

# Inteligencia Artificial

## Informe Final: Milk Collection Problem with Blending

[Belén González Baradit]

11 de enero de 2021

### Evaluación

Mejoras 1ra Entrega (10 %):	_____
Código Fuente (10 %):	_____
Representación (15 %):	_____
Descripción del algoritmo (20 %):	_____
Experimentos (10 %):	_____
Resultados (10 %):	_____
Conclusiones (20 %):	_____
Bibliografía (5 %):	_____
<b>Nota Final (100):</b>	_____

### Resumen

El presente informe presenta el estado del arte del problema de recolección de leche con mezcla (MB: Milk collection problem with Blending), cuyo problema consiste en recolectar leche cruda de diferentes productores de leche y llevarlos a la planta para su posterior proceso. Cada granja produce una de cada tres calidades posibles de leche. Los ingresos aumentan con la calidad y hay un requisito mínimo en la planta para cada calidad. Se pueden mezclar diferentes calidades de leche en los camiones, lo que reduce los ingresos, pero también los costos de transporte, lo que genera mayores ganancias. Para resolver este problema se hace una revisión exhaustiva de la literatura en torno al transporte de leche, donde se pondrá especial atención a las técnicas y consideraciones utilizadas para resolverlo.

## 1. Introducción

El transporte de la leche desde las granjas hasta las plantas de procesamiento implica importantes costos logísticos. La leche debe tratarse con cuidado debido a su naturaleza perecedera. Los productores de leche cruda se distribuyen principalmente por grandes áreas geográficas rurales, distancias que dificultan el problema. Además, diferentes granjas producen diferentes calidades de leche. Estas diferentes calidades de leche son la materia prima para la elaboración de diferentes productos y, por tanto, tienen una demanda diferente por tipo que los productores deben cumplir. En la mayoría de los casos, la recogida de leche de diferentes calidades implica el uso de diferentes camiones o camiones con compartimentos, uno para cada tipo de calidad.

Una limitante es la capacidad de cada camión y compartimiento. Un procedimiento alternativo es permitir la mezcla de leches de diferentes calidades, esta opción reduce los ingresos porque la mezcla reduce la calidad de la mejor leche, pero también reduce los costos de transporte del proceso de recolección. Por lo que el propósito de este informe es investigar los problemas que giran en torno al Milk Collection Problem (MCP), motivados por encontrar las tendencias más recurrentes en la literatura para su resolución, además de rescatar los avances más novedosos que se han desarrollado sobre el mismo, con el fin de buscar una clara tendencia a futuro.

Este artículo está organizado como sigue, la sección 2 describe el problema en cuestión, la sección 3 hace una revisión de la literatura con técnicas y heurísticas utilizadas para resolver las diferentes variantes del problema de la leche, la sección 4 contiene el modelo matemático para el problema del Milk collection problem with Blending (MB), la sección 5 contiene la representación del problema a tratado, la sección 6 describe el funcionamiento del algoritmo en forma general. La sección 7 contiene los experimentos realizados, y la sección 8 los resultados, finalmente la sección 9 se muestran las conclusiones del trabajo.

## 2. Definición del Problema

MCP es un problema recurrente en la industria lechera, en la que las empresas procesadoras de leche deben pagar un servicio de tercero de camiones cisterna para transportar la leche cruda desde los productores de leche hasta la planta procesadora. El costo de recolectar la leche cruda en la cadena de suministro de producción de leche tiene un impacto significativo en las ganancias. Los productores de leche se encuentran frecuentemente dispersos en extensas áreas rurales, a veces lejos de las plantas de procesamiento, lo que hace que el costo de transporte sea un componente relevante del costo total.

Este problema se presentó por primera vez en la literatura en el trabajo de Sankaran y Ubgade (1994) [28], el cual lo trató como una variante del VRP (Vehicle Routing Problem). Este problema consistía en minimizar el costo de la ruta escogida para vehículos motorizados, es decir, minimizando la distancia o ruta recorrida. Sin embargo, el problema es considerablemente más complejo que un VRP, pues se agrega la capacidad del camión para recolectar leche, horarios de producción, colas de espera para dejar la leche en la planta, y una flota de camiones que puede ser heterogénea o de diferentes características para recolectar la leche, entre las cuales se destacan aquellas con un solo compartimiento de igual (Dayarian et al. (2015a) [11]) o diferentes capacidades (Ruiz et al. (2020) [26]) o camiones con múltiples compartimentos, de igual (Dooley et al. (2005) [13]) o diferentes capacidades (Polat et al (2019) [23]). Los clientes pueden ser visitados solo una vez por el camión recolector (Igbaria et al. (1996) [15]) o más de una (Claassen y Hendriks (2007) [9]).

Un proceso común en este problema es que los pequeños productores se organicen en cooperativas, como en el trabajo de Prasertsri y Kilmer (2004) [27], capaces de obtener mejores condiciones comerciales con, generalmente, un solo comprador. Cada cooperativa vende la leche producida por sus miembros al comprador, o empresa, que realiza el proceso de recolección. Este arreglo es conveniente para los miembros de la cooperativa, pero plantea algunos desafíos para el comprador, que debe recolectar leche de todas las granjas de la cooperativa, aunque algunas pueden estar ubicadas lejos de la planta. Diferentes soluciones se han creado a partir de diferentes variantes del mismo problema, por ejemplo para reducir los costos del transporte de leche del camión de los productores a la planta, se han creado puntos de recolección de leche como centros de acopio para las pequeñas cooperativas o plantas productoras para que el camión evite hacer más recorridos, el problema ahora consiste en saber dónde colocar esos centros de acopio para luego minimizar el problema de enrutamiento de camiones, problema que aborda el trabajo de Muntaz et al. (2014) [19]. Otras variantes interesantes es que los vehículos tengan

características especiales para acceder a territorios donde geográficamente sean de difícil acceso (Hoff y Løkketangen (2007) [14])

Otra consideración importante es que la leche producida por diferentes granjas puede tener diferentes calidades o grados, utilizados para diferentes productos finales, Caramia y Guerriero (2010) [6]. Actualmente, las empresas abordan las diferencias en la calidad utilizando camiones separados para recolectar diferentes calidades o usando tanques con compartimentos separados para diferentes calidades. Ambas soluciones son caras, especialmente si algunas granjas producen pequeñas cantidades de leche. Un enfoque diferente, consiste en mezclar las diferentes calidades de leche siempre y cuando esto conlleve a un beneficio para la empresa, Paredes-Belmar et al. (2016) [21]. La mezcla de leche producida degrada la calidad de parte de la leche recolectada, ya que el producto mezclado se clasifica como su componente de menor calidad, lo que reduce los ingresos de la empresa. Sin embargo, los ahorros en el costo de transporte superan la reducción de los ingresos y, en última instancia, aumentan las ganancias.

La combinación de diferentes productos o calidades de productos en el mismo vehículo también se aplica a otras industrias. Bing et al. (2014) [3] resuelven un problema de recolección de residuos, en el que la recolección se realiza después de separar algunos reciclables en los puntos de recolección, o cargando diferentes clases de residuos en el mismo camión y clasificándolos en el sitio de procesamiento. Estas también son posibles aplicaciones para el enfoque de este estudio.

Finalmente, Sethanan y Pitakaso (2016) [29] afirman que la mezcla de leche cruda de diferentes centros de recolección en el mismo compartimento sería una extensión valiosa de su investigación, ya que en su trabajo utilizan mezcla, lo que podría aumentar la capacidad de su técnica para modelar problemas del mundo real.

Como es posible apreciar, existen numerosas técnicas y consideraciones para resolver este problema, sin embargo, éstas se verán más a fondo en el siguiente capítulo.

### 3. Estado del Arte

El milk collector problem with blending es una sub-rama de un problema más general conocido como milk collector problem (MCP) que está a su vez es considerado un caso especial del vehicle routing problem (VRP), pero este es mucho más complejo comparado al VRP general.

En la literatura estudiada para el problema de recolección de leche se han encontrado diversas variantes que es posible separarlas en 2 grandes grupos, aquellas que tratan el problema con solo una calidad de leche, y aquellas que consideran más de un tipo de leche. En esta última, a su vez, se pueden subdividir en aquellas que permiten que la leche se mezcle en el camión (o en la planta) y aquellas que no.

En paralelo a estas subdivisiones, y debido a que MCP es un caso especial de VRP, problemas importantes derivados de VRP también son aplicados en MCP, y por esta misma razón, las restricciones involucradas en aquellos problemas son heredadas por el milk collection problem.

