

**UNIVERSIDAD NACIONAL DE PIURA**

**Facultad de Ciencias**

**Escuela Profesional de Matemática**



**TESIS**

**OPTIMIZACIÓN DE RUTAS DE DOS NIVELES MEDIANTE UN  
MODELO INTEGRADO DE LOCALIZACIÓN Y RUTEO:  
APLICACIÓN EN LA LOGÍSTICA URBANA**

**Presentada por:**

**YOSBI JHON GOLLÉS PAICO**

**Para optar el título profesional de**

**LICENCIADO EN MATEMÁTICA**

**Línea de investigación: Matemática y estadística**

**Sublínea de investigación: Inteligencia Artificial**

**PIURA, PERÚ**

**2021**

**UNIVERSIDAD NACIONAL DE PIURA**

**Facultad de Ciencias**

**Escuela Profesional de Matemática**



**TESIS**

**OPTIMIZACIÓN DE RUTAS DE DOS NIVELES MEDIANTE UN  
MODELO INTEGRADO DE LOCALIZACIÓN Y RUTEO:  
APLICACIÓN EN LA LOGÍSTICA URBANA**

**Linea de investigación: Matemática y estadística**

**Sublinea de investigación: Inteligencia Artificial**

---

Br. Yosbi Jhon Gollés Paico  
Tesisista

---

Dr. Flabio Alfonso Gutiérrez Segura.  
Asesor

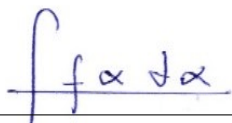
## DECLARACIÓN JURADA DE LA ORIGINALIDAD DE LA TESIS

Yo: Yosbi Jhon Gollés identificado con DNI N° 72787508, Bachiller de de Escuela Profesional de Matemáticas, de la Facultad de Ciencias y domiciliado en Av. Cesar Vallejo mk 25 urb las Mercedes, del Distrito de Piura, Provincia de Piura, Departamento de Piura, Celular: 916641577, Email: gollesyosbi@gmail.com.

**DECLARO BAJO JURAMENTO:** que la tesis que presento es original e inédita, no siendo copia parcial ni total de una tesis desarrollada, y/o realizada en el Perú o en el Extranjero, en caso contrario de resultar falsa la información que proporciono, me sujeto a los alcances de lo establecido en el Art. N° 411, del código Penal concordante con el Art. 32° de la Ley N° 27444, y Ley del Procedimiento Administrativo General y las Normas Legales de Protección a los Derechos de Autor.

En fe de lo cual firmo la presente.

Piura, Agosto del 2021.



---

Yosbi Jhon Gollés Paico  
DNI N° 72787508

UNIVERSIDAD NACIONAL DE PIURA

Facultad de Ciencias

Escuela Profesional de Matemática



TESIS

**OPTIMIZACIÓN DE RUTAS DE DOS NIVELES MEDIANTE UN  
MODELO INTEGRADO DE LOCALIZACIÓN Y RUTEO:  
APLICACIÓN EN LA LOGÍSTICA URBANA**

**Linea de investigación: Matemática y estadística**

**Sublinea de investigación: Inteligencia Artificial**

---

Dr. Luis Aguilar Ibañez,  
Presidente

---

Dr. José del Carmen Silva Mechato  
Secretario

---

M.Sc. María Isabel Hidalgo Tinedo,  
Vocal



*A Mery, mi madre,  
por darme la vida.*

# Agradecimientos

Creo conveniente y considerado expresar mi agradecimiento explícito a las personas que, de forma directa o indirecta, contribuyeron al desarrollo de esta tesis la cual representa el final de una fase académica y el inicio de otra.

Principalmente agradezco a mi asesor y padre académico, el Dr. Flabio Gutierrez Segura, por su invaluable guía personal y profesional que me ha inspirado y motivado a estudiar el uso de la ciencia matemática como medio para el mejoramiento de las sociedades.

Así también agradezco la iniciativa del Fondo Nacional de Desarrollo Científico y Tecnológico (FONDECYT), que financió esta tesis mediante el proyecto (125-2018-FONDECYT-BM-IADT-AV), MODELOS Y META HEURÍSTICAS PARA EL RUTEAMIENTO DEL PROCESO DE DISTRIBUCIÓN DE PRODUCTOS Y COLECTA TRANSPORTE DE RESIDUOS SÓLIDOS EN EL CONTEXTO DE LA LOGÍSTICA URBANA.

# Índice general

|  |             |
|--|-------------|
| <b>Índice general</b>  | <b>III</b>  |
| <b>Índice de tablas</b>  | <b>V</b>    |
| <b>Índice de figuras</b>                                       | <b>VI</b>   |
| <b>Resumen</b>   | <b>VII</b>  |
| <b>Abstract</b>  | <b>VIII</b> |
| <b>Introducción</b>  | <b>1</b>    |
| <b>1. Aspectos de la problemática</b>                          | <b>3</b>    |
| 1.1. Descripción de la realidad problemática . . . . .         | 3           |
| 1.1.1. Formulación del problema de investigación . . . . .     | 4           |
| 1.2. Justificación e importancia de la investigación . . . . . | 4           |
| 1.3. Objetivos . . . . .                                       | 5           |
| 1.3.1. Objetivo general . . . . .                              | 5           |
| 1.3.2. Objetivos específicos . . . . .                         | 5           |
| 1.4. Delimitación de la investigación . . . . .                | 5           |
| <b>2. Marco teórico</b>  | <b>6</b>    |
| 2.1. Antecedentes de la investigación . . . . .                | 6           |
| 2.2. Bases teóricas . . . . .                                  | 7           |
| 2.2.1. Teoría de grafos . . . . .                              | 7           |
| 2.2.2. Optimización combinatoria . . . . .                     | 8           |
| 2.2.3. Complejidad algorítmica . . . . .                       | 9           |
| 2.2.4. Complejidad del problema . . . . .                      | 10          |
| 2.2.5. Métodos de solución . . . . .                           | 11          |
| 2.2.6. Problema de localización de facilidades . . . . .       | 12          |
| 2.2.7. Problema de ruteo de vehículos . . . . .                | 13          |
| 2.2.8. Problema de localización-ruteo de N niveles . . . . .   | 15          |
| 2.3. Hipótesis . . . . .                                       | 16          |

|   |           |
|---|-----------|
| 2.3.1. Variables . . . . .                              | 16        |
| <b>3. Marco metodológico</b>                            | <b>17</b> |
| 3.1. Enfoque y diseño . . . . .                         | 17        |
| 3.2. Métodos y procedimientos . . . . .                 | 17        |
| 3.2.1. Metodología . . . . .                            | 17        |
| 3.2.2. Equipos y herramientas computacionales . . . . . | 18        |
| 3.3. Fase de precisión . . . . .                        | 18        |
| 3.3.1. 2E-LRP . . . . .                                 | 19        |
| 3.3.2. Entrega y colecta simultáneas . . . . .          | 20        |
| 3.3.3. Consideraciones ambientales . . . . .            | 20        |
| 3.4. Fase de Modelado . . . . .                         | 21        |
| 3.4.1. Descripción del problema . . . . .               | 21        |
| 3.4.2. Formulación del modelo . . . . .                 | 22        |
| 3.4.3. Refinamiento del modelo . . . . .                | 27        |
| 3.4.4. Implementación del modelo . . . . .              | 28        |
| 3.4.5. Desarrollo del generador de instancias . . . . . | 33        |
| 3.4.6. Implementación del generador . . . . .           | 35        |
| <b>4. Resultados y Discusión</b>                        | <b>39</b> |
| 4.1. Resultados . . . . .                               | 39        |
| 4.1.1. Descripción de las instancias . . . . .          | 39        |
| 4.1.2. Validación del modelo . . . . .                  | 39        |
| 4.1.3. Evaluación de la eficiencia del modelo . . . . . | 41        |
| 4.2. Discusión . . . . .                                | 43        |
| <b>Conclusiones</b>                                     | <b>44</b> |
| <b>Recomendaciones</b>                                  | <b>45</b> |
| <b>Referencias bibliográficas</b>                       | <b>49</b> |

# Índice de tablas

|  |    |
|--|----|
| 3.1. Parámetros y variables utilizadas en nuestro modelo matemático del<br>2E-LRPSPD-E . . . . . | 23 |
| 4.1. Instancia simulada de diez clientes. . . . .  | 40 |
| 4.2. Solución óptima. . . . .  | 41 |
| 4.3. Resultados computacionales. . . . .   | 42 |

# Índice de figuras

|  |    |
|--|----|
| 2.1. La división de las clases de complejidad bajo la hipótesis $\mathcal{P} \neq \mathcal{NP}$ . . . . .  | 11 |
| 2.2. Un ejemplo de FLP estándar con tres depósitos potenciales y siete clientes. La solución propone usar sólo los depósitos 0 y 1. . . . .  | 13 |
| 2.3. Un ejemplo de VRP estándar con un depósito y ocho clientes. La solución propone es uso de tres rutas. . . . .   | 13 |
| 2.4. Representación de un sistema de distribución de N niveles . . . . .   | 15 |
| 3.1. Genealogía del 2E-LRP . . . . .   | 19 |
| 3.2. Ubicaciones periféricas de las instalaciones satélite . . . . .   | 20 |
| 3.3. Modelo céntrico multi-escalón de la estructura de una ciudad . . . . .  | 34 |
| 3.4. Ejemplo de costos de viaje: distancia mínima de $i$ a $j$ (rojo), distancia mínima de $j$ a $i$ (azul), tiempo mínimo de $i$ a $j$ (verde), tiempo mínimo de $j$ a $i$ (amarillo). . . . .      | 35 |
| 4.1. Visualización de la instancia simulada (los puntos azules, verdes y rojos representan la instalación maestra, las potenciales instalaciones satélites y los clientes, respectivamente). . . . . | 40 |
| 4.2. Visualización de la solución óptima. . . . .  | 41 |
| 4.3. Evolución del tiempo de ejecución requerido. . . . .  | 42 |
| 4.4. Evolución del costo óptimo. . . . .   | 43 |

# Resumen

La creciente demanda de los sistemas de distribución, ahora amplificada por la crisis de salud Covid-19, por parte de las economías urbanas, ha derivado en un aumento sustancial de las externalidades negativas como las emisiones contaminantes y la congestión. Esto plantea la necesidad de implementar sistemas más ecológicos para minimizar el impacto negativo sobre el medio ambiente y el bienestar social. Se propuso un modelo de programación de enteros puro (PILP) para el problema de localización-ruteo de dos niveles para entrega y colecta simultáneas con consideraciones ambientales (2E-LRPSPD-E), desde el enfoque de la logística urbana. Se obtuvo un modelo respetuoso del medio ambiente, que considera la carga y la distancia por carreta para estimar y minimizar las emisiones de dióxido de carbono (CO<sub>2</sub>).

**Palabras clave:** Sistemas de distribución de dos niveles, Problema de localización-ruteo, Emisión de CO<sub>2</sub>, Optimización matemática.

# Abstract

The growing demand for distribution systems, now amplified by the Covid-19 health crisis, by urban economies, has led to a substantial increase in negative externalities such as polluting emissions and congestion. This raises the need to implement more ecological systems to minimize the negative impact on the environment and social well-being. A pure integer programming model (PILP) was proposed for the two-echelon location-routing problem for simultaneous delivery and collection with environmental considerations (2E-LRPSPD-E), from the urban logistics approach. An environmentally friendly model was obtained, which considers the freight and distance road to estimate and minimize carbon dioxide (CO<sub>2</sub>) emissions.

**Keywords:** Two-echelon distribution systems, Location-routing problem, CO<sub>2</sub> emission, Mathematical optimization.

---

# Introducción

Como consecuencia de la creciente y continua demanda de los servicios de transporte de carga, las áreas urbanas se han visto afectadas por una serie de externalidades negativas (congestión, emisiones contaminantes, consumo de combustible, ruido y otros factores ambientales). La logística urbana surge como una iniciativa para gestionar los sistemas de distribución de carga, teniendo en consideración tales externalidades, sin interrumpir las actividades económicas.

Los sistemas de distribución multinivel, y en particular los de dos niveles, son sistemas de frecuente estudio e implementación por parte de la logística urbana. Gestionar esta clase de sistemas de distribución implica optimizar dos tipos de decisiones interdependientes: una estratégica (localizar depósitos centrales e intermedios) y otra táctica-operacional (asignación de clientes y diseño de rutas). [Salhi and Rand \(1989\)](#) cuantificaron por primera vez los beneficios potenciales que se obtuvieron al incluir decisiones de ruteo de vehículos en la localización de depósitos. Sin embargo, dentro del contexto urbano, el enfoque típico para la planificación de estos sistemas ha sido el secuencial, ignorándose los beneficios de una planificación global.

En este trabajo se desarrolla e implementa un modelo integrado de localización y ruteo para optimizar rutas de dos niveles como aplicación en la logística urbana. Se formula el problema de localización-ruteo de dos niveles para entrega y colecta simultáneas con consideraciones ambientales (2E-LRPSPD-E), y se propone un modelo de programación lineal de enteros puro (PILP) para su optimización. El modelo se implementa en la herramienta de optimización GLPK y se evalúa sobre instancias simuladas y generadas aleatoriamente. El modelo minimiza tanto los costes económicos del sistema como la emisión de dióxido de carbono (CO<sub>2</sub>) producida por las operaciones de transporte, tomando en cuenta la carga y la distancia por carretera.

Esta tesis se estructura de la siguiente manera: en el Capítulo [1](#) se describe la realidad problemática sobre la que surge el problema de optimización de dos rutas. En el Capítulo [2](#) se comentan las teorías que permiten abordar el diseño de los modelos de optimización para problemas localización de instalaciones y programación de rutas. En el Capítulo [3](#) se plantea el modelo de programación lineal de

enteros puro para la optimización de rutas de dos niveles y se construye un generador que permitió validar y evaluar este modelo. En el Capítulo 4 se presentan y se discuten los resultados de la validación y evaluación desarrollada. Finalmente se exponen conclusiones generales de la investigación y las recomendaciones para futuras investigaciones.

# Capítulo 1

## Aspectos de la problemática

### 1.1. Descripción de la realidad problemática

El acelerado crecimiento demográfico y espacial de las áreas urbanas las ha transformado en importantes ejes económicos. La demanda creciente de bienes y servicios dentro de las ciudades, ha derivado en un mayor nivel de demanda y dependencia de los servicios de transporte de carga por parte de las economías urbanas.

