

Modelado, Simulación y Optimización - Proyecto 1

Amalia Cabonell - 202122079
Santiago Arenas - 202220359
Santiago Casasbuenas - 202214932

Problema a resolver - Proyecto C

Objetivo: El presente proyecto tiene como objetivo diseñar una estrategia integral de rutas para el transporte de carga a nivel nacional, que permita minimizar los costos operativos y de combustible. Se deberá garantizar la disponibilidad de recarga mediante estaciones de servicio estratégicamente ubicadas, a la vez que se respeta la restricción de peso impuesta en cada municipio por razones de seguridad vial.

Parte 1 - Formulación Matemática Completa

Definición de conjuntos

Se especificará de forma clara los conjuntos relevantes vinculados al problema.

1. Conjunto de Puntos de Acceso (Puertos)

P : Conjunto de nodos origen donde se ingresan las mercancías.

Ejemplo: $P = p_1$ donde $p_1 =$ Puerto Barranquilla

2. Conjunto de Destinos (Centros de Consumo)

D : Conjunto de destinos finales o municipios consumidores.

Ejemplo: $D = d_1, d_2, d_3$ donde:

$d_1 =$ Bogotá

$d_2 =$ Guasca

$d_3 =$ Cogua

3. Conjunto de Estaciones de Servicio

E : Conjunto de estaciones de recarga de combustibles disponibles en la red.

Ejemplo: $E = e_1, e_2, e_3$ donde:

$e_1 =$ Estación La Estrella

$e_2 =$ Estación El Sol

$e_3 =$ Estación Caribe

4. Conjunto de Vehículos

V : Conjunto de vehículos disponibles para el transporte.

Ejemplo: $V = v_1, v_2, v_3$ donde:

v_1 = Camión de 30 ton

v_2 = Camión de 25 ton

v_3 = Camión de 20 ton

5. Conjunto de "Arcos" Posibles

A : Conjunto de arcos entre nodos (puertos, estaciones, destinos) que representan posibles trayectos.

- Cada arco (i, j) indica que es posible moverse del nodo i al nodo j .
- Este conjunto se infiere del grafo de la red vial nacional y será determinado con base en la conectividad real y los peajes existentes.

6. Conjunto de Municipios con Restricción de Peso

$M \subseteq D$: Subconjunto de destinos con restricción de peso para los vehículos.

Ejemplo: $E = d_2, d_3$ donde Guasca y Cogua tienen límites de peso.

Definición de Parámetros

Lista de todas las constantes necesarias. Para este escenario, vamos a clasificar los parámetros en grupos de acuerdo con su tema, así serán más fáciles de ubicar y diferenciar.

1. Parámetros Geográficos

- lat_i : Latitud del nodo i
- $long_i$: Longitud del nodo i

2. Red y Costos de Trayecto

- $dist_{ij}$: Distancia en km entre el nodo i y el nodo j , con $(i, j) \in A$
- $PeajeBase_{ij}$: Tarifa base de peaje entre nodos i y j (COP).
- $PeajeTon_{ij}$: Costo adicional por tonelada transportada (COP/ton) entre i y j
 - Si no aplica, se considera 0.
- F_t : Tarifa de flete por km (COP/km) -> 5.000
- C_m : Costo de mantenimiento por km (COP/km) -> 700

3. Estaciones de servicio

Para cada estación $e \in E$:

- $PrecioCombustible_e$: Precio de combustible en estación e (COP/galón)

4. Vehículos

Para cada vehículo $v \in V$:

- $Capacidad_v$: Capacidad útil (ton)
- $Autonomía_v$: Autonomía máxima (km)

5. Restricciones de Municipios

Para cada destino $d \in D$:

- $PesoMax_d$: Peso máximo permitido en el municipio d (ton)
 - Si no tiene restricción, puede tomarse como un valor muy alto o inf.