#### 3.1. Una calidad de leche

Sankaran y Ubgade (1994) [28], fueron los primeros en abordar este problema como un caso especial del VRP. Diseñaron rutas de recolección para minimizar los costos de transporte, utilizando un Sistema de Soporte de Decisiones (DSS: Decision Support System) llamado CARS que utiliza heurísticas de construcción de rutas para resolver un caso de 70 centros de recolección de leche en Etah, India, obteniendo ahorros anuales de USD \$15,000.

Igbaria et al. (1996) [15], también utilizan un DSS, pero en este caso llamado FleetManager, donde abordan un problema de enrutamiento de camiones cisterna para la recolección de leche en la industria láctea de Nueva Zelanda. El objetivo de su trabajo era poder crear rutas que minimizaran los costos de entregar la leche a la fábrica, esto es en su trabajo buscando las rutas más cortas.

Butler et al. (1997) [5], propuso resolver el problema de una granja lechera en Dublín, Irlanda mediante el uso de una extensión del problema del vendedor viajero simétrico (STSP: Symmetric Travelling Salesman Problem). Una de las características más importantes de su trabajo fue que algunos productores debían ser visitados cada 2 días (u otra cantidad alterna de días, ya que hasta el momento todos los trabajos anteriores consideraban solo una visita), mientras que el resto de los productores eran visitados todos los días. Utilizaron la estrategia de branch-and-cut para resolver el problema de programación entera (IP: Integer Programming) del STSP. Su objetivo es encontrar la ruta más corta para minimizar el costo del transporte de la leche.

Branch and cut es un método de optimización combinatoria para resolver problemas de programación lineal (LP) donde algunas o todas las incógnitas están restringidas a valores enteros. Branch and cut implica ejecutar el algoritmo de branch and bound y usar planos de corte (cutting planes) para ajustar las relajaciones de programación lineal.

Basnet et al. (1999) [2], proponen un algoritmo exacto para resolver un problema de planificación y secuenciación (Scheduling and Sequencing problem) y un procedimiento heurístico para asignar camiones a rutas predefinidas en Nueva Zelanda. Su objetivo es minimizar el tiempo en el que el último camión entrega su carga a la planta. El algoritmo propuesto utiliza la estructura del algoritmo branch and bound para resolver el el IP.

Tarantilis y Kiranoudis (2001) [30], desarrollaron un algoritmo rápido y robusto para resolver el problema de distribución de leche fresca para una de las empresas más grandes de Grecia. Este problema en particular se formuló como un problema heterogéneo de generación de rutas para vehículos de flota fija (HFFVRP: Heterogeneous Fixed Fleet Vehicle Routing Problem) para el cual, debido a su alta complejidad computacional, hasta ese momento, nunca se había utilizado un algoritmo exacto para resolverlo. El algoritmo que se desarrollo para resolver este problema está basado en la aceptación de umbrales (threshold-accepting based algorithm) con el objetivo de satisfacer las necesidades de la empresa, la cual planea utilizar esta metodología repetidamente para programar su distribución de camiones y rutas varias veces a la semana. El algoritmo creado es conocido como BATA (Backtracking Adaptive Threshold Accepting), el cual ahora conocido como un algoritmo metaheurístico de búsqueda estocástica que pertenece a la clase de algoritmos de aceptación de umbral. Una consideración importante a este problema, es que no resuelve el problema de camiones cisterna de la granja a la fábrica, sino de la leche procesada, entre otros alimentos perecibles al supermercado.

Prasertsri y Kilmer (2004) [27], el objetivo de su estudio es determinar si es económicamente factible implementar un enrutamiento y planificación más eficiente de la recolección de leche de una cooperativa de leche en Florida, USA. Su problema es categorizado como un problema de vendedor viajero con ventanas de tiempo (TSPTW: Traveling Salesman Problem with Time Windows). Para resolver este problema utilizan un DSS llamado ALR (ArcLogistics Route 2.0). En su función objetivo minimiza tanto la distancia recorrida como el tiempo de espera en las ventanas de tiempo.

Butler et al. (2005) [4], analizaron cómo un DSS basado en un Sistema de Información Geográfica (GIS) permite que un planificador interactúe con algoritmos de optimización para planificar rutas de recolección de leche para encontrar rutas a un costo mínimo. El documento continúa discutiendo cómo dicho DSS puede integrarse con dispositivos de captura automática

de datos y sistemas de gestión de bases de datos para proporcionar una gestión eficaz de las operaciones de recogida de leche.

Hoff y Løkketangen (2007) [14], resuelven un caso real en Noruega, utilizando el modelo definido por Chao (2002) [8] donde se describe por primera vez el problema de camiones y remolques (TTRP: Truck and Trailer Routing Problem también es conocido como TTVRP: Truck and Trailer Vehicle Routing Problem ). En este problema, los clientes se dividen en clientes a los que se puede llegar con un vehículo completo con remolque (Truck and Trailer) y clientes a los que solo se puede llegar con un solo camión (Trunk). Una de las restricciones más importantes que consideraron fueron las condiciones geográficas de las granjas que restringían la accesibilidad de diferentes vehículos, en tales casos el camión se separaba del remolque y continuaba su camino hasta buscar la leche, luego volvía donde estaba el remolque, donde se acoplaban y continuaban la ruta juntos. Desarrollaron un algoritmo de búsqueda Tabú (TS) para resolver el problema, donde el objetivo era minimizar el costo de todas las rutas.

Claassen y Hendriks (2007) [9], estudiaron el desarrollo de un DSS piloto para recolectar leche de cabra en los Países Bajos y propusieron un enfoque basado en la investigación operativa (IO) para apoyar la recolección de leche. El problema lo trataron como un problema de generación de rutas periódicas para vehículos (PVRP: Periodic Vehicle Routing Problem) y lo resolvieron proponiendo un modelo de programación lineal entera mixta (MIP: Mixed-Integer Programming) el cual resuelven utilizando el DSS SOS1: Special Ordered Sets of type 1 (LP solver). Una de las consideraciones importantes a este problema es que los productores son visitados de acuerdo a frecuencias individuales, en lugar de recolectar la leche diariamente. Los autores además, buscan minimizar las desviaciones, excedentes o deficitarias, entre recolección y producción.

Dayarian et al. (2013) [10] y Dayarian et al. (2015a)[11], resuelven un problema de recolección y distribución de leche en múltiples períodos, considerando variaciones estacionales significativas de producción en un horizonte de tiempo táctico. Los autores minimizan las desviaciones, excedentes o deficitarias, entre recolección y producción.

Dayarian et al. (2013) [10] considera el problema como una clase particular de problemas de generación de rutas de vehículos de períodos múltiples (MPVRP: Multi-period Vehicle Routing Problem) y lo resuelven desarrollando el algoritmo ALNS (Adaptive Large-Neighborhood Search) que utiliza heurísticas basadas en VRP.

Dayarian et al. (2015a) [11], presentan un problema de enrutamiento con múltiples depósitos, una flota heterogénea y ventanas de tiempo, motivados por un problema real de recolección de leche en Canadá. Este problema se considera como una planificación táctica para una clase de problemas de generación de rutas de vehículos de períodos múltiples (MPVRP: Multi-Period Vehicle Routing Problems). El objetivo es minimizar el costo de transporte y satisfacer las demandas de las plantas. Para resolver el problema usan una formulación de programación de enteros (IP) mediante la generación de columnas y un algoritmo de branch-and-price.

Dayarian et al. (2015b) [12], consideran el problema como una variante del problemas de generación de rutas de vehículos de atributos múltiples (MAVRP: multi-attribute vehicle routing problems). Este problema consiste en la planificación de rutas para una flota heterogénea de vehículos que parten de diferentes depósitos. Los vehículos deben visitar un conjunto de productores en ventanas de tiempo específicas, y el producto recolectado luego se entrega a las plantas de procesamiento. Finalmente, los vehículos regresan a sus depósitos de origen. El modelo más similar en la literatura es el problema de enrutamiento de vehículos heterogéneos de múltiples depósitos con ventanas de tiempo (MDHVRPTW: multi-depot heterogeneous vehicle routing problem with time windows). No fue posible resolver este problema como un MIP ya que era muy grande, por lo que lo resuelven modelándolo como un set covering y utilizando

el método de generación de columnas. Una de las grandes contribuciones que hicieron, es que, aunque la leche debe recolectarse todos los días, las variaciones en la producción diaria debido al clima, enfermedades del ganado, etc. son relativamente insignificantes. Por lo tanto, las rutas de recolección no tienen que ser reprogramadas diariamente.