La gestión, mayoritariamente privada, de estos sistemas de transporte persigue objetivos exclusivamente económicos sin tener en cuenta los efectos negativos sobre la congestión, la seguridad o el medio ambiente. Esto ha motivado el desarrollo del concepto de logística urbana, cuyo fin es regular las actividades de distribución de carga para minimizar las posibles interrupciones en el paisaje urbano (Rodríguez et al., 2018). Esta reconoce explícitamente que el transporte de mercancías en áreas urbanas tiene un impacto negativo –así como positivo– en la vida de las personas que viven en estas áreas (Verma and Raghubanshi, 2018).

Entre los principales esfuerzos de la logística urbana por reducir las molestias asociadas con el transporte de carga en áreas urbanas, se encuentra el uso de instalaciones satélites como puntos de transferencia. Estas instalaciones satélites definen un sistema de distribución multinivel que implica utilizar: (i) varios tipos de vehículos en diferentes niveles, y (ii) instalaciones satélites configuradas para consolidar y transbordar carga (Li et al., 2018).

En la práctica, los sistemas de distribución de dos niveles se han convertido en la alternativa más frecuente, puesto que su implementación produce grandes ventajas desde el punto de vista ambiental. Tiene la función de limitar el tránsito vehicular pesado hasta instalaciones satélites ubicadas en zonas periféricas de la ciudad, donde se descargan, clasifican y consolidan las cargas, y a partir de las cuales, vehículos más pequeños terminan por distribuir la carga a los diferentes destinos dentro del área urbana (Cuda et al., 2015).

El diseño óptimo de un sistema de distribución plantea dos problemas difíciles de optimización combinatoria, a saber, determinar la localización de depósitos y satélites, y diseñar las rutas para el transporte de carga en cada nivel. Estos dos tipos de decisiones se han abordado por separado durante mucho tiempo, pero el progreso continuo en las técnicas de optimización ha hecho posibles enfoques integrados conocidos bajo el nombre de problemas localización-ruteo ([Prodhon and Prins, 2014](#)).

Lo antes expuesto motivó la realización de este trabajo de investigación, en el que se propone un modelo integrado de localización y ruteo para la optimización de rutas de dos niveles como aplicación en la logística urbana, que busca reducir las externalidades negativas producidas por el transporte de carga dentro de las ciudades. Esta investigación se ajusta a los propósitos de la logística urbana precisados con anterioridad.

### **1.1.1. Formulación del problema de investigación**

¿Cómo desarrollar e implementar un modelo integrado de localización y ruteo para optimizar rutas de dos niveles como aplicación en la logística urbana?

## **1.2. Justificación e importancia de la investigación**

Existen escasas investigaciones sobre la optimización de rutas de dos niveles abordadas mediante modelos integrados de localización y ruteo, que registren las consideraciones necesarias para alinearse con los requerimientos de la logística urbana. En particular, no se encontraron trabajos de este tipo en Perú.

El principal aporte de este trabajo es académico. En efecto, el conocimiento organizado (la síntesis conceptual necesaria para estudiar la optimización de rutas, el desarrollo de un nuevo modelo de optimización y la implementación en un lenguaje de modelado para su evaluación computacional) de este trabajo servirán como base para ulteriores investigaciones en el campo de la optimización de sistemas de distribución multinivel como aplicación a la logística urbana.

Otro aporte importante es el social. Se espera que los aportes de este trabajo sirvan como herramientas que las instituciones locales (y nacionales) de gestión pública puedan usar para diseñar sistemas de distribución multinivel que se ajusten al propósito de la logística urbana, en pro del desarrollo sostenible de las ciudades.

## **1.3. Objetivos**

### **1.3.1. Objetivo general**

Diseñar e implementar un modelo integrado de localización y ruteo para optimizar rutas de dos niveles aplicado a la logística urbana.

### **1.3.2. Objetivos específicos**

- Definir el problema general de localización-ruteo de dos niveles en el contexto de la logística urbana.
- Diseñar un modelo integrado de localización y ruteo para optimizar rutas de dos niveles como aplicación en la logística urbana.
- Diseñar un generador aleatorio de instancias simuladas.
- Implementar el modelo en paquete libre de optimización GLPK.
- Evaluar el modelo sobre instancias simuladas.

## **1.4. Delimitación de la investigación**

La investigación se desarrolló en la Universidad Nacional de Piura, entre los años 2019 y 2020, y es financiado por el FONDECYT mediante el proyecto (125-2018-FONDECYT-BM-IADT-AV), MODELOS Y META HEURÍSTICAS PARA EL RUTEAMIENTO DEL PROCESO DE DISTRIBUCIÓN DE PRODUCTOS Y COLECTA TRANSPORTE DE RESIDUOS SÓLIDOS EN EL CONTEXTO DE LA LOGÍSTICA URBANA.

El trabajo se limita a los problemas de logística urbana que se pueden tratar como problemas de localización y ruteo de vehículos, los cuales son formulados como problemas de optimización combinatoria. El problema se modela como un problema de Programación Lineal Entero Puro (PILP), es implementado en la herramienta libre de optimización denominada GLPK que resuelve modelos de programación lineal (LP) a gran escala con los métodos clásicos de programación matemática, y se evalúa en instancias simuladas.

# Capítulo 2

## Marco teórico

En esta capítulo se presenta una base conceptual para abordar, en particular, el diseño de un *modelo integrado de localización y ruteo* para la *optimización de rutas de dos niveles*, y en general, problemas de logística de transporte.

### 2.1. Antecedentes de la investigación

[Soysal et al. \(2015\)](#) presentan una formulación MILP integral para un problema de ruteo de vehículos capacitados de dos niveles dependientes del tiempo (2E-CVRP). Este modelo tiene en cuenta el tipo de vehículo, la distancia recorrida, la velocidad del vehículo, la carga, múltiples zonas horarias y emisiones de gas. Se evalúan varias versiones del modelo que difieren respecto a la función objetivo, resolviendo una instancia real en CPLEX para mostrar la aplicabilidad del modelo. Sin embargo, la localización de la instalación maestra y las instalaciones satélites se suponen establecidas. Además, no se desarrolla ningún método o enfoque para solucionar el problema.

[Wang et al. \(2017\)](#) mejoran el modelo 2E-CVRP-E desarrollado en [Soysal et al. \(2015\)](#) sustituyendo el modelo de arco utilizado en el segundo nivel por un modelo de ruta. Esto resulta útil para abordar formulaciones con funciones objetivo complejas. En particular, la que resulta de considerar el consumo de combustible y las emisiones de  $CO_2$  en las decisiones de ruta. Además, diseñan una metaheurística basada en la búsqueda de vecindad variable (VNS) y la programación de enteros para resolver el 2E-CVRP-E. La programación entera en la metaheurística se utiliza como una técnica de optimización posterior para encontrar mejores soluciones perdidas por el algoritmo VNS o para construir las rutas de primer nivel de menor costo. No obstante, el enfoque de este trabajo es secuencial; las ubicaciones de los depósitos y satélites de transferencia se suponen establecidas a priori.

[Zhao et al. \(2018\)](#) presentan un modelo para el problema capacitado de localización-ruteo de un sistema de distribución de dos niveles (2E-CLRP), que considera

algunas limitaciones prácticas, como la capacidad del vehículo, las horas de trabajo y las restricciones de tráfico en el área urbana. Además, se presenta un algoritmo heurístico de aproximación cooperativa que permite resolver de forma óptima instancias más grandes a diferencia de los métodos desarrollados por la mayoría de las investigaciones desarrolladas con anterioridad. La formulación función objetivo de este modelo no cuantifica de forma explícita costos ambientales como la emisión de  $CO_2$ .

## 2.2. Bases teóricas

En las secciones 2.2.1 y 2.2.2 presentamos una descripción concisa de la teoría de grafos y de optimización combinatoria; repasamos los conceptos de complejidad algorítmica, complejidad del problema, y resumimos los métodos de solución para problemas de optimización combinatoria. Estos conceptos son la base teórica que subyace a la representación, formulación, análisis y solución de los problemas de localización y ruteo. En las secciones 2.2.6, 2.2.7, y 2.2.8 presentamos el *problema de localización de facilidades*, el *problema de ruteo de vehículos* y el *problema de localización-ruteo de  $N$  niveles*, respectivamente.

### 2.2.1. Teoría de grafos

La teoría de grafos permite representar muchos problemas de la logística de transporte. Esta representación es efectiva porque describe sin ambigüedades la relación por pares entre los elementos de una red de rutas (Rahmani, 2015).

En general, una red de rutas se puede modelar, según su naturaleza direccional, como un grafo dirigido o un grafo no-dirigido.

Un *grafo dirigido*  $G = (N, A)$  es un dupla de conjuntos finitos y no vacíos, donde los elementos de  $N$  son llamados nodos y los elementos de  $A$  son pares ordenados de nodos llamados *arcos*. En una red de rutas, un nodo puede corresponder a una instalación o a un cliente, mientras que un arco simboliza un segmento de ruta unidireccional que conecta dos nodos. Este modelo considera una variable de decisión binaria con dos índices  $x_{ij}$  que es igual a 1 si el arco asociado al par ordenado  $(i, j)$  forma parte del diseño óptimo de la red.

Por otro lado, un *grafo no-dirigido*  $G = (N, E)$  es un dupla de conjuntos finitos y no vacíos, donde los elementos de  $E$  son pares desordenados de nodos llamados *aristas*. Para este caso, una arista simboliza un segmento de ruta bidireccional que conecta dos nodos. Este modelo puede considerar una variable de decisión binaria de dos índices  $x_{ij}$  que es igual a 1 si en el diseño óptimo de la red se usa la arista

asociada al par desordenado  $(i, j)$ , o considerar una variable binaria de un solo índice  $x_k$  que es igual a 1 si la arista  $k \in E$  forma parte de la red óptima.

Una red de rutas se define en un grafo dirigido (o no-dirigido) que no contiene ni arcos (aristas) asociados a pares de nodos iguales, ni arcos (aristas) múltiples asociados con un mismo par de nodos. Un grafo en tales condiciones es un grafo *simple*. A menudo también modelamos la red como un grafo *completo*, esto es, un grafo que contiene todas los arcos (o aristas) posibles. En adelante daremos por hecho que los grafos que modelan nuestros problemas son simples y completos.

Es posible asignar un peso a los arcos (aristas) de un grafo dirigido (no-dirigido) a través de una *función de ponderación* no negativa  $C : A \mapsto \mathbb{R}$ . Los grafos dotados de una función de ponderación se llaman grafos *ponderados*. La ponderación de un arco o arista puede corresponder a distancias, tiempos de recorrido, capacidad de flujo o un límite de velocidad.

En el proceso de distribución de carga, con frecuencia, los vehículos describen rutas (camino o trayectorias) cerradas. Formalmente, un *camino* (*trayectoria*) es un conjunto de aristas (arcos) ordenables en una secuencia. Un *ciclo* (*circuito*) es un *camino* (*trayectoria*) que conecta un nodo consigo mismo. Muchas veces conviene especificar el número de arcos o aristas que componen un camino o trayectoria, es decir la *longitud* de una ruta, añadiendo un prefijo numérico, e.g., un 2-ciclo es un ciclo de longitud 2. Por otro lado, la ponderación de un camino o trayectoria será la suma de las ponderaciones de todos sus arcos o aristas componentes.

### 2.2.2. Optimización combinatoria

Los problemas de logística de transporte pertenecen a una clase bien definida de problemas de optimización llamados problemas de optimización combinatoria.

Un problema de optimización combinatoria (POC) se caracteriza por: (i) sus variables son discretas y (ii) determina un espacio de soluciones finito. Los POCs tienen una representación concisa con estructura combinatoria. Tal es el caso de los problemas de optimización de rutas que representaremos a través de grafos completos.

Desde una perspectiva teórica los POCs se resuelven trivialmente mediante la enumeración exhaustiva de sus soluciones. Sin embargo, en la práctica, donde el tiempo es el factor más importante, esta técnica es inviable, ya que las representaciones de tamaño realista determinan espacios de soluciones de enorme tamaño. En un problema de rutas, la adición de un único nodo, produce un incremento exponencial en el conjunto de rutas posibles.

Formalmente, un *problema de optimización combinatoria* es un conjunto de pares ordenados  $(S, f)$  llamados *instancias*, que se definen usando: (i) un conjunto finito

de variables  $x_1, \dots, x_n$  asociadas a un conjunto de dominios discretos  $D_1, \dots, D_n$ , respectivamente y (ii) un conjunto finito de relaciones  $R_1, \dots, R_n$  entre variables llamadas *restricciones*.

- El conjunto  $S$  es llamado *espacio de búsqueda* y es tal que:

$$S = \{(x_1, \dots, x_n) \in D_1 \times \dots \times D_n : R_i(x_1, \dots, x_n), \text{ para } i = 1, \dots, n\}$$

Los elementos de  $S$  se llaman *soluciones factibles*.

- La función  $f$  es llamada *función objetivo* y es tal que:

$$f : D_1 \times \dots \times D_n \longrightarrow \mathbb{R}$$

Decimos que una solución  $r \in S$  es una *solución óptima de minimización* para la instancia  $(S, f)$  si se cumple que

$$f(r) \leq f(s), \text{ para todo } s \in S.$$

Análogamente, decimos que  $r$  es una *solución óptima de maximización* cuando la desigualdad se cumple en sentido inverso.

En muchos casos es posible formular un POC como un *modelo de programación lineal de enteros puro* (PILP). En una formulación de este tipo las variables son enteras y tanto la función objetivo como las restricciones son expresiones lineales. Su forma general es

$$\text{mín } \sum_{j \in B} c_j x_j \tag{2.1}$$

*sujeto a:*

$$\sum_{j \in B} a_{ij} x_j \leq b_i, \quad \forall i \in M \tag{2.2}$$

$$x_j \in \mathbb{Z}, \quad \forall j \in I_n \tag{2.3}$$

donde  $I_n = \{1, \dots, n\}$  es el conjunto de índices de las variables enteras  $x_j$ .

### 2.2.3. Complejidad algorítmica

En un contexto computacional un *problema* es la descripción general de una disposición junto a una cuestión sobre ésta a ser respondida. Esta descripción se formula mediante un conjunto de parámetros (datos de entrada) y la respuesta, si existe, se formula usando un conjunto de variables (conjunto de salida).