6. Parámetros Derivados o Preprocesados

Dependiendo del nivel de detalle del modelo, podríamos pensar en definir parámetros calculados como:

- $CostoTotal_{ijv} = dist_{ij} * (F_t + C_m) + PeajeBase_{ij} + PeajeTon_{ij} * Capacidad_v$

VARIABLES DE DECISIÓN

Detalle de las variables principales y las auxiliares que permiten capturar el comportamiento esperado del sistema. Para facilitar el reconocimiento de variables, las agruparemos en cuatro grupos clave.

1. Variables de Ruta

Variables que nos indican si se utiliza un arco (trayecto) en una ruta para un vehículo determinado.

$$x_{ij}^v \in \{0,1\}$$

$$x_{ij}^v = \begin{cases} 1 & \text{si el vehículo } v \in V \text{ se desplaza del nodo } i \text{ al nodo } j \\ 0 & \text{dlc.} \end{cases}$$

$$\forall (i,j) \in A, v \in V$$

2. Variables de Carga Transportada

Variables que permiten modelar el peso que transporta cada vehículo por cada arco:

$$w_{ij}^v \geq 0$$

- Toneladas de carga transportadas por el vehículo v en el arco (i,j) .
- Esto ayuda a validar restricciones de capacidad, peso en municipios, peajes variables por tonelada.

3. Variables de Recarga

Variables que permiten representar la "estrategia" de recarga de combustible.

$$r_i^v \geq 0$$

- Galones de combustible recargados por el vehículo v en la estación $i \in E$.
- Si el nodo no es una estación, entonces $r_i^v = 0$

4. Variables Auxiliares (para consumo y autonomía)

Control del combustible disponible.

$$r_i^v \geq 0$$

- Cantidad de combustible (galones) disponibles en el tanque del vehículo v al llegar al nodo i .

Restricciones

Restricciones necesarias para representar adecuadamente la realidad del problema. Se agruparán las restricciones para mayor calidad.

1. Restricciones de Flujo

Nos aseguramos de que cada vehículo tenga una ruta continua:

- **Conservación en nodos intermedios (flujo balanceado):**

$$\sum_{j:(j,i) \in A} x_{ij}^v = \sum_{j:(i,j) \in A} x_{ij}^v \quad \forall i \in N, v \in V$$

- **Salida del punto de acceso:**

$$\sum_{j:(p,j) \in A} x_{pj}^v \leq 1 \quad \forall p \in P, v \in V$$

- **Llegada a un destino:**

$$\sum_{j:(i,d) \in A} x_{id}^v \leq 1 \quad \forall d \in D, v \in V$$

2. Restricciones de Capacidad y Peso

- **Límite de carga por vehículo:**

$$\sum_{(i,j) \in A} w_{ij}^v \leq Capacidad_v \quad \forall v \in V$$

- **Restricción de peso en municipios:**

$$\sum_{i:(i,d) \in A} w_{id}^v \leq PesoMax_d \quad \forall d \in (M \subseteq D), v \in V$$

3. Restricciones de Autonomía y Consumo

- **Estado de combustible al llegar al nodo j:**

$$f_j^v = f_i^v - Consumo_v * dist_{ij} + f_j^v \quad \forall (i,j) \in A$$

- **Autonomía máxima del taque:**

$$f_i^v = Autonomía_v \quad \forall i \in N, v \in V$$

- **No recarga si el nodo no estación:**

$$r_i^v = 0 \quad \forall i \notin E$$

4. Cosotos de Peaje con peso

OJO: Esto se puede incorporar directamente en la función objetivo como término de costo

- **Si hay costo por tonelada:**

$$\text{Peaje}_{ij}^v = \text{PeajeBase}_{ij} + \text{PeajeTon}_{ij} * w_{ij}^v$$

5. Entrega Total de Carga

(Si se conoce la demanda total a transportar. Toca preguntar pq no entendí esa parte o no leí bien, la dejo por si acaso).

$$\sum_{v \in V} \sum_{i:(i,d) \in A} w_{id}^v = \text{Demanda}_d \quad \forall d \in D$$

6. Consistencia entre variables

- **Solo puede haber carga o recarga si el arco (ruta) es usado:**

$$w_{ij}^v \leq \text{Capacidad}_v * x_{ij}^v, r_i^v \leq R_{max} * \sum_j x_{ij}^v$$

Parte 2 - Función Objetivo y Proceso de indagación

Función objetivo

Planteamiento explícito de la función a optimizar considerando posibles penalizaciones por incumplimiento de restricciones.

$$\sum_{v \in V} \sum_{i \in N} \sum_{j \in N} x_{ij}^v * \text{CostoTotal}_{ijv}$$

Descripción del proceso de indagación

Con el fin de definir y justificar la función objetivo, incluyendo la revisión de literatura y análisis comparativo con casos de estudio relevantes.