Mumtaz et al. (2014) [19], presentan por primera vez en la literatura del MCP asignar centros de acopio de leche a las granjas, para que los productores depositen la leche en esos centros de acopio para así disminuir el costo del transporte del camión cisterna, sin tener que pasar por todos los productores, sino a los centros de acopio, los cuales se colocan estratégicamente. Por lo tanto, primero resuelven el problema de ubicación de los centros de acopio (MCNLRP: Milk Collection Network Location Routing Problem) y luego las rutas que deberán seguir los camiones (MCNVFMP: Milk Collection Network Vehicle Fleet Mix Problem). El modelo matemático planteado minimiza el número de depósitos, el número de vehículos y las distancias recorridas por los vehículos. Finalmente resolvieron el problema con el solver CPLEX.

Pasha et al. (2014) [22], consideraron el problema como un TTVRP para un problema MCP de una fábrica de leche en Noruega. utilizando un enfoque de solución híbrida que incluye la técnica de fusión y división (MST: Merging and Splitting Technique) en el algoritmo de búsqueda Tabu (TS). El método propuesto puede encontrar una solución rentable mediante el uso de agrupaciones geográficas naturales y la elección de tipos de vehículos que se adapten mejor a la producción de las granjas y la demanda de las plantas lecheras. La nueva solución sugerida produce una combinación de vehículos y una frecuencia de recolección más adecuadas, una mejor utilización de los vehículos y menores costos variables que la estructura de rutas que poseía la empresa.

Vásquez y Valencia (2014) [32], el artículo presenta la utilización de un método metaheurístico algoritmos genéticos (genetic algorithms), para la evaluación de un modelo de rutas de recolección de leche cruda. Se implementó un modelo a partir de datos reales, recolectados por medio de trabajo de campo, siguiendo el método del problema del vendedor viajero (TSP), usando el toolbox de algoritmos genéticos de Matlab® (Matlab® genetic algorithm).

Masson et al. (2015) [17], estudian un problema anual de transporte de productos lácteos, inspirado en un problema de recolección de leche canadiense. Generalizan el problema propuesto por Lahrichi et al. (2013) [16] conocido como Dairy Transportation Problem (DTP), que considera variaciones en función de la demanda diaria, mejorando el enfoque produciendo un ahorro de hasta el 4% en el costo, y presentando además, un nuevo problema de asignación y enrutamiento conocido como ADTP, que a diferencia de DTP, este utiliza información semanal durante el diseño anual de rutas, de hecho, durante cada semana del año, cada planta específica su demanda de leche. La cantidad total de leche que recibe al final de la semana debe estar cerca de la cantidad solicitada (se especifica una tolerancia para cada planta). Este cambio conduce a ahorros adicionales. En la primera parte de su trabajo, al resolver el DTP utiliza el algoritmo ALNS que supera al algoritmo GUTS propuesto anteriormente en este problema. Para la segunda etapa, para resolver el ADTP, se necesita conocer cómo agrupar las semanas, para ello utilizan un algoritmo de agrupamiento (k-means). El objetivo final es minimizar el costo de las rutas.

Caria et al. (2018) [7], Los autores desarrollaron una herramienta de apoyo a la toma de decisiones para recolectar leche de oveja en Italia. Mejoran las rutas actuales (en términos de costes y emisiones de CO<sub>2</sub>) utilizando la metaheurística de colonia de hormigas (ACO: Ant Colony Optimization). Cuyo modelo se basó en el problema del vendedor viajero (TSP). Los resultados se muestran en un sistema de posicionamiento geográfico (GIS).

Montero et al. (2019) [18], abordó un caso del mundo real para una empresa de procesamiento

de leche ubicada en el sur de Chile y consideró este problema como un problema de enrutamiento de vehículos recolectores de premios (PCVRP). El objetivo es encontrar rutas de recolección eficientes, satisfaciendo la cantidad mínima requerida y minimizando la sobre-demanda y los costos de transporte. El problema se formuló como un modelo de programación de enteros (IP) lineales para representar todos los requisitos del problema y resolver casos de prueba simples usando CPLEX. Para resolver esta versión del problema de recogida de premios, propusieron un modelo de programación de enteros para resolver instancias más simples y una metaheurística GRASP para resolver instancias más complejas en tiempo reducido. La selección de agricultores genera dos tipos de costos: transporte y sobre-demanda de leche. Los costos totales se basan en las distancias recorridas y el volumen de leche cargado en cada visita. También este problema considera a los pequeños productores que se pueden asociar en una cooperativa lechera o vender su producción a varias plantas. Se estudió el desempeño de estas propuestas sobre algunos posibles escenarios de expansión de la planta de procesamiento. Para ello, aumentaron tanto la demanda de leche recolectada como la flota de camiones, utilizando los métodos anteriormente mencionados para su resolución.

Polat y Topaloglu (2019) [25], propusieron un modelo matemático difuso para satisfacer la naturaleza probabilística del problema de recolección de leche. Diseñaron el modelo para recolectar la leche de los proveedores dividiéndola con vehículos multicompartimentales. Sin embargo, diseñaron este modelo sin considerar diferentes tipos de leches. El modelo tiene como objetivo minimizar el tiempo de duración de recolección de leche a los camiones cisterna, considerando el volumen variable de la producción de leche.

Ruiz et al. (2020) [26], resuelven un problema de recolección de leche en Galeras (Sucre), Colombia, modelando el problema como un problema de generación de rutas para vehículos con recogida y entrega simultáneas (VRPSPD: Vehicle Routing Problem with Simultaneous Pickup and Delivery). Su problema constaba de 17 granjas productoras de leche y 2 empresas procesadoras de leche. 3 tipos de vehículos con capacidades de carga de 1200, 1100 y 1000 kg. Se resolvió con el solver CPLEX, utilizando instancias de iniciales creadas utilizando GAMS.

El General Algebraic Modeling System (GAMS es un software de alto nivel para el modelado de sistema para la optimización matemática. GAMS está diseñado para modelar y resolver problemas lineales, no lineales y optimización entera mixta.

## 3.2. Más de una calidad de leche

### 3.2.1. Calidades diferentes de leche que no permiten mezcla

Al ser un problema donde no se permiten mezclas de calidades diferentes de leche, los camiones cisterna vienen con diferentes compartimentos para permitir conservar la calidad de la leche y generar productos lácteos de mayor calidad.

Dooley et al. (2005) [13], en una aplicación de Nueva Zelanda, clasifica la leche en dos calidades. Cada calidad de leche se recolecta por separado y se minimiza el costo de transporte. Se utilizó un software de algoritmo genético (Genetic algorithm software) para buscar el orden de los camiones de recolección de leche en cada granja. El orden de las paradas para recoger leche utilizadas para inicializar el software de algoritmo genético se derivaron utilizando una heurística codiciosa (greedy heuristic). Las paradas se clasificaron visitando la parada más cercana a la fábrica, luego pasando a la siguiente parada más cercana y así sucesivamente hasta que se hayan visitado todas las paradas. Se generaron seis órdenes de clasificación iniciales, ordenando paradas por tipo de leche dentro del orden de la granja y, granja dentro del orden del tipo de leche, tanto en sentido horario como antihorario.

Caramia y Guerriero (2010) [6], en su trabajo encuentra las mejores rutas de camiones para

recolectar tres calidades posibles de leche, sin mezclar. En su problema, los camiones tienen compartimentos que contienen cada calidad de leche. No se pueden combinar diferentes calidades. Los camiones tienen remolques, por lo que su problema se clasificaría como un TTPR. Al realizar el recorrido, cuando la locación geográfica para acceder a la granja es de difícil acceso el remolque se desprende del camión, y el camión sigue adelante para recoger la leche, luego este se devuelve al punto donde dejó el remolque para acoplarse a él y continuar con el viaje. Su trabajo lo resolvieron en 2 etapas: Primero resuelven el problema como un problema de asignación de camiones cisterna (tanker assignment problem) aplicando una heurística de búsqueda local, que primero asigna a los productores a los compartimentos de los camiones para minimizar el número de camiones. Luego, resuelven el problema de las rutas como un VRP minimizando la distancia del viaje. En su método, consideraron limitaciones específicas al problema, lo que produjo que se obtuvieran soluciones más prácticas. Sin embargo, al resolver el problema de manera secuencial la calidad de la solución se reduce.

Lahrichi et al. (2013) [16], abordaron un caso real en Canadá, en el que camiones con dos compartimentos comienzan su ruta en un solo depósito de camiones en múltiples períodos de tiempo en el cual recolectan tres calidades diferentes de leche. La leche se transporta a un conjunto de plantas y los camiones deben volver de regreso a los depósitos, con un costo de viaje mínimo. Los autores resuelven el problema utilizando una heurística de búsqueda tabú (TS), modelando el problema como un DTP y utilizando el algoritmo GUTS.

