
Optimización del control de semáforos utilizando algoritmos genéticos para la mejora del flujo vehicular

Pablo Uriel Benítez Ramírez

pablo_benitez@comunidad.unam.mx

Posgrado en Ciencias e Ingeniería de la Computación

Universidad Nacional Autónoma de México

1. Introducción

En los últimos años, el desarrollo de la urbanización ha facilitado la movilidad diaria, mejorando el acceso a las redes de transporte y potenciando la actividad económica. Sin embargo, al mismo tiempo, la congestión vehicular en las áreas urbanas se ha convertido en un problema cada vez más complejo, exasperante y urgente. Los modelos tradicionales de gestión del tráfico y los sistemas de control de señales de tráfico se han vuelto cada vez más inadecuados para manejar las complejidades de la movilidad urbana, lo que conduce a consecuencias económicas, ambientales y sociales significativas.

La Ciudad de México ocupa el primer lugar como la zona urbana más congestionada, con conductores que pierden un promedio de 152 horas al año debido al tráfico ([TomTom](#)). Las ineficiencias en los mecanismos de control de tráfico existentes resultan en retrasos prolongados en los viajes, un aumento en el consumo de combustible y un incremento en las emisiones de gases de efecto invernadero.

El control adaptativo de semáforos es esencial para mejorar el flujo vehicular y reducir tiempos de viaje y de espera en entornos urbanos. En este trabajo se implementa un algoritmo genético multi-objetivo que optimiza simultáneamente dos métricas: *tiempo medio de viaje* y *tiempo medio de espera*. La simulación se realiza en SUMO (Simulation of Urban MObility) ([1](#)) y la interacción se lleva a cabo con TraCI.

1.1. Objetivos

- Diseñar un algoritmo genético multiobjetivo para el control de semáforos en una intersección urbana.
- Representar genéticamente las configuraciones semaforicas considerando parámetros como duración de fases y orden de cambio.
- Evaluar la eficiencia de cada configuración mediante simulación discreta del tráfico.
- Obtener un conjunto de soluciones de Pareto que equilibren entre múltiples criterios: tiempo de espera, longitud de fila y flujo vehicular.

2. Desarrollo

En esta sección se explica cada uno de los experimentos realizados para llegar al objetivo esperado.

Kernel: Python 3.11.5

Se genera un archivo `AG_simple`, que contiene los operadores, parámetros y procesos necesarios para el desarrollo de éste proyecto, se encuentra en Github [🔗](#).

2.1. Optimización mono-objetivo

2.1.1. Simulación base base

Se considera una simulación base estándar con una configuración de fases/tiempos de semáforo comunes. Esta simulación será útil para realizar las comparaciones con los resultados posteriormente obtenidos. La configuración de las fases del semáforo tienen la siguiente estructura:

```
prueba = [30, 'GGGGgrrrrrGGGGgrrrrr',  
           5, 'yyyyrrrrrrrrrrrrrrrrrr',  
           30, 'rrrrrGGGGgrrrrrGGGGg',  
           5, 'rrrrrrrrrrrrrrrrrrrrrr']
```

La duración de una fase de un semáforo por lo general dura de 5 a 60 segundos, para la representación de la duración se toman 7 bits, para considerar de 0 a 127. Cada luz del semáforo es representada con dos bits, ya que se tiene

$$m = [G, \quad g, \quad y, \quad r]. \quad (1)$$

Cada fase tiene 20 luces, por lo tanto

$$7 + (20 \times 2) = 47 \rightarrow \text{por 4 fases} \rightarrow 47 \times 4 = 188. \quad (2)$$