Para la empresa LogistiCo: National Logistics Division lo mas importante es la optimizacion de costos ya que no su objetivo no es la velocidad de entrega si no la completa garantia de que los paquetes van a llegar a su destino de forma optima para la compañía.

Parte 3 - Preprocesamiento de Datos y Análisis

Procesos utilizados para extraer y procesar datos adicionales

Extraemos todas las coordenadas de los lugares a visitar

```
In [1]: from geopy.geocoders import Nominatim
puerto_barraquilla = [-74.7964, 10.9639] #Coordenadas Barranquilla
puntos_acesso = [puerto_barraquilla]
```

```

geolocator = Nominatim(user_agent="MyApp")
centros_consumo = []
bogota = geolocator.geocode("Bogota") #Usamos geopy para obtener coordenadas de centros de c
bogota = [bogota.longitude, bogota.latitude]
guasca = geolocator.geocode("Guasca")
guasca = [guasca.longitude, guasca.latitude]
cogua = geolocator.geocode("Cogua")
cogua = [cogua.longitude, cogua.latitude]
centros_consumo.extend([bogota, guasca, cogua])

#estaciones_servicio = [[-74.5000,12.000],[74.2000, 10.8000],[-75.0000, 10.5000]] #Las que e
estaciones_servicio = [[-74.28409651016692,11.005912925098826],[-74.16238561225475,10.649143

# Si se tuviera un csv de varios datos se cargaria adecuadamente aqui se hace para ejercicio
coordinates = puntos_acceso + centros_consumo + estaciones_servicio
place_names = ['Puerto Barranquilla', 'Bogotá', 'Guasca', 'Cogua', 'Estación La Estrella', 'E

print(coordinates)

```

WARNING:urllib3.connectionpool:Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='nomi natim.openstreetmap.org', port=443): Read timed out. (read timeout=1)": /search?q=Bogota&format=json&limit=1'
 WARNING:urllib3.connectionpool:Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='nomi natim.openstreetmap.org', port=443): Read timed out. (read timeout=1)": /search?q=Cogua&format=json&limit=1'
 WARNING:urllib3.connectionpool:Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='nomi natim.openstreetmap.org', port=443): Read timed out. (read timeout=1)": /search?q=Cogua&format=json&limit=1'
 [[-74.7964, 10.9639], [-74.0836333, 4.6533816], [-73.8776944, 4.8664391], [-73.9599328011159, 5.10852035], [-74.28409651016692, 11.005912925098826], [-74.16238561225475, 10.649143797497034], [-74.82508479065856, 10.492299407024618]]

TODO Escribir api_key

In [3]: #api_key =

Utilizamos API de open route service para tener la matriz de distancias entre puntos de interes

```

In [4]: import requests
import json

def get_distance_matrix(coordinates, api_key):
    """
    Retrieve distance matrix from OpenRouteService API

    Parameters:
    - coordinates: List of [longitude, latitude] coordinates
    - api_key: Your OpenRouteService API key

    Returns:
    - Distance matrix as a 2D list
    """
    # OpenRouteService matrix service endpoint
    url = "https://api.openrouteservice.org/v2/matrix/driving-car"

    # Prepare the request payload
    payload = {
        "locations": coordinates,
        "metrics": ["distance", "duration"], # You can also use "duration"
        "units": "km" # Optional: specify units
    }

