

Performance of Logic-Based Benders Decomposition for Improving the Scalability of a VRP with Time Windows

Diego Nicolás Gimenez Irusta , María Analía Rodríguez 

Instituto de Investigación y Desarrollo en Ingeniería de Procesos y Química Aplicada,
Universidad Nacional de Córdoba – CONICET, Córdoba, Argentina
diego.gimenez@unc.edu.ar, analia.rodriguez@unc.edu.ar

Abstract

The long-haul transportation planning problem addressed in this work involves determining the order in which destinations are visited, as well as the scheduling of loading and unloading operations. These decisions must ensure that weekly demand is satisfied under various operational constraints, with the objective of minimizing the required fleet size while maximizing the truck utilization. Initially, a monolithic formulation was developed that faithfully represents the problem and is capable of solving small instances satisfactorily. However, this formulation poses a significant challenge for realistic instances, as it requires excessive computational resources. In such cases, the size of the feasible region grows rapidly, substantially increasing the difficulty of solving the model to optimality. In this work, two strategies aimed at improving the scalability of the model are analyzed. Given the high level of symmetry present in the formulation, the first strategy consists in eliminating equivalent solutions through symmetry-breaking constraints. Finally, the most effective approach proved to be a decomposition technique known as Logic-Based Benders Decomposition, which improves scalability without compromising optimality guarantees. The results indicate that this decomposition strategy is beneficial, as it enables the solution of real-scale instances using a reasonable amount of computational resources.

Keywords: Vehicle routing problem; logic-based Benders decomposition; combinatorial optimization; symmetry; time windows

Desempeño de la descomposición de Benders basada en lógica en la escalabilidad de un problema VRP con ventanas temporales

Resumen

El problema de planificación del transporte de larga distancia abordado en este trabajo consiste en decidir el orden de visitas a los destinos, así como los horarios de carga y descarga. Estas decisiones deben permitir satisfacer una demanda semanal bajo diversas restricciones operativas, con el objetivo de minimizar el tamaño de la flota requerida y lograr su máximo aprovechamiento. Inicialmente se desarrolló una formulación monolítica que representaba fielmente el problema, resolviendo instancias pequeñas

satisfactoriamente. Sin embargo esta formulación presenta un desafío significativo para instancias realistas, pues requería excesivos recursos computacionales. En estos casos, el tamaño de la región factible crece rápidamente, lo que incrementa de manera considerable la dificultad para la convergencia del modelo. En este trabajo se analizan dos estrategias orientadas a mejorar su escalabilidad. Como el modelo presentaba un alto nivel de simetría, la primera estrategia consistió en eliminar algunas soluciones equivalentes con restricciones específicas. Por último, la más efectiva, resultó ser una técnica conocida como descomposición de Benders basada en lógica (*Logic-Based Benders Decomposition*, LBBD), la cual permite mejorar la escalabilidad del modelo sin perder la garantía de optimalidad. Los resultados obtenidos indican que esta estrategia de LBBD resulta beneficiosa, ya que permite resolver instancias de escala real utilizando una cantidad de recursos computacionales aceptable.

Palabras clave: Problema de ruteo de vehículos; descomposición de Benders basada en lógica; optimización combinatoria; simetría; ventanas de tiempo

1. Introducción

Acorde a lo planteado por la FADEEAC (nd), el índice de costos del transporte de cargas en Argentina se encuentra en constante crecimiento. En particular, el incremento acumulado registrado durante los últimos años evidencia una tendencia sostenida al alza en los costos operativos del sector. Esta dinámica impacta directamente sobre las decisiones logísticas y de planificación, ya que modificaciones en los costos de transporte pueden alterar la rentabilidad de determinadas rutas, la asignación de recursos y la conveniencia relativa de distintas alternativas operativas. En consecuencia, resulta relevante considerar estos costos como una variable dinámica dentro de los modelos de optimización y análisis de escenarios.

En este contexto, los modelos de optimización matemática son una herramienta central para mejorar la eficiencia del transporte. Sin embargo, estos modelos escalan rápidamente debido a su naturaleza combinatoria: pequeñas variaciones en la cantidad de nodos, vehículos o periodos generan incrementos significativos en el número de configuraciones posibles. Además, muchos problemas clásicos de transporte son NP-difíciles, por lo que incluso instancias moderadas pueden requerir tiempos de resolución elevados y un uso intensivo de recursos computacionales.

En aplicaciones reales, la necesidad de representar fielmente las operaciones —incluyendo múltiples circuitos, ventanas horarias y restricciones operativas— incrementa significativamente el tamaño del modelo y su complejidad (Toth y Vigo, 2014).

