

An integer programming model for exploring the existence of green waves in urban settings

Abstract. In this work, we propose an integer linear programming model for traffic signal synchronization in urban environments, with the aim of maximizing the number of green waves on avenues. This model is applied to portions of the City of Buenos Aires, and the results are evaluated through a traffic simulation. The results show that optimal solutions to the model can be found in reasonable times, and that the provided solutions generate improvements in the simulation environment.

Keywords: traffic light scheduling, green waves, integer linear programming

Un modelo de programación entera para explorar la existencia de ondas verdes en entornos urbanos

Abstract. En este trabajo proponemos un modelo de programación lineal entera para la sincronización de semáforos en entornos urbanos, con el fin de maximizar la existencia de ondas verdes en las avenidas. Se aplica este modelo a porciones de la Ciudad de Buenos Aires y se evalúan los resultados por medio de una simulación de tránsito. Los resultados muestran que se pueden encontrar soluciones óptimas al modelo en tiempos razonables, y que las soluciones provistas generan mejoras en el entorno de simulación.

Keywords: programación de semáforos, onda verde, programación lineal entera

1 Introducción

La programación de semáforos en entornos urbanos es un problema ampliamente estudiado, dado que afecta directamente a la circulación del tránsito en la ciudad. Una técnica muy utilizada en este contexto es la implementación de *ondas verdes*, que consisten en la sincronización de los semáforos a lo largo de una calle para que un vehículo pueda recorrer la calle a una velocidad constante sin ser interrumpido por un semáforo en rojo. En general no es posible programar los semáforos para lograr que todas las calles o avenidas de una ciudad tengan onda verde. No obstante, la búsqueda de ondas verdes tan largas como sea posible es un objetivo deseable de esta programación, dado que pueden contribuir fuertemente a la calidad de vida de la ciudad.

Existen trabajos previos que estudian esta problemática, tanto desde el punto de vista del flujo del tránsito (Bao et al., 2024; Goryaev et al., 2018; Talluri et al., 2025; Wu et al., 2014) como aplicando técnicas heurísticas de optimización (Ma and He, 2019). No obstante, no estamos al tanto de trabajos similares que apliquen técnicas de programación matemática con el objetivo de garantizar la inexistencia de ondas verdes, o bien tratando de minimizar en forma exacta los puntos de corte de las ondas verdes.

Motivados por la observación anterior, en este trabajo estudiamos la existencia de ondas verdes en avenidas conectadas por intersecciones, utilizando un modelo de programación entera que busca maximizar las cuadras sincronizadas con una onda verde, sujeto a condiciones idealizadas de tránsito. Aplicamos este modelo a porciones de la Ciudad de Buenos Aires, con un particular interés en la existencia de ondas verdes en grupos de avenidas con intersecciones entre ellas. Los resultados del modelo son evaluados por medio de una simulación del tránsito a través de estas avenidas, con distintos niveles de congestión y velocidades para los vehículos.

2 El modelo

Sea S el conjunto de semáforos, y sea $A \subseteq S \times S$ el conjunto de cuadras, cada una definida como un par ordenado entre dos semáforos. Suponemos que cada intersección afecta a solamente dos avenidas, y que se tiene un semáforo en S para cada avenida de la intersección. Llamamos $I \subseteq S$ al conjunto de semáforos en intersecciones. Si $i \in I$ es uno de estos semáforos, llamamos $\tau(i) \in I$ al *semáforo opuesto* sobre la avenida que genera la intersección. Si una avenida es de doble mano, entonces el conjunto A deberá contener dos arcos paralelos entre cada par de semáforos consecutivos. Consideramos la función $t : A \rightarrow \mathbb{R}_+$ de *tiempo de viaje* en cada cuadra, para un vehículo que circula a una velocidad predeterminada para la avenida.

Para cada semáforo $i \in S$, llamamos $v_i^{\min} \in \mathbb{R}_+$ y $v_i^{\max} \in \mathbb{R}_+$ a la cantidad mínima y máxima de segundos que el semáforo puede estar en verde, respectivamente. En forma similar, llamamos $r_i^{\min} \in \mathbb{R}_+$ y $r_i^{\max} \in \mathbb{R}_+$ a la cantidad

mínima y máxima de segundos que el semáforo puede estar en rojo, respectivamente. Suponemos que estos valores son datos para el modelo. Además, consideramos un parámetro $\rho \in \mathbb{R}_+$ de tolerancia para la sincronización entre semáforos consecutivos, expresado en segundos.

