



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA**

**DIVISIÓN DE INGENIERÍA ELÉCTRICA
SISTEMAS OPERATIVOS (2016)**



PROYECTO 2: UNA SITUACIÓN COTIDIANA PARALELIZABLE:

**CONTROL DEL FLUJO DE TRENES EN LAS LÍNEAS 1,5,9,A DEL METRO DE LA
CDMX**

ASIGNATURA: SISTEMAS OPERATIVOS (2016)

GRUPO: 4

PROFESOR: GUNNAR EYAL WOLF ISZAEVICH

ALUMNOS:

**LÓPEZ SOTO MIGUEL ANGEL
DURÁN ROMERO JOSÉ ARTURO**

28 / 03 / 2019

➤ Introducción:

El 20 de octubre de 1975 aproximadamente a las 9:40 de la mañana se suscitó el peor accidente de la historia del STC Metro, pues ocurrió una colisión de dos trenes en la estación Viaducto de la Línea 2 del Metro de la Ciudad de México, y tuvo como consecuencia la muerte de al menos 31 personas y 70 lesionados, siendo el accidente más mortífero de este sistema de transporte.¹ A partir de este hecho fue instalado el sistema de pilotaje automático en la red.

A las 9:36 el tren modelo MP-68 con el número 10 había salido de la estación Chabacano, la cual para entonces sólo servía a la Línea 2, ya no se consideraba hora pico y el tren transportaba entre 130 y 140 personas aproximadamente. En la estación Viaducto, el tren número 8 del mismo modelo se encontraba estacionado en espera de su salida. A las 9:40 el tren número 10 estaba muy cerca de la estación Viaducto, aún sin que partiera el tren 8. El tren 10 que llegaba a la estación embistió al tren 8 estacionado, aproximadamente a 70 km/h.

Los coches motores de los trenes terminaron destruidos, mientras que los coches restantes sufrieron daños moderados (y aunque pudieron ser reconstruidos ya no pudieron funcionar eficientemente) dentro de los coches quedaron atrapadas las presuntas 130 personas que había en cada tren. Las instalaciones de la estación presentaron daños severos que fueron reparados posteriormente. 31 personas murieron y 70 resultaron heridas, algunas de gravedad.



Imágenes de la tragedia ocurrida el 20 de octubre de 1975 | archivo



Rescate de cuerpos y lesionados |Créditos de la imagen: Excélsior TV

El conductor sobreviviente del tren 10, identificado como Carlos Fernández, fue destituido de su cargo y sentenciado a doce años de prisión debido a su negligencia, ya que según testigos, se le había ordenado detener la marcha del tren con anticipación. A su vez, el conductor afirmó no haber podido escuchar dichas peticiones.

A partir de tan lamentable situación se decidió implementar un sistema de pilotaje automático en cada uno de los trenes de todas las líneas de la red que su función es controlar el avance de los trenes, moderar la velocidad de cada tren con base en la ubicación de cada tren es decir hoy en día los conductores no conducen manualmente los trenes (a menos que se active la conducción manual) como ocurría antes de 1975, si no que ellos solo supervisan el estado general del tren, atienden contingencias y controlan la apertura y cierre de puertas.

El pilotaje automático se conecta en red para determinar la ubicación actual de los trenes sus velocidad y el congestionamiento de la línea, lo que permite controlar a cada tren, cuando ocurre un evento inesperado se activa una señal de DBO es decir si un tren se prolonga detenido en alguna estación, es por la presencia de esta señal, que evita que el tren continúe su marcha, es colocada desde el puesto de control y su fin es mantener la frecuencia de paso de los trenes (cuando ocurre algún retraso por incidente, esta señal evita que los trenes se aglomeren en algún punto de la línea)



Operación de DBO en semáforo de avance de tren

➤ Planteamiento

En el presente proyecto se pretende emular el comportamiento de una parte de la red del metro con datos reales de operación, es decir considerando tiempos reales de avance con base en una función de distribución probabilística, pasajeros abordando en cada estación y avance entre interestaciones basadas en distancias reales entre estaciones y la velocidad promedio de un modelo de tren el NM-16.

