Luis Enrique Güitrón A01018616
Juan Pablo Güitrón A01019936
Álvaro Vázquez A01207516
Dr. Jesús Arturo Pérez
Semana i

Oracle mobility optimization challenge

Diseño de solución y algoritmo

Objetivo de su algoritmo

El objetivo general del algoritmo es presentar una propuesta de optimización de rutas que, se originan en cinco puntos de partida específicos y que se dirigen hacia el mismo destino, Oracle MDC. Las métricas proporcionadas para realizar esta optimización son tiempos de traslado y distancia entre nodos.

El objetivo de nuestro algoritmo será la optimización de los costos de las rutas establecidas.

Descripción del funcionamiento de su algoritmo.

Calcularemos *costos* entre nodos a partir de una combinación lineal entre tiempos y distancias. Cabe mencionar que para este cálculo se asignará un mayor peso al tiempo que a la distancia debido a que consideramos que los principales activos de la organización son sus empleados y su satisfacción y mejor rendimiento laboral tendrá mayor importancia para la compañía que la diferencia relacionada al costo de transporte.

La fórmula que se utilizará para calcular el costo es la siguiente:

$$c = t + \frac{d}{100}$$

Donde:

c : Costo entre nodos
t : Tiempo entre nodos (segundos)
d : Distancia entre nodos (metros)

Con la fórmula anterior se consigue que el valor de tiempo sea aproximadamente 9 veces mayor que el de distancia, considerando una velocidad promedio de 40km/h. Por lo que la distribución de pesos sería del 90% al tiempo y de 10% a la distancia.

La mayor parte del tiempo de ejecución del programa será realizada por parte de un algoritmo genético. Antes de comenzar la ejecución del mismo se realizarán varios pasos que ayudarán al algoritmo a converger a una solución eficiente rápidamente.

La estructura de nuestro algoritmo se compone de los siguientes pasos:

1. Obtener información de base de datos: Guardar tiempos y distancias proporcionados en objetos.

2. Calcular matriz de costos: Combinación lineal tiempo y distancia entre nodos a partir de las matrices del paso 1.

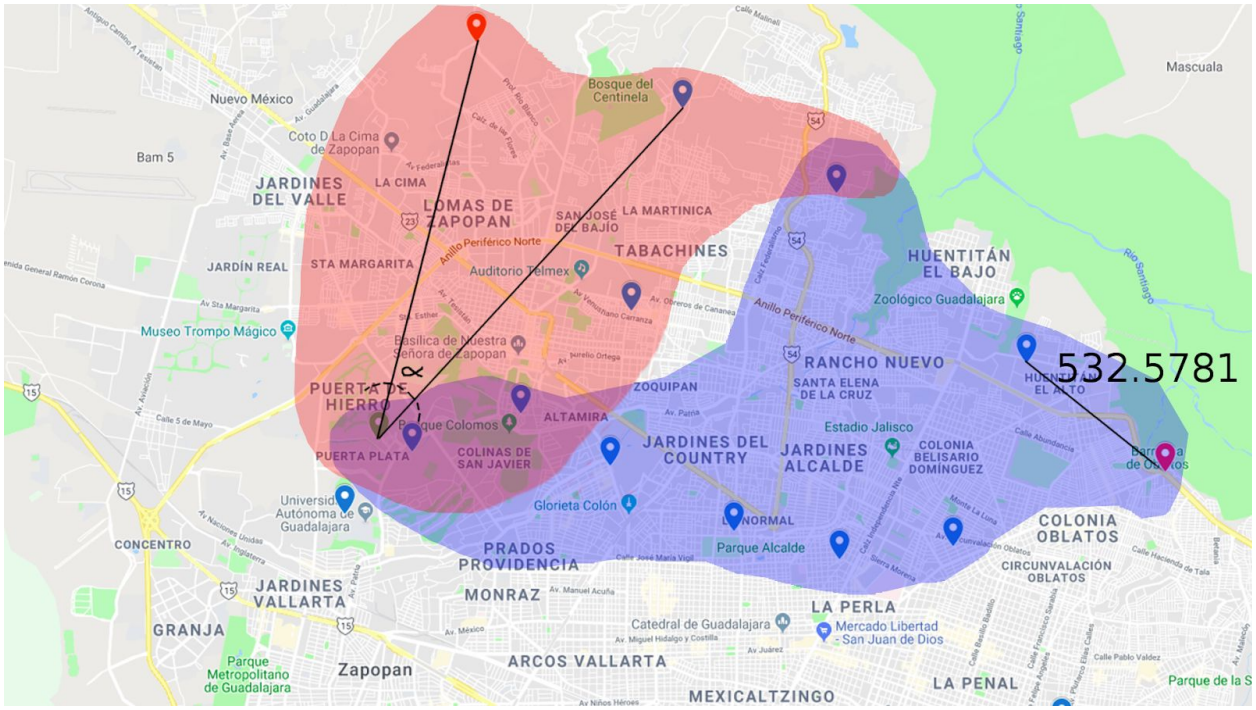
3. Reducción de espacio de búsqueda: Definir cuáles nodos pueden estar en cada ruta. A partir de este paso evitamos que el algoritmo considere la asignación de nodos a rutas que se encuentran demasiado alejadas (Reducción de la complejidad de búsqueda del problema).