Llamamos *instancia* de un problema a una descripción concreta de una disposición para la cual existe una respuesta. En los POCs los datos de entrada y salida son

las instancias  $(S, f)$  y las soluciones asociadas  $r \in S$ , respectivamente. Llamamos ejemplo de un problema a cualquier par  $(I, r)$  ordenado donde  $I$  es una instancia y  $r$  su solución. Tanto los datos de entrada como los de salida están codificadas como cadenas de caracteres de cierto alfabeto. Definimos el *tamaño* de una instancia de un problema como la longitud de la cadena que la codifica.

Un *autómata* es un dispositivo con la capacidad de realizar de modo automático ciertas operaciones básicas. Un *algoritmo* que resuelve un problema es un conjunto finito de ordenes diseñadas para dirigir un autómata desde cualquier instancia del problema hacia su solución.

Podemos definir una medida de eficiencia de un algoritmo  $A$  para resolver un problema, como una función  $C_A$  de variable natural, que asigna el máximo de número de órdenes  $C_A(n)$  que requiere el algoritmo para resolver una instancia de tamaño  $\leq n$ . Con frecuencia expresamos  $C_A$  asintóticamente respecto a  $n$  usando la notación  $\Theta$  para permitir la comparación entre algoritmos. Decimos que un algoritmo  $A$  es de complejidad  $\Theta(f(n))$  si  $\exists M > 0, \exists n_0$  tal que  $\forall n \geq n_0, C_A \leq Mf(n)$ . Las funciones algorítmicas más utilizadas son:

- $\Theta(1)$ : constante e independiente del tamaño de la instancia,
- $\Theta(\log(n))$ : logarítmico al tamaño de la instancia,
- $\Theta(n)$ : lineal al tamaño de la instancia,
- $\Theta(nc)$ : ( $c \geq 2$  una constante): polinomio del tamaño de la instancia,
- $\Theta(a^n)$  ( $a > 1$  una constante): exponencial al tamaño de la instancia.

#### 2.2.4. Complejidad del problema

La Teoría de la Complejidad tiene por objetivos determinar que tan difícil es resolver un problema estudiando la complejidad de su mejor algoritmo exacto, y estudiar las garantías de rendimiento viables en enfoques no exactos (Riera, 2002).

Para cada problema de optimización  $\min\{f(s) : s \in S\}$  existe un problema de decisión asociado: ¿existe una solución  $r \in S$  con valor  $f(r) \leq K$ ? Los problemas de decisión se dividen en las siguientes clases según su complejidad (Lafifi, 2015):

- Clase  $\mathcal{P}$ : un algoritmo se dice polinomial si su complejidad es orden  $f(n)$ , con  $f$  acotado superiormente por un polinomio en  $n$ . Un problema se dice *polinomial* si existe un algoritmo polinomial que lo resuelva. Definimos la clase  $\mathcal{P}$  como la familia de problemas polinomiales.

- Clase  $\mathcal{NP}$ : decimos que un problema pertenece a la clase  $\mathcal{NP}$  si existe un algoritmo polinomial que confirma si una salida  $r$  es solución de una instancia  $I$  del problema. Se cumple que  $\mathcal{P} \subseteq \mathcal{NP}$ .
- Clase  $\mathcal{NP}$ -Completo: la clase  $\mathcal{NP}$ -Completo es una subclase de  $\mathcal{NP}$  en la cuál para cada problema existe un algoritmo polinomial que transforma cualquier ejemplo de este en ejemplo de otro problema de la misma clase. Un problema de optimización cuyo problema de decisión asociado es  $\mathcal{NP}$ -Completo, es  $\mathcal{NP}$ -Difícil.

Bajo el orden de complejidad los problemas de la clase  $\mathcal{NP}$ -Difícil son a lo menos los problemas más difíciles de resolver, tal es el caso de los POC considerados en este trabajo (ver Figura 2.1).

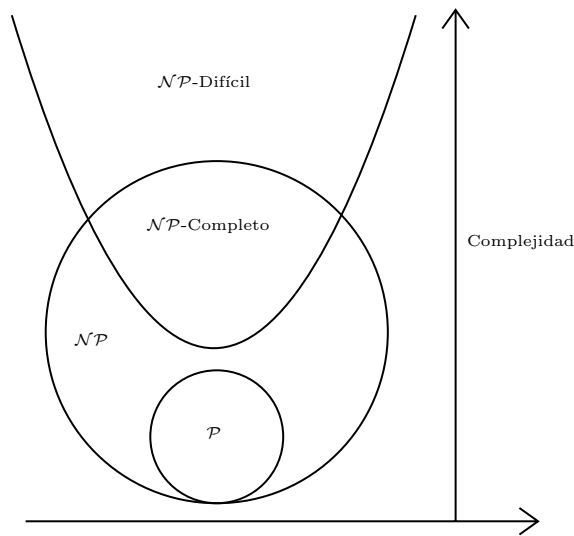


Figura 2.1: La división de las clases de complejidad bajo la hipótesis  $\mathcal{P} \neq \mathcal{NP}$

### 2.2.5. Métodos de solución

Podemos clasificar los métodos de solución para POCs en *métodos exactos* y *métodos aproximados* según ofrezcan garantías de optimalidad (esto es, proporcionar una solución factible que optimice la función objetivo) o no, respectivamente. Por otro lado, dividimos los métodos aproximados en *heurísticas* y *metaheurísticas*.

- **Métodos exactos:** teóricamente los métodos exactos son capaces de proporcionar una solución óptima para POCs, pero en la práctica esto depende mucho de la complejidad del problema. Para problemas de baja complejidad es posible encontrar algoritmos exactos *eficientes* que obtengan soluciones óptimas en un tiempo de ejecución polinomial de grado pequeño. Para problemas de alta

complejidad, antes de evaluar el uso de enfoques no exactos, se intenta plantear una formulación de programación lineal de enteros (ILP) del problema y el uso de solucionadores LP que proporcionan soluciones mediante algoritmos exactos. Sin embargo, debido a la alta complejidad computacional de estos algoritmos el tiempo de ejecución crece exponencialmente con el tamaño de las instancias. De forma que en este último caso el enfoque exacto resulta impráctico para problemas con instancias de tamaño realista. Los principales métodos exactos son la *programación dinámica* y los algoritmos de *búsqueda de árboles*.

- **Métodos de aproximados:** a razón de que la complejidad de la mayoría de problemas que modelan procesos reales es  $\mathcal{NP}$ -Difícil y a lo expuesto en nuestro análisis anterior sobre los métodos exactos, para instancias de gran tamaño en problemas de esta complejidad, se recurre a enfoques aproximados que proporcionan soluciones *buenas* en tiempos razonables pero sin garantía de optimalidad. Este es un enfoque razonable para COPs que modelan procesos en los que se necesita una evaluación rápida como, por ejemplo, los sistemas de tiempo real. El conjunto de estos métodos se pueden dividir en *heurísticas* y *metaheurísticas*.
  - *Heurísticas:* son esquemas algoritmos hechos para COPs concretos. La calidad de sus soluciones depende de los atributos especiales y el conocimiento del problema que puede explotar el algoritmo.
  - *Metaheurísticas:* son esquemas algorítmicos que, a diferencia de los heurísticos, se elaboran independientemente de un COP específico e incluyen fases de mejora o retroceso.

### 2.2.6. Problema de localización de facilidades

Un *problema de localización de facilidades* (FLP) consiste en seleccionar un conjunto de instalaciones maestras (ver Figura 2.2) de un conjunto potencial de instalaciones dispersas geográficamente, para satisfacer la demanda total (real o potencial) de un conjunto de clientes. El FLP en los grafos generales es  $\mathcal{NP}$ -difícil de resolver de manera óptima.

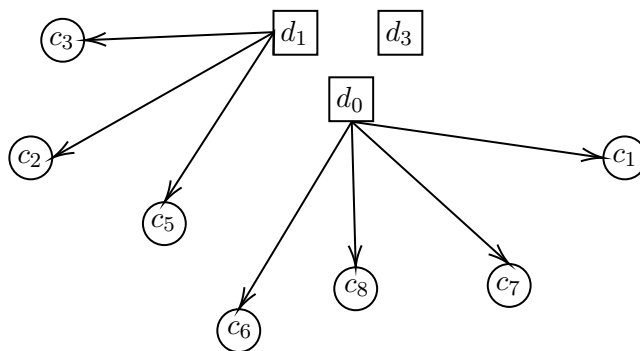


Figura 2.2: Un ejemplo de FLP estándar con tres depósitos potenciales y siete clientes. La solución propone usar sólo los depósitos 0 y 1.

### 2.2.7. Problema de ruteo de vehículos

El problema de ruteo de vehículos (VRP) consiste en encontrar un conjunto de rutas (ver Figura 2.3) a recorrer para un número de vehículos (establecidos o por establecer) a un costo mínimo (distancia recorrida, tiempo recorrido, etc.) comenzando y terminando la ruta en una instalación maestra, de modo que la demanda (real o potencial) de todos los clientes se cumplen, y cada cliente es visitado sólo una vez. El VRP es  $\mathcal{NP}$ -difícil de resolver de manera óptima.

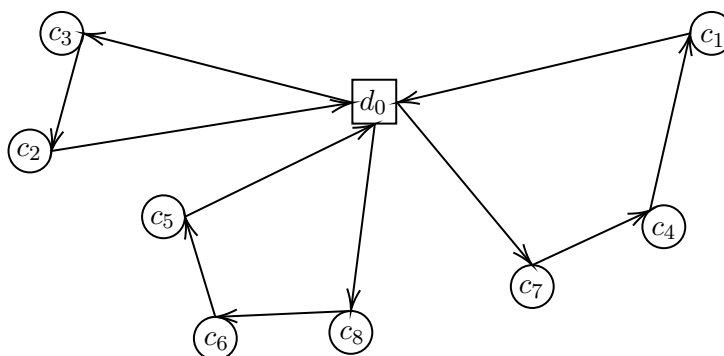


Figura 2.3: Un ejemplo de VRP estándar con un depósito y ocho clientes. La solución propone el uso de tres rutas.

A continuación presentamos una formulación matemática de VRP capacitado (CVRP), la clase más básica de los VRPs.

Formalmente, modelamos el CVRP como un grafo ponderado y no-dirigido  $G = (N, E, f)$ , donde  $N = \{0\} \cup N_1$  es el conjunto de nodos y  $E$  es el conjunto aristas  $(i, j)$ ,  $i \neq j$ , que conectan el vértice  $i \in N$  con el vértice  $j \in N$ . Los subconjuntos  $\{0\}$  y  $N_1$  representan la ubicación de la instalación maestra y las ubicaciones de los clientes. La función  $f$  asocia a cada arista un costo positivo (típicamente la distancia de viaje). En el nivel de la instalación maestra se encuentra disponible una

flota homogénea de vehículos  $V$  con capacidad limitada  $CV$ , usada para transportar carga hacia los clientes. Cada cliente  $c \in N_1$  tiene unas demandas conocidas de entrega  $D_c$  a ser atendidas.

- **Variables de decisión:** definimos la variable binaria  $X_{i,j,v}$ , con valor 1 cuando el vehículo  $v \in V$  recorre la arista  $(i, j) \in E$ , 0 en otro caso. Y la variable binaria  $Z_{iv}$ , con valor 1 cuando el nodo  $i \in N$  es visitado por el vehículo  $v \in V$ , 0 en otro caso.
- **Función objetivo:** la función objetivo (2.4) minimiza los costos de ruteo producidos por el transporte.

$$\text{mín} \quad \sum_{i \in N_1, j \in N_1, v \in V} f_{ij} X_{ijv} \quad (2.4)$$

- **Restricciones:**

$$\sum_{i \in N_1} Z_{0i} = |V|, \quad (2.5)$$

$$\sum_{j \in N_1} X_{ijv} = \sum_{j \in N_1} X_{jiv} = Z_{iv}, \quad \forall i \in N, \quad \forall v \in V, \quad (2.6)$$

$$\sum_{i \in N_1} Z_{iv} = 1 \quad \forall i \in N_1, \quad (2.7)$$

$$\sum_{i \in C, j \in C} X_{ijv} \leq |C| - 1, \quad \forall 2 \leq |C|, C \subseteq N_1 \quad \forall v \in V, \quad (2.8)$$

$$\sum_{i \in N_1} D_i Z_{iv} \leq CV, \quad \forall v \in V, \quad (2.9)$$

$$Z_{iv} \in \{0, 1\} \quad \forall i, j \in N, \quad (2.10)$$

$$X_{ijv} \in \{0, 1\} \quad \forall i, j \in N, \quad \forall v \in V, \quad (2.11)$$

La ecuación (2.5) garantiza que se realicen  $|V|$  rutas. La ecuación (2.6) garantiza que cada vehículo ingresa y sale luego de satisfacer la demanda de un cliente. La ecuación (2.7) garantiza que cada cliente es visitado una única vez. La ecuación (2.8) es la restricción de eliminación del subtours. La ecuación (2.9) garantiza las limitaciones de capacidad de los vehículos. Las ecuaciones (2.10)-(2.11) son las restricciones de valor de las variables de decisión.

### 2.2.8. Problema de localización-ruteo de N niveles

Los costos logísticos constituyen una porción importante del presupuesto que se destina a la gestión de una cadena de suministro. Tales costos pueden disminuirse mediante el diseño inteligente del sistema de distribución subyacente. Eso implica tomar dos decisiones de planificación de distinta dimensión: una decisión estratégica para determinar el número y la ubicación de las instalaciones que componen el sistema, y una decisión táctica-operativa para realizar asignaciones de cliente a instalación y establecer las rutas de transporte. Ambas decisiones plantean problemas de optimización combinatorios  $\mathcal{NP}$ -Difíciles: el FLP y el VRP.