Las etapas básicas de la configuración estándar es la siguiente

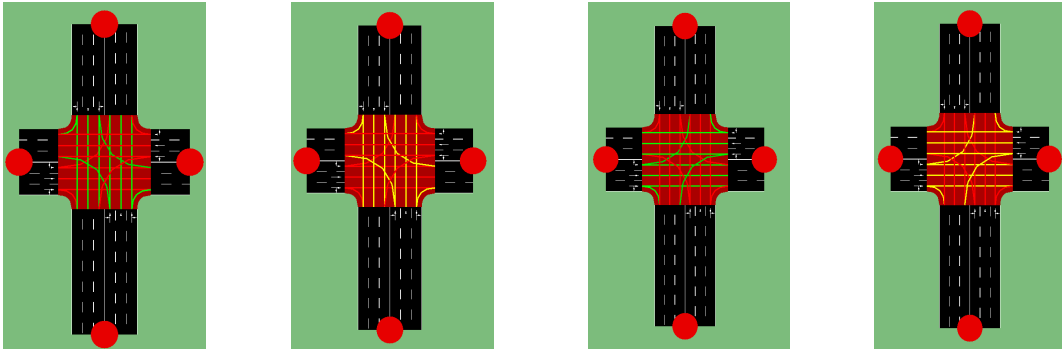


Figura 1: Fases de un semáforo, configuración estándar.

Simulación: video exp1 AG mono base. Tiempo de espera: 322.8 segundos.

Visualización: <https://youtube.com/shorts/2kCsTdTRLgk?feature=share>.

Visualización, velocidad: <https://youtube.com/shorts/25hlklQaz10?feature=share>.

2.1.2. Experimento 1

Para el primer acercamiento se considera un algoritmo genético simple que minimice el tiempo de espera promedio. Las consideraciones son las siguientes

- 1000 generaciones en total, 50 individuos con 20 generaciones de población.
- Cada gen tiene 188 bits.
- Probabilidad de cruce de 0.9 y probabilidad de muta de 1/188.
- Cruza en un punto.
- Se considera elitismo.

2.1.3. Resultados

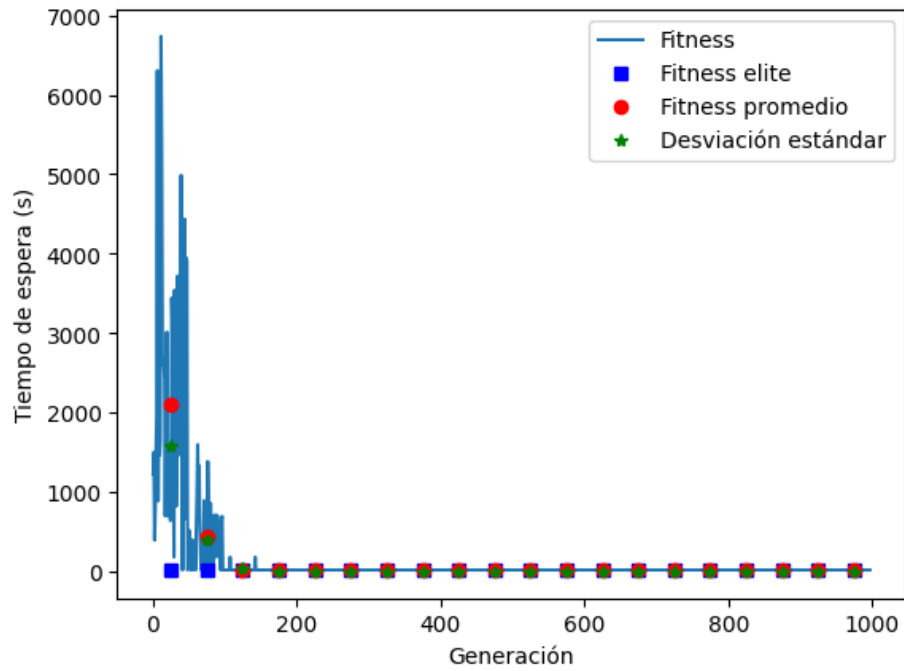


Figura 2: Evolución del fitness a lo largo de la simulación, experimento 1

	dur	state
0	111.00	GryGGrryGGGGyggGGrgG
1	109.00	yGgrGryrGgyrGgygGgyy
2	41.00	GgGggGGryryryrgGGGyy
3	14.00	ggrGGrrGryyygyGyyGGG

Figura 3: Configuración obtenida, experimento 1, monobjetivo

2.1.4. Simulación Exp 1

Simulación: video exp1 AG mono. Tiempo de espera: 18.2 segundos.

Visualización: <https://youtube.com/shorts/b3bTXj00x-E?feature=share>.

Visualización, velocidad: <https://youtube.com/shorts/abBnglNc2lo?feature=share>.

