

# Traslación del Modelo y Operación

## Entregable 5

### Fase 5. Traslación del Modelo

Se utilizó el lenguaje Python para realizar la traslación del modelo a la simulación, pues es un lenguaje con una sintaxis simple y es multiplataforma.

Se estructuró el código con un enfoque *orientado a objetos*, donde se definen las clases Metro y Tren para controlar su estado.

A pesar de que en el diagrama de flujo propuesto viene la lógica de ejecución, en el programa se especificaron las estructuras para manejar las diversas variables, entre ellas, listas para colas de espera y abordaje de pasajeros en los vagones.

Una corrida de simulación es pausada parcialmente por una función de `sleep()` para apreciar los cambios de estado al sistema.

El parámetro que se puede variar en cada corrida de la simulación es el intervalo en el que se envían trenes de la primera estación: variable que subsecuentemente afecta al sistema con el paso del tiempo.

### Fase 6. Simulación del Modelo (validación)

Comparando los datos obtenidos del número de pasajeros que se quedan en la estación esperando al siguiente tren se nota una fuerte similaridad:

NCE (real)	NCE (simulación)
21	27
16	23
26	27
25	32
27	27
15	29
16	24

Como ejercicio de validación, se intentó correr instancias en las que se enviaban trenes con un retraso muy corto, y esto resultó en datos irreales. Un comportamiento similar se esperaría en

el sistema real, pues debe haber un mínimo tiempo de espera entre un tren y el siguiente. Para tal fin, se le agregó un límite inferior del intervalo de llegadas de 210 segundos.

## Fase 7. Análisis de los datos simulados

Se simularon distintas instancias del sistema, variando el parámetro de intervalo de envío de trenes, y para cada instancia, se hicieron 10 iteraciones.

Envío Trenes:250		Regreso Trenes:340		
TEM	TET	NCLL	NCE	Trenes
155.79	1121702.14	8073	14	27
156.46	1126512.46	7942	17	27
161.99	1166309.12	7991	19	27
156.05	1123560.54	7826	18	27
170.34	1226429.40	8106	25	27
169.81	1222652.70	8272	25	27
167.03	1202625.55	8251	20	27
172.49	1241933.69	8265	26	27
156.78	1128791.75	8033	15	27
164.53	1184597.65	8116	21	28

*Instancia con 250 segundos en llegadas del tren*

Envío Trenes:280		Regreso Trenes:370		
TEM	TET	NCLL	NCE	Trenes
185.67	1336826.52	8111	25	24
188.14	1354607.60	8122	26	24
189.43	1363924.61	8160	29	24
190.77	1373519.82	8137	28	24
186.46	1342499.40	8075	28	24
187.38	1349159.92	8050	27	24
188.27	1355535.31	8258	26	24
186.56	1343209.48	8169	25	24
185.96	1338908.14	8143	26	24
185.78	1337588.46	8170	26	25

*Instancia con 250 segundos en llegadas del tren*

Envío Trenes:330		Regreso Trenes:420		
TEM	TET	NCLL	NCE	Trenes
248.36	1788221.56	8070	63	21
262.67	1891255.59	8052	75	21
247.16	1779546.32	8308	55	21
266.79	1920860.74	8142	78	21
273.41	1968523.68	8162	82	21

246.47	1774589.41	7991	61	21
246.32	1773538.75	7978	62	21
245.10	1764717.33	8052	57	21
256.67	1848007.00	8145	64	21
241.37	1737844.98	8050	55	21

*Instancia con 330 segundos en llegadas del tren*

Observando los datos obtenidos de la simulación, naturalmente es indirectamente proporcional el tiempo de espera y los clientes en espera con el número de trenes que se envían.

Es decir, entre más trenes se están enviando en el mismo lapso de tiempo, menos tiempo de espera habrá.

Evaluando las iteraciones de las instancias anteriores, el caso de 280 segundos es el que actualmente se maneja en el sistema real; sin embargo, reduciendo ese tiempo a 250 segundos, hay un beneficio significativo con el costo de 3 o 4 más envíos extras de trenes.

## Fase 8. Identificación de mejora/optimización del sistema

Tras evaluar distintas instancias de la simulación variando los parámetros, se identificaron algunos puntos clave que pueden mejorar para acercarse aún más a la operación del sistema real. Entre estos puntos, algunos son:

- Muestreo de tiempos de llegadas para cada estación. Por restricciones de recursos, se generalizó la muestra de llegadas para todas las estaciones; sin embargo, lo más preciso sería hacer un muestreo de datos para cada estación estudiada.
- Ampliar el alcance del estudio del sistema para poder evaluar la operación en las subsecuentes estaciones.
- Permitir tener más de 1 en operación. Al estudiarse sólo tres estaciones del metro, no fue relevante el caso de uso donde hay más de 1 tren en operación simultáneamente.
- Incluir el caso en el que los pasajeros bajan del tren. Para el caso estudiado, la cantidad de pasajeros que bajaba era insignificante, por lo que no se incluyó en la simulación. Sin embargo, puede ser relevante para un estudio aún más preciso, y que incluye más de 3 estaciones.
- Movimiento de pasajeros entre colas durante el abordaje. Si llega el caso en que el vagón al que se desea abordar está lleno, sería más aproximado a la realidad si el pasajero busca un vagón vecino que aún tenga cupo para su abordaje.

## Fase 9. Análisis de costo beneficio

Entre los datos de salida se incluye el número de trenes utilizados en el transcurso de la simulación. Un equipo de finanzas puede tomar estos datos y evaluar cuál es la instancia en la que se maximiza el costo beneficio.

Dada la naturaleza del sistema, no fue posible tener a disposición la información de los costos operativos para predecir el costo de la implementación.

## Fase 10. Conclusiones

Tomando como evidencia el conjunto de datos mostrados anteriormente, se puede comparar entre el caso de operación real del metro, y los casos en los que se incrementa o decrementa el tiempo de retraso de llegadas de trenes.

Evidentemente, la solución no es incrementar el tiempo de llegadas, pues causaría un embotellamiento. Por otro lado, el disminuir este retraso en aproximadamente 30 segundos ( $TLT = 250$ ), resulta en menos clientes esperando al siguiente tren, asimismo como un incremento pequeño de trenes enviados (3 o 4 trenes extra). Sumado a la optimización que se haría para distribuir mejor a los pasajeros en otros vagones, se alcanzaría un estado favorable del sistema.