



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

Algoritmos de control de semáforos auto-organizados para incrementar el flujo de tránsito vehicular basados en autómatas celulares

T E S I S

QUE PARA OBTENER EL GRADO DE:

DOCTOR EN CIENCIAS (COMPUTACIÓN)

PRESENTA:

JORGE LUIS ZAPOTECATL LÓPEZ

DIRECTOR DE TESIS: CARLOS GERSHENSON GARCÍA
I I M A S

CIUDAD UNIVERSITARIA, CD. MX.

ENERO, 2019



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Resumen

Los *semáforos auto-organizantes* regulan el tránsito vehicular de manera adaptativa. En este trabajo proponemos un “*sistema de detección deliberativo*” y un “*sistema de control basado en impulso vehicular*” para diseñar semáforos auto-organizantes. El sistema de detección deliberativo infiere las posiciones y velocidades de los vehículos que se aproximan o alejan de la intersección mediante un *entorno virtual*. El entorno virtual representa el estado del tránsito vehicular y depende de la historia percibida por un *sensor* colocado al inicio de la cuadra. El sistema de control basado en impulso vehicular decide los cambios de luz del semáforo con base en la cantidad de vehículos sobre una intersección y el tiempo acumulado que dichos vehículos esperan detrás de una luz roja. El sistema de control ejecuta tres reglas: *otorgar la luz verde a la calle con mayor impulso vehicular, evitar obstruir la intersección y acotar tiempos de espera*.

A partir de los sistemas de *detección deliberativo* y de control *basado en impulso vehicular* proponemos dos tipos de semáforos auto-organizantes: SARIV (semáforo auto-organizante reactivo basado en impulso vehicular) y SADIV (semáforo auto-organizante deliberativo basado en impulso vehicular). Nuestras simulaciones por computadora muestran que el SARIV tiene el mejor desempeño en términos de flujo vehicular en comparación con otros sistemas de semáforos propuestos en la literatura. El SADIV tiene un desempeño peor que el SARIV en términos de flujo vehicular; sin embargo, es más robusto frente a fallas de los sensores y el costo de su implementación podría ser menor debido a que solo requiere de un sensor simple. Además, en este trabajo exploramos cómo los vehículos autónomos reducen los *congestionamientos fantasma* al evitar el sobrefrenado que ocurre cuando los vehículos que son operados por humanos.

Agradecimientos

Agradezco las facilidades brindadas por el Posgrado en Ciencia e Ingeniería de la Computación y del Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas.

En especial agradezco a mi tutor el Dr. Carlos Gersehson García por sus enseñanzas y su apoyo a lo largo de estos años. Sus observaciones y consejos me ayudaron a crecer tanto en lo profesional como en lo personal. A mi comité académico y sinodales, Dr. David Arturo Rosenblueth Laguette, Dra. María Elena Lárraga Ramírez, Dr. Christopher Rhodes Stephens y Dr. Luis Agustín Álvarez Icaza Longoria, sus observaciones siempre enriquecieron mi trabajo.

Agradezco a mis amigos Gustavo Carreón Vásquez, Luis Enrique Cortes Berrueco, Jesús Mario Siqueiros García y Hyobin Kim, aprendí mucho de ustedes y fue un gusto conocerlos.

Al personal del Posgrado en Ciencia e Ingeniería de la Computación, al Coordinar Dr. Javier Gómez Castellanos, María de Lourdes González Lora, Cecilia Mandujano Gordillo y Amalia Arriaga Campos, gracias por todo su apoyo.

A mi familia por su comprensión y apoyo a lo largo de este periodo.

Esta investigación tuvo el soporte de Conacyt con la beca número 380281 y del Google Research Grant Program con el número de proyecto 23327.

Índice general

Índice general	7
1. Introducción	9
1.1. Problema	10
1.2. Objetivo general	12
1.2.1. Objetivos específicos	12
2. Marco de referencia	13
2.1. Regla 184	14
2.2. Ola verde	17
2.3. Métricas de desempeño	18
2.3.1. Curvas de optimalidad con la regla 184	19
3. Estado del arte	21
3.1. Dinámica vehicular	21
3.1.1. El modelo LAI	22
3.2. Control de semáforos	26
3.2.1. Semáforo auto-organizante reactivo	27
3.3. Vehículos autónomos	29
3.3.1. <i>Intersección basada en ranura</i>	30
4. Modificaciones al modelo LAI	37
4.1. Vehículos en las intersecciones	37
4.2. Semáforos en las intersecciones	38
4.2.1. Algoritmo LAI modificado para una ciudad tipo Manhattan	41

5. Sistema de detección	45
5.1. Componentes del sistema de detección deliberativo	46
5.2. Algoritmo de detección deliberativo	49
5.3. Experimentos y resultados con la regla 184	55
5.3.1. Errores en los sensores	57
5.4. Experimentos y resultados con el modelo LAI	59
6. Sistema de control	63
6.1. Estados del sistema de control	64
6.2. Otorgar la luz verde a la calle con mayor impulso vehicular	65
6.2.1. Calcular τ	68
6.2.2. Corrección del Algoritmo 6	69
6.3. Evitar obstruir la intersección	73
6.3.1. Corrección del Algoritmo 8	76
6.4. Acotar tiempos de espera	84
6.4.1. Corrección del Algoritmo 10	85
6.5. Integración de las reglas	91
6.6. Experimentos y resultados	91
6.6.1. Evaluación de las reglas	93
6.6.2. SAR vs SARIV	94
7. Comparación entre sistemas que coordinan el tránsito	97
7.1. Configuración de los experimentos	98
7.2. Regla 184	99
7.3. Modelo LAI	101
7.3.1. Sistemas de coordinación con vehículos manuales	101
7.3.2. Sistemas de semáforos con vehículos autónomos	103
7.4. Errores en los sensores	105
7.5. Ciudad de Manhattan con tránsito mixto	107
8. Conclusiones y trabajo futuro	111
Bibliografía	113

Capítulo 1

Introducción

El automóvil ha cambiado nuestras sociedades porque ha ofrecido una manera cómoda de desplazarnos. Su uso desmedido, sin embargo, ha tenido como consecuencia el excesivo tránsito vehicular en las grandes urbes. Los congestionamientos vehiculares tienen diversos efectos negativos en las zonas urbanas: el aumento de la contaminación, el consumo excesivo de combustible y el tiempo perdido de los usuarios.

La congestión del tránsito vehicular ocurre principalmente cuando la densidad (número de vehículos por unidad de longitud) crece hasta la capacidad máxima de las calles, inhibiendo el flujo de los vehículos. Los gobiernos han reglamentado el tránsito vehicular con la intención de reducir los congestionamientos. Algunas reglas han sido implementadas para mediar conflictos potenciales entre los vehículos. Por ejemplo, los carriles muestran el adecuado uso del espacio, los señalamientos de tránsito indican una conducción segura y los semáforos coordinan las intersecciones.

En comparación con métodos de control de semáforos tradicionales y conductores humanos, los *semáforos inteligentes* y *vehículos autónomos* tienen un potencial significativo para mejorar el tránsito vehicular porque tienen la capacidad de detectar las condiciones del entorno por medio de sensores y comunicarse entre sí.

Los *semáforos inteligentes* están compuestos por: 1) semáforos tradicionales; 2) una serie de sensores como bucles inductivos o cámaras; y 3) técnicas de inteligencia artificial como toma de decisiones, planificación o visión por computadora. Los principales sistemas propuestos que utilizan *semáforos inteligentes* detectan vehículos aproximándose y alejándose de cada intersección para conceder la preferencia a la dirección con mayor cantidad de vehículos aguas abajo [1, 2, 3, 4].

Los vehículos autónomos perciben su entorno por medio de sensores y clasifican los tipos de objetos que detectan, toman decisiones cuando las condiciones del ambiente cambian y planifican sus movimientos de acuerdo con los señalamientos de las carreteras. El desarrollo de los vehículos autónomos ha tenido un rápido progreso en los últimos años [5]. Actualmente, ya se encuentran en el mercado vehículos semi-autónomos que aplican la tecnología *autonomous cruise control* (ACC) con la finalidad de incrementar la seguridad y maximizar el espacio en las carreteras mediante el ajuste de la distancia entre los vehículos.

Las capacidades de los semáforos inteligentes y vehículos autónomos ofrecen la posibilidad de coordinar el tránsito de manera eficiente y segura porque, al contrario de vehículos operados por humanos, los vehículos autónomos no pierden concentración y mantienen distancias de seguridad. Por estas razones, es conveniente analizar y diseñar sistemas que aprovechen los beneficios que pueden ofrecer los semáforos inteligentes y los vehículos autónomos para mejorar el tránsito vehicular.

1.1. Problema

Los sistemas de control de semáforos y los comportamientos de conducción convencionales contribuyen a incrementar los congestionamientos vehiculares. Un sistema de semáforos es más eficiente si reduce los tiempos de espera promedio de los vehículos y si aumenta la utilización de las intersecciones, reduciendo así los tiempos de espera, el consumo de combustible y la contaminación.

Durante décadas, investigadores han utilizado métodos matemáticos y computacionales para encontrar períodos y fases apropiadas que regulen el cambio de luces de los semáforos [6, 7, 8, 9, 10]. Un sistema de control de lazo abierto denominado *ola verde* (OV) ha sido implementado en algunas áreas de muchas ciudades para controlar los cambios de luz de los semáforos. La OV intenta establecer periodos de tiempo de cambios de luz, de modo que los vehículos no tengan que detenerse.

La OV es útil porque es mejor que no tener ningún tipo de coordinación. Sin embargo, la OV no examina las condiciones del tránsito. En densidades bajas, algunos vehículos esperan detrás de una luz roja mientras la intersección no es utilizada. En densidades altas, se forman cuellos de botella en las intersecciones porque el semáforo no examina la saturación de la cuadra (ver Figura 1.1). Debido a que el sistema OV no se adapta a las condiciones del tránsito, es común que los vehículos se desplacen más lentamente que los cambios de luz sucesivos generados por la OV, provocando que los tiempos de espera se incrementen.



(a) En bajas densidades se presentan intersecciones ociosas



(b) En altas densidades se presentan intersecciones estancadas

Figura 1.1: Dos desventajas del sistema *ola verde*.

Una alternativa para mejorar el desempeño del tránsito vehicular es utilizar semáforos inteligentes [1, 2, 3]. En simulaciones por computadora, un semáforo inteligente denominado *semáforo auto-organizante* ha mostrado un desempeño superior al semáforo tradicional controlado por *ola verde* en una *ciudad de Manhattan* [4]. En este trabajo el semáforo auto-organizante se denomina *semáforo auto-organizante reactivo* (SAR), porque basa sus cambios de luz solo en los vehículos que detecta a cada instante de tiempo. Además, cuando nos referimos a una ciudad tipo Manhattan aludimos a una ciudad que tiene la estructura cuadriculada de la mayoría de las manzanas de la isla de Manhattan.

El SAR por medio de sus sensores detecta vehículos aproximándose a y alejándose de la intersección. El SAR ofrece un desempeño superior al sistema OV porque concede la preferencia al tránsito vehicular en la calle con mayor cantidad de vehículos esperando. Sin embargo, el SAR presenta inconvenientes y limitaciones:

- Retrasa oportunos cambios de luz en algunas densidades porque sólo concede la preferencia al tránsito, si la cantidad y el tiempo de espera de los vehículos en una calle con luz roja supera un umbral constante, incluso aunque en la dirección transversal que tiene la luz verde no haya vehículos.
- Requiere detectar k vehículos en una zona de aproximadamente 70 metros. Por esta razón, requiere de elaborados sensores para detectar los k vehículos aproximándose y alejándose de la

intersección. Dichos requerimientos limitan la implementación del SAR en las ciudades debido a su costo económico.

- No está diseñado para soportar fallas en los sensores.

Por los inconvenientes y limitaciones expuestas anteriormente, es necesario el diseño de *semáforos auto-organizantes* que incrementen el flujo vehicular en distintas densidades, reduzcan el número de sensores requeridos para implementarlos en las ciudades y sean tolerantes a fallas de los sensores. Además, es pertinente investigar el impacto que tendrá en el tránsito vehicular la utilización de vehículos autónomos en las ciudades.

1.2. Objetivo general

El objetivo general de esta tesis es **diseñar *semáforos auto-organizantes* para incrementar el flujo vehicular en las intersecciones.**

1.2.1. Objetivos específicos

- Desarrollar un modelo de tránsito vehicular basado en autómatas celulares para ciudades del tipo Manhattan.
- Diseñar un *sistema de detección* que detecte las condiciones del tránsito con sensores simples y que sea tolerante a fallas de dichos sensores.
- Diseñar un *sistema de control* que adapte los cambios de luz de acuerdo con las condiciones del tránsito vehicular.
- Evaluar el desempeño en términos del flujo vehicular sobre ciudades que cuenten con: miles de intersecciones, distintas densidades, distintos tipos de tránsito y distintos tipos de control de semáforos.

Capítulo 2

Marco de referencia

En este trabajo, el modelado del tránsito vehicular está basado en autómatas celulares (ACs). Un autómata celular (AC) es un modelo matemático para un sistema dinámico, compuesto por un conjunto de celdas que adquieren distintos estados. Dichos estados son modificados de un instante a otro en unidades de tiempo discreto. Cada estado de la celda es sensible a los estados de las celdas vecinas y evoluciona de acuerdo con una regla de transición local.

Los ACs han sido utilizados con éxito en distintas disciplinas. Por ejemplo, en Física es una técnica ampliamente usada para simular fenómenos de dinámica de fluidos. Los ACs tienen un papel destacado en el modelado del tránsito vehicular porque utilizan reglas de transición simples que son suficientes para simular el comportamiento observado en la dinámica vehicular. Además, los ACs son computacionalmente eficientes porque simulan el espacio, el tiempo y los estados de las celdas mediante variables discretas. Dichas características de los ACs permiten simular ciudades en el orden de miles de intersecciones.

Definición 2.0.1. Autómata celular finito. Formalmente, un autómata celular finito es una 4-tupla (Z^d, S, V, T) donde Z^d es una *rejilla* de tamaño finito, d es un entero que especifica la dimensión del AC, S es un *conjunto de estados* finito, V es un *operador de vecindad* definido por una serie de n elementos de Z^d y T es una función de transición.

La rejilla se define como el producto cartesiano Z^d en el espacio R^d . En una dimensión $d = 1$, una fila de celdas forman la rejilla y cada celda se indiza por un entero i . En dos dimensiones $d = 2$, las celdas son organizadas en una cuadrícula, y Z^d es el conjunto de índices (i, j) de enteros que representan la fila y la columna de una celda.

El operador de vecindad V es una serie de n elementos aplicados a una celda I . Por ejemplo, en una dimensión, el operador del vecindad $V = (-1, 0, 1)$ nos permite obtener los vecinos de la celda 72 mediante 3 desplazamientos (izquierda, centro y derecha): $i \rightarrow i - 1$, $i \rightarrow i$ y $i \rightarrow i + 1$. Por lo tanto, el vecindario: $V(72) = (71, 72, 73)$.

En dos dimensiones, el tipo de vecindad $V4 = ((0, 1), (0, -1), (-1, 0), (1, 0))$ se llama vecindario de Von Neumann (ver Figura 2.1) y hace posible el acceso a las cuatro celdas situadas: derecha, izquierda, abajo y arriba de la celda (i, j) . De manera que $V4(i, j) = ((i, j + 1), (i, j - 1), (i - 1, j), (i + 1, j))$.

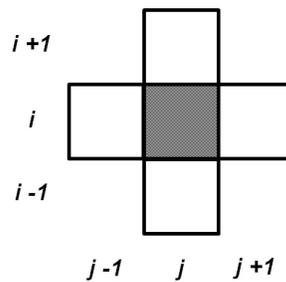


Figura 2.1: Vecindario de Von Neumann.

La entrada a_i de una celda i es la serie de n estados correspondientes a las celdas del operador de vecindad $V(i)$. Es decir, $a_i(s_j)_{j \in V(i)}$. Por lo tanto, para la vecindad $V(i) = (i - 1, i, i + 1)$ la serie de de sus estados es $a_i = (s_{i-1}, s_i, s_{i+1})$. El vecindario de una celda puede o no incluir a la celda misma. En cada celda i se tiene la forma $M_i = (S, T)$ donde S es el conjunto de estados y T es el mecanismo de transición que se denota como $s_i(t + 1) = T(a_i(t))$.

2.1. Regla 184

La regla 184 ha sido usado comúnmente para modelar la dinámica vehicular en un AC [9, 10, 11], donde los vehículos están representados por los 1s y los espacios están representados por los 0s. El operador del vecindad $V = (-1, 0, 1)$ se usa en la regla 184. A fin de simular el movimiento de los vehículos hacia la derecha, la regla 184 está basada en la siguiente función de transición:

$$T_{184}(a_i(t)) = \begin{cases} 1, & \text{si } (s_i(t) = 1 \wedge s_{i+1}(t) = 1) \vee (s_i(t) = 0 \wedge s_{i-1}(t) = 1) \\ 0, & \text{en caso contrario} \end{cases} \quad (2.1)$$

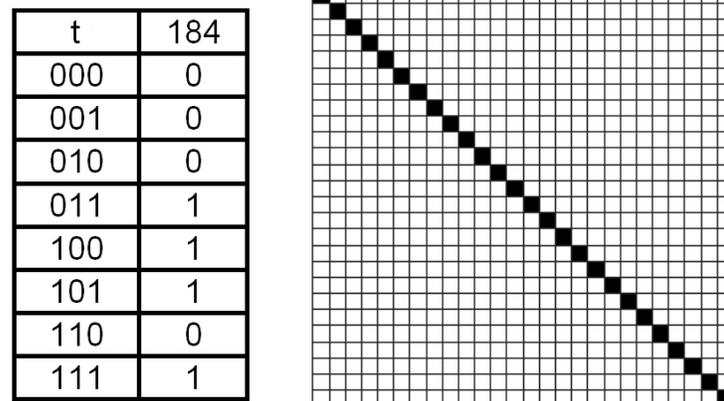


Figura 2.2: Tabla de transición y evolución de la regla 184.

La tabla de transición y la evolución de la regla 184 se muestra en la Figura 2.2.

En esta tesis utilizamos un autómata celular en dos dimensiones para representar una ciudad tipo Manhattan (ver Figura 2.3). Las calles de la ciudad de Manhattan bajo estudio, tienen un solo carril con un sentido asignado de cuatro posibles (este, oeste, norte y sur). Además, cada intersección tiene un semáforo.

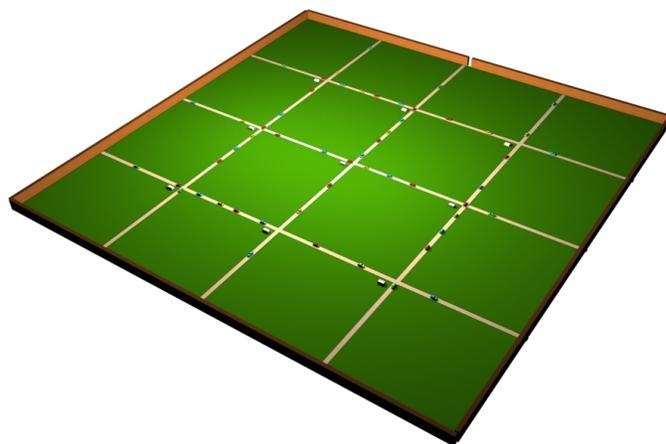


Figura 2.3: Ciudad de Manhattan de tamaño 3×3 calles y nueve intersecciones. En cada intersección hay un semáforo.

En la ciudad del tipo Manhattan se aplica la regla 184 en todas las celdas. Sin embargo, hay dos excepciones: Si un semáforo tiene luz roja, la regla 252 se aplica en la celda anterior a la intersección y la regla 136 se aplica en la celda posterior a la intersección. Las funciones de transición de las reglas 252 y 136 son las siguientes:

$$T_{252}(a_i(t)) = \begin{cases} 1, & \text{si } (s_i(t) = 1) \vee (s_i(t) = 0 \wedge s_{i-1}(t) = 1) \\ 0, & \text{en caso contrario} \end{cases} \quad (2.2)$$

$$T_{136}(a_i(t)) = \begin{cases} 1, & \text{si } (s_i(t) = 1 \wedge s_{i+1}(t) = 1) \\ 0, & \text{en caso contrario} \end{cases} \quad (2.3)$$

La Figura 2.4 muestra las tablas de transición para las reglas 252 y 136.

t	252	136
000	0	0
001	0	0
010	1	0
011	1	1
100	1	0
101	1	0
110	1	0
111	1	1

Figura 2.4: Tablas de transición de la regla 252 y 136.

La regla de transición 252 tiene la finalidad de detener el flujo vehicular antes de la intersección. La regla de transición 136 tiene la finalidad de prevenir que los vehículos que están en la intersección y que se mueven en la calle con luz verde, entren a la calle con luz roja. La celda correspondiente a la intersección es un caso especial debido a que tiene cuatro vecinos posibles, la regla en la intersección nunca cambia (siempre se aplica 184), lo que cambia son sus vecinos. La celda correspondiente a la intersección toma como vecinos a las dos celdas de la calle que tiene el semáforo en luz verde. Un diagrama de las celdas alrededor de una intersección se muestra en la Figura 2.5.

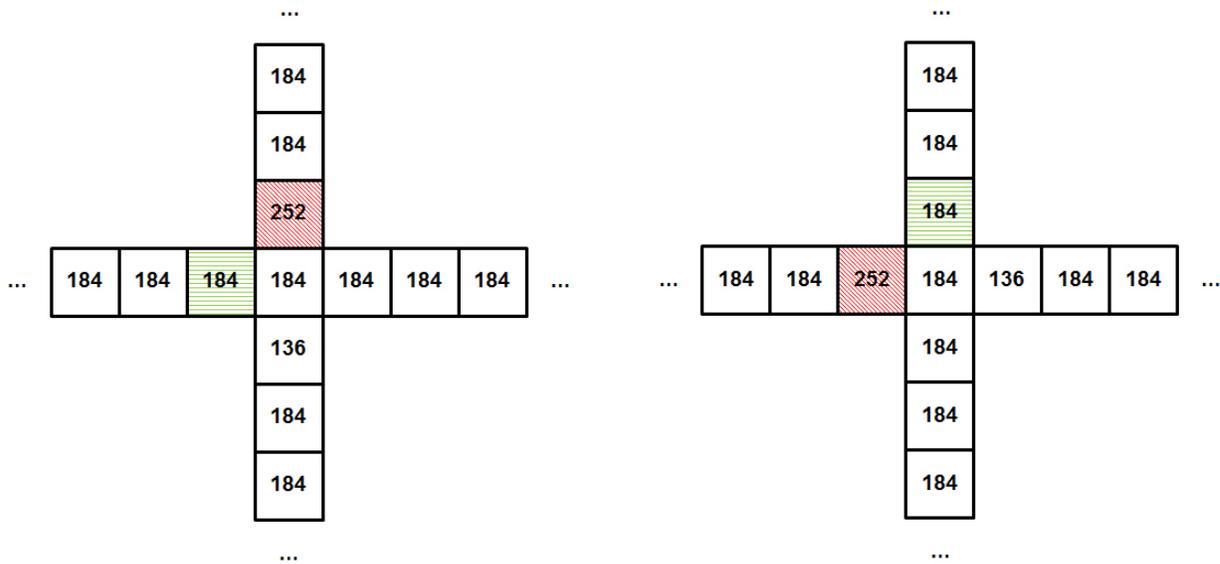


Figura 2.5: Tablas de transición de la regla 252 y 136.

2.2. Ola verde

El sistema de control *ola verde* (OV) ha sido implementado en algunas áreas de muchas ciudades para coordinar el tránsito vehicular en las intersecciones. La idea esencial del sistema *ola verde* (OV) es que las luces cambian consecutivamente de acuerdo con el tiempo de viaje esperado en que los vehículos arribarán a las intersecciones. Así, la ola de luces verdes se mueve a través de la calle a la misma velocidad esperada que los vehículos.

Este método tiene ventajas, por ejemplo, cuando la mayor parte del tránsito fluye en el sentido de la ola verde en densidades bajas. Sin embargo, los vehículos que fluyen en la dirección perpendicular a la ola verde se retrasarán. Además, si el tránsito fluye a una velocidad más baja de lo esperado, la ola verde se difundirá más rápidamente que los vehículos y éstos tendrán que esperar.

Para implementar el sistema OV en nuestro autómeta celular se requiere sincronizar el ciclo de tiempo T del semáforo con el tiempo de viaje de los vehículos. La sincronización se logra solo si la longitud de la calle es múltiplo de T y los bordes son periódicos. El ciclo T se divide en $\frac{T}{2}$ para la luz verde y roja, respectivamente.

Al estado del semáforo σ_j puede asignarse *verde* en la dirección horizontal (este u oeste) y por lo tanto la luz en *roja* estará en la dirección vertical (norte o sur). El estado σ_j de los semáforos se inicializa aplicando la ecuación (2.4).

$$\sigma_j = \begin{cases} verde_{vertical} & \text{Si } \lfloor ((x - y) \text{ mód } T) + 0.5 \rfloor \geq \frac{T}{2} \\ verde_{horizontal} & \text{en caso contrario} \end{cases} \quad (2.4)$$

La ecuación (2.4) inicializa el estado de un semáforo para que tenga una luz vertical *verde* si el entero más próximo de la coordenada x menos la coordenada y , módulo del ciclo T , es mayor o igual que medio ciclo. En caso contrario, al estado se asigna *verde* en la dirección horizontal. Con la ecuación (2.4), el 50 % de las intersecciones de cada calle (vertical y horizontal) tendrán un estado vertical en *verde* y la otra mitad un estado vertical en *roja*.