2.1.5. Experimento 2

Se considera un algoritmo genético simple que minimice el tiempo de espera promedio. Las consideraciones son las siguientes

- 1000 generaciones en total, 50 individuos con 20 generaciones de población.
- Estructura de gen se mantiene como en Experimento 1.
- Probabilidad de cruce de 0.9 y probabilidad de muta de 0.1.

- Cruza en un punto.
- Se considera elitismo.

2.1.6. Resultados

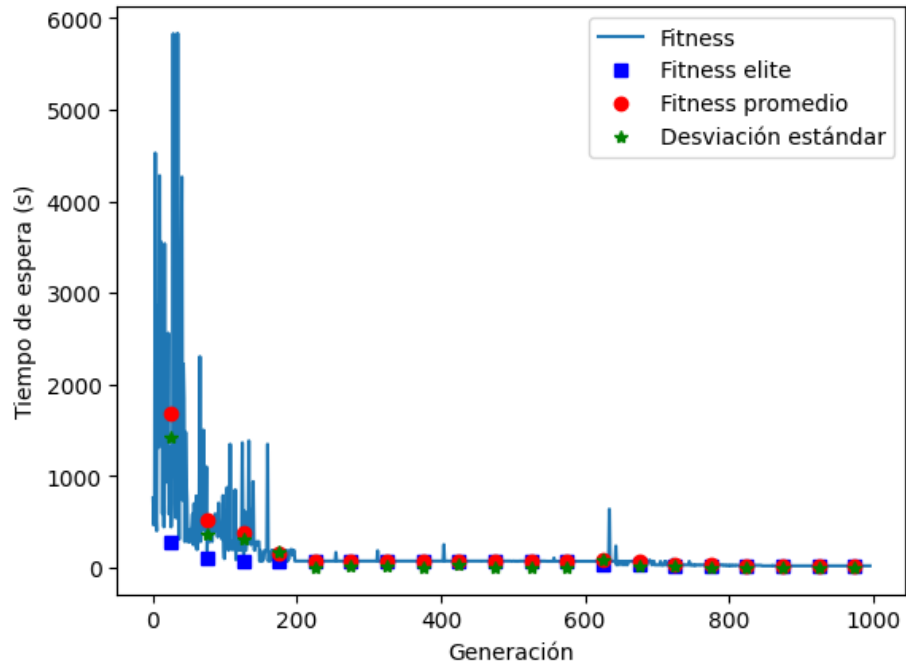



Figura 4: Evolución del fitness a lo largo de la simulación, experimento 2

	dur	state
0	43.00	GyGGgGGGGGgGGGGgGgrr
1	2.00	ygrYGrGyGGrygygrGrGg
2	11.00	ggGgGrrggggygyryGyGy
3	111.00	ggGGyyryyrGGgrGGGyGg

Figura 5: Configuración obtenida, experimento 2, monobjetivo

2.1.7. Simulación Exp 2


Evolución:  *Nombre de archivo:* exp2 AG mono.tex

Simulación: video exp2 AG mono. Tiempo de espera: 18.8 segundos.

Visualización: <https://youtube.com/shorts/FZ62PCZF0Vo?feature=share>.

Visualización, velocidad: <https://youtube.com/shorts/Qy4Gdz0bECo?feature=share>.

2.1.8. Código

Proyecto base.ipynb .

2.2. Optimización multi-objetivo

En la optimización multiobjetivo se consideran dos espacios principales. El *espacio de búsqueda* o espacio de decisión, denotado como $\mathcal{X} \subseteq \mathbb{R}^n$, es el conjunto de todas las variables. Por otro lado, el *espacio objetivo*, compuesto por los valores de la evaluación sobre una función $F(x) \in \mathbb{R}^k$, representa la imagen de un elemento del espacio de decisión.

Definición 2.1 (2) *Un Problema de Optimización Multiobjetivo (MOP) se define como*

$$\begin{aligned} & \min_{x \in \mathcal{X}} F(x), \\ \text{sujeto a } & g_j(x) \leq 0, \quad j = 1, \dots, m, \\ & h_i(x) = 0, \quad i = 1, \dots, p, \end{aligned} \quad (3)$$

donde $x \subseteq \mathbb{R}^n$ es la región factible de

$$\mathcal{X} = \{x \in \mathbb{R}^n : g_i(x) \leq 0 \wedge h_j(x) = 0\}, \quad (4)$$

y $F : \mathcal{X} \rightarrow \mathbb{R}^k$ es la familia de vectores de las funciones objetivo

$$f_i : \mathcal{X} \rightarrow \mathbb{R}, \quad i = 1, \dots, k, \quad (5)$$

$x \in \mathcal{X}$ es conocido como un vector de decisión, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, I$, son restricciones de desigualdad y $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $j = 1, \dots, J$, restricciones de igualdad.

