

# [193045] Optimización de Recorridos y Frecuencias en un Sistema BRT

Álvaro Andrés Rodríguez Galiano<sup>a,c</sup>, Juan Camilo Ibarra Cifuentes<sup>a,c</sup>,

Nicolás Rincón García<sup>b,c</sup>

<sup>a</sup>Estudiante de Ingeniería Industrial

<sup>b</sup>Profesor, director del Proyecto de Grado, Departamento de Ingeniería Industrial

<sup>c</sup>Pontificia Universidad Javeriana, Bogotá, Colombia

---

## Resumen

Currently, the transport of passengers within large cities has become a priority problem for their development and operation, for this reason, different approaches have been implemented to solve this problem known as TNDP (Transit Network Design Problem). The city of Bogotá has the Transmilenio BRT system, which is considered one of the great feats of mobility in Latin America, however, the level of user satisfaction with the system does not exceed 20%, mainly due to overcrowding of users in the trunk system, the number of users has exceeded the capacity of both stations and buses during peak hours.

For this reason, this undergraduate project seeks to take advantage of the incorporation of the new bus fleet, which has bi-articulated vehicles, and to propose, through the implementation of a multi-objective optimization method, a new set of routes and frequencies that focus on regulating occupancy average within the buses. On the other hand, a proposal is presented for the size of the stations according to the new distribution of the routes in order to improve the experience of the users of the trunk service by reducing both the average occupancy of the buses and the waiting times inside the stations without losing sight of the needs of the service provider, trying to minimize the necessary number of buses.

*Palabras claves: BRT, Transmilenio, Optimización multiobjetivo, TNDP (Transit Network Design Problem), satisfacción usuarios.*

---

## 1. Justificación y planteamiento del problema

El transporte es una fuerza motriz fundamental del desarrollo económico y social; genera oportunidades para los pobres y habilita a las economías a ser más competitivas (Banco Mundial, 2017). En la ciudad de Bogotá, se dispone de cuatro categorías de transporte: público, privado, humano e informal; generando un total de 17 millones de viajes diarios durante el año 2015, liderados por el transporte humano y público (Secretaría de Movilidad, 2015). Sin embargo, el transporte privado ha presentado un aumento constante, entre el año 2008 y el año 2015 el porcentaje de crecimiento de vehículos particulares ascendió a un 164,78%, y para motos a un 319,81%, para un total de 1.074.408 de automóviles y 449.823 motocicletas (Movilidad, 2015).

Estos aumentos del parque automotor en la ciudad de Bogotá han convertido la movilidad en un caos durante los últimos años, según el índice de tráfico TomTom para el año 2018 la ciudad se ubicó en el segundo puesto a nivel mundial en congestión vehicular, siendo superada solo por Bombay, India. Esto como reflejo del creciente número de personas que abandonan el transporte público, para el año 2017 el 68% de los habitantes de la ciudad usaban el transporte público como principal medio para sus desplazamientos, sin embargo, para el 2018 esta cifra disminuyó al 61%. Por el contrario, el porcentaje de habitantes que utiliza carro particular para

sus desplazamientos tuvo un aumento de 5 puntos porcentuales entre los años 2017 y 2018. (Bogotá cómo vamos; 2018).

Lo anterior se da a pesar de que Bogotá cuenta con el sistema BRT de TransMilenio, el cual en principio fue reconocido como “el proyecto de movilidad urbana más importante del mundo” (McCormick, 2005), sumado a esto, ganó el premio Stockholm Partnerships en 2002, por un modelo de transporte el cual pudiese ser utilizado como base para replicar en otros países que ayude al medio ambiente, la igualdad social y el desarrollo económico en ciudades del tercer mundo, según Edgar Enrique Sandoval, quien era el gerente de TransMilenio en 2002, fue como haber ganado el premio nobel en movilidad.

Además, en una entrevista realizada por la revista semana en el 2005 a Guido Robazza, investigador italiano del instituto universitario de arquitectura de Venecia, Guido (2005) comentó:

“TransMilenio es el proyecto de movilidad urbana más importante que se está realizando a escala mundial por varias razones: los retos que se propone en un contexto tan complejo como el bogotano, la magnitud de su tamaño, el corto plazo en que fue implementado y su modelo económico. Lo interesante y novedoso de un sistema como TransMilenio es que ofrece unas potencialidades de funcionamiento muy eficientes que le permiten ser competitivo con los sistemas masivos como los metros, y por otra parte cuesta infinitamente menos.”

Sin embargo, en el año 2018 el nivel de satisfacción de los habitantes de Bogotá con el servicio troncal de Transmilenio fue de un escaso 13%, frente al 49% que representaba en el 2008 este mismo indicador. Esta constante disminución del nivel de satisfacción de los usuarios de Transmilenio es una de las razones que ha motivado la migración a distintos medios de transporte en especial el carro particular, generando así en palabras de Kennedy *et al.* (2005) “ciudades cada vez más disfuncionales, con una siempre creciente congestión, pérdida de calidad de vida y más probablemente como resultado, productividad económica y competitividad decrecientes”. (p.394)

Este aumento de carros en las vías ha llevado a los ciudadanos de Bogotá a soportar un promedio de congestión vehicular del 63% en el 2018, esto quiere decir que aquellos viajes que deberían durar 30 minutos terminan durando aproximadamente 49 minutos, ahora bien, este porcentaje de congestión aumenta en horas pico, llegando a las 6 p.m. a un promedio de 104% de demoras en los tiempos de traslado por culpa de los embotellamientos, llevando así el mismo recorrido de 30 minutos a durar más de una hora.(TomTom, 2018)

Por lo anterior, la Alcaldía se planteó en el Plan Distrital de Desarrollo 2016-2020, Bogotá Mejor Para Todos, la meta de aumentar el nivel de satisfacción de los ciudadanos frente al sistema de Transmilenio hasta llegar al 30%, con el fin de frenar la migración de usuarios del sistema BRT y disminuir el creciente número de vehículos particulares que salen diariamente a las calles de la ciudad, mejorando así la calidad de vida de los habitantes. A pesar de esto, el distrito no ha conseguido tener mejoras considerables frente a este problema, sino todo lo contrario, la percepción de calidad sigue disminuyendo. (Bogotá como vamos, 2019)

Para entender los principales focos de problema en el servicio integrado de transporte, la Cámara de Comercio realizó la encuesta de percepción a los usuarios sobre las condiciones, calidad y servicio del sistema en el año 2016 la cual arrojó que el 66% de los pasajeros se quejan del sobrecupo en los buses, siendo este el principal problema, y en el segundo lugar se encuentra la falta de buses con un 40% de quejas. Aun así, esto no es nada nuevo, ya para el 2007 Bogotá como vamos reveló que el 89% de los consultados se quejaban por la capacidad del sistema, fuera de esto, ya en el 2006 se presentaron manifestaciones por parte de la comunidad en contra de la irregularidad en la frecuencia de los buses.

Ahora bien, en el informe *Para evaluar los cambios de la calidad de vida en la ciudad* (2019) se presenta la siguiente aclaración:

“El análisis de la ocupación de los sistemas de transporte público se ha empezado a considerar internacionalmente dado que tiene impacto en la velocidad de operación, tiempo de espera, frecuencia, percepción de comodidad y consecuentemente en la selección de rutas por parte de los usuarios.”

En este sentido, el manejo de la demanda en Transmilenio se convierte en uno de los factores principales al momento de querer mejorar el nivel de satisfacción de los usuarios del sistema. La factibilidad operacional del sistema se realizó para un máximo de 6 personas/m<sup>2</sup> sin embargo este valor tiene su máximo en 5 y 4 personas/m<sup>2</sup> en Estados Unidos y Europa respectivamente.

En respuesta al problema, el gobierno autorizó el cambio de flota para los buses de Transmilenio “lo que conlleva un proceso de reingeniería para la operación” aseguraron los de Transmilenio (El Tiempo, 2019), ya que el 67% de estos nuevos buses serán biarticulados. Ahora bien, si los servicios troncales de Transmilenio no quieren volver a caer en el mismo problema de sobre cupo e irregularidades en la frecuencia de los buses deberá replantear la manera en la cual se está prestando el servicio, teniendo en cuenta esta nueva flota de buses y teniendo como prioridad para los usuarios la disminución en los tiempos de espera de los buses con el fin de disminuir la congestión y aumentar así la percepción en la calidad del servicio por parte de los usuarios.

Teniendo en cuenta el cambio en la flota, este proyecto pretende realizar una propuesta de nuevas rutas para el sistema troncal de Transmilenio con sus respectivas frecuencias y capacidades necesarias de las diferentes estaciones. Esto mediante un enfoque multiobjetivo el cual en principio busca priorizar la experiencia de los usuarios teniendo en cuenta tanto la ocupación media como la ocupación máxima de los buses, fuera de esto se deben tener en cuenta los tiempos de espera tanto en los recorridos como en las estaciones. Lo anterior sin dejar de pensar en los costos que implican estas mejoras, razón por la cual el segundo punto de priorización será el uso mínimo de buses que permitan generar mejoras en la percepción de calidad del servicio.

## 2. Antecedentes

De acuerdo con el ITDP, el BRT (Bus Rapid Transit) es un sistema de transporte público de alta calidad basado en buses que proporciona movilidad urbana rápida, cómoda y costo efectiva a través de la provisión de infraestructura de derecho de vía segregada, operaciones rápidas y frecuentes, y excelencia en el mercadeo y servicio al cliente/usuario (2010). Para la planeación e implementación del BRT, se cuenta con seis áreas de planificación: preparación del proyecto, diseño operacional, diseño físico, integración, plan de negocios y evaluación e implementación (ITDP, 2010). En la ilustración 1 se muestran algunas de las características más importantes de las áreas de planificación.

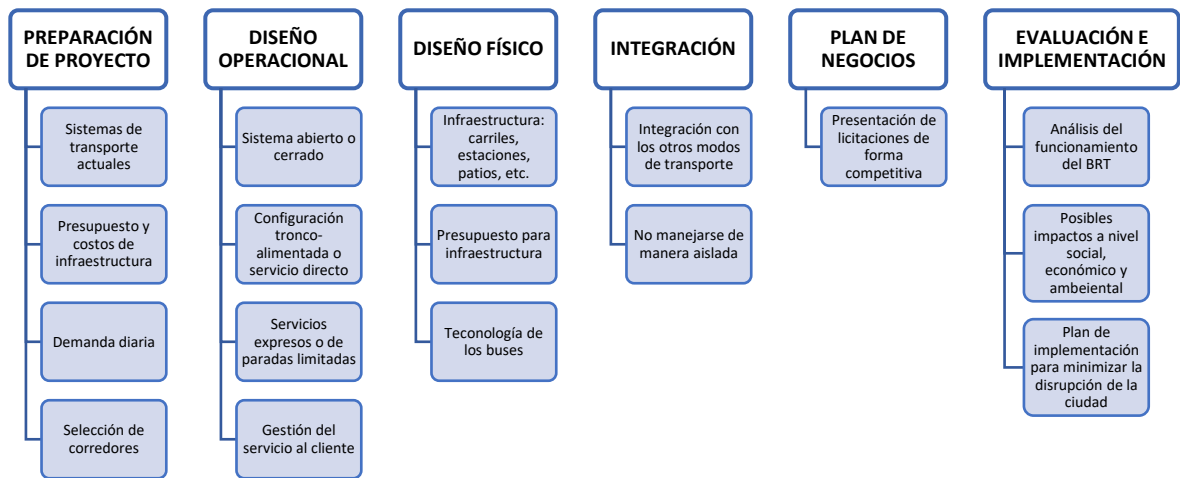
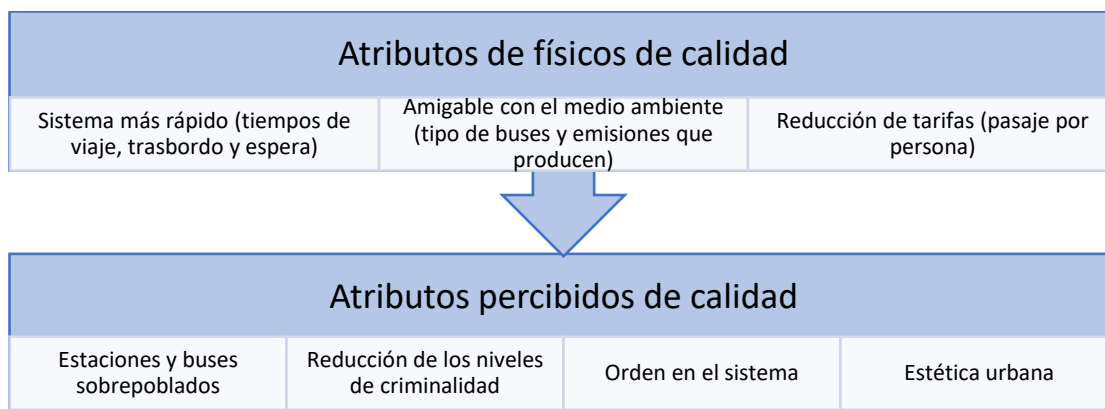


Ilustración 1. Áreas de planificación del sistema BRT  
Fuente: Insitute for Transportation and Development Policy. (n.d.)

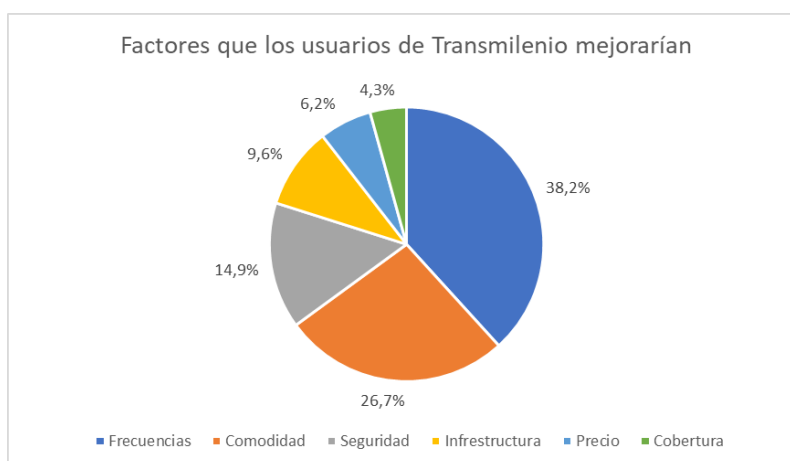
Aunque con el sistema BRT se busca proporcionar una alternativa para mejorar la movilidad en las ciudades, en la realidad, esto se ha conseguido cumplir a cabalidad. Según un estudio realizado en la ciudad de Bogotá, se logró identificar las expectativas y atributos de calidad esperados por parte de los usuarios del sistema BRT (Transmilenio) y del transporte público colectivo (Rincon-García, Navarro-Gomez, Alvarado, Aguirre-Mayorga., Salazar-Arrieta, 2016), y su preferencia con respecto a los sistemas de transporte ya mencionados y

la posible implementación del metro. Por medio de la aplicación de un cuestionario semiestructurado a 719 usuarios de transporte público, la mayoría de los usuarios del sistema BRT y de transporte público colectivo tienen una mayor inclinación a la construcción del metro, con un 65% y 74% respectivamente, debido a que las expectativas que se tienen con los sistemas existentes no son cumplidas y los atributos de calidad tampoco son los esperados. Los resultados obtenidos en el estudio son clasificados en dos tipos de atributos de calidad: físicos y percibidos por los usuarios, los cuales muestran claramente las falencias que tiene el sistema de transporte en la actualidad. Estos resultados son explicados en la ilustración 2.