Una ruta considerará a un nodo como “asignable” si se cumplen alguna de las siguientes condiciones:

1. El costo entre el nodo de ruta y el nodo de transición es la mínima en relación con los demás nodos de ruta.

2. El ángulo que se forma entre las siguientes rectas es el menor (en comparación con el ángulo de las otras rutas):
    a. Recta 1 - Formada entre punto de destino final (MDC) y el nodo de partida de esta ruta.
    b. Recta 2 - Formada entre el punto de destino final (MDC) y el nodo de transición evaluado.

Un ejemplo de estos dos criterios se muestra en la imagen a continuación:



En el caso de que algún nodo de ruta tenga el menor costo y el menor ángulo con respecto a un nodo de transición, sólamente se establecerá al nodo de transición como asignable a ese nodo de ruta (no habrá overlap).

Identificar los puntos con overlap

A partir del paso anterior se revisarán todos los nodos y se identificarán aquellos que hayan sido considerados “asignables” para más de una ruta.

Representación de solución

Las soluciones propuestas serán representadas como individuos en el algoritmo genético, estos individuos estarán compuestos por:

- 5 cromosomas de permutación: 1 por ruta representando el orden en el que se visita cada uno de los nodos.
- 1 cromosoma de bits: Representa la ruta a la que es asignado cada uno de los nodos que tienen overlap.

Inicialización de individuos

La inicialización de los individuos del algoritmo genético consistirá de:

- Selección de rutas iniciales para cada una de las rutas del problema.
- Asignación de nodos con overlap a a las rutas consideradas asignables.

Con el fin de introducir variedad en los individuos del algoritmo genético, estas inicializaciones se realizarán probabilisticamente de la siguiente manera:

- Asignación de nodos con overlap
    - La probabilidad de que un nodo sea asignado a cierta ruta se calcula con la siguiente fórmula:

      $1 - \frac{ca}{ca+cb}$

      Donde:
      $ca:$ Costo entre nodo actual y nodo inicial de esta ruta
      $cb:$ Costo entre nodo actual y nodo que también fue considerado como asignable para el nodo actual.

    Tras calcular estas probabilidades se generan múltiples individuos para formar la población inicial del algoritmo genético.