```

```

# Headers including your API key
headers = {
    'Accept': 'application/json, application/geo+json, application/gpx+xml, img/png; cha
    'Content-Type': 'application/json; charset=utf-8',
    'Authorization': api_key
}

try:
    # Make the POST request to the API
    response = requests.post(url, json=payload, headers=headers)

    # Check if the request was successful
    response.raise_for_status()

    # Parse the JSON response
    matrix_data = response.json()

    # Extract the distance matrix
    distance_matrix = matrix_data.get('distances', [])
    duration_matrix = matrix_data.get('durations', [])

    return distance_matrix, duration_matrix

except requests.exceptions.RequestException as e:
    print(f"Error making request to OpenRouteService: {e}")
    return None

# Get the distance matrix
distance_matrix, duration_matrix = get_distance_matrix(coordinates, api_key)

```

Imprimimos tabla de distancias

```

In [5]: from tabulate import tabulate

table_data = []
for i, row in enumerate(distance_matrix):
    # Create a row with place name as first column, followed by distances
    table_row = [place_names[i]] + [round(dist, 2) for dist in row]
    table_data.append(table_row)

# Print the table
print("Distance Matrix (kilometers):")
print(tabulate(
    table_data,
    headers=['From/To'] + place_names,
    tablefmt='pretty'
))

```

Distance Matrix (kilometers):

From/To	Puerto Barranquilla	Bogotá	Guasca	Cogua	Estación La Estrella
Estación El Sol					
Estación Caribe					
Puerto Barranquilla	0.0	990.92	955.45	935.42	63.38
116.91	58.92				
Bogotá	985.63	0.0	47.59	61.88	922.85
867.66	1017.3				
Guasca	964.07	48.08	0.0	40.31	901.29
846.1	995.74				
Cogua	935.73	60.55	39.83	0.0	872.95
817.76	967.4				
Estación La Estrella	63.69	928.45	892.98	872.95	0.0
54.44	115.73				
Estación El Sol	118.01	874.05	838.58	818.55	55.23
0.0	170.04				
Estación Caribe	59.36	1021.74	986.27	966.24	115.36
168.89	0.0				

In [9]: # Imprimimos tabla de duraciones

```
table_data = []
for i, row in enumerate(duration_matrix):
    # Create a row with place name as first column, followed by durations
    table_row = [place_names[i]] + [round(duration/3600, 2) for duration in row]
    table_data.append(table_row)

# Print the table
print("Duration Matrix (hours):")
print(tabulate(
    table_data,
    headers=['From/To'] + place_names,
    tablefmt='pretty'
))
```

Duration Matrix (hours):

From/To	Puerto Barranquilla	Bogotá	Guasca	Cogua	Estación La Estrella
Estación El Sol					
Estación Caribe					
Puerto Barranquilla	0.0	13.04	12.78	12.15	0.85
1.46	0.99				
Bogotá	12.96	0.0	1.01	1.18	12.16
11.48	13.37				
Guasca	12.75	1.09	0.0	0.96	11.95
11.27	13.16				
Cogua	12.14	1.36	1.04	0.0	11.33
10.66	12.55				
Estación La Estrella	0.86	12.25	12.0	11.36	0.0
0.67	1.71				
Estación El Sol	1.48	11.59	11.33	10.7	0.68
0.0	2.33				
Estación Caribe	0.96	13.48	13.22	12.59	1.64
2.25	0.0				

Integración de la información en el modelo

Esta información hace parte integral del modelo pues es la matriz $dist_{ij}$ que representa la distancia a recorrer entre el nodo i y el nodo j

Parte 4 - Ejemplos Ilustrativos

Casos de aplicación del modelo

1. Caso imposible

Un camino no alcanzable por LogistiCo sería la ruta entre el puerto de Barranquilla y el municipio de Cogua. Entre estos dos hay una distancia de 935.42 kilómetros. Con los camiones actualmente en la empresa ninguno de ellos sería capaz de llegar hasta Cogua sin realizar paradas. Escogemos el vehículo 3 con una autonomía de 800 kilómetros y una capacidad de 30 toneladas. Cogua tiene un peso máximo de entrada de 25, por lo que solo cargaremos el camión con 25 toneladas. El vehículo alcanzará a hacer aproximadamente el 85% del recorrido, quedando varado al rededor del municipio de Gualilo en (6.1876197, -73.7143039)

Ruta planeada Puerto Barranquilla → Cogua

Información Vehículo	Id Vehículo	Capacidad (ton)	Autonomía (km)
V3	25	800	

Información Destino	Id Destino	Municipio	Peso Máximo (ton)
D3	Cogua		25

En la siguiente imagen se pude la ruta calculada por medio de google maps. Se muestra la ruta original, no se muestra donde se debe quedar el vehículo sin gasolina.