La simulación contempla a las líneas 1,5 ,9 y A debido a que originalmente queríamos que el proyecto hablara sobre la confluencia en la estación Pantitlán con base en la llegada de trenes, sin embargo dicha idea fue cambiada por el enfoque actual. Se utilizarán hilos que crearán trenes independientes que irán avanzando en la red desde las terminales y su avance es independiente, mediante una lista de mutex emulamos una parte del funcionamiento del sistema de pilotaje automático pues el mutex bloquea el acceso a la posición actual y la anterior, precisamente para evitar que otro tren lo alcance y ocurra una colisión, aparecen semáforos en el código pero no se emplean pues en un momento previo se pensaban utilizar para el control de trenes en Pantitlán acorde al enfoque original del proyecto..

La función de probabilidad es controlada por parámetros de entrada definidos al inicio del programa en el panel de control gráfico que incluye día de operación, hora de operación y velocidad de ejecución mismos que determinan la afluencia de pasajeros en cada estación, como salidas del programa se tiene una salida de texto en consola que muestra el avance y pasajeros que abordan y descenden de cada tren, el tren actual, su posición en cuanto a estaciones, su comportamiento con respecto a mutex y el avance del tiempo en la simulación.

Por otro lado tenemos una salida gráfica que muestra las posiciones de cada tren en un mapa con las estaciones que conforman nuestra red de estudio, cada posición/cuadrado que 'emana' del centro actualiza la posición del tren en cuestión, debido a la cantidad de trenes que hay en cada línea (que relaciona el número de hilos) la simulación pareciera cortarse pero eso ya es cuestión del procesador al tratar de dibujar muchos puntos a la vez.

➤ Desarrollo

Para la realización de este proyecto lo primero en que nos enfocamos es en crear una simulación del sistema que queremos analizar.

En primera instancia analizamos los requerimientos para la simulación por lo que requerimos:

- Control del tiempo transcurrido.
- Número total de trenes disponibles por cada línea.
- Tasa de afluencia según la hora del día y demás factores que intervienen en este parámetro.

Para analizar el tiempo sabemos que el tiempo cómo lo medimos es cíclico (midiendo en 24 horas) y así se repite formando un grupo. Para mantener esta forma tenemos que la unidad mínima de medida es el segundo por lo que:

1 día = 24 horas * 60 minutos/hora * 60 segundos/minuto = 86 400 segundos.

Ahora habíamos dicho que debemos mantener una estructura de grupo por lo que aplicamos la operación de módulo por lo que tenemos:

$$t = x \bmod 86\,400$$

donde:

t = el tiempo transcurrido desde el inicio del día

x = tiempo transcurrido desde el inicio de la ejecución hasta el fin.

También debemos tener en cuenta los días de la semana ya que influye en la afluencia.

Ahora que ya podemos simular días procedemos a obtener los datos de los trenes. para esto obtenemos información real que el STC Metro nos proporciona mediante su página web.

DISTRIBUCIÓN DE TRENES

La red del STC tiene un total de 390 trenes asignados (321 neumáticos y 69 férreos), para proporcionar el servicio a los usuarios en horas punta, se tiene un polígono de operación de 282 trenes, los 108 trenes restantes se encuentran distribuidos en mantenimiento sistemático, mantenimiento mayor, rehabilitación, proyectos especiales y como reserva.

Línea	Trenes
1	49
2	40
3	50
4	12
5	25
6	17
7	33
8	30
9	29
A	39
B	36
12	30
Total	390

Nota: La cantidad de trenes por Línea puede variar por una redistribución del parque vehicular en función de las necesidades del servicio. Con objeto de proporcionar un mejor servicio, el STC modificó trenes de 6 carros por trenes de 9 carros con el fin de incrementar la frecuencia del paso de los convoyes.

Información tomada de <https://www.metro.cdmx.gob.mx/parque-vehicular>

En este problema únicamente trabajaremos con las líneas que inciden en la estación pantitlán (1,5,9,A).

Como podemos ver en la información no todos los trenes están habilitados para su uso diario por lo que solo contamos con 282 para todas las estaciones, esto nos deja sin 108 trenes sin servicio divididos en 12 líneas, por lo tanto para cada línea perdemos 9 trenes en operación.

Recuperando la información y eliminando los trenes faltantes tenemos que:

Línea 1 : 40 trenes.

Línea 5 : 16 trenes.

