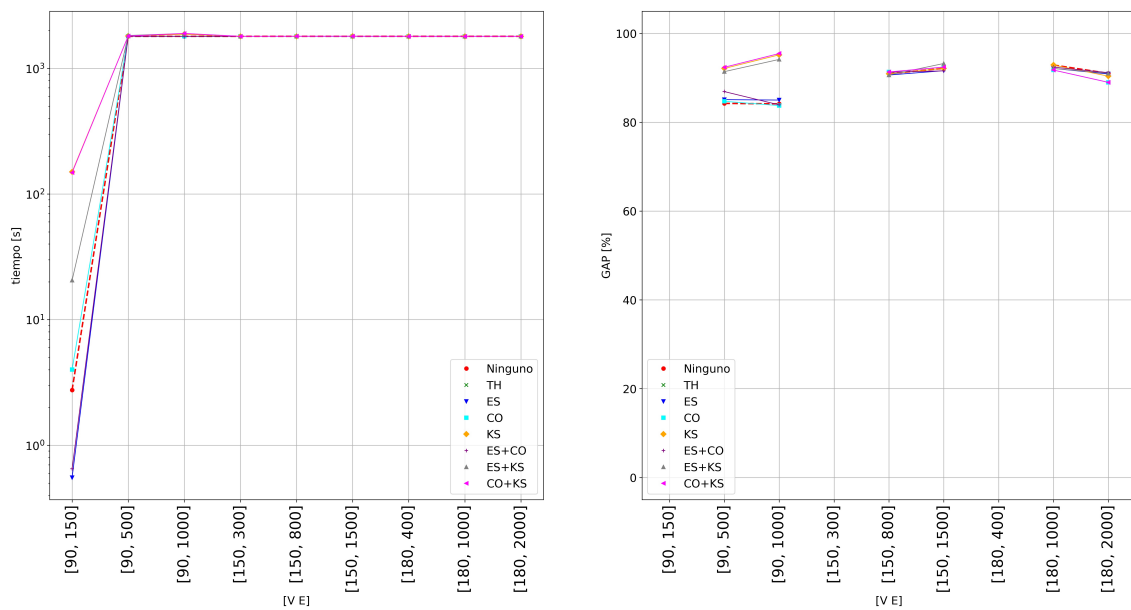


Figura 4.4: Resultados modelo  $\mathcal{F}_1$ ,  $|e_{max}| = 20$ .

## 4.2.2. Resultados formulación $\mathcal{F}_2$

Los resultados completos se presentan en la Tabla A.2 del Anexo A. A continuación se presentan los resultados de las instancias y planos cortantes, se han dividido las instancias por el tamaño máximo de hiperarista:

En la Figura 4.5 se presentan las instancias de grafos ( $|e_{max}| = 2$ ), las cuales se resuelven hasta la optimalidad en menos de 1 segundo y resolviéndose en el nodo raíz, mientras que las instancias  $[90, 150]$  y  $[150, 300]$  no resultaron factibles.

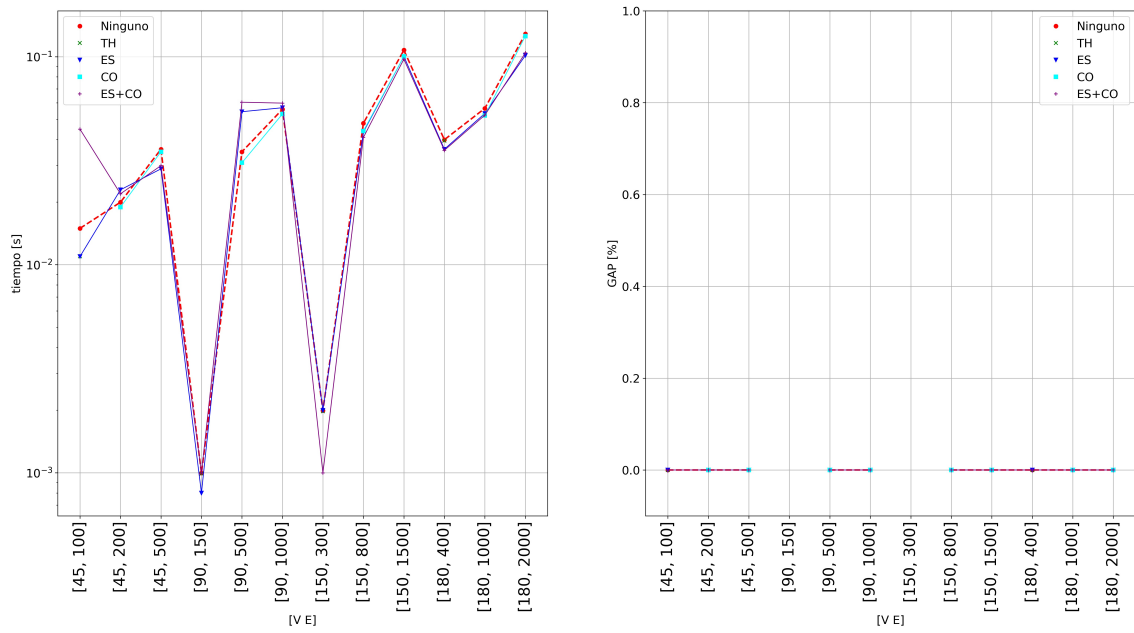


Figura 4.5: Resultados modelo  $\mathcal{F}_2$ ,  $|e_{max}| = 2$ .

Todas las instancias con  $|e_{max}| = 5$  se resuelven hasta la optimalidad con un tiempo máximo de 39 segundos. El plano cortante con mejores resultados en la mayoría de instancias es el de eliminación de simetrías **ES**, seguido por la combinación **ES+CO**. A pesar de no mostrar los mejores tiempos, con el plano cortante **CO** es con el que se explora menor cantidad de nodos, como se señala en la Tabla A.2. Los resultados se muestran en la Figura 4.6.

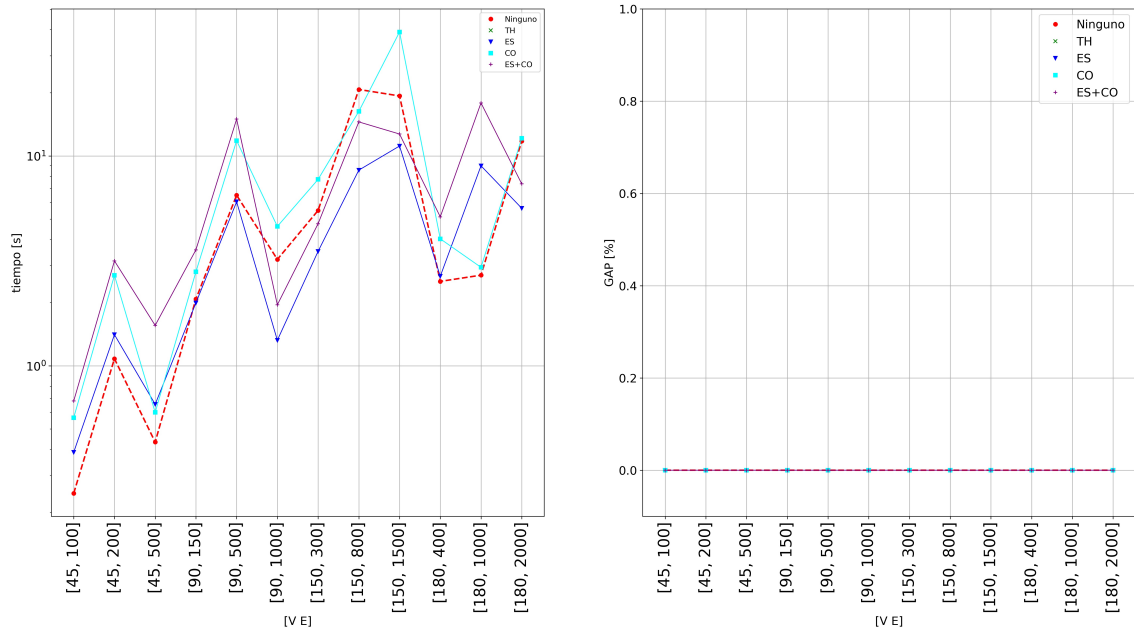


Figura 4.6: Resultados modelo  $\mathcal{F}_2$ ,  $|e_{max}| = 5$ .

En las instancias con  $|e_{max}| = 10$ , hasta la instancia  $[90, 500]$  se resuelven hasta la optimalidad en tiempos menores a 300 segundos. En la instancia  $[90, 1000]$  se logra la optimalidad con plano cortante **ES**. Desde la instancia  $[150, 300]$ , el plano cortante **CO** es el que presenta mejoras significativas tanto en el GAP de optimalidad como en el número de nodos explorados. En la Figura 4.7 se muestra los resultados obtenidos.

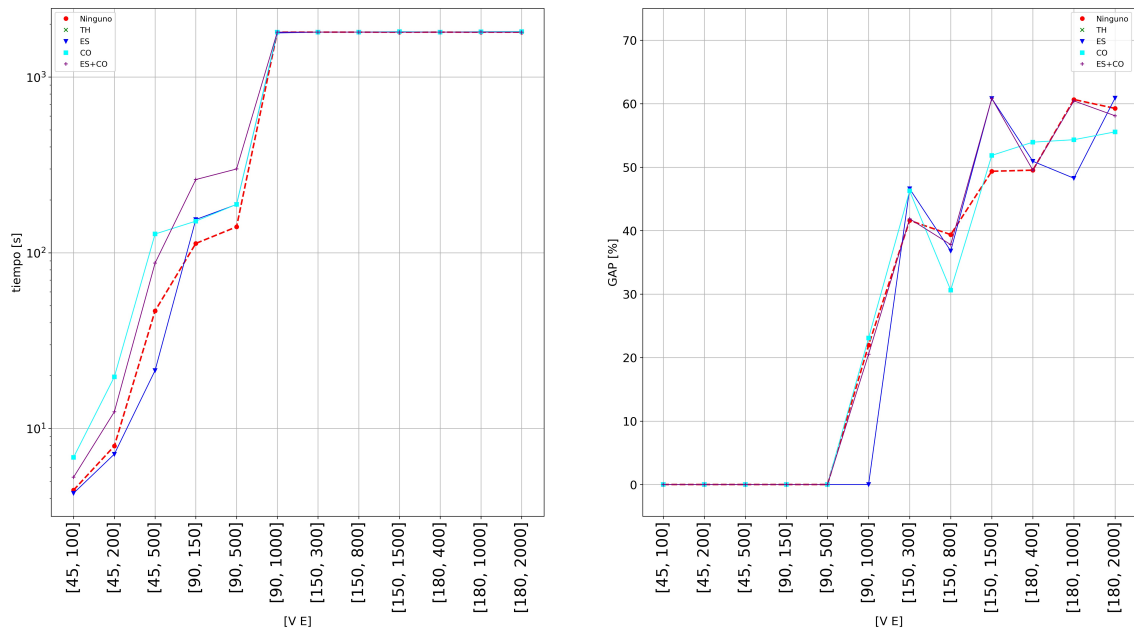


Figura 4.7: Resultados modelo  $\mathcal{F}_2$ ,  $|e_{max}| = 10$ .