Además, es necesario un desfase individual w_j para cada semáforo que es calculado aplicando la ecuación (2.5).

$$w_j = \left\lfloor \left((x - y) \text{ mód } \frac{T}{2} \right) + 0.5 \right\rfloor \quad (2.5)$$

La ecuación (2.5) establece el desfase redondeando al entero más cercano a la coordenada x menos la coordenada y , módulo de medio ciclo. Los desfases crean olas verdes al sur y al este. La idea esencial del establecimiento del desfase es tener intersecciones que contienen el mismo valor de desfase de manera diagonal, con el fin de cambiar sus luces consecutivamente.

2.3. Métricas de desempeño

El comportamiento del modelo de tránsito vehicular depende de la densidad de vehículos que hay en la ciudad. En nuestra ciudad de Manhattan el rango de la densidad es $\rho \in [0, 1]$. Los casos triviales son los extremos, $\rho = 0$ donde no hay vehículos, $\rho = 1$ donde hay la mayor cantidad de vehículos, de modo que no hay espacio para moverse y el flujo es cero. En nuestra ciudad de Manhattan se aplican *condiciones de frontera periódicas*. Por esta razón, siempre se conserva la misma densidad.

La densidad se calcula dividiendo el número de vehículos $\sum s_i$, entre la máxima cantidad de vehículos $|S|$ que puede contener la ciudad, ver ecuación (2.6).

$$\rho = \frac{\sum s_i}{|S|} \quad (2.6)$$

El desempeño del sistema es medido con la velocidad promedio v , que es simplemente la suma de las velocidades de los vehículos $\sum v_i$ por el número de vehículos $\sum s_i$, ver ecuación (2.7).

$$v = \frac{\sum v_i}{\sum s_i} \quad (2.7)$$

El flujo J del sistema representa la cantidad de espacio utilizado por los vehículos en movimiento. El flujo J se obtiene multiplicando la densidad de vehículos por la velocidad promedio, ver ecuación (2.8).

$$J = \rho v \quad (2.8)$$

Si hay pocos vehículos, entonces el flujo disminuye porque no hay movimiento en el espacio. Si hay muchos vehículos, entonces el flujo también disminuye porque los vehículos no pueden moverse.

Los métricas presentadas anteriormente enfatizan el comportamiento promedio de los vehículos en términos de densidad, velocidad y flujo. Dichas métricas son macroscópicas porque son capaces de capturar a lo largo del tiempo la cantidad de flujo promedio que se desplaza en el espacio total de la ciudad tipo Manhattan. No obstante, como trabajo futuro es posible aplicar métricas microscópicas que permitan evaluar otras características del tránsito vehicular a nivel local. Por ejemplo, evaluar la dinámica e interacción de los vehículos para obtener histogramas de velocidades, intervalos entre vehículos, congestionamientos en determinadas intersecciones, etc.

2.3.1. Curvas de optimalidad con la regla 184

Si cada intersección en un sistema de semáforos tiene el desempeño máximo de vehículos fluyendo, entonces el sistema logra teóricamente el desempeño máximo total. Un desempeño total inferior implica que hay interferencia entre los semáforos. Dadas las propiedades de la regla 184, las ecuaciones 2.9 y 2.10 describen las curvas de optimalidad para intersecciones respecto a la velocidad y el flujo, que dependen del flujo máximo $J_{m\acute{a}x}$ permitido por una intersección.

$$v_{\acute{o}ptimo} = \begin{cases} 1 & \text{Si } \rho \leq J_{m\acute{a}x} \\ \frac{J_{m\acute{a}x}}{\rho} & \text{Si } J_{m\acute{a}x} < \rho < 1 - J_{m\acute{a}x} \\ \frac{1-\rho}{\rho} & \text{Si } 1 - J_{m\acute{a}x} \leq \rho \end{cases} \quad (2.9)$$

$$J_{\text{óptimo}} = \begin{cases} \rho & \text{Si } \rho \leq J_{\text{máx}} \\ J_{\text{máx}} & \text{Si } J_{\text{máx}} < \rho < 1 - J_{\text{máx}} \\ 1 - \rho & \text{Si } 1 - J_{\text{máx}} \leq \rho \end{cases} \quad (2.10)$$

Si $\rho \leq J_{\text{máx}}$, entonces la intersección soporta una velocidad máxima de 1 celdas/paso. Con base en que $J = \rho v$, el flujo es igual a la densidad ρ , ya que todos los vehículos se están moviendo. Si $J_{\text{máx}} < \rho < 1 - J_{\text{máx}}$, entonces el flujo de la intersección está restringido por la capacidad máxima de la intersección $J_{\text{máx}}$. Esto implica que los vehículos estarán usando la intersección en todo momento, y la velocidad promedio será $J_{\text{máx}}/\rho$. Si la densidad de las calles es tan alta que restringe el flujo de vehículos en las calles, entonces el flujo se reduce a $1 - \rho$ y la velocidad a $\frac{1-\rho}{\rho}$.

Observe que el flujo óptimo J_{ptimo} es simétrico, porque hay una simetría en el modelo de tránsito vehicular con la regla 184 entre vehículos (1s) moviéndose en un sentido y espacios (0s) moviéndose en el sentido opuesto.

Capítulo 3

Estado del arte

3.1. Dinámica vehicular

La simulación computacional es un medio importante para evaluar, controlar y administrar el tránsito vehicular [12]. Los autómatas celulares (ACs) tienen un papel destacado en el modelado del tránsito vehicular porque utilizan reglas de transición simples que son suficientes para simular el comportamiento observado en la dinámica vehicular. Además, los ACs son computacionalmente eficientes porque simulan el espacio, el tiempo y los estados de las celdas mediante variables que tienen valores discretos.

El modelo de tránsito vehicular NaSch está basado en un autómata celular (AC) y fue propuesto por Nagel y Schreckenberg [13]. Los vehículos se caracterizan por tener una velocidad variable discreta, una aceleración, una desaceleración (debido a la presencia de otros vehículos) y una tendencia aleatoria a reducir la velocidad. El modelo NaSch simula la tendencia humana a reaccionar de forma exagerada al desacelerar y simula los *congestionamientos fantasma* dependiendo de la densidad vehicular. Se han propuesto diversas variantes del modelo NaSch, cada una con diferentes grados de detalle. Por ejemplo, Nagel y Paczuski modelan vehículos que si viajan a la velocidad máxima no aplican una desaceleración aleatoria [14]. Esta variación elimina los congestionamientos fantasma.

El modelo de tránsito vehicular BML es un AC en dos dimensiones y fue propuesto por Biham et al. [15]. En el modelo BML cada celda representa un espacio vacío o un vehículo que fluye en una determinada dirección. El BML es determinista, a excepción de las posiciones aleatorias iniciales de los vehículos en la ciudad. En el modelo BML se aplican condiciones periódicas en la frontera de modo que el número de vehículos se conserva. En pasos pares, solo se mueven los vehículos que

fluyen hacia el Norte, mientras que en pasos impares, solo se mueven los vehículos que fluyen al este, a menos que haya un espacio no vacío adelante.

Algunos modelos de tránsito vehicular se han desarrollado como una generalización de los modelos BML [16, 17, 18, 19]. Dichas generalizaciones son esencialmente una extensión del modelo BML con calles de longitud arbitraria y vehículos que se comportan de acuerdo con el modelo NaSch. Simon y Nagel desarrollaron una combinación más elaborada de los modelos NaSch y BML [16]. Estos autores modelan calles con diferentes capacidades. Por ejemplo, calles con diferente número de carriles.

El modelo LAI fue propuesto por Lárraga y Álvarez-Icaza [20]. El modelo LAI simula distancias de seguridad entre los vehículos y el tiempo de reacción al momento de acelerar o frenar. Además, en el modelo LAI los vehículos se desplazan en diferentes velocidades. En este trabajo se utiliza el modelo LAI porque la dinámica del movimiento de los vehículos es más acorde con la dinámica del tránsito vehicular real que los modelos de tránsito vehicular mencionados previamente. El modelo LAI se describe con mayor detalle en la siguiente sección.

3.1.1. El modelo LAI

La regla 184 simula el comportamiento de la dinámica vehicular de forma poco realista. Al aplicar la regla 184, los vehículos se mueven con velocidad constante (una celda por paso de tiempo), sin importar las distancias que hay entre sí (ver sección 2.1). El modelo LAI simula distancias de seguridad y tiempos de reacción al momento de acelerar o desacelerar de los vehículos. Además, los vehículos se desplazan a diferentes velocidades. Por estas razones, la simulación de la dinámica vehicular mediante el modelo LAI es más acorde con la dinámica de los vehículos. Las variables y los parámetros del modelo LAI se muestran en la Tabla 3.1.

El modelo LAI está basado en el cálculo de tres distancias de seguridad para el vehículo i con respecto al vehículo de adelante $i + 1$, en el tiempo t . Dichas distancias permiten que el vehículo i tome la decisión de: acelerar, mantener su velocidad o desacelerar. Las distancias de seguridad se denotan como d_a , d_m y d_d y están definidas en las ecuaciones 3.1, 3.2 y 3.3.

$$d_a = \max \left(0, \sum_{k=0}^{(v_i(t)+\Delta v) \operatorname{div} M} [(v_i(t) + \Delta v) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \operatorname{div} M} [(v_{i+1}(t) - M) - kM] \right) \quad (3.1)$$

Variables		Parámetros	
x_i	Posición del vehículo i .	v_{max}	Velocidad máxima
v_i	Velocidad del vehículo i .	v_s	Velocidad baja.
d_a	Distancia de aceleración segura.	l_s	Longitud del vehículo (en celdas).
d_d	Distancia de desaceleración segura.	M	Disminución máxima de la velocidad en un paso de tiempo.
d_m	Distancia de mantenimiento de velocidad segura.	R_s	Probabilidad de desacelerar.
R_a	Probabilidad de acelerar.	R_0	Probabilidad de acelerar a baja velocidad.
R_d	Probabilidad de desacelerar.	Δv	Capacidad de acelerar o desacelerar en condiciones normales.

Tabla 3.1: Parámetros del modelo LAI.

$$d_m = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)) \text{ div } M} [v_i(t) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \text{ div } M} [(v_{i+1}(t) - M) - kM] \right) \quad (3.2)$$

$$d_d = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)-\Delta v) \text{ div } M} [(v_i(t) - \Delta v) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \text{ div } M} [(v_{i+1}(t) - M) - kM] \right) \quad (3.3)$$

El vehículo i tiene una velocidad $v_i(t)$ en el paso de tiempo t , una magnitud Δv para aumentar o disminuir su velocidad. El vehículo i y el vehículo de adelante $i + 1$ tienen una disminución máxima de la velocidad M en un paso de tiempo. El vehículo de adelante $i + 1$ tiene una velocidad $v_{i+1}(t)$ en el paso de tiempo t . $X \text{ div } Y$ indica la división entera, es decir, $X \text{ div } Y = \lfloor X/Y \rfloor$, donde $\lfloor z \rfloor$ es la función suelo. En las sumatorias, si el límite superior es menor que el límite inferior, entonces la suma se define como 0.

El segundo término de las ecuaciones 3.1, 3.2 y 3.3 representa la distancia que recorre el vehículo de adelante $i + 1$ en el peor de los casos cuando disminuye su velocidad $(v_{i+1}(t) - M) \text{ div } M$ veces hasta que $v_{i+1}(t) = 0$. El primer término de las ecuaciones 3.1, 3.2 y 3.3 representa la distancia que recorre el vehículo i cuando disminuye su velocidad $(v_i(t) + \Delta v) \text{ div } M$; $(v_i(t)) \text{ div } M$; y $(v_i(t) - \Delta v) \text{ div } M$ veces hasta que $v_i(t) = 0$, respectivamente.

La ecuación (3.1) representa la distancia d_a que determina si el vehículo i puede aumentar o disminuir su velocidad en Δv . La ecuación (3.2) representa la distancia d_m que determina si el vehículo

i puede o tiene que disminuir su velocidad en Δv . La ecuación (3.3) representa la distancia d_d que determina si el vehículo i tiene que disminuir su velocidad en Δv o en M .

El siguiente paso es obtener el valor correspondiente al ruido estocástico R_a que indica la *probabilidad de acelerar* basado en la velocidad del vehículo v_i . La fórmula para calcular el ruido estocástico es:

$$R_a = \text{mín}(R_d, R_0 + v_i(t) \cdot (R_d - R_0/v_s)), \quad (3.4)$$

donde v_s es constante y ligeramente por arriba de 0, $0 < R_a \leq 1$, y $0 < R_0 < R_d \leq 1$. Los parámetros estocásticos R_0 y R_d controlan las fluctuaciones de las velocidades de los vehículos. El parámetro R_a interpola linealmente entre R_0 y R_d ($R_0 < R_d$). Además, es más pequeño que la velocidad v_s .

El parámetro R_a simula cómo los conductores actúan de manera diferente al momento de acelerar, dependiendo de las condiciones del tránsito. Los vehículos cuya velocidad $v_i < v_s$ en un paso de tiempo anterior tienen una probabilidad más baja de acelerar que el resto de los vehículos en movimiento. Por un lado, cuando $R_a = 1$ el conductor reacciona de manera inmediata y siempre acelera. Por otro lado, cuando $R_a = 0$ el conductor nunca acelera y el vehículo tiende a detenerse.

El parámetro R_s indica la probabilidad de frenar más de lo necesario al aproximarse al vehículo que hay adelante. El parámetro R_s simula cómo los conductores reducen su velocidad debido a frenar más de lo necesario por cansancio o distracciones. Por un lado, cuando $R_s = 1$ el conductor siempre frena más de lo necesario. Por otro lado, cuando $R_s = 0$ el conductor mantiene una velocidad acorde a las condiciones del tránsito por lo que nunca desacelera más de lo necesario.

Posteriormente se determina la velocidad del vehículo i dependiendo de la distancia que hay con el vehículo de adelante $i + 1$. La distancia desde el vehículo i al vehículo $i + 1$, se define como $d_i(t) = x_{i+1}(t) - x_i(t) - l_s$; donde l_s indica la longitud del vehículo. La velocidad de los vehículos se modifica de acuerdo con las siguientes reglas:

- **Acelerar**, si $d_i(t) \geq d_a$, la velocidad del vehículo i se incrementa aleatoriamente por una probabilidad R_a .

$$v_i(t + 1) = \begin{cases} \text{mín}(v_i(t) + \Delta v, v_{max}), & \text{si randf() } \leq R_a \\ v_i(t), & \text{en caso contrario} \end{cases} \quad (3.5)$$

Esta regla modela una velocidad máxima que los conductores tratan de alcanzar. Además, esta regla modela una aceleración no uniforme porque los conductores actúan de manera diferente

dependiendo del tránsito local. La función $randf()$ retorna valores enteros aleatorios, distribuidos uniformemente en el intervalo cerrado $[0, 1]$.

- **Desacelerar aleatoriamente**, si $d_a > d_i(t) \geq d_m$, la velocidad del vehículo i se reduce por una probabilidad R_s .

$$v_i(t+1) = \begin{cases} \text{máx}(v_i(t) - \Delta v, 0), & \text{si } randf() \leq R_s \\ v_i(t), & \text{en caso contrario} \end{cases} \quad (3.6)$$

Esta regla modela una velocidad constante que los conductores tratan de mantener si la distancia con respecto al vehículo de adelante es segura. Además, esta regla también introduce perturbaciones de tránsito que causan los conductores al reducir su velocidad sin una razón aparente.

- **Frenar**, si $d_m > d_i(t) \geq d_d$ y $v_i(t) > 0$, la velocidad del vehículo i se reduce.

$$v_i(t+1) = \text{máx}(v_i(t) - \Delta v, 0) \quad (3.7)$$

Esta regla modela una desaceleración que el conductor aplica si la distancia que separa su vehículo del vehículo de adelante es moderadamente pequeña.

- **Frenar urgentemente**, si $v_i(t) > 0$ y $d_i(t) < d_d$, si la velocidad del vehículo i no es inferior a 0, entonces se reduce en M .

$$v_i(t+1) = \text{máx}(v_i(t) - M, 0) \quad (3.8)$$

Esta regla modela un frenado de emergencia que el conductor aplica ante situaciones urgentes relacionadas con su seguridad.

Posteriormente, la posición $x_i(t+1)$ denota la posición del vehículo i en el paso de tiempo $t+1$ y se calcula como $x_i(t+1) = x_i(t) + v_i(t+1)$. Cada vehículo se mueve hacia adelante en función de su nueva velocidad $v_i(t+1)$ determinada por las reglas anteriormente mencionadas. Por último, Δx indica la longitud de la celda y se utiliza para discretizar el sistema.

3.2. Control de semáforos

En comparación con métodos de control de semáforos tradicionales como la ola verde, los *semáforos inteligentes* tienen un potencial significativo para mejorar el tránsito vehicular porque tienen la capacidad de detectar las condiciones del entorno por medio de sensores y comunicarse entre sí. Los semáforos inteligentes tienen una serie de sensores y técnicas de inteligencia artificial para detectar vehículos aproximándose y alejándose de cada intersección y conceder la preferencia a la dirección basado en determinadas reglas [1, 2, 3, 4].

El semáforo inteligente propuesto por Helbing y Lämmer está inspirado en la auto-organización de los flujos peatonales que ocurre en los cuellos de botella [21]. La auto-organización surge de las diferencias de presión entre los grupos de vehículos que esperan en ambos lados del cuello de botella. La presión se acumula en el lado donde hay más vehículos que están esperando, mientras que se reduce en el lado donde los vehículos logran pasar el cuello de botella. La dirección de paso cambia cuando la presión en un lado excede la presión en el otro lado en una cantidad suficiente. Dicha idea es aplicada en este semáforo inteligente para otorgar la preferencia a cierto flujo vehicular. Las interacciones locales consideradas conducen a patrones de coordinación emergentes tales como olas verdes y logran un control de semáforos eficaz y descentralizado.

El semáforo inteligente propuesto por Gier et al. adquiere el estado del tránsito vehicular y con base en información estadística otorga la preferencia a un determinado flujo vehicular [22]. Este semáforo inteligente muestra en simulaciones por computadora mayor eficiencia que los semáforos tradicionales.

Goel et al. investigan el uso de un algoritmo de control de semáforos basado en la auto-organización [23]. Los algoritmos basados en la auto-organización se comportan mejor que los algoritmos tradicionales. Se desarrolla un modelo de tránsito vehicular que tiene una carretera con múltiples intersecciones y se usan datos reales del tránsito vehicular.

Las investigaciones de Cesmea y Furth exploran un paradigma de semáforos auto-organizantes basado en reglas locales [24]. La simulación de los semáforos auto-organizantes en VISSIM muestra reducciones del tiempo de viaje. Las calles simuladas son los corredores arteriales de Massachusetts y Arizona.

Un sistema de semáforos inteligentes que ha mostrado mejoras en el aumento del flujo vehicular sobre distintas densidades en comparación con la *ola verde* (OV) es el *semáforo auto-organizante reactivo* (SAR) [4]. Además, es un sistema que se adapta a las condiciones del tránsito y que no requiere comunicación con sus vecinos. El SAR se describe con mayor detalle en la siguiente sección.

3.2.1. Semáforo auto-organizante reactivo

Un sistema de semáforos que pretende optimizar las fases del cambio de luces entre todos los semáforos tiene un alto costo computacional porque la búsqueda de posibles soluciones es un problema EXP-completo [25]. Además, los flujos de tránsito y densidades vehiculares cambian constantemente. Por esta razón, es preferible un semáforo que se adapte a las densidades actuales del tránsito vehicular.

El semáforo auto-organizante reactivo (SAR) esencialmente otorga preferencia a la calle que tiene más vehículos. Sin requerir comunicación, los vehículos se auto-organizan en grupos que fluyen más rápidamente que los vehículos con la *ola verde*. Los grupos pueden activar una luz verde antes de llegar a la intersección, deteniéndose solo cuando otro grupo está cruzando en la dirección transversal. El SAR está compuesto por un *sistema de detección* y un *sistema de control*.

El sistema de detección del SAR se denomina *sistema de detección reactivo*. El *sistema de detección reactivo* detecta las posiciones de k vehículos en una zona de detección compuesta por las distancias d , e , y la intersección (ver Figura 3.1). El sistema de detección se denomina *reactivo* porque las posiciones de los vehículos son enviadas directamente al *sistema de control* para provocar reacciones inmediatas. Los parámetros del SAR son descritos en la Tabla 3.2.

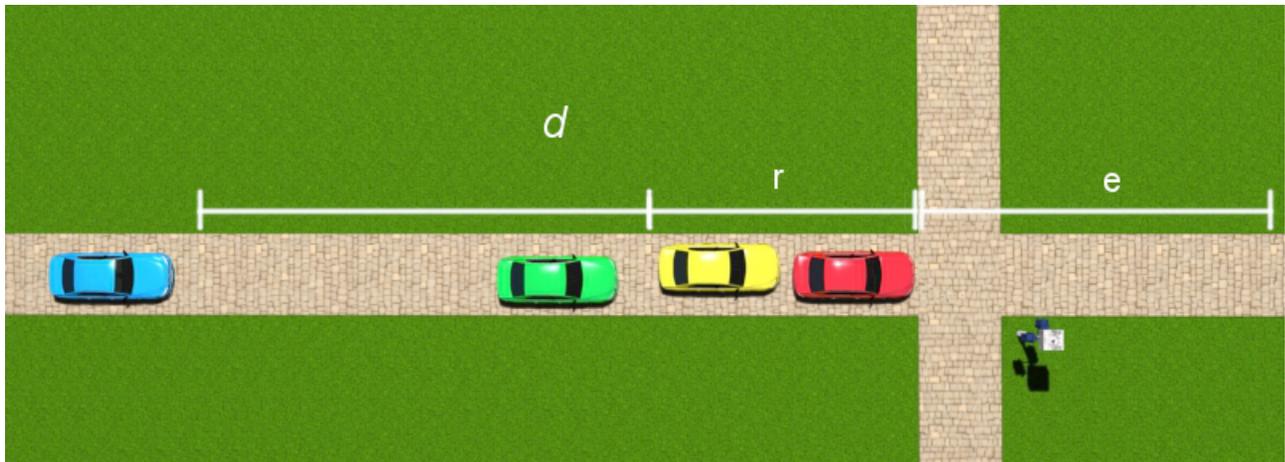


Figura 3.1: El semáforo auto-organizante reactivo detecta vehículos en las distancias d , r y e .

El *sistema de control* del SAR tiene seis reglas que controlan los cambios de luz. Las reglas son numeradas de mayor a menor orden de ejecución; es decir, la regla con un número mayor se ejecuta antes que las reglas con un número inferior. Por ejemplo, la regla 5 se ejecuta antes que las reglas 1 – 4. El conjunto de reglas del *sistema de control* se muestra a continuación.

Parámetros	Descripción
Δt	Paso de tiempo en la simulación.
d	Distancia en donde los vehículos son detectados al aproximarse a la intersección.
r	Distancia corta en donde los vehículos son detectados al aproximarse a la intersección.
e	Distancia donde los vehículos detenidos son detectados.
u	Tiempo mínimo de espera en una luz verde.
w	Tiempo máximo de espera en una luz roja.
n	Número acumulado de vehículos que están en la distancia d .
m	Número de vehículos que están en la distancia r .

Tabla 3.2: Parámetros usados por el SAR.

Sistema de control auto-organizante

1. A cada paso, se suma a un contador el número de vehículos que se aproximan o esperan en un semáforo en rojo a una distancia d . Cuando este contador supera un umbral n , la luz del semáforo cambia a verde y el contador se restablece a 0.
2. El semáforo permanece en luz verde durante un tiempo mínimo u y un tiempo máximo w .
3. Si algunos vehículos m o menos, pero más que cero están cruzando una luz verde a una corta distancia r , entonces la luz no cambia.
4. Si ningún vehículo se aproxima a una luz verde dentro de una distancia d y al menos un vehículo se acerca a la luz roja dentro de una distancia d , entonces la luz cambia.
5. Si hay un vehículo detenido más allá de una luz verde a una corta distancia e , entonces la luz cambia a roja.
6. Si hay vehículos detenidos más allá de la intersección en ambas direcciones a corta distancia e , entonces las dos luces cambian a rojo. Una vez que una de las direcciones esté libre se restaura la luz verde en esa dirección.

Cabe mencionar que un tiempo máximo w se agregó a la regla 2 en este trabajo, porque de otra manera cuando ocurren errores de detección, los vehículos en bajas densidades pueden permanecer indefinidamente detrás de una luz roja.

La regla 1 trata de cambiar una luz roja a verde cuando muchos vehículos se aproximan a la intersección. La regla 2 impide el cambio excesivamente rápido de las luces en altas densidades y la permanencia por demasiado tiempo de una luz verde. La regla 3 evita que las colas de los grupos se corten, promoviendo la integridad de los grupos. La regla 4 permite un rápido cambio de luces en bajas densidades, ya que vehículos aislados pueden activar el cambio de luces cuando se acercan a una intersección sin requerir que formen un grupo. Las reglas 5 y 6 evitan cuellos de botella causados por vehículos detenidos en una intersección.