En este trabajo se analizan dos estrategias para mejorar la escalabilidad de un modelo de optimización monolítico, el cual, si bien representa adecuadamente el problema real, presenta altos requerimientos computacionales. En particular, se proponen dos enfoques: (i) la incorporación de restricciones de rotura de simetría para eliminar soluciones equivalentes, y (ii) la aplicación de LBBD, que permite obtener soluciones óptimas de manera más eficiente.

2. Descripción del problema

2.1. Contexto operativo

El problema abordado consiste en la planificación semanal del transporte de cargas desde una planta de origen hacia un conjunto de centros de distribución. El sistema de transporte opera mediante una flota de camiones de terceros con los cuales existe un compromiso de uso mínimo en término de distancias semanales por camión, que debe ser honrado, y considera ventanas temporales asociadas a las operaciones de carga en la planta y descarga en los centros de distribución. A diferencia del enfoque tradicional en un problema de ruteo de vehículos, donde una ruta o circuito se define por una secuencia de visitas a destinos o clientes de forma consecutivas con inicio y fin en el nodo origen, en este trabajo, un circuito se define como una secuencia ordenada de destinos que pueden ser visitados por un camión dado a lo largo del horizonte de planificación (una semana en el caso de estudio). El objetivo de asignar un circuito a un camión radica en mantener un uso eficiente y balanceado de la flota que decide utilizarse. Dado que cada visita implica la entrega de un camión completo, la misma implica un viaje redondo (ida y vuelta), razón por la cual un dado circuito está formado por múltiples viajes ida y vuelta que deben ser coordinados para cumplir con los turnos de carga y descarga. Este tipo de lógicas operativas es típica de industrias de consumo masivo que deben atender con grandes volúmenes centros de distribución o almacenes propios para mantener ciertos niveles de stock de acuerdo a sus pronósticos de venta.

Antes de resolver el problema se conocen diversos elementos del sistema logístico. Entre ellos se encuentran los destinos a visitar, la demanda semanal asociada a cada destino, expresada en términos de cargas completas de camión, la distancia desde la planta de origen hasta cada centro de distribución, la flota máxima disponible para el proceso de transporte y los turnos de trabajo de la planta principal y los destinos. Asimismo, se consideran parámetros operativos tales como las capacidades de carga y descarga por turno, las ventanas horarias para las operaciones de carga y descarga, los tiempos requeridos para dichas operaciones y los días de trabajo dentro del horizonte semanal de planificación.

Dada esta información, el objetivo del proceso de planificación consiste en determinar la asignación de camiones a los distintos circuitos de transporte a lo largo del horizonte semanal. El resultado del proceso es un plan de transporte detallado representado mediante un diagrama de Gantt (Wilson, 2003), como el ilustrado en la Figura 1. En dicho diagrama se visualiza, para cada camión, la secuencia de destinos asignados junto con los tiempos asociados a las actividades: carga, viaje de ida, espera antes de la descarga debida a las ventanas horarias de los destinos, descarga y viaje de regreso.

2.2. Estructura de decisión del problema

El proceso de planificación se apoya en una etapa de preprocesamiento en la cual se definen previamente los circuitos que pueden ser recorridos por los camiones.

En el enfoque metodológico propuesto, estos circuitos se construyen previamente a la ejecución del modelo, determinando el universo de las posibles secuencias de destinos de manera que se cumplan diversos criterios operativos. Para que un circuito sea factible



Figura 1: Diagrama de Gantt generado a partir de la solución del modelo de optimización

debe respetar distancias mínima y máxima, cantidad máxima de viajes por circuito, y jornadas laborales de los conductores. Esta estrategia permite descartar circuitos inviables desde el punto de vista operativo y simultáneamente, evita introducir la decisión de secuenciación directamente dentro del proceso de optimización, reduciendo así la complejidad del modelo.

A partir de este conjunto de circuitos, el problema consiste en determinar qué circuitos son utilizados y en qué momentos se realizan las operaciones de carga, viaje y descarga dentro del horizonte semanal de planificación para satisfacer la demanda de los destinos. Adicionalmente, las decisiones de planificación deben satisfacer diversas restricciones operativas que incluyen ventanas temporales de carga y descarga, horarios de trabajo, disponibilidad máxima de vehículos y limitaciones de capacidad carga en planta y descarga en los centros de distribución, así como la factibilidad temporal de los recorridos respecto a las jornadas laborales de los conductores.

Las decisiones del modelo pueden agruparse en dos grandes categorías: decisiones binarias y decisiones continuas. Las decisiones binarias corresponden principalmente a la asignación de circuitos a los camiones y a la determinación de los turnos para realizar cada operación de carga y descarga. Por su parte, las decisiones continuas están asociadas a la determinación precisa de los tiempos de las distintas actividades logísticas, tales como los instantes de inicio de carga, inicio y fin de los viajes y comienzo de las operaciones de descarga.