Para las instancias con  $|e_{max}| = 20$ , la instancia  $[90, 150]$  fue infactible, mientras que en las instancias  $[150, 300]$  y  $[180, 400]$  no se obtuvo una solución factible en el tiempo máximo de ejecución. El plano cortante **CO** es el que obtiene mejores valores de la función objetivo en un menor número de nodos explorados. En la Figura 4.8 se muestra los resultados obtenidos.

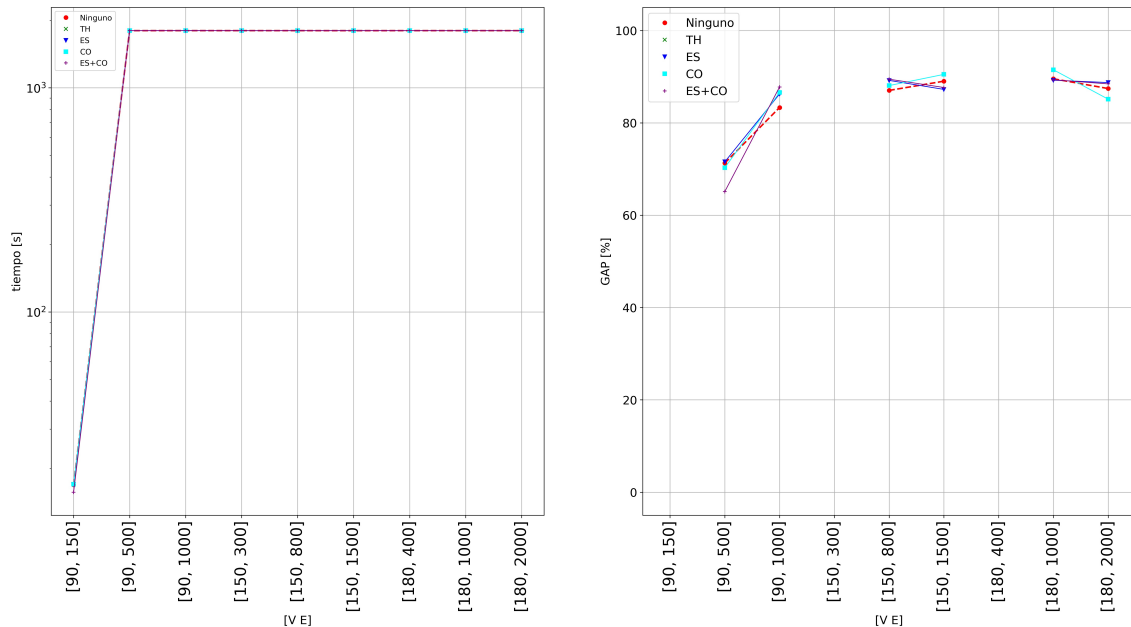


Figura 4.8: Resultados modelo  $\mathcal{F}_2$ ,  $|e_{max}| = 20$ .

### 4.2.3. Resultados formulación $\mathcal{F}_3$

Los resultados completos se presentan en la Tabla A.3 del Anexo A. A continuación se presentan los resultados de las instancias y planos cortantes, se han dividido las instancias por el tamaño máximo de hiperarista:

En la Figura 4.9 se presentan las instancias de grafos ( $|e_{max}| = 2$ ), las cuales se resuelven hasta la optimalidad en tiempos menores a 1 segundo y resolviéndose en el nodo raíz. Las instancias  $[90, 150]$  y  $[150, 300]$  resultaron infactibles.

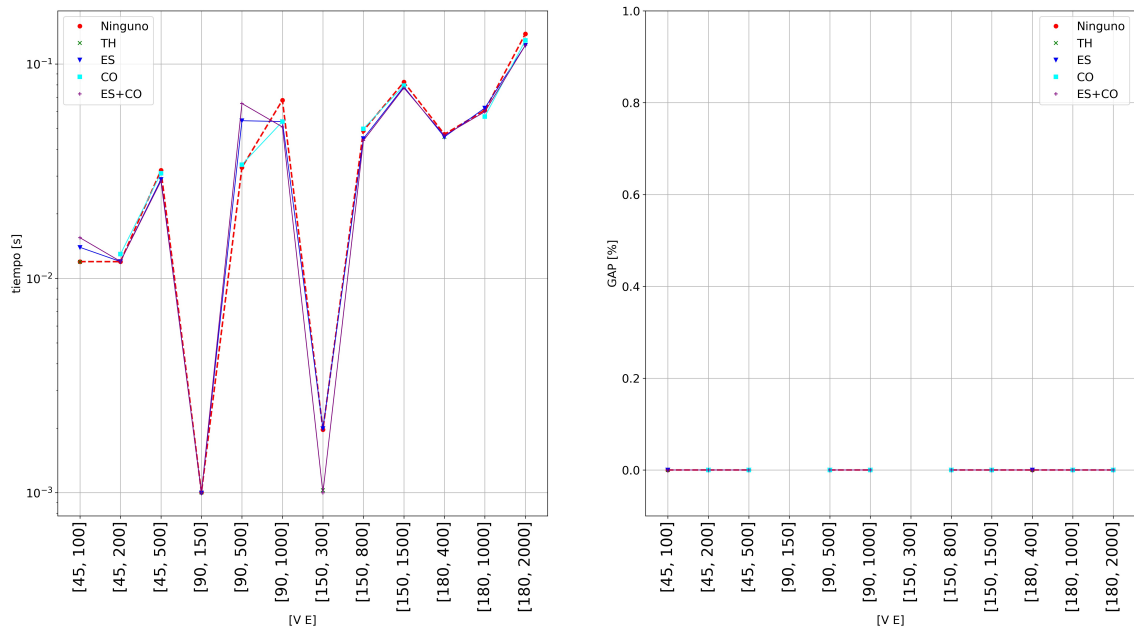


Figura 4.9: Resultados modelo  $\mathcal{F}_3$ ,  $|e_{max}| = 2$ .

Todas las instancias con  $|e_{max}| = 5$  se resuelven hasta la optimalidad con tiempos máximos de aproximadamente de 100 segundos. Los planos cortantes que presentan mejores resultados en la mayoría de instancias es el de eliminación de simetrías **ES** y la combinación **ES+CO**. Los resultados se muestran en la Figura 4.10.

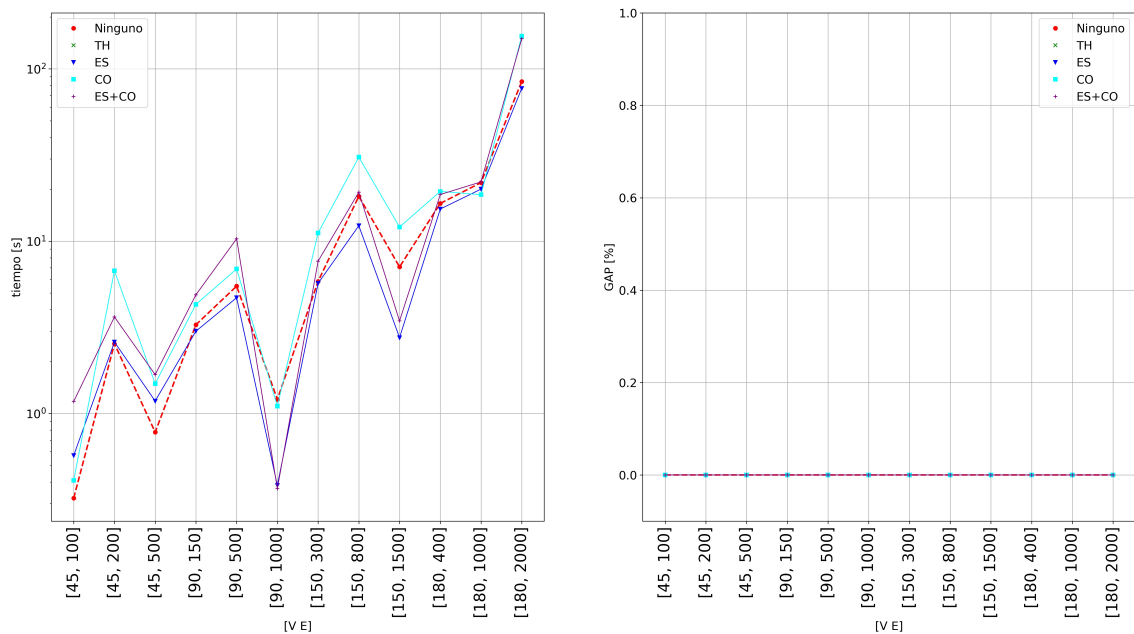


Figura 4.10: Resultados modelo  $\mathcal{F}_3$ ,  $|e_{max}| = 5$ .

En las instancias con  $|e_{max}| = 10$ , se evidencia que el modelo sin planos cortantes posee los mejores resultados a excepción en las instancias  $[90, 500]$ ,  $[150, 800]$  y  $[180, 1000]$ . El plano cortante **CO** es el que presenta menor número de todos en la mayoría de instancias. En la Figura 4.11 se muestra los resultados obtenidos.