3.3. Vehículos autónomos

Debido a los avances en el desarrollo de los vehículos autónomos, se han propuesto diversos sistemas para administrar el tránsito vehicular en las intersecciones [26]. En esta sección se presentan los trabajos relacionados a la simulación de vehículos autónomos o semi-autónomos y mecanismos de administración propuestos para coordinar el tránsito vehicular en las intersecciones. A diferencia de un vehículo autónomo, un vehículo manual es un vehículo operado por un humano.

En el trabajo de Yuan et al. [27] se muestra el flujo vehicular de un modelo de tránsito híbrido que está compuesto de vehículos con tecnología *autonomous cruise control* (ACC) y vehículos manuales. En el modelo de tránsito híbrido, los vehículos ACC se simulan mediante el uso de un modelo *car-following* y los vehículos manuales se modelan por un autómata celular. Los tres estados del tránsito vehicular son reproducidos (flujo libre, flujo sincronizado, flujo congestionado) mediante vehículos manuales.

El trabajo de Au et al. [28] supone que habrá un período de transición en el que coexistirán en las ciudades vehículos manuales y vehículos semi-autónomos, por lo que se propone un nuevo protocolo denominado *administración de intersección semi-autónoma*, que permite a los vehículos con características parcialmente autónomas, como el ACC, entrar en una intersección con diferentes direcciones en las calles. Los experimentos muestran que este protocolo puede disminuir los congestionamientos de tránsito cuando la mayoría de los vehículos son semi-autónomos. El estudio muestra que a medida que ingresan más vehículos con características autónomas el flujo vehicular aumenta.

Se han propuesto protocolos de control que explotan las capacidades de los vehículos autónomos para coordinarlos en las intersecciones [26]. La idea esencial consiste en desarrollar un protocolo basado en un enfoque de reservación. Un servidor es colocado en cada intersección denominado *administrador de la intersección* (AI). Los vehículos autónomos que se aproximan a la intersección se comunican con el AI para solicitarle tiempo y espacio dentro de la intersección. El AI considera todas las solicitudes y determina que vehículos son aceptados y rechazados. Los resultados obtenidos con este sistema muestran que el tiempo de espera por parte de los vehículos autónomos es menor en comparación con sistemas de control de semáforos convencionales.

El sistema *Intersección basada en ranura* (IBR) administra los tiempos de acceso de vehículos autónomos en las intersecciones [29]. En la siguiente sección se explica dicho sistema.

3.3.1. *Intersección basada en ranura*

El sistema *Intersección basada en ranura* (IBR) administra los tiempos de acceso de vehículos autónomos en las intersecciones [29]. El sistema IBR asigna a cada vehículo autónomo un intervalo de tiempo de acceso a la intersección.

En el sistema IBR hay dos tipos de algoritmos para asignar el acceso a la intersección de los vehículos autónomos, denominados *FAIR* y *BATCH*. El algoritmo *FAIR* asigna un tiempo de acceso a los vehículos autónomos de acuerdo con el orden de llegada en la intersección; es decir, con base en el criterio *primero en llegar primero en ser servido*. En lugar de una estrategia uno a uno, el algoritmo *BATCH* concede el acceso a la intersección a un lote de vehículos para aumentar el flujo vehicular [29]. La estrategia *BATCH* es equivalente a la estrategia *FAIR* si el lote de vehículos formado se restringe a un solo vehículo. Por esta razón, en las secciones siguientes se explica brevemente solo el sistema IBR aplicando el algoritmo *BATCH*.

La idea esencial del algoritmo *BATCH* es la siguiente: si aumenta la densidad vehicular, entonces el formar lotes de vehículos para conceder su acceso a la intersección es más eficiente que conceder a cada vehículo acceso a la intersección de manera individual [29]. Un lote de vehículos es formado con base en un intervalo de tiempo de retraso Δ . Por ejemplo, en la Figura 3.2 se muestran cinco vehículos que solicitan un tiempo de acceso a la intersección en el orden de arribo. El primer vehículo tiene un retardo Δ . Por esta razón, las solicitudes de acceso son organizadas por lotes.

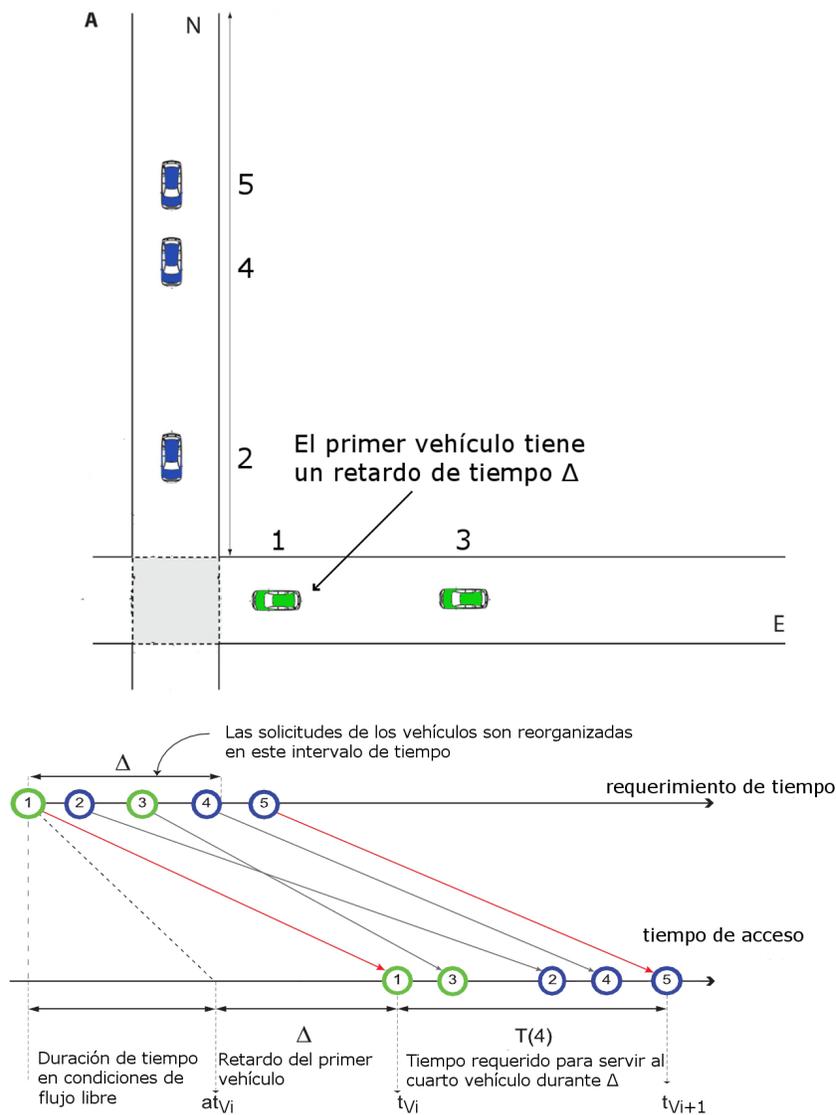


Figura 3.2: La intersección tiene dos flujos de entrada norte (N) y este (E). Todos los vehículos tienen la misma longitud l_s . El tiempo de acceso a la intersección de un vehículo se define como el momento en que su frente ingresa al área de la intersección. El valor del intervalo de tiempo para organizar las solicitudes de los vehículos es igual al retraso $\Delta = (t_{Vi} - at_{Vi})$ del vehículo 1. Los tiempos de acceso de los vehículos 2, 3 y 4 son organizados dentro del tiempo de retraso Δ . El vehículo 1 y el vehículo 3 pertenecen al mismo flujo. Por esta razón, el vehículo 3 tiene acceso a la intersección antes que el vehículo 2. El proceso se repite utilizando el vehículo 5 como referencia para calcular el siguiente Δ . A fin de evitar grupos de tamaño arbitrario, *BATCH* determina un límite N en el número total de vehículos para un grupo. Cuando $N = 1$, *BATCH* es equivalente a *FAIR*.

Un vehículo autónomo V recibe un tiempo de acceso T dependiendo de su tipo de flujo, norte (N) o este (E), y del tipo de flujo del vehículo autónomo V' que tiene adelante. Si ambos vehículos pertenecen al mismo flujo, entonces $T = T_1$ donde T_1 es calculado a partir de una distancia d_{tail} . En caso contrario, el tiempo $T = T_2$ donde T_2 es calculado a partir de una distancia d_{stop} (ver Figura 3.3).

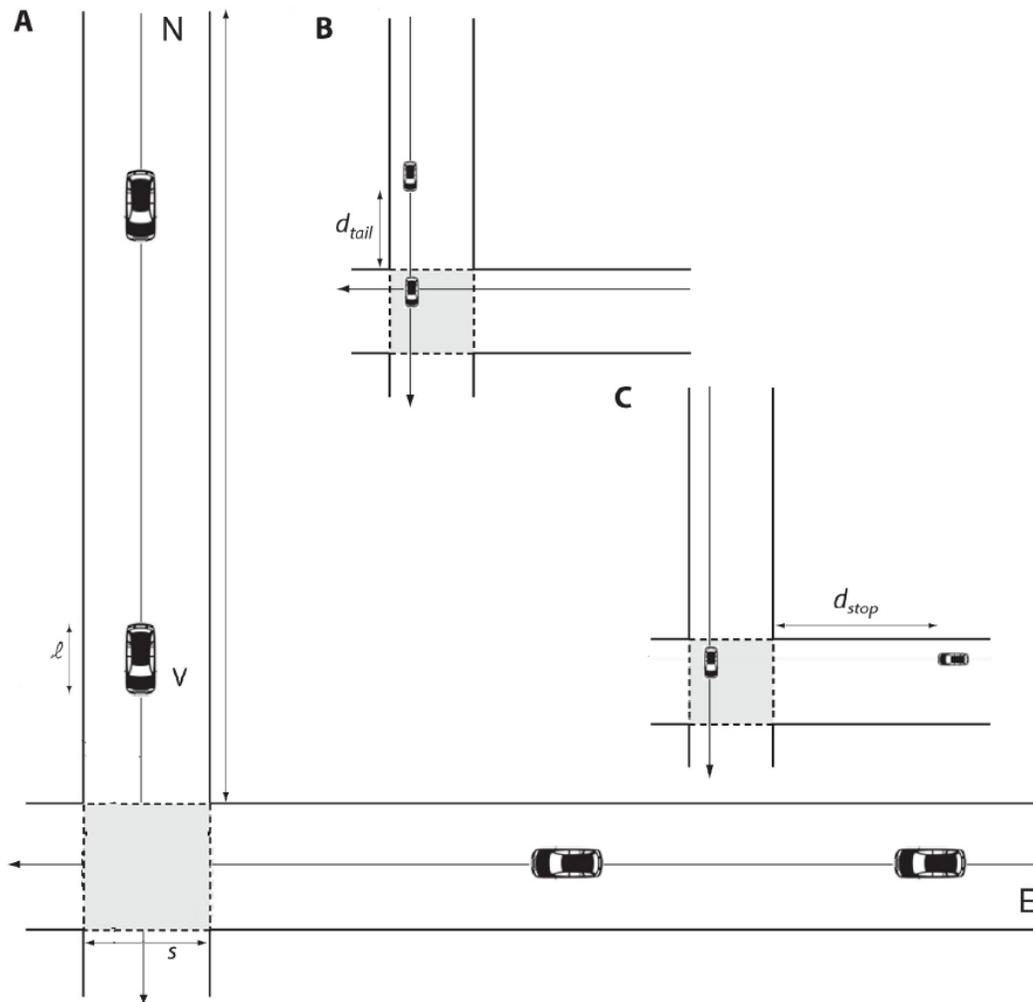


Figura 3.3: Estructura de la intersección. (A) El área de la intersección es cuadrada y de lado s . El tiempo de acceso a la intersección de un vehículo se define como el momento en que el frente del vehículo ingresa al área de la intersección. (B) Los criterios de seguridad del tránsito especifican que dos vehículos que acceden consecutivamente a la intersección y pertenecen al mismo flujo deben estar separados por la distancia d_{tail} . (C) Si los dos vehículos consecutivos pertenecen a diferentes flujos, entonces deben estar separados por la distancia d_{stop} , donde d_{stop} es mayor que d_{tail} .

El tiempo T_1 se calcula a partir de la distancia d_{tail} y es independiente de la velocidad v . La ecuación de la distancia d_{tail} es:

$$d_{tail} = v(t_{res} + \Delta) \quad (3.9)$$

$$T_1 = \left\lceil \frac{d_{tail}}{v} \right\rceil = \left\lceil \frac{v(t_{res} + \Delta)}{v} \right\rceil = \lceil t_{res} + \Delta \rceil, \quad (3.10)$$

donde t_{res} es un tiempo de reacción del vehículo autónomo V y Δ es un parámetro de tolerancia. En el trabajo de Tachet et al. [29] se sugiere ajustar $T_1 = t_{res} + \Delta = 1$ segundo.

El tiempo T_2 se calcula suponiendo que el vehículo autónomo V se aproxima a una distancia d_{stop} de la intersección y la parte trasera del vehículo autónomo V' abandona la intersección. La ecuación de la distancia d_{stop} es:

$$d_{stop} = v(t_{res}) + \frac{v^2}{2a_{brake}}, \quad (3.11)$$

donde $\frac{v^2}{2a_{brake}}$ es la distancia requerida por el vehículo V para detenerse antes de la intersección.

El tiempo de acceso T_2 a la intersección entre los vehículos se determina con la siguiente ecuación:

$$T_2 = T_2(v) = \left\lceil \frac{d_{stop}}{v} + \frac{l_s + s}{v} \right\rceil = \left\lceil t_{res} + \frac{v}{2a_{brake}} + \frac{l_s + s}{v} \right\rceil, \quad (3.12)$$

donde $\frac{l_s + s}{v}$ es el tiempo requerido por el vehículo V' para salir de la intersección. El tiempo $T_2(v)$ es una función continua en v , cuyo mínimo se logra en:

$$v = v^* = \sqrt{2a_{brake}(l_s + s)} \quad (3.13)$$

Las solicitudes de acceso a la intersección son representadas por el conjunto $(\{V, at_V, X_V\})_{V \in C}$, donde C es el conjunto de vehículos autónomos, $X_V \in \{N, E\}$ es el flujo norte o este y at_V es el tiempo de arribo mínimo del vehículo V a la intersección en condiciones de flujo libre. El conjunto $(\{V, at_V, X_V\})_{V \in C}$ tiene la propiedad de estar ordenado respecto a at_V . Una solución al problema de la administración de la intersección es un conjunto $(\{V, t_V\})_{V \in C}$, donde t_V es el tiempo de acceso a la intersección del vehículo autónomo V . El vehículo V' está adelante del vehículo V y su tiempo de arribo es at'_V . Si no hay vehículo adelante de V , entonces $at'_V = 0$. Los tiempos de acceso deben cumplir las restricciones de seguridad. Si $at'_V < at_V$ y $X'_V = X_V$, entonces $t'_V < t_V$.

La velocidad v_x de cada vehículo cuando entra a la intersección es calculada por la ecuación (3.14).

$$v_x = \min(v_{máx}, \left[\sqrt{v_{0x}^2 + 2a(x - x_0)} \right]) \quad (3.14)$$

donde v_{0x} es la velocidad actual del vehículo, x es la posición del comienzo de la intersección, x_0 es la posición actual del frente del vehículo y a es la aceleración; ver Figura 3.4.

El tiempo de arribo at_V del vehículo V a la intersección está determinado por la ecuación (3.15).

$$at_V = \max\left(\left[\frac{x - x_0}{\left(\frac{v_{0x} + v_x}{2}\right)} \right], at'_V + T_1\right), \quad (3.15)$$

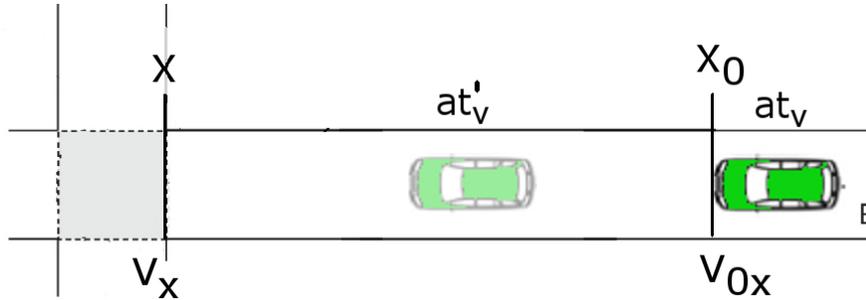


Figura 3.4: La posición del comienzo de la intersección es x . La posición frontal, la velocidad inicial, la velocidad final en la posición x y el tiempo de arribo del vehículo V son x_0 , v_{0x} , v_x y at_V , respectivamente. El tiempo de arribo del vehículo V' a la intersección es at'_V .

La estrategia *BATCH* comunica a los vehículos autónomos sus respectivos tiempos de acceso a la intersección. La planificación de los tiempos de acceso a la intersección de los vehículos autónomos se realiza de acuerdo con sus intervalos de tiempo de arribo at_V .

Inicialmente, la estrategia *BATCH* planifica los tiempos de acceso a la intersección de los vehículos autónomos con base en el criterio *primero en arribar primero en ser servido*. En el instante que un vehículo autónomo V experimenta un retraso $\Delta = (t_V - at_V)$ mayor que cero, es decir, el tiempo de acceso asignado t_V es mayor que su tiempo de arribo at_V , los vehículos autónomos que arriban a la intersección durante el intervalo de tiempo Δ se procesan como un lote (*BATCH*). Después de dicho lote, el acceso de los vehículos autónomos a la intersección nuevamente se asigna por orden de llegada, hasta que otro vehículo autónomo se retrase y el proceso anteriormente mencionado se repite. El número máximo de vehículos asignados a un lote es de N . Si se alcanza N durante un solo lote, entonces el vehículo $(N + 1)$ que está necesariamente retrasado servirá como pivote para la próxima planificación (ver Algoritmo 1).

Algoritmo 1 BATCH

- 1: V_0 es el primer vehículo procesado
 - 2: $t_l = at_{V_0}, d_l = X_{V_0}$
 - 3: $n_p = 1$ ▷ Número de vehículos procesados
 - 4: **mientras** $n_p < |C|$ **hacer**
 - 5: \bar{V} es el vehículo $(n_p + 1) \in C$
 - 6: **si** $X_{\bar{V}} = d_l$ **entonces**
 - 7: $T_{sep} = T_1$
 - 8: **si no**
 - 9: $T_{sep} = T_2$
 - 10: **fin si**
 - 11: $t_l = \max(t_l + T_{sep}, at_{\bar{V}})$
 - 12: $B = \{V \in C, at_{\bar{V}} \leq at_V \leq t_l = at_{\bar{V}} + \Delta\}$, con $|B| \leq N$
 - 13: BATCHPROCESSING(t_l)
 - 14: V_l es el último vehículo procesado en B
 - 15: $t_l = t_{V_l}, d_l = X_{V_l}, n_p = n_p + |B|$
 - 16: **fin mientras**
-

Algoritmo 2 Batch Processing

```
1:  $t_f = t_l$ 
2:  $X$  es el flujo de primer vehículo en  $B$ 
3: para  $V \in B \wedge X$  hacer                                ▷ Para todos los vehículos  $V$  en  $B$  con flujo  $X$ 
4:    $t_V = t_f$ 
5:   si  $V$  es el ultimo vehículo del flujo  $X$  entonces
6:     si Hay un vehículo con flujo  $Y \neq X$  entonces
7:        $t_f = t_f + T_2$ 
8:     fin si
9:   si no
10:     $t_f = t_f + T_1$ 
11:  fin si
12: fin para
13: para  $V \in B \wedge Y \neq X$  hacer                            ▷ Para todos los vehículos  $V$  en  $B$  con flujo  $Y$  distinto de  $X$ 
14:    $t_V = t_f$ 
15:   si  $V$  no es el ultimo vehículo del flujo  $Y$  entonces
16:      $t_f = t_f + T_1$ 
17:   fin si
18: fin para
```

Capítulo 4

Modificaciones al modelo LAI

En este capítulo se presentan las modificaciones que se realizaron al modelo LAI para que soporte una ciudad tipo Manhattan. El modelo LAI solo simula vehículos moviéndose en una calle. En la ciudad de Manhattan hay intersecciones, semáforos, varias calles y sentidos. Por esta razón, las modificaciones al modelo LAI permiten que el vehículo i determine si debe detenerse, reducir la velocidad o continuar moviéndose al aproximarse una intersección j .

4.1. Vehículos en las intersecciones

En el modelo LAI el ajuste de la velocidad del vehículo i depende de la posición y velocidad del vehículo $i + 1$ que hay adelante. El modelo LAI se modificó para que el vehículo i verifique si hay un vehículo fluyendo perpendicularmente en alguna de las intersecciones que hay adelante. La verificación se realiza para identificar cuál es el vehículo $i + 1$ y se aplique la velocidad adecuada para evitar colisionar (ver Figura 4.1).

Las proposiciones lógicas 4.1 y 4.2 verifican si hay un vehículo en la intersección j cuando los vehículos fluyen de derecha a izquierda y de izquierda a derecha, respectivamente.

$$(x_i(t) < I_j < x_{i+1}(t)) \wedge (s_{I_j}(t) = 1) \quad (4.1)$$

$$(x_{i+1}(t) < I_j < x_i(t)) \wedge (s_{I_j}(t) = 1) \quad (4.2)$$

donde I_j representa la posición de la intersección j . El estado de la celda en la j -ésima intersección es denotada con $s_{I_j}(t)$. El valor de $s_{I_j}(t)$ es 1 si hay un vehículo y es 0 en caso contrario.



Figura 4.1: Respecto al vehículo que está a la izquierda, el vehículo que hay adelante es el vehículo que ocupa la intersección.

Si dichas condiciones se cumplen, entonces la posición del vehículo precedente x_{i+1} al vehículo x_i será la posición de la intersección. Es decir, $x_{i+1}(t) = I_j$ y su velocidad es asignada a $v_{i+1}(t) = 0$ porque el vehículo de adelante se mueve de manera perpendicular.

4.2. Semáforos en las intersecciones

Los vehículos deciden si deben detenerse o continuar en movimiento dependiendo del estado de los semáforos y de la ocupación de la intersección. El estado del semáforo $\sigma_j(t)$ es *verde* en luz verde y *roja* en luz roja. En la misma manera que un vehículo i ajusta su velocidad con respecto al vehículo de adelante $i + 1$, el vehículo i ajusta su velocidad en dependencia de las distancias de seguridad d_{a_j} , d_{m_j} y d_{d_j} con respecto a la posición y el estado del semáforo j , definidas en (4.3), (4.4) y (4.5).

$$d_{a_j} = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)+\Delta v) \text{ div } M} [(v_i(t) + \Delta v) - kM] \right) \quad (4.3)$$

$$d_{m_j} = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)) \text{ div } M} [v_i(t) - kM] \right) \quad (4.4)$$

$$d_{d_j} = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)-\Delta v) \text{ div } M} [(v_i(t) - \Delta v) - kM] \right) \quad (4.5)$$

A diferencia de las ecuaciones con respecto al vehículo de adelante $i + 1$ (ver 3.1, 3.2 y 3.3), las ecuaciones 4.3, 4.4 y 4.5, no incluyen la segunda sumatoria porque el valor de la velocidad del semáforo j siempre es cero.

La ecuación (4.3) representa la distancia d_{a_j} que determina si el vehículo i puede aumentar o disminuir su velocidad en Δv . La ecuación (4.4) representa la distancia d_{m_j} que determina si el vehículo i puede o tiene que disminuir su velocidad en Δv . La ecuación (4.5) representa la distancia d_{d_j} que determina si el vehículo i tiene que disminuir su velocidad en Δv o en M .

La distancia desde el vehículo i al semáforo j se define como $d_{ij}(t) = x_j(t) - x_i(t) - l_s$; donde l_s indica la longitud del vehículo. Las reglas 3.5, 3.6, 3.7 y 3.8 modifican la velocidad v_i dependiendo de la distancia entre el vehículo i y el vehículo $i + 1$. De la mismas manera, las reglas 4.6, 4.7, 4.8 y 4.9 modifican la velocidad v_{ij} dependiendo de la distancia entre el vehículo i y el semáforo j .

- **Acelerar**, si $d_{ij}(t) \geq d_{a_j}$, la velocidad v_{ij} se incrementa aleatoriamente por una probabilidad R_a .

$$v_{ij}(t+1) = \begin{cases} \min(v_i(t) + \Delta v, v_{max}), & \text{si randf() } \leq R_a \\ v_i(t), & \text{en caso contrario} \end{cases} \quad (4.6)$$

- **Desacelerar aleatoriamente**, si $d_{a_j} > d_{ij}(t) \geq d_{m_j}$, la velocidad v_{ij} se reduce por una probabilidad R_s .

$$v_{ij}(t+1) = \begin{cases} \max(v_i(t) - \Delta v, 0), & \text{si randf() } \leq R_s \\ v_i(t), & \text{en caso contrario} \end{cases} \quad (4.7)$$

- **Frenar**, si $d_{m_j} > d_{ij}(t) \geq d_{d_j}$ y $v_i(t) > 0$, la velocidad v_{ij} se reduce.

$$v_{ij}(t+1) = \max(v_i(t) - \Delta v, 0) \quad (4.8)$$

- **Frenar urgentemente**, si $v_i(t) > 0$ y $d_{ij}(t) < d_{d_j}$, si la velocidad del vehículo i no es inferior a 0, entonces la velocidad v_{ij} se reduce en M .

$$v_{ij}(t+1) = \max(v_i(t) - M, 0) \quad (4.9)$$

Además, si la distancia entre un vehículo i y el semáforo j es suficiente para que el vehículo frene antes de la posición del semáforo j , entonces el vehículo frenará. En caso contrario, el vehículo i continuará su viaje. Por esta razón, es necesario calcular la distancia d_{s_j} que representa la distancia necesaria en dependencia de la velocidad del vehículo i en la que es posible detenerse antes del semáforo j (ver Figura 4.2), definida en la ecuación (4.10).