- Selección de rutas
    - Cada ruta se construirá paso a paso (visitando un nodo a la vez), para cada paso se asignan probabilidades de visita para los nodos restantes a partir de las siguientes características:
        - Distancia del último nodo del recorrido con nodo evaluado (Nearest Neighbor)
        - Distancia del nodo evaluado con nodo de destino final (Furthest from the goal)

      $$Score_i = \left(\frac{ca}{cg}\right)^p$$

      Donde:
      $ca:$ Costo entre último nodo visitado y el nodo que se considera para visitar
      $cg:$ Costo entre el nodo que se considera para visitar y el nodo del destino final (MDC).
      p: Hiper Parámetro utilizado para dar un diferente nivel de importancia a la diferencia de Score base de los nodos. Un mayor valor hace que los nodos con mayor score de base (ca/cg) reciban mayor probabilidad, mientras un menor valor hace que los nodos tengan probabilidades similares a pesar de la diferencia de score de base.

      La probabilidad para visitar un nodo *i* se calcula a partir del Score del nodo i dividido entre la suma de los Scores de los nodos que faltan por visitar.

Algoritmo Genético

Los valores para los hiper parámetros de nuestro algoritmo serán definidos de forma empírica y son los siguientes:

- Tamaño de generación
- Probabilidad de cruce
- Probabilidad de mutación
- Criterio de convergencia
- Función de selección

Las operaciones genéticas que se utilizan para los cromosomas de permutación son:

- Crossover - Partially matched crossover (PMX)
- Mutation - Swap mutator

Las operaciones que se utilizan para los cromosomas de bits son:

- Crossover - Two point crossover

Antes de comenzar con la ejecución de nuestro algoritmo genético se realiza:

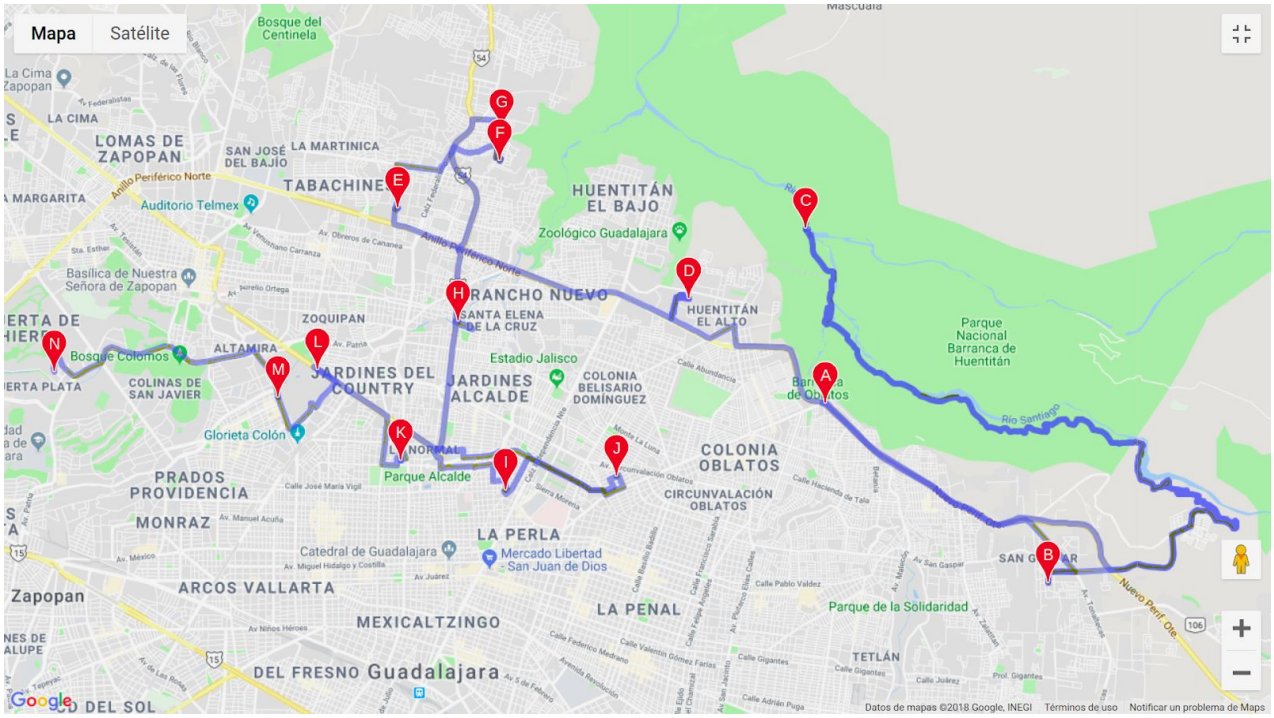
- Inicialización de individuos compuestos por cromosomas de permutación
- Inicialización de individuos compuestos por un cromosoma de bits.

