DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico I

Investigacion Opeartiva
Primer Cuatrimestre de 2017

Integrante	LU	Correo electrónico
Axel Lew	627/14	axel.lew@hotmail.com
Brian Bohe	706/14	brianbohe@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3359
http://www.fcen.uba.ar

Índice

1. Introducción

En este trabajo práctico se resuelve un conjunto de problemas utilizando conceptos vistos en la materia sobre programación lineal y lineal entera.

En cada problema se plantea un modelo y se implementa una solución para poder responder preguntas de cada enunciado. Se utiliza la biblioteca de *CPLEX* en *C++* para los ejercicios 1 y 23. El ejercicio 6 se resolveremos utilizando *LINDO*, y el resto utilizando *CPLEX Studio*.

Para los problemas 1 y 23 se realiza un análisis sobre las diferentes políticas u opciones de cortes, branching, elección de nodos y preprocesamiento con el objetivo de encontrar para cada problema la combinación que obtenga los tiempos más bajos de ejecución.

Se resuelve sobre el final y de manera adicional el ejercicio 10 de la práctica 8.

2. Ejercicio 12.1

2.1. Descripción del problema

Se tiene un conjunto de $Aceites = AceitesVegetales \cup AceitesNoVegatales$, con $AceitesVegetales$ teniendo dos subtipos de aceites ($\{\text{VEG1}, \text{VEG2}\}$), $AceitesNoVegatales$ teniendo tres subtipos de aceites ($\{\text{OIL1}, \text{OIL2}, \text{OIL3}\}$), y ambos disjuntos. Para cada aceite se conoce:

1. Precio para comprar una tonelada del i-*esimo* aceite en el j-*esimo* mes: $Precio_{ji}$

2. Dureza del j-*esimo* aceite: $Dureza_j$

Donde la dureza se combina linealmente al mezclar aceites.

A su vez se tiene también un conjunto de $Meses$ = {Enero, Febrero, Marzo, Abril, Mayo, Junio} y una cantidad de stock al comenzar Enero del i-*esimo* aceite $Stock_{0i}$ de 500 toneladas. El costo de almacenaje es de 5 libras por tonelada al mes.

Se quiere mezclar estos aceites para generar un producto final que se vende a 150 libras la tonelada. No se pierde aceite al realizar la mezcla y se puede mezclar aceites de distintos tipos.

El problema presenta las restricciones:

R1.1 No se puede almacenar producto final.

R1.2 La dureza del producto final debe estar entre 3 y 6.

R1.3 No es posible almacenar mas de 1000 toneladas de cada aceite.

R1.4 El stock al terminar el mes de Junio tiene que ser igual al inicial para todos los aceites.

R1.5 No es posible utilizar mas de 200 toneladas de aceites del tipo $AceitesVegetales$.

R1.6 No es posible utilizar mas de 250 toneladas de aceites del tipo $AceitesNoVegetales$.

2.2. Asunciones

Observación 2.1. *Para cumplir con la primer restricción, se asume que todo producto final producido es vendido instantáneamente.*

Observación 2.2. *Tiene sentido asumir que aunque el precio sea por tonelada, comprar* $t \in \mathbb{R}_{\geq 0}$ *toneladas del* i*-esimo aceite en el* j*-esimo mes tiene un costo de* $t \times Precio_{ji}$.

Observación 2.3. *Se puede utilizar cantidades no enteras de toneladas para producir el producto final.*

2.3. Variables del Modelo

Se definen las variables:

1. $Stock_{ji}$: Cantidad en toneladas del i-*esimo* aceite almacenado en el j-*esimo* mes. Notar que utilizamos el subíndice 0 para el stock inicial, por lo que $j \in [1, 6]$.
2. $Compra_{ji}$: Cantidad en toneladas del i-*esimo* aceite comprado en el j-*esimo* mes.
3. Uso_{ji}: Cantidad en toneladas del i-*esimo* aceite usado en el j-*esimo* mes para producir el producto final.

2.4. Restricciones del Modelo

$(\forall i \in Aceites)(\forall j \in Meses)$

$$\sum_{i' \in AceitesVegetales} Uso_{ji'} \leq 200 \tag{1a}$$

$$\sum_{i' \in AceitesNoVegetales} Uso_{ji'} \leq 250 \tag{1b}$$

Para cumplir las restricciones **R1.4** y **R1.5** sobre la cantidad máxima posible de producción

$$\sum_{i' \in Aceites} Uso_{ji'} \times Dureza_{i'} - \sum_{i' \in Aceites} Uso_{ji'} \times 3 \geq 0 \tag{2a}$$

$$\sum_{i' \in Aceites} Uso_{ji'} \times 6 - \sum_{i' \in Aceites} Uso_{ji'} \times Dureza_{i'} \geq 0 \tag{2b}$$

Para cumplir las restricción **R1.2** la combinación lineal de lo utilizado debe estar contenida en el rango [3,6]:

$$(0 < j < 6)\, Stock_{ji} \leq 1000 \tag{3a}$$

$$Stock_{0i} - Stock_{6i} = 0 \tag{3b}$$

$$Stock_{0i} = 500 \tag{3c}$$

Para cumplir las restricciones de fin y cota de stock **R1.3** y **R1.4**.

Solo queda modelar el cambio de Stock durante los meses, lo cual puede hacerse de la siguiente manera:

$$(j < 6)\, Stock_{j+1i} - Stock_{ji} - Compra_{ji} + Uso_{ji} = 0 \tag{4}$$

Que al ser todas variables positivas limitan en conjunto el uso del mes. No podrá $Uso_{ji} > Stock_{ij}$ a menos que $Compra_{ji} > 0$. Tampoco podrá $Stock_{j+1i}$ crecer tanto como quiera porque la ecuación se iguala a 0, el stock inicial esta limitado, y la cantidad comprada impacta negativamente en los beneficios finales, que es lo que se pide maximizar en este problema.

Y por último resta decir que todas las variables son reales mayoreso iguales a 0:

$$Compra_{ji}, Uso_{ji}, Stock_{ji} \in \mathbb{R}_{\geq 0} \tag{5a}$$

Meses	VEG 1	VEG 2	OIL 1	OIL 2	OIL 3
Enero	22,222	177,77	0	250	0
Febrero	0	200	0	250	0
Marzo	159.259	40.74	0	250	0
Abril	159.259	40.74	0	250	0
Mayo	159.259	40.74	0	250	0
Junio	159.259	40.74	0	250	0

Tabla 1: Cantidad en toneladas a vender de cada aceite en cada mes.

Meses	VEG 1	VEG 2	OIL 1	OIL 2	OIL 3
Enero	0	0	0	0	0
Febrero	0	0	0	750	0
Marzo	0	0	0	0	0
Abril	0	0	0	0	0
Mayo	0	0	0	0	0
Junio	659.259	540.74	0	750	0

Tabla 2: Cantidad en toneladas a comprar de cada aceite en cada mes.

2.5. Función Objetivo

Se quiere maximizar el beneficio obtenido al vender, pagando los costos de compra y almacenaje de aceite necesarios:

Maximizar

$$\sum_{i \in Aceites} \sum_{j \in Meses} Uso_{ji} \times 150 - Compra_{ji} \times Precio_{ji} - 5 \times Stock_{ji} \tag{6}$$

2.6. Solución del Problema

El problema se implementó con *CPLEX Studio* y en *C++* utilizando la librería. El obtenido al valor de la función objetivo es $\$107842, 593$.

Las políticas de compra y venta que la empresa debería tomar para maximizar el beneficio al final de los 6 meses están reflejas respectivamente en las tablas 2 y 1

3. Ejercicio 12.2

3.1. Descripción del problema

Se Extiende el problema 2.1 con las siguientes restricciones:

R2.1 Si el *i-esimo* aceite es utilizado en el *j-esimo* mes, $Uso_{ji} \geq 20$ toneladas.

R2.2 Si algún aceite en *AceiteVegetal* es utilizado, entonces el tercer subtipo de aceite de *AceiteNoVegetal* (OIL 3) es también utilizado.

R2.3 No pueden utilizarse mas de 3 aceites distintos en un mes para producción.

3.2. Asunciones

No hay nuevas Asunciones

3.3. Variables del Modelo

A las variables definidas en 2.3 se añaden:

1. Se_Usa_{ji}: $\begin{cases} 1 & \text{si se utiliza el } i\text{-esimo aceite en el } j\text{-esima mes} \\ \\ 0 & \text{si no} \end{cases}$

Para controlar la cota mínima de uso por aceite en cada mes.

3.4. Restricciones del Modelo

$(\forall i \in Aceites)(\forall j \in Meses)$
Se mantienen todas las restricciones de 2.4 y se agregan nuevas.

$$\sum_{i' \in Aceites} Se\_Usa_{ji'} \leq 3 \tag{7}$$

Para restringir la cantidad de aceites usados en un mismo mes (restricción **R2.3**).

$$Uso_{ji} \geq 20 \times Se\_Usa_{ji} \tag{8}$$

Para cumplir la restricción **R2.1**.

$$Uso_{ji} \leq \mathtt{M} \times Se\_Usa_{ji} \tag{9}$$

Para regular que la cantidad de toneladas usadas de un aceite sea igual a 0 cuando *Se_Usa* vale 0. Se puede tomar cualquier $\mathtt{M} \geq 250$ dado que ese numero es mayor o igual al máximo posible a utilizar en un mes de un aceite.

$$Se\_Usa_{j\,AceitesVegetales_1} \leq Se\_Usa_{j\,AceitesNoVegetales_3} \tag{10a}$$
$$Se\_Usa_{j\,AceitesVegetales_2} \leq Se\_Usa_{j\,AceitesNoVegetales_3} \tag{10b}$$

Para cumplir las restricción **R2.2**.

Meses	VEG 1	VEG 2	OIL 1	OIL 2	OIL 3
Enero	85,185	114,81	0	0	250
Febrero	0	200	0	230	20
Marzo	85,185	114,81	0	0	250
Abril	155	0	0	230	20
Mayo	155	0	0	230	20
Junio	0	200	0	230	20

Tabla 3: Cantidad en toneladas a usar de cada aceite en cada mes.

Meses	VEG 1	VEG 2	OIL 1	OIL 2	OIL 3
Enero	0	0	0	0	0
Febrero	0	0	0	190	0
Marzo	0	0	0	0	40
Abril	0	0	0	0	0
Mayo	0	0	0	0	540
Junio	480.37	629.63	0	730	0

Tabla 4: Cantidad en toneladas a comprar de cada aceite en cada mes.

3.5. Función Objetivo

La misma definida en al sección 2.6 del problema anterior.

3.6. Solución del problema

Este problema se implementó con la herramienta *CPLEX Studio* y se obtuvo un valor de la función objetivo de $\$100278,704$.

Las políticas de compra y venta que la empresa debería tomar para maximizar el beneficio al final de los 6 meses están reflejas respectivamente en las tablas 4 y 3

4. Ejercicio 12.6

4.1. Descripción del problema

4.2. Asunciones

Observación 4.1. *Se puede utilizar fracciones de barriles, es decir, en nuestro modelo las variables serán continuas*

4.3. Variables del Modelo

Se definen las variables:

1. $\mathit{CrudoDestilado}_i$: Cantidad de barriles utilizados de crudo tipo i.
2. *Otras variables*: El resto de las variables tienen nombre descriptivo, y corresponden a la cantidad de barriles utilizados o generados por tipo y uso. Por ejemplo, $LightNaphthaForBlendingPremium$ es la nafta liviana utilizada para el blending de petróleo premium.