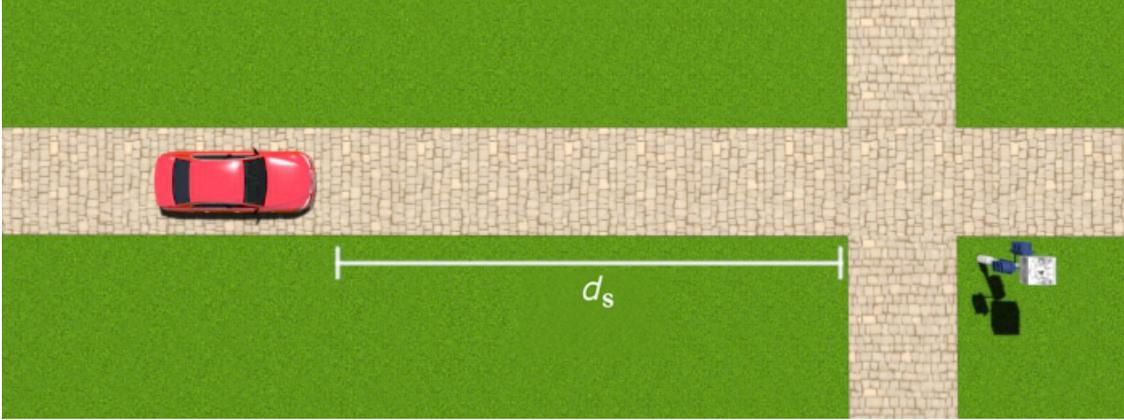


Figura 4.2: La distancia d_{s_j} representa la distancia límite en la que el vehículo puede frenar dependiendo de su velocidad.

$$d_{s_j} = \sum_{k=0}^{(v_i(t)-M) \text{ div } M} [(v_i(t) - M) - kM] \quad (4.10)$$

Por un lado, si la distancia del vehículo i al semáforo j es menor que d_{s_j} , entonces el vehículo i mantiene su velocidad porque no tiene la capacidad de frenar antes del semáforo j en luz *roja*, definida en la ecuación (4.10). Es decir, el vehículo i ignorará la luz roja y continuará su viaje. Por otro lado, si la distancia del vehículo i al semáforo j es mayor que d_{s_j} , entonces el vehículo i frenará antes del semáforo j .

- **Continuar viaje**, si $d_{ij}(t) < d_{s_j}$, la velocidad v_{ij} se mantienen en v_i .

$$v_{ij}(t+1) = v_i(t) \quad (4.11)$$

El vehículo i decide si aplicar la velocidad respecto al vehículo $i+1$ o al semáforo j mediante el mínimo de las dos velocidades como se expresa en la ecuación (4.12).

$$v_i(t+1) = \min(v_{ii}(t+1), v_{ij}(t+1)) \quad (4.12)$$

donde la velocidad del vehículo i respecto al vehículo $i + 1$ se denota como $v_{ii}(t + 1)$. La posición $x_i(t + 1)$ denota la posición del vehículo i en el paso de tiempo $t + 1$ y se calcula como $x_i(t + 1) = x_i(t) + v_i(t + 1)$.

4.2.1. Algoritmo LAI modificado para una ciudad tipo Manhattan

El Algoritmo 3 soporta una ciudad tipo Manhattan que incluye semáforos (los detalles del modelo LAI se muestran en la sección 3.1.1). En el Algoritmo 3, cada vehículo i aplica las velocidades correspondientes respecto al vehículo de adelante $i + 1$ y al semáforo j .

El Algoritmo 3 asigna el valor de los parámetros $R_d = 1.0$, $R_0 = 0.8$ y $R_s = 0.01$ (línea 1). Dichos valores de los parámetros simulan el comportamiento de vehículos operados por humanos en el modelo LAI y fueron propuestos por Lárraga y Álvarez-Icaza [20].

Si hay un vehículo detectado en la intersección j , entonces el vehículo de adelante $i + 1$ adquiere la posición de la intersección $x_{i+1}(t) = I_j$. Además, la velocidad $v_{i+1}(t) = 0$ porque el vehículo se desplaza perpendicularmente (líneas 2–5). Las distancias de seguridad son calculadas con respecto al vehículo de adelante $i + 1$ (líneas 6–8).

El parámetro R_a se calcula para asignar una mayor probabilidad de acelerar a los vehículos que se desplazan a mayores velocidades que los demás (línea 9). La distancia $d_i(t)$ representa la separación que hay entre el vehículo i y el vehículo de adelante $i + 1$. La función **CALCULARVELOCIDADVEHÍCULO** determina qué regla debe usarse para calcular una velocidad segura $v_{ii}(t + 1)$ en dependencia de la distancia $d_i(t)$ y la velocidad $v_i(t)$ (línea 11, ver Algoritmo 4).

Si la luz del semáforo j es roja (línea 13), entonces las distancias de seguridad son calculadas con respecto al semáforo j (líneas 14–17). La distancia $d_j(t)$ representa la separación que hay entre el vehículo i y el semáforo j . La función **CALCULARVELOCIDADSEMÁFORO** determina qué regla debe usarse para calcular una velocidad segura $v_{ij}(t + 1)$ en dependencia de la distancia $d_j(t)$ y la velocidad $v_i(t)$ (línea 19, ver Algoritmo 4).

El vehículo i decide si aplicar la velocidad respecto al vehículo $i + 1$ o al semáforo j mediante el mínimo de las dos velocidades $v_{ii}(t + 1)$ y $v_{ij}(t + 1)$ (línea 21). La posición $x_i(t + 1)$ del vehículo i depende del sentido en que se dirija (este u oeste).

Algoritmo 3 LAI en una ciudad del tipo Manhattan

-
- 1: $\Delta v = 2.5, M = 2, l_s = 2, R_d = 1.0, R_0 = 0.8, v_s = 3.0, R_s = 0.01$
 - 2: **si** $(x_i(t) < I_j < x_{i+1}(t) \wedge s_{I_j} = 1) \vee (x_{i+1}(t) < I_j < x_i(t) \wedge s_{I_j} = 1)$ **entonces**
 - 3: $x_{i+1}(t) = I_j$
 - 4: $v_{i+1}(t) = 0$
 - 5: **fin si**
 - 6: $d_a = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)+\Delta v) \text{ div } M} [(v_i(t) + \Delta v) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \text{ div } M} [(v_{i+1}(t) - M) - kM] \right)$
 - 7: $d_m = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)) \text{ div } M} [v_i(t) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \text{ div } M} [(v_{i+1}(t) - M) - kM] \right)$
 - 8: $d_d = \text{máx} \left(0, \sum_{k=0}^{(v_i(t)-\Delta v) \text{ div } M} [(v_i(t) - \Delta v) - kM] - \sum_{k=0}^{(v_{i+1}(t)-M) \text{ div } M} [(v_{i+1}(t) - M) - kM] \right)$
 - 9: $R_a = \text{mín}(R_d, R_0 + v_i(t) \cdot (R_d - R_0/v_s))$
 - 10: $d_i(t) = x_{i+1}(t) - x_i(t) - l_s$
 - 11: $v_{ii}(t+1) = \text{CALCULARVELOCIDADVEHÍCULO}(v_i(t), d_i(t))$ ▷ Respecto al vehículo
 - 12: $v_{ij}(t+1) = 0$
 - 13: **si** $\sigma_j(t) = \text{roja}$ **entonces** ▷ Sólo si el semáforo está en rojo
 - 14: $d_{a_j} = \sum_{k=0}^{(v_i(t)+\Delta v) \text{ div } M} [(v_i(t) + \Delta v) - kM]$
 - 15: $d_{m_j} = \sum_{k=0}^{(v_i(t)) \text{ div } M} [v_i(t) - kM]$
 - 16: $d_{d_j} = \sum_{k=0}^{(v_i(t)-\Delta v) \text{ div } M} [(v_i(t) - \Delta v) - kM]$
 - 17: $d_{s_j} = \sum_{k=0}^{(v_i(t)-M) \text{ div } M} [(v_i(t) - M) - kM]$
 - 18: $d_j(t) = x_j^I(t) - x_i(t) - l_s$
 - 19: $v_{ij}(t+1) = \text{CALCULARVELOCIDADSEMÁFORO}(v_i(t), d_j(t))$ ▷ Respecto al semáforo
 - 20: **fin si**
 - 21: $v_i(t+1) = \text{mín}(v_{ii}(t+1), v_{s_j}(t+1))$
 - 22: **si** $(x_i(t) < x_{i+1}(t))$ **entonces** ▷ Los vehículos pueden fluir en ambas direcciones
 - 23: $x_i(t+1) = x_i(t) + v_i(t+1)$
 - 24: **si no**
 - 25: $x_i(t+1) = x_i(t) - v_i(t+1)$
 - 26: **fin si**
-

Algoritmo 4 Velocidades respecto a vehículos de adelante y semáforos en luz roja

```

1: función CALCULARVELOCIDADVEHÍCULO( $v_i(t), d_i(t)$ )
2:   si  $d_i(t) \geq d_a$  entonces
3:      $v_i(t+1) = \begin{cases} \text{mín}(v_i(t) + \Delta v, v_{max}) & \text{si } \text{randf}() \leq R_a \\ v_i(t) & \text{en caso contrario} \end{cases}$ 
4:   si no si  $d_a > d_i(t) \geq d_m$  entonces
5:      $v_i(t+1) = \begin{cases} \text{máx}(v_i(t) - \Delta v, 0) & \text{si } \text{randf}() \leq R_s \\ v_i(t) & \text{en caso contrario} \end{cases}$ 
6:   si no si  $d_m > d_i(t) \geq d_d \wedge v_i(t) > 0$  entonces
7:      $v_i(t+1) = \text{máx}(v_i(t) - \Delta v, 0)$ 
8:   si no si  $d_d > d_i(t) \wedge v_i(t) > 0$  entonces
9:      $v_i(t+1) = \text{máx}(v_i(t) - M, 0)$ 
10:  fin si
11:  devolver  $v_i(t+1)$ 
12: fin función
13: función CALCULARVELOCIDADSEMÁFORO( $v_i(t), d_j(t)$ )
14:   si  $d_j(t) \geq d_{a_j}$  entonces
15:      $v_i(t+1) = \begin{cases} \text{mín}(v_i(t) + \Delta v, v_{max}) & \text{si } \text{randf}() \leq R_{a_i} \\ v_i(t) & \text{en caso contrario} \end{cases}$ 
16:   si no si  $d_{a_j} > d_j(t) \geq d_{m_j}$  entonces
17:      $v_i(t+1) = \begin{cases} \text{máx}(v_i(t) - \Delta v, 0) & \text{si } \text{randf}() \leq R_s \\ v_i(t) & \text{en caso contrario} \end{cases}$ 
18:   si no si  $d_{m_j} > d_j(t) \geq d_{d_j} \wedge v_i(t) > 0$  entonces
19:      $v_i(t+1) = \text{máx}(v_i(t) - \Delta v, 0)$ 
20:   si no si  $d_{d_j} > d_j(t) \geq d_{s_j} \wedge v_i(t) > 0$  entonces
21:      $v_i(t+1) = \text{máx}(v_i(t) - M, 0)$ 
22:   si no si  $d_j(t) < d_{s_j}$  entonces
23:      $v_i(t+1) = v_i(t)$ 
24:   fin si
25:   devolver  $v_i(t+1)$ 
26: fin función

```

Capítulo 5

Sistema de detección

En este capítulo describimos un *sistema de detección* basado en sensores simples que determina las posiciones y velocidades de los vehículos.

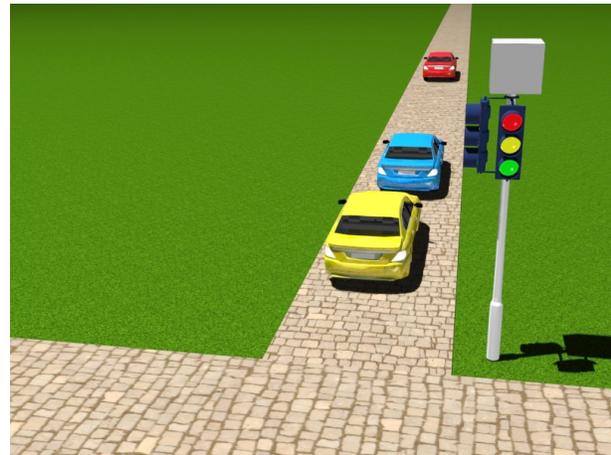
Los principales *semáforos inteligentes* propuestos utilizan sensores situados en la intersección para contar vehículos que se aproximan a una determinada distancia [1, 2, 3]. Sin embargo, las principales desventajas que impiden la implementación de dichos semáforos están relacionadas con el costo y en algunos casos con la privacidad.

El *semáforo auto-organizante reactivo* (SAR) tiene un *sistema de detección reactivo*. El *sistema de detección reactivo* detecta las posiciones de k vehículos en una zona de detección de aproximadamente 70m. Por esta razón, requiere de múltiples y elaborados sensores para detectar vehículos aproximándose y alejándose de la intersección. Dichos requerimientos limitan la implementación del *semáforo auto-organizante* en las ciudades debido a su costo. Además, no está diseñado para soportar fallas en los sensores.

Como una alternativa a los sensores requeridos por el *sistema de detección reactivo*, nosotros proponemos un *sistema de detección deliberativo* que utiliza sensores simples. El *sistema de detección deliberativo* detecta la posición y la velocidad de un vehículo en una zona de detección de 5m. Las posiciones y velocidades de los vehículos que se aproximan o alejan de la intersección son inferidas por un *entorno virtual* que representa el estado del tránsito vehicular, que depende de la historia percibida.



(a) Detección de los vehículos que arriban a la intersección



(b) Detección de los vehículos que dejan la intersección

Figura 5.1: Los principales inconvenientes para la implementación de semáforos inteligentes están relacionados a su costo, y en algunos casos a que violan la privacidad de los peatones porque son grabados.

5.1. Componentes del sistema de detección deliberativo

En el sistema de detección deliberativo se coloca al principio de cada cuadra una computadora y un sensor de presencia que en conjunto denominamos *sensor*. De la misma manera, una computadora y un semáforo son colocados en cada intersección que en conjunto denominamos *semáforo*.

El sensor detecta solo un vehículo en cada paso de tiempo (ver Figura 5.2). En contraste con el SAR, que detecta varios vehículos que se aproximan a una intersección en cada paso de tiempo. El sensor detecta vehículos al principio de cada cuadra y construye un *entorno virtual* que simula el movimiento de los vehículos para pronosticar qué vehículos llegan y salen de la intersección *aguas abajo*. Los vehículos simulados en el entorno virtual por el sensor se denominan *vehículos virtuales*.

El semáforo recibe información a través de una red de datos desde los sensores de las cuadras vecinas y ejecuta el *sistema de control* propuesto por Gershenson (seis reglas) para controlar los cambios de luz (ver sección 3.2.1).

Para cada cuadra, el sistema se compone de un sensor colocado antes de la intersección aguas abajo que envía información al semáforo aguas abajo y al semáforo aguas arriba (ver Figura 5.3).

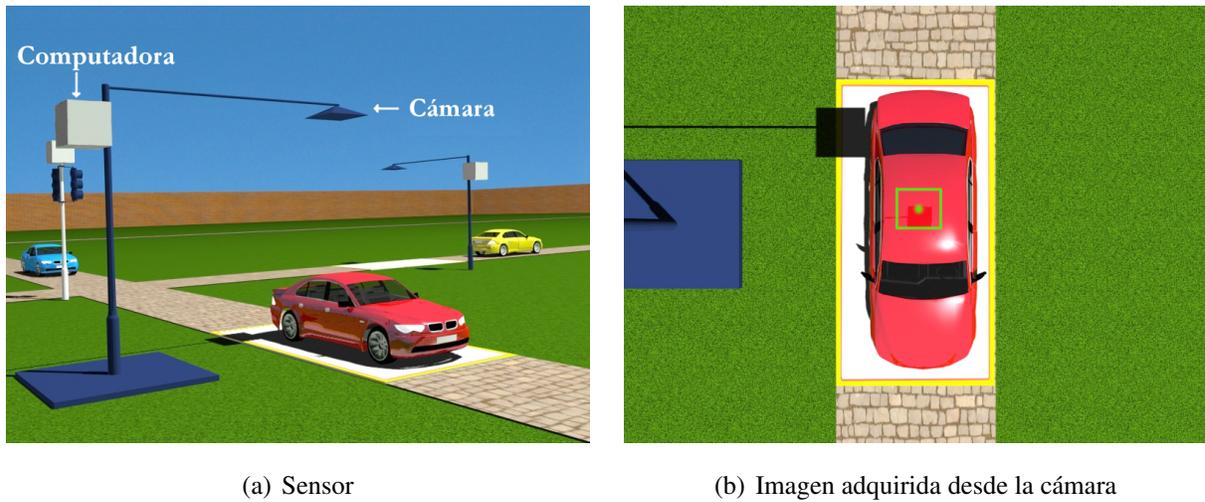


Figura 5.2: El sensor mostrado en la Figura (a) está compuesto por una computadora y un sensor que puede ser una cámara. El ejemplo ilustra una posible implementación del sensor. No obstante, el sensor puede tratarse de un bucle inductivo o un lidar. La Figura (b) muestra la imagen obtenida por el sensor, en la que únicamente se reconoce un vehículo en la zona de detección.

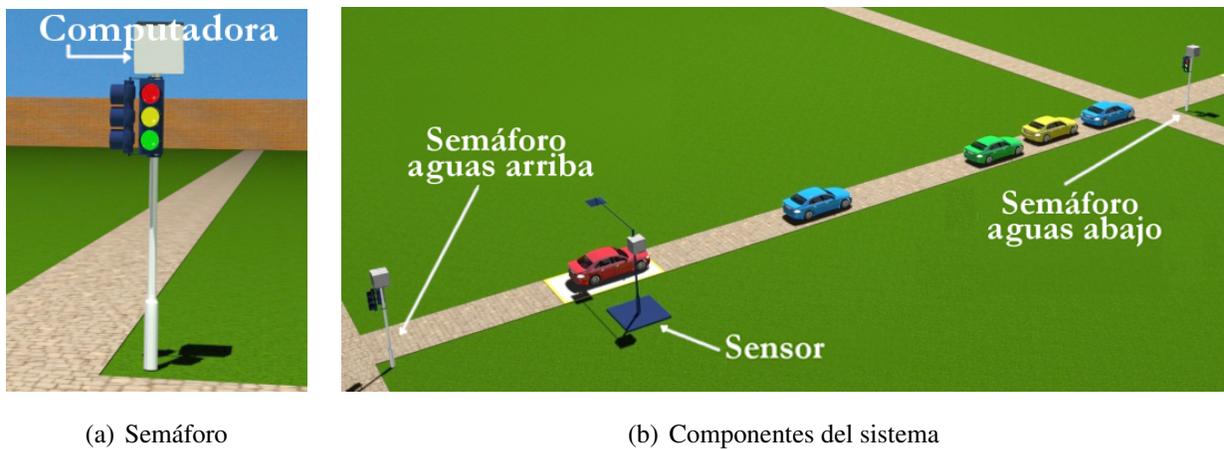


Figura 5.3: El sensor envía información de las posiciones de los vehículos virtuales predichos al semáforo aguas abajo. El sensor también envía un mensaje al semáforo aguas arriba para informar si un vehículo está detenido en la zona de detección. Con la información de los sensores, cada semáforo aplica un determinado sistema de control (por ejemplo, las seis reglas del semáforo auto-organizante reactivo).

Los vehículos virtuales que hay en o después de la posición del sensor y antes de la intersección forman parte del conjunto *vehículos virtuales recibidos*. Los vehículos virtuales que hay en o después de la intersección y antes de la posición del semáforo aguas abajo forman parte del conjunto *vehículos virtuales enviados* (ver Figura 5.4).

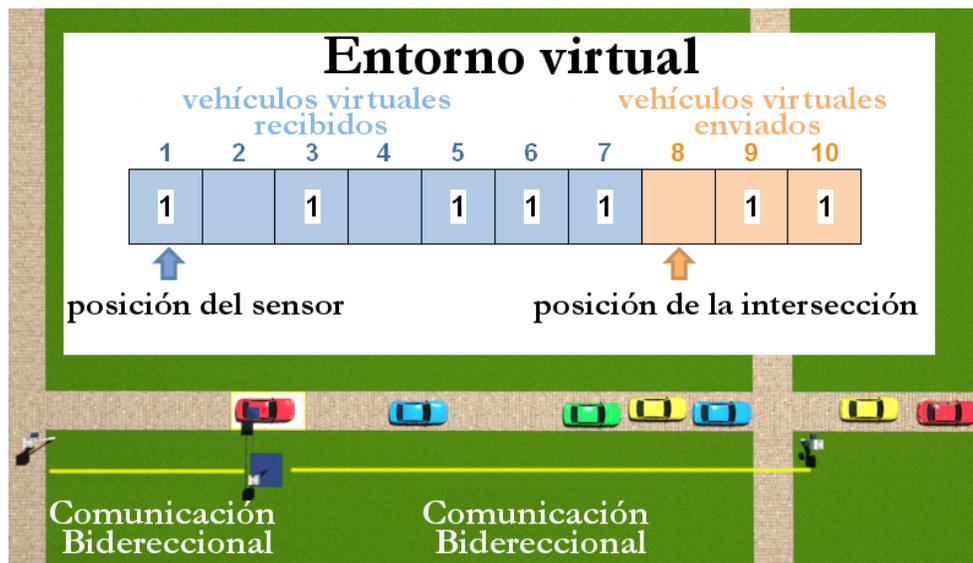


Figura 5.4: Cada sensor utiliza una simulación local para pronosticar el movimiento de los vehículos entre el sensor y el semáforo aguas abajo.

Este enfoque comparado con el sistema de detección reactivo que utiliza sensores elaborados tiene las siguientes ventajas:

- El costo de la infraestructura podría ser menor que la utilizada en el sistema de detección reactivo porque solo debe detectarse un vehículo en una zona de detección de 5m. Por ejemplo, el sensor puede ser integrada por un Raspberry Pi y una cámara de bajo costo.
- Al mantener un modelo interno (entorno virtual) del mundo real es posible que el sistema pueda pronosticar o deducir información no observable y recuperarse de errores de detección por parte de los sensores.
- La privacidad es respetada porque el sensor está dirigido sólo a una pequeña zona de detección; por lo tanto, los usuarios (peatones o conductores) no pueden ser identificados.

Sin embargo, con este enfoque se requiere una red de datos para el intercambio de información. En el caso de sensores elaborados no se requiere una red de datos, aunque es deseable para la supervisión del sistema.

Denominamos *semáforo auto-organizante deliberativo* (SAD) al semáforo que utiliza el *sistema de detección deliberativo* y el *sistema de control* propuesto por Gershenson.

5.2. Algoritmo de detección deliberativo

El algoritmo que ejecuta el *sensor* es responsable de simular el entorno virtual tanto con la información detectada como con los mensajes recibidos por el *semáforo aguas abajo*. El sensor envía la información al semáforo aguas abajo para ejecutar un determinado sistema de control de semáforos. Además, el sensor envía información al *semáforo aguas arriba* con el fin de compartir indirectamente esta información con el *sensor en la cuadra aguas arriba*. Las variables utilizadas en el algoritmo que ejecuta el sensor se muestran en la Tabla 5.1.

Name	Descripción
luz_{estado}	Si el valor es asignado a verdadero, el semáforo está en verde. Por otro lado, la luz es roja.
$enviados$	Número de vehículos virtuales enviados a la cuadra aguas abajo.
$recibidos$	Número de vehículos recibidos.
$detenido$	Si el valor es asignado a verdadero, un vehículo está detenido en la zona de detección.
$recibidos_{abajo}$	Número de vehículos recibidos en la cuadra aguas abajo.
$detenido_{abajo}$	Si el valor es asignado a verdadero, un vehículo está detenido en la zona de detección de la cuadra aguas abajo.
ϵ	Diferencia absoluta entre el número de vehículos virtuales enviados y número de vehículos recibidos en la cuadra aguas abajo.

Tabla 5.1: Variables utilizadas por el algoritmo de detección deliberativo.

Al inicio, el sensor no tiene información sobre la densidad de los vehículos que hay en su cuadra. Por esta razón, el sensor asume el caso medio y llena a la mitad el espacio reservado para el conjunto de *vehículos virtuales recibidos* en el entorno virtual. Además, las variables *recibidos* y *enviados* son asignadas a cero. Posteriormente, en cada paso de tiempo se ejecuta el Algoritmo 5.