2.3. Objetivo del modelo

Como se ha mencionado, dado que la demanda de producto semanal se mide en cargas completas, los trayectos de los camiones incluyen la ida desde planta hacia los depósitos o centros de distribución y el regreso a planta para su próximo viaje. Esto hace que la distancia total a recorrer para satisfacer la demanda semanal esté determinada entre cotas y no sea un objetivo de minimización. De hecho, se pretende hacer un uso eficiente de la flota, minimizando la cantidad requerida de camiones y liberando flota no necesaria para que quienes proveen el servicio de transporte puedan usar el excedente para otros fines.

En consecuencia, el problema incorpora un criterio de equidad en la asignación del trabajo entre los transportistas, con el objetivo de lograr una distribución lo más homogénea posible de las distancias totales recorridas por la flota. Este criterio resulta consistente con el contexto operativo, en el cual existen límites sobre la duración de las jornadas laborales y la longitud de los circuitos, y donde asignaciones significativamente desbalanceadas pueden generar inconvenientes desde el punto de vista operativo y organizacional.

En particular, la función objetivo del modelo busca maximizar la distancia recorrida por el camión con menor carga de trabajo. Este criterio promueve una asignación más equitativa de las tareas entre los transportistas y contribuye a un uso más balanceado de la flota requerida.

3. Modelo de optimización base

En esta sección se describen las principales características del modelo de optimización original, así como las limitaciones que motivaron la búsqueda de técnicas más adecuadas para abordar instancias de mayor escala.

3.1. Formulación general

A continuación se presentan algunas restricciones representativas del modelo, seleccionadas por su relevancia estructural. El resto de las restricciones, vinculadas a detalles operativos y de implementación, se omiten por limitaciones de espacio.

El modelo se define sobre un conjunto de índices discretos y ordenados que representan circuitos candidatos (C), destinos (D), camiones de la flota (R), viajes dentro de cada circuito (V), días del horizonte de planificación (T) y turnos de trabajo (S). Estos índices permiten capturar tanto la estructura secuencial de los circuitos como la dimensión temporal del problema de planificación.

Las variables y parámetros del modelo reflejan directamente las decisiones introducidas en la Secciones 2.2 y 2.3. Asimismo, algunos parámetros utilizados en el modelo provienen de una etapa de preprocesamiento previa y se consideran datos de entrada del problema. A continuación se listan únicamente las variables y parámetros necesarios para interpretar las expresiones presentadas en esta sección.

- $y_{c,r}^{\text{circuito}} \in \{0, 1\}$: variable binaria que vale 1 si el circuito c es asignado al camión r .

- $y_{r,v,t,s}^{\text{carga}} \in \{0, 1\}$: variable binaria que indica si el camión r realiza una operación de carga asociada al viaje v en el turno s del día t .
- $xic_{r,v,t,s} \in \mathbb{R}$: variable continua que indica el momento exacto del inicio de la actividad de carga para un camión r en su viaje v el día t y turno de carga s , $s \in S$. Otras continuas son: xiv y xfv para el inicio y fin de viaje respectivamente; xid para el momento de la descarga, xir y xfr para el inicio y fin del retorno a la planta principal, respectivamente.
- $Z_{\text{Obj}} \in \mathbb{R}$: variable para modelar el criterio de equidad, representando la distancia recorrida por el camión con menor carga de trabajo.
- $yv_{c,d,v} \in \{0, 1\}$: parámetro binario precalculado que indica si el viaje v del circuito c es hacia el destino d .

Debe destacarse que el modelo toma decisiones de las actividades logísticas tanto de forma discreta (con variables binarias sobre turnos y días) como continuas (la hora absoluta respecto a la semana entera donde se realiza cada actividad). El modelado del tiempo es por tanto híbrido, considerando la asignación de periodos (discreto) y momentos de tiempo (continuo).

La función objetivo del modelo se define como la Ecuación 1, donde Z_{Obj} representa la distancia recorrida por el camión con menor carga de trabajo. Este valor se acota mediante la Ecuación 2. Esta asegura que la variable Z_{Obj} sea menor o igual que todas las distancias que recorren los camiones, representando al camión que menos distancia recorre.

$$\text{máx } Z_{\text{Obj}} \quad (1)$$

$$Z_{\text{Obj}} \leq \sum_{d,v} yv_{c,d,v} \cdot \text{distancia}_d \cdot 2 + M (1 - y_{c,r}^{\text{circuito}}) \quad \forall c \in C, r \leq \text{RMax} \quad (2)$$

Entre las restricciones que garantizan la coherencia entre la selección de circuitos y las operaciones logísticas, se impone que si un camión r realiza un viaje v perteneciente al circuito c , la carga asociada a dicho viaje debe efectuarse. Esto lo representa la Ecuación 3. Notar que r está acotado por el parámetro FlotaMaxima , que es la cantidad máxima de camiones disponibles para enviar.