Amiama et al (2015) [1], desarrollaron un sistema de apoyo a la decisión espacial (SDSS) para resolver el problema de recolección de leche en dos etapas. Primero, el sistema heurístico produce una solución, y luego el sistema gráfico mejora la solución al permitir ciertos cambios en la ruta. SDSS es una herramienta útil para que el planificador de rutas vea los resultados de diferentes escenarios. El problema de este algoritmo es que no contenía herramientas para optimizar las rutas.

Sethanan y Pitakaso (2016) [29], los autores consideran este problema como un problema de enrutamiento de vehículos (VRP) con servicios de recogida y entrega (PDP). Para resolver este problema desarrollaron algoritmos DE modificados para determinar las rutas de recolección de leche cruda, con el objetivo de minimizar los costos totales considerando los costos de viaje y los costos de limpieza y desinfección de los tanques. El artículo sugiere, como trabajo futuro, la mezcla de leche de diferentes puntos de recogida en un mismo compartimento.

La Evolución Diferencial (ED) es un método de optimización perteneciente a la categoría de computación evolutiva, aplicado en la resolución de problemas complejos.

Polat et al (2019) [25], El problema presentado es modelado como un multi-compartment vehicle routing problem with split deliveries (MC-VRP-SD). Presentan un modelo matemático integrado que incluye desigualdades nuevas y conocidas para el problema de recolección de leche. Se llevó a cabo un caso de estudio hipotético basado en datos de la vida real que tiene 8 granjas o centros de recolección, 1 centro de procesamiento de leche y 3 tipos de calidades de leche. Se ha resuelto este problema utilizando el solver CPLEX. Consideraciones importantes en este problema, es que se consideran 4 camiones con tanques con características especiales, uno de ellos solo tiene un compartimento, 2 de ellos tienen 3 compartimentos, y el último tiene 2 compartimientos.

Chokanat et al. (2019) [24], consideraron el problema como un RTMP (raw milk transportation problem), básicamente es lo mismo que el MCP, pero con distinto nombre. Su problema considera camiones cisterna compuestos por tres compartimentos. El objetivo de esta investigación es minimizar el costo total de transporte y los costos de limpieza de camiones y tanques. El costo de transporte depende directamente del uso de combustible. El consumo de combustible



ocurre durante el transporte de la leche y durante los tiempos de espera cuando el camión llega a la fábrica y no puede transferir la leche cruda al tanque de producción. Para resolver el problema propuesto se presentó el algoritmo de evolución diferencial modificado (MDE).

### 3.2.2. Calidades diferentes de leche que permiten mezcla

Ahora está permitido que las diferentes calidades de leche se puedan mezclar en el camión siempre y cuando esto genere un beneficio a la empresa. El problema de la mezcla es que si yo tengo diferentes calidades de leche A, B, y C, donde la calidad de A es superior a la de B, y B es superior a la de C, al mezclarse la leche en los camiones, la mezcla realizada queda con la calidad más baja de la leche mezclada, es decir, si mezclo leche tipo A y B, la calidad de la leche resultante será de tipo B. La calidad de la leche baja, pero esto disminuye el costo del transporte.

Paredes-Belmar et al. (2016) [21], presentó un problema de recolección de leche para una instancia real en Chile con tres calidades de leche, lo más importante de este estudio es que se presenta por primera vez en la literatura el problema de recolección de leche con mezcla (MB: Milk Collection Problem with Blending). La lechería permite mezclarlos en el mismo camión siempre y cuando se tenga un impacto positivo en los beneficios de la empresa. El artículo analiza los beneficios de la combinación y reporta aumentos en las ganancias. Los resultados se comparan con el procedimiento de recolección actual de la lechería y con la recolección segregada por calidad (sin mezcla), lo que demuestra importantes mejoras. Este documento entrega grandes contribuciones al problema. Primero, para cada camión de una flota heterogénea, la solución MB indica qué agricultores debe visitar cada camión y la ruta que debe seguir para entregar toda la leche producida a la planta. Esto también especifica si es más conveniente realizar la mezcla en los camiones o en la planta. El objetivo es maximizar las ganancias de la empresa. En segundo lugar, se propone una formulación de enteros mixtos para el problema (MIP: Mixed Integer Programming), así como un algoritmo de ramificación y corte (branch-and-cut), utilizando un nuevo corte y cortes conocidos para resolver instancias de tamaño mediano de manera óptima. En tercer lugar, se diseña un procedimiento heurístico para resolver grandes instancias. Esta heurística divide un conjunto de granjas en grupos, cada uno con un número menor de granjas. Resolver el problema por agrupamiento no es trivial, ya que hay cuotas de leche que deben cumplirse y una flota de camiones determinada; por lo tanto, los camiones y las cuotas de leche deben asignarse a los clusters. Se propone una formulación de programación matemática que i) asigna los requerimientos de leche a cada cluster, de manera de asegurar el cumplimiento de los requerimientos mínimos globales para cada calidad de leche en la planta, y ii) encuentra la mejor asignación de camiones para cada cluster, de acuerdo con la flota disponible. Finalmente, el problema de recolección se resuelve para cada clúster usando el algoritmo de ramificación y corte (branch-and-cut). Se resuelven instancias de prueba de hasta 100 nodos y se resuelve una instancia real que incluye 500 granjas.

Paredes-Belmar et al (2017) [20], se basa en el trabajo de Paredes-Belmar et al (2016) en el que además de considerar la mezcla de calidades de la leche, aborda el número de punto de recolección y las decisiones de ubicación. La formulación resultante combina así un problema de enrutamiento con un problema de asignación y un problema de ubicación, y en este sentido tanto el modelo como el método de solución para instancias de diferentes tamaños son bastante novedosos. La primera contribución de este paper es introducir el problema de recolección de leche con mezcla y puntos de recolección (MBCP: milk collection problem with blending and collection points), el cual resuelve modelando el problema como un mixed integer linear programming y se utiliza el algoritmo branch-and-cut para resolver pequeñas instancias de hasta 40 nodos (granjas o puntos de recolección). La segunda contribución que entrega este paper es el desarrollo de un procedimiento de tres etapas que resuelve el MBCP para instancias grandes, el cual se aplica al caso real de una empresa láctea chilena que recolecta leche de 500 produc-

tores con una flota de 80 camiones y 112 puntos de recolección candidatos. La primera de las tres etapas, resuelve de manera óptima un problema de cobertura que asigna a los pequeños productores a los puntos de recolección. La segunda etapa genera rutas factibles utilizando la metaheurística de la colonia de hormigas. Finalmente, la tercera etapa elige las mejores rutas de las que genera la metaheurística para visitas directas a fincas y puntos de acopio. El objetivo de la formulación propuesta es maximizar las ganancias de la operación de recolección de leche con base en: (i) los ingresos obtenidos de la leche entregada a la planta, (ii) el costo de transporte en camión, (iii) el costo del punto de recolección (es decir, la operación más el costo de instalación) y (iv) costo de acceso al punto de recolección de las granjas.

Villagran et al (2020) [31], el problema que resuelven puede considerarse como un problema de generación de rutas de vehículos multiproducto con la mezcla. En esta versión se pueden mezclar diferentes productos generando una importante reducción en los costos de viaje pero un menor deterioro de la calidad de la leche que va a la planta. Para resolver este problema, se usó un enfoque de búsqueda local iterada (local search approach). Este enfoque trabaja con soluciones inviables que se penalizan en la función de evaluación. Las penalizaciones que se introducen en la función de evaluación son: Ponderar cada litro de leche insatisfecho para las cuotas de planta. La segunda sanción pondera cada litro de leche que excede las capacidades de los camiones. Estas penalizaciones se incorporan en la función de evaluación para permitir que el algoritmo busque áreas inviables del espacio de búsqueda. El modelo se probó en casos conocidos de la literatura además de un caso real aplicado en el sur de Chile con 500 nodos. El enfoque de búsqueda local resultó ser eficiente en problemas pequeños reduciendo el tiempo de su solución, además de encontrar soluciones de alta calidad.

Al hacer una revisión exhaustiva de la literatura del problema es fácil notar que ciertos métodos, heurísticas y solver tienden a repetirse en diversas variantes del problema, tales modelar el problema como un IP o MIP para problemas pequeños del MCP como los métodos y heurística Tabu Search, Branch and cut, la utilización y/o desarrollo de un DSS y el uso del solver CPLEX como herramienta de resolución favorita en varios trabajos.