4.4. Restricciones del Modelo

No se usa mas de 30.000 barriles del crudo tipo 2.

$$\mathit{CrudoDestilado2} \leq 30000 \tag{11}$$

No se destilan más de 45.000 barriles de crudo.

$$\mathit{CrudoDestilado1} + \mathit{CrudoDestilado2} \leq 45000 \tag{12}$$

Se reforman hasta 10.000 barriles de nafta.

$$\mathit{LightNaphthaReformed} + \mathit{MediumNaphthaReformed} + \mathit{HeavyNaphthaReformed} \leq 10000 \tag{13}$$

Se realiza cracking de hasta 8.000 barriles de aceite.

$$\mathit{LightOilCracked} + \mathit{HeavyOilCracked} \leq 8000 \tag{14}$$

La producción de aceite lubricante esta entre 500 y 1.000 barriles.

$$\mathit{LubeOil} \leq 1000 \tag{15a}$$
$$\mathit{LubeOil} \geq 500 \tag{15b}$$

La producción de petroleo premium tiene que ser al menos 40 % de la produccion de petroleo regular.

$$0{,}4 * \mathit{RegPetrol} - \mathit{PremPetrol} \leq 0 \tag{16}$$

Resultado de la destilación de crudo.

$$LightNaphtha - 0{,}1 * CrudoDestilado1 - 0{,}15 * CrudoDestilado2 = 0 \tag{17a}$$
$$MediumNaphtha - 0{,}2 * CrudoDestilado1 - 0{,}25 * CrudoDestilado2 = 0 \tag{17b}$$
$$HeavyNaphtha - 0{,}2 * CrudoDestilado1 - 0{,}18 * CrudoDestilado2 = 0 \tag{17c}$$
$$LightOil - 0{,}12 * CrudoDestilado1 - 0{,}08 * CrudoDestilado2 = 0 \tag{17d}$$
$$HeavyOil - 0{,}2 * CrudoDestilado1 - 0{,}19 * CrudoDestilado2 = 0 \tag{17e}$$
$$Residuum - 0{,}13 * CrudoDestilado1 - 0{,}12 * CrudoDestilado2 = 0 \tag{17f}$$

Usos de la nafta.

$$LightNaphthaReformed + LightNaphthaForBlending - LightNaphtha \leq 0 \tag{18a}$$
$$MediumNaphthaReformed + MediumNaphthaForBlending - MediumNaphtha \leq 0 \tag{18b}$$
$$HeavyNaphthaReformed + HeavyNaphthaForBlending - HeavyNaphtha \leq 0 \tag{18c}$$
$$\tag{18d}$$

Uso del residuo.

$$2 * LubeOil + ResiduumJetFuel + ResiduumFuelOil - Residuum \leq 0 \tag{19}$$

Creación de la gasolina reformada.

$$\begin{split}0{,}6 * LightNaphthaReformed + 0{,}52 * MediumNaphthaReformed + \\ 0{,}45 * HeavyNaphthaReformed - ReformedGasoline = 0\end{split} \tag{20}$$

Uso del aceite pesado.

$$\begin{split}HeavyOilJetFuel + HeavyOilFuelOil + HeavyOilCrackedGasoline + \\ HeavyOilCrackedOil - HeavyOil \leq 0\end{split} \tag{21}$$

Uso del aceite liviano

$$\begin{split}LightOilJetFuel + LightOilFuelOil + LightOilCrackedGasoline + \\ LightOilCrackedOil - LightOil \leq 0\end{split} \tag{22}$$

Creación del cracked oil.

$$0{,}68 * LightOilCrackedOil + 0{,}75 * HeavyOilCrackedOil - CrackedOil = 0 \tag{23}$$

Creación del cracked gasoline.

$$0{,}28 * LightOilCrackedGasoline + 0{,}2 * HeavyOilCrackedGasoline - CrackedGasoline = 0 \tag{24}$$

Uso del cracked oil.

$$CrackedOilFuelOil + CrackedOilJetFuel - CrackedOil = 0 \tag{25}$$

Creación del jet fuel.

$$\begin{aligned} LightOilJetFuel + HeavyOilJetFuel + CrackedOilJetFuel + \\ ResiduumJetFuel - JetFuel = 0 \end{aligned} \tag{26}$$

Composición del jet fuel.

$$\begin{aligned} LightOilJetFuel + 0{,}6 * HeavyOilJetFuel + 1{,}5 * CrackedOilJetFuel + \\ 0{,}05 * ResiduumJetFuel - JetFuel \leq 0 \end{aligned} \tag{27}$$

Uso de la nafta liviana para blending.

$$\begin{aligned} LightNaphthaForBlendingRegular + LightNaphthaForBlendingPremium - \\ LightNaphthaForBlending = 0 \end{aligned} \tag{28}$$

Uso de la nafta mediana para blending.

$$\begin{aligned} MediumNaphthaForBlendingRegular + MediumNaphthaForBlendingPremium - \\ MediumNaphthaForBlending = 0 \end{aligned} \tag{29}$$

Uso de la nafta pesada para blending.

$$\begin{aligned} HeavyNaphthaForBlendingRegular + HeavyNaphthaForBlendingPremium - \\ HeavyNaphthaForBlending = 0 \end{aligned} \tag{30}$$

Uso de la gasolina reformada.

$$ReformedGasolineRegular + ReformedGasolinePremium - ReformedGasoline = 0 \tag{31}$$

Uso de la cracked gasoline.

$$CrackedGasoline - CrackedGasolinePremium - CrackedGasolineRegular = 0 \tag{32}$$

Creación del petróleo regular.

$$\begin{aligned} LightNaphthaForBlendingRegular + MediumNaphthaForBlendingRegular + \\ HeavyNaphthaForBlendingRegular + ReformedGasolineRegular + \\ CrackedGasolineRegular - RegPetrol = 0 \end{aligned} \tag{33}$$

Creación del petróleo premium.

$$\begin{aligned} LightNaphthaForBlendingPremium + MediumNaphthaForBlendingPremium + \\ HeavyNaphthaForBlendingPremium + ReformedGasolinePremium + \\ CrackedGasolinePremium - PremPetrol = 0 \end{aligned} \tag{34}$$

$$\begin{aligned} 90 * LightNaphthaForBlendingRegular + 80 * MediumNaphthaForBlendingRegular + \\ 70 * HeavyNaphthaForBlendingRegular + 115 * ReformedGasolineRegular + \\ 105 * CrackedGasolineRegular - 84 * RegPetrol \geq 0 \end{aligned} \tag{35}$$

$$90 * LightNaphthaForBlendingPremium + 80 * MediumNaphthaForBlendingPremium + 70 * HeavyNaphthaForBlendingPremium + 115 * ReformedGasolinePremium + 105 * CrackedGasolinePremium - 94 * PremPetrol \geq 0 \tag{36}$$

$$1/10 * LightOilFuelOil - FuelOil \geq 0 \tag{37a}$$

$$1/4 * CrackedOilFuelOil - FuelOil \geq 0 \tag{37b}$$

$$1/3 * HeavyOilFuelOil - FuelOil \geq 0 \tag{37c}$$

$$ResiduumFuelOil - FuelOil \geq 0 \tag{37d}$$

4.5. Función Objetivo

Se quiere maximizar el beneficio de la venta de los distintos tipos de productos.
Maximizar

$$700 \times PremPetrol + 600 \times RegPetrol + 400 \times JetFuel + 350 \times FuelOil + 150 \times LubeOil \tag{38}$$

4.6. Solución del problema

La solución al problema, es decir, la ganancia producida, es 20.425.870.
Las producciones son:

1. $PremPetrol$: 6.077,668
2. $RegPetrol$: 15.194,17
3. $JetFuel$: 17.450
4. $FuelOil$: 0
5. $LubeOil$: 500

5. Ejercicio 12.13

5.1. Descripción del problema

Se tiene un conjunto de $Minoritas = \{M_1, \ldots, M_{23}\}$ y de cada uno se conoce:

1. La cantidad de combustible en galones que vende: $Venta_i$
2. La cantidad de puntos de delivery: $Delivery_i$
3. Categoría de crecimiento. $EsCategoriaA_i \begin{cases} 1 & \text{si pertenece a la categoría A} \\ 0 & \text{si pertenece a la categoría B} \end{cases}$
4. La cantidad de licor en galones que vende: $Licores_i$
5. La región a la que pertenece. $Minoritas = R_1 \cup^D R_2 \cup^D R_3$

Que se generalizarán como un conjunto de *Propiedades* que contiene cada minorista.

Se quiere repartir *Minoristas* en dos grupos, de forma que uno controle el %40 de cada uno de los puntos:

1. Cantidad de combustible que vende en la región 1.
2. Cantidad de combustible que vende en la región 2.
3. Cantidad de combustible que vende en la región 3.
4. Número total de puntos de delivery.
5. Control de mercado de licores.
6. Número de minoristas de categoría A.
7. Número de minoristas de categoría B.

5.2. Asunciones

Observación 5.1. *Si sobre alguna propiedad se distribuye entre el sector 1 y 2 un 43 % y 57 % respectivamente, se dice que el desvío es de 3 % y no 6 %.*

Observación 5.2. *Se establece que puede haber un error de +/-5 % sobre el porcentaje que cubre cada sector en cada uno de los puntos anteriormente descritos.*

5.3. Variables del Modelo

En un comienzo sólo interesa saber que minorista pertenecerá a cada sector por lo que se puede simplemente declarar las variables:

1. D_i: $\begin{cases} 1 & \text{si el } \textit{i-esimo} \text{ minorista esta en el sector 1} \\ \\ 0 & \text{si está en el sector 2} \end{cases}$

2. *Delta'*$_j$ y *Delta*$_j$: desvíos de la j*-esimo* propiedad sobre el porcentaje final de minoristas en el sector 1.

El objetivo de dos Deltas es modelar el posible desvío en cada propiedad del %40 en ambas direcciones.

5.4. Restricciones del Modelo

Para satisfacer la condición de cobertura del mercado de combustibles en R_1, se establece:

$$\begin{aligned}\sum_{i \in R_1} Combustible_i \times (0{,}4 + Delta_{Combustible_1} - Delta'_{Combustible_1}) \\ = \sum_{i \in R_1} Combustible_i \times D_i\end{aligned} \tag{39}$$

Esta restricción se repite por para cada una de las regiones R_2 y R_3 con $Delta_{Combustibles_2}$, $Delta'_{Combustibles_2}$, $Delta_{Combustibles_3}$ y $Delta'_{Combustibles_3}$ respectivamente.

Las demás propiedades no requieren ser discriminadas por región y pueden definirse genéricamente como:

$$\sum_{i \in Minoristas} Prop_i \times (0{,}4 + Delta_{Prop} - Delta'_{Prop}) = \sum_{i \in Minoristas} Prop_i \times D_i \tag{40}$$

Donde *Prop* puede ser cualquiera de las *Propiedades* menos *combustibles.*

Para asegurar que no se exceda el +/-5 % de desvío se agrega sobre los deltas:
$(\forall j \in Propiedades)$

$$Delta_j \leq 0{,}05 \tag{41a}$$
$$Delta'_j \leq 0{,}05 \tag{41b}$$

Resta especificar el rango de valores de las variables:
$(\forall i \in Minoristas)$ $(\forall j \in Propiedades)$

$$Delta'_j, Delta_j \in \mathbf{R}^+ \tag{42a}$$
$$D_i \in \{0, 1\} \tag{42b}$$

5.5. Función Objetivo

En un comienzo interesa saber si es posible hacer dicha partición sin necesidad de minimizar o maximizar alguna función. Se pueden utilizar varias funciones objetivo, arbitrariamente se opta por:

Maximizar

$$\sum_{i \in Minoristas} D_i \tag{43}$$

5.6. Solución del problema

Sector 1 = $\{M_4, M_6, M_8, M_{10}, M_{14}, M_{16}, M_{18}, M_{20}, M_{22}\}$
Sector 2 = *Minoristas* \ Sector 1

5.7. Extensión I

Se parte del problema descrito y se quiere minimizar el error +/-5 % o desvío del porcentual 40/60 respectivamente. Para conseguir esto se puede sólo cambiar la función objetivo.