*Ilustración 2. Atributos de calidad físicos y percibidos del sistema de transporte en Bogotá*  
*Fuente: Rincon-Garcia et al, 2016*

Adicionalmente, en un artículo publicado el año 2018 se definió el valor de los usuarios y la satisfacción con respecto a Transmilenio desde una aproximación a los principios del Lean Thinking (García-Suarez, Rivera-Perez, Rodriguez-Valencia, 2018). La metodología de este estudio se realizó desde un enfoque cualitativo y uno cuantitativo, por entrevistas y encuestas a usuarios de Transmilenio respectivamente. Los resultados mostraron gran similitud con el estudio mencionado anteriormente, pues se mencionan, en su mayoría, los mismos factores que los usuarios esperan del sistema de transporte y aumentaría su satisfacción con respecto a este. Estos resultados son mostrados en la ilustración 3. Por otro lado, según un informe de movilidad de Bogotá Cómo Vamos (2018), el 63% de los encuestados está insatisfecho con el sistema troncal de Transmilenio para el año 2018, un porcentaje con alto incremento comparado con el año 2008, cuya insatisfacción estaba cerca del 50%.



*Ilustración 3. Factores que los usuarios de TransMilenio mejorarían.*  
*Fuente: Garcia-Suarez et al, 2018*

Con el fin de proporcionar soluciones que contribuyan a mejorar la movilidad y el tránsito en las ciudades, se han propuesto diferentes metodologías y diseños de algoritmos basados en la optimización de operaciones

para la red de transporte, más específicamente, para el Transit Network Design Problem (TNDP). Dentro de las alternativas propuestas para problemas de TNDP, se ha empleado con frecuencia el algoritmo genético que es una herramienta de optimización que trabaja sobre el principio de evolucionar mejores soluciones, al intercambiar características de buenas soluciones existentes (Chakroborty, 2003). De acuerdo con Agrawal y Mathew (2004), esta metodología de optimización es empleada para resolver problemas complejos, utilizando un conjunto de soluciones posibles para una función objetivo, las cuales representan una codificación del problema como una forma analógica a los cromosomas de un sistema biológico. Este algoritmo es frecuentemente implementado para problemas multiobjetivo, debido a su simplicidad y robustez, como lo es el TNDP.

El algoritmo genético fue implementado por Jitendra Agrawal y Tom V. Mathew en el 2004, para generar un conjunto de rutas y frecuencias para la red de tránsito de la ciudad de Nueva Delhi, India. Esta red está compuesta por 1322 estaciones de bus (nodos) y 4076 arcos totales entre los nodos, que, en estudios anteriores, con la ayuda de herramientas computacionales, arrojó como resultado un total de 651 rutas, un tamaño de flota de 4609 buses, una longitud de ruta de 9,2 km y una frecuencia promedio de 5,5 buses por hora. Con la implementación del algoritmo genético se logró una disminución en: el número de rutas generadas a 638 y el tamaño de la flota de buses a 4571, obteniendo una satisfacción de la demanda del 96% y un tiempo promedio de viaje de 27 minutos.

Otra implementación del algoritmo genético fue realizada por Shashi Bhushan Jha, J.K. Jha y Manoj Kumar Tiwari (2019), tomando como referencia la red suiza estudiada por Christopher Mandl (1980) compuesta por 15 paradas y una demanda de 15.570 viajes por día. Jha et al (2019) plantean el problema de esta red como una variación del TNDP, llamada TNDFS (Transit Network Design and Frequency Setting), y, a partir de esto, el problema se divide en dos etapas. La primera corresponde al diseño de la red tránsito, compuesta por un conjunto de rutas que mejoren la cobertura del servicio y reduzca el tiempo promedio de viaje para los pasajeros. La segunda está asociada a la frecuencia de buses en cada ruta para minimizar el tiempo de espera y los costos de operación de manera simultánea. Los resultados obtenidos para ambas fases son comparados con resultados existentes de diferentes estudios y autores sobre el caso Mandl (1980). Para la fase del diseño de red, se realizaron diferentes conjuntos de ruta, compuestas desde 4 hasta 12 nodos, permitiendo comparar el porcentaje de viajes directos, de viajes con trasbordos desde 1 hasta 3, y el tiempo de viaje incluyendo los tiempos de trasbordo. Para la fase de configuración de frecuencias, se realizó la comparación mediante el Frente de Pareto para cada tamaño de ruta y los resultados existentes, buscando la relación entre el costo de operación y el tiempo de los pasajeros en la ruta. Finalmente, los resultados obtenidos arrojaron que el mejor tamaño de ruta es de 12, ya que el porcentaje de viajes directos corresponde a un 99,68% y un 0,32% para viajes con un trasbordo y su tiempo de viaje asociado corresponde a 10,08 minutos, y, por consiguiente, obtuvo el mejor Frente de Pareto con relación a costos de operación y tiempo del pasajero en la ruta (Jha et al, 2019).

Por otro lado, la capacidad de pasajeros en el BRT juega un papel fundamental, pues, como se observó en los resultados obtenidos en las encuestas de percepción de calidad del servicio en los sistemas de transporte de Rincon-Garcia et al (2016) y Garcia-Suarez et al (2018), los usuarios lo consideran como un factor importante en el momento de elegir uno de los modos de transporte disponibles. Hidalgo, Lleras y Hernández (2012) plantean una metodología para calcular la capacidad de pasajes en el sistema BRT de Bogotá (Transmilenio), teniendo en cuenta el número de paradas, el tiempo de abordaje y salida de los pasajeros, el intervalo de tiempo entre dos buses sucesivos, el porcentaje de buses que no paran en las estaciones, la capacidad máxima de los buses y la saturación aceptable a nivel de colas y la capacidad de recibirlas en las estaciones.

Configuración de estaciones	Capacidad (Pax/hora)
3 Plataformas sencillas	42.915
2 Plataformas dobles y una sencilla - Despacho coordinado	55.710
2 Plataformas dobles y una sencilla - Despacho no coordinado	45.309
2 Plataformas dobles y una sencilla - Despacho no coordinado, sin servicios asignados a plataformas específicas	48.809

*Tabla 1. Capacidad de pasajeros en las estaciones  
Fuente: Hidalgo et al (2012)*

De acuerdo con la metodología y la formulación planteada por Hidalgo et al (201), las capacidades correspondientes a las diferentes configuraciones de estaciones son presentadas en la tabla 1, mostrando que la mejor configuración de estación es aquella en la que se manejan dos plataformas dobles y una sencilla, sin coordinados y los servicios no son asignados a plataformas específicas, ya que soporta una capacidad de 48.809 pasajeros/hora/sentido. Además, esta configuración es la más conveniente, debido a que la forma en la que opera se ajusta al funcionamiento real del sistema de Transmilenio al contar con servicios que comparten plataformas y tienen diferentes orígenes y paradas a lo largo de la ruta (Hidalgo et al, 2012), la cual es la diferencia principal con una configuración de despachos coordinados.

Tomando como referencia la información anterior, es posible concluir que el TNDP para el sistema BRT es un reto en el que se ven involucradas diferentes variables y factores importantes, tanto para los usuarios como para los operadores del servicio. Es por esto, que el presente proyecto de grado es presentar una metaheurística que dé respuesta al TNDP para el sistema de Transmilenio en la ciudad de Bogotá, con el fin de proporcionar una nueva serie de rutas, una mejor asignación de frecuencias para los buses y una adecuación para las estaciones troncales, teniendo en cuenta los buses articulados y biarticulados, con el fin de aumentar la capacidad de usuarios a atender y, por consiguiente, aumentar la satisfacción del usuario, sin descuidar los objetivos de los operadores de servicio, como lo es el costo de operación, la cantidad de buses a utilizar y la adecuación de sus estaciones troncales.

### 3. Objetivos

#### General:

Diseñar un aplicativo para el problema de optimización multiobjetivo de un sistema BRT que permita determinar rutas y frecuencias con el fin de minimizar tanto los tiempos de los usuarios en el sistema y la ocupación media de los buses como el número de buses necesarios, mostrando la operación del sistema a través de una interfaz de visualización.

#### Específicos:

- 1) Diseñar una interfaz empleando un sistema de información geográfico para la visualización de la operación de un sistema BRT.
- 2) Implementar una metaheurística para el diseño de rutas con características multiobjetivo teniendo en cuenta la calidad del servicio para el usuario y el costo de operación de la flota para un sistema BRT.
- 3) Medir el impacto de los resultados obtenidos por el conjunto de rutas propuestas, frente al servicio prestado actualmente en el sistema BRT de Transmilenio.

### 4. Cuerpo del documento

El presente trabajo de grado se divide en tres objetivos específicos, sin embargo, el tercer objetivo hace referencia al impacto del desarrollo del trabajo, por esta razón solo se toma en cuenta, para este apartado, los

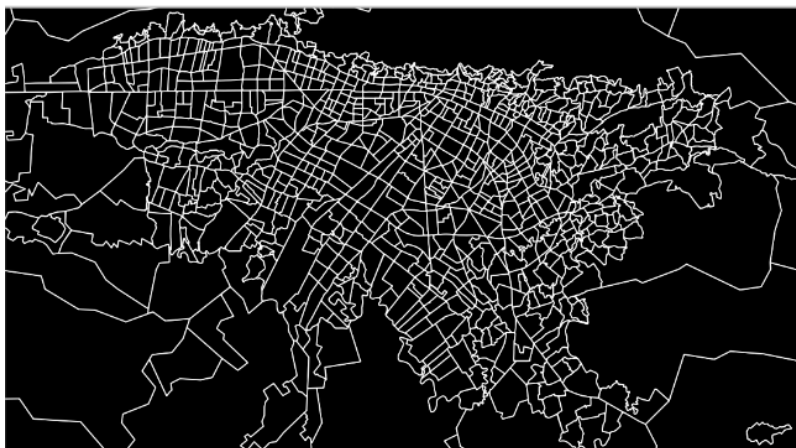
objetivos uno y dos que hacen referencia a la interfaz de visualización y el código de optimización respectivamente.

#### 4.1 Interfaz de visualización

La visualización del sistema se realiza en una aplicación que permita hacer un seguimiento georreferenciado de los buses a lo largo de los recorridos, para este trabajo se decidió utilizar la aplicación QGIS, la cual es un sistema de información geográfica de software libre que trabaja con Python como su lenguaje de programación.

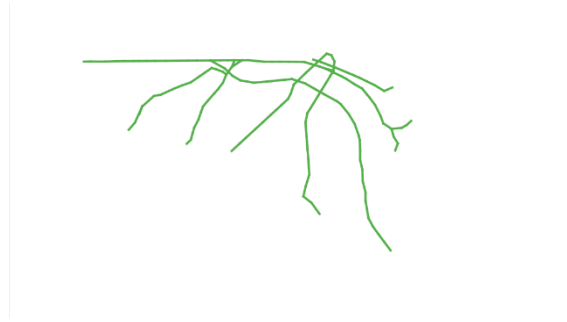
Para el desarrollo de la visualización se debe generar en primer lugar el mapa o plano que contendrá la información tanto de las estaciones como de las vías troncales. En este caso el mapeo consiste en la creación de 3 capas distintas. La primera capa es el mapa de la ciudad de Bogotá, la segunda capa son las vías troncales del sistema Transmilenio y la tercera capa hace referencia a los nodos que representan cada una de las 149 estaciones en el sistema.

La primera capa se crea con base en las bibliotecas con las que cuenta la aplicación de QGIS, descargando de esta manera el croquis vial de la ciudad de Bogotá como se puede ver a continuación:



*Ilustración 4. Capa 1 - Croquis vial de la ciudad de Bogotá en QGIS  
Fuente: Autoría propia*

Una vez se tiene la capa inicial se procede a la creación de las capas 2 y 3. Para esto se descarga la ubicación georreferenciada para cada una de las estaciones que hacen parte del sistema troncal, cabe aclarar que existen estaciones con longitudes diferentes, por esta razón y con la intención de facilitar el proceso se decidió tomar como punto de referencia lo que sería aproximadamente el punto medio de cada estación. Una vez se tienen las coordenadas estas se cargan a la aplicación de tal manera que genere un nodo en cada una de las ubicaciones y se guarda esta como la capa 3. Para la capa dos se utiliza como base la capa 3 previamente realizada y se conectan a través de vectores cada par de nodos, esto cuidando la realidad del sistema, para asegurar que este proceso se hiciese de manera correcta se realizó con la ayuda de una matriz que marca los nodos que tienen conexión y los que no la tienen, de esta manera se pueden ver las dos capas a continuación.

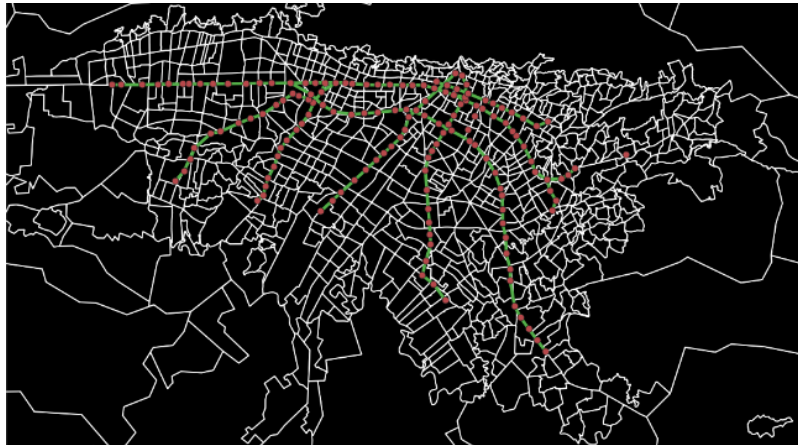


*Ilustración 5. Capa 2 - Arcos de conexión de nodos en QGIS*  
*Fuente: Nicolás Rincón García (2019)*



*Ilustración 6. Capa 3 - Nodos que simulan las estaciones de Transmilenio en QGIS*  
*Fuente: Nicolás Rincón García (2019)*

Finalmente se activan las tres capas al mismo tiempo en su respectivo orden, capa 1, capa 2 y capa 3; las cuales hacen referencia al croquis, las troncales y las estaciones respectivamente con el fin de tener el resultado que se muestra a continuación:



*Ilustración 7. Representación de la ciudad de Bogotá y el sistema de Transmilenio en QGIS*

*Fuente: Nicolás Rincón García (2019)*

Una vez se tiene el mapa terminado se puede empezar a cargar las diferentes rutas que se quieren visualizar. Para la realización de este proceso se debe de recibir dos datos, en primer lugar, se debe saber cuál es la ruta que se quiere visualizar, es decir un vector que determine el nodo inicio, el nodo fin y cada uno de los nodos en los cuales para la ruta. En segundo lugar, se necesita la frecuencia de la ruta, dado que el horizonte de estudio es de una hora, la frecuencia determinara cada cuanto parte un bus para realizar ese recorrido específico.