$$yv_{c,d,v} \cdot y_{c,r}^{\text{circuito}} \leq \sum_{t \in T} \sum_{s \in S} y_{r,v,t,s}^{\text{carga}} \quad \forall c \in C, d \in D, r \leq \text{RMax}, v \in V. \quad (3)$$

Los parámetros DL y DU introducen una relajación controlada en el cumplimiento de la demanda semanal. En lugar de exigir su satisfacción exacta, se admite una variación acotada respecto de su valor nominal, lo que aporta flexibilidad al modelo y favorece la obtención de soluciones factibles bajo las restricciones operativas. Este enfoque resulta adecuado bajo el supuesto de grandes volúmenes, típico de industrias de consumo masivo.

En efecto, se requiere que la cantidad total de viajes asignados se mantenga dentro de las cotas de demanda semanal, lo cual se logra con la Ecuación 4.

$$\text{Dem}_d \cdot \text{DL} \leq \sum_{\substack{c \in C, v \in V \\ r \leq \text{RMax}}} y_{c,d,v}^v \cdot y_{c,r}^{\text{circuito}} \leq \text{Dem}_d \cdot \text{DU}, \forall d \in D \quad (4)$$

Ademas, para garantizar la secuenciación adecuada de los viajes realizados por cada camión, se impone que un viaje no pueda asignarse en un periodo dado sin que el anterior haya sido asignado a un periodo previo, como se representa en la Ecuación 5.

$$y_{r,v+1,tt,ss}^{\text{carga}} \leq \sum_{\substack{(t,s): t < tt \\ \vee (t=tt \wedge s < ss)}} y_{r,v,t,s}^{\text{carga}}, \quad \forall r \leq \text{RMax}, v < |V| \quad (5)$$

Finalmente, se introduce una restricción que considera una de las variables continuas temporales. En ella se asegura que el momento en el que inicia la carga sea posterior al inicio de la ventana correspondiente. La Ecuación 6 refleja esto.

$$y_{r,v,t,s}^{\text{carga}} \cdot (\text{IVC}_s + 24 \cdot (t - 1)) \leq \text{xic}_{r,v,t,s} \quad \forall r \leq \text{RMax}, v \in V, t \in T, s \in S \quad (6)$$

3.2. Aspectos de crecimiento del modelo y diseño de experimentos

El desempeño del modelo fue evaluado mediante una serie de experimentos incrementales, comenzando con instancias reducidas en términos de cantidad de destinos y demanda a satisfacer. Posteriormente, se incrementó progresivamente la demanda hasta alcanzar un valor máximo, tras lo cual se aumentó la cantidad de destinos considerados y se reinició el análisis con niveles de demanda más bajos. Esta estrategia permitió aislar el impacto relativo de cada factor sobre el tamaño y la complejidad del modelo.

Respecto al primer factor, cabe destacar que el incremento de la demanda tiene un impacto sobre el tamaño del problema ya que a mayores demandas, mayor es el número de camiones requeridos, lo que impacta sobre el conjunto de flota disponible. Desde un punto de vista teórico, si no se aplicara ningún tipo de filtrado operativo y se consideraran todas las secuencias ordenadas posibles de hasta n viajes seleccionados de un conjunto de m destinos, permitiendo repeticiones, la cantidad de circuitos candidatos crecería exponencialmente. En particular, el número de secuencias posibles está dada por $\sum_{k=1}^n m^k$, donde m representa la cantidad de destinos y n la cantidad máxima de viajes permitida para un circuito, cota que se da por la practica operativa. Cada término m^k corresponde al número de secuencias ordenadas de longitud k .

En la práctica, el número de circuitos generados es considerablemente menor debido a los filtros operativos aplicados durante el preprocesamiento; sin embargo, esta expresión muestra el crecimiento exponencial potencial del espacio de circuitos candidatos.

Los experimentos realizados confirman este comportamiento: al aumentar la cantidad de destinos, también crece significativamente el número de circuitos candidatos generados durante la etapa de preprocesamiento. Este incremento repercute en el tamaño del modelo y en el espacio de búsqueda que debe explorar el *solver*, afectando el desempeño computacional del procedimiento de resolución.