La nueva función objetivo será: *Minimizar*

$$\sum_{j \in Propiedades} Delta_j + Delta'_j \tag{44}$$

La mínima suma de desvíos obtenida es de 5,3 % y definiendo un Sector 1 = $\{M_1, M_2, M_3, M_4, M_{11}, M_{16}, M_{18}, M_{21}$ y $M_{22}\}$.

5.8. Extensión II

Se parte del problema descrito y se quiere minimizar el máximo error o desvío del porcentual 40/60 respectivamente.

Para conseguir esto se puede agregar una nueva variable *MaximoDelta* que acote superiormente a todos los deltas y modificar nuevamente la función objetivo.

Las restricciones necesarias para que esto suceda son: ($\forall j \in Propiedades$)

$$Delta'_j \leq MaximoDelta \tag{45a}$$

$$Delta_j \leq MaximoDelta \tag{45b}$$

$$MaximoDelta \in \mathbf{R}^+ \tag{45c}$$

Que se pide sea lo menor posible por lo que La nueva función objetivo será: *Minimizar*

$$MaximoDelta \tag{46}$$

El mínimo valor para *MaximoDelta* obtenido es de 2,5 % y definiendo un Sector 1 = $\{M_3, M_6, M_7, M_{11}, M_{12}, M_{13}, M_{17}, M_{20}, M_{22}\}$. Se espera que la suma de los desvíos sea mayor o igual a la obtenida en 5.7, y es así ya que en este caso es de 8,11 %.

Ambos resultados nos permiten concluir que no existe una forma de partir el conjunto de *Minoristas* en dos conjuntos de forma tal que no se acepte ningún error o desvío del 40 % para ninguna de las *Propiedades*. De existir, el mínimo *MaximoDelta* y la suma serían 0.

6. Ejercicio 12.15

6.1. Descripción del problema

Se tiene tres tipos de *Generadores* = { GT_1, GT_2, GT_3} y de cada uno se conoce:

1. Cantidad de máquina disponibles: $\mathit{Disponibles}_i$
2. Minimo de MW que genera una máquina: $MinimoMW_i$
3. Maximo de MW que genera una máquina: $MaximoMW_i$
4. El costo por hora por máquina produciendo el mínimo de MW: $\mathit{CostoMinimo}_i$
5. El costo por hora por máquina por MW extra producido: $\mathit{CostoExtra}_i$
6. El costo de encender una máquina: $\mathit{CostoEncendido}_i$

A su vez se tiene un conjunto $\mathit{Horas} = \{1, \ldots, 24\}$ del que se sabe la cantidad de MW demandados en cada una:

1. Cantidad de MW demandada en la j*-esima* hora: $DemandaMW_j$

Se pide minimizar el costo de producción sujeto a las restricciones:

R6.1 Se debe satisfacer la demanda de MW de cada hora.

R6.2 Cada generador, de estar encendido, deberá producir una cantidad de MW entre su mínimo y máximo.

R6.3 En todo momento debe poderse satisfacer con los *Generadores* encendidos hasta un 15 % de demanda del establecido.

6.2. Asunciones

Observación 6.1. *No se puede conservar la energía.*

Observación 6.2. *Se puede generar una cantidad de MW no entera y el cálculo del costo es lineal, es decir, generar 14.5 MW tiene un costo igual a la suma de los costos de 14 MW y 15 MW divido 2.*

Observación 6.3. *No se establece ningún tiempo de encendido de una máquina, por lo que se considera instantáneo.*

Observación 6.4. *No se establece el estado de las máquinas al inicio del día, se asume que todas las máquinas comienzan apagadas.*

6.3. Variables del Modelo

Se definen las variables:

- $ExtraMW_{ij}$: Cantidad de MW extra producidos por los generadores del *i-esimo* tipo en la *j-esima* hora.
- $Usados_{ij}$: Cantidad de generadores del *i-esimo* tipo usados en la *j-esima* hora.
- $Prendidos_{ij}$: Cantidad de generadores del *i-esimo* tipo prendidos en la *j-esima* hora.
- $Apagados_{ij}$: Cantidad de generadores del *i-esimo* tipo apagados en la *j-esima* hora.

Es necesario notar que como todos los generadores prendidos producen al menos el mínimo de MW que su tipo puede producir, sólo es necesario modelar el extra producido por el mismo. De estar manera no es necesario crear restricciones para satisfacer **R6.2**.

6.4. Restricciones del Modelo

$(\forall i \in Generadores)$ $(\forall j \in Horas)$

Como todas las máquinas comienzan apagadas, en la primer hora la cantidad de generadores usados es igual a la cantidad prendida, y estas no pueden superar la cantidad de máquinas disponibles de cada tipo:

$$Usados_{i1} - Prendidos_{i1} = 0 \tag{47a}$$

$$Usados_{i1} <= Disponibles_i \tag{47b}$$

Para las horas posteriores el cálculo de cantidad de máquinas usadas dependerá de cuantas se apagaron o prendieron en la nueva hora. La cantidad de máquinas usadas no puede superar la disponibilidad, y tampoco tiene sentido que las usadas en una hora sumadas a las prendidas en la siguiente supere esta cantidad:

$$(j < 24)\ Usados_{ij+1} - Usados_{ij} - Prendidos_{ij+1} + Apagados_{ij+1} = 0 \tag{48a}$$

$$(j < 24)\ Usados_{ij} + Prendidos_{ij+1} \leq Disponibles_i \tag{48b}$$

Para cubrir la demanda (**R6.1**) de cada hora es necesario pedir que la suma de todas las máquinas prendidas con su producción al mínimo más la producción extra supere la demanda. En el caso de los picos de demanda donde debe ser atendida con los generadores prendidos (**R6.3**), sólo es necesario superar este valor sumando las maquinas prendidas asumiendo que producen el máximo de su capacidad:

$$\sum_{i' \in Generadores} Usados_{i'j} \times MinimoMW_{i'} + ExtraMW_{i'j} \geq DemandaMW_j \tag{49a}$$

$$\sum_{i' \in Generadores} Usados_{i'j} \times MaximoMW_{i'} \geq DemandaMW_j \times 1{,}15 \tag{49b}$$

La cantidad extra de MW producidos de cada por cada tipo de generadores está acotado por la cantidad de máquinas utilizadas:

$$ExtraMW_{ij} \leq Usados_{ij} \times (MaximoMW_i - MinimoMW_i) \tag{50}$$

Y por último resta pedir que las variables sean enteras o reales pero positivas:

$$Usados_{ij}, Prendidos_{ij}, Apagados_{ij} \in \mathbb{N}_0 \tag{51a}$$

$$ExtraMW_{ij} \in \mathbb{R}_{\geq 0} \tag{51b}$$

6.5. función Objetivo

El costo total incurrido es la suma de los costo por mantener máquinas al mínimo de su capacidad, el costo extra de cada máquina utilizada si se utiliza y los costos por encender una máquina. *Minimizar*

$$\sum_{i \in Generadores} \sum_{j \in Horas} Usados_{ij} \times CostoMinimo_i + ExtraMW_{ij} \times CostoExtra_i + Prendidos_{ij} \times CostoEncendido_i \tag{52}$$

6.6. Solución del problema

El mínimo costo incurrido es de $1015150. La cantidad de máquinas usadas por hora de cada tipo es la siguiente:

Horas	Tipo 1	Tipo 2	Tipo 3
1	12	2	0
2	12	2	0
3	12	2	0
4	12	2	0
5	12	2	0
6	12	2	0
7	12	8	0
8	12	8	0
9	12	8	0
10	12	8	0
11	12	8	0
12	12	8	0
13	12	8	0
14	12	8	0
15	12	8	0
16	12	9	2
17	12	9	2
18	12	9	2
19	12	9	0
20	12	9	0
21	12	9	0
22	12	9	0
23	12	9	0
24	12	9	0

7. Ejercicio 12.16

7.1. Descripción del problema

Se extiende el problema de la sección 6.1 con un nuevo tipo de generadores llamado GTW que utilizan un reservorio de agua para funcionar y de los cuales se conoce:

1. Cantidad de máquinas disponibles: $Disponibles_i$
2. Cantidad constante de MW que genera: $ConstanteMW_i$
3. El costo por hora por producir energía: $CostoConstante_i$
4. La cantidad en metros que se reduce el reservorio por hora de trabajo: $CostoReservorio_i$
5. El costo por encender el *i-esimo* generador: $Encendido_i$

El reservorio puede llenarse con 1 metro de agua en una hora aplicando una energía de 3000 MW con los otros generadores.

Se añaden nuevas restricciones relacionadas al reservorio:

R7.1 El reservorio debe tener una profundidad de 16 metros todos los días al final del día.

R7.2 El reservorio debe tener una profundidad contenida en el rango [15,20] durante todo el día.

Se modifica la restricción **R6.3** de manera que no se pueden considerar para esto la energía proveniente de generadores que se estén prendiendo en ese momento. Si se puede utilizar una combinación de las siguientes:

1. Generadores GTW aunque el nivel de profundidad no este en los rangos [15,20]. Su tiempo de encendido si se lo considera instantáneo.
2. Generadores que estaban prendidos en algún momento anterior. Consideraremos para esto aquellos prendidos al menos la hora anterior.
3. Energía dedicada a llenar el reservorio.

7.2. Asunciones

Observación 7.1. *Al iniciar el problema la profundidad del reservorio es de 16 metros.*

Observación 7.2. *Se puede aplicar una cantidad de energía E no múltiplo de 3000 MW en una hora para llenar el reservorio pudiendo así llenar* $\frac{E}{3000}$ *metros.*

7.3. Variables del Modelo

A las variables previamente definidas en la sección 6.3 se agregan:

1. $MinimoR_{ij}$: Cantidad de MW tomados de la mínima producción de generadores del *i-esimo* tipo en la *j-esima* hora dedicados a llenar el reservorio.
2. $ExtraR_{ij}$: Cantidad de MW tomados de la producción extra de generadores del *i-esimo* tipo en la *j-esima* hora dedicados a llenar el reservorio.

3. P_j: Profundidad en metros del reservorio al terminar la j-*esima* hora.

No será necesario definir la variable $ExtraMW_{ij}$ para GTW dado que siempre producen una cantidad constante de energía.

7.4. Restricciones del Modelo

Se descartan todas las restricciones de la sección 6.4 y se definen nuevas.
($\forall i' \in Generadores \setminus \mathrm{GTW}$) ($\forall i \in Generadores$) ($\forall j \in Horas$)
Para satisfacer la relación entre $Usados_{i'j}$, $MinimoR_{i'j}$, $ExtraR_{i'j}$ y $ExtraMW_{i'j}$, se añaden:

$$ExtraMW_{i'j} + ExtraR_{i'j} \leq Usados_{i'j} \times (MaximoMW_{i'} - MinimoMW'_i) \tag{53a}$$
$$MinimoR_{i'j} \leq Usados_{i'j} \times MinimoMW'_i \tag{53b}$$

Para cubrir la demanda por hora se tiene en cuenta el restar la energía dedicada a llenar el reservorio:

$$\begin{aligned} \sum_{g \in Generadores \setminus \mathrm{GTW}} & Usados_{gj} \times MinimoMW_g - MinimoR_{gj} + ExtraMW_{gj} \\ & + \sum_{w \in GTW} Usados_{wj} \times ConstanteMW_w \geq DemandaMW_j \end{aligned} \tag{54}$$

Para cubrir los posibles picos de demanda con los cambios en la restriccion **R6.3**, se agrega:

$$\begin{aligned} \sum_{g \in Generadores \setminus \mathrm{GTW}} & (Usados_{gj} - Prendidos_{gj}) \times MaximoMW_g \\ & + \sum_{w \in GTW} ConstanteMW_w \geq DemandaMW_j \times 1{,}15 \end{aligned} \tag{55}$$