Ahora bien, generar la visualización completa las rutas deben ser ingresadas una por una para que cada cual quede definida dentro de una capa diferente, una vez se tienen las capas de las diferentes rutas que se quieren visualizar se procede a cargar estas capas en la extensión de QGIS llamada Time Manager, esta se encarga de generar la animación según las especificaciones que se hayan dado en las capas cargadas. En este sentido la creación de las capas de rutas se realiza de la siguiente manera:

1. Se genera una capa nueva y vacía la cual sirve como base para la ruta.
2. Se lee el vector que contiene los nodos de la ruta en un nuevo vector el cual es llenado con todos los nodos intermedios entre cada par de nodos, es decir si el vector inicia es [1,3,4,5,8] el nuevo vector sería [1,2,3,4,5,6,7,8]. Esto se hace con el fin de que la visualización siga la ruta asignada y se mantenga sobre las vías troncales.
3. Los nodos del nuevo vector leen las coordenadas geográficas del nodo que corresponde desde la capa 3 de estaciones que fue creada al inicio.
4. Una vez se tiene la ubicación para cada uno de los nodos se calcula el tiempo de viaje que transcurre entre un nodo y otro, para esto se usan sus coordenadas y se calculan los kilómetros entre cada par de estaciones, luego teniendo en cuenta una velocidad promedio de 23 km/h se calcula el tiempo de viaje.
5. Una vez se tiene el tiempo de duración de una ruta y la hora a cuál visita cada uno de los nodos se procede a cargar la ruta en la extensión Time Manager, en la cual se pide al programa que genere una animación la cual muestre la transición de un punto (como representación del bus) entre cada par de nodos teniendo en cuenta la hora en la que pasa por cada uno, dato que se estipuló anteriormente.

Este proceso debe realizarse tantas veces como rutas quiera visualizarse al mismo tiempo. Una vez se tengan montadas cada una de las rutas se espera ver el camino de cada una sobre el mapa como se muestra a continuación.

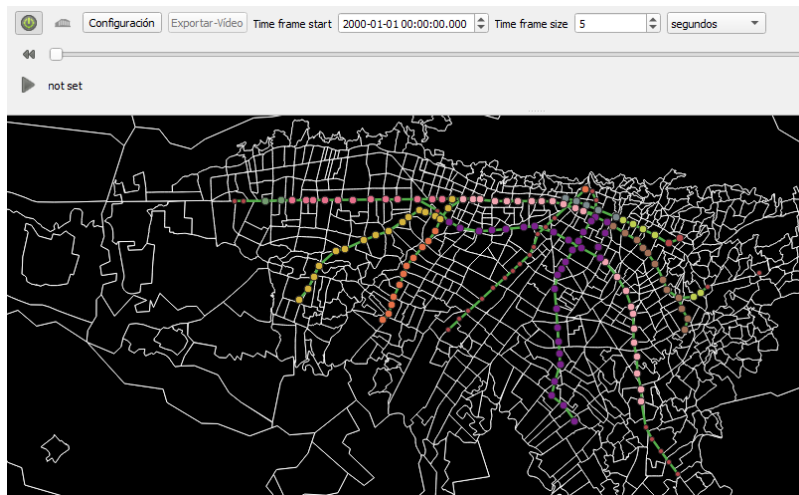


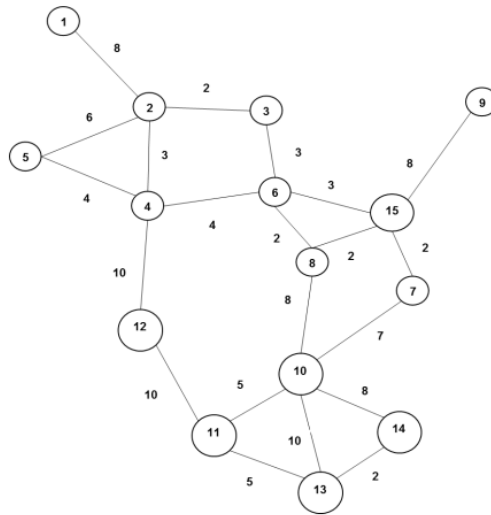
Ilustración 8. Visualización de los nodos que componen una ruta en QGIS  
Fuente: Autoría propia

Una vez empiece a correr la visualización del mapa se eliminarán todos los nodos y solo aparecerán los que estén activos en ese momento. Durante el tiempo entre nodos se ve el círculo (que representa al bus) moverse por sobre la vía troncal. Fura de esto se determina que los nodos donde para cada ruta tengan un tiempo de pausa o parada de 30 segundos para así mostrar cuando se detienen los buses y donde lo hacen.

#### 4.2 Código de optimización

Para el correcto desarrollo del código de optimización del caso de estudio, se realizó una búsqueda previa de información sobre estudios anteriores que hayan trabajado con la temática del TNDP. En esta búsqueda, se

destaca el estudio de Mandl (1979), que es un problema que se ha utilizado para medir la intensidad de los algoritmos de optimización en este tipo de problemas. Este caso de ejemplo cuenta con una red de 15 nodos y 21 arcos como se muestra a continuación y cuenta con una matriz de demanda origen-destino simétrica.



*Ilustración 9. Representación de la red de transporte del caso Mandl*  
Fuente: Mandl (1979)

Con el motivo de medir la capacidad del algoritmo de optimización propuesto se desarrolla un código inicial para abordar este caso de ejemplo y, luego, se realizan las modificaciones necesarias para adaptar el código al caso de estudio Transmilenio. Cabe resaltar que el ejemplo representa un problema TNDP, sin embargo, no es un sistema BRT como lo es Transmilenio, por esta razón se hace imprescindible la diferenciación entre los dos tipos de problemas, ya que, para el caso de estudio de Transmilenio, se cuenta con una red de 149 nodos y 159 arcos y una matriz de demanda origen-destino no simétrica.

Este código de optimización, tanto para el caso de Mandl como para el caso de estudio del sistema BRT, cuenta con tres fases:

- 1) De construcción o creación de rutas: Crea un conjunto de rutas que conforman una solución inicial, en donde se manejan, para ambos casos, parámetros que toman valor de manera aleatoria y que contribuyen a una mayor variedad para las rutas a generar.
- 2) De evaluación: Evalúa la función objetivo de la solución inicial mediante el modelo de asignación generado.
- 3) De mejora: Realiza modificaciones a la solución inicial teniendo en cuenta la variable de decisión de Frecuencias para cada una de las rutas y determinar si eliminarlas o mantenerlas en la solución inicial.

A continuación, se presentan los algoritmos respectivos para los problemas TND y BRT, es decir, el caso de Mandl y el de Transmilenio respectivamente:

#### **4.2.1 Problema TND – Caso de Mandl**

Como se explicó anteriormente, el caso de Mandl, para el problema TND, se tomó como referencia para medir la capacidad del algoritmo de optimización propuesto en este trabajo, mediante la comparación de los resultados obtenidos con respecto al de otros autores en estudios previos. Esta comparación se basa en los indicadores del tiempo de funcionamiento total del sistema y la cantidad de buses necesarios para satisfacer la totalidad de la demanda, cuya notación es asignada como Z1 y Z2 respectivamente. Adicionalmente, los algoritmos desarrollados para el caso de Mandl son la estructura base para el caso de estudio de este trabajo, Transmilenio, con sus respectivos ajustes de acuerdo a la complejidad de la red de transporte.

#### 4.2.1.1 Fase de construcción

Inicialmente, para el caso de Mandl, el algoritmo realiza una construcción de rutas “express”, es decir, rutas con nodos seguidos inmediatamente, en el cual se prioriza el cubrimiento de la mayor demanda posible y se establece un tiempo de recorrido total máximo ( $TiempoMax_i$ ) de 25 a 35 minutos por ruta, tiempo elegido de manera aleatoria. Este tiempo máximo de recorrido es implementado como una diversificación para el algoritmo, pues permite realizar una mayor exploración de nodos a agregar y aumentar la posibilidad de construir rutas diferentes.

A continuación, se presentan los datos de entrada, el pseudocódigo del algoritmo de generación de rutas:

Algoritmo 1. Generación de rutas – Caso Mandl

---

**Datos de entrada:** Demandas = Matriz de demandas entre par de nodos (i,j)  
MatrizD = Matriz de tiempos de viaje entre par de nodos (i,j)  
Paths = Camino más corto entre par de nodos (i,j)  
ParesOrdenados = Lista de pares de nodos ordenada descendentemente por demanda  
Grupos = Lista para cada nodo con las posiciones en las que aparece en ParesOrdenados  
Verificado = Lista de pares de nodos que no han sido utilizados

**Inicio**

1. **Mientras**  $i \leq NumRutas$  **hacer**
2. Inicializar  $TiempoRuta_i$ ,  $TiempoMax_i$  y  $NoVacía$  ( $NoVacía = 0$ )
3. Inicializar  $Ruta_i$  como vacía
3. **Mientras**  $j \leq ParesOrdenados$  **hacer**
4. **Si**  $NoVacía = 0$  **entonces**
5. **Si**  $Verificado_j = 0$  **entonces**
6. Agregar  $Paths_j$  en  $Ruta_i$
7.  $Verificado_j = 1$
8. Actualizar  $SumaTiempo_i$
9. **Terminar si**
10. **Terminar si**
11. **Si**  $NoVacía = 1$  **entonces**
12.  $Primero \leftarrow$  Nodo en la primera posición de  $Ruta_i$
13.  $Último \leftarrow$  Nodo en la última posición de  $Ruta_i$
14. Buscar en  $ParesOrdenados$  un par de nodos disponible que contenga a  $Primero$  ( $Primero, m$ ) y otro a  $Último$  ( $Último, n$ ) y que  $Paths_{Primero, m}$  y  $Paths_{Último, n}$  no tenga nodos en común con  $Ruta_i$
15.  $DemTimePrimero \leftarrow Demandas_{Primero, m} / MatrizD_{Primero, m}$
16.  $DemTimeÚltimo \leftarrow Demandas_{Último, n} / MatrizD_{Último, n}$
17.  $SumaPrimero \leftarrow MatrizD_{Primero, m}$
18.  $SumaÚltimo \leftarrow MatrizD_{Último, n}$
19. **Si**  $DemTimePrimero \geq DemTimeÚltimo$  **entonces**
20. **Si**  $TiempoRuta_i + SumaPrimero \leq TiempoMax_i$  **entonces**
21. Agregar  $Paths_{Primero, m}$  a  $Ruta_i$
22.  $TiempoRuta_i \leftarrow TiempoRuta_i + SumaPrimero$
23.  $Verificado_{Primero, m} = 1$
24. **Terminar si**
25. **De lo contrario**
26. **Si**  $TiempoRuta_i + SumaÚltimo \leq TiempoMax_i$  **entonces**
27. Agregar  $Paths_{Último, n}$  a  $Ruta_i$
28.  $TiempoRuta_i \leftarrow TiempoRuta_i + SumaÚltimo$
29.  $Verificado_{Último, n} = 1$
30. **Terminar si**
31. **Terminar si**
32. **Terminar si**
33. **Terminar Mientras**
34. Agregar  $Ruta_i$  a  $RutasSolucion$
35. **Terminar Mientras**

Fin

---

<i>NumRutas</i> : Número de rutas a crear	<i>Ruta<sub>i</sub></i> : Ruta <i>i</i> que compone la solución
<i>TiempoRuta<sub>i</sub></i> : Contador de tiempo total de la ruta <i>i</i>	<i>TiempoMax<sub>i</sub></i> : Tiempo máximo para la ruta <i>i</i>
<i>NoVacía</i> : Señal que indica si una ruta es nueva	<i>Primero</i> : Nodo de la primera posición de Ruta <sub><i>i</i></sub>
<i>Último</i> : Nodo de la última posición de Ruta <sub><i>i</i></sub>	

---

#### 4.2.1.2 Fase de evaluación

Para la evaluación de rutas se debe seleccionar un algoritmo de asignación, el cual determinara la manera en la cual se distribuye la demanda entre las rutas y asegura así el mayor cumplimiento de la demanda posible según los parámetros estipulados. En este caso se ha seleccionado como base el algoritmo de asignación de Baaj y Mahmassani (1991), el cual genera un conjunto de frecuencias para un determinado conjunto de rutas. Este algoritmo recibe como parámetros los siguientes datos:

- Red G (Identifica la unión entre los distintos nodos)
- Matriz de demanda origen-destino entre cada par de nodos D
- Solución para evaluar, compuesta por un conjunto de rutas y sus frecuencias respectivas  $S = (R,F)$

Cabe aclarar que las frecuencias de entrada para todas las soluciones se representan como un vector de 1 ( $F=[1,1,\dots,1]$ ) esto debido a que el código necesita un valor inicial que irá corrigiendo a medida que itera.

Una vez se ejecuta el código se obtienen diferentes valores, los cuales no se reducen únicamente a las variables sugeridas en el algoritmo de Baaj y Mahmassani (1991), sino que se han agregado más variables con el fin de mejorar el análisis para las diferentes soluciones encontradas. Dentro de las variables de salida se encuentran:

- *tv*: Tiempos de viaje totales en bus.
- *te*: Tiempo de espera totales según frecuencia.
- *tt*: Penalidad por transbordo que corresponde a  $p_t$  multiplicado por el número de viajes.
- $N_k$ : Corresponde al número de buses necesarios para cubrir la ruta  $r_k$  de la solución S. Al sumar todos los  $N_k$  se obtiene el número total de buses necesarios para cubrir la demanda del sistema.
- Proporción de demanda cubierta de manera directa, cubierta con un transbordo y no cubierta,  $D_0$ ,  $D_{01}$  y  $N_0$  respectivamente.
- $\Phi$ : Representa el flujo de personas por cada arco o unión entre cada par de nodos. Este flujo se da tanto en cantidad de demanda como en número de buses en tránsito.
- *Ocu*: Representa el porcentaje de ocupación media de cada uno de los buses que realizan los diferentes recorridos. Además, se presenta el porcentaje de ocupación máxima que llevan los buses en cada uno de los recorridos.
- *Km*: Se muestran los kilómetros totales recorridos en un conjunto solución.
- Frecuencias mínimas factibles para cada uno de los recorridos en una solución.