Tras estas inicializaciones se ejecutan alternadamente algoritmos genéticos, en estos se mantiene fija una de las poblaciones mientras se realizan las operaciones genéticas en la otra población.

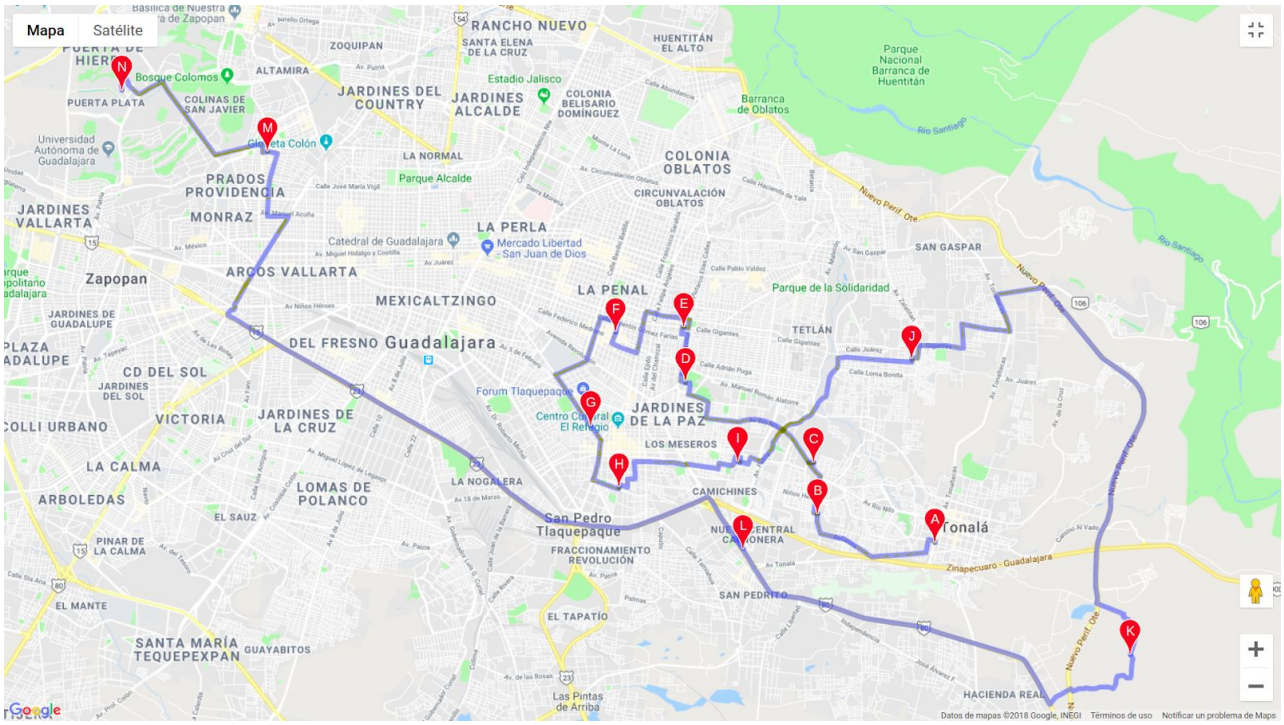
Resultados obtenidos (Nodos 0 - 49)