Se sabe que el reservorio empieza y termina el día con 16 metros de profundidad, y que no puede en ningún momento del día tener una profundidad menor a 15 o mayor a 20 metros por las restricciones **R7.1** y **R7.2**. Esto se puede hacer de la siguiente manera:

$$P_0 = 16 \tag{56a}$$
$$P_{24} - P_0 = 0 \tag{56b}$$
$$P_j \geq 15 \tag{56c}$$
$$P_j \leq 20 \tag{56d}$$

Durante el día la profundidad dependerá de la energía aplicada a llenarlo y el uso de las máquinas que necesitan del mismo:

$$\begin{aligned} P_j - P_{j-1} - & \sum_{g \in Generadores \setminus GTW} \frac{MinimoR_{g\,j} + ExtraR_{g\,j}}{3000} \\ & + \sum_{w \in GTW} Usados_{w\,j} \times CostoReservorio_w = 0 \end{aligned} \tag{57}$$

Para poder modelar el costo de prender un generador que no utiliza el reservorio:

$$Usados_{i1} - Prendidos_{i1} = 0 \tag{58a}$$
$$Usados_{i1} \leq Disponibles_i \tag{58b}$$
$$(j < 24)\, Usados_{ij+1} - Prendido_{ij+1} + Apagado_{ij+1} - Uso_{ih} = 0 \tag{58c}$$
$$Usados_{ij} + Prendido_{ij+1} \leq Disponibles_i \tag{58d}$$

Y por ultimo queda indicar los rangos de las variables:

$$Usados_{ij}, Prendidos_{ij}, Apagados_{ij} \in \mathbb{N}_0 \tag{59a}$$
$$MinimoR_{ij}, ExtraMW_{ij}, ExtraR_{ij}, MWR_{ij}, P_j \in \mathbb{R} \geq 0 \tag{59b}$$

7.5. Función Objetivo

Al costo descrito en la sección 6.5 es ahora necesario agregarle los costos incurridos por el uso y prendido de los nuevos generadores GTW. Es importante notar que el extra de MW producidos por cada tipo de generador es ahora en realidad la suma de $ExtraMW_{i'j}$ y $ExtraR_{i'j}$.

Minimizar

$$\begin{aligned}\sum_{j \in Horas} & \sum_{g \in Generadores \setminus \mathrm{GTW}} Usados_{gj} \times CostoMinimo_g + Prendidos_{gj} \times CostoPrendido_g \\ & + (ExtraMW_{gj} + ExtraR_{ij}) \times CostoExtra_i \\ & + \sum_{w \in GTW} Usados_{wj} \times ConstanteMW_w + Prendidos_{wj} \times CostoPrendido_w\end{aligned} \tag{60}$$

7.6. Solución del problema

Observación 7.3. *En la primera hora, pueden usarse las máquinas que recién se prenden para soportar picos de demanda. De lo contrario, para algunos inputs como el que se especifica en el enunciado, el problema no tiene una solución factible.*

El valor obtenido es de \$1014490. La cantidad de máquinas usadas por hora de cada tipo es la siguiente:

Horas	Tipo 1	Tipo 2	Tipo 3	GTW 1	GTW 2
1	12	3	0	0	0
2	12	3	0	0	0
3	12	3	0	0	0
4	12	3	0	0	0
5	12	3	0	0	0
6	12	5	0	0	0
7	12	9	0	0	1
8	12	9	0	0	1
9	12	9	0	0	1
10	12	9	0	0	1
11	12	9	0	0	1
12	12	9	0	0	1
13	12	9	0	0	1
14	12	9	0	0	1
15	12	9	1	0	1
16	12	9	1	0	1
17	12	9	1	0	1
18	12	9	1	0	1
19	12	9	0	0	1
20	12	9	0	0	1
21	12	9	0	0	1
22	12	9	0	0	1
23	12	9	0	0	1
24	12	9	0	0	0

8. Ejercicio 12.23

8.1. Descripción del problema

Se desea recolectar leche de 20 granjas diferentes, partiendo de un deposito y volviendo al mismo.

Algunas granjas son pequeños, por solo requieren ser visitadas cada dos días. Las otras requieren ser visitadas todos los días. Se definen dos conjuntos, $NoDiarias$ y $Diarias$ que contienen las granjas según su frecuencia de visita.

Cada granja produce una cantidad diferente de leche, y el camión que la recolecta tiene capacidad de hasta 80.000 litros. Se busca minimizar la distancia recorrida por el camión para la recolección de la leche.

Son dato del problema:

- $Distancia_{ij}$: Distancia entre la *i-ésima* granja y la *j-ésima* granja.
- $Recolectar_i$: Cuantos litros de leche a recolectar de la *i-ésima* granja.

8.2. Asunciones

Observación 8.1. *El camión de recolección sale sólo una vez al día.*

Observación 8.2. *Se define M como un número suficientemente grande.*

Observación 8.3. *Se simplifican las restricciones de subtour las cuales garantizan que el resultado es un ciclo conexo. Para esto se agregan dos restricciones que (viendo luego la solución) aseguran el cumplimiento del mismo.*

8.3. Variables del Modelo

Se definen las variables:

1. E_{dij}: Variable booleana que indica si la ruta elegida para el día d tiene el tramo que va de la granja i a la granja j.
2. Y_i: Variable booleana que indica si no se visita la granja i en el día 0.

8.4. Restricciones del Modelo

No se recolectan más de 80.000 litros de leche por día.
$(\forall d \in \textit{Días})$

$$\sum_{i \neq j \in Granjas} Recolectar_i \times E_{dij} \leq 80000 \tag{61}$$

Todos los días, el camión entra y sale una sola vez de las granjas diarias.
$(\forall d \in \textit{Dias})(\forall i \in \textit{Diarias})$

$$\sum_{j \in Granjas} E_{dij} = 1 \tag{62a}$$

$$\sum_{j \in Granjas} E_{dji} = 1 \tag{62b}$$

Un solo día, el camión sale a las granjas no diarias. Un solo día, el camión entra a las granjas no diarias.
$(\forall i \in \mathit{NoDiarias})$

$$\sum_{d \in \mathit{Días}} \sum_{j \in \mathit{Granjas}} E_{dij} = 1 \tag{63a}$$

$$\sum_{d \in \mathit{Días}} \sum_{j \in \mathit{Granjas}} E_{dji} = 1 \tag{63b}$$

Si el camión sale de la granja i en el día 0, entonces también entra en el día 0.
$(\forall i \in \mathit{NoDiarias})$

$$\sum_{j \in \mathit{Granjas}} E_{0ji} + M \times Y_i \geq 1 \tag{64}$$

$(\forall i \in \mathit{NoDiarias})$

$$-\sum_{j \in \mathit{Granjas}} E_{0ij} - M \times Y_i \geq -M \tag{65}$$

Si el camión entra de la granja i en el día 0, entonces también sale en el día 0.
$(\forall i \in \mathit{NoDiarias})$

$$\sum_{j \in \mathit{Granjas}} E_{0ij} + M \times Y_i \geq 1 \tag{66}$$

$(\forall i \in \mathit{NoDiarias})$

$$-\sum_{j \in \mathit{Granjas}} E_{0ji} - M \times Y_i \geq -M \tag{67}$$

Restricciones de subtour: No hay ciclos de tamaño 2
$(\forall d \in \mathit{Días})(\forall i \in \mathit{Granjas})(\forall j \in \mathit{Granjas})$

$$E_{dji} + E_{dij} \leq 1 \tag{68}$$

Restricciones de subtour: No hay ciclos de tamaño 3
$(\forall d \in \mathit{Días})(\forall i \in \mathit{Granjas})(\forall j \in \mathit{Granjas})(\forall h \in \mathit{Granjas})$

$$E_{dij} + E_{djh} + E_{dhi} \leq 2 \tag{69}$$

8.5. Función Objetivo

Se quiere minimizar la distancia recorrida para la recolección de la lecha de las granjas, partiendo y volviendo al deposito.
Min

$$\sum_{d \in \mathit{Días}} \sum_{i \neq j \in \mathit{Granjas}} E_{dij} \times \mathit{Distancia}_{ij} \tag{70}$$

8.6. Solución del problema

El valor de la función objetivo, es decir, la distancia recorrida en los dos días para la recolección, es 121,377.

Los recorridos para los días 1 y 2 son:
$0 \to 9 \to 5 \to 19 \to 6 \to 8 \to 20 \to 7 \to 12 \to 3 \to 14 \to 2 \to 13 \to 4 \to 1 \to 0$
y
$0 \to 9 \to 5 \to 6 \to 10 \to 8 \to 7 \to 11 \to 3 \to 15 \to 18 \to 2 \to 4 \to 17 \to 1 \to 16 \to 0$

9. Experimentación Ejercicio 12.1

Se quiere analizar el valor aportado por diferentes políticas de selección de nodos, heurísticas y cortes a medida que el tamaño de entrada del problema aumenta en este problema.

El problema formulado es lineal con lo que no se necesita utilizar una técnica de branch & bound ni resolver un árbol MIP. Es por esto esperado que diferentes formas de selección de nodos y cortes obtengan tiempos similares.

En cuanto al agregado de heurísticas, no se tiene conocimiento detallado sobre el orden de ejecución de la herramienta. Utilizando o no una heurística se deberá ejecutar simplex (o algoritmo equivalente) para resolver el sistema lineal. Si primero intenta obtener una solución utilizando una heurística, aunque esta pueda ser una solución básica factible, se espera que los tiempos por correr primero esta y luego simplex sean mayores a solo utilizar simplex. Dado que las variables son reales si primero intenta resolver el problema utilizando simplex antes de correr una heurística, los tiempos obtenidos deberían ser del mismo orden.

Para el caso de cortes se interesa conocer la diferencia en tiempos de ejecución aplicando o no cortes covers y clique, limitando o no la cantidad de cortes de planos en el nodo raíz del modelo.

9.1. Generación de Instancias

Con el modelo presentado y dados m meses y a aceites se tiene $m\times a\times 3$ variables de decisión y $m\times 4+m\times a\times 2$ restricciones, 2 por las restricciones en cantidades usadas, 2 por las restricciones de dureza, y luego tantas como variables de $Stock_{ma}$ haya para calcularlo en base al anterior y acotarlo superiormente.

Crecer en uno 1 cantidad de meses impacta en el modelo agregando $3 \times |Aceites|$ variables de decisión y $4 + |Aceites| \times 2$ restricciones nuevas, 2 para restringir el límite de uso, 2 para la dureza y $|Aceites| \times 2$ para los $Stock_{ma}$ del nuevo mes.

Por otro lado crecer con un nuevo tipo de aceite impacta en el modelo agregando $3 \times |Meses|$ variables de decisión nuevas y $|Meses| \times 2$ restricciones nuevas correspondientes al $Stock_{ma}$ del nuevo aceite en todos los meses.

Se decide generar instancias del problema con:

1. Igual cantidad de *Aceites* y *Meses*.
2. Con $|AceitesVegetales| = Aceites \times \frac{2}{5}$ del total de aceites. Elegido de forma de mantener el mismo porcentaje presente en el ejemplo del enunciado.
3. Valores random para la $Dureza_j$ entre 2 y 8 para no introducir un aceite inviable de utilizar.
4. Valores random de $Precio_{ji}$ entre 50 y 150.

Es interesante notar que al comenzar y al terminar el stock de aceites debe ser el mismo por lo que no comprar ni utilizar aceite es una solución factible básica con todas las variables en 0, lo que hace que estas instancias sean solubles sin importar los valores de $Dureza_j$ y $Precio_{ji}$.

9.2. Metodología de Experimentación

Se corrió en una computadora del laboratorio del DC un archivo cpp utilizando la librería de *CPLEX* provisto en la entrega con este informe. Las instancias, también provistas, tienen un tamaño creciente de 20 en 20 con un tamaño inicial de 6 *Aceites* y *Meses*. Se corrió cada una 10 veces midiendo sus tiempos, se quito outliers y se tomo promedio.

CPLEX incluye en la salida por standard output el tiempo de corrida, pero esto no es accesible a través de métodos de la librería. Se decidió tomar tiempos desde el código *C++* antes y después de la llamada a *CPLEX* para resolver el sistema, siendo esto recomendado por *IBM* en su documentación[1].