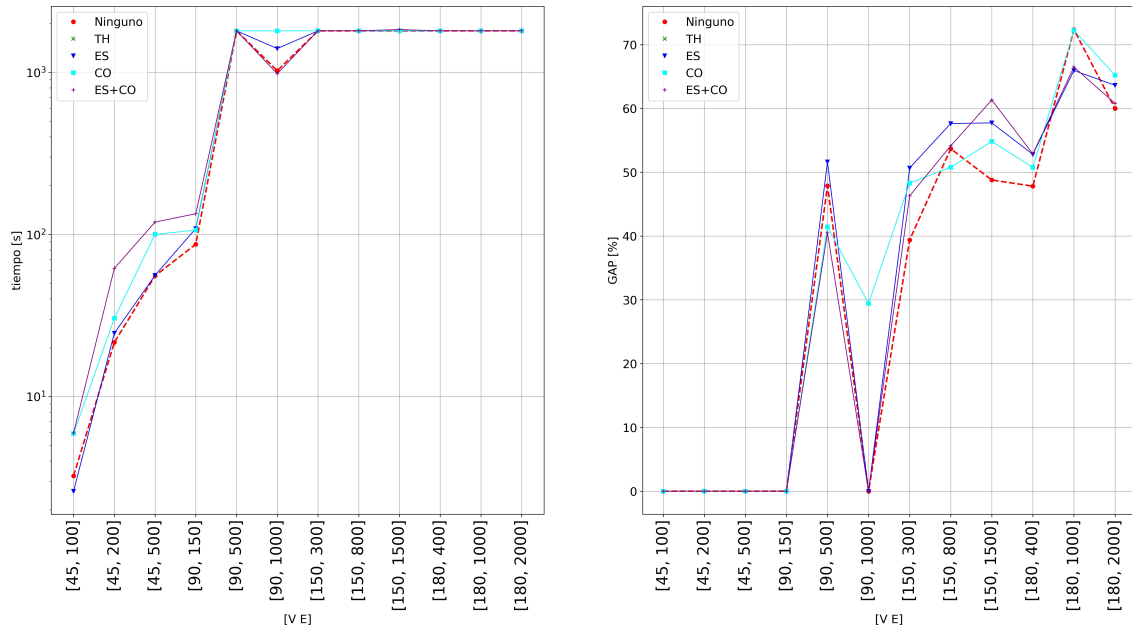


Figura 4.11: Resultados modelo  $\mathcal{F}_3$ ,  $|e_{max}| = 10$ .

Para las instancias con  $|e_{max}| = 20$ , la instancia  $[90, 150]$  fue infactible, mientras que en las instancias  $[150, 300]$  y  $[180, 400]$  no se obtuvo una solución factible en el tiempo máximo de ejecución. En las dos instancias más grandes se observa mejoras con todos planos cortantes, en el resto de instancias las mejoras no son significativas. En la Figura 4.12 se muestra los resultados obtenidos.

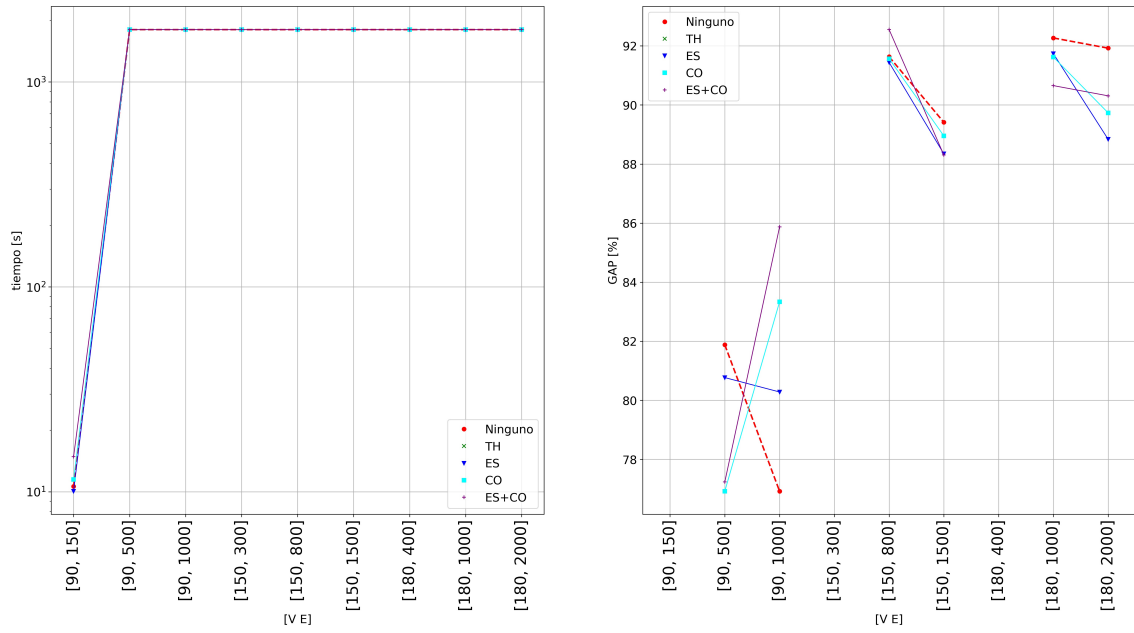


Figura 4.12: Resultados modelo  $\mathcal{F}_3$ ,  $|e_{max}| = 20$ .

#### 4.2.4. Resultados formulaci3n $\mathcal{F}_4$ y $\mathcal{F}_5$

Los resultados completos se presentan en la Tabla A.4 del Anexo A. A continuaci3n se presentan los resultados de las instancias y planos cortantes, se han dividido las instancias por el tama1o m1ximo de hiperarista:

En la Figura 4.13 se presentan las instancias de grafos ( $|e_{max}| = 2$ ), la instancia [150, 300] es infactible y el modelo  $\mathcal{F}_5$  presenta mejores tiempos de resoluci3n. Mientras que no existe diferencia en el gap de optimalidad hasta la instancia [90, 1000], a partir de la instancia [150, 800] el modelo  $\mathcal{F}_5$  es el 1nico que llega a la optimalidad.

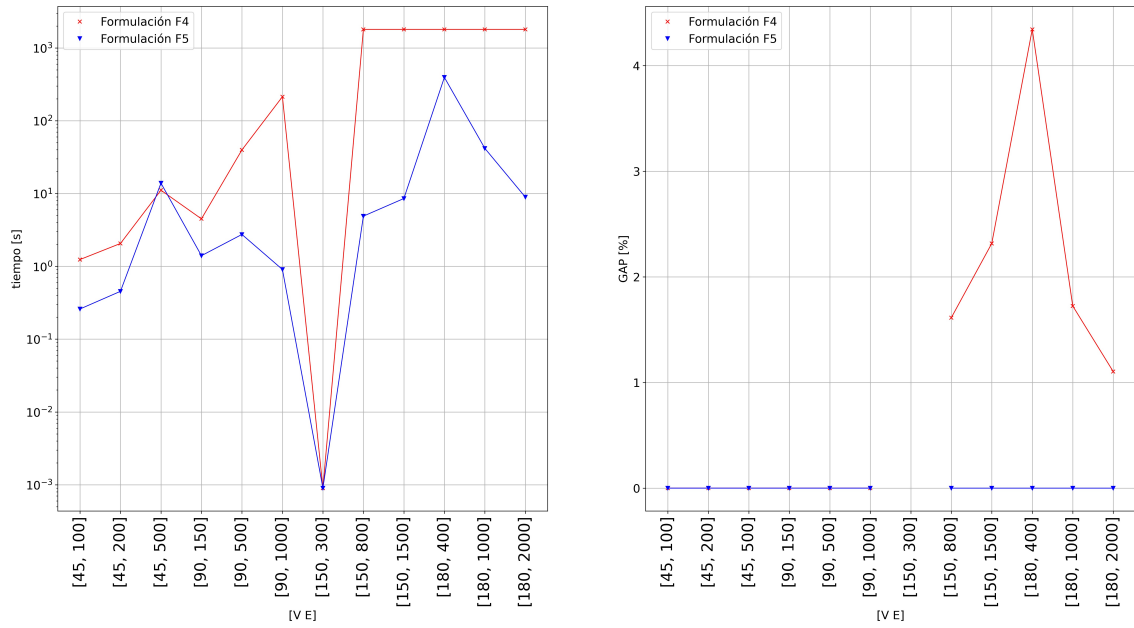


Figura 4.13: Resultados comparación modelos  $\mathcal{F}_4$  y  $\mathcal{F}_5$ ,  $|e_{max}| = 2$ .

En las instancias con  $|e_{max}| = 5$ , se resuelven hasta la optimalidad hasta la instancia [150, 300]. En los tiempos de ejecución los dos modelos muestran resultados similares pero en gap de optimalidad y número de nodos explorados el modelo  $\mathcal{F}_5$  tiene un mejor comportamiento. Los resultados se muestran en la Figura 4.14.

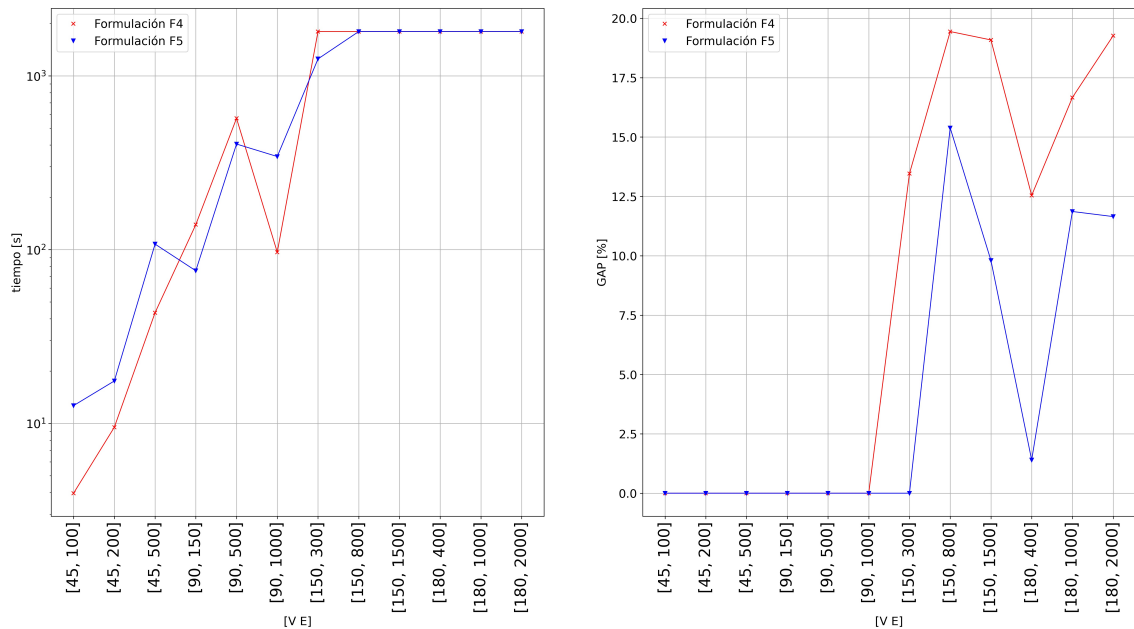


Figura 4.14: Resultados comparación modelos  $\mathcal{F}_4$  y  $\mathcal{F}_5$ ,  $|e_{max}| = 5$ .

En las instancias con  $|e_{max}| = 10$ , la instancia  $[90, 150]$  es infactible y en 7 de las 12 instancias no se obtiene una solución factible en el tiempo máximo de ejecución. En el resto de instancias, el modelo  $\mathcal{F}_4$  es el que presenta mejores tiempos de resolución. En la Figura 4.15 se muestra los resultados obtenidos.