El Algoritmo 5 asigna el valor de las variables *detenido* y *detenido_{abajo}* a falso (líneas 1–2). Si hay un vehículo detectado por el sensor, entonces la función **HayVehículo** retorna verdadero cuando

tal condición es cumplida y falso en caso contrario (línea 3). Si un vehículo es detectado, entonces el sensor adquiere la velocidad del vehículo con la función **AdquirirVelocidadVehículo** (línea 5).

Si hay un *vehículo virtual* en la posición de la zona de detección del entorno virtual, entonces la función **HayVehículoVirtual** retorna verdadero cuando tal condición es cumplida y falso en caso contrario (línea 6).

Si un vehículo es detectado y un vehículo virtual no está registrado en la posición correspondiente del entorno virtual, la función **CrearVehículoVirtual** crea un vehículo virtual en esa posición (línea 8). La verificación se realiza para evitar remover un vehículo virtual y volver a crearlo en la misma posición.

La función **Detenido** verifica si hay un vehículo detenido en la zona de detección. Si tal condición es verdadera, retorna verdadero y falso en caso contrario (línea 10). Si el vehículo está detenido, la variable *detenido* es asignada a verdadero para indicar que hay un vehículo detenido en la zona de detección, al principio de la cuadra. En caso contrario, si el vehículo está en movimiento el valor de la variable *recibidos* es aumentada en uno para indicar que se recibió un vehículo en la zona de detección (línea 14).

El sensor por medio de la función **RecibirMensajeSemáforoAguasAbajo** recibe un paquete por parte del semáforo aguas abajo. El paquete contiene información acerca de: 1) el estado de la luz del semáforo; 2) si un vehículo está detenido; y 3) el número de vehículos recibidos en la cuadra aguas abajo. Posteriormente, la información se asigna a las variables *luz_estado*, *detenido_abajo* y *recibidos_abajo*, respectivamente (líneas 18–20).

La función **SimularEntornoVirtual** simula en el entorno virtual el movimiento de los vehículos (línea 20).

- El movimiento de cada vehículo virtual del conjunto vehículos virtuales recibidos depende de: 1) el valor de la variable *luz_estado* (roja o verde); 2) la posición y velocidad del vehículo virtual; y 3) las posiciones y velocidades de los vehículos virtuales vecinos del vehículo virtual. Si un vehículo virtual recibido cruza una luz verde y se encuentra en o después de la intersección se volverá parte del conjunto de vehículos virtuales enviados.
- El movimiento de cada vehículo virtual del conjunto de vehículos virtuales enviados depende de: 1) el estado de la variable *detenido_abajo* (verdadero o falso); 2) la posición y velocidad del *vehículo virtual*; y 3) las posiciones y velocidades de los *vehículos virtuales* vecinos del *vehículo virtual*. El estado de la variable *detenido_abajo* opera de manera similar al estado de un

semáforo, si $detenido_{abajo}$ es verdadera y hay vehículos virtuales en el conjunto de vehículos virtuales enviados, entonces los vehículos virtuales no puede avanzar. Si un vehículo virtual alcanza la posición del sensor aguas abajo, entonces el vehículo virtual es eliminado del conjunto de vehículos virtuales enviados y la función **SimularEntornoVirtual** retorna un valor en el paquete *resultados* para indicar que un vehículo virtual arribó (línea 21).

Si la simulación registra un vehículo virtual que llega al sensor aguas abajo en el conjunto *vehículos virtuales enviados*, el sensor aumenta el valor de la variable *enviados* en uno (líneas 22–24).

Si la simulación registra un *vehículo virtual detenido* en la zona de detección y la variable *detenido* tiene el valor `falso`, entonces el valor de la variable *detenido* se asigna a `verdadero` porque el sensor escoge la posibilidad de que ocurrió un error de detección (líneas 25–27).

Si la simulación registra un vehículo virtual detenido en el conjunto de vehículos virtuales enviados y la variable $detenido_{abajo}$ tiene el valor `falso`, entonces el valor de la variable $detenido_{abajo}$ se asigna a `verdadero` porque el sensor escoge la posibilidad de que ocurrió un error de detección en el sensor aguas abajo con la finalidad de evitar congestionamientos en la intersección (líneas 28–30).

Es posible que la simulación local del entorno virtual no coincida con la simulación global de la ciudad de Manhattan principalmente por las siguientes razones:

- Los modelos vehiculares que introducen ruido estocástico en la dinámica vehicular pueden tener diferencias con el pronóstico de la simulación del entorno virtual. Por ejemplo, el modelo LAI introduce perturbaciones de tránsito al reducir la velocidad aleatoriamente.
- Al inicio de la simulación el sensor no tiene información por lo que asume el caso medio y llena la mitad del conjunto vehículos virtuales recibidos. Además, el sensor asume que no hay ningún vehículo virtual enviado. Por esta razón, el estado del tránsito no corresponde al estado del entorno virtual local del sensor y el algoritmo debe implementar un mecanismo para rectificar dicha situación.
- Cuando el sensor recibe una luz verde, asume que los vehículos comienzan a avanzar. No obstante, si un vehículo tiene una luz verde y hay un segundo vehículo en la dirección perpendicular obstruyendo la intersección, entonces el vehículo no avanzará y el pronóstico del entorno virtual estará desfasado.
- Los sensores pueden tener errores de detección y consecuentemente algunos vehículos no serán detectados.

La función **CambiarLuzVerde** determina si la luz del semáforo aguas abajo cambia a verde, cuando la condición es verdadera se calcula la diferencia absoluta entre el número de vehículos recibidos en la cuadra aguas abajo y el número de vehículos virtuales enviados. La diferencia se asigna a la variable ϵ (línea 33). Por un lado, si el número de vehículos recibidos en la variable *recibidos_{abajo}* es igual número de vehículos virtuales enviados de la variable *enviados*, entonces el valor de la variable ϵ es cero porque el número predicho entre los vehículos recibidos por el sensor aguas abajo y los vehículos virtuales enviados por el sensor es el mismo. Por otro lado, ocurren errores en el pronóstico, si el número de vehículos recibidos y vehículos virtuales enviados no es el mismo, es decir, el valor de ϵ no es cero. El valor se asigna a cero en la variable *enviados*.

La función **ContarVehículosEntornoVirtual** cuenta el número de vehículos virtuales recibidos que hay en la distancia d y r desde la posición de la intersección y los asigna a las variables n y m de la estructura de datos *resultados*. La estructura de datos *resultados* es el valor retornado por dicha función (línea 36). Posteriormente, es creado un paquete con el atributo *paquete₂.n* asignado a $n + \epsilon$, el atributo *paquete₂.m* asignado a m y el atributo *paquete₂.detenido* asignado a *detenido_{abajo}* (líneas 37–39). El paquete es enviado por medio de la función **EnviarMensajeSemáforoAguasAbajo** al semáforo aguas abajo (línea 40). El semáforo aguas abajo con los datos contenidos en el paquete aplicará un sistema de control (por ejemplo, el propuesto por Gershenson) para determinar los cambios de luz.

El número de vehículos virtuales recibidos en la distancia d y r es aumentado con ϵ para ajustar la diferencia entre el número de vehículos recibidos y el número de vehículos virtuales enviados (línea 37). El aumento en el número de vehículos recibidos mitiga dos situaciones desfavorables: 1) cuando el número de *vehículos virtuales* enviados es inferior al de recibidos, se infiere que hay vehículos restantes en la cuadra anterior; 2) cuando el número de *vehículos virtuales* enviados es mayor al de recibidos, se infiere que hay vehículos sobrantes en la cuadra anterior.

El sensor, por medio de la función **EnviarMensajeSemáforoAguasArriba**, envía un paquete con los valores de las variables *detenido* y *recibidos* al semáforo aguas arriba (líneas 40–42). El paquete contiene información acerca de si un vehículo se encuentra detenido y del número de vehículos recibidos en la cuadra del sensor.

Finalmente, el sensor por medio de la función **RecibirMensajeSemáforoAguasArriba** recibe un paquete del *semáforo aguas arriba* para informar que la luz del semáforo aguas arriba cambió a verde y asigna el valor de la variable *recibidos* a cero (líneas 44–47).

Algoritmo 5 Algoritmo de detección deliberativo

```

1: detenido = falso
2: detenido_abajo = falso
3: vehiculo_estado = HayVehículo()
4: si vehiculo_state = verdadero entonces
5:   v = AdquirirVelocidadVehículo()
6:   virtual_estado = HayVehículoVirtual()
7:   si virtual_estado = falso entonces
8:     CrearVehículoVirtual(v)
9:   fin si
10:  vehiculo_detenido = Detenido()
11:  si vehiculo_detenido = true entonces
12:    detenido = verdadero
13:  si no
14:    recibidos = recibidos + 1
15:  fin si
16: fin si
17: paquete1 = RecibirMensajeSemáforoAguasAbajo()
18: luz_estado = paquete1.luz_estado
19: detenido_abajo = paquete1.detenido_abajo
20: recibidos_abajo = paquete1.recibidos_abajo
21: resultados = SimularEntornoVirtual(luz_estado, detenido_abajo)
22: si resultados.enviados = verdadero entonces
23:   enviados = enviados + 1
24: fin si
25: si resultados.detenido_recibidos = verdadero  $\wedge$  detenido = falso entonces
26:   detenido = verdadero
27: fin si

```

```

28: si resultados.detenidoenviados = verdadero  $\wedge$  detenidoabajo = falso entonces
29:     detenidoabajo = verdadero
30: fin si
31: luzcambiar = CambiarLuzVerde(luzestado)
32: si luzcambiar = verdadero entonces
33:      $\epsilon = |\text{recibidos}_{\text{abajo}} - \text{enviados}|$ 
34:     enviados = 0
35: fin si
36: resultados = ContarVehículosEntornoVirtual()
37: paquete2.n = resultados.n +  $\epsilon$ 
38: paquete2.m = resultados.m
39: paquete2.detenido = detenidoabajo
40: EnviarMensajeSemáforoAguasAbajo(paquete2)
41: paquete3.stop = stop
42: paquete3.received = received
43: EnviarMensajeSemáforoAguasArriba(paquete3)
44: paquete4 = RecibirMensajeSemáforoAguasArriba()
45: si paquete4.luzcambiar = verdadero entonces
46:     recibidos = 0
47: fin si

```

Un sistema en la ciudad de Duisburg (Alemania) ha sido desarrollado con el propósito de pronosticar el estado del tráfico mediante la simulación en línea de vehículos virtuales con autómatas celulares [30, 31]. Al igual que nuestro planteamiento con los *semáforos auto-organizantes deliberativos*, en el *sistema Duisburg* es necesario colocar sensores (bucles inductivos) en las calles para detectar la presencia de los vehículos. Sin embargo, a diferencia de los semáforos auto-organizantes deliberativos, el sistema de Duisburg tiene una arquitectura centralizada porque una computadora recopila toda la información obtenida por los sensores y otra computadora ejecuta la simulación a nivel global. Nuestro planteamiento con los semáforos auto-organizantes deliberativos, por el contrario, es distribuido porque cada sensor recopila la información y simula el tráfico correspondiente sólo en su cuadra, es decir, a nivel local. Los semáforos auto-organizantes deliberativos tienen la ventaja de que si un sensor falla, no implica que el sistema entero colapse, lo que ocurre en el sistema de Duisburg cuando la

computadora que recopila la información de los sensores o la computadora que ejecuta la simulación falla.

Además, la simulación en línea del sistema de Duisburg tiene como objetivo planificar rutas según el estado del tránsito para enviarlas a los conductores. La simulación en cada sensor de los semáforos auto-organizantes deliberativos tiene como objetivo controlar los cambios de luz del semáforo.

5.3. Experimentos y resultados con la regla 184

Realizamos experimentos aplicando la regla 184 con el *sistema ola verde* (OV), el *semáforo auto-organizante reactivo* (SAR) y *deliberativo* (SAD), en una ciudad del tipo Manhattan. La longitud de todas las calles es de 1,700 celdas, es decir, 8,500 metros. Cada cuadra tiene una distancia de 16 celdas (sin la intersección). Así, hay 80 metros simulados entre las calles. Hay 330,000 celdas en la simulación, 10,000 intersecciones y 200 calles: 50 en cada dirección cardinal. Cabe mencionar que la isla de Manhattan tiene 3,360 intersecciones, por lo que nuestra *ciudad de Manhattan* es aproximadamente tres veces el tamaño de la isla Manhattan. Los vehículos no tienen permitido incorporarse a otra calle en las intersecciones, es decir, siempre se desplazan sobre la misma calle.

Los experimentos consistieron en ejecutar la simulación en 50 densidades distintas tanto para la OV, SAR y SAD. En cada densidad se promediaron 10 ejecuciones. En las simulaciones con las reglas de transición del ACE cada paso de tiempo equivale a $1/3$ de segundo. Cada ejecución consistió en lo siguiente: se simuló media hora (5,400 pasos de tiempo) para estabilizar el sistema donde los vehículos son colocados aleatoriamente en las calles de la ciudad. Posteriormente, se simuló otra media hora donde la velocidad se mide en cada paso de tiempo. Al final de la simulación, se promedian las velocidades de la segunda media hora para obtener la velocidad media v y el flujo promedio J .

La Tabla 5.2 muestra los valores de los parámetros utilizados por el SAR y el SAD. Dichos valores fueron propuestos por Gershenson [4].

El simulador con el que realizamos nuestros experimentos fue desarrollado en el lenguaje C. El lector puede acceder al código en la URL: <https://github.com/Zapotecat1/Traffic-Light>.

Los resultados se muestran en la Figura 5.5. En el sistema *ola verde* se encontraron las siguientes fases:

- La fase *intermitente* está entre $\rho > 0$ y $\rho \lesssim 0.22$. En esta fase algunos vehículos se detienen detrás de una luz roja. La transición de fase se produce en ≈ 0.22 . Donde hay un flujo máximo $J \approx 0.17$ (ver la Figura 5.5).

Parámetros	Valores	Escala
d	10 celdas	50m
r	5 celdas	25m
e	2 celdas	10m
u	10 pasos	3.33s
w	600 pasos	200s
n	40 veh · pasos	13.33 veh · s
m	2 vehículos	2 vehículos

Tabla 5.2: Parámetros usados por los sistemas auto-organizantes. El tamaño de la cuadra es de 16 celdas.

- La fase *congestionada* ocurre en densidades $\rho \gtrsim 0.22$, los vehículos que fluyen en las calles con la ola verde (hacia el sur y hacia el este) no pueden mantener la velocidad de la ola verde, de modo que los congestionamientos comienzan a moverse en la dirección opuesta del tránsito. Las colas de los grupos de vehículos que esperan detrás de una luz roja crecen en direcciones opuestas a la ola verde y bloquean las intersecciones. Consecuentemente, se originan los cuellos de botella (ver Figura 5.5).

Para el SAR y SAD se encontraron las siguientes fases:

- La fase *flujo libre* se presenta en densidades bajas en el SAR y SAD en $0 < \rho \lesssim 0.000006$, donde $v = 1$ y ningún vehículo tiene que detenerse.
- La fase *cuasi-flujo libre* se presenta en el SAR y SAD en $0.001 < \rho \lesssim 0.121$, donde pocos vehículos se detienen por poco tiempo. La mayoría de las intersecciones tienen sólo un grupo de vehículos solicitando una luz verde, por lo que el grupo es capaz de fluir sin tener que esperar.
- La fase *intermitente subutilizada* se presenta en el SAR y SAD entre $\rho \approx 0.12$ y $\rho \approx 0.22$, donde las intersecciones están ociosas una parte del tiempo, es decir, ningún vehículo las usa. La diferencia con la fase de *cuasi-flujo libre* es que en la fase *intermitente subutilizada* hay dos grupos de vehículos que solicitan una luz verde en la mayoría de los casos. Así, un grupo tiene que esperar hasta que el otro cruce.
- La fase *capacidad cuasi-completa intermitente* se presenta en el SAR y SAD entre $\rho \approx 0.22$ y $\rho \approx 0.4$. Hay siempre dos grupos de vehículos que solicitan una luz verde; sin embargo, hay

una cierta probabilidad que los grupos sean de tamaños diferentes. El grupo más pequeño genera menor demanda y conducirá a intersecciones ociosas.

- La fase *capacidad completa intermitente* se presenta en el SAR y SAD entre $\rho \approx 0.4$ y $\rho \approx 0.68$ con un flujo máximo $J \approx 0.257$. Esto implica que las intersecciones se están utilizando a plena capacidad. En la fase *capacidad completa intermitente* no hay recursos “perdidos” de espacio libre porque hay vehículos cruzando en todas las intersecciones.
- La fase *intermitente superutilizada* se presenta en el SAR y SAD entre $\rho \approx 0.68$ y $\rho \approx 0.86$, donde la densidad es tal que la regla 6 a veces obliga a ambas direcciones a detenerse, reduciendo así el flujo de las intersecciones. Esta fase es similar a la fase *intermitente subutilizada*, en el sentido de que las intersecciones no pueden utilizarse a su capacidad de flujo total. En el caso *intermitente subutilizada*, es porque no hay vehículos suficientes. En el caso *intermitente superutilizada*, es porque hay demasiados vehículos y el uso de las intersecciones debe esperar antes de que una calle pueda obtener la luz verde.
- La fase *cuasi-congestionada* se presenta en el SAR y SAD entre $\rho \approx 0.8$ y $\rho < 1.0$. La mayoría de los vehículos están detenidos, pero los espacios libres se mueven en la dirección opuesta de los vehículos entre los congestionamientos a una velocidad de una celda por paso de tiempo.
- La fase *congestionada* se presenta para altas densidades en el SAR y SAD en $\rho = 1.0$, es decir, $v = 0$. Los vehículos no pueden moverse porque la densidad es demasiado alta y las intersecciones son bloqueadas desde el inicio.

En conclusión, el SAR y SAD tienen un desempeño similar. Sin embargo, el SAD utiliza menos sensores.

5.3.1. Errores en los sensores

La función **HayVehículo** se ejecuta k veces para verificar si hay un vehículo en la zona de detección en el sistema de detección *reactivo*. La función **HayVehículo** se ejecuta una vez en el sistema de detección *deliberativo*. Para cada celda, el sistema de detección (reactivo y deliberativo) tiene asociada una probabilidad P que representa la precisión para detectar vehículos. Por ejemplo: si $P = 0.9$, entonces el sensor tiene una precisión del 90 % de detectar vehículos. Dicha precisión está implementada dentro de la función **HayVehículo**. Además, un vehículo permanecerá imperceptible durante toda su trayectoria en la zona de detección desde el momento en que el sensor no lo detectó.

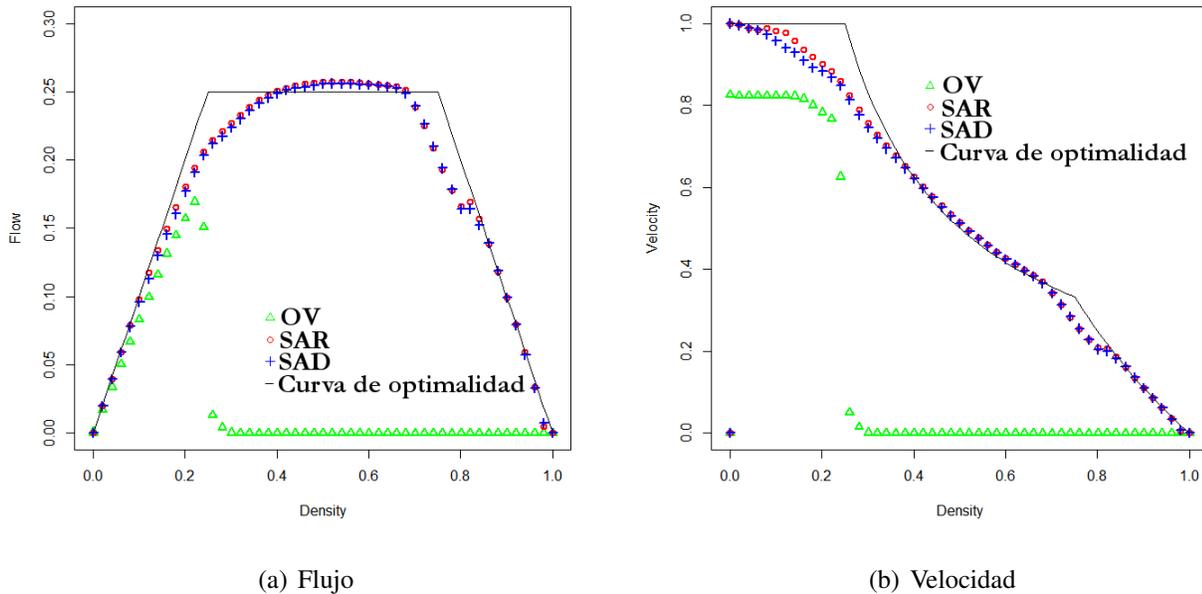


Figura 5.5: Tanto el sistema auto-organizante reactivo como el deliberativo logran un buen rendimiento para todas las densidades en comparación con el sistema ola verde.

Los experimentos consistieron en ejecutar la simulación en seis grados distintos de precisión del sensor para evaluar la robustez del SAR y SAD ante errores de detección. Es decir, con $P = 1.0$, $P = 0.9$, $P = 0.8$, $P = 0.7$, $P = 0.6$ y $P = 0.5$.

La configuración utilizada para cada ejecución es la misma de los experimentos de la sección anterior para evaluar el rendimiento de los tres sistemas para controlar los semáforos. Los resultados del SAR comparado con el SAD se muestra en la Figura 5.6.

Los resultados de la simulación con sensores a diferentes niveles de precisión muestran que el SAD es más robusto ante los errores de detección. Esto se debe a que el SAD mantiene el estado del tránsito en memoria y el error de detección sólo afecta a una celda. En el SAR el estado del tránsito no se mantiene y el error se distribuye en todas las celdas porque es inherente al tamaño de la zona de detección. Por lo tanto, el sistema reactivo no tiene forma de recuperarse y requiere un alto grado de precisión del sensor porque incluso en precisiones de 90 % el sistema colapsa.

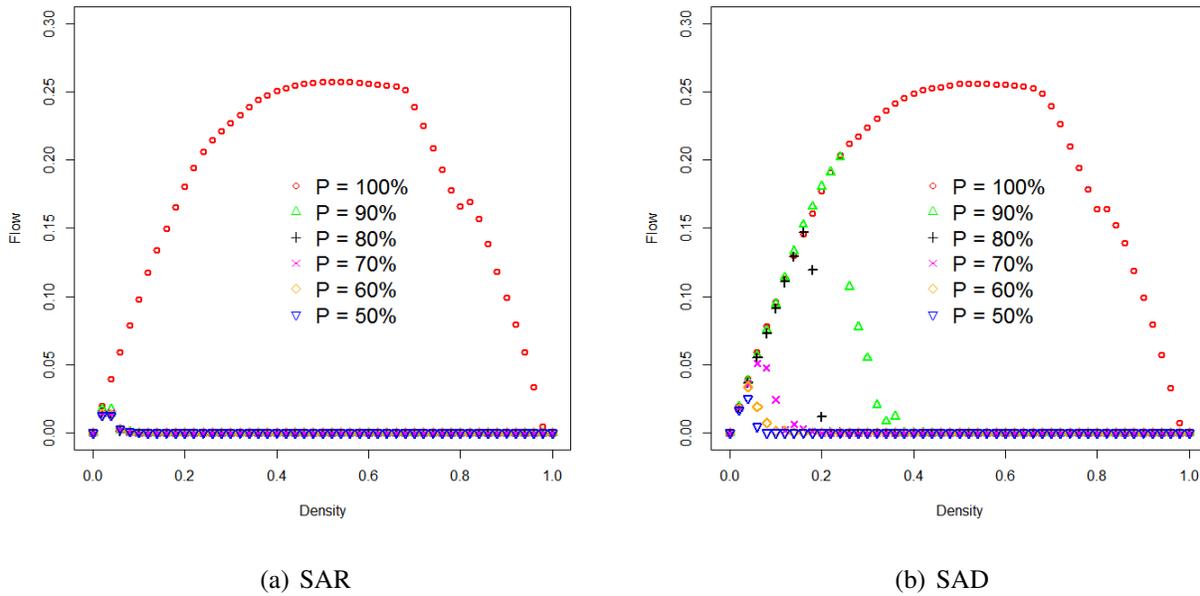


Figura 5.6: Resultados de aplicar sensores con diversas precisiones en la detección de vehículos. El parámetro P representa la precisión del sensor.

5.4. Experimentos y resultados con el modelo LAI

Realizamos experimentos aplicando el modelo vehicular Lárraga y Álvarez-Icaza (LAI) [20] con los *semáforos auto-organizantes reactivos* (SAR) y *deliberativos* (SAD) en una ciudad del tipo Manhattan de 10×10 intersecciones, es decir, hay 100 intersecciones. La distancia de todas las calles es de 330 celdas donde cada celda corresponde a $\Delta x = 2.5\text{m}$. Cada cuadra tiene una distancia de 32 celdas (sin la intersección).