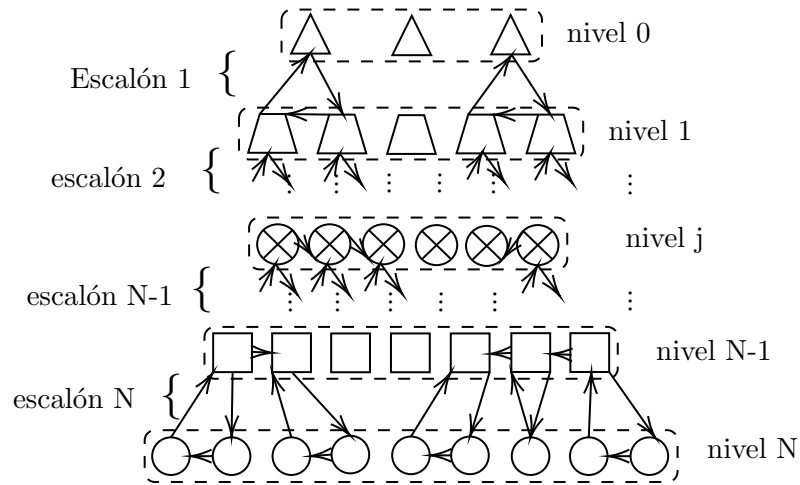


Figura 2.4: Representación de un sistema de distribución de N niveles

Un *sistema de distribución de N niveles* es, en su forma general, una disposición de instalaciones divididas en  $N$  *niveles* o capas según su funcionalidad, a través de las cuales se realizan actividades de transporte y distribución de carga (véase Figura 2.4). Estas actividades inician en una o más *instalaciones maestras* (nivel 0), pasan por varios niveles intermedios formados por *instalaciones satélites* que sirven para consolidar, transbordar o fusionar carga, y llegan hasta las *instalaciones locales* (nivel  $N-1$ ) desde las cuales finalmente se satisfacen las demandas de los clientes (nivel  $N$ ). El flujo de carga entre dos niveles consecutivos se denomina *escalón*; la carga debe transportarse a través de los  $N$  escalones definidos por el sistema. Además, para cada escalón existe una flota de vehículos que transporta carga entre los niveles que lo definen.

Varios trabajos han demostrado que una planificación combinada de las ubicaciones de las instalaciones y las rutas de distribución, produce constantemente soluciones de mayor calidad que una planificación secuencial (Salhi and Rand, 1989; Salhi and Nagy, 1999). Esto ha motivado varios estudios que abordan el diseño óptimo de sistemas de distribución considerando ambas decisiones de manera integral

(Prodhon and Prins, 2014; Drexl and Schneider, 2015). En concreto, dado un sistema de distribución, se define el *problema de localización-ruteo de  $N$  niveles* (nE-LRP) como la clase de problemas que estudian cómo planificar de forma integral todos los escalones del sistema tomando las decisiones de localización y ruteo de manera combinada a fin de reducir los costos logísticos –y/o de otra naturaleza– producidos por la distribución de carga.

## 2.3. Hipótesis

Se postula que utilizando un modelo integrado de localización y ruteo se pueden optimizar rutas en dos niveles como una aplicación en la logística urbana.

### 2.3.1. Variables

- Variable dependiente: sistema de rutas de distribución de dos niveles.
- Variable independiente: Modelo integrado de localización y ruteo

## Capítulo 3

# Marco metodológico

En la presente investigación se estudió el diseño de un modelo lineal entero puro para un problema de localización de instalaciones y programación de rutas.

### 3.1. Enfoque y diseño

El trabajo se desarrolló bajo un enfoque de investigación de análisis prescriptivo que presenta un enfoque mixto de diseño experimental.

### 3.2. Métodos y procedimientos

#### 3.2.1. Metodología

En la investigación se utilizó el método de programación matemática correspondiente a la investigación de operaciones para modelar el problema. Para nuestro propósito el diseño del modelo se dividió en tres fases: fase de precisión, fase de modelado, y fase de validación-evaluación.

**Fase de precisión.** En esta fase se definen los constructos o elementos del problema a modelar:

- El problema de localización-ruteo de dos niveles,
- El servicio de entrega y colecta simultáneas,
- Las consideraciones ambientales.

**Fase de modelado.** En esta fase se formula el modelo matemático:

- Definición matemática del problema,
- Parámetros y variables del modelo,

- Función objetivo del modelo,
- Restricciones del modelo,
- Refinamiento del modelo,
- Implementación del modelo,
- Desarrollo de un generador de instancias,
- Implementación del generador.

**Fase de validación y evaluación.** En esta fase se valida y evalúa el modelo:

- Definición de los criterios de validación,
- Evaluación del modelo,
- Discusión de los resultados.

Los resultados de esta fase se presentan y discuten en el Capítulo 4 siguiendo la estructura de tesis correspondiente al reglamento de la Universidad Nacional de Piura.

### 3.2.2. Equipos y herramientas computacionales

Los experimentos computacionales se ejecutaron en un computador Core (TM) i7 de 3.20 GHz, con 6 procesadores principales, 12 procesadores lógicos, y 16 GB de RAM en un entorno Windows 10. Para la implementación del modelo se usó el paquete libre GLPK (GNU Linear Programming Kit). Para el desarrollo e implementación del generador de instancias se usó el lenguaje de programación Python; para la descarga y manipulación de datos geoespaciales se usó el paquete OSMnx de Python.

## 3.3. Fase de precisión

Siendo el objetivo principal de esta tesis la optimización de rutas en dos niveles mediante un modelo de localización y ruteo en el contexto de la logística urbana, consideramos el problema de optimización asociado de mayor generalidad, el problema de localización-ruteo de dos niveles para entrega y colecta simultáneas con consideraciones ambientales (2E-LRPSPD-E). Los elementos principales de este problema son: el problema de localización-ruteo de dos niveles (2E-LRP), el servicio de entrega y colecta simultáneas (SPD) y las consideraciones ambientales (E).

### 3.3.1. 2E-LRP

En la última década se ha prestado especial atención al *problema de localización-ruteo de dos niveles* (2E-LRP) por las diversas aplicaciones de su estructura teórica (Romeijn et al., 2007; Cuda et al., 2015; Guastaroba et al., 2016). Estas incluyen modelar sistemas reales como el transporte multimodal, distribución de productos de supermercado e hipermercado, distribución de periódicos y paquetería, servicios de comercio electrónico y entrega a domicilio, entre otros (Gonzalez-Feliu et al., 2008).

En la literatura se destaca su aplicación en la logística urbana por los beneficios a nivel económico, social y medioambiental: las instalaciones satélites que conforman el nivel intermedio del sistema se ubican en puntos periféricos de la ciudad, desde donde se distribuye la carga hacia los clientes usando flotas de vehículos pequeños y ecológicos que permiten reducir el ruido, el tráfico y la contaminación (Soysal et al., 2015).

Desde el punto de vista teórico el 2E-LRP es una doble extensión por *integración* del VRP (véase Figura 3.1). Por lo tanto, pertenece a la clase de problemas  $\mathcal{NP}$ -Difíciles. Además, hereda la propiedad de que el número de problemas variantes que resultan de considerar múltiples atributos crece exponencialmente (Vidal et al., 2013). Esto apertura un campo amplio de investigaciones sobre modelos que consideran múltiples y nuevos atributos de sistemas reales.

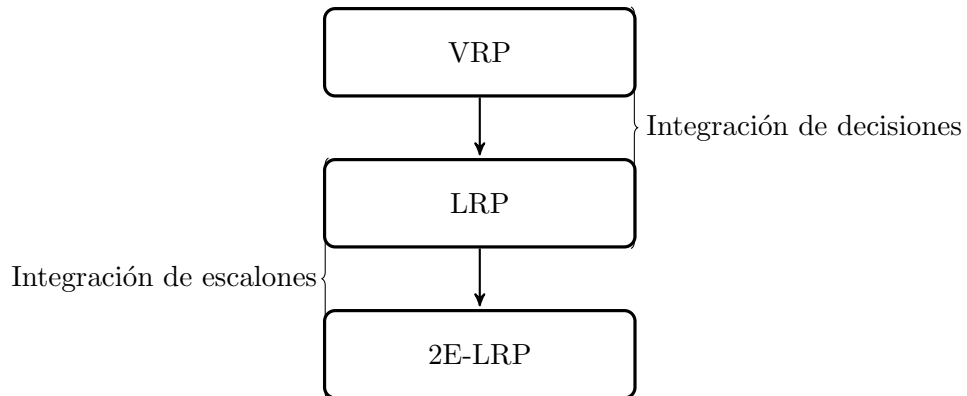


Figura 3.1: Genealogía del 2E-LRP

El 2E-LRP capacitado (2E-CLRP) es la clase más básica de los 2E-LRPs, de modo que, cualquier otra clase se obtiene por añadidura de atributos a alguna de sus formulaciones clásicas. La mayoría de problemas de la literatura considerada en este trabajo se compone especificando el nivel de la localización, el tipo de operación sobre la carga y la consideración de algunos atributos. La descripción de nuestro problema sigue dicha línea de composición.

### 3.3.2. Entrega y colecta simultáneas

En un problema clásico de rutas existen clientes que demandan la entrega ó clientes que demanda la colecta de carga. Sin embargo, en un contexto real, los clientes pueden demandar –no necesariamente a la vez– ambos tipos de servicios. Para nuestro problema consideramos el servicio de *entrega y colecta simultáneas* (SPD), en el cual todos los clientes demandan la entrega y colecta de carga en una única visita. Las aplicaciones de este servicio se pueden encontrar en varios contextos de la logística inversa (Demircan-Yildiz et al., 2016).

### 3.3.3. Consideraciones ambientales

- **Ubicaciones periféricas para instalaciones satélites:** una consideración ambiental no solo debe estar dirigida a minimizar el impacto negativo sobre el medio ambiente, sino también a evitarlo. En ese sentido, considerar únicamente puntos periféricos de la ciudad como potenciales ubicaciones de las instalaciones satélites, es implícitamente una consideración ambiental. En efecto, esta decisión estratégica sobre la disposición estructural del sistema impide el transito de grandes vehículos dentro de las zonas urbanas evitando las externalidades que eso conlleva.

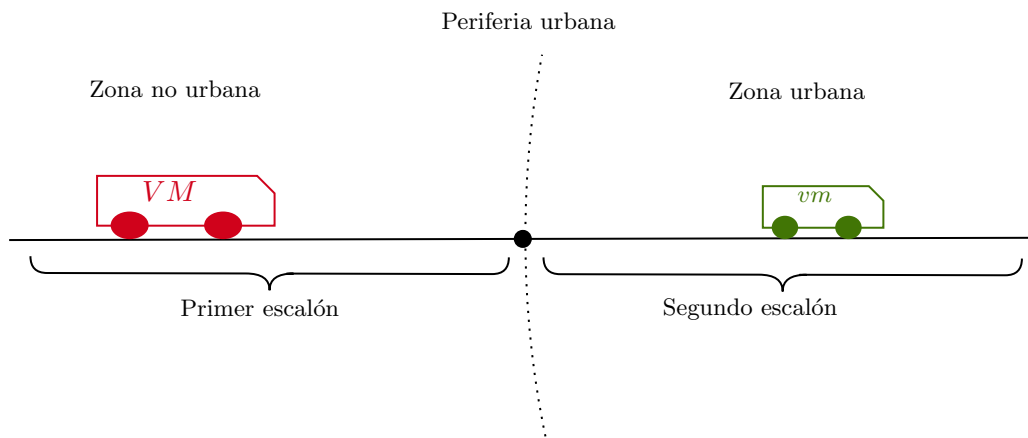


Figura 3.2: Ubicaciones periféricas de las intalaciones satélite

- **Consumo de combustible y emisión de CO<sub>2</sub>:** la cantidad de CO<sub>2</sub> emitida por un vehículo es proporcional a la cantidad de combustible consumido (Fontaras et al., 2017). Por tanto, minimizar las emisiones de CO<sub>2</sub> equivale a minimizar el consumo de combustible. En la literatura se observa que el consumo de combustible depende de múltiples factores relativos al vehículo, estado de las rutas y políticas de conducción: la masa del vehículo (Aronsson and Brodin, 2006), la velocidad y aceleración del vehículo (Bickel et al., 2006), tecnología del vehículo (Liu and Helfand, 2009) y las condiciones ambientales

y del tráfico (Fontaras et al., 2017).

Si bien es deseable considerar modelos que integren un mayor número de factores con el fin de obtener estimaciones más realista del consumo de combustible, para la mayoría de estos factores se dispone de estimaciones de baja precisión y el cálculo de estimaciones más precisas requiere un tiempo considerable. Estos modelos son poco viables para problemas para los que es mejor –y/o suficiente– tener una buena solución en un tiempo corto.

### 3.4. Fase de Modelado

#### 3.4.1. Descripción del problema

En el 2E-LRPSPD-E se consideran dos niveles. El primer escalón está conformado por el flujo de carga desde la instalación maestra ya localizada hasta un conjunto instalaciones satélites que deben aperturarse, y el segundo, por el flujo de carga desde las instalaciones satélites aperturadas hasta los clientes.

- **Configuración del problema**

Formalmente, modelamos el 2E-LRPSPD-E como un grafo ponderado y no-dirigido  $G = (N, E, f)$ , donde  $N = N_0 \cup N_1 \cup N_2$  es el conjunto de nodos y  $E$  es el conjunto aristas  $(i, j)$ ,  $i \neq j$ , que conectan el vértice  $i \in N$  con el vértice  $j \in N$ . Los subconjuntos  $N_0$ ,  $N_1$  y  $N_2$  representan la ubicación de la instalación maestra, las ubicaciones potenciales de las instalaciones satélites y las ubicaciones de los clientes, respectivamente. La función  $f$  asocia a cada arista un costo positivo (típicamente la distancia de viaje). Cada potencial instalación satélite  $s \in N_1$  tiene un costo de apertura  $a_s$  y una capacidad  $w_s$ . En el nivel de la instalación maestra se encuentra disponible una flota homogénea de vehículos  $V_1$  con capacidad limitada  $CV_1$ , costo fijo  $FV_1$  de ruteo, costo variable  $k_1$  de ruteo y costo de combustible unitario  $K_1$ , usada para transportar carga hacia los satélites. En el nivel de instalaciones satélites se encuentra disponible otra flota homogénea de vehículos  $V_2$  con capacidad limitada  $CV_2$ , costo fijo  $FV_2$  de ruteo, costo variable  $k_2$  de ruteo y costo de combustible unitario  $K_1$ , usada para transportar carga hacia los clientes. Cada cliente  $c \in N_2$  tiene unas demandas conocidas de entrega  $D_c$  y una de colecta  $P_c$  ser atendidas en simultáneo.