Se fijó de manera arbitraria una configuración de *CPLEX* para los siguientes parámetros con el fin de obtener una medición más afín:

1. *CPX_PARAM_TILIM*: 900 segundos
2. *CPX_PARAM_PARALLELMODE*: 1 (determinista)
3. *CPX_PARAM_THREADS*: 1 thread
4. *CPX_PARAM_MIPSEARCH*: CPX_MIPSEARCH_TRADITIONAL
5. *CPX_PARAM_MIPCBREDLP*: CPX_OFF
6. *CPX_PARAM_PRELINEAR*: 0 (deshabilitado la reducción linear de variables)
7. *CPX_PARAM_REDUCE*: 0 (ninguna reducción)
8. *CPX_PARAM_REPEATPRESOLVE*: 0
9. *CPX_PARAM_PROBE*: -1

9.3. Resultados Obtenidos con el Modelo Lineal

Primero se quiere saber que política de selección de nodos es la mejor. Para esto se deshabilitaron heurísticas y cortes.

Selección de Nodos

Como se puede ver en la figura 1, los tiempos obtenidos para las diferentes políticas de selección de nodos es el mismo (una diferencia menor a un 500 milisegundos en el peor de los casos). También, revisando la salida de *CPLEX* se verificó que nunca se recorriera un nodo diferente a la raíz, que era lo esperado.

Heurísticas

Para comparar los tiempos utilizando distintas heurísticas se deshabilitaron todos los cortes y se decidió arbitrariamente por *DFS* como política de selección de nodos.

Los resultados de la figura 2 tampoco presentan una diferencia en tiempos de ejecución entre las diferentes heurísticas (nuevamente menor a un segundo).

La salida de *CPLEX* refleja que se corrieron cada una de las heurística previo a resolver el sistema a pesar de ser real. También muestra que el tiempo aplicado en obtener una solución utilizando las heurísticas crece con el tamaño de la entrada, pero llegando a ser a lo sumo 80

[1] http://www-01.ibm.com/support/docview.wss?uid=swg21401522

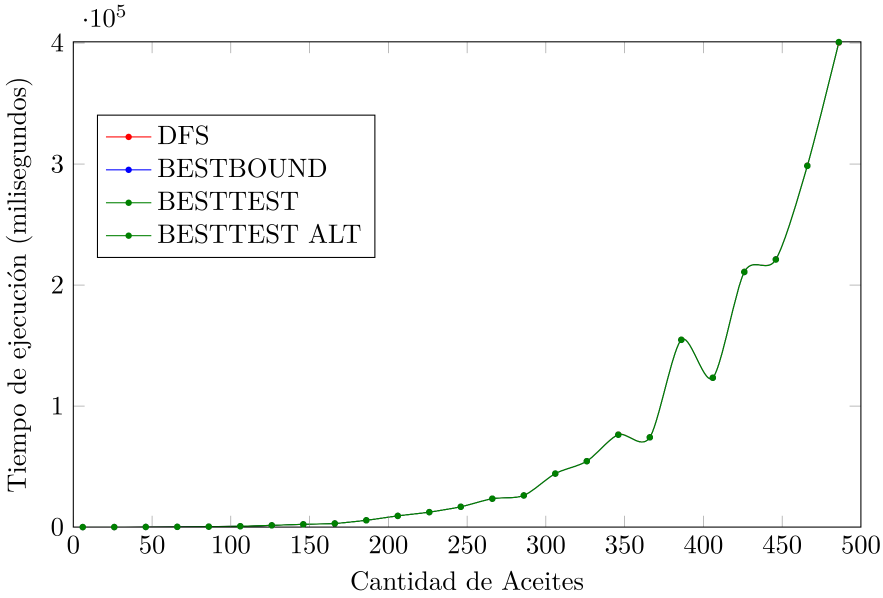

Figura 1: Tiempos de ejecución con diferentes selección de nodos sin cortes ni heurísticas sobre el modelo lineal.

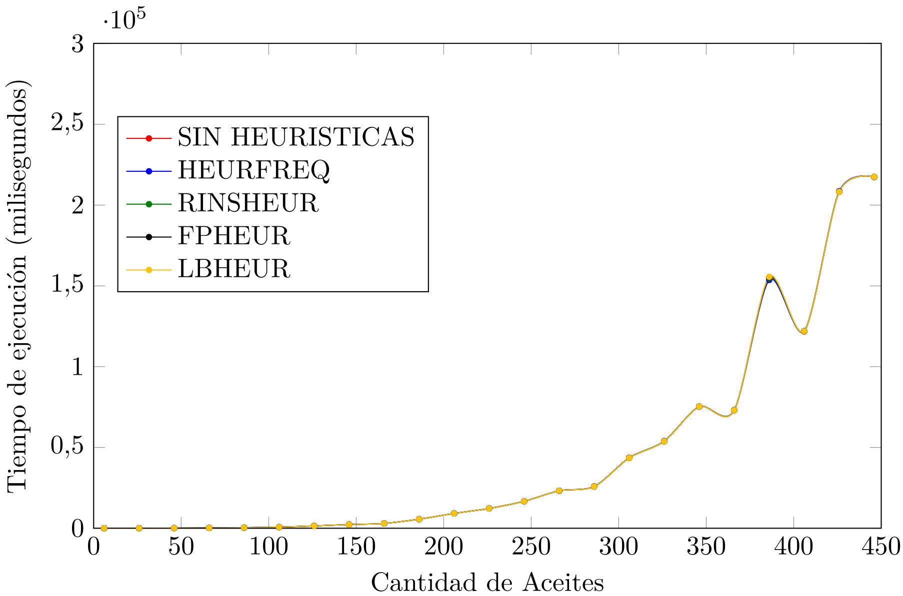

Figura 2: Tiempos de ejecución con diferentes heurísticas sin cortes sobre el modelo lineal.

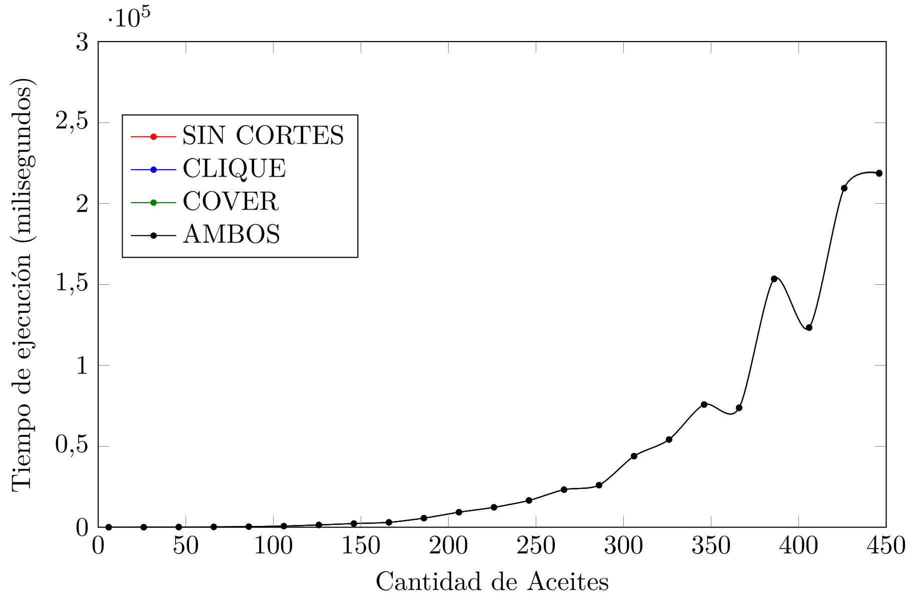

Figura 3: Tiempos de ejecución con diferentes cortes sin heurísticas ni planos de cortes en la raíz y usando DFS como selección de nodos para el modelo lineal.

milisegundos para con las instancias más grandes, lo que representa aproximadamente el $\%0,03$ del total del tiempo de corrida.

Cortes

Para comparar los cortes se deshabilitaron las heurísticas y se eligió nuevamente *DFS* como política de selección. Se tiene en cuenta el parámetro *CPX_PARAM_CUTPASS* de CPLEX, el cual limita la cantidad de cortes de planos en la raíz.

Como se ve en las figuras 3 y 4 los tiempos de todas las combinaciones obtuvieron el mismo resultado con una diferencia de a lo sumo un segundo en el peor de los casos. El output de *CPLEX* tampoco muestra registros de cortes. Este al igual que los resultados previos, eran de esperarse dado que no se recorre más de un nodo.

Combinación

Dado los resultados, la combinación que mejor aplica a este problema es no utilizar heurísticas ni cortes, y cualquier política de selección.

Algo que se podría analizar ahora es si haber cambiado alguno de los flags seteados en la sección 9.2 podría reducir los tiempos.

La figura 5 refleja que las reducciones en el modelo previas a la ejecución de simples obtienen mejores resultados. Lo mismo se constata con la salida de *CPLEX*.

Durante la etapa de preprocesamiento, no utilizar los flags toma $\%5$ del tiempo total incurrido haciendo preprocesamiento. Esto se debe a que la corrida con los flags realiza más de un intento por borrar filas y columnas.

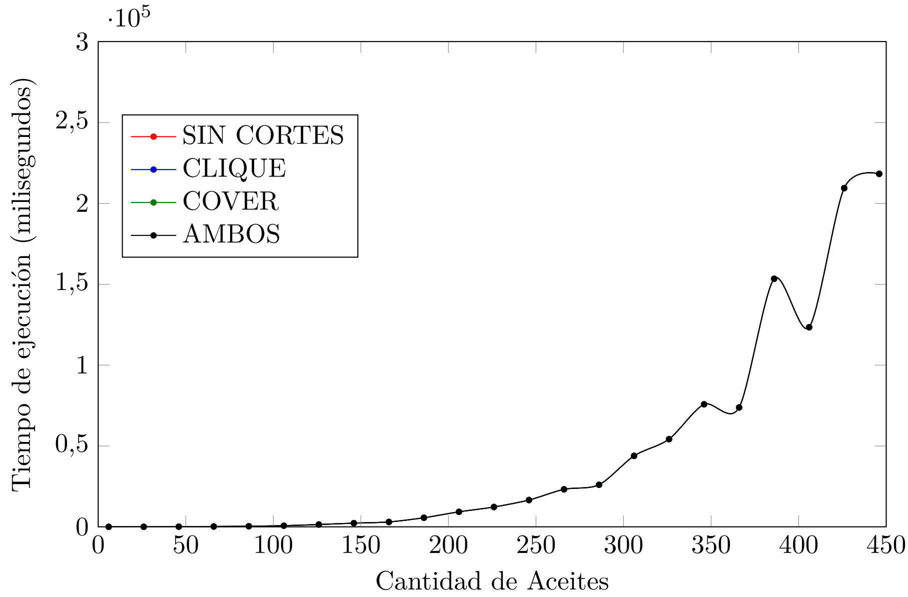

Figura 4: Tiempos de ejecución con diferentes cortes sin heurísticas, con planos de cortes en la raíz y usando DFS como selección de nodos para el modelo lineal.