Cada vehículo tiene una longitud $l_s = 2$ (5.0m). El paso de tiempo t se asigna a $1s$. Las velocidades se cambian de acuerdo con las reglas del modelo LAI y los valores relacionados a la velocidad son: $v_{max} = 12$, $v_s = 3$, $\Delta v = 1$ y $M = 2$. Los valores de los parámetros que aplican el comportamiento de los vehículos son: $R_d = 1.0$, $R_0 = 0.8$ y $R_s = 0.01$, dichos valores fueron propuestos por Lárraga y Álvarez-Icaza [20]. Los vehículos no tienen permitido incorporarse a otra calle en las intersecciones, es decir, siempre se desplazan sobre la misma calle.

Los experimentos consistieron en ejecutar la simulación en 50 densidades distintas tanto para el SAR y SAD. En cada densidad se promediaron veinte ejecuciones. Cada ejecución consistió en lo

siguiente: se simuló hora y media (5, 400) pasos donde los vehículos son colocados aleatoriamente en las calles de la ciudad. Posteriormente, es simulada otra hora y media donde la velocidad se mide en cada paso de tiempo. Al final de la simulación, se promedian las velocidades para obtener la velocidad media v y el flujo promedio J .

La Tabla muestra los valores de los parámetros utilizados en el SAR y SAD.

Parámetros	Valores	Escala
d	20 celdas	50m
r	10 celdas	25m
e	8 celdas	10m
u	10 pasos	10s
w	600 pasos	600s
n	13 veh · pasos	13 veh · s
m	2 vehículos	2 vehículos

Tabla 5.3: Parámetros usados por el SAR y SAD. El tamaño de la cuadra es de 32 celdas

El simulador con el que realizamos nuestros experimentos fue desarrollado en el lenguaje C. El lector puede acceder al código en la URL: <https://github.com/Zapotecat1/Traffic-Light>.

Los resultados se muestran en la Figura 5.7. El SAR tiene un desempeño en términos de flujo vehicular de 0.348. El SAD tiene un desempeño en términos de flujo vehicular de 0.307. En conclusión, el SAD tiene un desempeño ligeramente menor que el SAR aplicando el modelo LAI. Sin embargo, el SAD utiliza menos sensores.

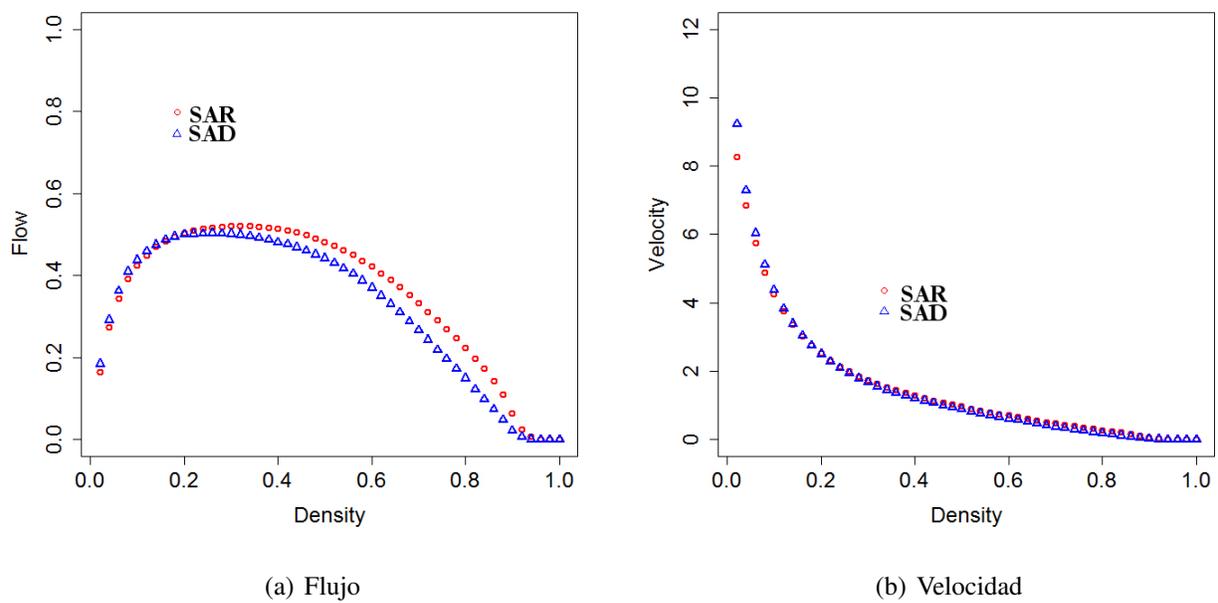


Figura 5.7: Los sistemas SAR y SAD tienen un rendimiento similar, sin embargo, el SAD utiliza un sensor simple.

Capítulo 6

Sistema de control

En este capítulo describimos un *sistema de control auto-organizante basado en impulso vehicular* que determina los cambios de luz de un semáforo. La idea esencial de este sistema de semáforos está basada en el concepto físico de *impulso*. El impulso es una fuerza que se aplica durante un determinado tiempo. Si la fuerza se aplica durante un tiempo largo, entonces produce más cambio de cantidad de movimiento que cuando se aplica durante un breve lapso de tiempo. Así, para cambiar la cantidad de movimiento de un objeto importan tanto la magnitud de la fuerza como el tiempo durante el cual actúa la fuerza. Formalmente, el impulso es la integral de una fuerza neta F en el intervalo de tiempo de t_1 a t_2 . De manera similar, en este sistema de control la asignación de luz verde depende de la cantidad de vehículos en una zona de detección y el tiempo que esperan durante una luz roja. Denominamos *impulso vehicular* a la integral del número de vehículos F_v en el intervalo de tiempo de t_1 a t_2 .

Definición 6.0.1. Impulso vehicular. El impulso vehicular J_v es la integral de la cantidad de vehículos F_v durante el intervalo de tiempo de t_c a t_f :

$$J_v = \int_{t_c}^{t_f} F_v dt \quad (6.1)$$

donde $[t_c, t_f]$ es un intervalo cerrado contenido en el conjunto de los números reales \mathbb{R} , t_c es el tiempo en que comienza una luz roja y t_f es el tiempo en que finaliza.

La asignación de luz verde cambia cuando el impulso vehicular en una calle con luz roja excede un *umbral de impulso* determinado por la cantidad de vehículos en la calle con luz verde. Las interacciones locales entre los semáforos con este sistema de control establecen patrones de coordinación emergentes como olas verdes. El sistema de control auto-organizante basado en impulso vehicular

ejecuta tres reglas: 1) otorgar la luz verde a la calle con mayor impulso vehicular, 2) evitar obstruir la intersección y 3) acotar tiempos de espera.

6.1. Estados del sistema de control

El sistema de control auto-organizante basado en impulso vehicular toma en cuenta los vehículos que hay en la región de longitud d_{total} . La distancia d_{total} incluye: la distancia d antes de la intersección; la distancia e después de la intersección; y la intersección (ver Figura 6.1). A fin de determinar cuántos vehículos hay en la distancia d_{total} se aplica el sistema de detección reactivo o deliberativo.

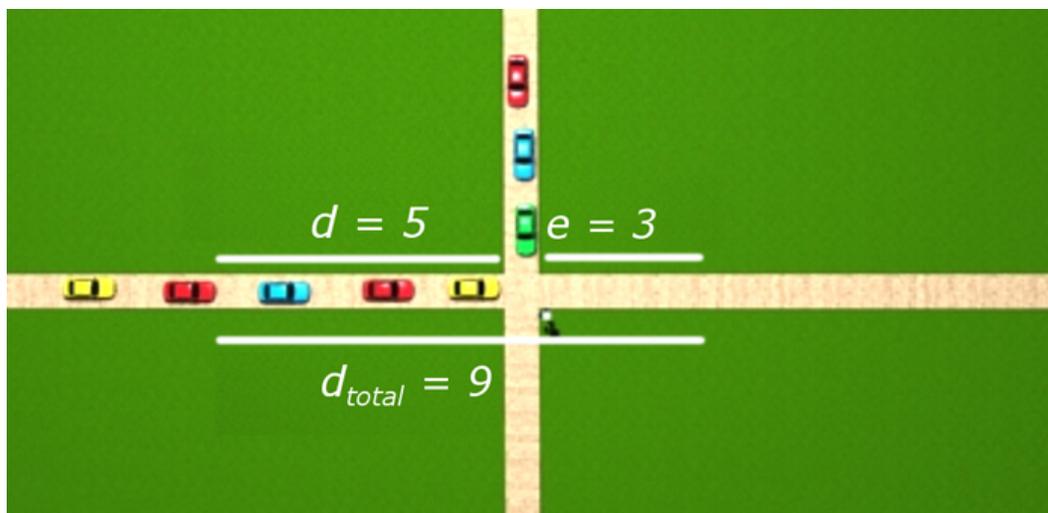


Figura 6.1: Distancias d_{total} , d y e

Los estados de las luces en la dirección horizontal y vertical son representados con las variables σ_h y σ_v .

Variable	Valor	Descripción
σ_h	verde	La luz del semáforo en la dirección horizontal es verde.
σ_v	verde	La luz del semáforo en la dirección vertical es verde.
σ_h	roja	La luz del semáforo en la dirección horizontal es roja.
σ_v	roja	La luz del semáforo en la dirección vertical es roja.

Las transiciones de los estados son determinadas por las tres reglas que se explican en las siguientes secciones.

6.2. Otorgar la luz verde a la calle con mayor impulso vehicular

La regla *otorgar la luz verde a la calle con mayor impulso vehicular* decide los cambios de luz de acuerdo con la cantidad de vehículos y el tiempo de espera. En el caso de un autómata celular el tiempo y el espacio son discretos. Por esta razón, consideramos la integral del impulso vehicular definido en la ecuación (6.1) mediante una sumatoria finita definida por la ecuación (6.2).

$$J_v = \sum_{i=1}^n F_v \Delta t \quad (6.2)$$

donde el intervalo de tiempo en luz roja $[t_c, t_f]$ es dividido en n pasos de tiempo por medio de los puntos $t_c = t_0 < t_1 < t_2 \dots < t_{n-1} < t_n = t_f$ y sea $\Delta t = t_i - t_{i-1} = 1$. La fuerza F_v representa el número de vehículos detectados a una distancia menor o igual que d en el paso de tiempo Δt .

Por ejemplo, considere la situación mostrada en la Figura 6.2. Si la dirección horizontal tiene luz roja, entonces un vehículo detectado durante $n = 6$ pasos de tiempo ejerce el mismo impulso que tres vehículos detectados en $n = 2$ pasos de tiempo.

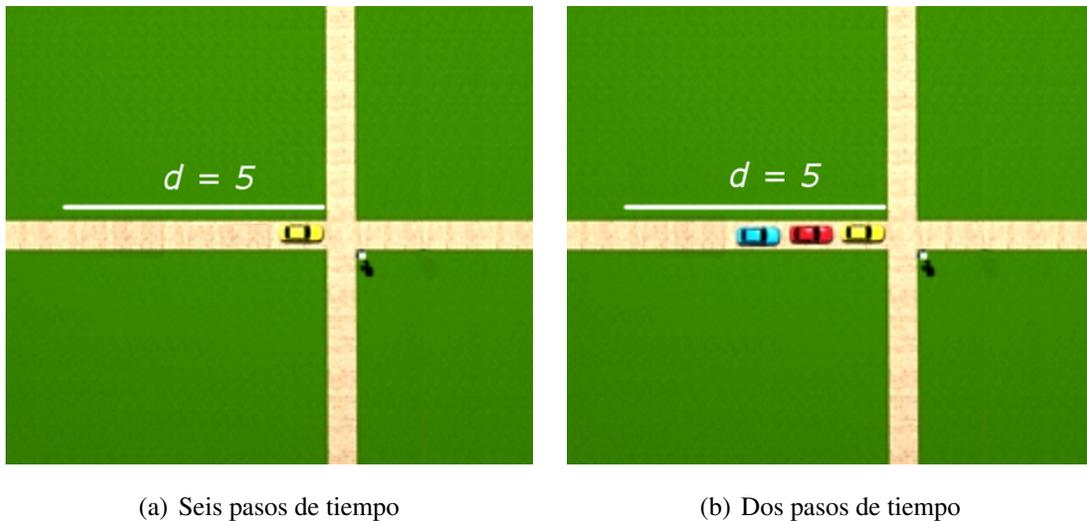


Figura 6.2: Los estados de las luces son $\sigma_h = roja$ y $\sigma_v = verde$. La distancia $d = 5$. (a) Un vehículo en $n = 6$ tiene un impulso igual a $J_v = 6$. (b) Tres vehículos en $n = 2$ tiene un impulso igual a $J_v = 6$.

El impulso vehicular determina si las luces deben cambiarse de rojo a verde. Por esta razón, es necesario establecer un umbral para el impulso vehicular que permita determinar los cambios de luz.

El umbral θ_{verde} representa la importancia en el instante t que tiene la dirección en luz verde y está definida por la ecuación (6.3).

$$\theta_{verde} = \tau F_{total} \quad (6.3)$$

donde F_{total} es el número de vehículos detectados a una distancia menor o igual que d_{total} y en el instante t , y la constante τ es el factor que determina qué importancia tiene el grupo de vehículos detectados. τ es definido en la sección 6.2.1.

A diferencia del impulso vehicular, donde son detectados vehículos en una región de longitud d antes de la intersección, en el umbral de impulso son detectados vehículos en una región de longitud d_{total} con el objetivo de identificar si un grupo de vehículos cruza la intersección al ocupar partes de la región determinada por las distancias d y e al mismo tiempo. De esta manera, el umbral de impulso es mayor en el caso donde los vehículos se detectan en la región que involucra ambas distancias que en los casos donde los vehículos se detectan solo en la región de longitud d antes de la intersección. Como resultado, el cambio de luz se retrasa para evitar que las colas, de los grupos se corten.

Por ejemplo, considere la situación mostrada en la Figura 6.3. Si la dirección horizontal tiene luz verde, entonces un vehículo detectado en el instante $t = 6$ ejerce menor importancia en comparación con tres vehículos detectados en $t = 2$.

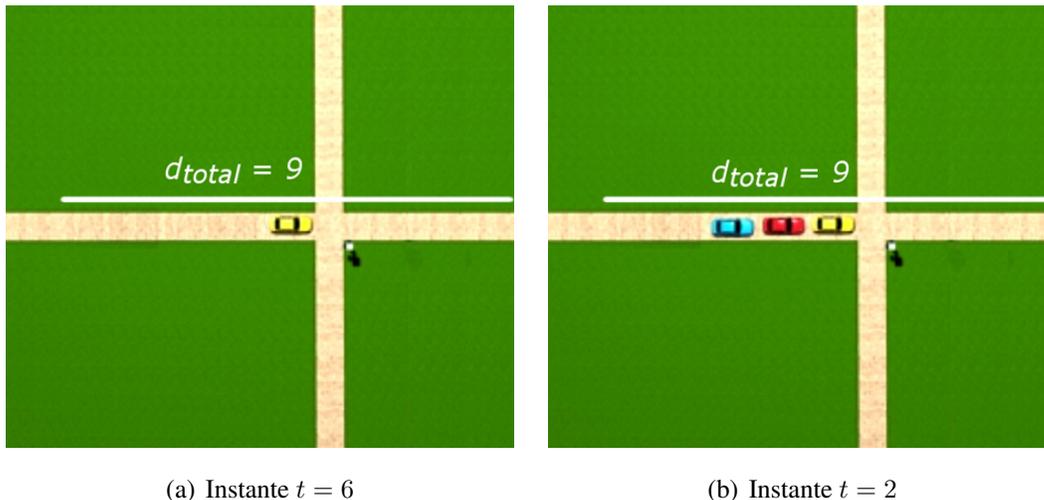


Figura 6.3: Los estados de las luces son $\sigma_h = verde$ y $\sigma_v = roja$. La distancia $d_{total} = 9$ y el factor de importancia es $\tau = 25$. Un vehículo en $t = 6$ tiene un umbral igual a $\theta_{verde} = 25$. (b) Tres vehículos en $t = 2$ tiene un umbral igual a $\theta_{verde} = 75$.

Finalmente, el semáforo ejecuta el cambio de luz cuando el impulso vehicular J_v es mayor que el umbral vehicular θ_{verde} , como se expresa en la desigualdad 6.4.

$$\theta_{verde} - J_v < 0 \quad (6.4)$$

La idea esencial de la desigualdad 6.4 es, por un lado, provocar que en densidades pequeñas los cambios de luz se produzcan rápidamente de manera parecida a la estrategia *primero en llegar, primero en ser servido* [29]. Por otro lado, la desigualdad provoca que en densidades medias los cambios de luz se produzcan con menor rapidez que en densidades bajas para generar grupos de vehículos de tamaño similar en ambas direcciones, de manera similar al sistema de semáforos auto-organizantes reactivos [4].

Por ejemplo, considere la situación mostrada en la Figura 6.4. Para que un grupo de tres vehículos en la calle vertical con luz roja genere el suficiente impulso vehicular sobre otro grupo de tres vehículos en la calle horizontal con luz verde, son necesarios nueve pasos de tiempo.

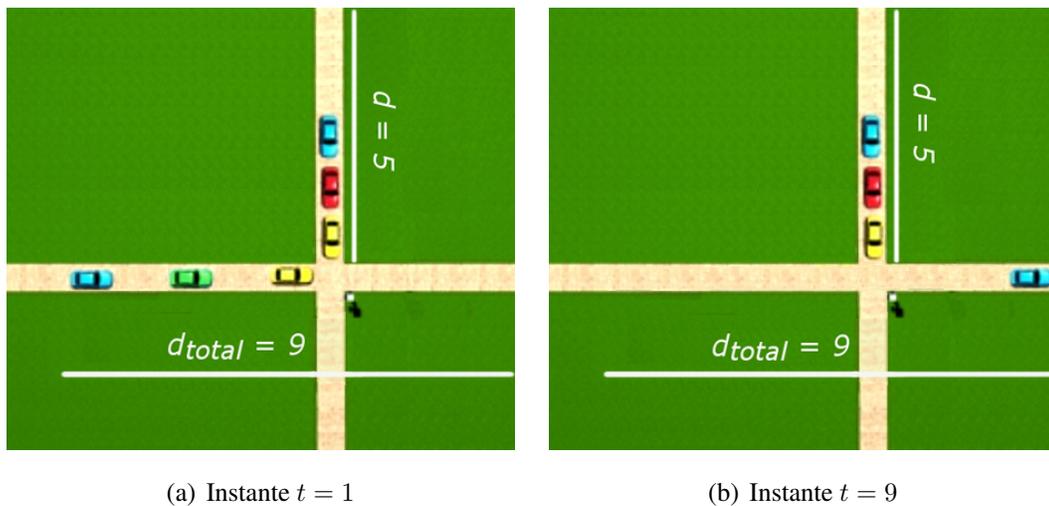


Figura 6.4: Los estados de las luces son $\sigma_h = verde$ y $\sigma_v = roja$. La distancia $d_{total} = 9$, el factor de importancia es $\tau = 25$ y la velocidad máxima es $v_{max} = 1$. En los pasos de tiempo $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, los valores del impulso vehicular J_v son asignados por la función $f(t)$ y los valores del umbral vehicular θ_{verde} son asignados por la función $g(t)$, de modo que $J_v = f(t) = \{(1, 3), (2, 6), (3, 9), (4, 12), (5, 15), (6, 18), (7, 21), (8, 24), (9, 27)\}$ y $\theta_{verde} = g(t) = \{(1, 75), (2, 75), (3, 75), (4, 75), (5, 75), (6, 50), (7, 50), (8, 25), (9, 25)\}$. En el instante $t = 9$ se cumple que $\theta_{verde} - p_{roja} = 25 - 27 < 0$ y consecuentemente el semáforo ejecuta el cambio de luces.

El Algoritmo 6 representa la regla *otorgar la luz verde a la calle con mayor impulso vehicular*. Los estados de las luces al inicio y al final de la ejecución de esta regla tiene que cumplir que $\sigma_h \neq \sigma_v$.

Algoritmo 6 Otorgar la luz verde a la calle con mayor impulso vehicular

función OTORGARVERDECALLEMAYORIMPULSOVEHICULAR()

$$J_v = J_v + F_v \Delta t$$

$$\theta_{verde} = \tau F_{total}$$

si $\theta_{verde} - J_v < 0$ **entonces**

$$\sigma_h = \neg \sigma_h$$

$$\sigma_v = \neg \sigma_v$$

$$J_v = 0$$

fin si

fin función

6.2.1. Calcular τ

El criterio para determinar el valor de τ consiste en repartir el tiempo de luz verde en altas densidades de manera equilibrada para ambas direcciones. Si hay el suficiente espacio para que los vehículos fluyan, entonces el peor de los casos es cuando hay la mayor cantidad de vehículos detenidos en la zona de detección d de la dirección en luz roja y la mayor cantidad de vehículos en movimiento en la zona de detección d_{total} de la dirección en luz verde. A fin de garantizar un tiempo máximo de espera n en dicha situación, el valor de τ tiene que determinarse. El impulso vehicular y el umbral vehicular están relacionados en la ecuación (6.5).

$$\sum_{i=1}^n F_v \Delta t = \tau F_{total} \quad (6.5)$$

La mayor cantidad de vehículos detenidos en la dirección de la luz roja está dada por la distancia de detección d dividida entre la longitud de los vehículos l_s . La mayor cantidad de vehículos moviéndose en la dirección de la luz verde está dada por la distancia de detección d_{total} dividida entre $l_s + 1$. Es decir, para que los vehículos puedan moverse necesitan que al menos la celda delantera de cada vehículo esté vacía (tanto para la regla 184 y LAI). Por esta razón, los términos de la ecuación (6.5) son sustituidos en la ecuación (6.6).

$$\frac{d}{l_s} n = \tau \frac{d_{total}}{l_s + 1} \quad (6.6)$$

Finalmente, el valor de τ está determinado por la ecuación (6.7).

$$\tau = n \frac{d + \frac{d}{l_s}}{d_{total}} \quad (6.7)$$

Si $\frac{d}{l_s} = \frac{d_{total}}{l_s + 1}$, entonces $\tau = n$. Si $\frac{d}{l_s} < \frac{d_{total}}{l_s + 1}$, entonces $\tau < n$. Si $\frac{d}{l_s} > \frac{d_{total}}{l_s + 1}$, entonces $\tau > n$. Por ejemplo, la Tabla 6.1 muestra los valores de τ a partir de diferentes d_{total} asumiendo que $l_s = 2$ y el valor deseado para cambiar a luz verde es de un tiempo máximo de espera $n = 30$ segundos.

Tabla 6.1: Valores de τ

d_{total}	d	$d_{total}/(l_s + 1)$	d/l_s	τ
30	20	10	10	30
31	20	10.333	10	29.032
29	20	9.666	10	31.034

La desigualdad $\theta_{verde} - J_v < 0$ que determina el cambio a luz verde se cumplirá como sigue:

- Si $\tau = 30$, entonces el cambio a luz verde en la dirección requerida se ejecutará en el segundo 31.
- Si $\tau = 29.032$, entonces el cambio a luz verde en la dirección requerida se ejecutará en el segundo 30.
- Si $\tau = 31.034$, entonces el cambio a luz verde en la dirección requerida se ejecutará en el segundo 32.

6.2.2. Corrección del Algoritmo 6

En esta sección el Algoritmo 6 se verifica formalmente mediante la lógica Hoare [32]. El estado inicial inválido de los semáforos para el Algoritmo 6 ocurre si ambas direcciones tienen luz verde o luz roja. Por esta razón, los estados de cada dirección son distintos y la precondition es $P = \{\sigma_h \neq \sigma_v\}$.

A fin de facilitar las operaciones de la demostración, el estado de las luces se especifica con 1 para el estado *verde* y 0 para el estado *rojo*. La condiciones de salida correctas para el Algoritmo 6 son las siguientes:

- Si el impulso vehicular es mayor que el umbral vehicular, entonces se intercambia el estado de las luces. En caso contrario, no se intercambia el estado de las luces. Por esta razón, la salida correcta es $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$.

La poscondición resultante es $Q = \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$. Si C representa el Algoritmo 6, entonces la demostración de la corrección del Algoritmo 6 consiste en verificar la terna de Hoare $\{\sigma_h \neq \sigma_v\} C \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$.

La demostración de dicha terna de Hoare consiste en verificar cada una de las ternas de Hoare especificadas en el Algoritmo 7.