Para cada $i \in S$, introducimos las variables continuas $v_i, r_i \in \mathbb{R}_+$, que contienen el tiempo en segundos de luz verde y el tiempo en segundos de luz roja para el semáforo i , respectivamente. Suponemos que el semáforo i tiene un *ciclo* formado por v_i segundos en verde seguidos de r_i segundos en rojo, y que este ciclo se repite indefinidamente. Además, introducimos la variable $s_i \in \mathbb{R}_+$, que representa en qué instante de este ciclo comienza el semáforo i cuando se inicia la simulación. Para $(i, j) \in A$, introducimos la variable binaria w_{ij} , de modo tal que $w_{ij} = 0$ si existe una onda verde entre los semáforos i y j , y esto sucede cuando el inicio de cada ciclo del semáforo j es igual al inicio del ciclo asociado del semáforo i más el tiempo de viaje t_{ij} entre los dos semáforos, con una tolerancia de ρ segundos. Además, para cada $i \in S$ introducimos las variables auxiliares $zs_i, zp_i \in \{0, 1\}$ y $ss_i, sp_i \in \mathbb{R}$, de modo tal que $ss_i = s_i$ si $zs_i = 0$ y $ss_i = ss_i - (v_i + r_i)$ en caso contrario, y $sp_i = s_i$ si $zp_i = 0$ y $sp_i = sp_i - (v_i + r_i)$ en caso contrario. Estas variables ayudan a modelar las restricciones que definen el comportamiento cíclico de los semáforos.

Con estas definiciones, el modelo propuesto es el siguiente.

$$\min \sum_{(i,j) \in A} w_{ij} \quad (1)$$

$$v_i^{\min} \leq v_i \leq v_i^{\max} \quad \forall i \in S \quad (2)$$

$$r_i^{\min} \leq r_i \leq r_i^{\max} \quad \forall i \in S \quad (3)$$

$$s_i \leq v_i + r_i \quad \forall i \in S \quad (4)$$

$$v_i = r_{\tau(i)} \quad \forall i \in I \quad (5)$$

$$s_{\tau(i)} = s_i - v_i \quad \forall i \in I \quad (6)$$

$$ss_i \leq s_i + M \quad zs_i \quad \forall i \in S \quad (7)$$

$$ss_i \geq s_i - M \quad zs_i \quad \forall i \in S \quad (8)$$

$$ss_i \leq s_i - (v_i + r_i) + M \quad (1 - zs_i) \quad \forall i \in S \quad (9)$$

$$ss_i \geq s_i - (v_i + r_i) - M \quad (1 - zs_i) \quad \forall i \in S \quad (10)$$

$$sp_i \leq s_i + M \quad zp_i \quad \forall i \in S \quad (11)$$

$$sp_i \geq s_i - M \quad zp_i \quad \forall i \in S \quad (12)$$

$$sp_i \leq s_i - (v_i + r_i) + M \quad (1 - zp_i) \quad \forall i \in S \quad (13)$$

$$sp_i \geq s_i - (v_i + r_i) - M \quad (1 - zp_i) \quad \forall i \in S \quad (14)$$

$$-sp_j \leq -ss_i + t_{ij} + \rho + M \quad w_{ij} \quad \forall (i, j) \in A \quad (15)$$

$$-sp_j \geq -ss_i + t_{ij} - \rho - M \quad w_{ij} \quad \forall (i, j) \in A \quad (16)$$

$$v_j + r_j \leq v_i + r_i + \rho + M \quad w_{ij} \quad \forall (i, j) \in A \quad (17)$$

$$v_j + r_j \geq v_i + r_i - \rho - M \quad w_{ij} \quad \forall (i, j) \in A \quad (18)$$