En la imagen que se muestra a continuación, se presenta el pseudocódigo del algoritmo de asignación implementado, los datos de entrada y los datos de salida. De igual manera, se presenta la aclaración pertinente para las siguientes partes del algoritmo:

Algoritmo 2. Evaluación de rutas

---

**Datos de entrada:** Demandas = Matriz de demandas entre par de nodos (i,j)  
MatrizD = Matriz de tiempos de viaje entre par de nodos (i,j)  
SRutas = Matriz que contiene las rutas pertenecientes a una solución  
FR = Frecuencias relacionadas a cada una de las rutas en una solución

**Inicio**

1.  $FR_{sal} \leftarrow FR$
2. **Hacer**
3.  $FR_{ent} \leftarrow FR_{sal}$

4. **Para i en n**
  5. **Para j en n**
  6.  $Dir_{ij} \leftarrow R_{ij}$
  7.  $P_{ijk} \leftarrow$  Proporción de la demanda del nodo i al j atendida por la ruta k
  8. **Si**  $|Dir_{ij}| > 0$  **entonces**
  9. Actualizar  $D_0$
  10. Actualizar  $T_v, T_e, km$
  12. Actualizar  $\Phi$
  13. **sino**
  14.  $Tr_{ij} \leftarrow$  Recorridos con un nodo en común que una la ruta de i y j
  15.  $n_t \leftarrow$  Nodo de transbordo
  16.  $P_{ijk} \leftarrow$  Proporción de la demanda del nodo i al j atendida por la ruta k
  17. **Si**  $|Tr_{ij}| > 0$  **entonces**
  18. Actualizar  $D_{0i}$ ;
  19. Actualizar  $T_v, T_e, T_t, km$
  20. Actualizar  $\Phi$
  21. **sino**
  22. Actualizar No
  23. **Terminar si**
  24. **Terminar si**
  25. **Siguiente j**
  26. **Siguiente i**
  27. Calcular  $F_{sal}, Ocu, N_k$
  28. **Hasta**  $F_{ent}$  Y  $F_{sal}$  no sean significativamente diferentes
  29. **Imprimir** ( $F_{sal}, Ocu, N_k, No, T_v, T_e, km, \Phi, D_{0i}, D_0$ )
- Final**

---

$Dir_{ij}$ : Recorridos en los que se encuentra el nodo i y el nodo j  
 $Tr_{ij}$ : Recorridos con un nodo en común que una la ruta de i y j  
 $P_{ijk}$ : Proporción de la demanda del nodo i al j atendida por la ruta k  
 $n_t$ : Nodo de transbordo  
n: Numero de nodos totales

---

**Línea 07:** Algoritmo de asignación directa.

Se verifica si existen recorridos comunes entre los conjuntos de rutas que pasan por i y los que pasan por j. En caso de no existir recorridos comunes, el vector  $Dir_{ij}$  quedara vacío. De lo contrario, se busca aquél recorrido que une a i con j que tenga el menor costo posible  $tv_{ij}$ . Si hay recorridos alternos, se seleccionan los que unen a i con j a un costo menor que  $tv_{ij} * \sigma_d$ . Tomando el conjunto restante de recorridos  $Dir_{ij}$ , se realiza la asignación de proporciones  $P_{ijk}$  de la demanda  $d_{ij}$  para cada recorrido  $r_k \in Dir_{ij}$  de acuerdo con las frecuencias de cada uno, usando la siguiente expresión:

$$P_{ijk} = \frac{f_k}{\sum_{r_m \in R_{ij}} f_m}$$

**Líneas 09-12:** Actualización de indicadores.

- Demanda satisfecha directamente:  $D_0 = D_0 + \frac{d_{ij}}{D_{tot}}$ ; donde  $D_{tot} = \sum_{i,j \in [1, \dots, n]} d_{ij}$
- $tv_{ij} = \sum_{r_k \in R_{ij}} P_{ijk} t_k^{ij}$ , donde  $t_k^{ij}$  es el tiempo de viaje en el vehículo por recorrido  $r_k$  entre el par de nodos (i,j).
- Tiempo de espera:  $te_{ij} = \frac{1}{2 \sum_{r_k \in R_{ij}} f_k}$

- $\Phi_{a,r_k} = \Phi_{a,r_k} + P_{ijk}d_{ij}$  para cada arco entre los nodos  $i$  y  $j$  de cada recorrido  $r_k$  del conjunto de líneas componentes  $Dir_{ij}$ . La actualización de flujos debe tener en cuenta el sentido del recorrido, donde el nodo  $i$  está antes del nodo  $j$ ; de esta forma se actualizarán solamente flujo en el sentido de ida o vuelta.

**Líneas 14-16:** Asignación de transbordo.

En caso de no existir una ruta que conecte directamente al par de nodos  $i,j$ , es necesario verificar los transbordos posibles, buscando los nodos en común entre los recorridos de los conjuntos  $R_i$  y  $R_j$ . De esta manera, se identifica un conjunto de trayectorias de transbordo posibles, donde cada trayectoria  $T$  será de la forma  $r_s n_t r_l$ , donde  $r_s$  y  $r_l$  son recorridos que pasar por  $i$  y  $j$  respectivamente (salida y llegada), y  $n_t$  es el nodo de transbordo. Al igual que en la asignación directa, se halla la trayectoria de transbordo más corta y se filtran aquellas que exceden el tiempo de viaje en vehículo en un factor  $\sigma_t$  del tiempo de la trayectoria más corta. Una vez obtenido el conjunto  $Tr_{ij}$  de trayectorias de transbordo competentes, se realiza la asignación a las distintas alternativas. Para eso, en primer lugar, el conjunto  $Tr_{ij}$  se particiona en clases  $C_{ij} = (C_1, \dots, C_l)$ , donde los elementos de una misma clase son tales que tienen en común el recorrido de salida de su trayectoria. De esta forma, la demanda se distribuye, en principio, entre las clases de la partición de forma proporcional a la frecuencia de los recorridos de salida de cada  $P_{ijk}$ . Dentro de una clase, la demanda se distribuye de manera uniforme entre los recorridos de la llegada  $P_{ijkm}$ .

**Líneas 18-20:** Actualización de indicadores.

- Demanda satisfecha con transbordos:  $D_{01} = D_{01} + \frac{d_{ij}}{D_{tot}}$ ; donde  $D_{tot} = \sum_{i,j \in [1, \dots, n]} d_{ij}$
- $tv_{ij} = \sum_{C_k \in C_{ij}} P_{ijk} (\sum_{T_m \in C_k} P_{ijkm} (t_s^{it} + t_l^{tj}))$ , donde  $t_s^{it}$  es el tiempo de viaje en un vehículo por recorrido de salida de la trayectoria  $T_m$  de la clase  $C_k$  entre los nodos  $i$  y  $n_t$ ,  $t_l^{tj}$  es el tiempo de viaje en vehículo por el recorrido de llegada de la trayectoria de transbordo  $T_m$  de la clase  $C_k$  entre los nodos  $n_t$  y  $j$ . Cada trayectoria de transbordo  $T_m$  tiene un nodo de transbordo  $n_t$  diferente.
- $te_{ij} = \frac{1}{2 \sum_{C_k \in C_{ij}} f_s} + \sum_{C_k \in C_{ij}} P_{ijk} \frac{\sum_{T_m \in C_k} P_{ijkm}}{2f_l}$ , donde  $f_s$  es la frecuencia del recorrido de salida de la clase  $C_k$  y  $f_l$  es la frecuencia del recorrido de llegada de la trayectoria de transbordo  $T_m$  de la clase  $C_k$ .
- Tiempo de transbordo:  $tt_{ij} = p_t$
- $\Phi_{a,r_k} = \Phi_{a,r_k} + P_{ijk}d_{ij}$ , para cada arco  $a$  entre los nodos  $i$  y  $n_t$  de los recorridos  $r$  de salida de cada clase  $C_k$ .
- $\Phi_{a,r_k} = \Phi_{a,r_k} + P_{ijk}P_{ijkm}d_{ij}$ , para cada arco  $a$  entre los nodos  $n_t$  y  $j$  de los recorridos  $r$  de llegada a cada trayectoria de cada clase.

**Línea 22:** Actualización de demanda no satisfecha.

De no ser posible asignar la demanda entre el par de nodos ( $i,j$ ) realizando un transbordo, la demanda se considera insatisfecha y se actualiza de la siguiente manera:

$$D_{NS} = D_{NS} + \frac{d_{ij}}{D_{tot}}$$

Al asignar la totalidad de la demanda, se realiza el cálculo de las frecuencias necesarias para cada recorrido, teniendo en cuenta el no sobrepasar el factor de carga máxima  $LF_{max}$  especificado en el modelo de optimización, de la siguiente manera:

$$f_k = \frac{\phi_k^*}{LF_{max} CAP}$$

Además, se obtiene la cantidad de buses necesarios para operar en cada recorrido  $r_k$ :

$$N_k = f_k t_k$$

### 4.2.1.3 Fase de mejora

Al realizar una comparación entre los resultados propuestos por el algoritmo para el caso Mandl y aquellos de otros autores y estudios consultados en la literatura, se encontró una mejora considerable con algunas de las soluciones propuestas a nivel de tiempo de funcionamiento del sistema (Z1) y el número de buses requeridos para satisfacer la demanda (Z2), es decir, se logró una disminución para estos indicadores haciendo uso solo de la fase de construcción de rutas y de evaluación de las mismas. Gracias a lo anterior, no se realizó una fase de mejora para el caso TND de Mandl, ya que, como se explicó anteriormente, se cumplió el propósito de disminuir estos indicadores y, por otro lado, la fase de mejora propuesta para el problema BRT de Transmilenio, ajustada al caso Mandl, no obtuvo mejoras significativas.

## 4.2.2 Problema BRT – Caso Transmilenio

Para el problema BRT de Transmilenio, se tomó como referencia los algoritmos propuestos para el caso TND, gracias a la comparación de los resultados obtenidos y contrastados con estudios previos, demostrando su capacidad a nivel de construcción de rutas y su posterior evaluación. Teniendo en cuenta lo anterior, para el caso de Transmilenio se realizaron los ajustes necesarios en los algoritmos de construcción y de evaluación de rutas, dada la alta complejidad de la red de transporte, además de la implementación de la fase de mejora para las soluciones generadas inicialmente. Cabe resaltar que, para este caso, la fase de mejora cobra gran importancia, pues además de buscar minimizar el tiempo total de funcionamiento del sistema (Z1) y la cantidad de buses para satisfacer la demanda (Z2), se tiene como propósito la disminución de rutas para el sistema y la ocupación media y máxima de personas por metro cuadrado para cada tipo de bus.

### 4.2.2.1 Fase de construcción

Para la creación de rutas para el sistema BRT, se realizó un algoritmo cuyo procedimiento es similar al del TND, pero contiene unos pequeños cambios referentes a los criterios de selección de los grupos de nodos iniciales que componen cada una de las rutas. Sin embargo, se mantiene la prioridad de cubrir la mayor cantidad de demanda posible y un tiempo máximo total de recorrido (TiempoMax<sub>i</sub>) de 40 a 60 minutos por ruta, tiempo establecido aleatoriamente para cada una de ellas, implementado como un método de diversificación de las rutas, dando la posibilidad de obtener rutas con diferentes cantidades de nodos y explorar diferentes caminos a agregar en las mismas. A continuación, se presenta el pseudocódigo de este proceso de creación de rutas:

Algoritmo 3. Generación de rutas – Caso Transmilenio

---

**Datos de entrada:** Demandas = Matriz de demandas entre par de nodos (i,j)  
MatrizD = Matriz de tiempos de viaje entre par de nodos (i,j)  
Paths = Camino más corto entre par de nodos (i,j)  
ParesOrdenados = Lista de pares de nodos ordenada descendientemente por demanda  
Grupos = Lista para cada nodo con las posiciones en las que aparece en ParesOrdenados  
Verificado = Lista de pares de nodos que no han sido utilizados

**Inicio**

1. **Mientras**  $i \leq \text{NumRutas}$  **hacer**
2. Inicializar TiempoRuta<sub>i</sub>, TiempoMax<sub>i</sub>, NoVacia (NoVacia = 0) y Ruta<sub>i</sub> como vacía
3. **Mientras**  $j \leq \text{ParesOrdenados}$  **hacer**
4. **Si** NoVacia = 0 **entonces**
5. **Si** Verificado<sub>j</sub> = 0 **entonces**
6. **Mientras**  $k \leq |\text{Paths}_j|$  **hacer**
7. Nodo1  $\leftarrow$  Paths<sub>j,k</sub>
8. Nodo2  $\leftarrow$  Paths<sub>j,k+1</sub>
9. **Si** TiempoRuta<sub>i</sub> + MatrizD<sub>Nodo1,Nodo2</sub>  $\leq$  TiempoMax<sub>i</sub> **entonces**
10. TiempoRuta<sub>i</sub>  $\leftarrow$  TiempoRuta<sub>i</sub> + MatrizD<sub>Nodo1,Nodo2</sub>
11. Agregar Nodo1 a Ruta<sub>i</sub>
12. **Si**  $k = |\text{Paths}_j|$  **entonces**
13. Agregar Nodo2 a Ruta<sub>i</sub>
14. **De lo contrario**

15. Salir de Mientras (línea 7)

16. **Terminar si**

17. **Terminar Mientras**

18. Verificado<sub>i</sub> = 1

19. Inicializar CantidadVerificar y CantidadVerificados

20. **Mientras**  $l \leq |\text{ParesOrdenados}|$  **y**  $\text{CantidadVerificados} \leq \text{CantidadVerificar}$  **hacer**

21. **Si**  $\text{ParesOrdenados}_{j,0} = \text{ParesOrdenados}_{k,0}$  **y**  $\text{ParesOrdenados}_{j,1} = \text{ParesOrdenados}_{j,1}$  **entonces**

22. Verificado<sub>i</sub> = 1

23.  $\text{CantidadVerificados} \leftarrow \text{CantidadVerificados} + 1$

24. **Terminar si**

25. **Terminar Mientras**

26. **Si**  $\text{TiempoRuta}_i + 5 \leq \text{TiempoMax}_i$  **entonces**

27. NoVacía = 1

28. **De lo contrario**

29. NoVacía = 0

30. **Terminar si**

31. **Terminar si**

32. **Terminar si**

33. **Terminar Mientras**

34. **Mientras**  $j \leq 5$  **hacer**

35. **Si** NoVacía = 1 **entonces**

36. Primero  $\leftarrow$  Nodo en la primera posición de Ruta<sub>i</sub>

37. Último  $\leftarrow$  Nodo en la última posición de Ruta<sub>i</sub>

38. Buscar en ParesOrdenados un par de nodos disponible que contenga a Primero (Primero,m) y otro a Último (Último,n) y que Paths<sub>Primero,m</sub> y Paths<sub>Último,n</sub> no tenga nodos en común con Ruta<sub>i</sub>