Línea 9 : 20 trenes.

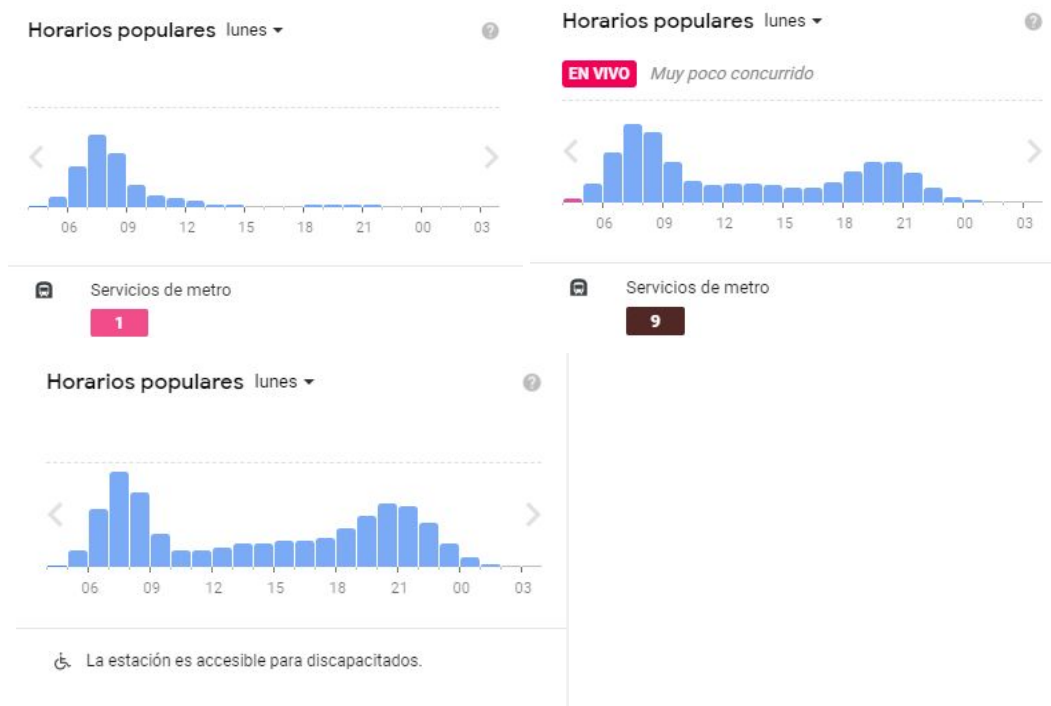
Línea A : 30 trenes.

Es decir ocuparemos alrededor de 160 hilos para cada tren

Estos para efectos prácticos de nuestra simulación se almacenan la mitad en el depósito de pantitlán y la otra mitad en sus respectivas terminales.

Ahora uno de los puntos más complicados es la determinación de la afluencia en partes de la línea. Para esto nos basamos en datos que proporciona Google Maps acerca de las estaciones.

Por lo que por ejemplo en la estación pantitlán tenemos.



Por lo que vemos, la afluencia varía según la línea donde se encuentre y si exploramos las demás líneas vemos que también varía conforme a la estación. Por ejemplo

Estación Chilpancingo Línea 9.

Estación Balderas Línea 1



Estación Acatitla Línea A:

Horarios populares **lunes** ▼



Servicios de metro

A

Estación Balbuena Línea 1.

Horarios populares **lunes** ▼



EN VIVO

Muy poco concurrido



Servicios de metro

1

Asimismo también influye el día de la semana que esté. algunos ejemplos.

Estación Chilpancingo Línea 9

Horarios populares **sábados** ▼



Horarios populares **domingos** ▼



Estación Acatitla Línea A.

Horarios populares **sábados** ▼

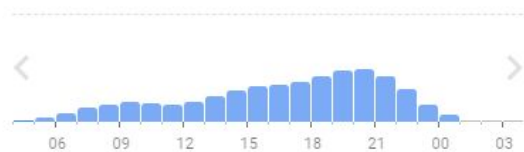


Servicios de metro

A

Estación Balbuena Línea 1.