- **Objetivo del problema**

Solucionar el 2E-LRPSPD-E consiste en determinar: (i) la ubicación de las instalaciones satélites, (ii) la asignación de instalaciones satélites a instalaciones maestras, y de clientes a instalaciones satélites, y (iii) la construcción de rutas en el primer y segundo escalón, bajo los siguientes supuestos:

1. Las instalaciones satélites y clientes son atendidos una única vez.
2. Los vehículos regresan a su instalación de partida luego de completar sus tareas de transporte.
3. La carga total de vehículos de ambos escalones en cualquier nodo de una ruta no excede su capacidad.
4. La demandas acumuladas de entrega y colecta asignadas a una instalación satélite no exceden su capacidad.
5. Se minimizan el costo de apertura de las instalaciones satélites, el costo de uso de los vehículos, el costo del consumo de combustible y la suma de los costos de las aristas que estos atraviesan.

### 3.4.2. Formulación del modelo

Formulamos el 2E-LRPSPD-E como un modelo de programación lineal de enteros puro (PILP) de tres índices, basado en [Demircan-Yildiz et al. \(2016\)](#) y [Fan et al. \(2020\)](#). A continuación presentamos el modelo seleccionado para estimación del consumo de combustible, y los parámetros, variables, función objetivo y restricciones del modelo formulado.

#### ■ Modelo para el consumo de combustible

Para la estimación del consumo de combustible usamos el modelo propuesto por [Xiao et al. \(2012\)](#), según el cual, el consumo de combustible por unidad de distancia al transportar  $U_1$  (kg) de carga, se calcula como

$$\rho(U_1) = \rho_0 + \frac{\rho^* - \rho_0}{CV} U_1$$

Dónde  $CV$  es la capacidad máxima del vehículo, y  $\rho_0$  (l/km),  $\rho^*$  (l/km), indican la tasa de consumo de combustible con carga plena y sin carga, respectivamente.

En general, el consumo de combustible utilizado para transportar  $U_1$  (kg) de carga entre dos puntos  $i$  y  $j$ , se calcula como

$$\rho_{ij}(U_1) = f_{ij}(\rho_0 + \alpha U_1)$$

dónde  $f_{ij}$  es la distancia que el vehículo recorre desde  $i$  hasta  $j$ , y  $\alpha = \frac{\rho^* - \rho_0}{CV}$  es la tasa media de consumo de combustible con carga.

#### ■ Parámetros y variables

Tabla 3.1: Parámetros y variables utilizadas en nuestro modelo matemático del 2E-LRPSPD-E

| Conjuntos             | Descripción   |
|-----------------------|---|
| $N$                   | Conjunto de nodos, $N = N_0 \cup N_1 \cup N_2$  |
| $N_0$                 | Ubicación de la instalación maestra (nodo 0)  |
| $N_1$                 | Ubicaciones potenciales de las instalaciones satélites  |
| $N_2$                 | Ubicación de los Clientes   |
| $V$                   | Conjuntos de vehículos, $V = V_1 \cup V_2$  |
| $V_1$                 | Flota de vehículos para el primer escalón   |
| $V_2$                 | Flota de vehículos para el segundo escalón  |
| $E$                   | Conjunto de aristas válidas, $E = E_1 \cup E_2$   |
| $E_1$                 | Subconjunto de aristas válidas para el primer escalón, $E_1 = \{(i, j) : i, j \in N_0 \cup N_1, i \neq j\}$         |
| $E_2$                 | Subconjunto de aristas válidas para el segundo escalón, $E_2 = \{(s, c) : p, q \in N_1 \cup N_2, p \neq q\}$        |
| Índices               | Descripción   |
| $i, j$                | Nodo en el primer escalón, $i, j \in N_0 \cup N_1, i \neq j$  |
| $s, c$                | Nodo en el segundo escalón, $p, q \in N_1 \cup N_2, p \neq q$   |
| $v$                   | vehículo $v \in V$  |
| Parámetros            | Descripción   |
| $f_{ij}$              | Distancia de viaje de la arista $(i, j)$  |
| $CV_1$                | Capacidad de los vehículos de la flota del primer escalón   |
| $CV_2$                | Capacidad de los vehículos de la flota del segundo escalón  |
| $FV_1$ y $FV_2$       | Costos fijos de ruteo para vehículos del primer y segundo escalón   |
| $k_1$ y $k_2$         | Costos variables de ruteo para vehículos del primer y segundo escalón   |
| $W_s$                 | Capacidad del satélite $s \in N_1$  |
| $A_s$                 | Costo de apertura del satélite $s \in N_1$  |
| $D_c$                 | Demanda de entrega del cliente $c \in N_2$  |
| $P_c$                 | Demanda de colecta del cliente $c \in N_2$  |
| $\rho^*, \rho_0$      | Consumo de combustible para un vehículos del primer escalón con y sin carga, respectivamente                        |
| $\mu^*, \mu_0$        | Consumo de combustible para un vehículos del segundo escalón con y sin carga, respectivamente                       |
| $K_1$ y $K_2$         | Costos unitarios del combustible en el primer y segundo escalón   |
| Variables adicionales | Descripción   |
| $U_{ijv}$             | Carga dinámica de $i$ a $j$ usando el vehículo $v \in V_1$ del primer escalón                                       |
| $U_{scv}$             | Carga dinámica de $s$ a $c$ usando el vehículo $v \in V_2$ del segundo escalón                                      |
| $D_s$                 | Demanda de entrega para el satélite aperturado $s \in N_1$  |
| $P_s$                 | Demanda de colecta para el satélite aperturado $s \in N_1$  |
| Variables de decisión | Descripción   |
| $X_{ijv}$             | Variable binaria con valor 1 cuando el vehículo $v \in V_1$ recorre la arista $(i, j) \in E_1$ ,<br>0 en otro caso. |
| $Y_{scv}$             | Variable binaria con valor 1 cuando el vehículo $v \in V_2$ recorre la arista $(s, c) \in E_2$ ,<br>0 en otro caso. |
| $Z_{sc}$              | Variable binaria con valor 1 cuando el cliente $c \in N_2$ es asignado al satélite $s \in N_1$ ,<br>0 en otro caso. |
| $L_s$                 | Variable binaria con valor 1 cuando el satélite $s \in N_2$ se apertura,<br>0 en otro caso.                         |

■ **Función objetivo:**

$$\text{mín Costo} = \sum_{s \in N_1} A_s L_s + \sum_{i \in N_0, j \in N_1, v \in V_1} FV_1 X_{ijv} + \sum_{s \in N_1, c \in N_2, v \in V_2} FV_2 Y_{scv} +$$

$$\begin{aligned}
& \sum_{i,j \in E_1, v \in V_1} k_1 f_{ij} X_{ijv} + \sum_{s,c \in E_2, v \in V_2} k_2 f_{sc} Y_{scv} + \\
& \sum_{i \in N_0, j \in N_1, v \in V_1} K_1 f_{ij} (\rho_0 X_{ijv} + \alpha_1 U_{ijv}) + \sum_{s \in N_1, c \in N_2, v \in V_2} K_2 f_{sc} (\mu_0 Y_{scv} + \alpha_2 U_{scv})
\end{aligned} \tag{3.1}$$

■ **Restricciones:**

$$\sum_{i \in E_1, v \in V_1} X_{ijv} = L_j, \quad \forall j \in N_1 \tag{3.2}$$

$$\sum_{i \in E_1, v \in V_1} X_{ijv} \leq 1, \quad \forall j \in N_1 \tag{3.3}$$

$$\sum_{s \in E_2, v \in V_2} Y_{scv} = 1, \quad \forall c \in N_2 \tag{3.4}$$

$$\sum_{i \in E_1} X_{ijv} = \sum_{i \in E_1} X_{jiv}, \quad \forall j \in E_1, \quad \forall v \in V_1, \tag{3.5}$$

$$\sum_{s \in E_2} Y_{scv} = \sum_{s \in E_2} Y_{csv}, \quad \forall c \in E_2, \quad \forall v \in V_2, \tag{3.6}$$

$$\sum_{i \in N_0, j \in N_1} X_{ijv} \leq 1, \quad \forall v \in V_1, \tag{3.7}$$

$$\sum_{s \in N_1, c \in N_2} Y_{scv} \leq 1, \quad \forall v \in V_2, \tag{3.8}$$

$$\sum_{i,j \in S} X_{ijv} \leq |S| - 1, \quad \forall S \subseteq N_1, \quad \forall v \in V_1, \tag{3.9}$$

$$\sum_{s,c \in C} Y_{scv} \leq |C| - 1, \quad \forall C \subseteq N_2, \quad \forall v \in V_2, \tag{3.10}$$

$$\sum_{s,c \in N_1} Y_{scv} = 0, \quad \forall v \in V_2, \tag{3.11}$$

$$\sum_{t \in E_2} Y_{stv} + \sum_{t \in E_2} Y_{ctv} \leq 1 + Z_{sc}, \quad \forall s \in N_1, \quad \forall c \in N_2, \quad \forall v \in V_2, \tag{3.12}$$

$$D_s = \sum_{c \in N_2} Z_{sc} D_c, \quad \forall s \in N_1, \quad (3.13)$$

$$P_s = \sum_{c \in N_2} Z_{sc} P_c, \quad \forall s \in N_1, \quad (3.14)$$

$$\max\{D_s, P_s\} \leq W_s L_s, \quad \forall s \in N_1, \quad (3.15)$$

$$\sum_{i \in N_0, j \in N_1} U_{ijv} = \sum_{i \in E_1, j \in N_1} X_{ijv} D_j, \quad \forall v \in V_1, \quad (3.16)$$

$$\sum_{i \in N_0, j \in N_1} U_{jiv} = \sum_{i \in E_1, j \in N_1} X_{ijv} P_j, \quad \forall v \in V_1, \quad (3.17)$$

$$\sum_{s \in N_1, c \in N_2} U_{scv} = \sum_{s \in E_2, c \in N_2} Y_{scv} D_c, \quad \forall v \in V_2, \quad (3.18)$$

$$\sum_{s \in N_1, c \in N_2} U_{csv} = \sum_{s \in E_2, c \in N_2} Y_{scv} P_c, \quad \forall v \in V_2, \quad (3.19)$$

$$\sum_{i \in E_1, v \in V_1} U_{ijv} - D_j = \sum_{i \in E_1, v \in V_1} U_{jiv} - P_j, \quad \forall j \in N_1, \quad (3.20)$$

$$\sum_{s \in E_2, v \in V_2} U_{scv} - D_c = \sum_{s \in E_2, v \in V_2} U_{csv} - P_c, \quad \forall c \in N_2, \quad (3.21)$$

$$0 \leq U_{ijv} \leq CV_1 X_{ijv}, \quad \forall i, j \in E_1, \quad \forall v \in V_1, \quad (3.22)$$

$$0 \leq U_{scv} \leq CV_2 Y_{scv}, \quad \forall s, c \in E_2, \quad \forall v \in V_2, \quad (3.23)$$

$$\sum_{c \in N_2} Z_{sc} \geq L_s, \quad \forall s \in N_1, \quad (3.24)$$

$$Z_{sc} \leq L_s, \quad \forall s \in N_1, \quad \forall c \in N_2, \quad (3.25)$$

$$\sum_{c \in N_2, v \in V_2} Y_{scv} \geq L_s, \quad \forall s \in N_1, \quad (3.26)$$

$$\sum_{c \in N_2} Y_{scv} \leq L_s, \quad \forall s \in N_1, v \in V_1, \quad (3.27)$$

$$L_s \in \{0, 1\}, Z_{sc} \in \{0, 1\}, \quad \forall s \in N_1, c \in N_2, \quad (3.28)$$

$$X_{ijv} \in \{0, 1\}, \quad \forall i, j \in E_1, v \in V_1, \quad (3.29)$$

$$Y_{scv} \in \{0, 1\}, \quad \forall s, c \in E_2, v \in V_2. \quad (3.30)$$

La función (3.1) representa la función objetivo. Se minimizan los costos de apertura, los costos fijos por el uso de vehículos, los costos de ruteo y los costos del consumo de combustible. La ecuación (3.3) se utiliza para garantizar que cada instalación satélite aperturada en el primer escalón sea servido por exactamente un vehículo del primer escalón. De manera similar, la ecuación (3.4) se utiliza para garantizar que cada cliente en el segundo escalón sea atendido por exactamente un vehículo del segundo escalón. Las ecuaciones (3.5) y (3.6) son las restricciones de equilibrio de flujo. Las ecuaciones (3.7) y (3.8) se utilizan para garantizar que los vehículos del primer y segundo escalón solo puedan realizar al menos un viaje y partan de un solo nodo, respectivamente. Las ecuaciones (3.9) y (3.10) son las restricciones de eliminación de los subtours. La ecuación (3.11) se utiliza para garantizar que dos instalaciones satélites cualesquiera del segundo escalón no están conectados. La ecuación (3.12) se utiliza para asegurar que los clientes asignados de cada instalación satélite en el primer escalón sean visitados por un vehículo del segundo escalón. Las ecuaciones (3.13) y (3.14) se utilizan para determinar las demandas de entrega y colecta de las instalaciones satélites. La ecuación (3.15) es la restricción de capacidad de las instalaciones satélites. Las demandas totales de entrega y colecta de todos los clientes asignados a una instalación satélite no pueden exceder su capacidad. Las ecuaciones (3.16) y (3.17) representan las cargas del vehículo en el primer escalón cuando sale y regresa de la instalación maestra, respectivamente. Las ecuaciones (3.18) y (3.19) representan las cargas del vehículo en el segundo escalón cuando sale del satélite y regresa a su instalación satélite, respectivamente. Las ecuaciones (3.20) y (3.21) representan los cambios en la carga dinámica del vehículo en el primer y segundo escalón, respectivamente. Las ecuaciones (3.22) y (3.23) son las limitaciones de capacidad de los vehículos en el primer y segundo escalón, respectivamente. Las ecuaciones (3.24) y (3.25) se utilizan para garantizar que cada cliente deba asignarse a una instalación satélite aperturada. Las ecuaciones (3.26) y (3.27) se utilizan para garantizar que los vehículos del segundo escalón sólo se asignen a instalaciones satélites aperturadas. Las ecuaciones (3.28) - (3.30) son las restricciones de integridad de las variables de decisión.

### 3.4.3. Refinamiento del modelo

- **Linealización de restricciones**

Para completar nuestra formulación del 2E-LRPSPD-E como un modelo de PILP se linealizan las restricciones 3.15, 3.16 y 3.17.