- *Ruta 101*

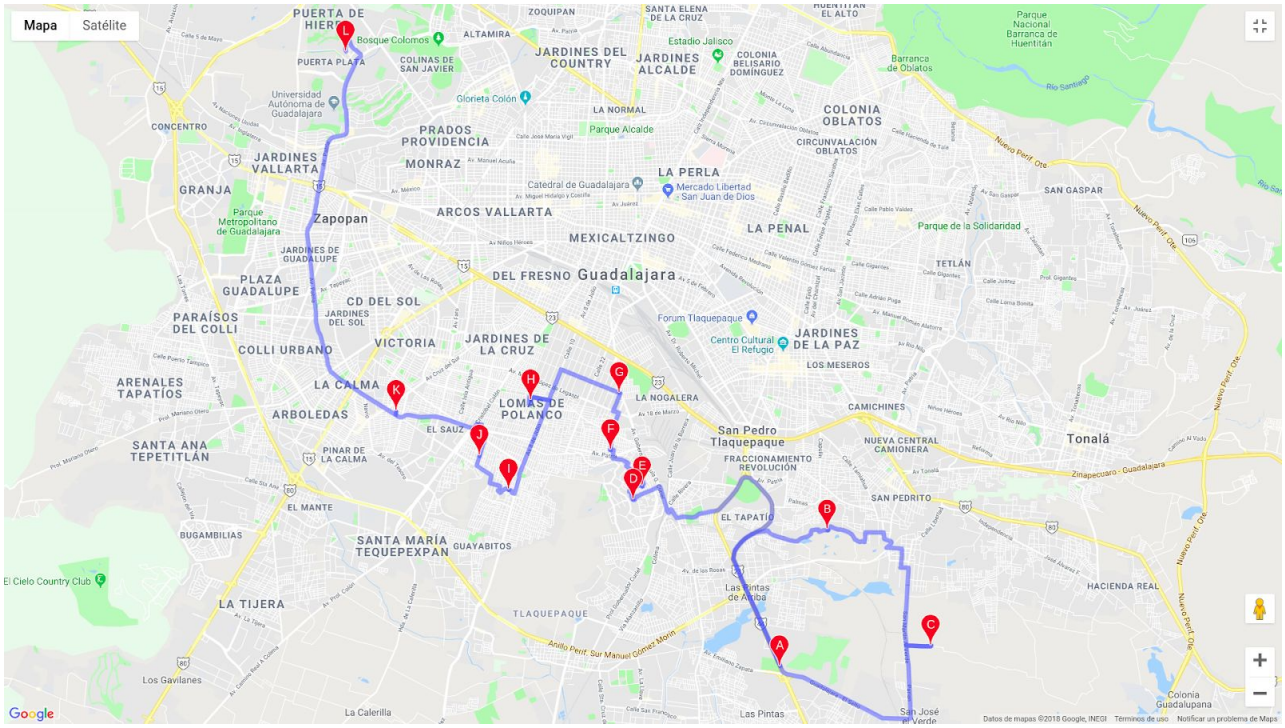
- *Ruta 102*



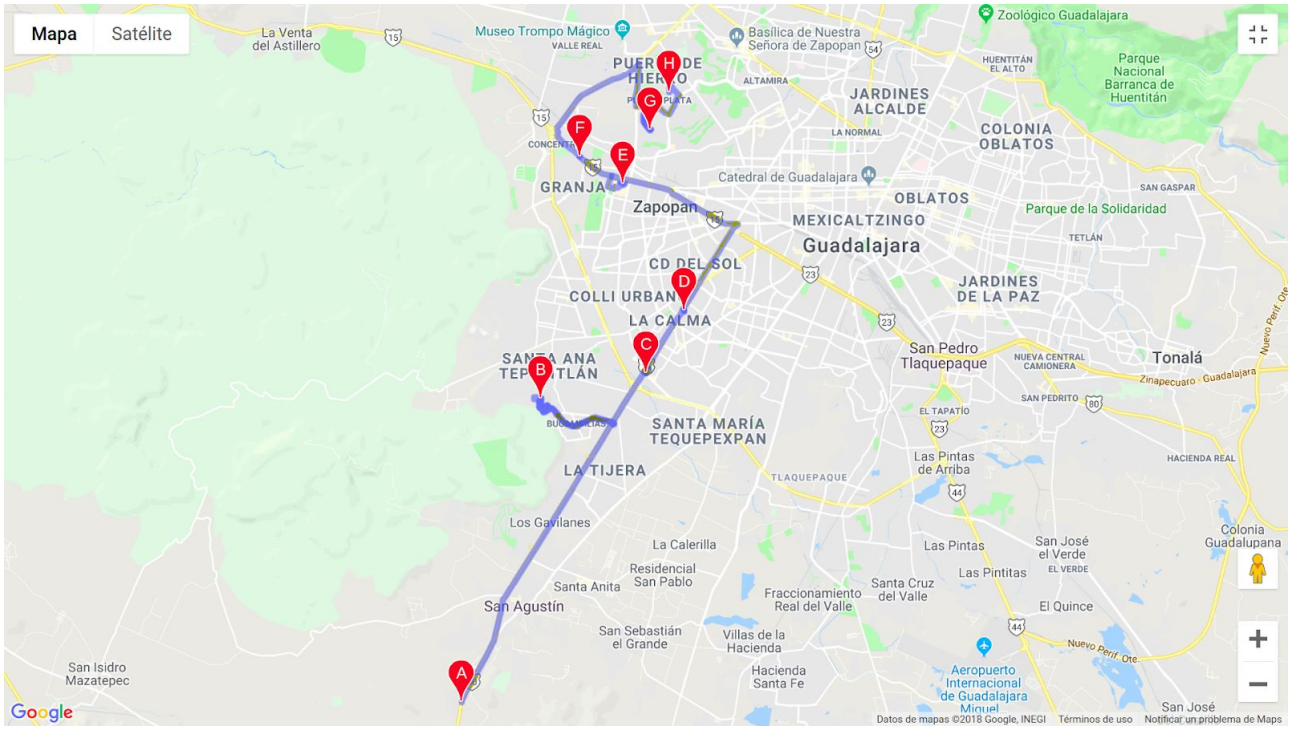
- *Ruta 103*



- *Ruta 104*


Mapa
Satélite
PUERTA DE HIERRO
Bosque Colomos
ALTAMIRA
ZOQUIPAN
SANTA ELENA DE LA CRUZ
JARDINES DEL COUNTRY
JARDINES ALCALDE
Estadio Jalisco
COLONIA BELISARIO DOMÍNGUEZ
Parque Nacional Barranca de Huentitán
Barranca de Oblatos
Río Santiago
PUERTA PLATA
COLINAS DE SAN JAVIER
Universidad Autónoma de Guadalajara
Glorieta Colón
LA NORMAL
Parque Alcalde
COLONIA OBLATOS
CIRCUNVALACIÓN OBLATOS
CONCENTRO
PRADOS PROVIDENCIA
JARDINES VALLARTA
MONRAZ
LA PERLA
Catedral de Guadalajara
Mercado Libertad o San Juan de Dios
SAN GASPAR
GRANJA
Parque Metropolitano de Guadalajara
Zapopan
ARCOS VALLARTA
LA PENAL
Parque de la Solidaridad
JARDINES DE GUADALUPE
MEXICALTZINGO