La solución a un problema de optimización multiobjetivo nos produce un conjunto de soluciones de Pareto, que no son comparables entre sí, pero cualquiera de los elementos de éste conjunto satisface el objetivo. Para la integración del enfoque se considera la escalarización de los objetivos propuestos.

Definición 2.2 Sea $F : \mathbb{R}^n \rightarrow \mathbb{R}^k$ el vector de los objetivos, con $F(x) = (f_1(x), \dots, f_k(x))^\top$, y sean w_1, \dots, w_k los pesos asociados a los objetivos $f_i(x)$ tal que $\sum_{i=1}^k w_i = 1$ y $w_i \geq 0$. Por lo tanto

$$F(x) = \sum_{i=1}^k w_i f_i(x) \quad (6)$$

Observación 2.3 Similar al caso mono objetivo los datos obtenidos por la configuración estándar del semáforo son los siguientes

(322.79999999999995, 684.9999999999999)

Con un tiempo de espera promedio de 322.8 y un tiempo de viaje promedio de 685.

2.2.1. Experimento 1

Se considera un algoritmo genético simple que minimice el tiempo de espera promedio y el tiempo total promedio de viaje. Las consideraciones son las siguientes

- 1000 generaciones en total, 50 individuos con 20 generaciones de población.
- Cada gen tiene 188 bits.
- Probabilidad de cruza de 0.9 y probabilidad de muta de 1/188.
- Cruza en un punto.
- Se considera elitismo.

2.2.2. Resultados

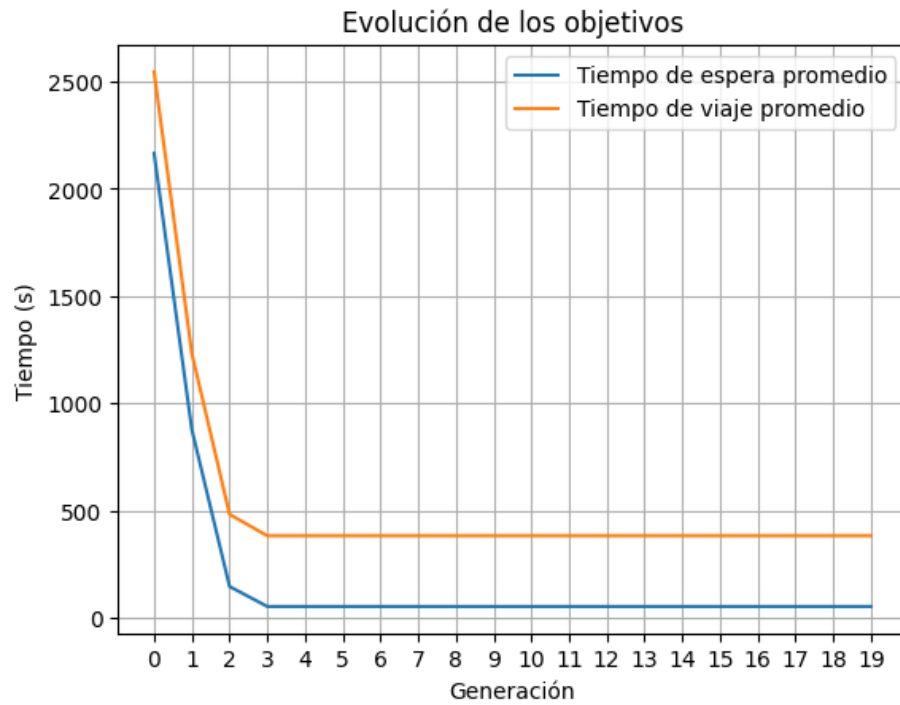


Figura 6: Evolución de los fitness a lo largo de la simulación, experimento 1

	dur	state
0	60.00	ggyGGygyggggGyggrGgy
1	51.00	grrrgGrGryyygyggrgyG
2	14.00	yrrrgGrrryrrgyGGGrgy
3	4.00	ygyrgyryrggggyrGyygGy