- Para linealizar la restricción 3.15 se sustituye por las siguientes restricciones:

$$D_s \leq W_s L_s, \quad \forall s \in N_1, \quad (3.31)$$

$$P_s \leq W_s L_s, \quad \forall s \in N_1, \quad (3.32)$$

- Para linealizar la restricción 3.16 se introduce la variable entera auxiliar  $X_{i,j,v}^* = X_{i,j,v} D_j$  y 3.16 se sustituye por las siguientes restricciones:

$$\sum_{i \in N_0, j \in N_1} U_{ijv} = \sum_{i \in E_1, j \in N_1} X_{i,j,v}^*, \quad \forall v \in V_1, \quad (3.33)$$

$$X_{i,j,v}^* \leq W_j X_{i,j,v}, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.34)$$

$$X_{i,j,v}^* \leq D_j, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.35)$$

$$X_{i,j,v}^* \geq D_j - W_j(1 - X_{i,j,v}), \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.36)$$

$$X_{i,j,v}^* \leq W_j, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.37)$$

- Para linealizar la restricción 3.17 se introduce la variable entera auxiliar  $X_{i,j,v}^{**} = X_{i,j,v} P_j$  y 3.17 se sustituye por las siguientes restricciones:

$$\sum_{i \in N_0, j \in N_1} U_{jiv} = \sum_{i \in E_1, j \in N_1} X_{i,j,v}^{**}, \quad \forall v \in V_1, \quad (3.38)$$

$$X_{i,j,v}^{**} \leq W_j X_{i,j,v}, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.39)$$

$$X_{i,j,v}^{**} \leq P_j, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.40)$$

$$X_{i,j,v}^{**} \geq P_j - W_j(1 - X_{i,j,v}), \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.41)$$

$$X_{i,j,v}^{**} \leq W_j, \quad \forall i \in E_1, j \in N_1, v \in V_1 \quad (3.42)$$

- **Redundancia de restricciones**

Se observa que la restricción 3.3 es perfectamente suprimible de nuestro modelo, puesto que se puede implicar de las restricciones 3.2 y 3.28.

#### 3.4.4. Implementación del modelo

El modelo se implementó en el paquete libre GLPK (GNU Linear Programming Kit) y se ejecutó en computador Core (TM) i7 de 3.20 GHz, con 6 procesadores principales, 12 procesadores lógicos, y 16 GB de RAM en un entorno Windows 10.

```
## Parámetros y variables
```

```
## Conjuntos
```

```
#Niveles
```

```
param n1; #Número de instalaciones satélites potenciales
```

```
param n2; #Número de clientes
```

```
set N0 := {0}; #Instalación maestra
```

```
set N1 := {1..n1}; #Instalaciones satelites potenciales
```

```
set N2 := {(n1+1)..(n1+n2)}; #Clientes
```

```
set N:= {0..(n1+n2)}; #Total de elementos
```

```
#Escalones
```

```
set E1:= N0 union N1; #Primer escalón
```

```
set E2:= N1 union N2; #Segundo escalón
```

```
#Flotas de vehículos
```

```
param v1;
```

```
param v2;
```

```
set V1 := {1..v1}; #vehículos del primer escalón
```

```
set V2 := {(v1+1)..(v1+v2)}; #vehículos del segundo escalón
```

```
## -- Parametros --
```

```
param f{i in N, j in N}; #Distancia de viaje
```

```
param CV1; #Capacidad de los vehículos del primer escalón
```

```
param CV2; #Capacidad de los vehículos del segundo escalón
```

```
param FV1; #Costo fijo de ruteo para vehículos del primer escalón
```

```
param FV2; #Costo fijo de ruteo para vehículos del segundo escalón
```

```
param k1; #Costo variable de ruteo para vehiculos del primer escalón
```

```
param k2; #Costo variable de ruteo para vehiculos del segundo escalón
```

```
param W{s in N1}; #Capacidad del satélite s
```

```

param A{s in N1}; #Costo de apertura del satélite s
param D{c in N2}; #Demanda de entrega del cliente c
param P{c in N2}; #Demanda de colecta del cliente c
param rho0; #Consumo de combustible sin carga para vehículos
del primer escalón

param rho1; #Consumo de combustible con carga para vehículos
del primer escalón

param alpha1:=(rho1-rho0)/CV1; #Consumo medio de combustible
con carga para vehículos del primer escalón

param mu0; #Consumo de combustible sin carga para vehículos
del primer escalón

param mu1; #Consumo de combustible con carga para vehículos
del segundo escalón

param alpha2:=(mu1-mu0)/CV2; #Consumo de combustible con carga
para vehículos del segundo escalón

param K1; #Costo unitario del combustible en el primer escalón

param K2; #Costo unitario del combustible en el segundo escalón

##Variables adicionales
var U1{i in E1, j in E1, v in V1: i<>j} >= 0; #Carga dinámica de
i a j usando el vehículo V1 del primer escalón

var U2{s in E2, c in E2, v in V2: s<>c} >= 0; #Carga dinámica de s a c
usando el vehículo V2 del segundo escalón

var DD{s in N1} >= 0; #Demanda de entrega para el satélite aperturado
var PP{s in N1} >= 0; #Demanda de colecta para el satélite aperturado

#Variable de linealización
var XD{i in E1, j in E1, v in V1: i<>j} >=0;
var XP{i in E1, j in E1, v in V1: i<>j} >=0;

## -- Variables de decisión --

```

```

var X{i in E1, j in E1, v in V1: i<>j}, binary;
var Y{s in E2, c in E2, v in V2: s<>c}, binary;
var Z{s in N1, c in N2}, binary;
var L{s in N1}, binary;

## -- Función Objetivo --

minimize Costo:
#Costos de apertura, uso de vehículos y de ruteo
sum{s in N1} A[s]*L[s]+
sum{i in N0, j in N1, v in V1: i<>j} FV1*X[i,j,v] +
sum{s in N1, c in N2, v in V2} FV2*Y[s,c,v]+
sum{i in E1,j in E1, v in V1: i<>j} k1*f[i,j]*X[i,j,v] +
sum{s in E2,c in E2, v in V2: s<>c} k2*f[s,c]*Y[s,c,v] +

#Costo del consumo de combustible
sum{i in N0, j in N1, v in V1: i<>j} K1*f[i,j]*(rho0*X[i,j,v] +
alpha1*U1[i,j,v]) +
sum{s in N1, c in N2, v in V2:s!=c} K2*f[s,c]*(mu0*Y[s,c,v] +
alpha2*U2[s,c,v]);

## -- Restricciones --

s.t. R1 {j in N1}: sum{i in E1, v in V1: i<>j} X[i,j,v] = L[j];

s.t. R2{j in N1}: sum{i in E1, v in V1: i<>j} X[i,j,v] <= 1;

s.t. R3{c in N2}: sum{s in E2, v in V2: s<>c} Y[s,c,v] = 1;

s.t. R4{j in E1, v in V1}: sum{i in E1: i<>j} X[i,j,v]=
sum{i in E1: i<>j} X[j,i,v];

s.t. R5{c in E2, v in V2}: sum{s in E2: s<>c} Y[s,c,v]=
sum{s in E2: s<>c} Y[c,s,v];

s.t. R6{v in V1}: sum{i in N0, j in N1} X[i,j,v] <= 1;

s.t. R7{v in V2}: sum{s in N1, c in N2} Y[s,c,v] <= 1;

s.t. R8{S in N1, v in V1}: sum{i in 1..S,j in 1..S: i<>j}

```

$X[i,j,v] \leq S-1;$

s.t.  $R9\{C \text{ in } N2, v \text{ in } V2\}: \sum\{s \text{ in } (n1+1)..C, c \text{ in } (n1+1)..C: s \langle c\} Y[s,c,v] \leq C-n1-1;$

s.t.  $R10\{v \text{ in } V2\}: \sum\{s \text{ in } N1, c \text{ in } N1: s \langle c\} Y[s,c,v] = 0;$

s.t.  $R11\{s \text{ in } N1, c \text{ in } N2, v \text{ in } V2\}: \sum\{t \text{ in } E2: s \langle t\} Y[s,t,v] + \sum\{t \text{ in } E2: c \langle t\} Y[c,t,v] \leq 1 + Z[s,c];$

s.t.  $R12\{s \text{ in } N1\}: DD[s] = \sum\{c \text{ in } N2\} Z[s,c]*D[c];$

s.t.  $R13\{s \text{ in } N1\}: PP[s] = \sum\{c \text{ in } N2\} Z[s,c]*P[c];$

#s.t.  $R14\{s \text{ in } N1\}: \max(DD[s], PP[s]) \leq W[s]*L[s];$

s.t.  $R14a\{s \text{ in } N1\}: DD[s] \leq W[s]*L[s];$

s.t.  $R14b\{s \text{ in } N1\}: PP[s] \leq W[s]*L[s];$

#s.t.  $R15\{v \text{ in } V1\}: \sum\{i \text{ in } N0, j \text{ in } N1: i \langle j\} U1[i,j,v] = \sum\{i \text{ in } E1, j \text{ in } N1: i \langle j\} X[i,j,v]*DD[j];$

#  $DD[j]*X[i,j,v] = XD [i,j,v]$  (variable auxiliar de linealización)

s.t.  $R15\{v \text{ in } V1\}: \sum\{i \text{ in } N0, j \text{ in } N1 : i \langle j\} U1[i,j,v] = \sum\{i \text{ in } E1, j \text{ in } N1 : i \langle j\} XD[i,j,v];$

s.t.  $R15a \{i \text{ in } E1, j \text{ in } N1, v \text{ in } V1: i \langle j\}: XD[i,j,v] \leq W[j]*X[i,j,v];$

s.t.  $R15b \{i \text{ in } E1, j \text{ in } N1, v \text{ in } V1 : i \langle j\}: XD[i,j,v] \leq DD[j];$

s.t.  $R15c \{i \text{ in } E1, j \text{ in } N1, v \text{ in } V1 : i \langle j\}: XD[i,j,v] \geq DD[j]-W[j]*(1-X[i,j,v]);$

s.t.  $R15d \{i \text{ in } E1, j \text{ in } N1, v \text{ in } V1 : i \langle j\}: XD[i,j,v] \leq W[j];$

#s.t.  $R16\{v \text{ in } V1\}: \sum\{i \text{ in } N0, j \text{ in } N1: i \langle j\} U1[j,i,v] = \sum\{i \text{ in } E1, j \text{ in } N1: i \langle j\} PP[j]*X[i,j,v];$

#  $PP[j]*X[i,j,v] = XP [i,j,v]$  (variable auxiliar de linealización)

s.t. R16{v in V1}:  $\sum\{i \text{ in } N0, j \text{ in } N1: i \langle \rangle j\} U1[j,i,v] = \sum\{i \text{ in } E1, j \text{ in } N1: i \langle \rangle j\} XP[i,j,v]$ ;

s.t. R16a {i in E1, j in N1, v in V1: i <> j}:  $XP[i,j,v] \leq W[j]*X[i,j,v]$ ;

s.t. R16b {i in E1, j in N1, v in V1 :i <> j}:  $XP[i,j,v] \leq PP[j]$ ;

s.t. R16c {i in E1, j in N1, v in V1 :i <> j}:  $XP[i,j,v] \geq PP[j]-W[j]*(1-X[i,j,v])$ ;

s.t. R16d {i in E1, j in N1, v in V1 :i <> j}:  $XP[i,j,v] \leq W[j]$ ;

s.t. R17{v in V2}:  $\sum\{s \text{ in } N1, c \text{ in } N2: s \langle \rangle c\} U2[s,c,v] = \sum\{s \text{ in } E2, c \text{ in } N2: s \langle \rangle c\} Y[s,c,v]*D[c]$ ;

s.t. R18{v in V2}:  $\sum\{s \text{ in } N1, c \text{ in } N2: s \langle \rangle c\} U2[c,s,v] = \sum\{s \text{ in } E2, c \text{ in } N2: s \langle \rangle c\} Y[s,c,v]*P[c]$ ;

s.t. R19{j in N1}:  $\sum\{i \text{ in } E1, v \text{ in } V1: i \langle \rangle j\} U1[i,j,v] - DD[j] = \sum\{i \text{ in } E1, v \text{ in } V1: i \langle \rangle j\} U1[j,i,v] - PP[j]$ ;

s.t. R20{c in N2}:  $\sum\{s \text{ in } E2, v \text{ in } V2: s \langle \rangle c\} U2[s,c,v] - D[c] = \sum\{s \text{ in } E2, v \text{ in } V2: s \langle \rangle c\} U2[c,s,v] - P[c]$ ;

s.t. R21a{i in E1, j in E1, v in V1: i <> j}:  $0 \leq U1[i,j,v]$ ;

s.t. R21b{i in E1, j in E1, v in V1: i <> j}:  $U1[i,j,v] \leq CV1*X[i,j,v]$ ;

s.t. R22a{s in E2, c in E2, v in V2: s <> c}:  $0 \leq U2[s,c,v]$ ;

s.t. R22b{s in E2, c in E2, v in V2: s <> c}:  $U2[s,c,v] \leq CV2*Y[s,c,v]$ ;

s.t. R23{s in N1}:  $\sum\{c \text{ in } N2\} Z[s,c] \geq L[s]$ ;

```

s.t. R24{s in N1, c in N2}: Z[s,c] <= L[s];

s.t. R25{s in N1}: sum{c in N2, v in V2: s<>c} Y[s,c,v] >= L[s];

s.t. R26{s in N1, v in V2}: sum{c in N2: s<>c} Y[s,c,v] <= L[s];

## -- Ejecución del solver --

solve;

```

### 3.4.5. Desarrollo del generador de instancias

Para evaluar el modelo propuesto se requirió de instancias y parámetros realistas. Para ello, se desarrolló un generador de instancias cuidando aspectos importantes del problema como la disposición geográfica y el costo de viaje.