39. DemPrimero  $\leftarrow$  Demandas<sub>Primero,m</sub>

40. DemÚltimo  $\leftarrow$  Demandas<sub>Último,n</sub>

41. Inicializar RutaAgregar

42. **Si** DemPrimero  $\geq$  DemÚltimo **hacer**

43. RutaAgregar  $\leftarrow$  Paths<sub>Primero,m</sub>

44. **Mientras**  $p \leq |\text{RutaAgregar}|$  **hacer**

45. Nodo1  $\leftarrow$  RutaAgregar<sub>p</sub>

46. Nodo2  $\leftarrow$  RutaAgregar<sub>p+1</sub>

47. **Si**  $\text{TiempoRuta}_i + \text{MatrizD}_{\text{Nodo1},\text{Nodo2}}$  **entonces**

48. Agregar Nodo1 a Ruta<sub>i</sub>

49.  $\text{TiempoRuta}_i \leftarrow \text{TiempoRuta}_i + \text{MatrizD}_{\text{Nodo1},\text{Nodo2}}$

50. **Terminar si**

51. **Terminar Mientras**

52. Verificado<sub>Primero,m</sub> = 1

53. Verificado<sub>m,Primero</sub> = 1

54. **De lo contrario**

55. RutaAgregar  $\leftarrow$  Paths<sub>Último,n</sub>

56. **Mientras**  $p \leq |\text{RutaAgregar}|$  **hacer**

57. Nodo1  $\leftarrow$  RutaAgregar<sub>p</sub>

58. Nodo2  $\leftarrow$  RutaAgregar<sub>p+1</sub>

59. **Si**  $\text{TiempoRuta}_i + \text{MatrizD}_{\text{Nodo1},\text{Nodo2}}$  **entonces**

60. Agregar Nodo1 a Ruta<sub>i</sub>

61.  $\text{TiempoRuta}_i \leftarrow \text{TiempoRuta}_i + \text{MatrizD}_{\text{Nodo1},\text{Nodo2}}$

62. **Terminar si**

63. **Terminar Mientras**

64. Verificado<sub>Último,n</sub> = 1

65. Verificado<sub>n,Último</sub> = 1

66. **Terminar si**

67. **Si**  $\text{TiempoRuta}_i + 5 \geq \text{TiempoMax}_i$  **entonces**

68. Salir de Mientras (línea 35)

69. Agregar Ruta<sub>i</sub> a RutasSolucion

70. **Terminar Mientras**

**Fin**



<i>NumRutas</i> : Número de rutas a crear	<i>Ruta<sub>i</sub></i> : Ruta <i>i</i> que compone la solución
<i>TiempoRuta<sub>i</sub></i> : Contador de tiempo total de la ruta <i>i</i>	<i>TiempoMax<sub>i</sub></i> : Tiempo máximo para la ruta <i>i</i>
<i>NoVacía</i> : Señal que indica si una ruta es nueva	<i>Primero</i> : Nodo de la primera posición de Ruta <sub><i>i</i></sub>
<i>Último</i> : Nodo de la última posición de Ruta <sub><i>i</i></sub>	<i>CantidadVerificar</i> : Cantidad máxima de pares a verificar (Valor aleatorio entre 30 y 60)

---

Debido a que el caso de estudio posee una red troncal mucho más extensa que la del caso Mandl, se realizaron las modificaciones respectivas para el correcto funcionamiento del algoritmo. A continuación, se presentan las diferencias del algoritmo para el caso de estudio y una breve explicación de estas:

- **Líneas 20-26:** Se genera un valor aleatorio para marcar como “Verificado” a los siguientes *n* pares de nodos que contengan el nodo inicio y nodo fin de la ruta a construir, ya que en el dato de entrada ParesOrdenados, se encuentra una gran cantidad de pares de nodos repetidos, esto se debe a su alta demanda. El objetivo de estas líneas es que el algoritmo descarte posibles pares de nodos a agregar, tanto en la construcción de una nueva ruta como en la adición de nuevos nodos a rutas que se están creando, ya que puedan se generar rutas muy similares y la solución inicial podría verse limitada por la falta de diversidad a nivel de rutas. [Esta parte del algoritmo cumple la función de diversificación para las soluciones propuestas, ya que permite explorar diferentes posiciones de la lista ParesOrdenados, encontrando otros nodos candidatos para agregar a la ruta, aumentar la posibilidad de construir rutas diferentes y salir de un óptimo local.](#)
- **Líneas 28 y 69:** Debido a que hay una gran cantidad de nodos ruta que conforman los pares de nodos a escoger para crear una nueva ruta, se puede obtener un tiempo de recorrido bastante alto, que puede ser muy cercano o incluso exceder el tiempo máximo de esa ruta. Al utilizar esta restricción, se busca la posibilidad de agregar al menos un nodo adicional a la ruta y no pasar inmediatamente a la siguiente ruta, teniendo una holgura de tiempo que permita extender su recorrido.
- **Líneas 54, 55, 66 y 67:** Con el fin de evitar crear rutas similares y de generar una mayor variedad de rutas, se descartan los nodos que ya han sido agregados a alguna ruta, marcando como “Verificado” tanto al par de nodos (*i,j*) como (*j,i*).

Finalmente, los datos de salida que genera este algoritmo corresponden a un conjunto de rutas que estarán sujetas a una posterior evaluación para determinar la calidad de la solución. Dicho conjunto de rutas recibe el nombre de RutasSolucion.

#### 4.2.3 Fase de evaluación

La fase de evaluación de rutas para el problema BRT utiliza el algoritmo empleado en el problema TND, ya que arroja los mismos resultados, esto gracias a la facultad que tiene de ser de aspecto genérico, sin embargo, es pertinente realizar cambios a ciertos parámetros como lo son: el número de nodos en el sistema, la capacidad de los buses y el número de iteraciones que se desea para evaluar y reajustar las frecuencias de la solución.

#### 4.2.4 Fase de mejora

En esta fase, se busca realizar modificaciones de la solución inicial, con el fin de explorar diferentes escenarios y encontrar nuevas soluciones que den lugar a mejores resultados a nivel de la función objetivo para el problema de optimización. [Esta fase de mejora funciona como una etapa de intensificación, es decir, realizar diferentes cambios sobre la solución inicial, buscando mejorar la función objetivo.](#)

Las modificaciones para implementar son:

- **Eliminación de rutas:** Para la eliminación de las rutas se toma como base el criterio de la frecuencia de estas. Teniendo en cuenta que la frecuencia determina la cantidad de buses por minuto que tiene una ruta, este valor se espera que no sea muy bajo dado que significa un mayor tiempo de espera por parte de la demanda que debe ser cubierta por esta ruta. En principio, se descartan aquellas rutas cuya frecuencia represente un valor mayor a los 60 minutos de diferencia entre la partida de cada bus, y esto

conlleva a una actualización en el análisis de las rutas disminuyendo la cantidad de rutas en la solución y las frecuencias de entrada al código, es decir  $S = (R,F)$

A continuación, se presenta el pseudocódigo correspondiente a esta modificación:

#### Algoritmo 4. Eliminación de rutas

---

**Datos de entrada:** SRutas = Matriz que contiene las rutas pertenecientes a una solución  
FR = Frecuencias relacionadas a cada una de las rutas en una solución

**Inicio**

1. **Para** j en M
2. **Para** i en |SRutas|
3. **Si**  $FR_i < \text{min}$  **entonces**
4.     Eliminar SRutas [i]
5.     Eliminar FR [i]
6.     Eliminar Ocu [i]
7.     **Romper** i
8.     **Terminar** si
9. **Siguiente** i
10. **Siguiente** j

**Final**

---

M: número grande

min: Nivel de frecuencia mínimo

---

- Eliminación de nodos: Para la eliminación de nodos se ha decidido tomar un parámetro de máximo número de nodos permitidos (max) el cual no permite que un nodo este en más de *max* rutas pertenecientes a una solución  $S = (R,F)$ . En caso de que una solución contenga el nodo i en un numero de rutas superior a max, será eliminado aleatoria mente de tantas rutas como sea necesario para que aparezca en un total de max rutas. A continuación, se muestra el pseudocódigo que permite este cambio:

#### Algoritmo 5. Eliminación de nodos

---

**Datos de entrada:** SRutas = Matriz que contiene las rutas pertenecientes a una solución

**Inicio**

1. **Para** i en n
2. **Para** j en |SRutas|
3. **Si** i está en SRutas[j] **entonces**
4.     ContNodo<sub>i</sub> ← Lleva la cuenta del número de rutas en las que se encuentra el nodo i
5.     **Terminar** si
6. **Siguiente** j
7. **Siguiente** i
8. **Para** i en n
9. **Si** ContNodo [i] > max **entonces**
10.     Posibles ← Nodos que sobrepasan max
11.     Quitar ← Cantidad de veces que se repite un nodo por encima de max
12. **Terminar** si
13. **Siguiente** i
14. **Para** i en |Posibles|
15. **Para** j en |SRutas|
16. **Si** Posibles [i] está en SRutas [j] **entonces**
17.     PosibleRu ← Rutas en las cuales se encuentra el nodo i
18.     **Terminar** si
19. **Siguiente** j
20. **Para** j en |Quitar [i]|

21. Eliminar SRutas [j][i]
  22. **Siguiente j**
  23. **Siguiente i**
- Final**

---

n: Numero de nodos totales

ContNodo<sub>i</sub>: Numero de rutas en las que está el nodo i en una S=(R,F)

Posibles: Vector con los nodos que se encuentran en más de max rutas en una S=(R,F)

Quitar: max - ContNodo<sub>i</sub>

PossibleRu: Rutas en las cuales se encuentra el nodo i

---

Cabe aclarar que en la línea 21 del pseudocódigo se determina la ruta de la cual se eliminara el nodo de forma aleatoria.

- **Tipo de bus:** Como parte del proceso de mejora de las soluciones se da la selección del tipo de bus que se utilizara para cada una de las rutas, para esto se inicia probando todas las rutas para un bus con capacidad de biarticulado y se prohíbe las ocupaciones mayores al 100% por lo cual el máximo de ocupación es de 6,5 personas por metro cuadrado, sin embargo se maneja un código adicional que aumenta la frecuencia de las rutas con una ocupación media del 90% con el fin de disminuirla. Fuera de esto, las rutas que tengan una ocupación media menor al 55% se pasan a buses articulados, de esta manera los buses articulados tienen un máximo nivel de ocupación de 85% que representa 5,1 personas por metro cuadrado.

## 5. Resultados

### 5.1 Resultados del TND – Caso de Mandl

En los resultados mostrados a continuación, se presentan aquellos correspondientes al problema TND, para la generación de diferentes soluciones fue necesario iterar los algoritmos en repetidas ocasiones, ya que se manejan datos de entrada aleatorios en el algoritmo de creación de rutas, permitiendo buscar soluciones más diversas a nivel de cantidad de rutas y tiempo máximo por ruta. [Para este caso del TND, los algoritmos desarrollados tenían por objetivo generar soluciones de rutas y frecuencias \(S=\(R,F\)\) que lograrán una disminución considerable respecto al tiempo de funcionamiento total del sistema \(Z1\) y la cantidad de buses requerida para satisfacer la demanda \(Z2\), siendo estos los indicadores con mayor importancia, para demostrar la calidad de las soluciones propuestas.](#)

Adicionalmente, se realizó la comparación de resultados obtenidos registrando aquellos que cumplieran con los siguientes criterios de factibilidad:

- El porcentaje de demanda a satisfacer (D01) tiene que ser cumplido en su totalidad, es decir, la demanda debe ser atendida al 100% entre viajes directos (D0) y viajes con trasbordo (D1).
- La frecuencia máxima entre todas las rutas de una solución (FR) no debe sobrepasar el máximo de 2.
- Al realizar una comparación entre diferentes soluciones, se deben conservar aquellas soluciones no dominadas a nivel del tiempo total en el sistema (Z1) y de la cantidad total de buses (Z2) para el funcionamiento adecuado del sistema.

Inicialmente, para el caso TND, fue necesario realizar una calibración en los datos de entrada, más específicamente en el rango de tiempos que se iba a destinar para cada una de las rutas, ya que esto generaría rutas más extensas y rutas más cortas. La calibración de estos datos fue realizada mediante prueba y error, en la cual se encontró que las mejores soluciones serían obtenidas manejando un rango de tiempo máximo de 25 a 35 minutos por ruta. A continuación, se presentan los resultados obtenidos:

<i>Solución</i>	<i>D0</i>	<i>D1</i>	<i>D01</i>	<i>Z1</i>	<i>tv</i>	<i>te</i>	<i>tt</i>	<i>Z2</i>	<i>Número de rutas</i>
1	97,7	2,3	100	165443	157708	5936	1800	69,26	17
2	95,6	4,4	100	171624	153967	14257	3400	67,8	17
3	93,4	6,6	100	176495	154267	17129	5100	67,62	12
4	95,9	4,1	100	178267	156117	18950	3200	67,58	16
5	96,8	3,2	100	178534	156918	19116	2500	67,55	17
6	94,5	5,5	100	180442	154249	21894	4300	66,73	15
7	94,9	5,1	100	181536	154400	23186	3950	66,67	16
8	94,9	5,1	100	191429	155234	32245	3950	66,57	16
9	92,5	7,5	100	191494	156661	28983	5850	66,5	11
10	93,2	6,8	100	194448	157097	32051	5300	66,31	11

*Tabla 2. Resultados de las soluciones propuestas para el caso Mandl*  
*Fuente: Autoría propia*

Como se puede observar, existen soluciones con la misma cantidad de rutas, a pesar de esto, se tomaron en consideración ya que, al tener aleatoriedad en los datos de entrada para el algoritmo de creación, las rutas obtenidas eran diferentes. Para este caso, se manejó una alta cantidad de rutas, siendo la máxima de 17 y la mínima de 11 rutas, las cuales consiguieron satisfacer la totalidad de la demanda, generando un tiempo promedio de 180.971 minutos de funcionamiento del sistema y un promedio de 67,23 buses.

De igual manera, se realizó una comparación de resultados con diferentes autores que en estudios anteriores han utilizado el caso de Mandl para la implementación de sus algoritmos de creación y mejora de soluciones. Cabe mencionar que, dentro de la literatura consultada, varios de los autores no registran detalladamente los tiempos de viaje, de espera o de trasbordo, motivo por el cual no se muestran en la siguiente tabla de comparación.