TETLÁN
PLAZA GUADALUPE
DEL FRESNO
Guadalajara
CD DEL SOL
JARDINES DEL SOL
PARAÍSOS DEL COLLI
Forum Tlaquepaque
Centro Cultural El Refugio
JARDINES DE LA PAZ
COLLI URBANO
VICTORIA
JARDINES DE LA CRUZ
LOS MESEROS
ARENALES TAPATÍOS
LA CALMA
LA NOGALERA
LOMAS DE POLANCO
CAMICHINES
ARBOLEDAS
EL SAUZ
San Pedro Tlaquepaque
NUEVA CENTRAL CAMIONERA
Tonalá
SANTA ANA TEPETITLÁN
PINAR DE LA CALMA
FRACCIONAMIENTO REVOLUCIÓN
Zinapecuaro - Guadalajara
EL MANTE
SAN PEDRITO
EL TAPATÍO
BUGAMBILIAS
SANTA MARÍA TEQUEPEXPAN
GUAYABITOS
El Cielo Country Club
Las Pintas de Arriba
HACIENDA REAL
LA TIJERA
TLAQUEPAQUE
Anillo Perif. Sur Manuel Gómez Morín
Los Gavilanes
La Calerilla
Las Pintas
San José el Verde
Colonia Guadalupana
Google
Datos de mapas ©2018 Google, INEGI
Términos de uso
Notificar un problema de Maps