2. Camino Posible

Un camino alcanzable por LogístiCo sería la ruta entre Puerto Barranquilla, Estación el Sol, Cogua y Guasca. En total las distancias recorridas por el vehículo serían de 975.29 kilómetros, en la tabla a continuación se muestran las distancias de cada recorrido. Esta ruta será realizada por un nuevo camión que adquiera la empresa que tenga una autonomía de mínimo 870 kilómetros y una capacidad mínima de 25 pero recomendada mayor. Este vehículo 4 saldría del Puerto Barranquilla transportando 25 toneladas pues es el máximo para entrar a Cogua y entregaría mínimo 7 toneladas. Cuando tenga menos de 18 toneladas, se dirigirá a Guasca.

Ruta planeada Puerto Barranquilla → Estación El Sol → Cogua → Guasca

Informacion vehiculo	Id Vehículo	Capacidad (ton)	Autonomía (km)	
	----- ----- ----- -----	V4	>=25	>=860

Informacion Ruta	Ruta	Distancia	Autonomía (km)	Recarga Gasolina
	----- ----- ----- -----	Puerto Barranquilla	900	No
	----- ----- ----- -----	Puerto Barranquilla → Estación el sol	116.91	783.09
	----- ----- ----- -----	Sí (vuelve a 900 la autonomía)	Estación el sol → Cogua	818.55
	----- ----- ----- -----	Cogua → Guasca	39.83	41.62
	----- ----- ----- -----	No	81.45	No

En la siguiente imagen ilustramos el recorrido planteado anteriormente. Los puntos amarillos en las estaciones o municipios representan las paradas realizadas en el camino.



Conclusion análisis factibilidad y consistencia

Caso Imposible: La ruta directa entre Puerto Barranquilla y Cogua no es viable con los vehículos actuales debido a la autonomía insuficiente. De hecho, con los datos planteados como están en este momento, ninguna de las rutas es posible. Todas las distancias entre tanto el puerto como las estaciones de servicio superan los 800 kilómetros lo que es la autonomía máxima de los vehículos de LogístiCo.

Caso Posible: Si a LogístiCo le interesa la adquisición del vehículo 4 que tenga una mayor autonomía y posiblemente una mayor capacidad, las rutas deseadas si serán posibles. Con una distancia total a recorrer de 975.29 kilómetros y una parada en la estación de servicio, es posible el recorrido.

Actualmente no tenemos la demanda de cada uno de los municipios para estimar con mayor precisión la ruta y la capacidad necesaria para cumplir dicha demanda. Sin embargo, con valores teóricos y un nuevo vehículo vemos que la ruta de puerto barranquilla a Cogua y Guasca es posible.

Análisis de factibilidad y consistencia de la solución

Parte 5 - Justificación y Discusión

Justificación de los elementos del modelo

Justificación de los elementos del modelo y relevancia de los datos, las decisiones metodológicas adoptadas y los posibles impactos operativos en el contexto de LogistiCo en Bogotá

Conjuntos

1. Conjuntos punto de Acceso

Se representa el nodo de entrada de mercancía al sistema, puerto desde el cual se va a distribuir los productos. En este caso solo tenemos un nodo inicial en el Atlántico, puerto Barranquilla. Se plantea como un conjunto pues en caso de que LogistiCo desee agregar un puerto de origen en algún momento, no se tiene que replantear todo el modelo, sino agregarlo al conjunto.

2. Conjunto destinos

Representa los municipios a donde se entregarán los productos transportados. Se separan de los puntos de acceso pues estos municipios tienen una demanda específica que debe ser satisfecha. Se plantea el conjunto de tal manera que en caso que LogistiCo desee agregar más municipios en su red, sea sencillo y no requiera mayor modificaciones del modelo.