Pese a que los métodos de resolución son repetidos para cierto tipo de problemas, la diferencia principal en mejora de resultados se lleva a cabo en cómo se plantea el problema y las consideraciones especiales y específicas que se tiene de cada problema que se intenta resolver. Un ejemplo claro de ello es el haber incorporado las mezclas de leche al problema original (Paredes-Belmar et al. (2016) [21] y trabajos posteriores de MB) para reducir los costos en el transporte de la leche. Para su resolución no utilizan métodos nuevos, ya que modelan el problema como un MIP para problemas pequeños, lo cual ya se vio en Claassen y Hendriks (2007) [9] o la utilización de la técnica branch-and-cut también utilizada por Butler et al. (1997) [5], sino que el solo hecho de considerar relajar ciertas restricciones al problema puede brindar nuevas y mejores soluciones al mismo. Y es así, como mejoran su mismo trabajo en Paredes-Belmar et al (2017)[20], esta vez incorporando puntos estratégicos en los que implementar centros de acopio de leche, lo se vio en el trabajo de Mumtaz et al. (2014)[19], pero el solo hecho de permitir mezclas de leche, mejora significativamente reduciendo los costos del transporte de la misma.

## 4. Modelo Matemático

### 4.1. Parámetros

- $A$ : Conjunto de arcos que representan caminos entre granjas de leche.
- $A^0$ : Conjunto de arcos que representan caminos entre planta y productores de leche
- $N = \{0, ..., n\}$  granjas de leche.

- $N^0$ : Conjunto de granjas y la planta.
- $K$ : Conjunto de camiones.
- $T$ : Conjunto de calidades de leche.
- $N^t$  : Conjunto de granjas de leche de calidad  $t \in T$ .
- $D^t$  : Resultado de la mezcla de leche de calidad  $r$  con leche de calidad  $t$ .
- $IT$  : Conjunto de pares ordenados  $(i, t)$  de granjas  $i$  y leche de calidad  $t$ , donde cada granja solo produce una calidad de leche.
- $Q^k$  : Capacidad de cada camión  $k$ .
- $q_i^t$  : Cantidad de leche  $t$  producida por la granja  $i$ .
- $c_{ij}^k$  : Costo de viaje de cada camión  $k$  sobre el arco  $(i, j) \in A \cup A^0$ .
- $\alpha^t$  : Ingreso por unidad de leche de calidad  $t$ .
- $P^t$  : Requerimientos de leche de calidad  $t$  de la planta.
- $K_i = \{k \in K : t \in T, q_i^t \leq Q^k \forall t\}$  Conjunto de camiones que pueden visitar la granja  $i$ .
- $K_{ij} = \{k \in K : q_i^t + q_j^t \leq Q^k \forall t\}$  Conjunto de camiones que puede viajar de la granja  $i$  a la granja  $j$  recolectando la leche que ambas granjas producen sin exceder la capacidad del camión, para cada arco  $(i, j) \in A \cup A^0$ .
- $0_k$  : Nodo en el que el camión  $k$  inicia la ruta.
- $AK = \{(i, j, k) : (i, j) \in A \cup A^0, k \in K_{ij}\}$  Conjunto de elementos que contiene un arco que va de la granja  $i$  a la granja  $j$  y un camión  $k$  que pertenece a  $K_{ij}$ .

## 4.2. Variables de decisión

Variables de tipo binaria

$$\begin{aligned}
 x_{ij}^k &= \begin{cases} 1, & \text{Si el camión } k \text{ viaja directamente del nodo } i \\ 0, & \text{En caso contrario.} \end{cases} \\
 y_i^{kt} &= \begin{cases} 1, & \text{Si el camión } k \text{ recoge la leche de calidad } t \text{ de la granja } i \\ 0, & \text{En caso contrario.} \end{cases} \\
 z^{kt} &= \begin{cases} 1, & \text{Si el camión } k \text{ entrega leche de calidad } t \text{ a la planta} \\ 0, & \text{En caso contrario.} \end{cases}
 \end{aligned}$$

### Variables de control de calidad de la leche entregada

- $w^{kt}$  = Volumen de leche de calidad  $t$  que el camión  $k$  entrega a la planta.
- $v^{tr}$  = Volumen de leche de calidad  $t$  entregada a la planta, mezclada para su uso como leche de calidad  $r$ .

### 4.3. Restricciones

- Cada camión no puede recolectar más leche que su capacidad máxima.

$$\sum_{t \in T} \sum_{i \in N: (i,j) \in IT} q^t y_i^{kt} \leq Q^k, \quad \forall k \in K$$

- La recolección de la leche de cada granja debe ser realizada por exactamente un camión. Esto implica que se debe recolectar la leche de todas las granjas y que cada granja no puede ser visitada más de una vez.

$$\sum_{k \in K_i} y^{kt} = 1, \quad \forall i \in N, t \in T : (i,j) \in IT$$

- Cada camión debe tener como máximo una ruta, que comienza desde la planta.

$$\sum_{j: (0_k, j, k) \in AK} x_{0_k j}^k \leq 1 \quad \forall k \in K$$

- Control de flujo para el orden de las visitas de los nodos por parte de cada camión.

$$\sum_{i: (i,j,k) \in AK} x_{ij}^k = \sum_{h: (j,h,k) \in AK} x_{jh}^k \quad \forall k \in K, j \in N_0$$

- Cada camión que visita cada granja debe detenerse y recoger su leche.

$$\sum_{p: (p,j,k) \in AK} x_{pi}^k = y_i^{kt} \quad \forall k \in K, i \in N, t \in T : (i,t) \in IT$$

#### Restricciones de mezcla de leche

- Tipo de leche según las granjas que ha visitado.

$$z^{kt} \leq 1 - \sum_{\substack{r \in D^t: r \neq t \\ (i,r) \in IT}} y_i^{kr} \quad \forall k \in K, i \in N, t \in T$$

- Cada camión solo entrega un tipo de leche a la planta.

$$\sum_{t \in T} z^{kt} \leq 1 \quad \forall k \in K$$

- La cantidad de leche entregada de cada tipo por cada camión a la planta no supera su capacidad.

$$w^{kt} \leq z^{kt} Q^k \quad \forall k \in K, t \in T$$

- La cantidad recolectada de cada tipo de leche por cada camión, considerando las granjas que ha visitado, no supera su capacidad.

$$w^{kt} \leq \sum_{r: i \in D^r} \sum_{h \in N^r} q_h^r y_h^{kr} \quad \forall k \in K, t \in T$$

- Cada camión debe llevarse toda la leche producida por cada granja a la planta.

$$\sum_{k \in K} \sum_{t \in T} w^{kt} = \sum_{(i,t) \in IT} q_i^t$$

- Se debe equilibrar la cantidad de leche de cada calidad que llega a la planta y la cantidad de leche de cada calidad restante después de la mezcla en la planta.

$$\sum_{r \in D^t} v^{tr} = \sum_{k \in K} w^{kt} \quad \forall t \in T$$

- Se deben satisfacer las cuotas de leche de cada tipo.

$$\sum_{t \in T} v^{tr} \geq P^r \quad \forall r \in D^t$$

- Se debe evitar la mezcla de leche prohibidas

$$y_i^{kt} + y_i^{kr} \leq 1 \quad \forall (t, r) \in PM; (i, t), (j, t) \in IT$$

- Se debe evitar la aparición de sub-ciclos en las rutas de cada camión.

$$\sum_{i \in S} \sum_{j \in S} x_{ij}^k \leq |S| - 1 \quad \forall S \subseteq N, k \in K$$

#### 4.4. Naturaleza de la variables

- Variables asociadas a los tipos de leche recolectados y mezclados en ruta.

$$y_i^{kt}, z^{kt} \in \{0, 1\} \quad \forall i \in N, k \in K, t \in T : (i, t) \in IT$$

- Variable que controla la ruta del camión

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j, k) \in AK$$

- Vómenes de leche entregados y mezclados en la planta no negativos.

$$w^{kt}, v^{tr} \geq 0 \quad \forall k \in K; t, r \in T, r \in D^t$$

#### 4.5. Función objetivo

La función objetivo corresponde a la maximización del beneficio. Se considera la diferencia entre la ganancia producida por la leche recolectada de acuerdo a tipo y los costos de transporte de las rutas construidas.

$$\text{MAX} \sum_{t \in T} \sum_{r \in T} \alpha^r v^{tr} - \sum_{(i, j, k) \in AK} c_{ij}^k x_{ij}^k \quad (1)$$

### 5. Representación

Para manejar el problema la representación utilizada es un struct de los diferentes componentes que yacen en el problema, es decir, una estructura para los camiones, para la planta y las granjas en conjunto con la planta, y de esta forma tener acceso a ellos durante su ejecución.