Comparación de instancias con otros autores - Caso Mandl									
Solución	D0	D1	Dun	Z1	tv	te	tt	Z2	Número de rutas
Propuesta (2020)	92,5	7,5	100	191.494	156.661	28.983	5.850	66,5	11
	93,4	6,6	100	176.495	154.267	17.129	5.100	67,62	12
	94,5	5,5	100	180.442	154.249	21.894	4.300	66,73	15
	95,9	4,1	100	178.267	156.117	18.950	3.200	67,58	16
	97,7	2,3	100	165.443	157.708	5.936	1.800	69,26	17
Buba and Lee (2018)	93,38	6,62	100	199.880	170.632	24.098	5.150	95	4
	96,92	3,08	100	191.035	162.080	24.705	4.250	92	6
	96,34	3,66	100	188.337	158.650	25.597	4.100	90	7
	97,17	2,83	100	188.519	159.832	25.487	3.200	94	8
	97,02	2,98	100	196.774	167.754	25.670	3.350	88	10
Arbex and da Cunha (2015)	98,27	1,73	100	223.430	-	-	-	79	4
	98,2	1,8	100	215.800	-	-	-	77	6
	98,52	1,48	100	215.022	-	-	-	77	7
	98,65	1,35	100	213.620	-	-	-	74	8
	99,42	0,58	100	210.039	-	-	-	77	10
Cancela et al. (2015)	90	10	100	191.027	-	-	-	63	20
Nikolić and Teodorović (2014)	95	4	99	190.798	-	-	-	85	6
Alt and Weidmann (2011)	92,7	7,1	99,8	188.314	-	-	-	76	4
	90	9,9	99,9	192.482	-	-	-	72	6
Bagloee and Ceder (2011)	86	13	99	207.814	-	-	-	62	12
Mauttone (2005)	93,51	6,49	100	189.280	162.534	21.647	5.100	79,4	-
	93,51	6,49	100	190.050	162.532	22.419	5.100	79,1	-
	93,25	6,75	100	190.242	160.876	24.065	5.300	73,7	-
	89,24	10,76	100	199.676	160.028	31.198	8.450	64,6	-
	89,24	10,76	100	201.221	160.071	32.700	850	64,4	-
	89,24	10,76	100	202.295	160.113	33.732	8.450	64,3	-

Tabla 3. Comparación de resultados obtenidos por diferentes autores para el caso de Mandl  
Fuente: Autoría propia

Como se puede observar, los resultados obtenidos en este trabajo, hay una gran diferencia a nivel de tiempo en el sistema (Z1) y número de buses necesarios para cumplir la demanda (Z2) con respecto a otros autores, ya que se logró una gran disminución de Z1, consiguiendo el más bajo dentro de los resultados comparados y se logró mantener el Z2 dentro del rango de las mejores soluciones. Para mostrar de mejor manera el rendimiento del algoritmo se decidió realizar dos pruebas: el hipervolumen y el frente de Pareto, con el fin de ilustrar la potencia del algoritmo.

En primer lugar, el hipervolumen es una metodología que permite medir la calidad de las soluciones en un problema de optimización multiobjetivo con dos variables de decisión, en este caso se seleccionaron las variables Z1 y Z2. En este sentido, el hipervolumen consiste en ubicar cada variable dentro de un eje del plano cartesiano, como se puede ver en la ilustración 10 se tomó Z1 en el eje X y Z2 en el eje Y, una vez hecho esto se determina un valor máximo para cada una de las variables, en este caso  $Z1 = 220.000$  y  $Z2 = 120$  y se toma esa coordenada como el punto  $W = (120, 220000)$  de referencia y se entiende que el área bajo la recta  $Y = 220000$  para valores positivos es el universo de posibles soluciones del problema.

Teniendo en cuenta lo anterior, el hipervolumen busca de manera gráfica determinar qué porcentaje del universo de soluciones delimitado por W domina cada una de las soluciones propuestas, en este sentido entre mayor sea este porcentaje mayor es la calidad de la solución o grupo de soluciones. A continuación, se muestra el porcentaje del hipervolumen para los conjuntos solución propuestos y los de la literatura.

<i>Solución</i>	<i>%</i>
Propuesta (2020)	10,91%
Buba and Lee (2018)	3,77%
Arbex and da Cunha (2015)	1,69%
Alt and Weidmann (2011)	5,70%
Bagloee and Ceder (2011)	2,68%
Cancela et al. (2015)	6,26%
Nikolić and Teodorović (2014)	3,87%
Antonio Mautone (2005)	6,09%

Tabla 4. Porcentaje de hipervolumen de las soluciones de los autores  
Fuente: Autoría propia

Como se puede notar en los porcentajes la solución propuesta es la que abarca una mayor área en el plano. Lo anterior ocurre a pesar de existir soluciones que sean mejores a las propuestas en algún Z, esto debido a que las soluciones propuestas son mejores para el valor de Z1 en gran medida, mostrando una reducción del 12,1% de la mejor solución propuesta frente a la mejor solución para Z1 en la literatura consultada. Esto se puede apreciar de mejor manera en el siguiente gráfico de dispersión, donde la separación en Z1 es mucho más pronunciada que la separación que se muestra para Z2 frente a las mejores soluciones para esa variable.

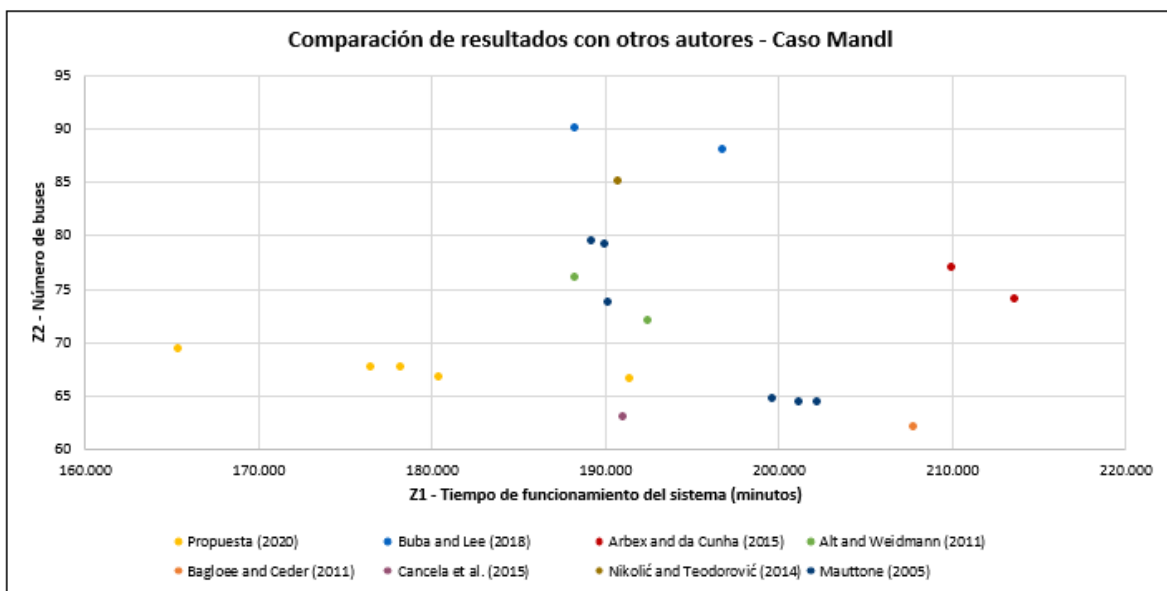


Ilustración 10. Gráfico de dispersión y comparación de los resultados obtenidos por los autores.  
Fuente: Autoría propia

En segundo lugar, el frente de Pareto consiste en ubicar todos aquellos resultados que no sean dominados dentro de un plano cartesiano, siguiendo la metodología del hipervolumen se decide continuar con la construcción del plano según los ejes explicados anteriormente y se procede a eliminar todas aquellas soluciones dominadas que se tenían en consideración por ser soluciones de diferentes autores. A continuación, se muestran únicamente aquellas soluciones que no son dominadas, es decir, que para al menos uno de los Z obtuvieron un resultado menor que otra solución sin importar que sean o no parte del grupo de soluciones propuestas por un mismo autor.

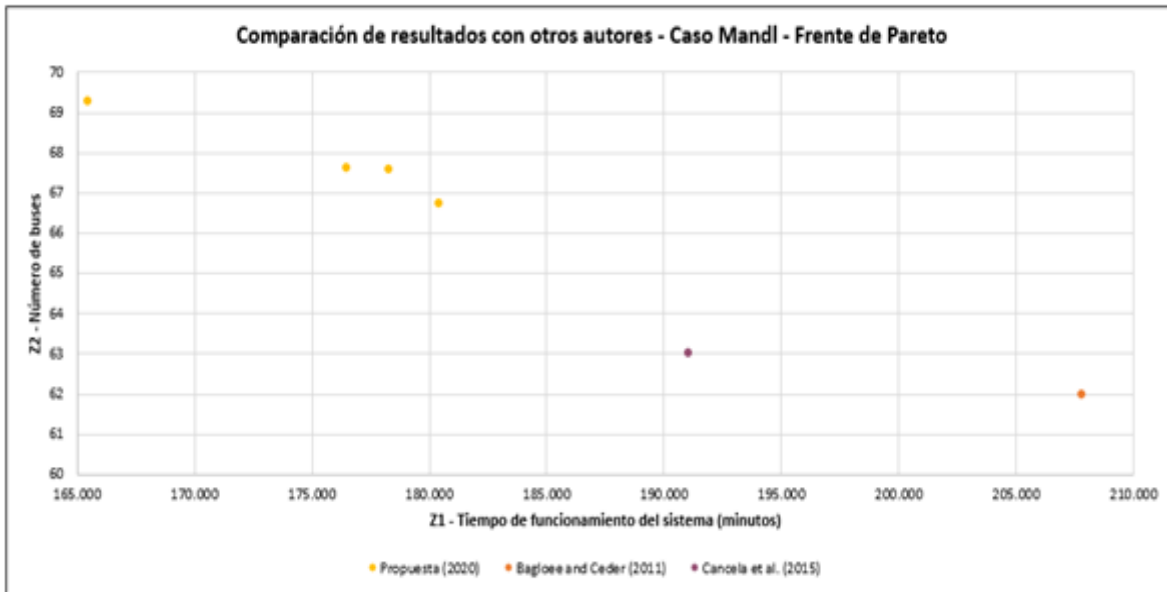


Ilustración 11. Frente de Pareto entre los resultados obtenidos y los de otros autores  
Fuente: Autoría propia

Finalmente, tras realizar reiteradas corridas del código de optimización para el caso del problema TND, con el cual se generaron 350 soluciones iniciales posibles, las cuales serían clasificadas según las condiciones de factibilidad, se obtuvo un tiempo de procesamiento del código de 39,1 segundos en promedio.

## 5.2 Resultados del problema BRT – Caso Transmilenio

Para el problema BRT del caso Transmilenio, se realizaron los ajustes necesarios de los algoritmos implementados para el caso de Mandl en las fases de construcción y de evaluación, debido a que, como se observó en el capítulo 5.1., estos algoritmos consiguieron una mejora significativa respecto a los indicadores de tiempo total de funcionamiento del sistema (Z1) y la cantidad de buses requerida para satisfacer la demanda (Z2), demostrando su capacidad para generar conjuntos de soluciones con buenos resultados. Por otro lado, la fase de mejora busca mejorar, en gran medida, la cantidad de rutas propuestas y la ocupación media y máxima, siendo indicadores que no fueron tenidos en cuenta para el caso de Mandl.

En adición, para este caso, se hicieron iteraciones de los algoritmos de construcción, evaluación y mejora, con el fin de encontrar las mejores soluciones dentro del conjunto de posibilidades, ya que, al igual que en el caso de Mandl, se utilizaron datos de entrada aleatorios, tales como: el tiempo máximo de recorrido de una ruta, el cual oscila entre 30 y 60 minutos, y un contador aleatorio que ayuda a buscar posibles segmentos de ruta a agregar, con la intención de evitar rutas repetitivas o muy similares.

Adicionalmente, se realizó la comparación de resultados obtenidos registrando aquellos que cumplieran con los siguientes criterios de factibilidad:

- El porcentaje de demanda a satisfacer (D01) tiene que ser cumplido en su totalidad, es decir, la demanda debe ser atendida al 100% entre viajes directos (D0) y viajes con trasbordo (D1).
- Al realizar una comparación entre diferentes soluciones, se deben conservar aquellas soluciones no dominadas a nivel del tiempo total en el sistema (Z1) y de la cantidad total de buses (Z2) para el funcionamiento adecuado del sistema.

Inicialmente, para cada iteración, se generó un conjunto de 100 rutas, las cuales serían evaluadas para determinar datos de interés como: la demanda satisfecha con viajes directos (D0), viajes con un trasbordo (D1) y demanda no satisfecha (NO), los tiempos de viaje ( $t_v$ ), de espera ( $t_e$ ) y de trasbordo ( $t_t$ ) y el tiempo final total (Z1) para cada conjunto de rutas, y el número de buses (Z2) a utilizar para dar cumplimiento a esa demanda y configuración de rutas. Posterior a ello, se realizó la evaluación de cada una de las rutas mediante un proceso

iterativo para el reajuste de las frecuencias en cada una de las rutas. Finalmente, se realizó un proceso de selección de rutas gracias las condiciones de factibilidad mencionadas con anterioridad y las soluciones obtenidas son mostradas a continuación:

<i>Solución</i>	<i>D0</i>	<i>D1</i>	<i>Dun</i>	<i>Z1</i>	<i>tv</i>	<i>te</i>	<i>tt</i>	<i>Z2</i>	<i>Número de rutas</i>
1	64,21	35,79	100	8493602	5488456	2540541	464605	889	79
2	62,98	37,02	100	9201351	5307970	3412820	4805600	845,8	78
3	66,14	33,86	100	15255506	5582274	9233722	439510	770,6	100

Tabla 5. Resultados obtenidos para el caso de Transmilenio  
Fuente: Autoría propia

Como se puede observar, se registraron 3 soluciones factibles no dominadas para el caso de estudio, sin embargo, con la finalidad de encontrar un conjunto de mejores soluciones, se realizó una evaluación de estas soluciones con una variación en los siguientes parámetros:

- En la eliminación de rutas de la fase de mejora, inicialmente se estableció que tras cada evaluación del conjunto de rutas se descartarían aquellas cuya frecuencia estuviese por debajo de 0,003 (línea 3 del algoritmo 4). Este parámetro para la eliminación fue modificado y probado con los siguientes valores: 0,01, 0,016 y 0,02 para ver el comportamiento de la solución.
- Como parte de la mejora de los algoritmos de mejora se agregó la siguiente restricción, la cual busca aumentar la frecuencia de los buses con mayor ocupación media y de esta manera disminuir el nivel de ocupación y aumentar el número de buses a utilizar.