3. Conjunto estaciones de servicio

Representa los puntos donde los vehículos pueden recargar combustible durante una ruta. Representarán una parada adicional en la ruta que se plantee según los municipios que cada vehículo debe visitar. Esto permite evaluar la restricción de recarga establecida por LogistiCo y la estrategia integral de recarga. Estas dos condiciones plantean la importancia de parar a cargar el camión que transporte los productos de gasolina a lo largo del trayecto, lo que permite la continuidad de las operaciones.

4. Conjunto de vehículos

Define el conjunto de camiones disponibles para el transporte. Cada uno de los camiones tiene capacidades y costos diferentes lo cual se tiene que tener en cuenta al planear los caminos por la distancia y peso que puede transportar cada uno de ellos.

5. Conjunto de arcos posibles

Se representan las conexiones existentes entre municipios. Incluye tanto conexiones con nodos de acceso o iniciales como con municipios normales y estaciones de parada. LogistiCo provee una tabla en la cual se muestran las condiciones de cada camión incluyendo la capacidad de peso y la autonomía en cuanto a kilómetros.

6. Conjuntos de municipios con restricción de peso

Este conjunto es un subconjunto de destinos que tienen limitaciones en el peso de los vehículos que pueden ingresar. Este conjunto se usará para establecer restricciones. LogistiCo está interesado en no poner en riesgo ninguna carretera para completar las rutas, por lo cual cuando un municipio tenga un límite de peso que puede ingresar, este debe ser valorado y tomado en cuenta para calcular las rutas.

Parametros

1. Parámetros geográficos

- Latitud del nodo i

Representa el valor de la latitud del nodo i

- Longitud del nodo i

Representa el valor de la longitud del nodo i \

Estos parámetros se utilizan para mostrar la ubicación geográfica del nodo i. Permiten calcular distancias reales entre los nodos para evaluar las distancias y seleccionar las mejores rutas. Tenemos los datos de latitud y longitud para las estaciones de servicio y para el puerto barranquilla, pero no tenemos los datos de las estaciones de los municipios. Para esto utilizamos geolocator en el código para obtener las coordenadas de los centros de consumo. Con ese valor encontrado, calculamos la latitud y longitud de cada punto de la red.

2. Red y Costos del trayecto

- **Distancia entre nodo i y j (dist(ij))**

Nos muestra la distancia en kilómetros entre el nodo i y j. Estas distancias son calculadas en el código por el método `get_distance_matrix(coordinates, api_key)`, la cual devuelve una matriz que nos muestra las distancias entre cada uno de los municipios, el puerto de origen y las estaciones de servicio. Esta matriz de distancia nos ayuda a cumplir con la optimización de las rutas que está buscando LogístiCo. La empresa mencionó que se están buscando rutas que minimicen el costo total del transporte considerando distancias, peajes, mantenimiento y combustible. Con esta matriz analizaremos la condición de distancia.

- **Tarifa de peaje entre i y j (Peaje Base(ij))**

Representa un costo fijo en el camino entre algunos municipios. El código analizará este costo en el cálculo de los caminos para asegurarse de que el costo del peaje sea tomado en cuenta y se trate de minimizar las veces que se pase por este.

- **Costo adicional por tonelada transportada entre i y j (PeajeTon(ij))**

Representa un costo variable según el peso del camión para un peaje entre municipios. Al igual que el valor mencionado anteriormente, este valor de peaje se analizará en el código para ser tenido en cuenta. Este peaje, a diferencia del anterior, puede cambiar su valor dependiendo en qué momento se pase pues el peso del camión cambiará durante el viaje y las paradas.

- **Tarifa por flete por km (F_t)**

Este valor es un parámetro operativo adicional que representa el costo de operación del camión por kilómetro. Este costo adicional es clave para la optimización de costos pues se debe tener en cuenta que el valor aumenta con los kilómetros recorridos por un vehículo.