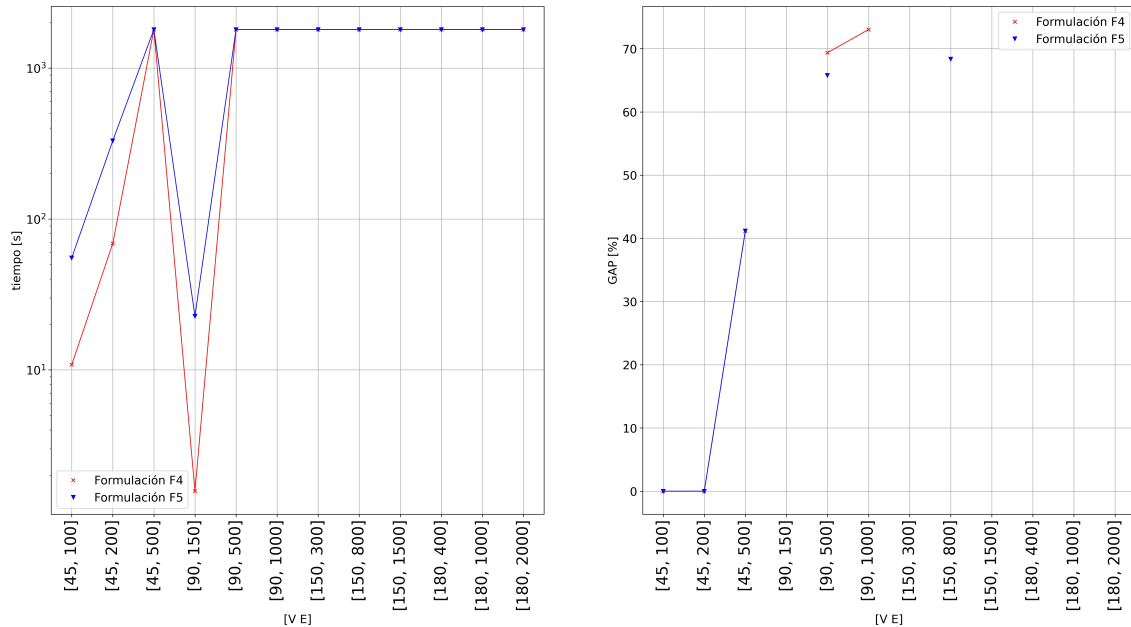


Figura 4.15: Resultados comparación modelos  $\mathcal{F}_4$  y  $\mathcal{F}_5$ ,  $|e_{max}| = 10$ .

Para las instancias con  $|e_{max}| = 20$ , las instancias  $[90, 150]$ ,  $[150, 300]$  y  $[180, 400]$  resultaron infactibles, siendo en la instancia  $[180, 400]$  que se comprueba la infactibilidad con el modelo  $\mathcal{F}_4$  al ser su tiempo menor que el tiempo límite. Las únicas dos instancias en donde se obtuvo un resultado factible fueron obtenidas mediante el modelo  $\mathcal{F}_5$ . En la Figura 4.16 se muestra los resultados obtenidos.

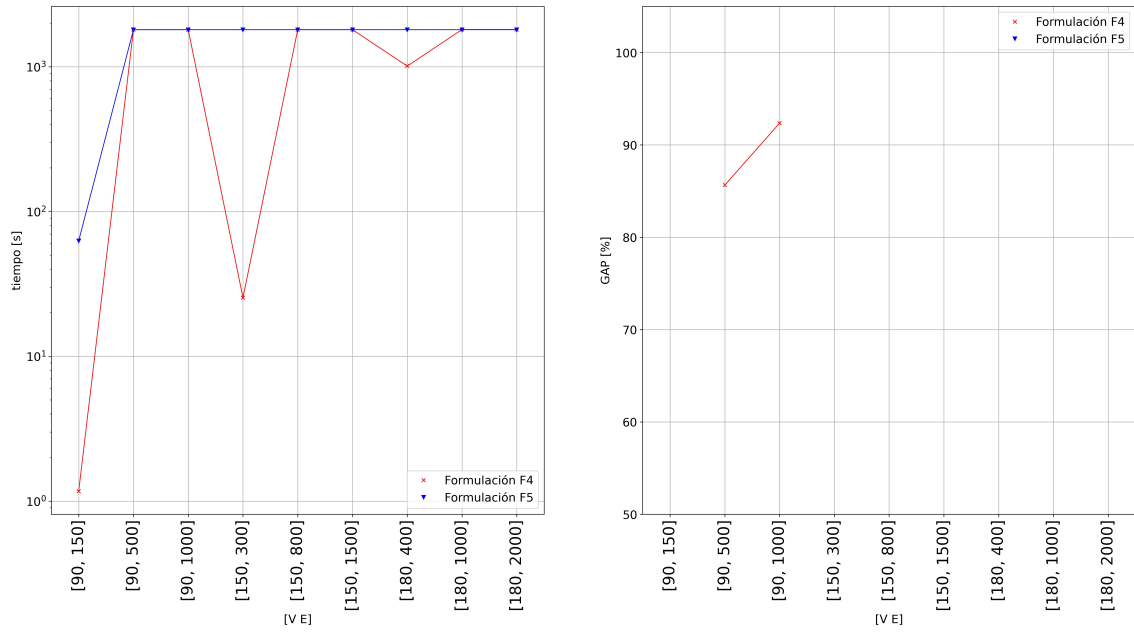


Figura 4.16: Resultados comparación modelos  $\mathcal{F}_4$  y  $\mathcal{F}_5$ ,  $|e_{max}| = 20$ .

#### 4.2.5. Resultados Heurística

Para las pruebas se consideró el caso del biparticionamiento de hipergrafos ( $k = 2$ ). Como se puede observar en los resultados de la Tabla 4.2, los tiempos del algoritmo son muy altos para tratarse de una heurística, además desde las instancias con 150 nodos, no se obtuvo ningún particionamiento factible con tiempos de ejecución menores a 3600 segundos.

Tabla 4.2: Resultados Heurística particionamiento factible.

Instancia (n,m)	$ e_{prom} $	$ e_{max} $	Tiempo [s]	Particionamiento factible
(40,100)	2.00	2	2.17	True
	3.56	5	2.56	True
	6.04	10	5.58	True
	10.99	20	33.54	False
(40,200)	2.00	2	2.11	True
	3.54	5	1.87	True
	5.84	10	3.51	True
	10.18	20	65.70	False
(40,500)	2.00	2	5.28	True
	3.52	5	4.51	True
	5.98	10	4.44	True

	10.80	20	8.49	True
(90,150)	2.00	2	956.49	False
	3.43	5	93.03	True
	5.82	10	1015.58	False
	11.37	20	893.73	False
	18.98	35	912.18	False
(90,500)	2.00	2	45.15	True
	3.48	5	50.93	True
	6.00	10	162.13	True
	10.98	20	3088.25	False
	18.03	35	2998.01	False
(90,1000)	2.00	2	94.13	True
	3.46	5	81.17	True
	6.04	10	59.22	True
	11.02	20	377.88	True
	18.58	35	>3600	False
(150,300)	2.00	2	>3600	False
	3.78	5	>3600	False
	6.08	10	>3600	False
	10.78	20	>3600	False
	28.07	55	>3600	False
(150,800)	2.00	2	>3600	False
	3.53	5	>3600	False
	6.03	10	>3600	False
	10.89	20	>3600	False
	27.69	55	>3600	False
(150,1500)	2.00	2	>3600	False
	3.47	5	>3600	False
	5.90	10	>3600	False
	11.17	20	>3600	False
	28.39	55	>3600	False
(180,400)	2.00	2	>3600	False
	3.55	5	>3600	False
	5.95	10	>3600	False
	11.44	20	>3600	False
	33.51	65	>3600	False
(180,1000)	2.00	2	>3600	False
	3.51	5	>3600	False
	5.99	10	>3600	False
	11.04	20	>3600	False
	33.64	65	>3600	False
(180,2000)	2.00	2	>3600	False
	3.49	5	>3600	False
	6.05	10	>3600	False
	10.88	20	>3600	False
	34.10	65	>3600	False



# Capítulo 5

---

## Conclusiones

---

Aún siendo un problema ampliamente estudiado, el particionamiento de hipergrafos con la función objetivo planteada en este trabajo ofrece nuevas posibilidades de aplicaciones y el desarrollo de técnicas resolutivas, tanto con métodos exactos o de aproximación.

Las diferentes formulaciones mostraron tiempos de resolución bajos para instancias con tamaños de hiperaristas menores o iguales a cinco. Por tal motivo, el uso de las formulaciones de hipergrafos para resolver problemas en grafos se vuelve atractivo. En la mayoría instancias de hipergrafos poco densos resultaron infactibles, esto refleja lo planteado previamente en la Sección 3.5, que en este problema la obtención de particionamientos factibles es más difícil que en otros problemas de particionamiento.

El impacto de los planos cortantes implementados en los modelos, respecto al tiempo de resolución y gap de optimalidad, es mayor para las instancias grandes y en donde el tamaño de hiperarista es mayor. En instancias de grafos (tamaño de hiperarista igual a dos), en cambio, el impacto de los planos cortantes puede llegar a ser negativo respecto a la mejora en el tiempo de resolución.

Al no ser directa la obtención de soluciones factibles a diferencia de otros problemas, donde es posible particionamientos aleatorios iniciales, el uso de heurísticas conocidas de particionamiento de hipergrafos no proporciona una ayuda en la mejora de los Programas Enteros

planteados. A pesar del desarrollo de una heurística, ésta no presentó resultados favorables para obtener particionamientos iniciales factibles en poco tiempo.

Al ser un problema NP-hard, el desarrollo de algoritmos de aproximación es necesario para mejorar la resolución de este problema de particionamiento de hipergrafos, especialmente cuando se trate de instancias grandes y con tamaños de hiperaristas mayores a 5.

Para poder representar problemas del mundo real a través de hipergrafos, en vez de generar hipergrafos aleatorios, es necesario tener información estadística de cada aplicación. Así, en vez de generar instancias con una distribución de probabilidad uniforme en los tamaños de hiperarista, se utilizaría la distribución de probabilidad obtenida en cada problema.

A partir de las formulaciones para el problema del k-equiparticionamiento en componentes conexas considerando soluciones con hiperárboles lineales, se pueden realizar trabajos posteriores para su generalización en donde las componentes del particionamiento sean hipergrafos conexos.