Figura 7: Configuración obtenida, experimento 1, multiobjetivo

La configuración obtenida es la siguiente

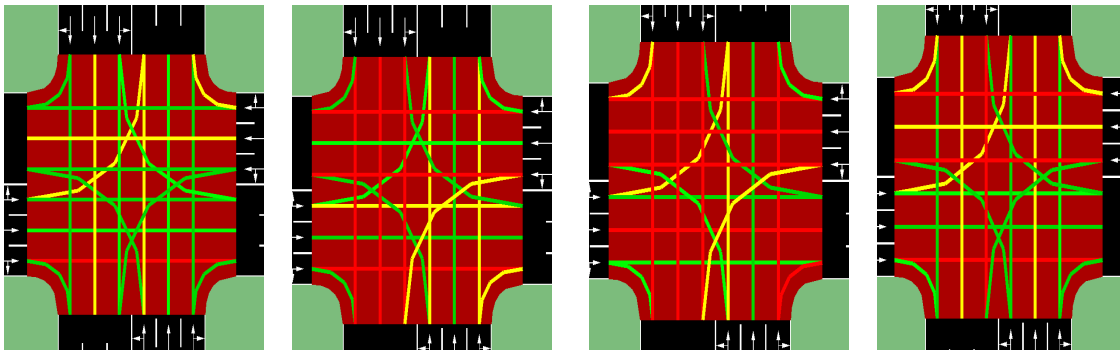


Figura 8: Fases del semáforo obtenidas del experimento 1 multiobjetivo.

2.2.3. Simulación Exp 1

Simulación: video exp1 AG multi.

Tiempo de espera = 52.8

Tiempo de viaje = 382.49999999999994 \approx 382.5

Visualización: https://youtube.com/shorts/IFV_Pi5_3kg?feature=share.

Visualización, velocidad: <https://youtube.com/shorts/s3e4jMBX4Ro?feature=share>.

2.2.4. Experimento 2

Se considera un algoritmo genético simple que minimice el tiempo de espera promedio y el tiempo total promedio de viaje. Las consideraciones son las siguientes

- 1000 generaciones en total, 50 individuos con 20 generaciones de población.
- Cada gen tiene 188 bits.
- Probabilidad de cruce de 0.9 y probabilidad de muta de 0.1.
- Cruza en un punto.
- Se considera elitismo.