La estructura de la planta es simple: posee un vector para indicar la cantidad de leche que se requiere de cada calidad, es decir, el valor que esté almacenada en la posición 0 del vector corresponderá a la cantidad que se requiere de leche A, la posición 1 del vector indicará la cantidad

que se requiere de leche de tipo B, y finalmente la posición 2 indica la cantidad que se requiere de la calidad de tipo leche C. Ya que al estudiar los datos de todos los archivos, todos poseían los mismos tipos de leche. Además, posee un vector para el valor que se le da a la calidad, esto se almacena en un vector double ya que el tipo de dato que se maneja es un flotante, y posee el mismo orden que el mencionado anteriormente.

La estructura de la granja es en realidad la que posee los datos tanto de las posiciones de las granjas como de la planta. Los siguientes componentes se almacenan en vectores independientes:

- Los ID de las granjas y la planta.
- Las coordenadas x de las granjas y la planta.
- Las coordenadas y de las granjas y la planta.
- Un vector string para las calidades de cada granja (para la planta se almacena un guión).
- Las cantidades de leche que produce cada granja (se almacena como un 0 para la planta, ya que no produce leche, solo la procesa).

Como cada componente se guarda por separado en un vector, la misma posición que comparten los vectores corresponde al mismo punto de la granja, es decir, la posición 3 de los vectores corresponde al ID 4 de la granja, a la coordenada X e Y para la granja 4, la calidad que produce y la cantidad se encuentran exactamente en la misma posición, esa característica nos será muy útil.

Además en esta misma estructura se guardan en un vector todas las posiciones de las granjas que producen leche de tipo A, en otro vector las que producen leche tipo B y por último las que producen leche tipo B. Asimismo 3 vectores donde se guarda la cantidad que se producen las granjas de cada tipo (en el mismo orden que el vector de las posiciones), y finalmente una variable que almacena el total de las cantidades de la leche de cada tipo.

La estructura de los camiones tiene parámetros como el nombre del camión (camión 1, camión 2, camión 3) como también la capacidad de cada camión. Además posee variables tales como:

- un vector de la ruta actual, el cual almacena las posiciones del vector de las granjas a visitar.
- un vector de los ID de las rutas que cursa actualmente, esto es ahora contado la salida desde la planta y la entrada a la misma al final del recorrido.
- un vector de strings que almacena la calidad de la leche de cada granja de donde la recolecta.
- un vector que almacena la cantidad de leche que recolectó de cada granja.
- una variable que almacena la calidad total de leche recolectada de la ruta.
- una variable que almacena la cantidad total de leche recolectada de la ruta.
- una variable que almacena el valor de la función de evaluación de la ruta actual escogida.

Además, como en el código se implementa los re-start es necesario almacenar la mejor solución de todos ellos, para ello se utilizan las variables rutaBestSol0, rutaBestSol1, rutaBestSol2 , para almacenar la mejor ruta encontrada, es decir, las posiciones de los vectores granja a visitar de la leche A, B y C respectivamente; y feBestSol para almacenar las funciones de evaluación de las mejores rutas.

La estructura escogida, si bien es cierto se ve algo engorrosa, para manejar el problema es altamente eficiente, ya que gracias a ella es mucho más sencillo saber donde encontrar cada componente con el cual se desea trabajar. Además, el utilizar vectores facilita el uso sencillo de la memoria dinámica, ya que no es necesario pedir de antemano cuánto es lo que se va a necesitar, a diferencia de las listas por ejemplo, por lo que su estructura es sencilla para manejar datos de largos diferentes. De la misma forma es fácil de modificar y acceder a datos específicos dentro de ella, incluso de eliminarlos, sin tener que preocuparse por el uso indebido de la memoria.

La solución será el vector de mejor ruta de cada uno de los camiones creados, pero para trabajar en la solución se crea un vector de vectores que tendrán las estructuras de los nodos creados de los camiones.

La función de evaluación matemáticamente se calcula como:

$fe = \text{calidad del tipo de leche} \times \text{cantidad de leche recolectada} - \text{costo} \times \text{ruta tomada}.$

El costo de la ruta se calcular como: Se transforma la ruta incluyendo la planta al inicio y al final. Se recorre la ruta calculando la distancia entre los dos puntos consecutivos de la misma usando la fórmula de la distancia:  $\sqrt{(x2 - x1)^2 + (y2 - y1)^2}$  Y se suman todas las distancias generadas.

Luego se calcula la calidad de la leche de la ruta, y se multiplica por su cantidad total. Luego simplemente se saca la resta.

Por lo que la representación escogida es muy útil al momento de resolver el problema.

## 6. Descripción del algoritmo

La técnica asignada para resolver este problema es Hill Climbing con mejor mejora. Hill Climbing necesita una representación, una solución inicial y un movimiento. La representación ya fue explicada en la sección anterior.

### Solución inicial

La solución inicial se implementa en la función `solIni()`;

Para implementar la solución inicial del problema de mezclas, se generaron 3 rutas, aquellas que abarcan sólo las granjas que producían la leche tipo A, aquellas que producían solo la leche de tipo B y luego las que producían solo la leche de tipo C. Siempre trataremos el problema con ese orden de las leches, A, B y C.

Y a esas rutas generadas le vamos a asignar un camión al azar, por ejemplo, a la ruta de la leche A le asigno el camión 3, a la ruta de la leche B le asigno el camión 1 y a la ruta de la leche C le asigno el camión 1. Para mantener este orden, manejaremos un vector de vectores que almacene las estructuras de los camiones en ese orden, camión 3, camión 1 y camión 2, para respetar que en la posición 0 siempre irá la ruta del camión que transporta la leche de tipo A, en la posición 1 siempre irá la ruta del camión que transporta leche tipo B y en la posición 2 siempre irá el camión que transporte la leche de tipo C.

Por lo cual habrán 3! combinaciones posibles de asignar camiones a las rutas

Una vez asignado el camión a las rutas de leche (se explica de esta forma para conservar la posición de las calidades de las leches) se procede a analizar si la capacidad del camión asignado

es suficiente para almacenar la cantidad total de leche de cada uno.

Acá viene la importancia del orden escogido, comprobar la capacidad del camión y hacerla factible.

Si la leche A excede la capacidad del camión asignado mueve granjas a la ruta de la leche B, de esta manera, A no infringirá la capacidad del camión y B no perderá la calidad de su leche.

Una vez analizado que A no infringe la capacidad de su camión, se analizará si B no infringe la capacidad de su camión, de esta forma, al igual que en el caso anterior se escogerán elementos al azar y los pasará a la ruta de C hasta satisfacer la restricción de capacidad.

Finalmente si C no es capaz de almacenar su propia leche en el camión, o en su defecto la leche enviada a él de otros camiones supera su capacidad se hace un re-start para asignar nuevamente camiones o encontrar una mejor combinación de leches que se adecue a las capacidades de los camiones.

Resulta fácil implementar esta funcionalidad de pasar cantidades de leche de una ruta a otra, porque ya no es necesario buscar o decidir donde posicionarlas, simplemente se pasa a la ruta de al lado, de esta forma cada ruta mantiene su calidad y es fácil de implementar con solo haber hecho esta representación. La función `movimiento1` es la que se encarga de pasar cantidades de leche al azar de un camión a otro.

Una vez vista la factibilidad de las capacidades de los camiones falta cerciorarse de que cada total de leche que llevan los camiones cumplan con la cuota de las plantas requeridas por cada tipo de leche.

Luego si el total de la leche A que lleva el camión, no es lo suficiente para cumplir la cuota de la planta, el camión asignado a la ruta de A no es el mejor por lo que se hace re-start para ver otra combinación de camiones.

Si la leche total de B que lleva el camión es menor a la requerida por la planta, éste pide leche al camión que contiene A hasta completar su cuota, sin que A infrinja su cuota y que la leche que A pasa a B no sobrepase la capacidad del camión de B.

Finalmente C, si no cumple su cuota pide leche a B primero y luego a A si B no alcanza a suplir la necesidad de leche de C, al pasar la leche verifican que tanto B como A no infrinjan su cuota con la planta y que la leche que le pasan a C no sobrepase su capacidad.

Si se infringe alguna restricción se hace un re-star para probar otra combinación de camiones o en su defecto si toca la misma combinación, pasa cantidades de leche diferentes.

## **Movimientos**

Se crearon 4 tipos de movimientos

### **Primer movimiento: movimiento2()**

El primero es mover una granja random de la ruta en todas las posiciones diferentes dentro de la misma, por ejemplo, si el random selecciona el valor 2, este se moverá en todas las posiciones posibles de la ruta:

### **Segundo movimiento: movimiento3()**



Este movimiento escoge un valor random y lo intercambia con otro dentro de la misma ruta.

**Tercer movimiento: movimientoTriple()**

Este movimiento involucra a las 3 rutas y su calidades, y solo funciona con la calidad A. Lo que hace es intercambiar una leche A de la ruta de A, con una A de la ruta de B con una A de la ruta C, A se la pasa a B, B se la pasa a C y C se la pasa a A.