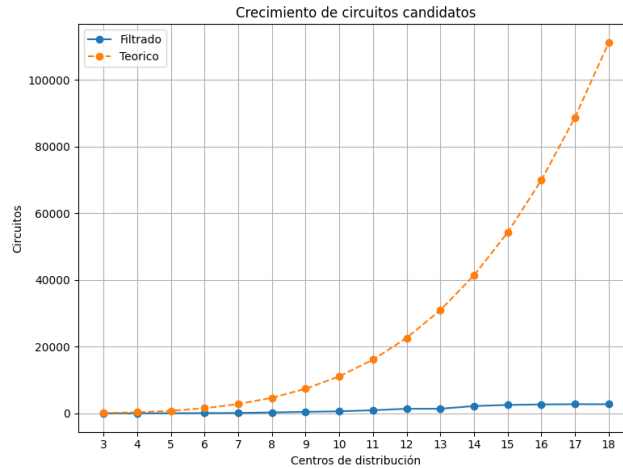


Figura 2: Crecimiento de la cantidad de circuitos en función del número de centros de distribución, comparando el caso teórico sin filtrado y el obtenido tras el preprocesamiento ($n = 4$ viajes).

La Figura 2 ilustra el crecimiento de la cantidad de circuitos en función del número de centros de distribución, tanto para la estimación teórica como para los circuitos efectivamente generados durante el preprocesamiento. En la estimación teórica se consideró una longitud máxima de circuito de cuatro viajes, en concordancia con el límite impuesto durante el preprocesamiento. Si bien la tendencia general es creciente, la relación no responde a un patrón simple, debido a que el proceso de generación de circuitos incorpora diversas restricciones operativas y temporales. En particular, el impacto de agregar un nuevo destino depende de sus características operativas, como su distancia a la planta y su posibilidad de integrarse en un mismo circuito con otros destinos sin incumplir las restricciones de distancia o de jornada laboral. Por esta razón, distintos destinos pueden generar incrementos muy diferentes en el número de circuitos factibles. Como consecuencia, pequeñas variaciones en la cantidad de destinos pueden producir incrementos abruptos en el número de circuitos generados, lo que refleja el carácter combinatorio del problema.

4. Evolución del modelo y del entorno de resolución

El modelo presentado en la sección anterior corresponde a una formulación monolítica desarrollada inicialmente en GAMS y resuelta mediante CPLEX . Posteriormente, el modelo fue migrado a una implementación basada en Python , utilizando la librería de modelado Pyomo y el solver Gurobi .

Este cambio de entorno permitió mejorar la integración con herramientas de procesamiento de datos, incrementar la flexibilidad de modelado y facilitar la experimentación con distintas estrategias de resolución. Estas mejoras sentaron las bases para el desarrollo

C.D.	Demanda	Circuitos	ZObj	Tiempo de resolución (s)	Flota	Restr.	Cont.	Disc.
3	33	15	2994	6.35	10	273748	19405	2013
3	66	15	2850	2764.60	19	547153	38809	4026
4	28	20	2850	632.67	8	263390	19657	1692
4	50	20	2994	847.53	16	467790	34945	3008
6	69	94	3000	7164.86	21	3395700	193201	6026

Tabla 1: Resultados de los experimentos computacionales

posterior de enfoques orientados a mejorar la escalabilidad del modelo.

4.1. Escritura del modelo en su versión monolítica

Para resolver el problema se desarrolló una formulación monolítica en GAMS y se la resolvió usando CPLEX. En este contexto fue posible resolver instancias de hasta seis centros de distribución. La principal limitación observada estuvo asociada al uso intensivo de memoria RAM, debido al elevado número de variables binarias y al tamaño del espacio de búsqueda resultante. En la Tabla 1 se pueden encontrar los resultados de algunos experimentos, que debido a limitaciones de espacio, están simplificados. Cabe destacar además que en todos ellos su *gap* es del 0 % y las siglas “C.D.” refiere a centros de distribución. En esta etapa inicial, el análisis se centró en identificar los cuellos de botella computacionales del modelo, sin realizar un estudio sistemático de desempeño ni un análisis comparativo detallado de tiempos de resolución.

Estas observaciones motivaron la necesidad de migrar el modelo a un entorno con mayor flexibilidad y soporte para la experimentación computacional, particularmente en lo referido a la definición dinámica de dominios y al preprocesamiento de datos, lo que condujo a su reimplementación en un entorno basado en Pyomo.

4.2. Migración de GAMS a Pyomo

La migración del modelo desde GAMS hacia Pyomo implicó la reescritura completa de la definición de conjuntos, variables y restricciones. Si bien en la mayoría de los casos este proceso consistió en una traducción directa, o requirió un esfuerzo relativamente reducido, la carga y manipulación de los datos representó inicialmente el principal desafío, dado que los mecanismos de gestión de datos difieren sustancialmente entre GAMS y Python. No obstante, el uso de bibliotecas especializadas para la manipulación de datos en Python permitió una gestión más flexible y transparente. La posibilidad de almacenar la información en estructuras explícitas facilitó su inspección, validación y preprocesamiento antes de la construcción del modelo. Esta capacidad resultó clave para detectar inconsistencias, verificar dominios factibles y mejorar la consistencia entre la información cargada y los valores asumidos por los parámetros del modelo.