---

## Referencias bibliográficas

---

- [1] Yaroslav Akhremtsev, Tobias Heuer, Peter Sanders, and Sebastian Schlag. Engineering a direct  $k$ -way hypergraph partitioning algorithm. In *2017 Proceedings of the Nineteenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 28–42. SIAM, 2017.
- [2] Giuseppe Alia and Piero Maestrini. An approach to optimal partitioning of hypergraphs. In *Proceedings of the 1974 Annual Conference - Volume 1*, ACM '74, page 133–139, New York, NY, USA, 1974. Association for Computing Machinery. ISBN 9781450374828. doi: 10.1145/800182.810393.
- [3] Calvin Beideman, Karthekeyan Chandrasekaran, and Weihang Wang. Counting and enumerating optimum cut sets for hypergraph  $k$ -partitioning problems for fixed  $k$ , 2023.
- [4] Austin R Benson, Rediet Abebe, Michael T Schaub, Ali Jadbabaie, and Jon Kleinberg. Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*, 115(48):E11221–E11230, 2018.
- [5] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 2013.
- [6] Ümit Çatalyürek, Karen Devine, Marcelo Faraj, Lars Gottesbüren, Tobias Heuer, Henning Meyerhenke, Peter Sanders, Sebastian Schlag, Christian Schulz, Daniel Seemaier, and Dorothea Wagner.

- More recent advances in (hyper)graph partitioning. *ACM Comput. Surv.*, 55(12), mar 2023. ISSN 0360-0300. doi: 10.1145/3571808.
- [7] Der-San Chen, Robert G Batson, and Yu Dang. *Branch-and-Cut Approach*, chapter 12, pages 305–333. John Wiley & Sons, 2009. doi: <https://doi.org/10.1002/9781118166000.ch12>.
- [8] Qionghai Dai and Yue Gao. *Hypergraph Computation*. Springer Nature Singapore, 2023. doi: 10.1007/978-981-99-0185-2.
- [9] C.M. Fiduccia and R.M. Mattheyses. A linear-time heuristic for improving network partitions. In *19th Design Automation Conference*, pages 175–181, 1982. doi: 10.1109/DAC.1982.1585498.
- [10] András Frank, Tamás Király, and Matthias Kriesell. On decomposing a hypergraph into  $k$  connected sub-hypergraphs. *Discrete Applied Mathematics*, 131(2):373–383, September 2003. doi: 10.1016/s0166-218x(02)00463-8.
- [11] Michael R Garey and David S Johnson. *Computers and intractability*, volume 174. freeman San Francisco, 1979.
- [12] Fred Glover. Tabu search—part i. *ORSA Journal on Computing*, 1(3): 190–206, 1989. doi: 10.1287/ijoc.1.3.190.
- [13] Lars Gottesbüren, Michael Hamann, Sebastian Schlag, and Dorothea Wagner. Advanced flow-based multilevel hypergraph partitioning, 2020.
- [14] George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekhar. Multilevel hypergraph partitioning: Application in vlsi domain. In *Proceedings of the 34th annual Design Automation Conference*, pages 526–529, 1997.
- [15] Brian W Kernighan and Shen Lin. An efficient heuristic procedure for partitioning graphs. *The Bell system technical journal*, 49: 291–307, 1970.
- [16] Dorothy Kucar, Shawki Areibi, and Anthony Vannelli. Hypergraph partitioning techniques. *Dynamics of Continuous Discrete and Impulsive Systems Series A*, 11:339–368, 2004.

- [17] Eugene L. Lawler. Cutsets and partitions of hypergraphs. Technical Report UCB/ERL M340, EECS Department, University of California, Berkeley, May 1972. URL <http://www2.eecs.berkeley.edu/Pubs/TechRpts/1972/28707.html>.
- [18] Geon Lee, Minyoung Choe, and Kijung Shin. How do hyperedges overlap in real-world hypergraphs?-patterns, measures, and generators. In *Proceedings of the Web Conference 2021*, pages 3396–3407, 2021.
- [19] Jiří Matoušek and Bernd Gärtner. *Understanding and using linear programming*, volume 33. Springer, 2007.
- [20] Shota Saito and Mark Herbster. Generalizing p-laplacian: spectral hypergraph theory and a partitioning algorithm. *Machine Learning*, 112(1):241–280, November 2022. doi: 10.1007/s10994-022-06264-y.
- [21] Daniel G Schweikert and Brian W Kernighan. A proper model for the partitioning of electrical circuits. In *Proceedings of the 9th Design Automation Workshop*, pages 57–62, 1972.

# Capítulo A

## Tablas de Resultados

Para las diferentes tablas, en la columna de planos cortantes añadidos el símbolo (\*) indica que los planos fueron añadidos en el nodo raíz del árbol de búsqueda. Adicionalmente, para las diferentes instancias se resaltan los planos cortantes con las mayores mejoras respecto a sus valores sin aplicar ningún plano cortante.

### A.1. Formulación $\mathcal{F}_1$

Tabla A.1: Resultados computacionales modelo  $\mathcal{F}_1$ .

Instancia (n,m)	$ e_{prom} $	$ e_{max} $	Plano cortante	Obj	Gap [%]	Nodos B&C	Planos Añad.	Tiempo [s]
2.00	2		Ninguno	153	0	1	-	0.05
			TH	153	0	1	-	0.02
			ES	153	0	1	3*	0.01
			KS	153	0	1	0	0.01
			ES+CO	153	0	1	0	0.01
			ES+KS	153	0	1	0	0.01
			CO+KS	153	0	1	0	0.01
(45,100)	3.52	5	Ninguno	86	0	1836	-	0.69
			ES	86	0	<b>1831</b>	3*	1.68
			CO	86	0	4256	54	2.17
			KS	86	0	1836	0	1.07
			ES+CO	86	0	2727	48	1.56
			ES+KS	86	0	<b>1831</b>	0	2.36
			CO+KS	86	0	4256	54	2.98

			Ninguno	147	0	1543	-	3.33
			ES	147	0	1503	3*	<b>2.96</b>
			CO	147	0	1778	17	4.19
			KS	147	0	1675	74	51.70
			ES+CO	147	0	2749	22	3.72
			ES+KS	147	0	4361	305	103.15
			CO+KS	147	0	1554	131	48.92
			Ninguno	90	0	1	-	0.03
			ES	90	0	1	3*	0.02
			KS	90	0	1	0	0.03
			ES+CO	90	0	1	0	0.02
			ES+KS	90	0	1	0	0.01
			CO+KS	90	0	1	0	0.04
			Ninguno	43	0	1291	-	1.21
			ES	43	0	1596	3*	<b>0.79</b>
			CO	43	0	2496	57	2.46
			KS	43	0	1291	0	1.97
			ES+CO	43	0	1623	35	1.70
			ES+KS	43	0	1596	0	1.34
			CO+KS	43	0	2496	57	2.40
			Ninguno	54	0	11572	-	18.84
			ES	54	0	<b>7211</b>	3*	<b>13.92</b>
			CO	54	0	36719	151	72.77
			KS	54	0	19530	62	1351.71
			ES+CO	54	0	19436	110	36.02
			ES+KS	54	0	25932	184	1675.56
			CO+KS	54	0	19093	148	1253.60
			Ninguno	49	0	1	-	0.04
			ES	49	0	1	3*	0.07
			KS	49	0	1	0	0.04
			ES+CO	49	0	1	0	0.07
			ES+KS	49	0	1	0	0.07
			CO+KS	49	0	1	0	0.04
			Ninguno	30	0	4591	-	3.85
			ES	30	0	<b>1781</b>	3*	<b>1.78</b>
			CO	30	0	4258	128	11.34
			KS	30	0	4591	0	8.06
			ES+CO	30	0	<b>1886</b>	66	5.10
			ES+KS	30	0	<b>1781</b>	0	3.34
			CO+KS	30	0	4258	128	10.90
			Ninguno	18	0	1892	-	9.38
			ES	18	0	2839	3*	<b>8.21</b>
			CO	18	0	4208	320	31.80
			KS	18	17	3262	5	1808.94
			ES+CO	18	0	7093	242	22.94
			ES+KS	18	0	<b>1688</b>	6	1072.40
			CO+KS	22	41	2796	236	1808.33
			Ninguno	386	0	1	-	0.05
			TH	386	0	1	-	0.03
			ES	386	0	1	3*	0.02
			KS	386	0	1	0	0.05
			ES+CO	386	0	1	0	0.02

		ES+KS	386	0	1	0	0.02
		CO+KS	386	0	1	0	0.05
		Ninguno	203	0	3988	-	3.38
3.47	5	ES	203	0	<b>3503</b>	3*	<b>1.66</b>
		CO	203	0	4134	18	5.34
		KS	203	0	3988	0	4.75
		ES+CO	203	0	<b>3380</b>	13	<b>2.87</b>
		ES+KS	203	0	<b>3503</b>	0	<b>2.88</b>
		CO+KS	203	0	4134	18	5.19
		Ninguno	285	0	7296	-	56.84
6.04	10	ES	285	0	15803	3*	121.16
		CO	285	0	<b>7292</b>	15	64.52
		KS	285	0	7296	0	63.30
		ES+CO	285	0	17544	11	159.12
		ES+KS	285	0	15803	0	129.20
		CO+KS	285	0	<b>7292</b>	15	64.88
		Ninguno	-	-	-	-	2.75
11.03	20	ES	-	-	-	-	0.55
		CO	-	-	-	-	4.00
		KS	-	-	-	-	149.83
		ES+CO	-	-	-	-	0.65
		ES+KS	-	-	-	-	20.57
		CO+KS	-	-	-	-	148.05
		Ninguno	144	0	1	-	0.03
2.00	2	ES	144	0	1	3*	0.03
		KS	144	0	1	0	0.03
		ES+CO	144	0	1	0	0.04
		ES+KS	144	0	1	0	0.03
		CO+KS	144	0	1	0	0.03
		Ninguno	63	0	1841	-	7.37
3.50	5	ES	63	0	<b>1722</b>	3*	<b>3.86</b>
		CO	63	0	2381	51	11.06
		KS	63	0	1841	0	9.87
		ES+CO	63	0	2981	45	9.92
		ES+KS	63	0	<b>1722</b>	0	<b>6.26</b>
		CO+KS	63	0	2381	51	10.93
		Ninguno	68	24	445282	-	1800.04
5.85	10	ES	72	29	<b>404803</b>	3*	1800.02
		CO	69	<b>23</b>	<b>317414</b>	862	1800.65
		KS	68	25	339739	0	1800.11
		ES+CO	72	32	305395	4267	1801.11
		ES+KS	73	32	320927	0	1800.66
		CO+KS	69	<b>23</b>	<b>317030</b>	861	1800.23
		Ninguno	146	84	72870	-	1800.04
10.53	20	ES	161	85	108844	3*	1800.04
		CO	144	85	<b>59702</b>	959	1801.50
		KS	203	92	2761	57	1812.02
		ES+CO	168	87	65096	816	1801.08
		ES+KS	186	91	2806	9	1828.31
		CO+KS	197	92	2684	97	1808.98
		Ninguno	65	0	1	-	0.06
2.00	2						