- **Disposición geográfica**

Las coordenadas geográficas de las ubicaciones de las instalaciones y clientes se generaron de forma aleatoria satisfaciendo los siguientes supuestos inspirados en [Crainic et al. \(2011\)](#):

- La estructura de la ciudad sigue un modelo concéntrico multi-escalón (ver Figura 3.3) con radios  $rc \leq rs \leq rm$ .
- Los clientes se ubican en el disco menor de radio  $rc$
- Las instalaciones satélites se ubican potencialmente en el aro interior al disco intermedio de radio  $rs$  y exterior al disco de radio  $rc$ ,
- La instalación maestra se ubica en el aro interior al disco mayor de radio  $rd$  y exterior al disco de radio  $rs$ ,
- Las ubicaciones de las instalaciones se encuentran en un sector circular de ángulo  $\alpha$ , para simular el espacio alrededor de la carretera de ingreso/salida comercial más importante,
- Las demandas de entrega y colecta varían en el intervalo de  $[d_{min}, d_{max}]$

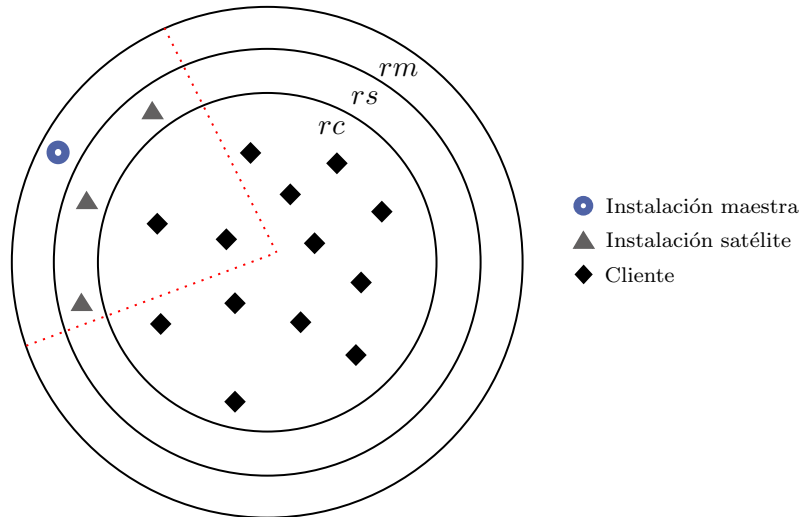


Figura 3.3: Modelo céntrico multi-escalón de la estructura de una ciudad

#### ■ Costo del viaje

La mayoría de la literatura conocida sobre problemas de rutas usan la distancia euclídeana para estimar el costo de viaje entre dos puntos. Sin embargo, se sabe que para la mayoría de países la relación entre la distancia en la red carretera respecto de la distancia euclídeana oscila entre 1,2 y 1,4 (Cole and King, 1968). Esto significa que en general las distancias euclídeanas son entre un 20 % y 40 % menor que las distancias en la red de carretera.

Para estimar con mayor precisión la ruta de mínima distancia (ó tiempo) de viaje entre dos nodos accedemos a datos geoespaciales de OpenStreetMap usando el paquete OSMnx (Boeing, 2017) de Python, y realizamos el siguiente procedimiento heurístico:

- Sean  $n_1$  y  $n_2$  dos nodos cualesquiera, calculamos los nodos  $nc_1$  y  $nc_2$  más cercanos sobre la carretera, respectivamente,
- Calculamos la ruta de distancia (ó tiempo) más corta sobre la carretera entre los nodos  $nc_1$  y  $nc_2$ ,
- Usamos esta ruta como la ruta de distancia (ó tiempo) más corta entre  $n_1$  y  $n_2$ .

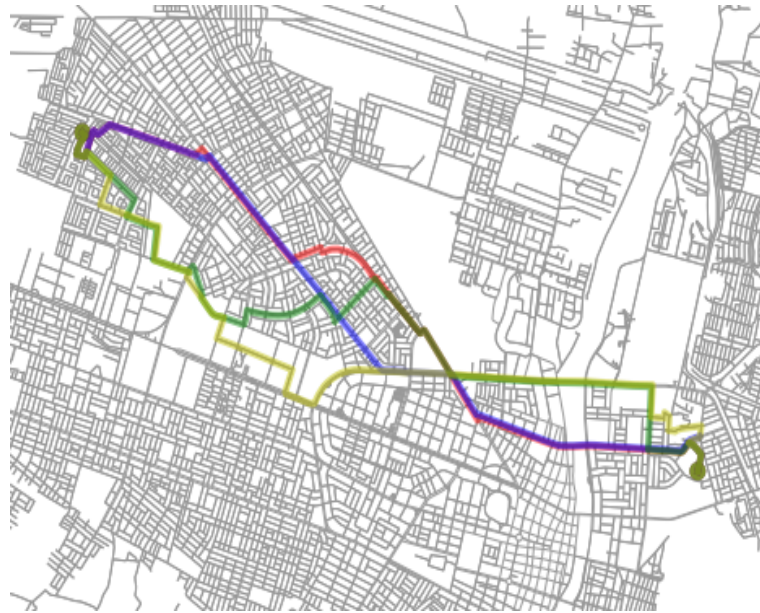


Figura 3.4: Ejemplo de costos de viaje: distancia mínima de  $i$  a  $j$  (rojo), distancia mínima de  $j$  a  $i$  (azul), tiempo mínimo de  $i$  a  $j$  (verde), tiempo mínimo de  $j$  a  $i$  (amarillo).

### 3.4.6. Implementación del generador

Se implementó el generador de instancias en el lenguaje de programación Python. Este nos permite obtener las ubicaciones de los nodos, las demandas de los clientes, la distancia en la red de carretera entre nodos y una gráfica realista la disposición geográfica.

- **Librerías**

```
import math
import osmnx as ox
import numpy as np
import networkx as nx
from random import randint
import matplotlib.pyplot as plt
```

- **Parámetros**

```
# G: grafo de la región geográfica
# P: punto central urbano del modelo céntrico
# a0: ángulo de referencia para el sector circular
# a0: ángulo central del sector circular
# m: número de instalaciones maestras
```

```

# s: número de instalaciones satélites
# c: número de clientes
# r_m: radio del disco urbano mayor
# r_s: radio del disco urbano intermedio
# r_c: radio del disco urbano menor
# dmin,dmax: extremo inferior de la demanda de entrega y colecta
# dmax: extremo superior de la demanda de entrega y colecta

```

- Ubicación de los nodos

```

def ubicaciones(p, a0 , a, m, r_m, s, r_s, c, r_c):

radio_m = np.random.uniform(low=r_s, high=r_m, size=m)
angulo_m = np.random.uniform(low=a0, high=a0+0.1*np.pi, size=m)
longitudes_m = list(p[1] + radio_m * np.cos(angulo_m))
latitudes_m = list(p[0] + radio_m * np.sin(angulo_m))

radio_s = np.random.uniform(low=r_c, high=r_s, size=s)
angulo_s = np.random.uniform(low=a0-(a/2), high=a0+(a/2), size=s)
longitudes_s = list(p[1] + radio_s * np.cos(angulo_s))
latitudes_s = list(p[0] + radio_s * np.sin(angulo_s))

radio_c = np.random.uniform(low=0, high=r_c, size=c)
angulo_c = np.random.uniform(low=0*np.pi, high=2*np.pi, size=c)
longitudes_c = list(p[1] + radio_c * np.cos(angulo_c))
latitudes_c = list(p[0] + radio_c * np.sin(angulo_c))

latitudes = latitudes_m + latitudes_s + latitudes_c
longitudes = longitudes_m + longitudes_s + longitudes_c

return print("latitudes", latitudes , " \n longitudes", longitudes)

```

- Demandas de los clientes

```

def demandas(dmin, dmax, c):
D=np.random.randint(low=dmin, high=dmax+1, size=(c,))
C=np.random.randint(low=dmin, high=dmax+1, size=(c,))
return print("Demanda de clientes:", "Entrega", D , "Colecta", C)

```

- Gráfica de la instancia

```

def instancia_grafica(longitudes,latitudes,m,s,c):

nodos_carretera=ox.distance.nearest_nodes(G, longitudes,
latitudes, return_dist=False)
rutas= ox.distance.shortest_path(G, nodos_carretera, nodos_carretera,
weight='length', cpus=1)

color_puntos = []
for num in range(0, m):
color_puntos.append('b')
for num in range(0, s):
color_puntos.append('g')
for num in range(0, c):
color_puntos.append('r')

grafo=ox.plot_graph_routes(G, rutas, route_colors=color_puntos,
node_size=0, figsize=(20, 20), bgcolor='#FFFFFF')
return grafo

```

- **Distancia en la red carretera entre nodos**

```

def distancia_nodos(lat_p1, lon_p1, lat_p2, lon_p2):

if lat_p1 == lat_p2 and lon_p1 == lon_p2:
return 0
origen=ox.distance.nearest_nodes(G, lon_p1, lat_p1, return_dist=False)
destino=ox.distance.nearest_nodes(G, lon_p2, lat_p2, return_dist=False)
DM= nx.shortest_path_length(G, origen, destino, weight='length')
DK=DM/1000
return round(DM,3)

```

```

def distancias_totales(latitudes,longitudes,m,s,c):

```

```

def distancia_indices(i,j):
return distancia_nodos(latitudes[i],longitudes[i],latitudes[j],
longitudes[j])

```

```

indices=[(i,j) for i in range(m+s+c) for j in range(m+s+c)]

```

```
distancias= [distancia_indices(x,y) for (x,y) in indices]
```

```
return distancias
```

- **Gráfica de las rutas**

```
def ruta(i,j):
```

```
    nodos_carretera=ox.distance.nearest_nodes(G, longitudes, latitudes,  
    return_dist=False)
```

```
    ruta_ij= nx.shortest_path(G, nodos_carretera[i], nodos_carretera[j],  
    weight='length')
```

```
    return ruta_ij
```

```
def grafica_rutas(lista_rutas):
```

```
    rutas = [ruta(lista_rutas[e],lista_rutas[e+1]) for e in  
    range(0,len(lista_rutas),2)]
```

```
    grafica = ox.plot_graph_routes(G, rutas, node_size=0,  
    figsize=(20,20), bgcolor='#FFFFFF')
```

```
    return grafica
```

## Capítulo 4

# Resultados y Discusión

En este capítulo se validó el modelo respecto a dos criterios: consistencia y funcionalidad. Además, se realizó una evaluación de la eficiencia del modelo respecto al tiempo de ejecución requerido para encontrar la solución óptima.

Un modelo de optimización es *consistente* si su correcta implementación produce un espacio de búsqueda no vacío y acotado, i.e., contiene soluciones óptimas. Por otro lado, la *funcionalidad* del modelo se comprueba cuando cualquier solución óptima se corresponde con la configuración y los objetivos del problema.

### 4.1. Resultados

Con el fin de validar y evaluar el modelo implementado en el software GLPK, este se aplica sobre un conjunto de instancias de distinto tamaño.

#### 4.1.1. Descripción de las instancias

Las instancias del problema se obtuvieron con el generador de instancias (ver Sección 3.4.6), utilizando como región geográfica la ciudad de Piura como área limitante para las ubicaciones de los nodos. Por lo cual todos los parámetros usados para generar estas instancias en particular se basan en características reales de la ciudad.

En la Tabla 4.1 y la Figura 4.1, se muestra una instancia compuesta por una instalación maestra, tres instalaciones satélites y diez clientes. Se generan las coordenadas geográficas de los nodos y las demandas de entrega y colecta de los clientes.

#### 4.1.2. Validación del modelo

Con el fin de validar la consistencia y funcionalidad, el modelo se aplica a un caso de estudio: una instalación maestra (Nodo: 0), tres instalaciones satélites potenciales (Nodos: 1,2,3), y ocho clientes (Nodos: 4,6,7,8,9,10,11,12), ver Tabla 4.1.

Tabla 4.1: Instancia simulada de diez clientes.

| Nodos | Latitud      | Longitud     | Entrega | Colecta | Tipo                 |
|-------|--------------|--------------|---------|---------|----------------------|
| 0     | -5.172749664 | -80.69015811 | -       | -       | instalación maestra  |
| 1     | -5.176564231 | -80.67642392 | -       | -       | instalación satélite |
| 2     | -5.162600086 | -80.67082297 | -       | -       | instalación satélite |
| 3     | -5.149887272 | -80.66023895 | -       | -       | instalación satélite |
| 4     | -5.161577874 | -80.63605486 | 5       | 6       | cliente              |
| 5     | -5.191104588 | -80.61860242 | 8       | 5       | cliente              |
| 6     | -5.246004625 | -80.62837855 | 7       | 6       | cliente              |
| 7     | -5.202825759 | -80.63082734 | 6       | 6       | cliente              |
| 8     | -5.189306445 | -80.61393078 | 8       | 5       | cliente              |
| 9     | -5.177340598 | -80.65774143 | 6       | 8       | cliente              |
| 10    | -5.183313696 | -80.6339345  | 7       | 6       | cliente              |
| 11    | -5.158422678 | -80.62050557 | 7       | 7       | cliente              |
| 12    | -5.188147416 | -80.60372039 | 6       | 5       | cliente              |
| 13    | -5.192809978 | -80.62700815 | 5       | 8       | cliente              |



Figura 4.1: Visualización de la instancia simulada (los puntos azules, verdes y rojos representan la instalación maestra, las potenciales instalaciones satélites y los clientes, respectivamente).

Para este caso de estudio, el modelo obtiene una solución óptima, que se muestra en la Tabla 4.2; en el primer escalón el modelo seleccionó dos de las tres instalaciones potenciales y utilizó un único vehículo para satisfacer las demandas de los satélites. En el segundo escalón el modelo utilizó dos vehículos para atender a los clientes

Tabla 4.2: Solución óptima.

| Vehículo | Ruta  | Escalón |
|----------|---|---------|
| 1        | 0 $\mapsto$ 1 $\mapsto$ 3 $\mapsto$ 0   | 1       |
| 3        | 1 $\mapsto$ 9 $\mapsto$ 10 $\mapsto$ 8 $\mapsto$ 12 $\mapsto$ 6 $\mapsto$ 7 $\mapsto$ 1 | 2       |
| 4        | 3 $\mapsto$ 4 $\mapsto$ 11 $\mapsto$ 3  | 2       |

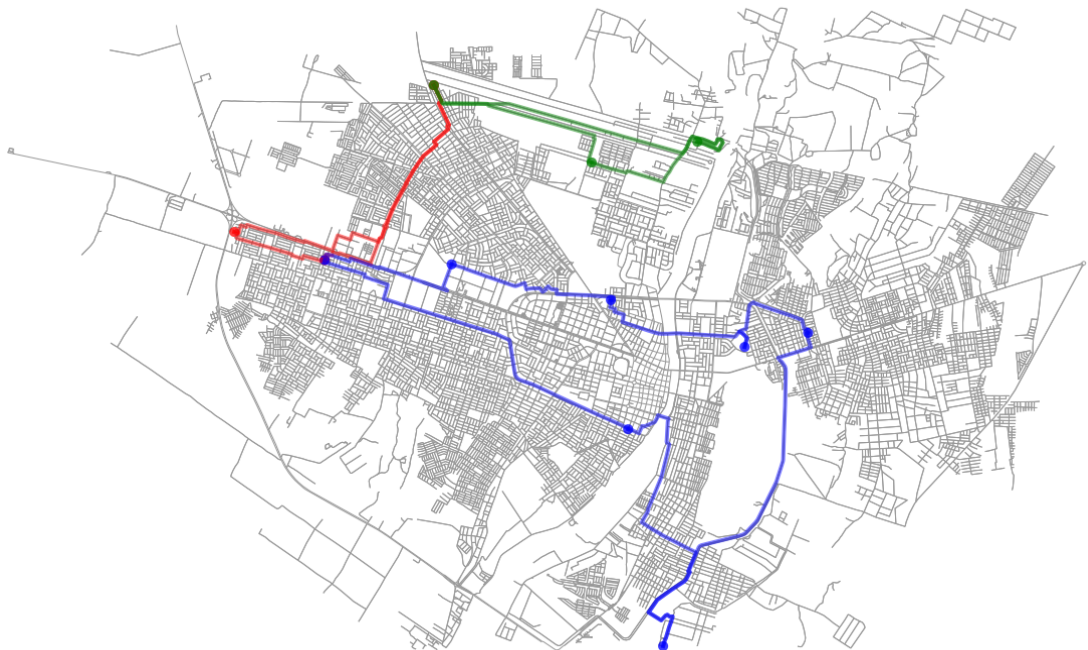


Figura 4.2: Visualización de la solución óptima.