Horarios populares **domingos** ▼



Servicios de metro

1

Para modelar este comportamiento podemos utilizar una función que varía según la hora del día y el día en sí mismo. Debido a que no tenemos datos que también influyen en la afluencia como el clima, eventos (Conciertos, festivales etc.), y para simplificar un poco las cosas excluimos estos factores del análisis.

Como podemos observar en las gráficas su comportamiento se asemeja mucho a dos “Campanas de Gauss” adelantadas una de la otra y con diferentes parámetros. Ésta función varía según dos parámetros, la posición donde se encuentra el punto más alto (en estadística es la media) y la variación de su “anchura y altura” (Desviación estándar) .

Estos parámetros modifican la curva y por consiguiente la tasa de llegada.

Al evaluar el valor de tiempo en esta función nos da una tasa de llegada que podemos usar para sincronizar los trenes y modificar la distribución de éstos según la hora.

Esta curva como ya mencionamos, se modifica principalmente con estos dos valores por lo que estos valores dependen de los factores ya mencionados pero reduciendo el problema, decimos que solo va a depender del día y el tipo de estación que definimos como concurrida o no concurrida.

Sabemos que en las mañanas las personas van hacia sus trabajos por lo que una función Gaussiana muestra un pico de confluencia a horas pico como las 7 u 8 de la mañana en cambio a las 10 de la noche la afluencia decrece

Modelado de el Tren.

Para esto consideramos un solo tipo de tren con las siguientes características:

Capacidad total : 1530 personas [1].

Dirección (si va hacia la estación pantitlán o a otra terminal)

Velocidad promedio : 50 km/h



Modelo Nacional Mexicano 2016, NM-16

Un hilo va a crear un objeto tren y va a manipularlo conforme al avance de los trenes sobre la línea.

Estimación del tiempo para lanzar un tren

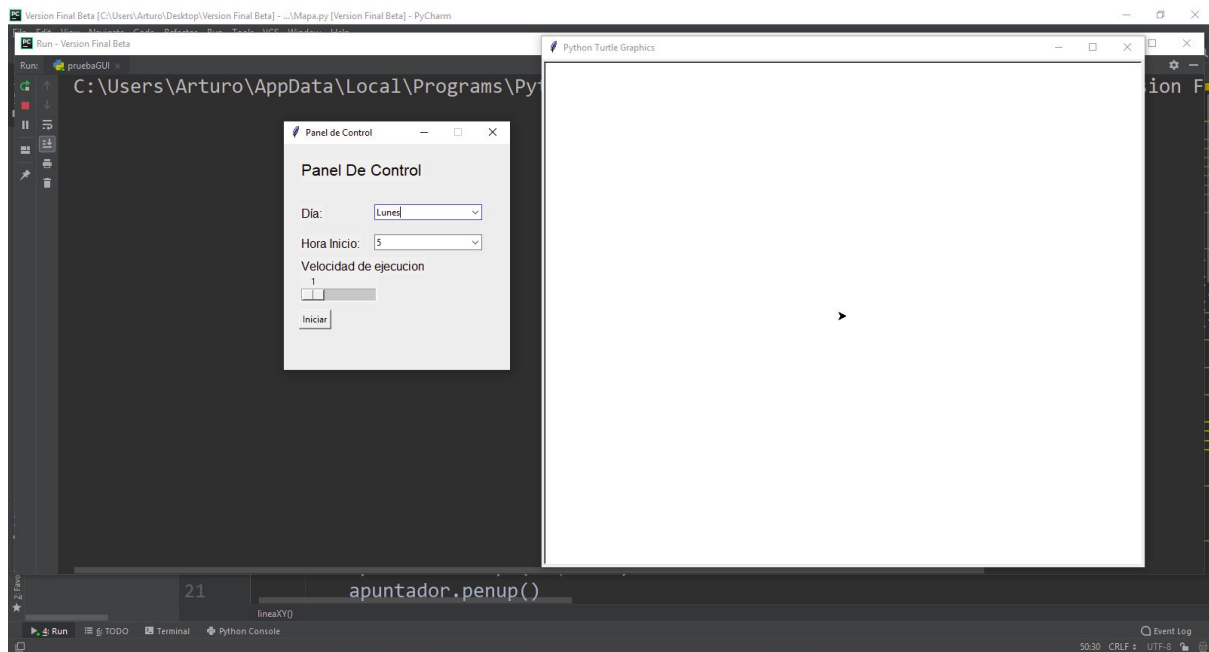
Para realizar esta estimación tenemos que considerar

$$r = (T_{\text{max de espera}} - \text{tasa de llegada}) * (\text{constante de tiempo} / \text{núm máx personas})$$