- *Ruta 105*


Mapa
Satélite
La Venta del Astillero
Museo Trompo Mágico
VALLE REAL
Basílica de Nuestra Señora de Zapopan
Zoológico Guadalajara
PUERTA DE HIERRO
PUERTA PLATA
ALTAMIRA
HUENTITÁN EL ALTO
Parque Nacional Barranca de Huentitán
JARDINES ALCALDE
LA NORMAL
COLONIA OBLATOS
CONCENTRO
GRANJA
Catedral de Guadalajara
OBLATOS
SAN GASPAR
Zapopan
Parque de la Solidaridad
MEXICALTZINGO
TETLÁN
Guadalajara
CD DEL SOL
JARDINES DE LA PAZ
COLLI URBANO
LA CALMA
San Pedro Tlaquepaque
NUEVA CENTRAL CAMIONERA
Tonalá
Zinapecuaro - Guadalajara
SANTA ANA TEPETITLÁN
SAN PEDRITO
EL TAPATÍO
BUGAMBILIAS
SANTA MARÍA TEQUEPEXPAN
Las Pintas de Arriba
HACIENDA REAL
LA TIJERA
TLAQUEPAQUE
Nuevo Perif. Ote
Los Gavilanes
La Calerilla
Las Pintas
San José el Verde
EL VERDE
Colonia Guadalupana
Residencial San Pablo
Santa Anita
Las Pintitas
Santa Cruz del Valle
Fraccionamiento Real del Valle
El Quince
San Agustín
San Sebastián el Grande
Villas de la Hacienda
Hacienda Santa Fe
San Isidro Mazatepec
Aeropuerto Internacional de Guadalajara Miguel
San José
Google
Datos de mapas ©2018 Google, INEGI
Términos de uso
Notificar un problema de Maps

- *Tabla de datos obtenidos*

Nodes 0 - 49			
	Time (hrs)	Distance (km)	Cost
Route 101	4.506	145.303	17675.03
Route 102	5.513	103.796	20884.96
Route 103	2.794	70.337	10761.37
Route 104	1.977	50.359	7621.59
Route 105	1.629	50.037	6364.37
Total	16.419	419.832	63307.32

Nodes 50 - 99			
	Time (hrs)	Distance (km)	Cost
Route 101	5.454	174.665	21382.65
Route 102	4.213	107.368	16240.68
Route 103	2.915	61.233	11107.33
Route 104	2.407	61.649	9281.49
Route 105	0.946	30.823	3715.23
Total	15.936	435.738	61727.38

Visualización de soluciones propuestas con Google Maps API

El programa devuelve información de las rutas en un JSON array que puede ser introducido en https://codepen.io/anon/pen/yRLNbw para poder ver gráficamente la solución propuesta.