En la Figura 4.2, se muestra sobre el mapa de la ciudad de Piura, los resultados de la Tabla 4.2, la ruta resultante para el primer escalón se señala de color rojo y las rutas resultantes para el segundo escalón se señalan con los colores azul y verde.

#### 4.1.3. Evaluación de la eficiencia del modelo

Para la evaluar la eficiencia del modelo se utilizaron nodos de la Tabla 4.1, se formaron instancias con una instalación maestra, tres ubicaciones potenciales y un número variable de clientes. Para cada una de las instancias se aplicó el modelo y de las soluciones obtenidas se registró: las localizaciones y rutas obtenidas, si la solución es óptima o no, el tiempo que se demora en obtener la solución y el valor de la función objetivo.

Los resultados obtenidos con un límite de ejecución de 3600 segundos, se muestran en la Tabla 4.3.

Por ejemplo, con una instalación maestra, tres satélites y dos clientes, el modelo obtuvo la solución óptima en un tiempo de 0.219 segundos. y una función objetivo de 4072.588176.

Tabla 4.3: Resultados computacionales.

| Satélites | Clientes | Solución óptima | Tiempo (segundos) | Objetivo (\$) |
|-----------|----------|-----------------|-------------------|---------------|
| 3         | 2        | Si              | 0.219             | 4072.588176   |
| 3         | 3        | Si              | 0.240             | 4113.044248   |
| 3         | 4        | Si              | 0.329             | 4172.607717   |
| 3         | 5        | Si              | 0.897             | 4570.747872   |
| 3         | 6        | Si              | 3.460             | 7238.189432   |
| 3         | 7        | Si              | 1.110             | 7286.024794   |
| 3         | 8        | Si              | 4.897             | 7332.719413   |
| 3         | 9        | Si              | 227.572           | 7396.627546   |
| 3         | 10       | Si              | 434.224           | 7450.337053   |
| 3         | 11       | No              | Fuera del límite  | -             |
| 3         | 12       | No              | Fuera del límite  | -             |

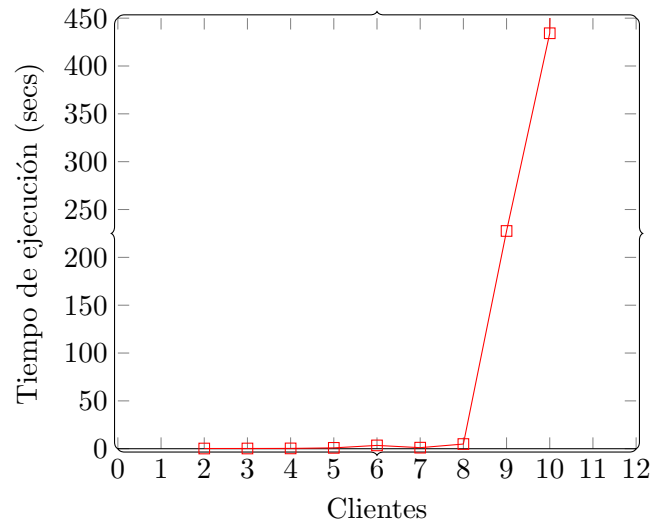


Figura 4.3: Evolución del tiempo de ejecución requerido.

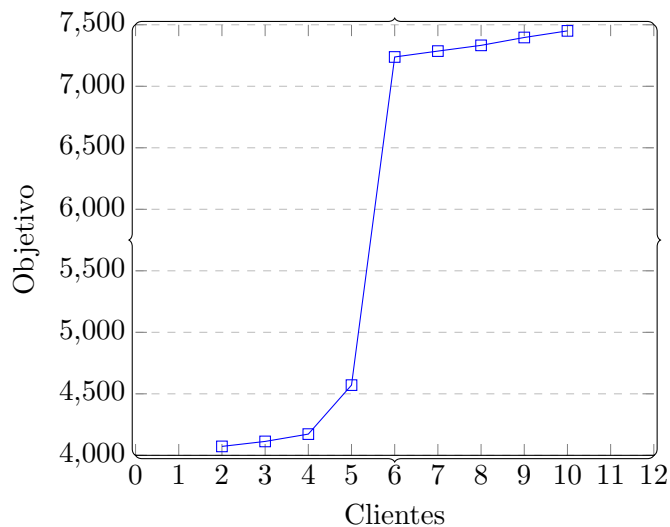


Figura 4.4: Evolución del costo óptimo.

El tiempo de ejecución y el valor de la función objetivo de la Tabla 4.3 se muestran en las Figuras 4.3 y 4.4.

## 4.2. Discusión

En el apartado 3.4.4 se implementó de forma satisfactoria el modelo propuesto en el paquete GLPK. No se obtuvo errores en la declaración de los parámetros, las variables, el objetivo y las restricciones del modelo. Esto prueba la correcta sintaxis del modelo.

El modelo es consistente, puesto que para una instancia obtiene al menos una solución óptima (ver Sección 4.1.2)

El modelo es funcional, puesto que las soluciones óptimas satisfacen la configuración y el objetivo del problema (ver Tabla 4.2 y Figura 4.2).

Respecto a la eficiencia del modelo, en la Tabla 4.3 se observa que el modelo propuesto, obtiene en pocos segundos la solución óptima hasta con 10 clientes, para más de 10 clientes no se obtiene ninguna solución en el tiempo límite de computo utilizado.

También se observa un cambio brusco en el tiempo que demora el modelo en obtener la solución, para 8 clientes el modelo obtiene la solución en 4.897 segundos, mientras que para 9 clientes demora 227.572 segundos.

# Conclusiones

- En el área de la logística urbana, donde se tiene un problema de localización-ruteo de dos niveles (2E-LRPSPD-E), con instalaciones (maestras, satélites y clientes), que requieren entrega y colecta de productos; el modelo de Programación Lineal Entero Puro (PILP) desarrollado en esta tesis, obtiene una solución óptima, es decir, el modelo realiza la mejor localización de las instalaciones satélites que permiten hacer el ruteo óptimo de la instalación maestra a los satélites y el ruteo óptimo de los satélites a los clientes para la entrega y colecta de productos.
- El modelo desarrollado tiene un aporte en el aspecto ambiental, pues minimiza las emisiones de CO<sub>2</sub> producidas por las operaciones de transporte, tomando en cuenta la carga transportada y la distancia en la red de carretera recorrida por los vehículos
- El modelo obtiene soluciones rápidas (pocos segundos), para instancias con un número pequeño de instalaciones.
- El generador de instancias desarrollado, logra que en una ciudad se puedan generar instancias con ubicaciones realistas y georreferenciadas de instalaciones (maestras, satélites y clientes), que permiten evaluar este modelo u otros relacionados con localizaciones y ruteo de vehículos.

Este generador tiene un aporte en el área ambiental, pues localiza las instalaciones satélites en la zonas periféricas de la ciudad; para evitar el uso de vehículos de gran tamaño y poco amigables con el medio ambiente, dentro de las zonas urbanas.

# Recomendaciones

El modelo propuesto es, en primera instancia, un artefacto, y por lo tanto está sujeto tanto a la evaluación como al mejoramiento. A continuación se presentan una lista de recomendaciones respecto a distintos aspectos de la investigación para futuras propuestas:

**Sobre el problema.** El problema abordado en esta investigación es una variante particular del 2E-LRP, de modo que podemos considerar otros atributos para obtener variantes más realistas. Respecto a esto se sugiere investigar en lo siguiente:

- Considerar ventanas de tiempo y objetivos múltiples,
- Considerar incertidumbre en los parámetros del problema,
- Se advierte que considerar más atributos e incertidumbre aumenta la complejidad del problema.

**Sobre el modelo.** Durante el diseño del modelo, el modelador elige modelos para estimar algunos parámetros según sus recursos de información. Respecto a esto se sugiere lo siguiente:

- Utilizar la distancia de carretera entre dos puntos, ya que estas representan una mejor estimación del costo de recorrido, que la distancia euclídeana,
- Considerar un modelo más realista de cálculo del consumo de combustible, siempre que se tenga acceso a estimaciones de alta precisión para los parámetros necesarios en un tiempo razonable .

**Sobre el método.** La complejidad del problema  $\mathcal{NP}$ -Difícil hace inviable el uso de solvers LP para instancias de tamaño realista. Respecto a esto se sugiere lo siguiente:

- Resolver el modelo utilizando heurísticas o metaheurísticas, para obtener buenas soluciones en tiempos razonables.

**Sobre las instancias** La construcción o selección de instancias para evaluar un modelo es una parte importante del diseño. Estas nos permiten obtener información útil, para corregir y mejorar el diseño propuesto. Respecto a esto se sugiere considerar varios modelos de la estructura de las ciudades en la elaboración de instancias simuladas, a fin de obtener mejores comparativas en la evaluación del modelo.

# Referencias bibliográficas

- Aronsson, H. and Brodin, M. H. (2006). The environmental impact of changing logistics structures. *The international journal of logistics management*.
- Bickel, P., Friedrich, R., Link, H., Stewart, L., and Nash, C. (2006). Introducing environmental externalities into transport pricing: measurement and implications. *Transport reviews*, 26(4):389–415.
- Boeing, G. (2017). Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65:126–139.
- Cole, J. and King, C. (1968). Quantitative geography john wiley. *London*, 692:200–201.
- Crainic, T. G., Sforza, A., and Sterle, C. (2011). *Location-routing models for two-echelon freight distribution system design*. CIRRELT Montréal.
- Cuda, R., Guastaroba, G., and Speranza, M. G. (2015). A survey on two-echelon routing problems. *Computers & Operations Research*, 55:185–199.
- Demircan-Yildiz, E. A., Karaoglan, I., and Altiparmak, F. (2016). Two echelon location routing problem with simultaneous pickup and delivery: mixed integer programming formulations and comparative analysis. In *International Conference on Computational Logistics*, pages 275–289. Springer.
- Drexl, M. and Schneider, M. (2015). A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2):283–308.
- Fan, H., Wu, J., Li, X., and Jiang, X. (2020). Presenting a multi-start hybrid heuristic for solving the problem of two-echelon location-routing problem with simultaneous pickup and delivery (2e-lrpspd). *Journal of Advanced Transportation*, 2020.

- Fontaras, G., Zacharof, N.-G., and Ciuffo, B. (2017). Fuel consumption and co2 emissions from passenger cars in europe—laboratory versus real-world emissions. *Progress in Energy and Combustion Science*, 60:97–131.
- Gonzalez-Feliu, J., Perboli, G., Tadei, R., and Vigo, D. (2008). The two-echelon capacitated vehicle routing problem.
- Guastaroba, G., Speranza, M. G., and Vigo, D. (2016). Intermediate facilities in freight transportation planning: A survey. *Transportation Science*, 50(3):763–789.
- Laffi, S. (2015). *Vehicle Routing Problems with Resources Synchronization*. PhD thesis, Université de Technologie de Compiègne.
- Li, H., Liu, Y., Chen, K., and Lin, Q. (2018). The two-echelon city logistics system with on-street satellites. *Computers & Industrial Engineering*.
- Liu, Y. and Helfand, G. E. (2009). The alternative motor fuels act, alternative-fuel vehicles, and greenhouse gas emissions. *Transportation Research Part A: Policy and Practice*, 43(8):755–764.
- Prodhon, C. and Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1):1–17.
- Rahmani, Y. (2015). *The Multi-product Location-Routing Problem with Pickup and Delivery*. PhD thesis.
- Riera, J. (2002). The traveling purchaser problem.
- Rodrigue, J. P., Dablanc, Laetitia, G. G., O’Brien, T., Conway, A., and Lee, J. S. (2018). City logistics: Concepts, policy and practice. *MetroFreight Consortium*.
- Romeijn, H. E., Shu, J., and Teo, C.-P. (2007). Designing two-echelon supply networks. *European Journal of Operational Research*, 178(2):449–462.
- Salhi, S. and Nagy, G. (1999). Consistency and robustness in location-routing. *Studies in Locational Analysis*, (13):3–19.
- Salhi, S. and Rand, G. K. (1989). The effect of ignoring routes when locating depots. *European journal of operational research*, 39(2):150–156.
- Soysal, M., Bloemhof-Ruwaard, J. M., and Bektaş, T. (2015). The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. *International Journal of Production Economics*, 164:366–378.
- Verma, P. and Raghubanshi, A. (2018). Urban sustainability indicators: Challenges and opportunities. *Ecological indicators*, 93:282–291.

- Vidal, T., Crainic, T., Gendreau, M., and Prins, C. (2013). Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research*, 231(1):1–21.
- Wang, K., Shao, Y., and Zhou, W. (2017). Matheuristic for a two-echelon capacitated vehicle routing problem with environmental considerations in city logistics service. *Transportation Research Part D: Transport and Environment*, 57:262–276.
- Xiao, Y., Zhao, Q., Kaku, I., and Xu, Y. (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & operations research*, 39(7):1419–1431.
- Zhao, Q., Wang, W., and De Souza, R. (2018). A heterogeneous fleet two-echelon capacitated location-routing model for joint delivery arising in city logistics. *International Journal of Production Research*, 56(15):5062–5080.