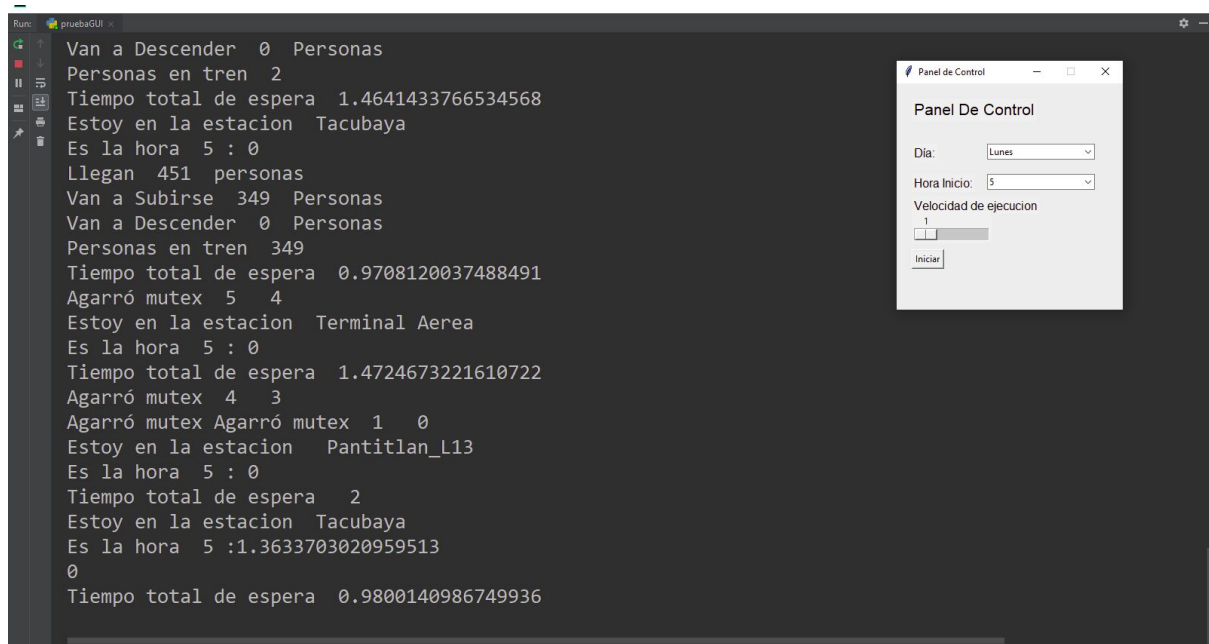
existen parámetros que se determinan aleatoriamente mediante randoms considerando 0.7 de el tiempo máximo por ejemplo.

➤ Ejecuciones y condiciones de ejecución:

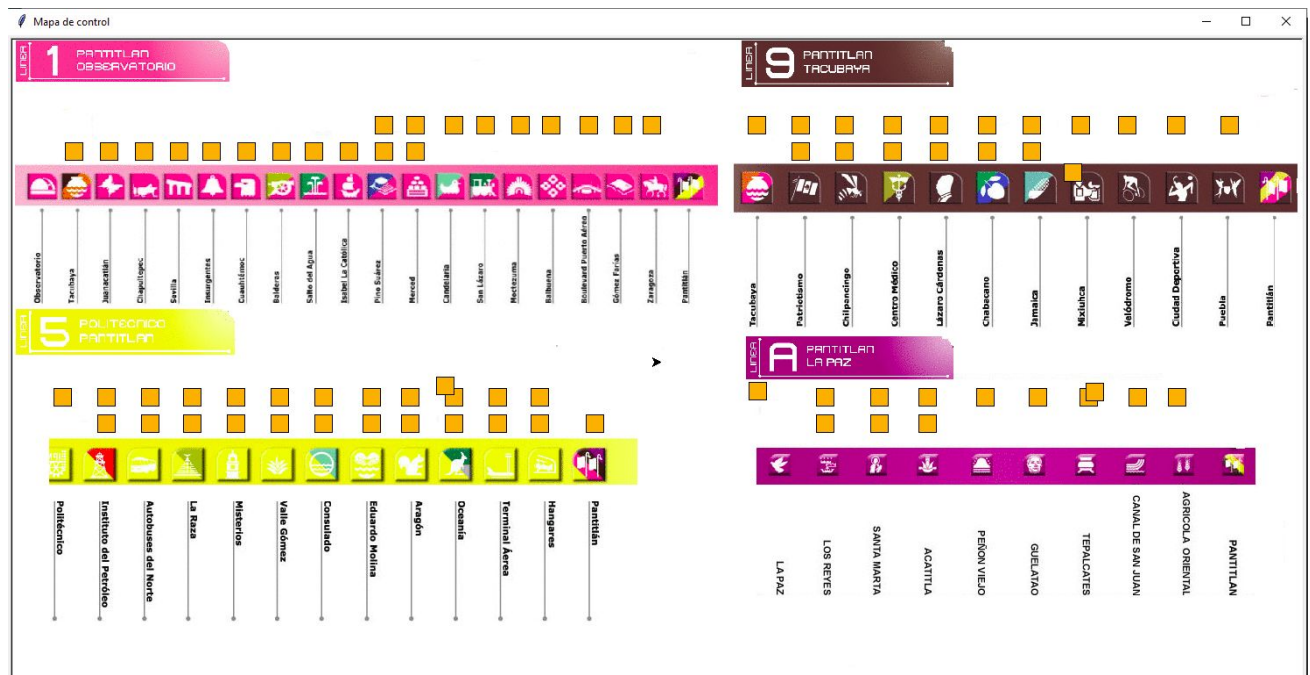
A continuación se muestran capturas de ejecución:



Inicio del programa, Panel de control e inicialización del módulo gráfico



Una vez configurado en la salida de consola tenemos el flujo de cada tren y la confluencia de pasajeros



Ejecución del mapa de control con cada tren.

Aclaración:

```
Run - Version Final Beta
Run: pruebaGUI
Estoy en la estacion Santa_Marta
Es la hora 5 : 0
Tiempo total de espera 1.0969196433483892
Agarró mutex 13 12
Estoy en la estacion Chabacano
Es la hora 5 : 0
Tiempo total de espera 1.244599695733724
Exception in thread LA:
Traceback (most recent call last):
  File "C:\Users\Arturo\AppData\Local\Programs\Python\Python37\lib\threading.py", line 917, in _bootstrap
    self.run()
  File "C:\Users\Arturo\AppData\Local\Programs\Python\Python37\lib\threading.py", line 865, in run
    self._target(*self._args, **self._kwargs)
  File "C:\Users\Arturo\Desktop\Version Final Beta\Sim.py", line 240, in GestorDelinea
    ActualD1 = Depositos[num_linea][0].pop(0)
IndexError: pop from empty list

Funcion Tren 15
Direccion -1
Tren Creado
Funcion Tren 34
Direccion 1
Tren Creado
```

Durante la ejecución puede ocurrir una excepción que se relaciona con un intento de pop de una lista, esto se debe a un problema al momento que un tren se acerca al depósito de

trenes después de la terminal, sin embargo ese error no afecta al desarrollo general de la simulación, pues esta no termina.

➤ **Condiciones de ejecución:**

El programa fue desarrollado utilizando el paradigma orientado a objetos y programación estructurada para la parte lógica de cálculos, la ejecución es mediante la ejecución de GUI_main.py

Versión y lenguaje de programación: Se desarrolló en y probó en **Python 3.7**

Para el desarrollo y pruebas

Existieron dos entornos:

1:

SO: Windows 10

Programación: editor de texto, sublime

Pruebas: ejecución por consola

2:

SO: Windows 10

Programación y pruebas: IDE Pycharm community

Módulos de python requeridos:

Se requieren los siguientes módulos de python:

-matplotlib

-numpy

-scipy

Se utilizan otros pero en Python 3.7 ya están incluidos en las bibliotecas básicas

Nota: La ventana del mapa de líneas tiene una resolución fija, nos disculpamos por ello

Fin del documento.