Asimismo, el cambio de entorno posibilitó la adopción de un *solver* diferente, Gurobi, lo que se tradujo en una mejora significativa del desempeño computacional. Esta ganancia resultó clave para abordar instancias de mayor escala sin modificar los datos operativos ni la lógica subyacente del modelo.

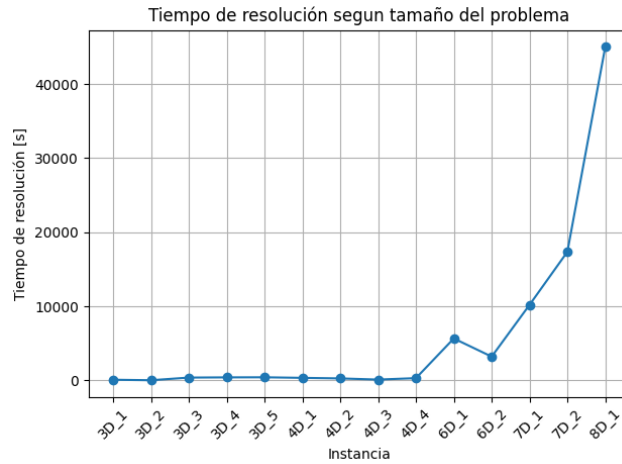


Figura 3: Tiempo de resolución en función del tamaño del problema utilizando Pyomo con Gurobi como *solver*.

Sin embargo, el uso de Gurobi también introdujo diferencias en el comportamiento del proceso de resolución. Distintos *solvers* implementan estrategias de *presolve*, generación de cortes y heurísticas de búsqueda propias, influyendo en la forma en que se explora la región factible. Este comportamiento puso en evidencia limitaciones implícitas del modelo original, motivando la incorporación de restricciones adicionales destinadas a controlar dichos efectos y a reforzar la completitud y robustez de la formulación.

4.3. Resultados del modelo representado en Pyomo

A partir de la versión del modelo escrita en Pyomo fue posible resolver instancias de hasta nueve centros de distribución, lo que representa una mejora significativa respecto al tamaño de los problemas abordables en la versión escrita en GAMS. No obstante, esta escala aún resultó insuficiente para representar escenarios operativos realistas.

En la Figura 3 se presentan los tiempos de resolución en función de la instancia considerada. Cada instancia se identifica mediante la notación XD_Y , donde X indica la cantidad de centros de distribución e Y corresponde a un identificador del nivel de incremento de demanda asociado. El tiempo requerido para resolver la instancia de mayor tamaño alcanzó los 127,039 segundos (aproximadamente 35.3 horas), registrándose además un pico de uso de memoria RAM cercano a los 100 GB.

Si bien la adopción de Gurobi permitió ampliar el tamaño de las instancias resolubles, los resultados obtenidos aún no resultaron operativamente satisfactorios, dado que la escala alcanzada se mantiene por debajo de la requerida para un escenario real. La principal limitación observada continuó siendo el elevado consumo de memoria RAM, lo que motivó la exploración de técnicas de resolución alternativas.

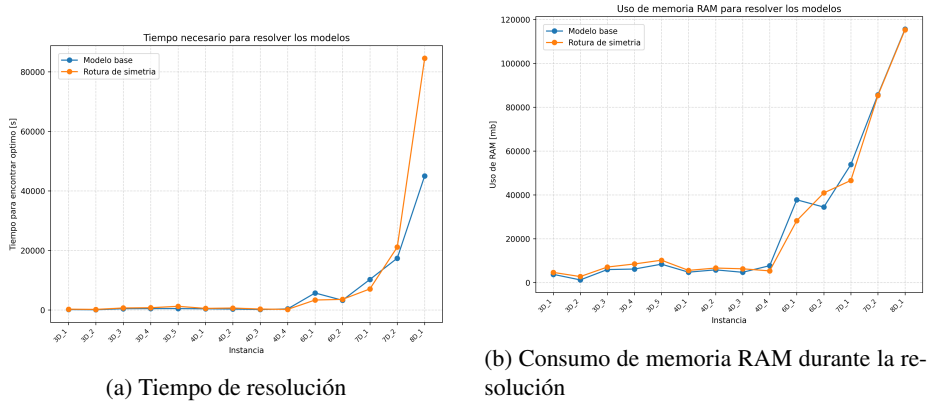
5. Técnicas alternativas de resolución

La ausencia de un modelo operativamente aceptable, junto con los problemas de escalabilidad identificados, motivó la exploración de distintas técnicas clásicas orientadas a mitigar las limitaciones observadas.