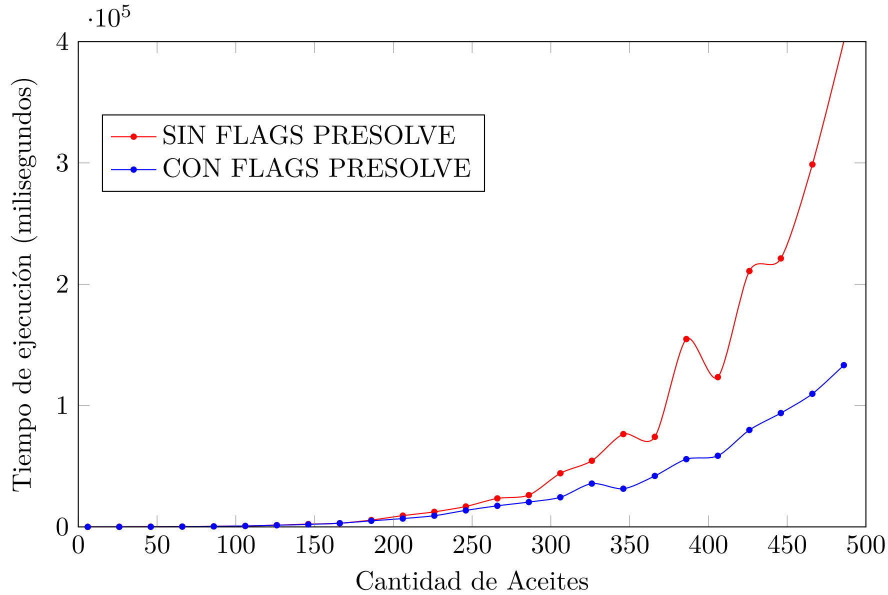

Figura 5: Tiempos de ejecución con la mejor combinación para el modelo lineal con y sin utilizar preprocesado.

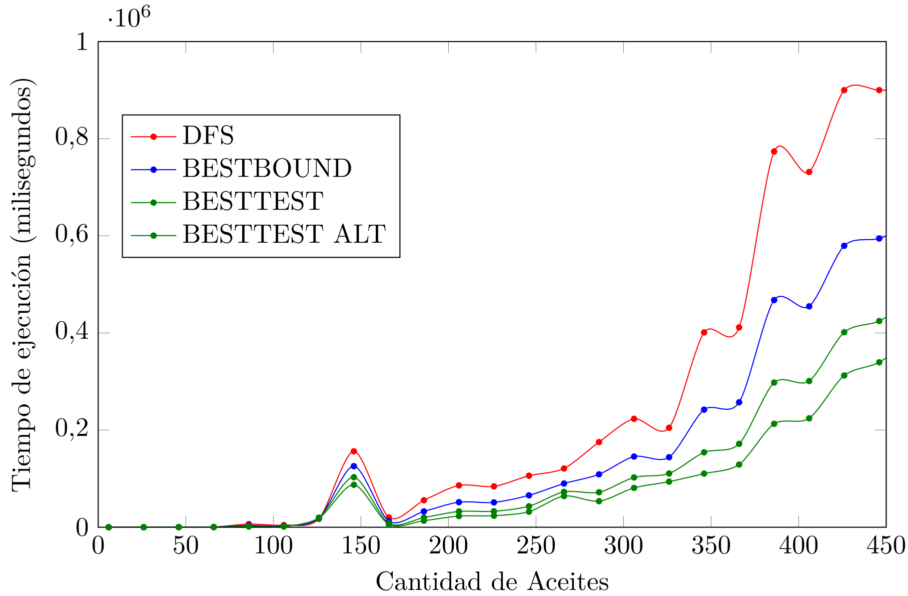

Figura 6: Tiempos de ejecución con diferentes selección de nodos sin cortes ni heurísticas para el modelo entero.

La salida de *CPLEX* refleja que aunque mayor es el tiempo de preprocesado utilizando los flags sea mayor (al rededor del %1833), este tiempo es pequeño (menor a 2 segundos en las instancias más grandes). Mientras que el tiempo de procesador en el nodo raíz sin utilizar este preprocesado es al menos 2 veces más grande a partir de la instancia de tamaño 386, que es considerablemente mayor al tiempo común incurrido en el preprocesamiento.

9.4. Resultados del Modelo Entero

Se quiere repetir el mismo análisis sobre el mismo modelo pero considerando ahora que todas las variables de decisión son enteras.

El método de experimentación es el mismo y se utiliza el mismo input. En este caso se espera que los tiempos de corrida sean mayores en general y que pueda apreciarse una diferencia considerable entre las diferentes selecciones de nodos, heurísticas y cortes.

Selección de Nodos

A diferencia de los resultados obtenidos en el modelo lineal, se puede observar en la figura 6 diferencias considerables entre las diferentes formas de selección de nodos desde instancias con un tamaño mayor a 186 Aceites.

Como se esperaba, los tiempos fueron considerablemente más grandes a los respectivos en el modelo lineal. La política *DFS*, elegida arbitrariamente en el modelo lineal dentro de la mejor combinación, fue la que más altos tiempos tuvo.

La salida de *CPLEX* muestra que mientras con *DFS* la exploración del árbol MIP toma más de 300 segundos a partir de la instancia de tamaño 346, las demás opciones toman menos de 35

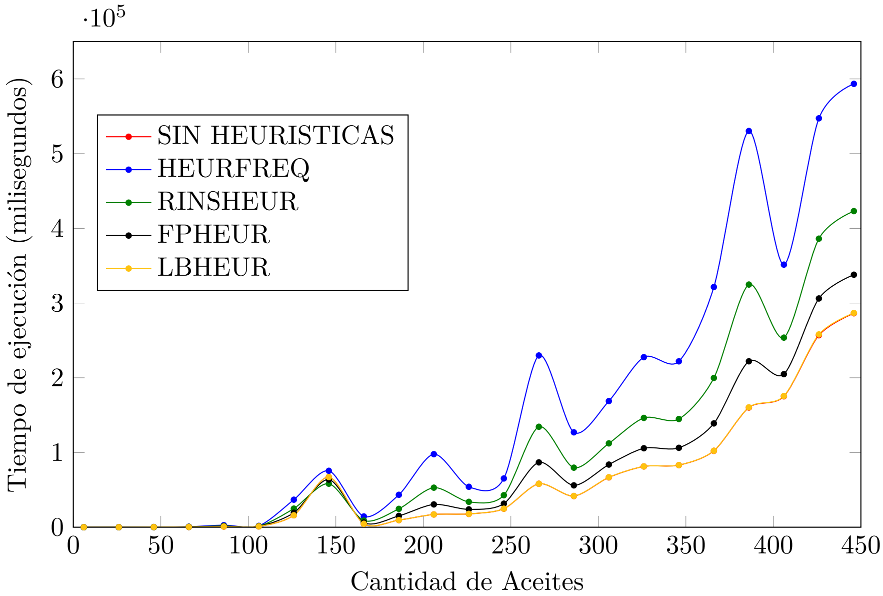

Figura 7: Tiempos de ejecución con diferentes heurísticas sin cortes y con BESTTEST ALT como selección de nodos para el modelo entero.

segundos para la instancia más grande (de tamaño 446).

La salida de *CPLEX* también permite ver que los tiempos de exploración del árbol para las otras 3 opciones varían en menos de 5 segundos. Lo cual no puede verse siguiendo la recomendación de *IBM* para tomar tiempos.

Sin embargo, los resultados muestran que la política de selección de nodos *BESTTEST ALT* en general obtiene tiempos más bajos y lo mismo se puede constatar desde la salida de *CPLEX* aunque la diferencia sea menor a 5 segundos. Se utiliza esta para los siguientes experimentos.

Heurísticas

Para comparar los tiempos utilizando distintas heurísticas se deshabilitaron todos los cortes y se decidió *BESTTEST ALT* como política de selección de nodos.

Los resultados de la figura 7 muestran que tanto no utilizar heurísticas como la utilización de la heurística *LBHEUR* consiguen los tiempos más bajos. Señalando también la elección de la heurística *HEURFREQ* como la peor en general para estas entradas.

El tiempo en la salida de *CPLEX* respaldan los vistos en la figura para *LBHEUR*, *SIN HEURÍSTICAS* y *HEURFREQ*. No sucede lo mismo con las otras dos, las cuales arrojan tiempos a lo sumo 10 segundos mayores a *LBHEUR* en todos los casos.

Ante estos datos, se opta por no utilizar heurísticas.

Cortes

Los resultados de las figuras 8 y 9 muestran que las diferentes alternativas de cortes obtienen tiempos similares con una diferencia menor a un segundo.

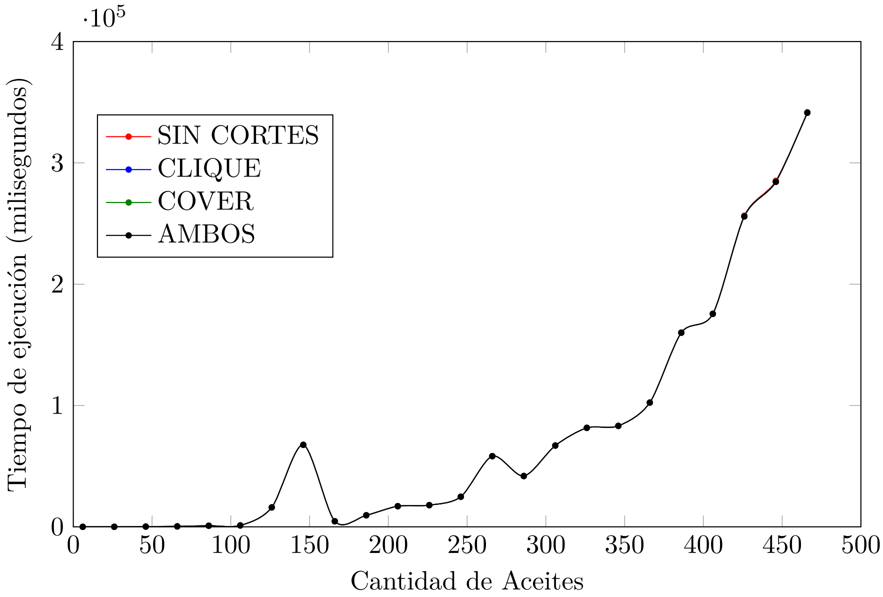

Figura 8: Tiempos de ejecución con diferentes cortes sin heurísticas ni planos de cortes en la raíz y con BESTTEST ALT como selección del próximo nodo para el modelo entero.

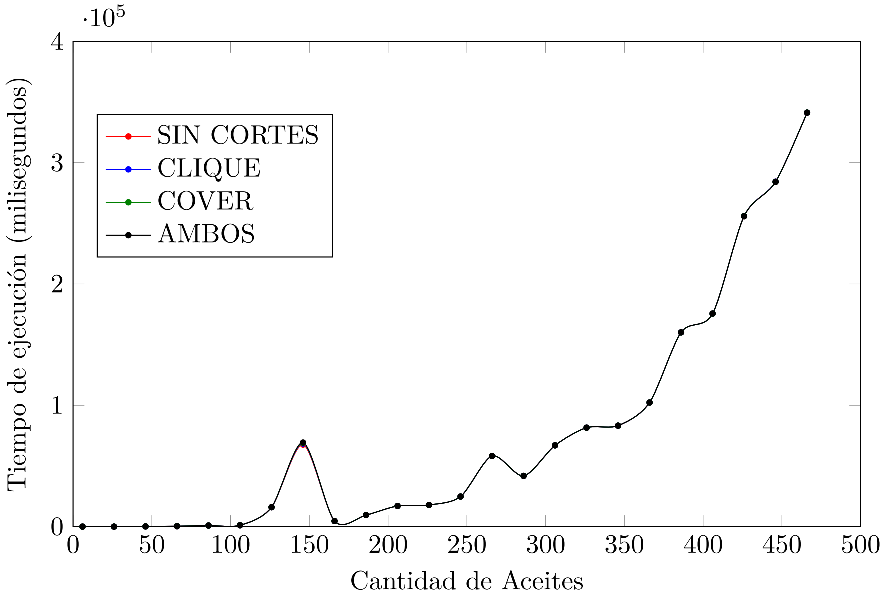

Figura 9: Tiempos de ejecución con diferentes cortes sin heurísticas, con planos de cortes en la raíz y y con BESTTEST ALT como selección del próximo nodo para el modelo entero