Algoritmo 6. Aumento de frecuencias.

---

**Datos de entrada:** Ocu = Ocupación promedio de las rutas en una solución  
FR = Frecuencias relacionadas a cada una de las rutas en una solución

**Inicio**

1. Si  $Ocu_i > 0,8$  entonces:
2.  $FR \leftarrow FR + 0,01$
3. Terminar si

**Final**

---

FR = Frecuencias relacionadas a cada una de las rutas en una solución

---

Gracias a la variación de estos parámetros, fue posible encontrar mejores soluciones a nivel de tiempo dentro del sistema (Z1), el número total de rutas a utilizar y, en algunos casos, el número de buses requeridos (Z2). Esas soluciones son mostradas a continuación:



<b>Mejores soluciones no dominadas</b>							
<b>Solucion</b>	1	2	3	4	5	6	7
<b>Z1</b>	8.337.452	8.341.074	8.355.647	8.548.397	9.082.874	14.675.503	16.721.014
<b>Z2</b>	899	893	892	888	844	761	665
<b>FR promedio</b>	0,14	0,13	0,13	0,14	0,12	0,10	0,07
<b>Minutos promedio entre bus</b>	7,22	7,95	7,62	6,99	8,00	10,16	15,03
<b>P/m<sup>2</sup> Bi</b>	3,91	4,13	4,13	4,38	4,10	4,65	5,64
<b>P/m<sup>2</sup> Ar</b>	3,98	3,63	3,60	3,88	3,33	2,98	1,78
<b># Rutas</b>	65	71	68	62	70	78	100
<b>Te</b>	2.358.015	2.372.886	2.384.532	2.502.742	3.283.551	8.537.891	10.699.811
<b>Te prom</b>	9,08	9,14	9,18	9,64	12,65	32,88	41,21
<b>Tv</b>	5.509.821	5.502.298	5.501.555	5.574.621	5.317.923	5.691.698	5.581.693
<b>Tv prom</b>	21,22	21,19	21,19	21,47	20,48	21,92	21,50
<b>D0</b>	63,82%	64,11%	63,82%	63,71%	62,91%	65,65%	66,14%
<b>D1</b>	36,18%	35,89%	36,18%	36,29%	37,09%	34,35%	33,86%

Tabla 6. Mejores soluciones no dominadas obtenidas para el caso de Transmilenio  
Fuente: Autoría propia

Para mayor claridad, se presenta una explicación de alguno de los ítems expuestos en la tabla 6:

- **FR promedio:** Frecuencia promedio del conjunto solución de rutas.
- **Minutos entre bus promedio:** Minutos promedio entre la salida de los buses para comenzar el recorrido de las rutas.
- **P/m<sup>2</sup> Bi:** Ocupación de personas por metro cuadrado para buses biarticulados.
- **P/m<sup>2</sup> Ar:** Ocupación de personas por metro cuadrado para buses articulados.

Como es posible observar, se encontró una disminución en Z1, Z2 y el número de rutas para la gran mayoría de los casos, con respecto a las soluciones iniciales. Teniendo en cuenta que una de las características destacadas de la metaheurística para el problema BRT es contribuir a mejorar la **percepción de** calidad del servicio para los usuarios a nivel de ocupación en buses y para cada tipo de bus, los resultados anteriores reportan que la cantidad de personas por metro cuadrado es en promedio de 4,42 y 3,31 para buses biarticulados y articulados respectivamente, lo cual indica que no se tendrá una alta ocupación dentro de los vehículos como la que se está manejando **actualmente** en el sistema, que puede estar dentro de 5 y 6 personas por metro cuadrado.

Por otro lado, es posible observar que los tiempos de espera (**te**) tienen un peso bastante importante sobre el cálculo de la función objetivo Z1, sobre todo en aquellos casos en los cuales esta variable sobrepasa los 10 millones **de minutos**. Adicionalmente, se percibe una relación inversamente proporcional entre el número de buses Z2 y el tiempo de espera de los usuarios Te, es decir, con un menor número de buses se obtienen mayores tiempos de espera y viceversa, ya que al empear una flota menor de buses se disminuye la frecuencia de salida promedio de cada uno de ellos. Cabe aclarar que esta espera promedio para el caso de las rutas con transbordo tiene en cuenta la espera que se da por cada uno de los buses del recorrido, tanto el de salida como el de llegada.

Otro factor importante dentro de los resultados obtenidos para cada una de las soluciones corresponde a los porcentajes de demanda satisfechos con un viaje directo y con un viaje con transbordo, ya que en todos los casos se obtiene un porcentaje de demanda directa por encima del 60%. Esto se ve reflejado directamente en Z1 ya que cada operación de transborde conlleva una multa de 5 min para el sistema. Fuera de esto es importante resaltar la similitud entre los tiempos de viaje promedio que a diferencia de los tiempos de espera no varían en gran **medida** a pesar de las diferentes soluciones que se obtienen.

En adición, se presenta una comparación de los resultados obtenidos para el caso de Transmilenio, tanto de las soluciones iniciales como de las soluciones mejoradas, con ayuda de un gráfico de dispersión en el que se

relaciona la variable Z1, tiempo de funcionamiento del sistema, y la variable Z2, número de buses, el cual es mostrado a continuación:

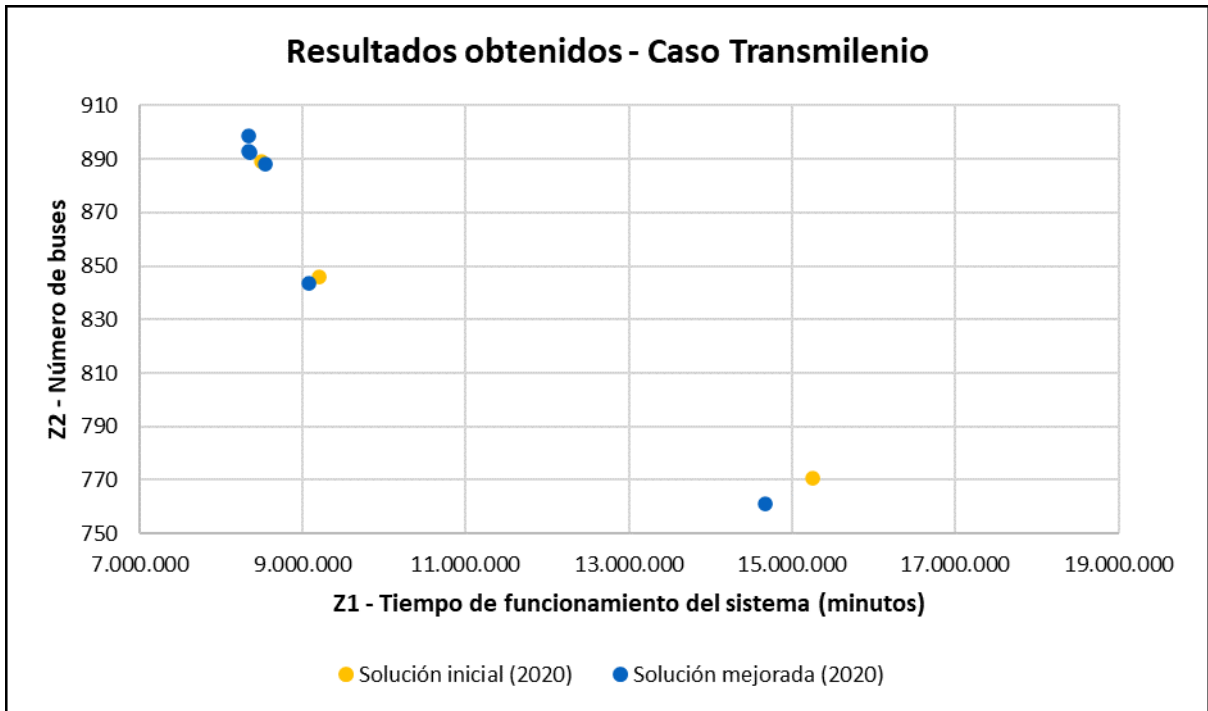


Ilustración 12. Gráfico de dispersión y comparación de resultados de las soluciones iniciales y las mejoradas  
Fuente: Autoría propia

Con ayuda de la anterior grafica se llevaron a cabo los dos análisis de intensidad de las soluciones, en primera instancia se tomó el indicador del hipervolumen, explicado anteriormente, en el cual las soluciones iniciales dominan un 5,35% del área determinada mientras que las soluciones mejoradas dominan el 7,22% de esta área. En segunda instancia, al realizar el frente de Pareto se determinó que tan solo una de las soluciones iniciales hacia parte de este, lo cual significa que se pasó de tener 3 a tener 6 soluciones no dominadas para el caso de estudio, eliminando dos de las soluciones iniciales.

Finalmente, para la generación de soluciones para el problema BRT, se realizaron las corridas del código de optimización, con el cual se genera solamente una solución a la vez, esto debido a que el tiempo de procesamiento para realizar una solución es más demorado y la complejidad del problema es más alta, con una duración promedio de 232,7 segundos por corrida.

### 5.3 Impacto de los resultados obtenidos

En principio los resultados obtenidos en el caso Mandl muestran la intensidad que tiene el algoritmo utilizado para ese caso, consiguiendo las mejores soluciones respecto al tiempo total en el sistema (Z1). Esto se consiguió gracias a la prioridad dada a las rutas directas y a la limitación de tener tan solo 1 transbordo entre cada par de nodos. Este enfoque quito prioridad al número de buses necesarios (Z2) pero demostró en las dos metodologías de análisis que supera en conjunto a los resultados en la literatura. Lo anterior refuerza la idea de que con un pequeño cambio en Z2 se pueden conseguir grandes variaciones para Z1 en este tipo de problemas.

En este sentido cabe resaltar el frente de Pareto mostrado en la ilustración 11 en el cual se tomaron en cuenta las soluciones tanto de la literatura como las propuestas, allí se observa que de las 6 soluciones no dominadas 4 hacen parte de las soluciones propuestas. Teniendo en cuenta lo anterior se observa que las otras dos soluciones mejoran a las propuestas en cuanto al número de buses utilizados (Z2) sin embargo al relacionar este indicador con el del hipervolumen se ve que las 4 soluciones Pareto propuestas dominan un 10,89% del área

mientras que las restantes dos de la literatura dominan el 6,30%, mostrando así que las mejoras en el tiempo total en el sistema (Z1) tienen un impacto porcentual mayor en el área que dominan las soluciones.

Fuera de lo anterior, este trabajo ha permitido ilustrar los resultados por medio de un software de información geográfica, lo cual permite ver con mayor claridad el funcionamiento de un sistema en conjunto, es decir, permite no solo ilustrar cada una de las rutas sino su comportamiento en un tiempo delimitado de estudio, mostrando así posibles errores como el exceso de buses en una parada o un elevado número de buses al mismo tiempo en un arco entre un par de nodos específico. Como apoyo al desarrollo del algoritmo esta herramienta de visualización entrega ventajas que solo desde python no se tendrían.

En cuanto al caso de Transmilenio cabe aclarar que los datos presentados deben de tomarse como parte de un ejercicio teórico para el cual se determinaron ciertas instancias que no son realmente las que tiene Transmilenio en este momento, como lo es la metodología de asignación de la demanda. Aparte se tiene el inconveniente de no poder replicar los resultados obtenidos con el funcionamiento actual de Transmilenio ya que no se cuenta con las frecuencias de las rutas. En este sentido los resultados para el caso Transmilenio deben tomarse de manera teórica y no pueden compararse directamente con el funcionamiento exacto de Transmilenio.

Sin embargo, este trabajo de grado tiene un segundo enfoque el cual es basado en la percepción de calidad de los usuarios frente al sistema de Transmilenio. Como se ha mencionado anteriormente el nivel de satisfacción de los usuarios con el servicio es bajo y esto se debe principalmente a la congestión que presenta el sistema, es decir a la sobreocupación en los buses. Actualmente Transmilenio cuenta con una ocupación media que llega a las 6 personas por metro cuadrado y es en este punto en específico que el proyecto toma mayor relevancia en cuanto al caso Transmilenio ya que como se mencionó anteriormente los promedios de ocupación para articulados y biarticulados son de 3,31 y 4,42 respectivamente, lo cual representa en teoría es un valor bajo, comparable a los estándares europeos de 4 personas por metro cuadrado.

Fuera de la ocupación de los buses se tiene la cantidad de rutas utilizadas en las propuestas presentadas, en su mayoría estas se encuentran por debajo de 80 rutas llegando a alcanzar tan solo 62 rutas en algunos casos, esto también representa una disminución teórica frente a las casi 100 rutas con las que cuenta Transmilenio.

Finalmente es importante recalcar que en el caso Transmilenio no ha sido posible comparar los resultados con el funcionamiento actual del sistema, sin embargo, el aporte teórico correspondiente a la utilización de la ocupación media y máxima de los buses como una variable de decisión y no como un parámetro le da un peso importante al trabajo en un aspecto investigativo y como primer acercamiento al uso de la percepción de calidad como variable dentro de los problemas de optimización multiobjetivo.

## **6. Conclusiones**

Como se mencionó anteriormente, el caso de Mandl ha sido un problema utilizado frecuentemente para la prueba y medición de la capacidad de metaheurísticas en problemas de transporte TND. De igual manera, la metaheurística propuesta en este trabajo de grado para el caso de Mandl demostró un alto desempeño en las fases de construcción y de evaluación de rutas, ya que cumplió con el propósito de minimizar los indicadores del tiempo total de funcionamiento del sistema y la cantidad de buses requeridos para satisfacer la totalidad de la demanda, lo cual se ve sustentado en los resultados obtenidos y contrastados frente a estudios previos de diferentes autores. Gracias a esto, se puede concluir que el aporte fundamental de la metaheurística propuesta para este caso está enfocado en la minimización de los indicadores mencionados anteriormente con las fases de construcción y de evaluación de rutas, además de brindar la facilidad de adaptarla para problemas con un mayor nivel de complejidad como es el caso de estudio de este trabajo de grado, Transmilenio.

Por otro lado, el ajuste elaborado en las fases de construcción y de evaluación de rutas del caso de Mandl para el caso de Transmilenio permitió obtener, en gran medida, buenas soluciones iniciales, simplificando la programación de algoritmos más complejos para estas dos etapas, gracias a su carácter genérico, y dando la oportunidad de enfocarse en la fase de mejora de la metaheurística propuesta para este trabajo de grado. Dicha metaheurística fue complementada y desarrollada con características de un algoritmo tabú en su fase final, es decir que el código busca mejoras a partir de una solución inicial a la cual, posteriormente, le realiza cambios que pueden empeorar los resultados, pero la llevan a otro espacio diferente dentro del universo de posibles

soluciones y, una vez allí, el algoritmo repite el proceso de intensificación o mejora sobre dicha solución de manera iterativa.