**Cuarto movimiento: movimientoDoble()**

Este movimiento hace lo mismo que el anterior pero de manera más simple, se intercambian entre dos camiones, funciona en los siguientes casos A con B intercambiando A, A con C intercambiando A, B con C intercambiando B.

Finalmente la función generar vecindario escoge al azar uno de los dos primeros movimientos, luego genera el vecindario, el cual sería las posiciones de las granjas. Se obtiene la función de evaluación de cada vecino, y se compara con la función de evaluación de la ruta actual. Si esta es mejor el cambio se hace definitivo, sino sigue iterando hasta encontrar algo mejor, como los vecindarios son parciales al comienzo del algoritmo siempre se encontraran soluciones mejores, a medida que el algoritmo avanza comienza a atascarse en óptimos locales, para salir de allí sin hacer re-start se usan los movimientos Triples o Dobles, dependiendo si existen valores para los cuales hacer el swap.

Finalmente, terminadas las iteraciones se guarda la ruta actual en la mejor solución arras-trada, se genera el re-start y el algoritmo comienza a iterar otra vez.

## 7. Experimentos

Entorno de trabajo:

Maquina virtual Virtual Box de Oracle con Fedora 33 de 64 bits instalado como sistema operativo.

La maquina virtual cuenta con acceso a los siguientes recursos:

- procesador: Ryzen 3 dual-core 2200U 2.5GHZ
- memoria ram: 4 Gb de acceso para la maquina virtual.

## 8. Resultados

Re-start	10			10			10		
Iteraciones	10			50			100		
a36.txt	15619	10711	4152	15658	10137	4139	15,658	10137	4139
a44.txt	-	-	-	-	-	-	-	-	-
a55.txt	11365	8132	4799	11510	8227	4925	11535	8206	4880
a64.txt	9674	9301	5134	10178	9380	5403	10138	9057	5440
a80.txt	15276	9418	3980	15670	9702	4221	15768	9727	4300
c50.txt	32772	23853	8620	32772	23853	8634	32859	23908	8692
c75.txt	17600	17270	7061	17708	17414	7403	17708	17414	7403
eil22.txt	9626	4863	1466	9626	4863	1466	9626	4863	1466
eil23.txt	5764	3987	328	5764	1524	328	5764	1524	328
eil30.txt	2130	3807	1278	2130	3807	1278	2130	3807	1278
eil33.txt	10865	7876	1672	10872	7876	1672	10872	7876	1672
eil76.txt	46234	33307	13410	46449	32356	13570	46449	32356	13570

Al ver los resultados se hacen notar diferentes errores que posee el algoritmo, como por ejemplo no arrastrar la mejor solución, problema que se creía solucionado, por falta de tiempo habría que re-revisar nuevamente el algoritmo completo, para ver en que secciones falla. Cabe destacar que a medida que se fue probando este funcionaba correctamente, pero a medida que se implementaron más iteraciones, el algoritmo tendría a elegir una solución no mejor a la que ya tenía, aunque sí cercana a los resultados obtenidos en (Paredes-Belmar et al. (2016) [21].

## 9. Conclusiones

El estudio realizado al Milk Collector Problem with Blending dejó en evidencia ser un problema relativamente nuevo para la literatura, sus trabajos anteriores a este no consideraban más de una calidad de leche o mezcla lo que colocaba restricciones a un problema que podía relajarse en pos de reducir costos.

Si bien es cierto cada autor aborda el problema de una manera diferente, no todas las técnicas resuelven el mismo problema, hay unas que se ajustan de mejor manera que otras para resolver cada tipo de problemas. Lo que sí se puede destacar en cuanto a técnicas de resolución para MCP es TB y Brunch and cut. Uno de los solver más utilizados el CPLEX. Y para modelar el problema, se puede utilizar IP o MIP para resolver problemas pequeños y para modelos más grandes se pueden utilizar heurísticas que permitan limitar las instancias del problema, una de las más renombradas era dividir las rutas y camiones en clusters.

La limitación de los problemas en general siempre es la misma, qué ocurre si el problema escala. Gracias a los potentes solvers, heurísticas y el avance tecnológico, además de analizar elementos específicos para cada tipo de problemas que se desea abordar, hace que el hallar una solución óptima local u óptima general hace que el problema sea un poco más sencillo de resolver, y por lo mismo cada problema estudiado es diferente, pues son distintas las condiciones en las que está envuelta, pese a que en general sea un mismo problema, por lo que no toda formulación sirve para todos los problemas, ya que el hallar una solución dependerá de cómo modele la situación, y el tamaño del problema. Por lo que muchas veces la resolución no dependerá de la técnica utilizada, sino de cómo halla modelado el problema.

Como el MCP con mezcla es un problema nuevo para la literatura (ya que salió por primera vez en la literatura en el año 2016) aun quedan diferentes gamas del problema que se pueden

abordar, un caso especial a investigar podría ser un MCP con mezcla y aplicado con un camión y remolque, considerar una flota de vehículos que no sea heterogénea, minimizando la contaminación de los camiones, generar ventanas de tiempo, etc. Por lo que hay muchas variantes del mismo problema que aun no se han estudiado.

En cuanto a los resultados:

Dentro de los experimentos realizados en los archivos dados, uno de los resultados más importantes corresponde al resultado entregado por el archivo a44.txt, ya al probar las instancias por primera vez estas no dejaban de iterar para hallar una solución inicial factible, forzando la máquina a parar. Sin embargo, para los otros archivos estos si tenían respuesta e iteraban correctamente.

Se procedió entonces a hacer un análisis más exhaustivo a las capacidades de los camiones, a los requerimientos de la planta y a las leches totales de cada calidad producidas por las granjas, donde al aplicar lógica básica se llegó a la conclusión que el problema no tenía solución.

El análisis que se empleó fue el siguiente: Camion capacidad

- 1:25.000
- 2:20.000
- 3:15.000

Planta requerimientos

- A:20.000
- B:16.000
- C:12.000

Total leche granja

- A:24.400
- B:17.000
- C:15.600

El procedimiento creado para saber que el problema no tiene solución es asignar camiones de acuerdo a los requerimientos de la planta, esta función inicialmente fue implementada para saber si existiría algún camión que supla los requerimientos de la planta, si es que no se encontraba ninguno quería decir que el problema no tenía solución, puesto que no había camión que llevara tal cantidad de leche a la planta.

Se modificó esa funcionalidad para comprobar algo más, la infactibilidad del problema a44.txt. Para ello se toma el requerimiento de la planta, partiendo por el de la calidad de leche A, y comprueba cual es el camión de menor capacidad que suple el requerimiento de la planta para la leche A. Este sería el camión 2. Como el total de la leche de A es 24.400, supongamos es posible sacar el remanente de forma exacta (en la práctica, en la mayoría de los casos debería ser un valor un poco mayor) 4.400 y esa cantidad la incorporamos en el total de la leche B.

Nuevamente se vuelve a hacer el ejercicio, esta vez con el requerimiento de la planta B, ahora solo quedan disponibles los camiones 1 y 3, camión 3 queda descartado ya que no cumple con la cuota de la planta, por lo que obligatoriamente tomaría el camión 1. Ahora sin embargo el

total de B no es 17.000 sino 21.400, pero esto no es problema ya que la capacidad del camión 1 es 25.000.

Ahora el único camión que queda disponible para satisfacer la demanda C de la planta es el 3, sin embargo, la cantidad de leche total C es 15.600 y el camión no puede llevar tal cantidad, sobrando 600, el problema es que esos 600 si se incorporan en el camión 1 o el 2, degradará la calidad de la leche no cumpliendo sus cuotas respectivas, por lo que el problema no tendría solución.

Aunque esta función cumpla con el hecho de detectar si un problema no tiene solución, para este caso específico, no es la panacea, podría darse el caso que al quitar elementos sobrantes de leche de un camión a otro, estos no sean en las cantidades exactas como lo hace este, ya que es un problema de características discretas, por lo que habría que calcular cual es la menor cantidad de leche a sacar similar al sobrante. En algunos casos puede dar la misma cantidad, pero no en todos. Por lo que se podría mejorar esta función para descubrir si un problema no tiene solución.

Cabe destacar que las técnicas incompletas no descubren si un problema tiene solución o no. Por lo que la función que se ha empleado aquí no es parte del Hill Climbing, sino una función que toma las premisas de los datos entregados, hace un manejo oportuno para saber si el problema en cuestión tiene solución o no.