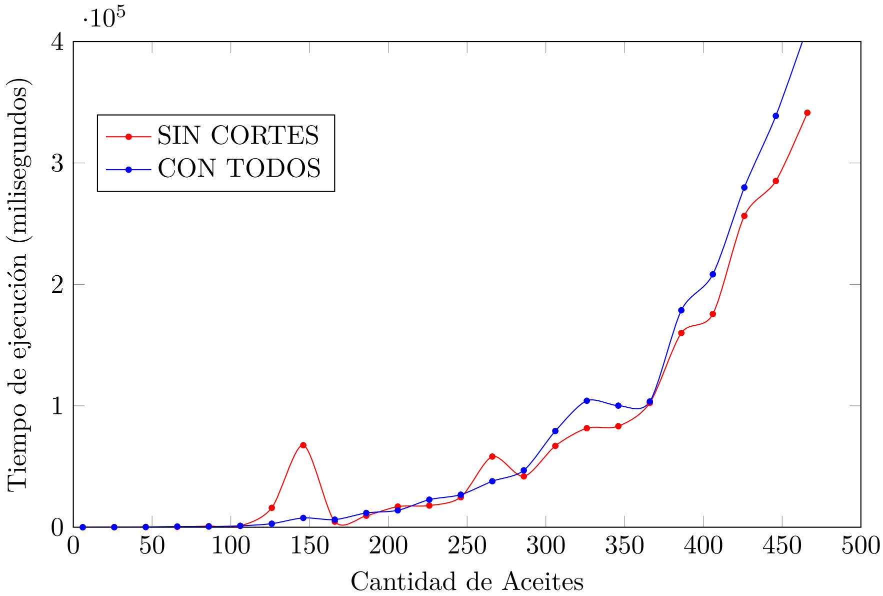

Figura 10: Tiempos de ejecución sin heurísticas y con BESTTEST ALT como selección de nodos para el modelo entero

La salida de *CPLEX* en este caso refleja los mismo resultados. También indica que no se han realizado cortes de ninguno de estos dos tipos, lo cual explica el resultado obtenido. Por lo tanto, los resultados no respaldan que el uso de cortes covers o clique realice una mejora en los tiempos.

Algo que se podría analizar es si incluyendo más cortes se podría disminuir los tiempos obtenidos. Para esto se corrió el mismo input para una configuración con todos los cortes habilitados, sin heurísticas y con la misma política de selección de nodos.

Los resultados de la figura 10 no indican una mejora que justifique utilizar esta alternativa. La salida de *CPLEX* refleja que la versión con todos los cortes aplica cortes gomory y mixed rounding. También en general es un %50 más grande el tiempo de procesamiento en la raiz del árbol MIP. El cual fue factor decisivo para la obtención de tiempos más altos en este caso.

Combinación

Dados los resultados, la combinación que mejor aplica al modelo entero es utilizar *BESTTEST ALT* como método de selección de nodos, no utilizar heurísticas y no utilizar cortes (o al menos no estos dos).

9.5. Conclusiones

Los datos favorecen el no aplicar cortes ni heurísticas cuando el problema es lineal, lo cual tiene sentido si simplemente se precisa ejecutar simplex o algún algoritmo equivalente. No sucede lo mismo con el preprocesamiento, el tiempo incurrido aplicándolo puede ser despreciable respecto al tiempo total que incurre el procesamiento en el nodo raíz y favorecer los tiempos totales.

Aunque se aplicó una cantidad aproximada de 100 cortes entre cortes gomory y mixed round en las instancias más grandes del problema, el tiempo que le llevó a *CPLEX* realizar esta tarea significó tiempos totales mayores. Es interesante para futuros experimentos probar solo estos cortes y limitando la cantidad máxima de cortes aplicados en la raíz.

Para el caso del modelo entero las heurísticas no demostraron favorecer a los tiempos de ejecución. Sería interesante como futuros experimentos proveer a *CPLEX* de soluciones iniciales con heurísticas desarrolladas para el problema.

10. Experimentación Ejercicio 12.23

Se quiere buscar los mejores parámetros de configuración de *CPLEX* respecto a selección de nodos, cortes y heurísticas sobre el modelo del problema 12.23. El objetivo es comparar y analizar si sería necesario customizar estos parámetros o si *CPLEX* ya resuelve esto.

Primero se extiende el problema para que utilice 40 granjas en vez de 20 y así para poder medir con mejor claridad los tiempos.

Resolviendo el sistema con todas las configuraciones default de *CPLEX* se obtiene un tiempo de ejecución de $71,03$ segundos en promedio. Se utilizará este valor para futuras comparaciones.

10.1. Cortes

Por defecto, *CPLEX* tiene activado todos los cortes disponibles, los cuales decide o no utilizar y que tan seguido.

El objetivo es sólo usar un tipo de corte, por lo que se toman tiempos para cada corte con los demás inhabilitados.

Se puede ver en la figura 11 que hay 3 cortes que se destacan por disminuir considerablemente el tiempo de ejecución. Son los cortes por cliques, Gomory fractional y zero-half. El corte zero-half obtuvo los menores tiempo de ejecución, lo cual fue aproximadamente 20 segundos menos que los obtenidos con los valores default.

Es de interés saber también, que es lo que sucede al combinar varios cortes distintos. Si es que al acumular cortes mejoran o se degradan los tiempos. Para esto se corre el programa aumentando la cantidad de cortes utilizados y tomando los cortes en el orden de tiempos que vimos en el análisis anterior.

Como se puede ver en la figura 12, salvo en el caso del uso de 4 cortes, los tiempos no mejoran con respecto a utilizar un solo corte (en este caso, zero-half).

10.2. Heurísticas

Al igual que antes con los cortes, se analizan los tiempos individuales de cada heurística y se comparan con el tiempo obtenido con los valores default y el tiempo al no utilizando heurísticas.

Por default, *CPLEX* utiliza todas las heurísticas, solamente si ve que están ayudando.

Como se puede ver en la figura 13 la heurística de local branching es la que obtiene tiempos más bajos, incluso que los valores default. Pero más notable es que no utilizar heurísticas es mejor que el default para este caso. Esto se puede deber a que el problema no es lo suficientemente grande para que valga la pena utilizar ciertas heurísticas.

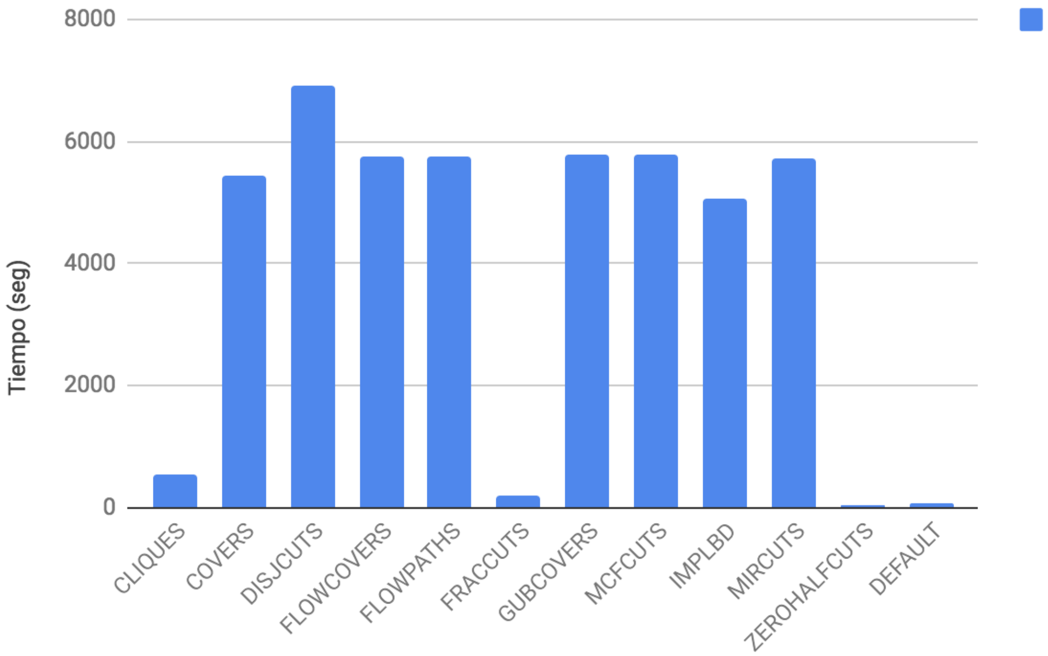

Figura 11: Tiempos de ejecución promedio sin heurísticas, habilitando solo un tipo de corte a la vez.

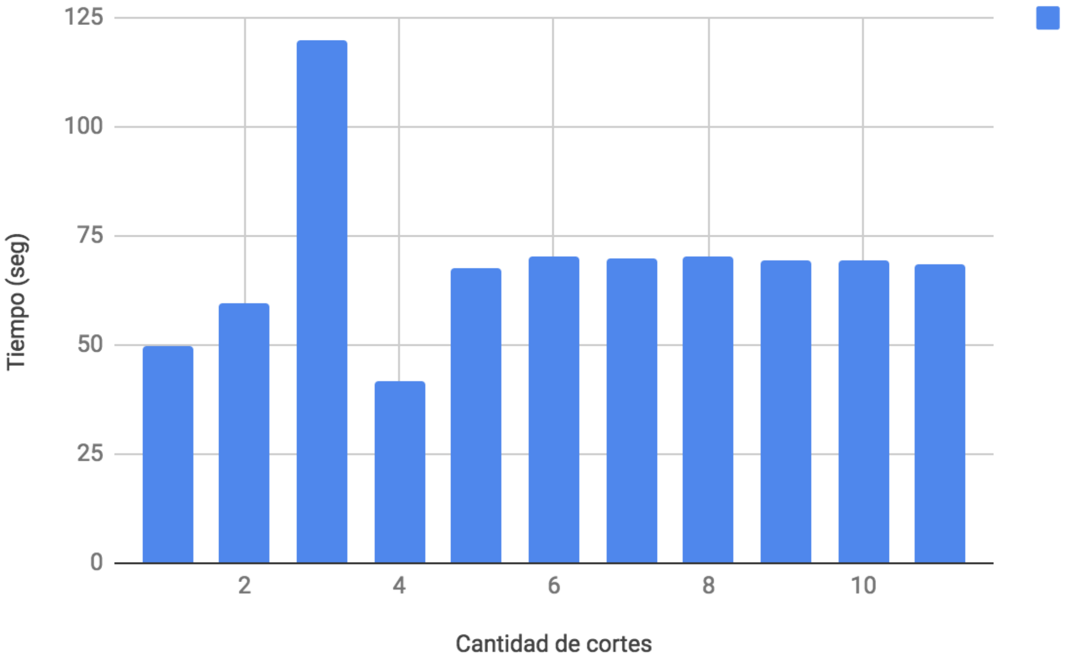

Figura 12: Tiempos de ejecución aumentando la cantidad de cortes zero-half.