Ejemplo de JSON array resultante:

[

[103,20.620651,-103.24745],

[28,20.599216,-103.206696],

[12,20.656195,-103.2521],

[27,20.636314,-103.2726],

[19,20.626451,-103.27164],

[46,20.619728,-103.28678],

[45,20.636541,-103.28808],

[49,20.651886,-103.2989],

[18,20.662537,-103.29923],

[26,20.661577,-103.31336],

[40,20.643332,-103.318535],

[20,20.631517,-103.312515],

[100,20.708206,-103.41582]

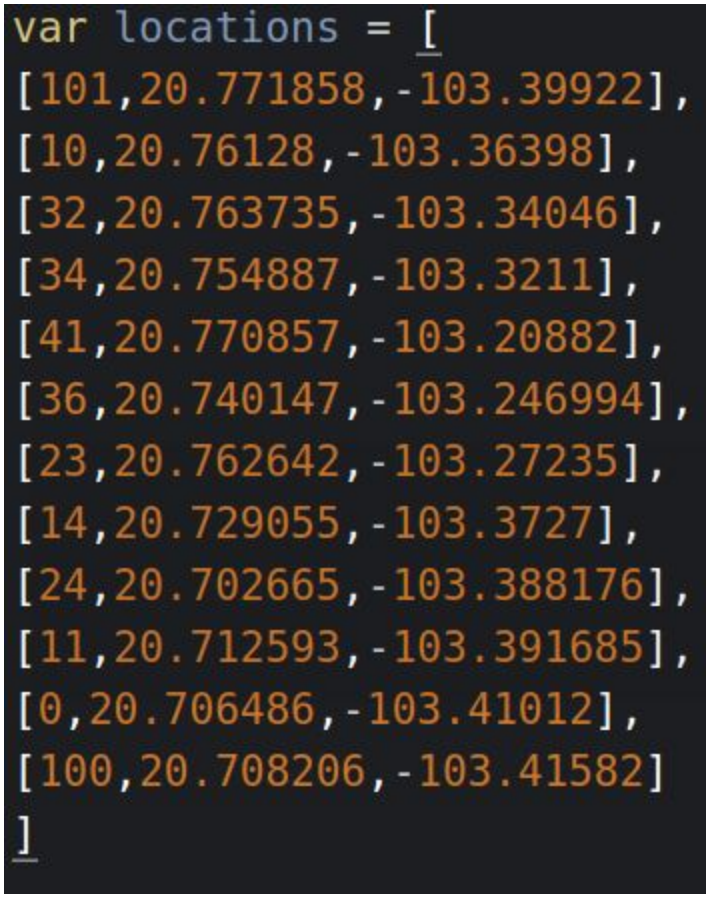
]

Para poder visualizarlo se deben cambiar las variables map y locations del archivo js:

En la variable map se debe cambiar el center por la latitud y longitud del nodo final que en este caso es 20.708206,-103.41582 respectivamente.

```
var map = new google.maps.Map(document.getElementById('map'), {
  zoom: 10,
  center: new google.maps.LatLng(20.708207, -103.4158208),
});
```

La variable locations se debe sustituir por el resultado en formato json del algoritmo.



```
var locations = [
[101,20.771858,-103.39922],
[10,20.76128,-103.36398],
[32,20.763735,-103.34046],
[34,20.754887,-103.3211],
[41,20.770857,-103.20882],
[36,20.740147,-103.246994],
[23,20.762642,-103.27235],
[14,20.729055,-103.3727],
[24,20.702665,-103.388176],
[11,20.712593,-103.391685],
[0,20.706486,-103.41012],
[100,20.708206,-103.41582]
]
```

Alcances del algoritmo propuesto

El algoritmo satisface los requerimientos del problema para las 5 rutas y toma en cuenta las siguientes consideraciones:

- El objetivo de minimización toma en cuenta tiempo y distancia
- Algunos nodos pueden ser asignados a diferentes rutas
- Las rutas establecidas son mejoradas con algoritmo genético

Escalabilidad de algoritmo.

Nuestro algoritmo es escalable para distintos tamaños de problemas, la principal razón de esto es que la mayor parte del tiempo de ejecución está destinada al algoritmo genético, por lo que una buena inicialización rápida puede ayudar a alcanzar una buena convergencia incluso en escenario que tienen varias rutas y varios puntos de parada.
Cabe mencionar que el algoritmo está contemplado de tal forma en que se puedan agregar *n* rutas y *m* paradas en algún momento.