2.2.5. Resultados

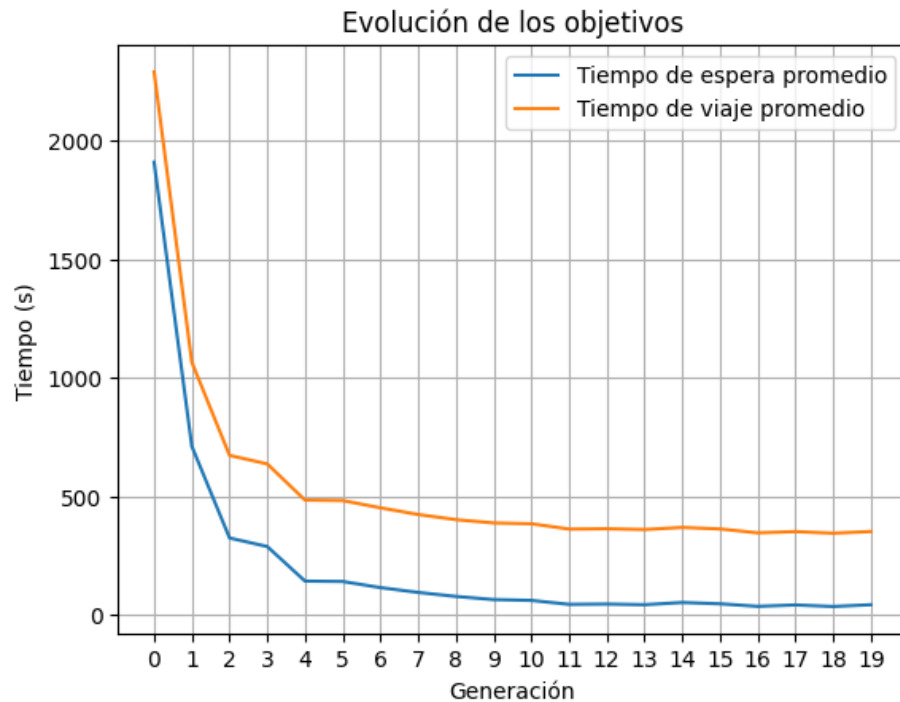


Figura 9: Evolución de los fitness a lo largo de la simulación, experimento 2

	dur	state
0	12.00	ryyrGGgGrrygyGGgGyGy
1	5.00	ggGGGGrGyggGGgrGGGgG
2	32.00	gGGGgGGyryGGgGGyGryG
3	12.00	GrGyyggyGGrryrrGrgGr
Σ	170.00	Links: 20

Figura 10: Configuración obtenida, experimento 2, multiobjetivo

La configuración obtenida es la siguiente

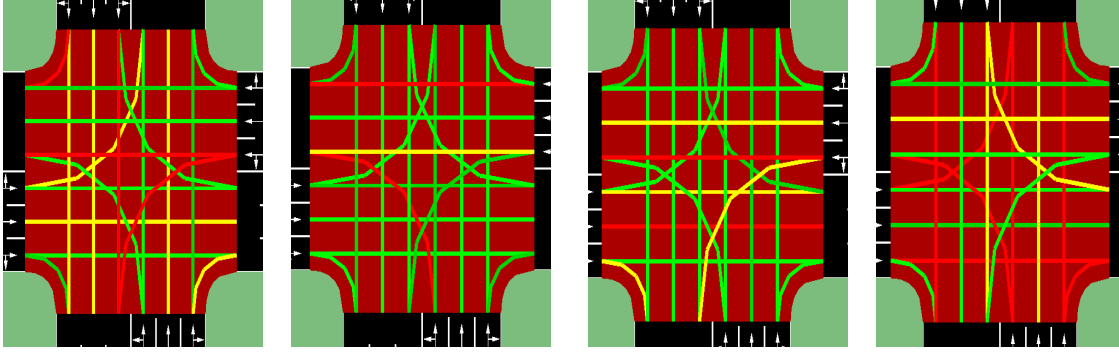


Figura 11: Fases del semáforo obtenidas del experimento 2 multiobjetivo.

2.2.6. Simulación Exp 2

Simulación: video exp2 AG multi.

Tiempo de espera = 35.0

Tiempo de viaje = 342.29999999999999 \approx 342.3

Visualización: <https://youtube.com/shorts/Rbxyj7nxJl0?feature=share>.

Visualización, velocidad: <https://youtube.com/shorts/sccewr202HQ?feature=share>.

2.2.7. Código

Proyecto base.ipynb [🔗](#).

3. Conclusión

En este trabajo se implementó un algoritmo genético para optimizar los tiempos de los semáforos en un entorno simulado con SUMO, utilizando una función de aptitud escalarizada que combina dos objetivos: el tiempo total de espera y el tiempo total de recorrido de los vehículos. Esta aproximación permitió convertir el problema multiobjetivo en uno de objetivo único, facilitando su resolución con técnicas evolutivas tradicionales.

Aunque se lograron mejoras en el flujo vehicular, este enfoque no representa una optimización multiobjetivo pura. La escalarización puede ocultar compromisos entre objetivos en conflicto y limitar la obtención del frente de Pareto completo. En consecuencia, los resultados dependen fuertemente de la elección de los pesos asignados a cada objetivo.

A pesar de ello, este tipo de optimización resulta útil como paso preliminar para la gestión eficiente del tráfico urbano, especialmente en escenarios con múltiples vehículos autónomos. Al reducir los tiempos de espera y de recorrido, se mejora la coordinación entre vehículos y semáforos, contribuyendo a una circulación más fluida y segura.

Referencias

- [1] DLR – Institute of Transportation Systems (2024). Simulation of Urban MObility (SUMO). <https://www.eclipse.org/sumo/>. Acceso: 2025-05-28.
- [2] Miettinen, K. (1998). *Nonlinear Multiobjective Optimization*. Springer, Boston, MA.
- [TomTom] TomTom. Traffic index ranking | tomtom traffic index. Acceso 2025-04-11.