- **Costo de mantenimiento por km (C_m)**

Este valor también es un parámetro operativo adicional que representa los gastos asociados con un vehículo por su desgaste y el mantenimiento que se debe realizar. Este costo adicional es clave para la optimización de costos pues se debe tener en cuenta que el valor aumenta con los kilómetros recorridos por un vehículo.

3. Estaciones de servicio

- Precio de combustible

Este parámetro está asociado con las estaciones de servicio y representa el valor de la gasolina en dicha estación. Este precio puede tener un efecto en la ruta escogida pues puede variar el precio de la ruta según la parada que se escoja. Esto es mencionado por LogístiCo en la estrategia de recarga pues ya no solo se tiene en cuenta la ubicación de la estación sino el precio de la gasolina.

4. Vehículos

- Capacidad

Nos muestra la capacidad de peso o carga que tiene un camión. Esto se tiene en cuenta para minimizar la cantidad de viajes que se puede hacer pero se tiene que tener en cuenta el peso máximo de cada municipio en la ruta del camión. Este valor es dado por Logística y está en toneladas.

- Autonomía

Este parámetro asociado a los vehículos nos muestra la autonomía que tiene un camión en kilómetros. Por autonomía entendemos la distancia que puede recorrer un camión con el tanque lleno y sin necesidad de parar a recargar.

5. Restricciones de municipios

- Peso Máximo (PesoMax)

Como mencionado anteriormente, los municipios tienen un peso máximo que es permitido entrar a él. Este valor es representado por medio del parámetro peso máximo y nos ayudará a planear las rutas y el orden de las paradas para cumplir con la restricción.

6. Parámetros derivados o preprocesados

- Costo total

Este parámetro derivado combina en una misma fórmula todos los costos relevantes para evaluar una ruta. Todo está basado en el objetivo de LogístiCo de diseñar una estrategia de rutas que permita minimizar los costos operativos y de combustible. La ecuación reúne todos estos costos de manera que se suma, la distancia multiplicada por la tarifa de flete y la de mantenimiento, se suma la tarifa fija del peaje y la variable por las toneladas del camión, multiplicado por la capacidad del camión.

Variable de decisión

- Variable de ruta \

Es una variable binaria que nos permite determinar si el vehículo V se desplaza por el arco i j. De esta manera podemos ir determinando el camino que sigue un camión. Para esta variable de decisión se toma un valor de 1 si el camión si pasa por la ruta o un valor de 0 si el camión no se transporta por esa ruta.

- Variable de carga transportada \

Es una variable numérica que nos permite analizar cuánto peso es transportado por un camión mientras pasa del municipio i al municipio j. Este valor debe ser mayor o igual a 0. La razón por la que incluimos el 0 es por el viaje final. El camión debe ser capaz de retornar al origen o puerto barranquilla después de realizar las entregas. El valor en ningún momento debe ser negativo.

- Variable de recarga \

Esta variable permite hacer realidad la estrategia de recarga planteada por LogístiCo. Está representada como un valor numérico que nos muestra la cantidad de galones de gasolina ingresados al camión en la estación de servicio escogida. Esta variable recorre todas los nodos pero se asigna un valor de 0 en caso de que la estación no sea una estación de servicio.

- Variable Auxiliar (para consumo y autonomía) \

Esta variable numérica representa la cantidad de gasolina en galones disponible en el tanque del vehículo para llegar al nodo i. Esta variable nos permite controlar el consumo y la autonomía pues al

ver la gasolina restante del vehículo podemos determinar la cantidad de kilómetros que podría recorrer y cuantos galones se gastaría.

Restricciones

1. Restricción de flujo

Se plantean 3 restricciones con respecto al manejo de flujo de los camiones

- Conservación de flujo nodos intermedios Esta restricción permite conectar los caminos y asegurarse de que al llegar un camión a un municipio, este mismo camión saliera de este. Esta condición permite la creación de rutas pues permite las conexiones y evita que los camiones se puedan "teletransportar" entre municipios.
- Salida de punto de acceso \

Esta restricción se asegura de que los vehículos salgan del nodo inicial. Es decir que salgan del puerto de Barranquilla hacia los demás municipios destino de la red. Como LogístiCo solo tiene un nodo de inicio la condición es muy simple. En caso de que se quiera modificar y tener más de un inicio sería muy sencillo agregar el nuevo origen. Como solo se tiene uno, se asegura que todos salgan de Puerto Barranquilla.