Adicionalmente, otro aporte en el que se centra la metaheurística propuesta en este trabajo de grado es una mejora a la percepción de calidad de servicio que se pretende prestar a los usuarios, la cual se basa en:

- Generar una menor cantidad de rutas frente a las que se están manejando actualmente y que sean más fáciles de entender.
- Una disminución de la ocupación media de pasajeros por metro cuadrado en los vehículos, con el fin de evitar posibles sobreocupaciones de pasajeros.
- Una propuesta de asignación de rutas para cada tipo de bus con base en la ocupación media para cada una de estas, ya que la metaheurística tiene en consideración los tipos de buses que están en funcionamiento: articulados y biarticulados.

En adición, el factor de percepción de la calidad por parte de los usuarios no se [ha trabajado](#) en los demás casos [revisados](#) en la literatura, [pues](#) en estos casos simplemente se determina el nivel de ocupación máximo del bus ([generalmente](#) mayor al 100%) y se realizan mejoras únicamente sobre los tiempos de los pasajeros y la cantidad de buses o kilómetros recorridos. En este sentido, el priorizar la ocupación media [en la metaheurística propuesta](#), como uno de los factores a optimizar, [aportó](#) buenos resultados al respecto con un alto impacto sobre los resultados actuales de un sistema como Transmilenio, a pesar del costo que esto tiene sobre los demás factores a optimizar.

Según lo anterior, se logró cumplir con el sentido principal de este trabajo, enfocado a [algunos factores que podrían mejorar](#) la percepción de la calidad del servicio, [tales como: minimización del tiempo de funcionamiento del sistema, que comprende el tiempo de viaje, de espera y de trasbordo; disminución en la cantidad de rutas disponibles para el sistema, y una](#) ocupación media de los buses debajo de las 5 personas por metro cuadrado y, para algunas rutas, debajo de las 4 personas por metro cuadrado.

Sin embargo, es importante realizar la aclaración que los resultados obtenidos por la metaheurística propuesta para el caso de Transmilenio son de carácter teórico, pues tal como [explicó](#) en el capítulo 5.3, el realizar una comparación directa y precisa entre los indicadores y objetivos de interés implicaría tener conocimiento de las frecuencias reales del sistema para determinar sus resultados, las cuales son [información no disponible para el público](#). Por otro lado, para determinar si los resultados teóricos obtenidos se [cumplen en realidad y en qué porcentaje lo hacen](#), se hace necesaria la implementación de las rutas y frecuencias propuestas para Transmilenio y una medición rigurosa a nivel de: el tiempo de espera, de viaje y de trasbordo, la cantidad de buses requeridos para satisfacer la demanda, la ocupación media por tipo de bus.

Finalmente, [se ha podido evidenciar que](#) el transporte público en las grandes ciudades es un problema que va más allá de la movilidad de las personas y de sus viajes diarios, pues la movilidad impacta directamente sobre la calidad de vida de las personas y esta fue una de las razones principales sobre las cuales se basó este [trabajo de grado](#). Durante la investigación de antecedentes y, como base teórica para el desarrollo del proyecto, se tomó en cuenta el problema TND para el cual se encontraron ciertas diferencias con el BRT las cuales se muestran a continuación:

- El caso de estudio de este trabajo cuenta con una gran diferencia respecto al tipo de problema que se está tratando, pues es un problema BRT-TND (Bus Rapid Transit – Transit Network Design) a diferencia del TND, pues los problemas BRT cuentan con una dificultad extra relacionada con las paradas de los buses, ya que este tipo de bus, que va por un carril propio, se detiene en estaciones y estas cuentan con un número limitado de vagones y paradas para los buses y estos pueden estar destinados a parar al mismo tiempo y en una misma parada, generando mayores congestiones.
- Los sistemas BRT tienen una capacidad máxima de pasajeros por hora sentido, y para el caso de estudio, este límite se ve superado en algunas troncales, como en el caso de la troncal Caracas. Por este motivo, se hace necesario el uso de otros medios de transporte públicos adicionales que ayuden a mitigar la demanda del sistema y complementar el sistema de transporte público.

- Los sistemas BRT deben tener en cuenta los principios presentados en este código con el fin de mejorar la percepción de calidad de los usuarios al tener en cuenta la ocupación media de los buses como uno de los factores clave a optimizar.
- Los sistemas BRT, al contar con vías segregadas, disminuyen la posibilidad de conexión entre estaciones, es decir, una estación por lo general está conectada directamente con dos estaciones adyacentes a diferencias de un sistema TND, cuya red de transporte puede tener dos o más estaciones o paradas adyacentes con las cuales conectar, lo cual contribuye a una menor cantidad de rutas diversas.

## 7. Recomendaciones y limitaciones

- Al tener un problema de mayor complejidad gracias a una alta cantidad de nodos y arcos y la manera en la que están conectados, se dificulta encontrar diferentes soluciones que cumplan con criterios de factibilidad tales como la satisfacción completa de la demanda, unas frecuencias y unos porcentajes de ocupación que no sobrepasen un límite definido, ya que se necesita restringir en mayor medida el funcionamiento del algoritmo y los estándares de aceptación de las soluciones propuestas se incrementan.
- La falta de documentación por parte de Transmilenio respecto a las frecuencias oficiales de sus rutas es una limitación importante, ya que impide realizar una comparación precisa y directa entre indicadores de interés tales como: el tiempo total de funcionamiento en el sistema, la cantidad de buses requeridos para satisfacer la demanda y el porcentaje de demanda satisfecha, tanto con viajes directos como con trasbordos.
- Para futuros trabajos de investigación relacionados a los sistemas BRT se propone la estandarización de un caso BRT de ejemplo que cumpla con condiciones de troncales y el bajo número de arcos con el fin de probar la fuerza de los algoritmos y que sea adaptable modelos de diferentes tamaños en BRT.
- La utilización de máximo un transbordo permitido dentro del sistema propuesto limita el universo de soluciones posibles para el problema, sin embargo, pensando en la percepción de calidad en el servicio por parte de los usuarios se dejó de esta manera. Por esta razón se propone la solución del caso sin un límite específico de transbordos con el fin de explorar de qué manera varían los resultados.
- Se recomienda para futuras investigaciones la evaluación estocástica de la demanda en las estaciones con mayor flujo de personas con el fin de mejorar las estrategias para la disminución de la ocupación media de los buses.
- Para la visualización simultánea de una mayor cantidad de rutas se requiere de un equipo de cómputo con mayor capacidad de procesamiento a nivel de datos y de gráficos, ya que los utilizados para este trabajo de grado limitaron estas dos características.

## 9. Referencias

- Agrawal, J., & Mathew, T. V. (2004). Transit route network design using parallel genetic algorithm. *Journal of Computing in Civil Engineering*, 18(3), 248-256.
- Alt, B. ( 1 ), & Weidmann, U. ( 2 ). (n.d.). A stochastic multiple area approach for public transport network design. *Public Transport*, 3(1), 65–87. <https://doi-org.ezproxy.javeriana.edu.co/10.1007/s12469-011-0042-0>
- Arbex, R. O., & da Cunha, C. B. (2015). Efficient transit network design and frequencies setting multi-objective optimization by alternating objective genetic algorithm. *Transportation Research Part B*, 81(Part 2), 355–376. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.trb.2015.06.014>

- Asadi Bagloee, S., & Ceder, A. (Avi). (2011). Transit-network design methodology for actual-size road networks. *Transportation Research Part B*, 45(10), 1787–1804. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.trb.2011.07.005>
- Baaj, M.H., & Mahmassani, H.S. (1991). An AI-based approach for transit route system planning and design.
- Banco Mundial (septiembre 22, 2017). Transporte. Recuperado el 10 de agosto de 2019, de <https://www.bancomundial.org/es/topic/transport/overview#1>
- Bogotá cómo vamos. (2018). Encuesta de percepción ciudadana 2018. Recuperado el 8 de agosto de 2019, de <http://www.bogotacomovamos.org/documentos/encuesta-de-percepcion-ciudadana-2018/>
- Buba, A. T., & Lee, L. S. (2018). A differential evolution for simultaneous transit network design and frequency setting problem. *Expert Systems With Applications*, 106, 277–289. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.eswa.2018.04.011>
- Cámara de comercio de Bogotá. (2016). Encuesta de percepción sobre las condiciones, calidad y servicio a los usuarios de Transmilenio, SITP y TPC. Recuperado el 5 de agosto de 2019, de [https://bibliotecadigital.ccb.org.co/bitstream/handle/11520/18746/Resultados\\_Movilidad\\_2016.pdf?sequence=1&isAllowed=y](https://bibliotecadigital.ccb.org.co/bitstream/handle/11520/18746/Resultados_Movilidad_2016.pdf?sequence=1&isAllowed=y)
- Cámara de comercio de Bogotá. (2016). Encuesta de percepción sobre las condiciones, calidad y servicio a los usuarios de Transmilenio, SITP y TPC. Recuperado el 5 de agosto de 2019, de [https://bibliotecadigital.ccb.org.co/bitstream/handle/11520/18746/Resultados\\_Movilidad\\_2016.pdf?sequence=1&isAllowed=y](https://bibliotecadigital.ccb.org.co/bitstream/handle/11520/18746/Resultados_Movilidad_2016.pdf?sequence=1&isAllowed=y)
- Cancela, H., Mauttone, A., & Urquhart, M. E. (2015). Mathematical programming formulations for transit network design. *Transportation Research: Part B*, 77, 17–37. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.trb.2015.03.006>
- Chakroborty, P. (2003). Genetic algorithms for optimal urban transit network design. *Computer-Aided Civil and Infrastructure Engineering*, 18(3), 184-200.
- El Tiempo. (2002). BOGOTÁ GANA NOBEL EN MOVILIDAD. *El Tiempo*. Recuperado el 8 de agosto de 2019, de <https://www.eltiempo.com/archivo/documento/MAM-1374282>
- Garcia-Suarez, C., Rivera-Perez, A. L., & Rodriguez-Valencia, A. (n.d.). Defining TransMilenio Users' Value and Satisfaction through the Lean Thinking Approach. *TRANSPORTATION RESEARCH RECORD*, 2672(8), 455–463. <https://doi-org.ezproxy.javeriana.edu.co/10.1177/0361198118787363>
- Gilbert, A. & Garcés M. T. (Ed.). (2008). BOGOTÁ: progreso, gobernabilidad y pobreza. Bogotá D.C.: Editorial Universidad del Rosario.
- Hidalgo, D., Lleras, G., & Hernández, E. (2013). Methodology for calculating passenger capacity in bus rapid transit systems: Application to the TransMilenio system in Bogotá, Colombia. *Research in Transportation Economics*, 39(1), 139–142. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.retrec.2012.06.006>
- Hidalgo, D., Pereira, L., Estupiñán, N. & Jimenes, P. L. (2012). TransMilenio BRT system in Bogota, high performance and positive impact – Main results of an ex-post evaluation. *Research in Transportation Economics*, Vol 39, pag. 133 – 138.
- Insitute for Transportation and Development Policy. (n.d.). What is BRT? [Archivo de vídeo]. Recuperado de <https://www.itdp.org/multimedia/what-is-brt/>

- Jha, S. B., Jha, J. K., & Tiwari, M. K. (2019). A multi-objective meta-heuristic approach for transit network design and frequency setting problem in a bus transit system. *Computers & Industrial Engineering*, 130, 166–186. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.cie.2019.02.025>
- Mandl, C. E. (1980). Evaluation and optimization of urban public transportation networks. *European Journal of Operational Research*, 5(6), 396–404.
- Mauttone, A. (2005). Optimización de Recorridos y Frecuencias en Sistemas de Transporte Público Urbano Colectivo. Departamento de Investigación Operativa, Universidad de La República.
- McCormick M. I. (2005). "TransMilenio es el proyecto de movilidad urbana más importante del mundo". [versión electrónica]. *Semana*. Recuperado el 8 de Agosto de 2019, de <https://www.semana.com/online/articulo/transmilenio-proyecto-movilidad-urbana-mas-importante-del-mundo/71251-3>
- Melgarejo, C. (2019). Preste atención a los cambios en las rutas de TransMilenio. *El Tiempo*. Recuperado el 12 de agosto de 2019, de <https://www.eltiempo.com/bogota/cambios-en-rutas-de-transmilenio-en-bogota-363008>
- Nikolić, M., & Teodorović, D. (2014). A simultaneous transit network design and frequency setting: Computing with bees. *Expert Systems With Applications*, 41(16), 7200–7209. <https://doi-org.ezproxy.javeriana.edu.co/10.1016/j.eswa.2014.05.034>
- Nueva ISO 9001:2015. (2017). Norma ISO 9001: Diseño y desarrollo en la industria del software. Recuperado el 28 de septiembre de 2019, de <https://www.nueva-iso-9001-2015.com/2017/02/norma-iso-9001-desarrollo-software/>
- Rincon-Garcia, Nicolás, Navarro-Gomez, David L., Alvarado Valencia, Jorge A., Aguirre-Mayorga, Hugo S., Salazar-Arrieta, Fernando. (2016). BRT and bus users quality expectations regarding metro design. *Revista Técnica de la Facultad de Ingeniería Universidad del Zulia*, 39, 412-422.
- Secretaría de movilidad. (2015). Encuesta de movilidad 2015. Recuperado el 12 de agosto de 2019, de <https://drive.google.com/file/d/0ByNoeWkPXuHpakpSeFVOdnBsQ3c/view>
- Secretaría de movilidad. (2015). Movilidad en cifras 2015. Recuperado el 12 de agosto de 2019, de [https://www.movilidadbogota.gov.co/web/SIMUR/ARCHIVOS/Movilidad\\_Cifras\\_2015\\_V4\\_marzo2017.pdf](https://www.movilidadbogota.gov.co/web/SIMUR/ARCHIVOS/Movilidad_Cifras_2015_V4_marzo2017.pdf)
- Susuki, H., Cervero, R. & Luchi, K. (2014). Transformando las ciudades con el transporte público. Recuperado el 10 de agosto de 2019, de <http://documents.worldbank.org/curated/en/895051468329469373/pdf/Informe-principal.pdf>
- Universidad de los Andes & Cámara de Comercio de Bogotá. (2017). Observatorio de Movilidad – Balance de Movilidad 2007-2016 (10). Recuperado de <https://bibliotecadigital.ccb.org.co/bitstream/handle/11520/19561/Observatorio%20de%20movilidad%20%2830-11-2017%29.pdf?sequence=14&isAllowed=y>
- Wright, L., Hook, W. (Ed.). (2007). Guía de planificación de sistemas BRT. New York, Estados Unidos: Institute for Transportation & Development Policy