Algoritmo 7 Ternas de Hoare del Algoritmo 6

- 1: $\{P\}$
 - 2: $J_v = J_v + F_v \Delta t$
 - 3: $\{P_1\}$
 - 4: $\theta_{verde} = \tau F_{total}$
 - 5: $\{P_2\}$
 - 6: **si** $\theta_{verde} - J_v < 0$ **entonces**
 - 7: $\{P_2 \wedge B\}$
 - 8: $\sigma_h = \neg \sigma_h$
 - 9: $\{S_1\}$
 - 10: $\sigma_v = \neg \sigma_v$
 - 11: $\{S_2\}$
 - 12: $J_v = 0$
 - 13: $\{Q\}$
 - 14: **fin si**
 - 15: $\{Q\}$
-

1. La precondition de la línea 7 es $\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} - J_v < 0)\}$
2. La terna de Hoare a verificar entre las líneas 7-13:

$$\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} - J_v < 0)\}$$

$$\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; J_v = 0;$$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$
3. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$J_v = 0;$$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$
4. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$\sigma_v = \neg\sigma_v; J_v = 0;$$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$
5. Axioma de asignación:

$$\{(\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 1 \wedge \sigma_v = 0)\}$$

$$\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; J_v = 0;$$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$
6. Fortalecimiento de la precondition:

$$\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} - J_v < 0)\} \Rightarrow \{(\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 1 \wedge \sigma_v = 0)\}$$
7. Por lo tanto la terna de Hoare

$$\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} - J_v < 0)\}$$

$$\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; J_v = 0;$$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

es correcta.
8. El caso negativo de la condición de la línea 6 implica Q :

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(\theta_{verde} - J_v < 0)\} \Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$
9. Regla de inferencia if/else para las líneas 6-14:

$$\frac{\begin{array}{l} \{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} - J_v < 0)\} \\ \sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; J_v = 0; \\ \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\} \\ \{(\sigma_h \neq \sigma_v) \wedge \neg(\theta_{verde} - J_v < 0)\} \Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\} \end{array}}{\{(\sigma_h \neq \sigma_v)\}}$$

Si $(\theta_{verde} - J_v < 0)$ entonces $\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; J_v = 0;$

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

10. La precondition de la línea 3 es $\{\sigma_h \neq \sigma_v\}$
11. La terna de Hoare a verificar entre las líneas 3-5:
 $\{\sigma_h \neq \sigma_v\}$
 $\theta_{verde} = \tau F_{total};$
 $\{\sigma_h \neq \sigma_v\}$
12. Axioma de asignación:
 $\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} = \tau F_{total})\}$
 $\theta_{verde} = \tau F_{total};$
 $\{\sigma_h \neq \sigma_v\}$
13. Debilitamiento de la precondition:
 $\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} = \tau F_{total})\} \Rightarrow \{\sigma_h \neq \sigma_v\}$
14. Por lo tanto la terna de Hoare
 $\{\sigma_h \neq \sigma_v\}$
 $\theta_{verde} = \tau F_{total};$
 $\{(\sigma_h \neq \sigma_v)\}$
 es correcta.
15. La precondition de la línea 2 es $\{\sigma_h \neq \sigma_v\}$
16. La terna de Hoare a verificar entre las líneas 1-3:
 $\{(\sigma_h \neq \sigma_v)\}$
 $J_v = J_v + F_v \Delta t;$
 $\{\sigma_h \neq \sigma_v\}$
17. Axioma de asignación:
 $\{(\sigma_h \neq \sigma_v) \wedge (\theta_{verde} = \tau F_{total})\}$
 $J_v = J_v + F_v \Delta t;$
 $\{\sigma_h \neq \sigma_v\}$
18. Debilitamiento de la precondition:
 $\{(\sigma_h \neq \sigma_v) \wedge (J_v = J_v + F_v \Delta t)\} \Rightarrow \{\sigma_h \neq \sigma_v\}$
19. Por lo tanto la terna de Hoare
 $\{\sigma_h \neq \sigma_v\}$
 $J_v = J_v + F_v \Delta t;$
 $\{\sigma_h \neq \sigma_v\}$
 es correcta.

Por lo tanto, queda demostrado que el Algoritmo 6 es correcto.

6.3. Evitar obstruir la intersección

La regla *evitar obstruir la intersección* asigna la luz roja a la dirección donde hay al menos un vehículo detenido en la intersección o más allá de la intersección a una distancia corta e . Si la regla asigna la luz roja en ambas direcciones, entonces asignará la luz verde hasta que una dirección sea liberada.

Las variables que registran si hay vehículos detenidos en la intersección o en la región de longitud e después de la intersección de las direcciones horizontal y vertical son E_h y E_v , respectivamente. El valor de E_h es verdadero si hay al menos un vehículo detenido en la intersección o en la región horizontal de longitud e , falso en caso contrario. De la misma manera, el valor de E_v es verdadero si hay al menos un vehículo detenido en la intersección o en la región vertical de longitud e , falso en caso contrario.

En la Tabla 6.2 se muestran los valores posibles para las variables E_h , E_v , el estado de las luces del semáforo y el valor de retorno de la regla.

Tabla 6.2: Evitar obstruir la intersección

E_h	E_v	Estados de las luces	Retorno
verdadero	falso	$\sigma_h = roja \wedge \sigma_v = verde$	falso
falso	verdadero	$\sigma_h = verde \wedge \sigma_v = roja$	falso
verdadero	verdadero	$\sigma_h = roja \wedge \sigma_v = roja$	falso
falso	falso	-	verdadero

Si el estado de las luces es roja en ambas direcciones y ambas direcciones son liberadas en el mismo instante, entonces la luz verde se asigna a la dirección que tenga el mayor tiempo de espera. El tiempo de espera en la dirección horizontal y vertical está denotado por las variables t_r^h y t_r^v , respectivamente.

Por ejemplo, la Figura 6.5 muestra las situaciones donde hay al menos un vehículo detenido en una dirección en la región de longitud e después de la intersección. La regla asigna luz verde en la dirección que no tiene vehículos detenidos en la región de longitud e después de la intersección. Además, como la regla detecta si no hay una calle obstruida, el valor de retorno es falso.

Si hay vehículos detenidos en las dos direcciones en la región de longitud e después de la intersección, entonces esta regla asigna luz roja en ambas direcciones y retorna falso (ver Figura 6.6). Únicamente si no hay vehículos detenidos en ambas direcciones y las dos direcciones no están en

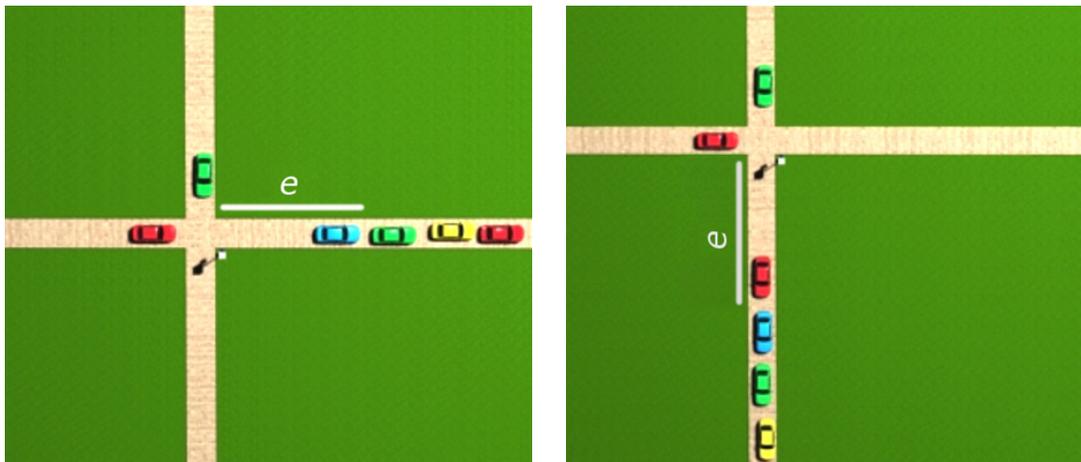
(a) $E_h = verdadero$.(b) $E_v = verdadero$.

Figura 6.5: Aplicación de la regla para evitar obstruir la intersección. (a) La regla asigna luz verde en la dirección vertical. (b) La regla asigna luz verde en la dirección horizontal.

luz roja, entonces la regla retorna *verdadero* para indicar que no hay problema respecto a obstruir la intersección (ver Figura 6.6).

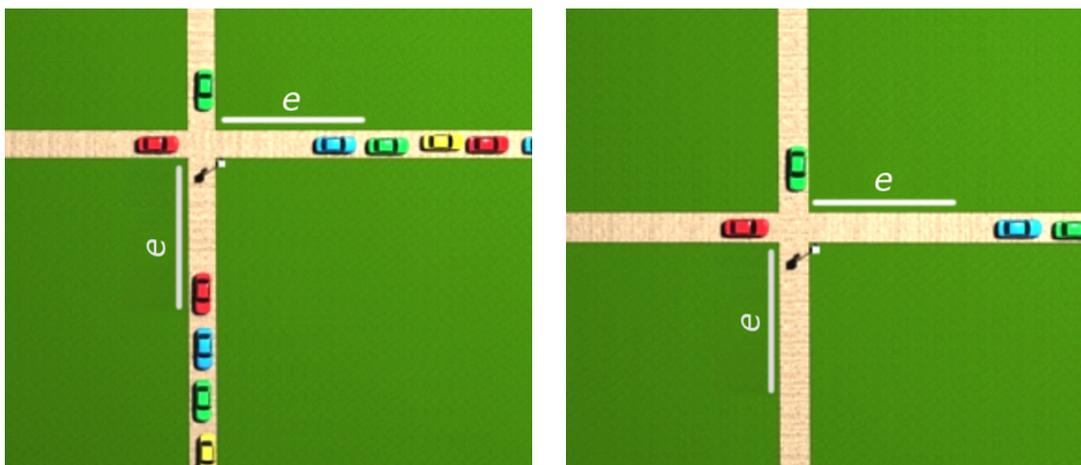
(a) $E_h = verdadero$ y $E_v = verdadero$.(b) $E_h = falso$ y $E_v = falso$.

Figura 6.6: Aplicación de la regla *evitar obstruir la intersección*. (a) La regla asigna luz roja en ambas direcciones. (b) Si en ambas direcciones no hay vehículos detenidos y las dos direcciones no están en luz roja, entonces la regla retorna *verdadero*.

El Algoritmo 8 representa la regla *evitar obstruir la intersección*.

Algoritmo 8 Evitar obstruir la intersección.

```

1: función EVITAROBSTRUIRINTERSECCIÓN()
2:   si  $E_h = verdadero \wedge E_v = verdadero$  entonces
3:      $\sigma_h = falso$ 
4:      $\sigma_v = falso$ 
5:      $r = falso$ 
6:   si no
7:     si  $E_h = falso \wedge E_v = falso$  entonces
8:       si  $\sigma_h = falso \wedge \sigma_v = falso$  entonces
9:         si  $t_r^h \geq t_r^v$  entonces
10:           $\sigma_h = verdadero$ 
11:           $\sigma_v = falso$ 
12:           $r = falso$ 
13:        si no
14:           $\sigma_h = falso$ 
15:           $\sigma_v = verdadero$ 
16:           $r = falso$ 
17:        fin si
18:      si no
19:         $\sigma_h = \sigma_h$ 
20:         $\sigma_v = \sigma_v$ 
21:         $r = verdadero$ 
22:      fin si
23:    si no
24:      si  $E_h = verdadero$  entonces
25:         $\sigma_h = falso$ 
26:         $\sigma_v = verdadero$ 
27:         $r = falso$ 
28:      si no
29:         $\sigma_h = verdadero$ 
30:         $\sigma_v = falso$ 
31:         $r = falso$ 
32:      fin si
33:    fin si
34:  fin si
35:  devolver  $r$ 
36: fin función

```

6.3.1. Corrección del Algoritmo 8

En esta sección el Algoritmo 8 se verifica formalmente mediante la lógica Hoare [32]. El estado inicial inválido de los semáforos para el Algoritmo 8 ocurre si las dos direcciones tienen luz verde. Por esta razón, la precondition es $P = \{(\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$.

A fin de facilitar las operaciones de la demostración, el estado de las luces se especifica con 1 para el estado *verde* y 0 para el estado *rojo*. De la misma manera, el estado verdadero y falso con 1 y 0, respectivamente. Las condiciones de salida correctas para el Algoritmo 8 son las siguientes:

- Si no hay vehículos detenidos en o más allá de la intersección, entonces no se cambia el estado de las luces y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $\{r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))\}$.
- Si no hay vehículos detenidos en o más allá de la intersección y ambas direcciones tienen luz roja, entonces se asigna la luz verde a la dirección que tiene más tiempo esperando una luz verde y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $\{r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$.
- Si hay al menos un vehículo detenido en o más allá de la intersección, entonces se asigna la luz verde a la dirección que no tiene dicho vehículo detenido y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $\{r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$.
- Si ambas direcciones tienen al menos un vehículo en o más allá de la intersección, entonces se asigna la luz roja en ambas direcciones y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $\{r = 0 \wedge (\sigma_h = 0 \wedge \sigma_v = 0)\}$.

La poscondición resultante es $Q = \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$

Si C representa el Algoritmo 8, entonces la demostración de la corrección del Algoritmo 8 consiste en verificar la terna de Hoare:

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

C

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

La demostración de dicha terna de Hoare consiste en verificar cada una de las ternas de Hoare especificadas en el Algoritmo 9.

Algoritmo 9 Evitar obstruir la intersección.

```

1:  $\{P\}$ 
2: si  $E_h = 1 \wedge E_v = 1$  entonces
3:    $\{P \wedge B\}$ 
4:    $\sigma_h = 0$ 
5:    $\{S_1\}$ 
6:    $\sigma_v = 0$ 
7:    $\{S_2\}$ 
8:    $r = 0$ 
9:    $\{Q\}$ 
10: si no
11:    $\{P \wedge \neg B\}$ 
12:   si  $E_h = 0 \wedge E_v = 0$  entonces
13:      $\{(P \wedge \neg B) \wedge B_1\}$ 
14:     si  $\sigma_h = 0 \wedge \sigma_v = 0$  entonces
15:        $\{((P \wedge \neg B) \wedge B_1) \wedge B_2\}$ 
16:       si  $t_r^h \geq t_r^v$  entonces
17:          $\{(((P \wedge \neg B) \wedge B_1) \wedge B_2) \wedge B_3\}$ 
18:          $\sigma_h = 1$ 
19:          $\{S_3\}$ 
20:          $\sigma_v = 0$ 
21:          $\{S_4\}$ 
22:          $r = 0$ 
23:          $\{Q\}$ 
24:       si no
25:          $\{(((P \wedge \neg B) \wedge B_1) \wedge B_2) \wedge \neg B_3\}$ 
26:          $\sigma_h = 0$ 
27:          $\{S_5\}$ 
28:          $\sigma_v = 1$ 
29:          $\{S_6\}$ 
30:          $r = 0$ 
31:          $\{Q\}$ 
32:       fin si
33:      $\{Q\}$ 

```

```
34:   si no
35:      $\{((P \wedge \neg B) \wedge B_1) \wedge \neg B_2\}$ 
36:      $\sigma_h = \sigma_h$ 
37:      $\{S_7\}$ 
38:      $\sigma_v = \sigma_v$ 
39:      $\{S_8\}$ 
40:      $r = 1$ 
41:      $\{Q\}$ 
42:   fin si
43:    $\{Q\}$ 
44: si no
45:    $\{(P \wedge \neg B) \wedge \neg B_1\}$ 
46:   si  $E_h = 1$  entonces
47:      $\{((P \wedge \neg B) \wedge \neg B_1) \wedge B_4\}$ 
48:      $\sigma_h = 0$ 
49:      $\{S_9\}$ 
50:      $\sigma_v = 1$ 
51:      $\{S_{10}\}$ 
52:      $r = 0$ 
53:      $\{Q\}$ 
54:   si no
55:      $\{((P \wedge \neg B) \wedge \neg B_1) \wedge \neg B_4\}$ 
56:      $\sigma_h = 1$ 
57:      $\{S_{11}\}$ 
58:      $\sigma_v = 0$ 
59:      $\{S_{12}\}$ 
60:      $r = 0$ 
61:      $\{Q\}$ 
62:   fin si
63:    $\{Q\}$ 
64: fin si
65:    $\{Q\}$ 
66: fin si
67:  $\{Q\}$ 
68: devolver  $r$ 
```

1. La precondition de la línea 55 es $\{(P \wedge \neg B) \wedge \neg B_1\} \wedge \neg B_4 =$
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge \neg E_h = 1\}$
2. La terna de Hoare a verificar entre las líneas 55-61:
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge \neg E_h = 1\}$
 $\sigma_h = 1; \sigma_v = 0; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
3. Axioma de asignación y ley de identidad:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$
 $r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
4. Axioma de asignación:
 $\{\sigma_h = 1 \vee \sigma_h = 0\}$
 $\sigma_v = 0; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
5. Axioma de asignación:
 $\{\text{verdadero}\}$
 $\sigma_h = 1; \sigma_v = 0; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
6. Fortalecimiento de la precondition:
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge \neg E_h = 1\}$
 $\Rightarrow \{\text{verdadero}\}$
7. Por lo tanto la terna de Hoare
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge \neg E_h = 1\}$
 $\sigma_h = 1; \sigma_v = 0; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
 es correcta.
8. La precondition de la línea 47 es $\{(P \wedge \neg B) \wedge \neg B_1\} \wedge B_4 =$
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge E_h = 1\}$
9. La terna de Hoare a verificar entre las líneas 47-53:
 $\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge E_h = 1\}$
 $\sigma_h = 0; \sigma_v = 1; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
10. Axioma de asignación y ley de identidad:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$
 $r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$
11. Axioma de asignación:
 $\{\sigma_h = 0\}$
 $\sigma_v = 1; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$

12. Axioma de asignación:

$$\{\text{verdadero}\}$$

$$\sigma_h = 0; \sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

13. Fortalecimiento de la precondition:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge E_h = 1\}$$

$$\Rightarrow \{\text{verdadero}\}$$

14. Por lo tanto la terna de Hoare

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge E_h = 1\}$$

$$\sigma_h = 0; \sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

es correcta.

15. Regla de inferencia if/else para las líneas 45-63:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge E_h = 1\}$$

$$\sigma_h = 0; \sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)) \wedge \neg E_h = 1\}$$

$$\sigma_h = 1; \sigma_v = 0; r = 0;$$

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0))\}$$

Si $E_h = 1$ entonces $\sigma_h = 0; \sigma_v = 1; r = 0;$ en caso contrario $\sigma_h = 1; \sigma_v = 0; r = 0;$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

16. La precondition de la línea 25 es $\{((P \wedge \neg B) \wedge B_1) \wedge B_2) \wedge \neg B_3\} =$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(t_r^h \geq t_r^v)\}$$

17. La terna de Hoare a verificar entre las líneas 29-35:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(t_r^h \geq t_r^v)\}$$

$$\sigma_h = 0; \sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

18. Axioma de asignación y ley de identidad:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

19. Axioma de asignación:

$$\{\sigma_h = 0\}$$

$$\sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

20. Axioma de asignación:

$$\{\text{verdadero}\}$$

$$\sigma_h = 0; \sigma_v = 1; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

21. Fortalecimiento de la precondition:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(t_r^h \geq t_r^v)\}$$

$$\Rightarrow \{\text{verdadero}\}$$

22. Por lo tanto la terna de Hoare

$$\begin{aligned} & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge \neg(t_r^h \geq t_r^v)\} \\ & \sigma_h = 0; \sigma_v = 1; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \\ & \text{es correcta.} \end{aligned}$$

23. La precondition de la línea 17 es $\{((P \wedge \neg B) \wedge B_1) \wedge B_2\} =$

$$\{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge (t_r^h \geq t_r^v)\}$$

24. La terna de Hoare a verificar entre las líneas 17-23:

$$\begin{aligned} & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge (t_r^h \geq t_r^v)\} \\ & \sigma_h = 1; \sigma_v = 0; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}$$

25. Axioma de asignación y ley de identidad:

$$\begin{aligned} & \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\} \\ & r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}$$

26. Axioma de asignación:

$$\begin{aligned} & \{\sigma_h = 1 \vee \sigma_h = 0\} \\ & \sigma_v = 0; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}$$

27. Axioma de asignación:

$$\begin{aligned} & \{\text{verdadero}\} \\ & \sigma_h = 1; \sigma_v = 0; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}$$

28. Fortalecimiento de la precondition:

$$\begin{aligned} & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge (t_r^h \geq t_r^v)\} \\ & \Rightarrow \{\text{verdadero}\} \end{aligned}$$

29. Por lo tanto la terna de Hoare

$$\begin{aligned} & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge (t_r^h \geq t_r^v)\} \\ & \sigma_h = 1; \sigma_v = 0; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \\ & \text{es correcta.} \end{aligned}$$

30. Regla de inferencia if/else para las líneas 15-33:

$$\begin{aligned} & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge (t_r^h \geq t_r^v)\} \\ & \sigma_h = 1; \sigma_v = 0; r = 0; \\ & \{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\} \\ & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0) \wedge \neg(t_r^h \geq t_r^v)\} \\ & \sigma_h = 0; \sigma_v = 1; r = 0; \\ & \{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\} \\ & \hline & \{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)\} \\ & \text{Si } (t_r^h \geq t_r^v) \text{ entonces } \sigma_h = 1; \sigma_v = 0; r = 0; \text{ en caso contrario } \sigma_h = 0; \sigma_v = 1; r = 0; \\ & \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}$$

31. La precondition de la línea 35 es $\{(P \wedge \neg B) \wedge B_1\} \wedge \neg B_2 =$

$$\{((((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)) \wedge \neg(\sigma_h = 0 \wedge \sigma_v = 0)\}$$

32. La terna de Hoare a verificar entre las líneas 35-41 es:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge \neg(\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

33. Axioma de asignación y ley de identidad:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

34. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

35. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

36. Fortalecimiento de la precondition:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge \neg(\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$\Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

37. Por lo tanto la terna de Hoare

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge \neg(\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

es correcta.

38. La precondition de la línea 15 es $\{(P \wedge \neg B) \wedge B_1\} \wedge B_2 =$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)\}.$$

El código C_1 representa las líneas 16-32 verificadas en pasos anteriores.

39. Regla de inferencia if/else para el código entre las líneas 13-43:

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge (\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$C_1;$$

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0) \wedge \neg(\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)\}$$

Si $(\sigma_h = 0 \wedge \sigma_v = 0)$ entonces C_1 ; en caso contrario $\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1$;

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

40. La precondition de la línea 11 es

$$\{(P \wedge \neg B) \wedge B_1\} = \{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)\}.$$

La precondition de la línea 45 es

$$\{(P_1 \wedge \neg B) \wedge \neg B_1\} = \{(((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)\}.$$

El código C_2 representa las líneas 12-44 y el código C_3 representa las líneas 45-64.

41. Regla de inferencia if/else para el código entre las líneas 11-65

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge (E_h = 0 \wedge E_v = 0)\}$$

C_2 ;

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)) \wedge \neg(E_h = 0 \wedge E_v = 0)\}$$

C_3 ;

$$\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\}$$

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)\}$$

Si $(E_h = 0 \wedge E_v = 0)$ entonces C_2 ; en caso contrario C_3 ;

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

42. La precondition de la línea 11 es $\{P \wedge \neg B\} =$

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)\}. \text{ El código } C_4 \text{ representa las líneas 11-64 y fue verificado anteriormente.}$$

43. La precondition de la línea 3 es $\{P \wedge B\} = \{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge (E_h = 1 \wedge E_v = 1)\}$

44. La terna de Hoare a verificar entre las líneas 3-9 es:

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge (E_h = 1 \wedge E_v = 1)\}$$

$$\sigma_h = 0; \sigma_v = 0; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

45. Axioma de asignación y ley de identidad:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)\}$$

$$r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

46. Axioma de asignación:

$$\{\sigma_h = 1 \vee \sigma_h = 0\}$$

$$\sigma_v = 0; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

47. Axioma de asignación:

$$\{\text{verdadero}\}$$

$$\sigma_h = 1; \sigma_v = 0; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

48. Fortalecimiento de la precondition:

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge (E_h = 1 \wedge E_v = 1)\}$$

$$\Rightarrow \{\text{verdadero}\}$$

49. Por lo tanto la terna de Hoare

$$\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge (E_h = 1 \wedge E_v = 1)\}$$

$$\sigma_h = 0; \sigma_v = 0; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\}$$

es correcta.

50. Regla de inferencia if/else para las líneas 2-66:

$$\frac{\begin{aligned} &\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge (E_h = 1 \wedge E_v = 1)\} \\ &\sigma_h = 0; \sigma_v = 0; r = 0; \\ &\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\} \\ &\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0)) \wedge \neg(E_h = 1 \wedge E_v = 1)\} \\ &C_4; \\ &\{(r = 1 \wedge \sigma_h = \sigma_h \wedge \sigma_v = \sigma_v) \vee (r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\} \end{aligned}}{\begin{aligned} &\{((\sigma_h \neq \sigma_v) \vee (\sigma_h = 0 \wedge \sigma_v = 0))\} \\ &\text{Si } (E_h = 1 \wedge E_v = 1) \text{ entonces } \sigma_h = 0; \sigma_v = 0; r = 0; \text{ en caso contrario } C_4; \\ &\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1) \vee (\sigma_h = 0 \wedge \sigma_v = 0)))\} \end{aligned}}$$

Por lo tanto, queda demostrado que el Algoritmo 8 es correcto.

6.4. Acotar tiempos de espera

La regla *acotar tiempos de espera* asegura que la luz verde permanezca durante un tiempo mínimo y durante un tiempo máximo. La variable t_v denota el tiempo transcurrido en luz verde. Las constantes T_{\min} y T_{\max} denotan los tiempos límite mínimo y máximo en luz verde, respectivamente. Los tres intervalos de tiempo determinados por T_{\min} y T_{\max} son: A , B y C (ver Figura 6.7). Los cambios de luces dependen del intervalo de tiempo transcurrido t_v .

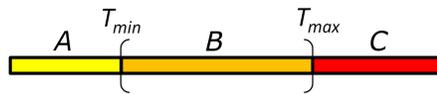


Figura 6.7: La cantidad de tiempo transcurrido en luz verde t_v determina los cambios de luces.