Otro caso particular relevante que se dio fue el caso del archivo a64.txt, la particularidad era que todos los camiones tenían la misma capacidad, pero una de las calidades tenía el mismo valor. El problema no se habría presentado si hubiese sido la calidad A, y que el total de la leche A hubiese sido mayor a 20.000, el problema se presentó porque la cantidad total de la leche C es menor 14.650, por lo que debería recibir una cantidad exacta de 5350 para completar la cuota de la planta, pero que además quepa justo en el camión, por lo que hay que hacer una combinación exacta de leche A y B para llegar a ese monto, lo cual genera un grado de dificultad más, ya que podría no haber cantidades de leche que generen ese número exacto para que el problema tenga solución.

Este problema fue relevante para la forma de implementación ya que la forma en que el algoritmo trabaja es moviéndose por soluciones factibles, al no haberlas, es imposible encontrar una solución inicial, por lo que el algoritmo no pararía de iterar.

Otro de los problemas que surgió al momento de probar los archivos es al hacer el movimiento doble o triple incurrió en una violación de segmento, esto se debe a que el movimiento doble y triple pueden pasar a una solución peor, pero mantienen en memoria la mejor solución hasta ese momento, pero al salir del for se genera el conflicto, por lo que se optó por comentar movimiento doble y triple, y generar más re-starts para hallar otra gama más de soluciones que antes se podían acceder con esos movimientos. cabe destacar que ambas funciones por separado cumplen su cometido, el problema surge al integrar la solución externa con la del auto, por lo que se considerará para un trabajo futuro.

En cuanto a los datos de los archivos restantes, es posible afirmar que si la capacidad de los camiones es mayor para cada tipo de cantidad total de leche, entonces el problema se transforma en uno donde acortar la ruta, siento un TSP para cada uno. Esto podría cambiar si la calidad de cada tipo de leche baja o el camino se hace aun más costoso.

Si faltan cantidades de leche para el requerimiento de la planta (ya sea B o C) sí o sí se producirá mezcla, siempre y cuando las cantidades a dar de leche por las otras plantas no infrinjan su cuota estipulada.

## 10. Bibliografía

### Referencias

- [1] C. Amiama, J. M. Pereira, L. Carpentier, and J. Salgado. Spatial decision support system for the route management for milk collection from dairy farms. *Transportation Letters*, 7(5):279–288, 2015.
- [2] C. Basnet, L. R. Foulds, and J. M. Wilson. An exact algorithm for a milk tanker scheduling and sequencing problem. *Annals of Operations Research*, 86(0):559—568, 1999.
- [3] X. Bing, M. de Keizer, J. M. Bloemhof-Ruwaard, and J. G. van der Vorst. Vehicle routing for the eco-efficient collection of household plastic waste. *Waste Manag*, 34(4):719—729, 2014.
- [4] M. Butler, P. Herlihy, and P. B. Keenan. Integrating information technology and operational research in the management of milk collection. *Journal of Food Engineering*, 70(3):341—349, 2005.
- [5] M. Butler, H. P. Williams, and L. A. Yarrow. The two-period travelling salesman problem applied to milk collection in ireland. *Computational Optimization and Applications*, 7(3):291–306, 1997.
- [6] H. Caramia and F. Guerriero. A milk collection problem with incompatibility constraints. *Computers and Electronics in Agriculture*, 40(2):130–143, 2010.
- [7] M. Caria, G. Todde, and A. Pazzona. Modelling the collection and delivery of sheep milk: A tool to optimise the logistics costs of cheese factories. *Agriculture*, 8(1):1–11, 2018.
- [8] I.-M Chao. A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29:33–51, 2002.
- [9] G. D. H. Claassen and T. H. B. Hendriks. An application of special ordered sets to a periodic milk collection problem. *European Journal of Operational Research*, 180(2):754—769, 2007.
- [10] I. Dayarian, T. G. Crainic, M. Gendreau, and W. Rei. An adaptive large neighborhood search heuristic for a multi-period vehicle routing problem. *Technical Report CIRRELT-2013-60, Interuniversity Research Center on Enterprise Networks, Logistics and Transportation (CIRRELT)*, 2013.
- [11] I. Dayarian, T. G. Crainic, M. Gendreau, and W. Rei. A branch-and-price approach for a multiperiod vehicle routing problem. *Computers and Operations Research*, 55:167–184, 2015.
- [12] I. Dayarian, T. G. Crainic, M. Gendreau, and W. Rei. A column generation approach for a multi-attribute vehicle routing problem. *European Journal of Operational Research*, 241(3):888—906, 2015.
- [13] A. E. Dooley, W. J. Parker, and H. T. Blair. Modelling of transport costs and logistics for on-farm milk segregation in new zealand dairying. *Computers and Electronics in Agriculture*, 48(2):75—91, 2005.
- [14] A. Hoff and A. Lokketangen. A tabu search approach for milk collection in western norway using trucks and trailers. In *6th Triennial Symposium on Transportation Analysis, TRISTAN, Pukett, Thailand*, 2007.

- [15] M. Igbaria, R. H. Sprague, C. Basnet, and L. Foulds. The impact and benefits of a dss: The case of fleetmanager. *Information and Management*, 31(4):215–225, 1996.
- [16] N. Lahrichi, T. G. Crainic, M. Gendreau, W. Rei, and L. M. Rousseau. Strategic analysis of the dairy transportation problem. *Journal of the Operational Research Society*, 66(1):44–56, 2013.
- [17] R. Masson, N. Lahrichi, and L. Rousseau. A two-stage solution method for the annual dairy transportation problem. *European Journal of Operational Research*, 251(1):36–43, 2015.
- [18] M. Montero, D. Canales, G. Paredes-Belmar, and R. Soto. A prize collecting problem applied to a real milk collection problem in chile. *Institute of Electrical and Electronics Engineers Inc.*, IEEE Congress on Evolutionary Computation (CEC):1415–1422, 2019.
- [19] M. K. Mumtaz, M. N. Jalil, and K. A. Chatha. Designing the milk collection network using integrated location routing approach. in international conference on industrial engineering and operations management. *Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management Bali, Indonesia, January 7 – 9, 2014*.
- [20] G. Paredes-Belmar, A. Lüer-Villagra, V. Marianov, and C. Cortés. The milk collection problem with blending and collection points. *Computers and Electronics in Agriculture*, 134:109–123, 2019.
- [21] G. Paredes-Belmar, V. Marianov, A. Bronfman, and C. Obreque. A milk collection problem with blending. *Transportation Research Part E Logistics and Transportation Review*, 94:26–43, 2016.
- [22] U. Pasha, A. Hoff, and A. Lokketangen. A hybrid approach for milk collection using trucks and trailers. *Annals of Management Science*, 3(1):87–110, 2014.
- [23] O. Polat, C. B. Kalayci, B. Bilgen, and D. Topaloglu. An integrated mathematical model for the milk collection problem. *Pamukkale University Journal of Engineering Sciences*, 25(9):1087–1096, 2019.
- [24] O. Polat, C. B. Kalayci, B. Bilgen, and D. Topaloglu. Methodology to solve a special case of the vehicle routing problem: A case study in the raw milk transportation system. *AgriEngineering*, 1(1):75–93, 2019.
- [25] O. Polat and D. Topaloglu. Milk collection network design in a fuzzy enviroment. *Economy and Business Journal*, 13(1):376–384, 2019.
- [26] O. Polat and D. Topaloglu. Vehicle routing problem with simultaneous pickup and delivery for milk collection in galeras, sucre. *18th LACCEI International Multi-Conference for Engineering, Education, and Technology: “Engineering, Integration, and Alliances for a Sustainable Development” “Hemispheric Cooperation for Competitiveness and Prosperity on a Knowledge-Based Economy”, Virtual EditionAt: Boca Raton, Florida, US, 2020*.
- [27] P. Prasertsri and R. L. Kilmer. Scheduling and routing milk from farm to processors by a cooperative. *Journal of Agribusiness*, 22(2):93–106, 2004.
- [28] J. Sankaran and R. Ubgade. Routing tankers for dairy milk pickup. *Interfaces*, 24(5):59–66, 1994.
- [29] K. Sethanan and R. Pitakaso. Differential evolution algorithms for scheduling raw milk transportation. *Computers and Electronics in Agriculture*, 121:245–259, 2016.
- [30] C. D. Tarantilis and C. T. kiranoudis. A meta-heuristic algorithm for the efficient distribution of perishable foods. *Journal of food Engineering*, 50(1):1–9, 2001.

- [31] J. Villagrán, E. Montero, and G. Paredes-Belmar. An iterated local search approach to solve the milk collection problem with blending. *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2020.
- [32] R. Vásquez and M. Valencia. Model of routing for raw milk collection using genetic algorithms. *Sistemas y Telemática*, 12(31):77–87, 2014.