		ES	65	0	1	3*	0.13
		KS	65	0	1	0	0.06
		ES+CO	65	0	1	0	0.13
		ES+KS	65	0	1	0	0.12
		CO+KS	65	0	1	0	0.06
		Ninguno	44	0	5839	-	23.07
3.53	5	ES	44	0	3559	3*	17.37
		CO	44	0	2400	177	19.77
		KS	44	0	3077	0	24.89
		ES+CO	44	0	3732	340	31.35
		ES+KS	44	0	3559	0	26.19
		CO+KS	44	0	2400	177	19.99
		Ninguno	47	38	146906	-	1800.05
5.92	10	ES	44	<b>39</b>	<b>103586</b>	3*	1800.05
		CO	48	42	81296	370	1801.66
		KS	47	40	115069	0	1801.47
		ES+CO	47	43	92121	391	1801.70
		ES+KS	46	41	88251	0	1801.46
		CO+KS	48	42	80956	368	1801.08
		Ninguno	82	84	38187	-	1800.08
11.00	20	ES	100	85	39797	3*	1800.07
		CO	80	<b>84</b>	<b>30314</b>	906	1802.32
		KS	187	95	795	0	1869.89
		ES+CO	87	<b>84</b>	<b>36632</b>	1239	1800.38
		ES+KS	154	94	807	14	1896.17
		CO+KS	199	95	816	351	1897.46
		Ninguno	-	-	-	-	0.00
2.00	2	TH	-	-	-	-	0.00
		ES	-	-	-	-	0.00
		KS	-	-	-	-	0.00
		ES+CO	-	-	-	-	0.00
		ES+KS	-	-	-	-	0.00
		CO+KS	-	-	-	-	0.00
(150,300)		Ninguno	236	0	2344	-	7.92
3.56	5	ES	236	0	2649	3*	7.13
		CO	236	0	2293	15	9.73
		KS	236	0	2344	0	10.04
		ES+CO	236	0	<b>2197</b>	47	8.09
		ES+KS	236	0	2649	0	9.28
		CO+KS	236	0	<b>2293</b>	15	9.34
		Ninguno	342	59	144235	-	1800.13
6.37	10	ES	328	<b>58</b>	<b>111848</b>	3*	1800.08
		CO	325	57	127590	271	1801.03
		KS	315	55	141948	0	1801.19
		ES+CO	364	62	97250	128	1800.59
		ES+KS	328	58	104385	0	1800.39
		CO+KS	317	<b>56</b>	<b>128994</b>	273	1800.30
		Ninguno	-	-	-	-	1800.02
10.77	20	ES	-	-	-	-	1800.03
		CO	-	-	-	-	1800.35
		KS	-	-	-	-	1800.75
		ES+CO	-	-	-	-	1800.07
		ES+KS	-	-	-	-	1800.11

			CO+KS	-	-	-	-	1800.04
(150,800)	2.00	2	Ninguno	219	0	1	-	0.11
			ES	219	0	1	3*	0.05
			KS	219	0	1	0	0.11
			ES+CO	219	0	1	0	0.05
			ES+KS	219	0	1	0	0.05
			CO+KS	219	0	1	0	0.11
	3.49	5	Ninguno	114	0	1792	-	13.04
			ES	114	0	2160	3*	<b>11.76</b>
			CO	114	0	2297	39	21.85
			KS	114	0	1792	0	16.29
			ES+CO	114	0	2210	43	17.43
			ES+KS	114	0	2160	0	16.30
	6.04	10	Ninguno	166	63	96830	-	1800.05
			ES	162	63	93069	3*	1800.07
			CO	168	62	101861	1195	1802.25
KS			166	<b>63</b>	<b>86671</b>	0	1800.58	
ES+CO			150	<b>58</b>	103376	970	1800.59	
ES+KS			162	63	81939	0	1800.66	
10.86	20	Ninguno	308	91	10671	-	1800.06	
		ES	299	91	11557	3*	1800.03	
		CO	323	91	10854	185	1800.22	
		KS	308	91	<b>9929</b>	0	1800.14	
		ES+CO	321	91	11507	165	1801.34	
		ES+KS	299	91	10827	0	1801.42	
(150,1500)	2.00	2	Ninguno	162	0	1	-	0.16
			ES	162	0	1	3*	0.22
			KS	162	0	1	0	0.14
			ES+CO	162	0	1	0	0.23
			ES+KS	162	0	1	0	0.23
			CO+KS	162	0	1	0	0.14
	3.48	5	Ninguno	76	0	4915	-	28.90
			ES	76	0	7621	3*	49.29
			CO	76	0	10831	135	90.55
			KS	76	0	4915	0	48.37
			ES+CO	76	0	9851	255	122.53
			ES+KS	76	0	7621	0	78.94
	5.99	10	Ninguno	81	48	38123	-	1800.09
			ES	87	53	44739	3*	1800.04
			CO	109	62	29350	534	1800.44
KS			91	54	33589	0	1801.10	
ES+CO			94	56	32861	462	1800.27	
ES+KS			92	55	38388	0	1800.26	
11.22	20	Ninguno	191	92	6410	-	1800.06	
		ES	167	92	6684	3*	1800.12	
		CO	198	92	7006	513	1800.14	

			KS	191	92	<b>5099</b>	2	1807.07
			ES+CO	178	92	7321	522	1801.88
			ES+KS	207	93	5341	0	1804.29
			CO+KS	198	92	5283	491	1804.88
			Ninguno	563	0	1	-	0.06
	2.00	2	TH	563	0	1	-	0.06
			ES	563	0	1	3*	0.04
			KS	563	0	1	0	0.06
			ES+CO	563	0	1	0	0.05
			ES+KS	563	0	1	0	0.04
			CO+KS	563	0	1	0	0.06
			Ninguno	261	0	3774	-	17.61
(180,400)	3.61	5	ES	261	0	2176	3*	10.50
			CO	261	0	3774	30	22.16
			KS	261	0	3774	0	21.72
			ES+CO	261	0	<b>1823</b>	6	<b>10.40</b>
			ES+KS	261	0	2176	0	11.96
			CO+KS	261	0	3774	30	21.93
			Ninguno	388	65	122973	-	1800.04
	6.05	10	ES	367	63	114538	3*	1800.04
			CO	370	64	95673	203	1800.46
			KS	374	63	124666	0	1800.08
			ES+CO	300	<b>55</b>	<b>120536</b>	261	1800.32
			ES+KS	367	63	108393	0	1800.80
			CO+KS	370	64	94623	203	1800.19
			Ninguno	-	-	-	-	1800.03
	10.45	20	ES	-	-	-	-	1800.05
			CO	-	-	-	-	1800.54
			KS	-	-	-	-	1800.05
			ES+CO	-	-	-	-	1800.63
			ES+KS	-	-	-	-	1800.25
			CO+KS	-	-	-	-	1800.76
			Ninguno	270	0	1	-	0.12
	2.00	2	ES	270	0	1	3*	0.09
			KS	270	0	1	0	0.11
			ES+CO	270	0	1	0	0.08
			ES+KS	270	0	1	0	0.08
			CO+KS	270	0	1	0	0.11
			Ninguno	126	0	3743	-	29.63
(180,1000)	3.49	5	ES	126	0	4026	3*	34.43
			CO	126	0	2539	161	31.07
			KS	126	0	2242	0	26.95
			ES+CO	126	0	<b>1665</b>	113	<b>19.78</b>
			ES+KS	126	0	4026	0	46.21
			CO+KS	126	0	2539	161	30.12
			Ninguno	225	68	55102	-	1800.09
	6.08	10	ES	207	67	59905	3*	1800.07
			CO	207	<b>65</b>	<b>49343</b>	266	1801.29
			KS	225	68	54553	0	1800.90
			ES+CO	209	67	51151	318	1803.10
			ES+KS	207	67	57137	0	1800.84
			CO+KS	207	<b>65</b>	49974	277	1801.50



			ES	60	0	1420	3*	0.39
			CO	60	0	499	2	0.57
			ES+CO	60	0	860	6	0.68
			Ninguno	133	0	3475	-	4.46
6.20	10.00		ES	133	0	2839	3*	4.29
			CO	133	0	3467	32	6.86
			ES+CO	133	0	<b>2695</b>	30	5.28
			Ninguno	103	0	1	-	0.02
2.00	2.00		ES	103	0	1	3*	0.02
			ES+CO	103	0	1	3	0.02
(45,200)			Ninguno	44	0	1904	-	1.08
3.63	5.00		ES	44	0	3298	3*	1.41
			CO	44	0	2365	54	2.70
			ES+CO	44	0	3066	25	3.16
			Ninguno	38	0	3369	-	7.95
6.03	10.00		ES	38	0	3629	3*	<b>7.15</b>
			CO	38	0	7178	103	19.67
			ES+CO	38	0	3923	125	12.44
			Ninguno	40	0	1	-	0.04
2.00	2.00		ES	40	0	1	3*	0.03
			ES+CO	40	0	1	3	0.03
(45,500)			Ninguno	18	0	66	-	0.43
3.49	5.00		ES	18	0	334	3*	0.66
			CO	18	0	66	0	0.60
			ES+CO	18	0	334	36	1.56
			Ninguno	19	0	11395	-	46.72
5.94	10.00		ES	19	0	<b>5491</b>	3*	<b>21.39</b>
			CO	19	0	16679	91	127.84
			ES+CO	19	0	14083	49	87.36
			Ninguno	-	-	-	-	0.00
2.00	2.00		TH	-	-	-	-	0.00
			ES	-	-	-	-	0.00
			ES+CO	-	-	-	-	0.00
(90,150)			Ninguno	187	0	1957	-	2.08
3.61	5.00		ES	187	0	6442	3*	1.99
			CO	187	0	<b>1668</b>	25	2.80
			ES+CO	187	0	2698	22	3.57
			Ninguno	153	0	70041	-	112.99
5.53	10.00		ES	153	0	98535	3*	154.54
			CO	153	0	70170	88	151.33
			ES+CO	153	0	106387	180	260.95
			Ninguno	-	-	-	-	16.92
10.87	20.00		ES	-	-	-	-	16.99
			CO	-	-	-	-	17.02