5.1. Rotura de simetrías

La formulación presenta un alto grado de simetría, ya que los camiones son indistinguibles al momento de asignar circuitos. En consecuencia, distintas permutaciones de camiones pueden producir soluciones operativamente equivalentes que el *solver* considera distintas, incrementando artificialmente el tamaño de la región factible explorada.

Para reducir estas simetrías se introdujeron restricciones de ordenamiento en la asignación de circuitos. En primer lugar, se impuso un uso contiguo de la flota, de modo que si un camión con índice k es utilizado, todos los camiones con índices menores deben encontrarse también asignados. Adicionalmente, los circuitos se ordenaron de manera decreciente según su distancia total y se restringió su asignación para que los circuitos más largos se asignen a camiones con índices inferiores. Estas restricciones eliminan soluciones que difieren únicamente en la permutación de camiones. Sin embargo, los resultados experimentales muestran que la incorporación de estas restricciones no mejora el desempeño computacional, resultando inferior a la formulación base.



de memoria asociada a la rotura de simetrías.

Finalmente, mientras que la formulación base logró resolver la instancia con nueve centros de distribución, la versión con rotura de simetrías no consiguió resolver dicha instancia ni permitió abordar configuraciones mayores. Estos resultados indican que esta técnica no mejora el desempeño ni amplía el rango de instancias resolubles en el contexto analizado.

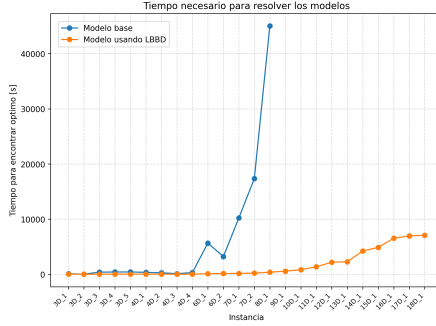
5.2. Descomposición de Benders basada en lógica

Una estrategia ampliamente utilizada para abordar problemas de optimización de gran escala es la descomposición de Benders (Taskin, 2010), que separa el modelo en un problema maestro y uno o varios subproblemas resueltos iterativamente mediante la incorporación de cortes que refinan la región factible del maestro. En este trabajo se adopta la LBBD propuesta por Hooker y Ottosson (2003), que generaliza el enfoque clásico al permitir subproblemas con variables enteras y la generación de cortes sin depender de resultados de dualidad fuerte (Wolsey y Nemhauser, 1999). Esta característica resulta adecuada para la estructura del problema considerado.

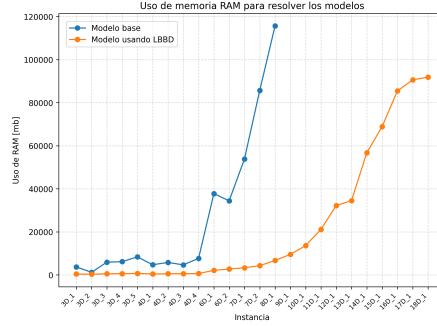
En la descomposición propuesta, el problema maestro determina la asignación de circuitos a los camiones, mientras que el subproblema verifica la factibilidad operativa de dicha asignación considerando las restricciones de carga y descarga y la programación temporal de las actividades. Cuando el subproblema resulta infactible, se agrega un corte de tipo *no-good* que excluye la solución del maestro responsable de dicha infactibilidad. Cuando el subproblema es factible, se ha encontrado la solución óptima dado que las variables incluidas en el mismo no están presentes en el master y no afecta la función objetivo. En este sentido, el procedimiento de resolución implementado corresponde a una traducción a Python del Algoritmo 2 presentado en Velásquez-Bermúdez et al. (2019), resolviendo iterativamente el maestro, evaluando la factibilidad mediante el subproblema y agregando cortes hasta alcanzar el criterio de terminación (solución factible en el subproblema).

Es importante destacar que la LBBD no modifica el conjunto factible del modelo original ni compromete la optimalidad global, sino que reorganiza el proceso de resolución mediante la incorporación iterativa de cortes que excluyen regiones inviables de la región factible explorada por el *solver*. La aplicación de esta técnica permitió obtener mejoras sustanciales respecto del modelo base. En particular, logró resolver la instancia de mayor tamaño disponible y, para la mayor instancia resuelta por ambos enfoques, redujo el tiempo de cómputo en aproximadamente 12.38 horas y el consumo de memoria en 108 GB.

Los resultados pueden observarse en la Figura 5. Se aprecia que el modelo descompuesto presenta tiempos de resolución significativamente menores y un consumo de memoria inferior al del modelo base. Asimismo, durante la etapa de instanciación del modelo (Figura 6) se observan reducciones de hasta 27,300 segundos (7.58 horas) en el tiempo de construcción y de 181 GB en memoria RAM. Cabe destacar además que se encontró el óptimo en la primer iteración del algoritmo antes mencionado y que en todas las instancias comparables, ambos enfoques alcanzaron el mismo valor óptimo de la función objetivo, por lo que las mejoras observadas corresponden exclusivamente a eficiencia computacional.