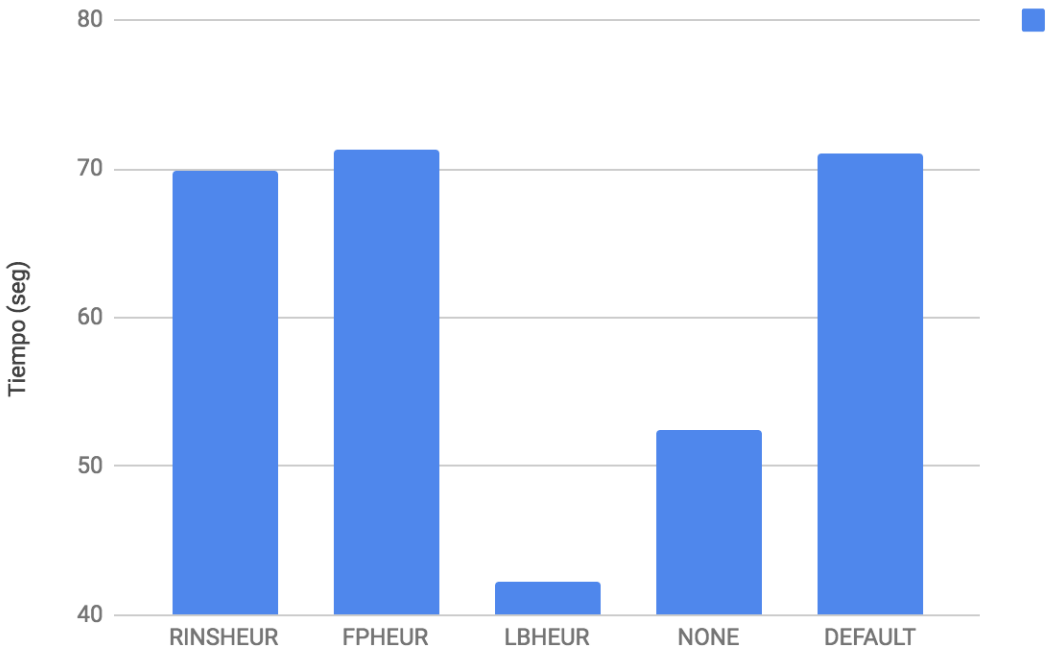

Figura 13: Tiempos de ejecución con distintas opciones de heurísticas por separado y sin ninguna.

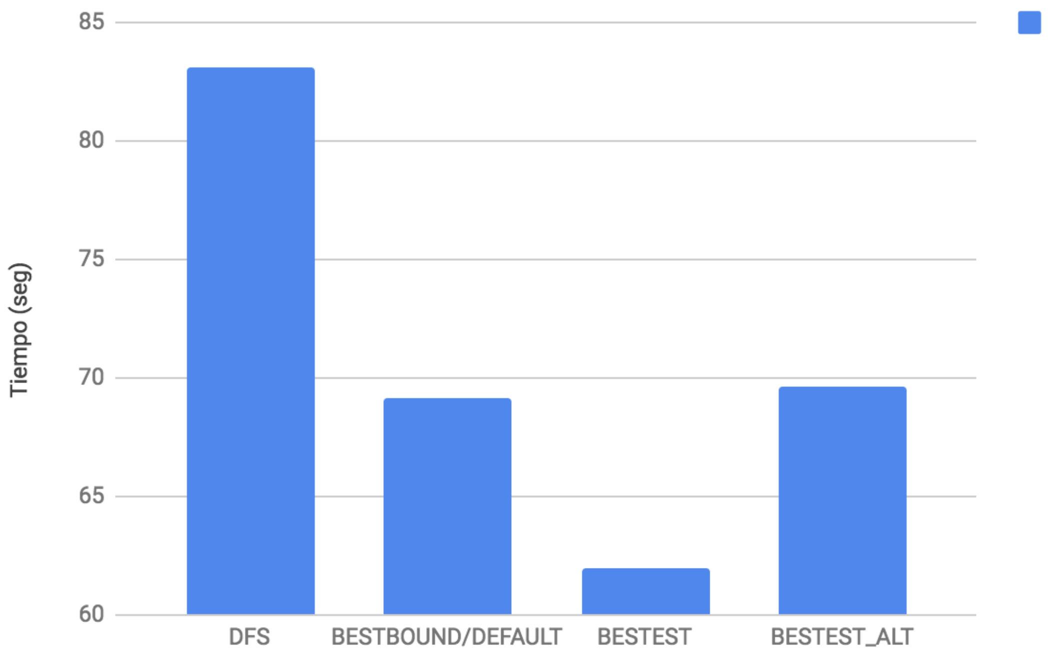

Figura 14: Tiempos de ejecución con distintas opciones de selección del próximo nodo.

10.3. Selección nodos

Se repite una vez más el análisis aplicado ahora sobre las reglas de selección del próximo nodo. Por default, *CPLEX* utiliza la regla de best-bound.

Se puede ver en la figura 14 que la regla de best-estimate search es la que mejor aplica para este caso.

10.4. Selección variables branching

Por último, se analiza la selección de variables en el branching. Por default, *CPLEX* selecciona la mejor regla basado en el problema y en su progreso.

Se puede ver en la figura 15 que la regla que obtiene los menores tiempo, incluso mejores que default, es minimum infeasibility. Esta elige la variable con el valor más cercano a un entero, pero todavía fraccionaria.

10.5. Conclusión

En cada eje de análisis se encontró una configuración de parámetros que obtiene mejores tiempos de ejecución en el sistema planteado que las default de *CPLEX*. En algunos casos con mejoras de hasta %40.

Esto permite concluir que en problemas suficientemente grandes vale la pena realizar un análisis para encontrar los mejores cortes, heurísticas, etc. para el sistema.

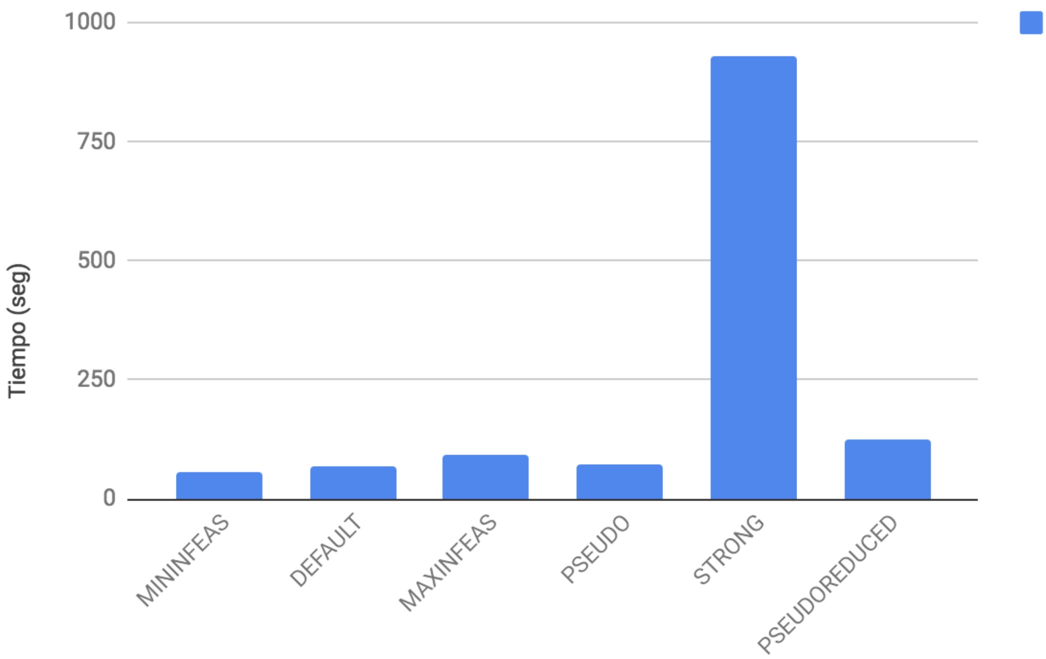

Figura 15: Tiempos de ejecución de distintas opciones de branching.

11. Ejercicio 10 de la Práctica 8

Dado el sistema *Maximizar*

$$x_1 + x_2 + x_3 + x_4 + x_5 \tag{71}$$

Sujeto a

$$3 \times x_1 + 2 \times x_2 + x_3 + 2 \times x_4 + x_5 \geq 4 \tag{72}$$

$$-2 \times x_1 + 6 \times x_2 + 5 \times x_3 - 2 \times x_4 - 4 \times x_5 \geq 2 \tag{73}$$

$$4 \times x_1 - 4 \times x_2 - 14 \times x_3 + 2 \times x_4 + 3 \times x_5 \geq 4 \tag{74}$$

$$x_1, x_2, x_3, x_4, x_5 \in \{0, 1\} \tag{75}$$

11.1. Resolver utilizando enumeración implícita

Si $x_3 = 1$ debe valer por ecuación 74 que $4 \times x_1 - 4 \times x_2 + 2 \times x_4 + 3 \times x_5 \geq 18$, lo cual no vale porque a izquierda como mucho puedo sumar 10 con $x_1 = 1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 1$. Entonces $x_3 = 0$.

Si $x_2 = 0$ debe valer por ecuación 73 que $-2 \times x_1 + 5 \times x_3 - 2 \times x_4 - 4 \times x_5 \geq 2$. Se que $x_3 = 0$, entonces debe valer que $-2 \times x_1 - 2 \times x_4 - 4 \times x_5 \geq 2$ lo cual no vale porque a izquierda como mucho puedo sumar 0 con $x_1 = 0$, $x_2 = 0$, $x_4 = 0$. Entonces $x_2 = 1$.

Si $x_5 = 0$ debe valer por ecuación 74 que $4 \times x_1 - 4 \times x_2 - 14 \times x_3 + 2 \times x_4 \geq 4$. Como $x_2 = 1$ y $x_3 = 0$, debe valer que $4 \times x_1 + 2 \times x_4 \geq 8$, lo cual no vale porque a izquierda como mucho puedo sumar 6 con $x_1 = 1$, $x_4 = 1$. Entonces $x_5 = 1$.

Si $x_1 = 1$ debe valer por ecuación 73 que $6 \times x_2 + 5 \times x_3 - 2 \times x_4 - 4 \times x_5 \geq 4$. Como $x_2 = 1$, $x_3 = 0$ y $x_5 = 1$, debe valer que $-2 \times x_4 \geq 2$, lo cual no vale nunca para ningún valor $\{0, 1\}$ de x_4. Entonces $x_1 = 0$. Pero entonces debe valer por ecuación 74 que $-4 \times x_2 - 14 \times x_3 + 2 \times x_4 + 3 \times x_5 \geq 4$, y como $x_2 = 1$, $x_3 = 0$, $x_5 = 1$, debe valer $2 \times x_4 \geq 5$, que no vale para los valores $\{0, 1\}$ x_4.

El problema no tiene solución factible.

11.2. Verificar que $5 \times x_1 + 4 \times x_2 - 8 \times x_3 + 2 \times x_4 \geq 10$ es una desigualdad válida

Podemos expresar nuestro problema como

$$min\{c \times x / x \in X\} \tag{76}$$

Con $c = [1, 1, 1, 1, 1]$ y $X = \{x / r_1 \times x \geq b_1, r_2 \times x \geq b_2, r_3 \times x \geq b_3, x \in \{0, 1\}^5\}$ donde r_i son los coeficientes de la $i - ésima$ restricciones del modelo y b_i el lado derecho de la misma.

Como $x \in X$ vive en la intersección de los 3 semi espacios, satisface las 3 restricciones al mismo tiempo y combinaciones lineales de estas.

Si en especial las combino linealmente multiplicadas por 1 a cada una, puedo decir que $x \in X$ satisface $r_1 \times x + r_2 \times x + r_3 \times x \geq b_1 + b_2 + b_3$. Que reemplazando por sus respectivos valores en el modelo queda

$$\begin{aligned} &(3 \times x_1 + 2 \times x_2 + x_3 + 2 \times x_4 + x_5) \\ &+(-2 \times x_1 + 6 \times x_2 + 5 \times x_3 - 2 \times x_4 - 4 \times x_5) \\ &+(4 \times x_1 - 4 \times x_2 - 14 \times x_3 + 2 \times x_4 + 3 \times x_5) \geq 2 + 4 + 4 \end{aligned} \tag{77}$$

que puede ser reescrito como

$$\begin{aligned} (3 - 2 + 4) \times x_1 + (2 + 6 - 4) \times x_2 + (1 + 5 - 14) \times x_3 \\ +(2 - 2 + 2) \times x_4 + (1 - 4 + 3) \times x_5 \\ \geq 10 \end{aligned} \tag{78}$$

que es lo mismo que

$$5 \times x_1 + 4 \times x_2 - 8 \times x_3 + 2 \times x_4 + 0 \times x_5 \geq 10 \tag{79}$$

que exactamente lo que se quería verificar.

11.3. Agregar la desigualdad anterior y resolver por enumeración implícita

La resolución utilizada en la primer parte de este ejercicio sigue siendo válida.