- Llegada a un destino \

Esta restricción asegura que los vehículos lleguen a algún nodo destino. Es decir, no una estación de servicio y no el nodo inicial. Se está buscando poder llegar a todos los municipios y asegurar que sea un destino.

2. Restricción Capacidad y Peso

- Límite de carga de un vehículo \

Esta restricción trabaja en conjunto con el parámetro de capacidad de un camión para asegurarse que en ningún momento se está cargando más peso del posible.

- Restricción por peso en municipios \

Esta restricción se basa en la capacidad de peso de cada municipio destino para asegurarse que en ningún momento un camión ingrese más peso del que es permitido. Se da como ejemplo que a Guasca no se pueden ingresar más de 18 toneladas.

3. Restricción de autonomía y consumo

- Estado combustible \

Al llegar un vehículo a algún destino, se debe analizar cuánta gasolina esta restante en el tanque del vehículo. Se calcula con el flujo encontrado en la estación anterior menos el consumo por la distancia. De esta manera se puede saber cuánta gasolina queda después de cada trayecto y cuando es necesario tanquear el vehículo.

- Autonomía máxima del tanque \

Esta restricción verifica la autonomía máxima que tendría un vehículo teniendo en cuenta las condiciones específicas del vehículo.

- Recargar solo en estacion

Para evitar que el programa cargue gasolina en cualquier nodo, se incluye una restricción que garantice que solo se pueda hacer recarga en las estaciones de servicio. De esta forma se trabaja en la estrategia

de recarga.

4. Costos de peaje con peso

Si en el peaje específico hay un costo adicional por el peso del camión, se agrega la condición de agregar el costo del peaje con el costo del precio por el peso. Existe la posibilidad de agregar esta restricción directamente a la función objetivo. Pero por su cuenta también funciona y se encarga de cumplir con la gestión de peajes y restricciones viales planteadas por LogístiCo.

5. Entrega total de carga

LogístiCo busca optimizar la entrega y minimizar los costos de transporte. Por esta razón, en ningún momento se va a transportar materiales que no deben ser entregados. Esta restricción se asegura de repartir todos los materiales dentro de él. Todos los materiales que hayan sido empacados en un camión deben ser entregados.

6. Consistencia entre variables

Esta restricción verifica que se recargue en la estación de servicio si y sólo si el arco en el que se encuentra la parada está siendo usado. Al igual que la condición de nodos intermedios, evita que los camiones puedan teletransportarse o llegar a estaciones de servicio de un camino que no se esté recorriendo.

Función Objetivo

La función objetivo busca minimizar el costo total de transporte en la red logística de LogístiCo.

$$\sum_{v \in V} \sum_{i \in N} \sum_{j \in N} x_{ij}^v * CostoTotal_{ijv}$$

- Sumatoria sobre el conjunto de vehículos garantiza que el modelo optimice el costo considerando a todos los vehículos disponibles. Esto permite que los recursos se repartan de forma eficiente y se reduzcan los costos operativos. Se tiene que tener en cuenta que puede ser más eficiente sacar todos los vehículos pero más costoso hacerlos.
- Sumatoria sobre los nodos tanto en i como en j, se aseguran de analizar todas las rutas posibles. De esta manera se puede encontrar la ruta óptima minimizando costos y distancias.

La variable de decisión, nos indica si un vehículo v utiliza el camino i j. De esta forma se puede seleccionar las rutas más rentables y analizar los caminos.

- Costo total nos muestra el costo asociado con el uso de la ruta i j. Este valor como mencionado anteriormente, tiene en cuenta el vehículo, la distancia, la gasolina, peajes, mantenimiento y costos operativos adicionales.

Esta formulación permite a LogistiCo garantizar que los paquetes lleguen a su destino de la manera más económica y sostenible posible, sin comprometer la viabilidad operativa del sistema.