			ES+CO	-	-	-	-	15.69
			Ninguno	137	0	1	-	0.03
	2.00	2.00	ES	137	0	1	3*	0.05
			ES+CO	137	0	1	3	0.06
(90,500)			Ninguno	67	0	6990	-	6.50
	3.43	5.00	ES	67	0	7276	3*	6.05
			CO	67	0	<b>3291</b>	63	11.83
			ES+CO	67	0	7280	178	14.98
	5.95	10.00	Ninguno	44	0	25306	-	140.42
ES			44	0	35468	3*	188.24	
CO			44	0	<b>24302</b>	140	188.51	
		ES+CO	44	0	38358	136	299.66	
10.74	20.00	Ninguno	101	71	185045	-	1800.05	
		ES	109	72	150609	3*	1800.05	
		CO	101	70	142139	2311	1801.35	
		ES+CO	86	<b>65</b>	<b>153252</b>	2258	1800.33	
(90,1000)			Ninguno	79	0	1	-	0.06
	2.00	2.00	ES	79	0	1	3*	0.06
			ES+CO	79	0	1	3	0.06
	3.46	5.00	Ninguno	41	0	392	-	3.21
			ES	41	0	256	3*	1.32
			CO	41	0	392	11	4.62
			ES+CO	41	0	256	5	1.95
	6.12	10.00	Ninguno	41	22	189195	-	1800.07
			ES	36	0	230831	3*	1781.43
			CO	39	23	<b>93530</b>	805	1801.61
		ES+CO	39	<b>21</b>	101871	358	1801.12	
11.20	20.00	Ninguno	72	83	53940	-	1800.40	
		ES	80	86	42800	3*	1800.05	
		CO	82	87	31586	655	1801.31	
		ES+CO	90	88	41368	682	1801.98	
(150,300)			Ninguno	-	-	-	-	0.00
	2.00	2.00	TH	-	-	-	-	0.00
			ES	-	-	-	-	0.00
			ES+CO	-	-	-	-	0.00
	3.58	5.00	Ninguno	249	0	1459	-	5.50
			ES	249	0	1511	3*	<b>3.51</b>
			CO	249	0	1689	14	7.73
			ES+CO	249	0	1478	5	4.75
	5.92	10.00	Ninguno	262	42	205531	-	1800.11
			ES	279	47	220793	3*	1800.11
CO			281	46	170230	304	1800.12	
		ES+CO	258	42	220450	275	1801.51	

			Ninguno	-	-	-	-	1800.02
	11.35	20.00	ES	-	-	-	-	1800.03
			CO	-	-	-	-	1800.04
			ES+CO	-	-	-	-	1800.68
			Ninguno	271	0	1	-	0.05
	2.00	2.00	ES	271	0	1	3*	0.04
			ES+CO	271	0	1	3	0.04
			Ninguno	124	0	3386	-	20.74
(150,800)	3.49	5.00	ES	124	0	<b>1664</b>	3*	<b>8.56</b>
			CO	124	0	2052	22	16.30
			ES+CO	124	0	1727	28	14.53
			Ninguno	127	39	108569	-	1800.06
	5.88	10.00	ES	125	37	157788	3*	1800.06
			CO	111	31	106217	1469	1800.89
			ES+CO	127	38	121158	2228	1801.30
			Ninguno	239	87	14969	-	1800.07
	10.98	20.00	ES	287	89	14730	3*	1800.07
			CO	260	88	13075	160	1800.46
			ES+CO	294	89	15466	153	1800.25
			Ninguno	156	0	1	-	0.11
	2.00	2.00	ES	156	0	1	3*	0.10
			ES+CO	156	0	1	3	0.10
			Ninguno	72	0	4305	-	19.32
(150,1500)	3.51	5.00	ES	72	0	<b>521</b>	3*	<b>11.15</b>
			CO	72	0	5443	278	38.98
			ES+CO	72	0	<b>521</b>	14	12.72
			Ninguno	77	49	39489	-	1800.09
	5.88	10.00	ES	97	61	31976	3*	1800.08
			CO	81	52	27387	434	1813.06
			ES+CO	97	61	33513	553	1801.11
			Ninguno	164	89	8346	-	1800.07
	10.95	20.00	ES	149	<b>87</b>	11085	3*	1800.21
			CO	200	91	9297	1134	1802.03
			ES+CO	154	88	9880	486	1800.84
			Ninguno	625	0	1	-	0.04
	2.00	2.00	TH	625	0	1	13*	0.04
			ES	625	0	1	3*	0.04
			ES+CO	625	0	1	3	0.04
			Ninguno	264	0	1958	-	2.53
(180,400)	3.49	5.00	ES	264	0	2179	3*	2.67
			CO	264	0	<b>1550</b>	36	4.03
			ES+CO	264	0	2565	66	5.13
			Ninguno	315	50	152303	-	1800.03
	6.08	10.00	ES	324	51	138452	3*	1800.03

			CO	343	54	131094	299	1800.04
			ES+CO	313	50	<b>110072</b>	200	1800.80
			Ninguno	-	-	-	-	1800.03
	10.96	20.00	ES	-	-	-	-	1800.04
			CO	-	-	-	-	1800.27
			ES+CO	-	-	-	-	1800.68
			Ninguno	303	0	1	-	0.06
	2.00	2.00	ES	303	0	1	3*	0.05
			ES+CO	303	0	1	3	0.05
			Ninguno	130	0	73	-	2.70
(180,1000)	3.51	5.00	ES	130	0	1797	3*	8.98
			CO	130	0	73	2	2.95
			ES+CO	130	0	1524	89	17.90
			Ninguno	183	61	48350	-	1800.37
	6.01	10.00	ES	143	<b>48</b>	72063	3*	1800.06
			CO	162	54	54494	1771	1811.18
			ES+CO	182	60	61843	1708	1802.45
			Ninguno	287	90	9973	-	1800.07
	10.90	20.00	ES	288	<b>89</b>	10234	3*	1800.14
			CO	353	92	8145	241	1800.30
			ES+CO	300	<b>89</b>	10328	207	1800.09
			Ninguno	190	0	1	-	0.13
	2.00	2.00	ES	190	0	1	3*	0.10
			ES+CO	190	0	1	3	0.10
			Ninguno	87	0	113	-	11.78
(180,2000)	3.52	5.00	ES	87	0	202	3*	<b>5.64</b>
			CO	87	0	113	8	12.14
			ES+CO	87	0	202	19	7.36
			Ninguno	108	59	17393	-	1800.07
	6.00	10.00	ES	115	61	16304	3*	1800.05
			CO	99	<b>56</b>	<b>14271</b>	505	1814.69
			ES+CO	105	58	15020	300	1801.62
			Ninguno	191	87	6175	-	1800.12
	11.21	20.00	ES	213	89	6671	3*	1800.09
			CO	148	<b>85</b>	<b>5645</b>	327	1800.15
			ES+CO	199	88	6197	1093	1800.33

### A.3. Formulación $\mathcal{F}_3$



Tabla A.3: Resultados computacionales modelo  $\mathcal{F}_3$ .

Instancia (n,m)	Plano		Obj	Gap [%]	Nodos B&C	Planos Añad.	Tiempo [s]	
	$ e_{\text{prom}} $	$ e_{\text{max}} $						cortante
(45,100)	2.00	2.00	Ninguno	140	0	1	-	0.01
			TH	140	0	1	1*	0.01
			ES	140	0	1	3*	0.01
			ES+CO	140	0	1	3	0.02
	3.27	5.00	Ninguno	110	0	288	-	0.32
			ES	110	0	2633	3*	0.57
			CO	110	0	288	7	0.41
			ES+CO	110	0	2241	46	1.17
	5.42	10.00	Ninguno	88	0	4423	-	3.24
			ES	88	0	<b>3269</b>	3*	2.61
			CO	88	0	5644	203	5.91
			ES+CO	88	0	5894	208	5.96
(45,200)	2.00	2.00	Ninguno	96	0	1	-	0.01
			ES	96	0	1	3*	0.01
			ES+CO	96	0	1	3	0.01
	3.46	5.00	Ninguno	50	0	7317	-	2.54
			ES	50	0	2803	3*	2.60
			CO	50	0	7698	130	6.74
			ES+CO	50	0	<b>2177</b>	21	3.63
	6.05	10.00	Ninguno	51	0	13100	-	21.56
			ES	51	0	19904	3*	24.57
			CO	51	0	<b>10976</b>	177	30.36
			ES+CO	51	0	28319	273	61.70
	(45,500)	2.00	2.00	Ninguno	42	0	1	-
ES				42	0	1	3*	0.03
ES+CO				42	0	1	3	0.03
3.54		5.00	Ninguno	20	0	313	-	0.78
			ES	20	0	<b>249</b>	3*	1.18
			CO	20	0	313	12	1.49
			ES+CO	20	0	<b>249</b>	31	1.68
6.13		10.00	Ninguno	26	0	17487	-	55.51
			ES	26	0	<b>16515</b>	3*	55.90
			CO	26	0	<b>14413</b>	199	100.01
			ES+CO	26	0	24337	393	119.11
(90,150)		2.00	2.00	Ninguno	-	-	-	-
	TH			-	-	-	-	0.00
	ES			-	-	-	-	0.00
	ES+CO			-	-	-	-	0.00
	3.63	5.00	Ninguno	178	0	3242	-	3.27
			ES	178	0	3003	3*	<b>3.00</b>

			CO	178	0	<b>2999</b>	36	4.30
			ES+CO	178	0	3382	43	4.88
			Ninguno	185	0	37433	-	86.99
5.77	10.00		ES	185	0	55233	3*	108.68
			CO	185	0	37433	9	106.35
			ES+CO	185	0	59117	48	134.08
			Ninguno	-	-	-	-	10.63
11.49	20.00		ES	-	-	-	-	10.07
			CO	-	-	-	-	11.48
			ES+CO	-	-	-	-	14.85
			Ninguno	113	0	1	-	0.03
2.00	2.00		ES	113	0	1	3*	0.05
			ES+CO	113	0	1	3	0.07
			Ninguno	68	0	1782	-	5.48
(90,500)	3.46	5.00	ES	68	0	3901	3*	<b>4.69</b>
			CO	68	0	<b>1593</b>	27	6.89
			ES+CO	68	0	3272	74	10.33
			Ninguno	94	48	428971	-	1800.04
6.09	10.00		ES	93	52	243914	3*	1800.05
			CO	82	<b>41</b>	<b>186454</b>	962	1800.06
			ES+CO	79	<b>41</b>	271225	2124	1800.81
			Ninguno	149	82	118609	-	1800.04
10.80	20.00		ES	130	81	145141	3*	1800.04
			CO	130	<b>77</b>	<b>107165</b>	2563	1800.12
			ES+CO	123	<b>77</b>	115856	1473	1801.06
			Ninguno	103	0	1	-	0.07
2.00	2.00		ES	103	0	1	3*	0.05
			ES+CO	103	0	1	3	0.05
			Ninguno	46	0	1	-	1.21
(,90,1000)	3.51	5.00	ES	46	0	1	3*	0.38
			CO	46	0	1	0	1.10
			ES+CO	46	0	1	3	0.37
			Ninguno	28	0	77046	-	1022.41
5.99	10.00		ES	28	0	105551	3*	1399.98
			CO	34	29	95199	727	1800.19
			ES+CO	28	0	<b>50429</b>	204	<b>980.11</b>
			Ninguno	52	77	35844	-	1800.08
11.12	20.00		ES	71	80	36910	3*	1800.07
			CO	72	83	26435	709	1800.12
			ES+CO	92	86	31305	1077	1802.06
			Ninguno	-	-	-	-	0.00
2.00	2.00		TH	-	-	-	-	0.00
			ES	-	-	-	-	0.00
			ES+CO	-	-	-	-	0.00