$$w_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (19)$$

$$z_i \in \{0, 1\} \quad \forall i \in S \quad (20)$$

$$v_i, r_i, s_i \geq 0 \quad \forall i \in S \quad (21)$$

La función objetivo (1) solicita minimizar los pares de semáforos que no están sincronizados. Las restricciones (2)-(3) especifican el tiempo mínimo y máximo en cada color para cada semáforo. Las restricciones (4) definen el rango del estado inicial de cada semáforo dentro del ciclo del semáforo. Las restricciones (5) especifican que semáforos opuestos deben tener tiempos opuestos de verde y rojo, respectivamente, mientras que las restricciones (6) piden que el estado inicial de cada semáforo en una intersección sea simétrico al estado inicial del semáforo opuesto. Las restricciones (7)-(14) definen ss_i y sp_i en función de s_i y del ciclo del semáforo i , dependiendo de los valores de zs_i y zp_i , para todo $i \in S$. Por su parte, las restricciones (15)-(18) definen que si $w_{ij} = 0$ entonces los semáforos i y j deben estar sincronizados en todos los ciclos, para todo $(i, j) \in A$. Las restricciones (15)-(16) solicitan la sincronización para el ciclo inicial y, en conjunto con (17)-(18), estas restricciones imponen la misma condición para todos los ciclos subsiguientes. Finalmente, las restricciones (19)-(21) definen los dominios de las variables.

3 Resultados computacionales

Se implementó el modelo en Java usando Cplex 22.1.1 como solver, realizando la interfaz con el solver a través de la API proporcionada por IBM ILOG Concert Technology. Los resultados computacionales preliminares sobre instancias obtenidas con arterias presentes en porciones de la Ciudad de Buenos Aires muestran que el modelo se puede resolver eficientemente para instancias pequeñas y medianas, proporcionando resultados adecuados. Por ejemplo, el ciclo conformado por las avenidas Independencia, Entre Ríos y Belgrano y por la calle San José (en el barrio de Congreso) admite onda verde para las cuatro arterias con velocidades entre 27 y 56 km/h, para $\rho = 0$. Con 26 y 57 km/h respectivamente, la solución óptima tiene un corte, pero se tiene nuevamente onda verde completa si se permite una tolerancia de $\rho = 0.5$ segundos, algo que en la práctica es imperceptible para los vehículos. Se implementó también un mecanismo sencillo de simulación de tránsito, considerando una velocidad máxima para cada carril y generando los movimientos de los autos en función del siguiente auto en el carril y del siguiente semáforo (en caso de estar en rojo). Esta implementación se realizó también en Java. Las simulaciones realizadas con los resultados del modelo confirman que las ondas verdes propuestas cumplen con criterios razonables de funcionamiento. Si se fuerza a que un semáforo sobre la Av. Belgrano esté fuera de secuencia, entonces el tiempo promedio de recorrida de esta avenida pasa de 228.34 a 257.32 segundos, y los contactos entre autos también aumentan significativamente.

4 Conclusiones

Se ha presentado en este trabajo un modelo para analizar la existencia de ondas verdes en calles o avenidas de una ciudad, con restricciones flexibles en cuanto a la duración del ciclo de cada semáforo y el cumplimiento de la onda verde. Los resultados computacionales muestran que el modelo proporciona soluciones aceptables en función de simulaciones de tránsito. Se prevé continuar el estudio de este modelo, con el objetivo de resolver instancias más grandes que la reportada en la sección anterior, y analizar el impacto de las soluciones sobre la programación de semáforos en situaciones reales.

References

- Bao, R., Huang, W., Lin, Y., Lian, P., Easa, S. M., & Chen, N. (2024). Green wave arterial cooperative control strategy based on through-traffic priority. *Electronics*, *13*, 2939.
- Goryaev, N., Myachkov, K., & Larin, O. (2018). Optimization of “green wave” mode to ensure priority of fixed-route public transport. *Transportation Research Procedia*, *36*, 231–236.
- Ma, C., & He, R. (2019). Green wave traffic control system optimization based on adaptive genetic-artificial fish swarm algorithm. *Neural Computing and Applications*, *31*, 2073–2083.
- Talluri, K. K., Stang, C., & Weidl, G. (2025). Green wave as an integral part for the optimization of traffic efficiency and safety: A survey. *2025 IEEE Intelligent Vehicles Symposium (IV)*, 2422–2429.
- Wu, X., Deng, S., Du, X., & Ma, J. (2014). Green-wave traffic theory optimization and analysis. *World Journal of Engineering and Technology*, *2*, 14–19.