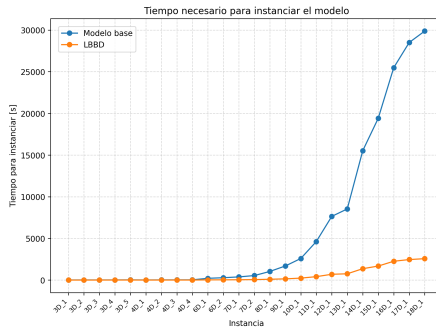


(a) Tiempo de resolución

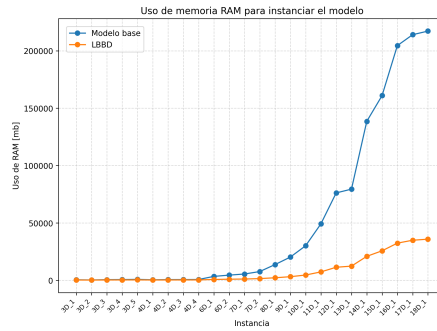


(b) Consumo de memoria RAM

Figura 5: Comparación entre el modelo base y el modelo con LBB en términos de tiempo de resolución y consumo de memoria RAM.



(a) Tiempo de instanciación



(b) Consumo de memoria RAM

Figura 6: Comparación entre el modelo base y el modelo con LBB durante la etapa de instanciación del modelo.

La aplicación de la LBB permitió obtener mejoras sustanciales respecto del modelo base, tanto en tiempo de cómputo como en consumo de memoria RAM. En particular, el enfoque descompuesto permitió resolver instancias de mayor tamaño manteniendo la optimalidad de las soluciones y reduciendo significativamente los requerimientos computacionales del proceso de resolución.

6. Conclusión

En este trabajo se estudió la escalabilidad de un modelo de optimización para la planificación del transporte de larga distancia. Tras analizar distintas estrategias de resolución, incluyendo rotura de simetrías y LBB, se observó que la última permite mejorar significativamente el desempeño computacional del modelo. En particular, esta técnica permitió reducir de manera sustancial los tiempos de cómputo y el consumo de

memoria, logrando resolver instancias de mayor tamaño que las abordables mediante la formulación monolítica.

No obstante, análisis sobre el consumo de memoria durante los experimentos computacionales indican que los requerimientos del modelo continúan creciendo de forma marcada a medida que aumenta la cantidad de centros de distribución. Este comportamiento refleja la sensibilidad estructural del problema y sugiere que la exploración de estrategias adicionales orientadas a reducir el tamaño efectivo del modelo o el espacio de búsqueda podría contribuir a mejorar aún más la escalabilidad del enfoque.

Además, la diferencia de tiempo entre los inicios de los viajes es notable. Por este motivo se propone considerar la dimensión tiempo como métrica a optimizar, adicionalmente al uso eficiente de la flota requerida. Asimismo se propone una evaluación mas sistemática de la escalabilidad, al incrementar tanto la demanda como la cantidad de destinos.

En conjunto, los resultados obtenidos muestran que la LBBD constituye una herramienta efectiva para mejorar la resolubilidad del problema considerado, manteniendo la optimalidad de las soluciones y permitiendo abordar instancias de mayor escala en un entorno computacional realista.

7. Agradecimientos

Este trabajo utilizó recursos computacionales de UNC Supercómputo (CCAD) de la Universidad Nacional de Córdoba (<https://supercomputo.unc.edu.ar>), que forman parte del Sistema Nacional de Computación de Alto Desempeño (SNCAD) de la República Argentina.

Referencias

- FADEEAC (n.d.). Estudios económicos y costos. Consultado el 22 de junio de 2026, <https://www.fadeeac.org.ar/estudios-economicos-y-costos/>.
- Hooker, J. N. y Ottosson, G. (2003). Logic-based benders decomposition. *Mathematical Programming*, 96(1):33–60.
- Taskin, Z. C. (2010). Benders decomposition. *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Malden (MA).
- Toth, P. y Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications*. SIAM, Philadelphia, PA.
- Velásquez-Bermúdez, J. M., Khakifirooz, M., y Fathi, M. (2019). *Large Scale Optimization in Supply Chains and Smart Manufacturing*. Springer.
- Wilson, J. M. (2003). Gantt charts: A centenary appreciation. *European Journal of Operational Research*, 149(2):430–437.
- Wolsey, L. A. y Nemhauser, G. L. (1999). *Integer and combinatorial optimization*. John Wiley & Sons.