			Ninguno	301	0	1928	-	5.84
	3.47	5.00	ES	301	0	1959	3*	<b>5.69</b>
			CO	301	0	2854	43	11.16
			ES+CO	301	0	2196	42	7.65
			Ninguno	259	39	225036	-	1800.11
	5.94	10.00	ES	302	51	218395	3*	1800.09
			CO	294	48	169421	1251	1801.77
			ES+CO	283	46	219798	1171	1801.33
			Ninguno	-	-	-	-	1800.03
	10.94	20.00	ES	-	-	-	-	1800.02
			CO	-	-	-	-	1800.50
			ES+CO	-	-	-	-	1800.17
			Ninguno	251	0	1	-	0.05
	2.00	2.00	ES	251	0	1	3*	0.04
			ES+CO	251	0	1	3	0.04
			Ninguno	131	0	2179	-	18.21
(150,800)	3.49	5.00	ES	131	0	2962	3*	<b>12.29</b>
			CO	131	0	2807	62	30.73
			ES+CO	131	0	2570	80	19.23
			Ninguno	136	54	103402	-	1800.07
	5.98	10.00	ES	144	58	101934	3*	1800.04
			CO	128	<b>51</b>	103161	1492	1800.13
			ES+CO	133	54	113201	2201	1801.30
			Ninguno	251	92	12301	-	1800.06
	11.03	20.00	ES	257	<b>91</b>	20737	3*	1800.08
			CO	249	92	13287	209	1801.63
			ES+CO	282	93	14224	166	1801.13
			Ninguno	175	0	1	-	0.08
	2.00	2.00	ES	175	0	1	3*	0.08
			ES+CO	175	0	1	3	0.08
			Ninguno	80	0	859	-	7.09
(150,1500)	3.50	5.00	ES	80	0	<b>72</b>	3*	<b>2.75</b>
			CO	80	0	925	40	12.08
			ES+CO	80	0	<b>72</b>	7	3.44
			Ninguno	82	49	36822	-	1800.07
	5.91	10.00	ES	97	58	40603	3*	1800.11
			CO	93	55	28848	800	1802.75
			ES+CO	106	61	30995	719	1834.49
			Ninguno	170	89	7860	-	1800.06
	11.01	20.00	ES	163	<b>88</b>	12161	3*	1800.07
			CO	172	89	8353	549	1800.24
			ES+CO	171	<b>88</b>	7900	747	1800.83
			Ninguno	580	0	1	-	0.05
	2.00	2.00	TH	580	0	1	11*	0.05

			ES	580	0	1	3*	0.05
			ES+CO	580	0	1	3	0.05
			Ninguno	321	0	2774	-	16.56
3.47	5.00		ES	321	0	<b>2135</b>	3*	<b>15.36</b>
			CO	321	0	2421	6	19.44
			ES+CO	321	0	2135	13	18.65
			Ninguno	320	48	68092	-	1800.05
5.75	10.00		ES	356	53	71162	3*	1800.03
			CO	337	51	58161	129	1801.45
			ES+CO	348	53	59686	122	1801.22
			Ninguno	-	-	-	-	1800.06
11.58	20.00		ES	-	-	-	-	1800.06
			CO	-	-	-	-	1800.12
			ES+CO	-	-	-	-	1800.10
			Ninguno	306	0	1	-	0.06
2.00	2.00		ES	306	0	1	3*	0.06
			ES+CO	306	0	1	3	0.06
			Ninguno	139	0	7718	-	21.86
(180,1000)	3.46	5.00	ES	139	0	8086	3*	20.04
			CO	139	0	<b>1660</b>	45	<b>18.60</b>
			ES+CO	139	0	1909	63	22.07
			Ninguno	228	72	39663	-	1800.09
5.98	10.00		ES	185	<b>66</b>	64846	3*	1800.09
			CO	227	72	38565	947	1802.75
			ES+CO	188	<b>66</b>	49180	729	1801.84
			Ninguno	388	92	8137	-	1800.12
11.00	20.00		ES	363	92	7503	3*	1800.06
			CO	358	92	7285	222	1800.24
			ES+CO	321	<b>91</b>	<b>7484</b>	169	1801.25
			Ninguno	169	0	1	-	0.14
2.00	2.00		ES	169	0	1	3*	0.12
			ES+CO	169	0	1	3	0.12
			Ninguno	93	0	8729	-	84.48
(180,2000)	3.52	5.00	ES	93	0	<b>7223</b>	3*	<b>77.02</b>
			CO	93	0	8775	131	155.02
			ES+CO	93	0	7970	79	149.50
			Ninguno	100	60	11701	-	1800.08
5.98	10.00		ES	110	64	14431	3*	1802.88
			CO	115	65	14054	319	1802.18
			ES+CO	102	61	13913	560	1802.42
			Ninguno	260	92	5465	-	1800.08
11.15	20.00		ES	215	<b>89</b>	5780	3*	1800.07
			CO	224	90	<b>4440</b>	503	1800.83
			ES+CO	227	90	6145	783	1800.77

## A.4. Formulación $\mathcal{F}_4$ y $\mathcal{F}_5$

Tabla A.4: Resultados computacionales modelo  $\mathcal{F}_4$  y  $\mathcal{F}_5$ .

Instancia (n,m)	e <sub>prom</sub>    e <sub>max</sub>		Modelo	Obj	Gap [%]	Nodos B&C	Tiempo [s]
(45,100)			F4	236	0	1967	1.24
	2.00	2.00	F5	236	0	803	0.26
			F4	116	0	4296	3.97
	3.49	5.00	F5	116	0	3757	12.66
			F4	246	0	5758	10.81
	5.57	10.00	F5	246	0	7065	55.21
(45,200)			F4	112	0	2283	2.06
	2.00	2.00	F5	112	0	495	0.45
			F4	57	0	9417	9.50
	3.52	5.00	F5	57	0	3310	17.54
			F4	71	0	34902	68.84
	5.91	10.00	F5	71	0	16636	330.17
(45,500)			F4	71	0	37883	11.11
	2.00	2.00	F5	71	0	88045	13.90
			F4	37	0	44540	43.36
	3.41	5.00	F5	37	0	8535	107.62
			F4	34	41	183398	1800.04
	5.93	10.00	F5	34	41	4745	1800.03
(90,150)			F4	565	0	5663	4.49
	2.00	2.00	F5	565	0	4116	1.40
			F4	230	0	152040	139.30
	3.37	5.00	F5	230	0	31645	75.60
			F4	-	-	-	1.57
	6.15	10.00	F5	-	-	-	22.66

			F4	-	-	-	1.17
	11.07	20.00	F5	-	-	-	62.66
			F4	252	0	20705	39.75
	2.00	2.00	F5	252	0	3230	2.74
(90,500)			F4	75	0	260906	569.85
	3.51	5.00	F5	75	0	27984	405.30
			F4	173	69	299716	1800.04
	6.25	10.00	F5	152	66	2769	1800.07
			F4	223	86	30868	1800.25
	11.07	20.00	F5	-	-	-	1800.07
			F4	171	0	374170	213.65
	2.00	2.00	F5	171	0	1	0.91
(90,1000)			F4	43	0	51971	96.68
	3.52	5.00	F5	43	0	9996	343.48
			F4	89	73	117267	1800.05
	5.99	10.00	F5	-	-	-	1800.06
			F4	157	92	4856	1800.17
	10.89	20.00	F5	-	-	-	1800.10
			F4	-	-	-	0.00
	2.00	2.00	F5	-	-	-	0.00
(150,300)			F4	312	13	210273	1800.07
	3.38	5.00	F5	298	0	108512	1252.36
			F4	-	-	-	1800.04
	6.34	10.00	F5	-	-	-	1800.02
			F4	-	-	-	25.37
	11.21	20.00	F5	-	-	-	1800.06
			F4	372	2	529350	1800.04
	2.00	2.00	F5	372	0	1270	4.89
(150,800)			F4	144	19	138720	1800.07
	3.52	5.00	F5	143	15	11563	1800.03
			F4	-	-	-	1800.07
	6.06	10.00	F5	218	68	545	1800.06
			F4	-	-	-	1800.07
	10.95	20.00					

			F5	-	-	-	1800.07
			F4	259	2	288499	1800.05
	2.00	2.00	F5	259	0	3240	8.54
(150,1500)			F4	110	19	102626	1800.08
	3.48	5.00	F5	102	10	2823	1800.05
			F4	-	-	-	1800.19
	6.07	10.00	F5	-	-	-	1800.03
			F4	-	-	-	1800.67
	11.01	20.00	F5	-	-	-	1800.07
			F4	1036	4	211397	1800.09
	2.00	2.00	F5	1031	0	648262	394.26
(180,400)			F4	303	13	106145	1800.07
	3.43	5.00	F5	286	1	99988	1800.05
			F4	-	-	-	1800.03
	5.79	10.00	F5	-	-	-	1800.04
			F4	-	-	-	1009.54
	11.30	20.00	F5	-	-	-	1800.04
			F4	464	2	287522	1800.05
	2.00	2.00	F5	464	0	82019	41.71
(180,1000)			F4	180	17	68364	1800.05
	3.43	5.00	F5	177	12	5157	1800.04
			F4	-	-	-	1800.10
	6.03	10.00	F5	-	-	-	1800.07
			F4	-	-	-	1804.65
	10.98	20.00	F5	-	-	-	1800.08
			F4	272	1	289679	1800.04
	2.00	2.00	F5	272	0	2792	8.93
(180,2000)			F4	109	19	102838	1800.09
	3.51	5.00	F5	103	12	559	1800.10
			F4	-	-	-	1800.13
	6.15	10.00	F5	-	-	-	1800.04
			F4	-	-	-	1805.13
	10.84	20.00	F5	-	-	-	1800.07