En la Tabla 6.3 se muestran los intervalos posibles para la variable t_{verde} . Además, el estado σ que adquiere el *semáforo* y el valor de retorno de la regla en relación con los intervalos de tiempo A , B y C . Únicamente si el tiempo transcurrido t_v pertenece al intervalo B , entonces la regla devuelve *verdadero*.

Tabla 6.3: Acotar tiempos de espera

t_{verde}	Estados de las luces	Retorno
A	$(\sigma_h = \sigma_h) \wedge (\sigma_v = \sigma_v)$ (no se cambian)	falso
B	— (se delega a otra regla)	verdadero
C	$(\sigma_h = \neg\sigma_h) \wedge (\sigma_v = \neg\sigma_v)$ (se cambian)	falso

El Algoritmo 10 representa la regla *acotar tiempos de espera*.

Algoritmo 10 Acotar tiempos de espera

```

1: función ACOTARTIEMPOSESPERA()
2:   si  $t_v \geq T_{min} \wedge t_v \leq T_{max}$  entonces
3:      $\sigma_h = \sigma_h$ 
4:      $\sigma_v = \sigma_v$ 
5:      $r = verdadero$ 
6:   si no
7:     si  $t_v < T_{min}$  entonces
8:        $\sigma_h = \sigma_h$ 
9:        $\sigma_v = \sigma_v$ 
10:       $r = falso$ 
11:    si no
12:       $\sigma_h = \neg\sigma_h$ 
13:       $\sigma_v = \neg\sigma_v$ 
14:       $r = falso$ 
15:    fin si
16:  fin si
17:  devolver  $r$ 
18: fin función

```

6.4.1. Corrección del Algoritmo 10

En esta sección el Algoritmo 10 se verifica formalmente mediante la lógica Hoare [32]. El estado inicial inválido de los semáforos para el Algoritmo 10 ocurre si las dos direcciones tienen luz verde o roja. Por esta razón, la precondition es $P = \{\sigma_h \neq \sigma_v\}$.

A fin de facilitar las operaciones de la demostración, el estado de las luces se especifica con 1 para el estado *verde* y 0 para el estado *rojo*. Las condiciones de salida correctas para el Algoritmo 10 son las siguientes:

- Si el tiempo en luz verde es mayor o igual que el tiempo mínimo y menor o igual que el tiempo máximo, entonces no se cambian las luces y se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $r = 1 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)$.
- Si el tiempo en luz verde es menor que el tiempo mínimo, entonces no se cambia el estado de las luces y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)$.
- Si el tiempo en luz verde es mayor que el tiempo máximo, entonces se cambian las luces y no se delega los cambios de luz a otra regla. Por esta razón, la salida correcta es $r = 0 \wedge (\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)$.

La poscondición resultante es $Q = \{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$.

Si C representa el Algoritmo 10, entonces la demostración de la corrección del Algoritmo 10 consiste en verificar la terna de Hoare:

$$\{(\sigma_h \neq \sigma_v)\}$$

C

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

La demostración de dicha terna de Hoare consiste en verificar cada una de las ternas de Hoare especificadas en el Algoritmo 11.

Algoritmo 11 Acotar tiempos de espera

```

1:  $\{P\}$ 
2: si  $t_v \geq T_{min} \wedge t_v \leq T_{max}$  entonces
3:    $\{P \wedge B\}$ 
4:    $\sigma_h = \sigma_h$ 
5:    $\{S_1\}$ 
6:    $\sigma_v = \sigma_v$ 
7:    $\{S_2\}$ 
8:    $r = verdadero$ 
9:    $\{Q\}$ 
10: si no
11:    $\{P \wedge \neg B\}$ 
12:   si  $t_v < T_{min}$  entonces
13:      $\{(P \wedge \neg B) \wedge B_1\}$ 
14:      $\sigma_h = \sigma_h$ 
15:      $\{S_3\}$ 
16:      $\sigma_v = \sigma_v$ 
17:      $\{S_4\}$ 
18:      $r = falso$ 
19:      $\{Q\}$ 
20:   si no
21:      $\{(P \wedge \neg B) \wedge \neg B_1\}$ 
22:      $\sigma_h = \neg \sigma_h$ 
23:      $\{S_5\}$ 
24:      $\sigma_v = \neg \sigma_v$ 
25:      $\{S_6\}$ 
26:      $r = falso$ 
27:      $\{Q\}$ 
28:   fin si
29:    $\{Q\}$ 
30: fin si
31:  $\{Q\}$ 

```

1. La precondition de la línea 21 es $\{(P \wedge \neg B) \wedge \neg B_1\} = \{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge \neg(t_v < T_{min})\}$
2. La terna de Hoare a verificar entre las líneas 21-27:
 $\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge \neg(t_v < T_{min})\}$
 $\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
3. Axioma de asignación y ley de identidad:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$
 $r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
4. Axioma de asignación:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$
 $\sigma_v = \neg\sigma_v; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
5. Axioma de asignación:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$
 $\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
6. Fortalecimiento de la precondition:
 $\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge \neg(t_v < T_{min})\}$
 $\Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$
7. Por lo tanto la terna de Hoare
 $\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge \neg(t_v < T_{min})\}$
 $\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
 es correcta.
8. La precondition de la línea 13 es $\{(P \wedge \neg B) \wedge B_1\} = \{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge (t_v < T_{min})\}$
9. La terna de Hoare a verificar entre las líneas 13-19:
 $\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge (t_v < T_{min})\}$
 $\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 0;$
 $\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$
10. Axioma de asignación y ley de identidad:
 $\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$

$$r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

11. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_v = \sigma_v; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

12. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

13. Fortalecimiento de la precondition:

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge (t_v < T_{min})\}$$

$$\Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

14. Por lo tanto la terna de Hoare

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge (t_v < T_{min})\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

es correcta.

15. Regla de inferencia if/else para las líneas 11-29:

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge (t_v < T_{min})\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max}) \wedge \neg(t_v < T_{min})\}$$

$$\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; r = 0;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max})\}$$

Si $(t_v < T_{min})$ entonces $\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 0;$ en caso contrario $\sigma_h = \neg\sigma_h; \sigma_v = \neg\sigma_v; r = 0;$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

16. La precondition de la línea 11 es $\{(P \wedge \neg B)\} =$

$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max})\}$. El código C_1 representa las líneas 12-28 que fueron verificadas anteriormente.

17. La precondition de la línea 3 es $\{(P \wedge B)\} =$

$$\{(\sigma_h \neq \sigma_v) \wedge (T_{min} \leq t_v \leq T_{max})\}$$

18. La terna de Hoare a verificar entre las líneas 3-9 es:

$$\{(\sigma_h \neq \sigma_v) \wedge (T_{min} \leq t_v \leq T_{max})\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

19. Axioma de asignación y ley de identidad:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

20. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

21. Axioma de asignación:

$$\{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

22. Fortalecimiento de la precondition:

$$\{(\sigma_h \neq \sigma_v) \wedge (T_{min} \leq t_v \leq T_{max})\}$$

$$\Rightarrow \{(\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)\}$$

23. Por lo tanto la terna de Hoare

$$\{(\sigma_h \neq \sigma_v) \wedge (T_{min} \leq t_v \leq T_{max})\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

es correcta.

24. Regla de inferencia if/else para las líneas 2-30:

$$\{(\sigma_h \neq \sigma_v) \wedge (T_{min} \leq t_v \leq T_{max})\}$$

$$\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1;$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

$$\{(\sigma_h \neq \sigma_v) \wedge \neg(T_{min} \leq t_v \leq T_{max})\}$$

$$C_1$$

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

$$\{\sigma_h \neq \sigma_v\}$$

Si $(T_{min} \leq t_v \leq T_{max})$ entonces $\sigma_h = \sigma_h; \sigma_v = \sigma_v; r = 1$; en caso contrario C_1 ;

$$\{(r = 1 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1))) \vee (r = 0 \wedge ((\sigma_h = 1 \wedge \sigma_v = 0) \vee (\sigma_h = 0 \wedge \sigma_v = 1)))\}$$

Por lo tanto, queda demostrado que el Algoritmo 10 es correcto.

6.5. Integración de las reglas

En la sección de experimentos se muestra que a fin de incrementar el flujo vehicular y garantizar la seguridad en la intersección es necesario aplicar las reglas como se muestra en el Algoritmo 12.

Las reglas previamente mencionadas son aplicadas con las funciones: **EvitarObstruirIntersección**, **AcotarTiemposEspera** y **OtorgarVerdeCalleMayorImpulsoVehicular**. Cada función retorna un valor verdadero o falso que es asignado a la variable r . La ejecución de las funciones depende de condiciones anidadas (ver Algoritmo 12).

Algoritmo 12 Control de semáforos basados en impulso vehicular

```

1:  $J_v = 0$ 
2: mientras verdadero hacer
3:    $r = \text{EVITAROBSTRUIRINTERSECCIÓN}()$ 
4:   si  $r == \text{verdadero}$  entonces
5:      $r = \text{ACOTARTIEMPOSESPERA}()$ 
6:     si  $r == \text{verdadero}$  entonces
7:        $\text{OTORGARVERDECALLEMAYORIMPULSOVEHICULAR}()$ 
8:     fin si
9:   fin si
10: fin mientras

```

6.6. Experimentos y resultados

El *semáforo auto-organizante reactivo* (SAR) está compuesto por el sistema de *detección reactivo* y el *sistema de control auto-organizante* propuesto por Gershenson [4]. El *semáforo auto-organizante reactivo basado en impulso vehicular* (SARIV) está compuesto por el *sistema de detección reactivo* y el *sistema de control auto-organizante basado en impulso vehicular* descrito en este capítulo.

Los experimentos consistieron en comparar el SAR y el SARIV en una ciudad del tipo Manhattan. La ciudad estaba compuesta por 100×100 intersecciones, es decir, había 10,000 intersecciones. La distancia de todas las calles es de 3,300 celdas donde cada celda corresponde a $\Delta x = 2.5\text{m}$. Cada cuadra tiene una distancia de 32 celdas (sin la intersección).

El desempeño del SAR y SARIV fue evaluado con la regla 184 y el modelo LAI (Lárraga y Álvarez-Icaza) [20]. Cada vehículo tenía una longitud $l_s = 2$ (5.0m). El paso de tiempo t fue asignado

a 1s. En el caso de la regla 184 la velocidad máxima de cada vehículo fue $v_{max} = 1$. En el caso del modelo LAI la velocidad máxima de cada vehículo fue $v_{max} = 12$, la velocidad baja fue $v_s = 3$, el cambio en la aceleración fue $\Delta v = 1$ y la capacidad de frenar urgentemente fue $M = 2$. Los valores de los parámetros que regulan las velocidades de los vehículos fueron: $R_d = 1.0$, $R_0 = 0.8$ y $R_s = 0.01$, dichos valores fueron propuestos por Lárraga y Álvarez-Icaza [20]. Los vehículos no tenían permitido incorporarse a otra calle en las intersecciones, es decir, siempre se desplazaban sobre la misma calle.

Los experimentos consistieron en ejecutar la simulación en 50 densidades distintas para el SAR y para el SARIV. En cada densidad se ejecutaron 20 simulaciones y los vehículos fueron colocados inicialmente en las calles de la ciudad de manera aleatoria. Cada simulación constó de dos fases. En la primera fase se simularon 5,400 pasos de tiempo (hora y media) para relajar el sistema. En la segunda fase se simularon otros 5,400 pasos de tiempo donde la velocidad se midió en cada paso de tiempo. Finalmente, para cada densidad se obtuvo el promedio de las 20 simulaciones respecto a la velocidad v y el flujo J .

Todos los experimentos de esta sección utilizan la configuración y los parámetros del sistema de detección especificados en la Tabla 6.4. El simulador con el que realizamos nuestros experimentos fue desarrollado en el lenguaje C. El lector puede acceder al código en la URL: <https://github.com/Zapotecat1/Traffic-Light>.

Parámetros	SAR	SARIV
d	20 celdas (50m)	20 celdas (50m)
r	10 celdas (25m)	-
e	8 celdas (15m)	8 celdas (15m)
u	10 pasos (10s)	10 pasos (10s)
w	60 pasos (60s)	60 pasos (60s)
n	13 veh · pasos	-
m	2 vehículos	-
τ	-	32

Tabla 6.4: Parámetros usados por los sistemas auto-organizantes. El tamaño de la cuadra es de 32 celdas

6.6.1. Evaluación de las reglas

En esta sección examinamos los resultados obtenidos cuando se aplican las reglas del SARIV individualmente. El primer experimento consistió en determinar el desempeño, en términos del flujo vehicular, de la regla *otorgar la luz verde a la calle con mayor impulso vehicular* en los modelos vehiculares 184 y LAI (ver Figura 6.8). El flujo promedio es $J \approx 0.069$ y $J \approx 0.070$ para la regla 184 y el modelo LAI, respectivamente.

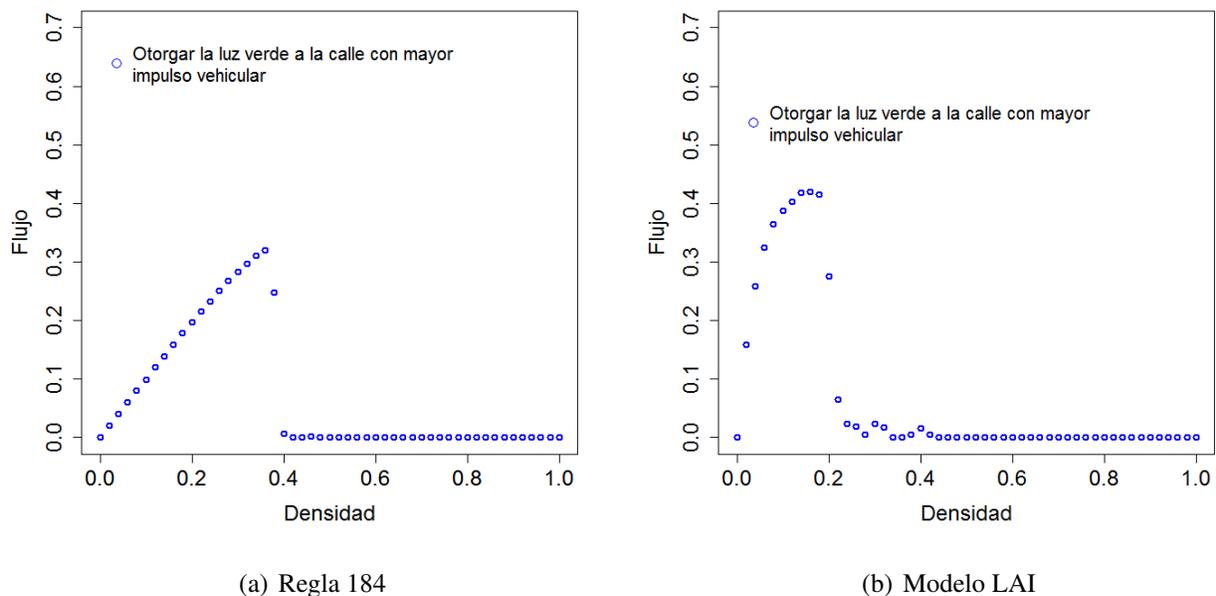


Figura 6.8: La regla *otorgar la luz verde a la calle con mayor impulso vehicular* con el modelo 184 y LAI.

El flujo vehicular disminuyó en densidades $\rho > 0.36$ y $\rho > 0.16$, para la regla 184 y el modelo LAI, respectivamente. El flujo disminuyó porque en cuadras con alta densidad de vehículos ocurre que un vehículo obstruía la intersección. La obstrucción en una intersección desencadenaba el interbloqueo (*deadlock*) en cuadras vecinas. Por esta razón, es necesario complementar la regla *otorgar la luz verde a la calle con mayor impulso vehicular* con la regla *evitar obstruir la intersección*.

El segundo experimento consistió en determinar el desempeño, en términos del flujo vehicular, de la regla *otorgar la luz verde a la calle con mayor impulso vehicular* y *evitar obstruir la intersección* con la regla 184 y el modelo LAI (ver Figura 6.9). El flujo promedio fue $J \approx 0.217$ y $J \approx 0.351$ para la regla 184 y el modelo LAI, respectivamente.

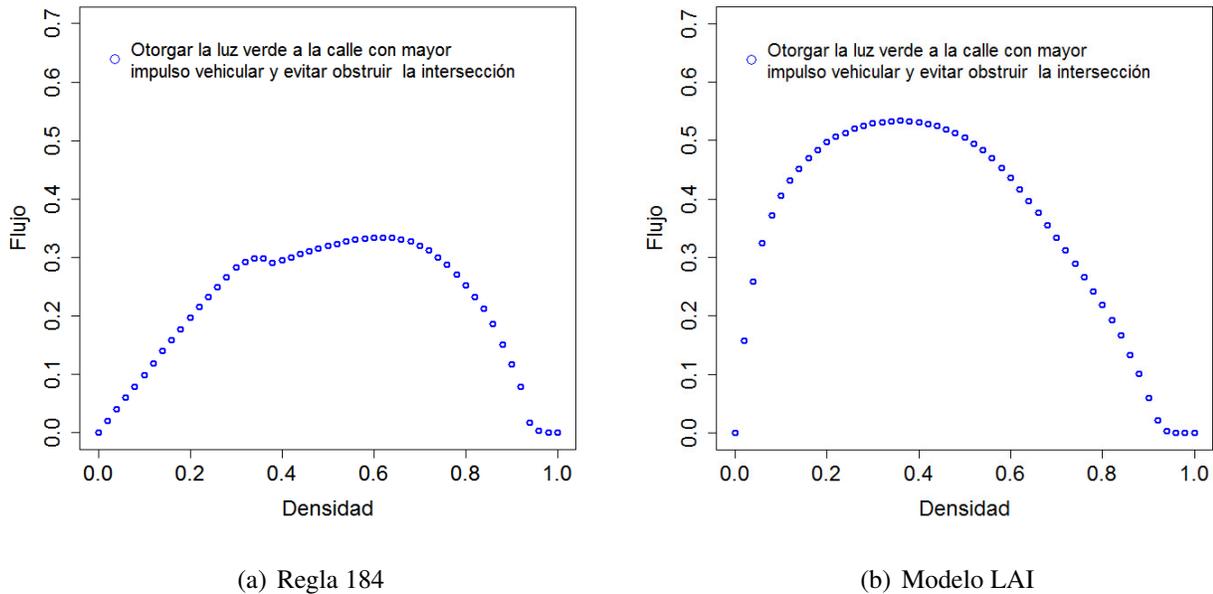


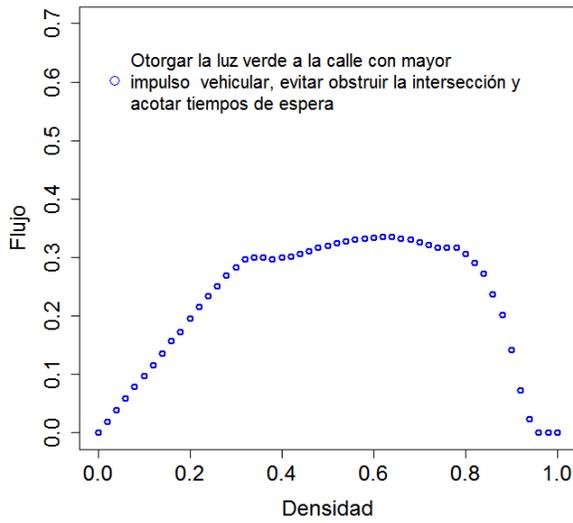
Figura 6.9: La regla *otorgar la luz verde a la calle con mayor impulso vehicular y evitar obstruir la intersección* con la regla 184 y el modelo LAI.

La regla *evitar obstruir la intersección* impide que los vehículos obstruyan las intersecciones. Por esta razón, el flujo vehicular aumentó en comparación con solo aplicar la primera regla. Sin embargo, si únicamente se aplicaban las dos reglas, entonces era posible que los cambios de luz se ejecutaran de inmediato, es decir, sin esperar al menos un tiempo mínimo. Por esta razón, la seguridad de un peatón o ciclista al momento de cruzar la calle no está garantizada.

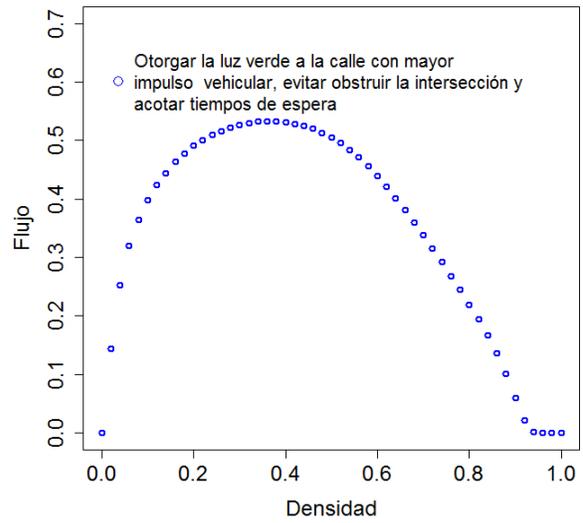
El tercer experimento consistió en determinar el desempeño, en términos del flujo vehicular, de la reglas integradas: *otorgar la luz verde a la calle con mayor impulso vehicular*, *evitar obstruir la intersección* y *acotar tiempos de espera* con la regla 184 y el modelo LAI (ver Figura 6.9). El flujo promedio era $J \approx 0.225$ y $J \approx 0.350$ para la regla 184 y el modelo LAI, respectivamente.

6.6.2. SAR vs SARIV

En esta sección se muestra la comparación entre el SAR y el SARIV (ver Figura 6.11). El flujo promedio del SAR es $J \approx 0.206$ y $J \approx 0.344$ con la regla 184 y el modelo LAI, respectivamente. El flujo promedio del SARIV es $J \approx 0.215$ y $J \approx 0.350$ para la regla 184 y el modelo LAI, respectivamente.

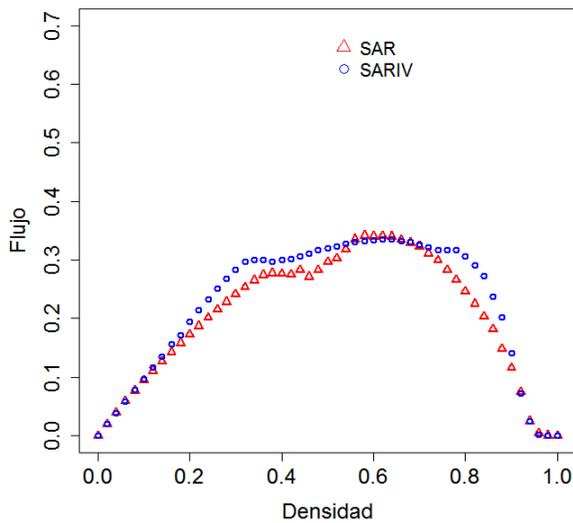


(a) Regla 184

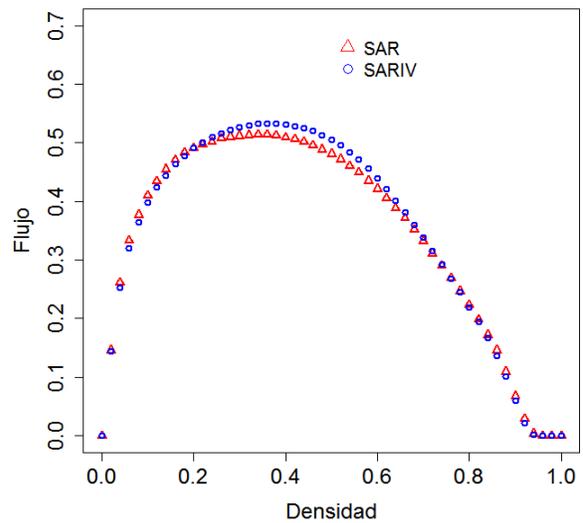


(b) Modelo LAI

Figura 6.10: La regla *otorgar la luz verde a la calle con mayor impulso vehicular, evitar obstruir la intersección y acotar tiempos de espera* con la regla 184 y el modelo LAI.



(a) Regla 184



(b) Modelo LAI

Figura 6.11: Comparación entre el SAR y el SARIV.

El SARIV tiene un desempeño superior al SAR con la regla 184 y con el modelo LAI. Además, las ventajas del sistema SARIV sobre el sistema SAR son las siguientes:

- El SARIV tiene un umbral adaptable, por esta razón, no necesita que previamente se determine el valor del umbral n . Además, no requiere de un umbral m .
- El diseño del SARIV es modular donde cada regla determina completamente cual es su funcionalidad.

Capítulo 7

Comparación entre sistemas que coordinan el tránsito

En este capítulo, se presenta la comparación de cinco sistemas que coordinan el tránsito vehicular en las intersecciones:

- Ola verde (OV). Cambia las luces consecutivamente de acuerdo con el tiempo de viaje esperado en que los vehículos arribarán a las intersecciones.
- Semáforo auto-organizante reactivo (SAR). Otorga la preferencia a la calle que acumula una mayor cantidad de vehículos y tiempo de espera que un *umbral fijo*. Utiliza el *sistema de detección reactivo* y el *sistema de control de Gershenson*.
- Intersección basada en ranura (IBR). Asigna un intervalo de tiempo de acceso a la intersección a cada vehículo autónomo.
- Semáforo auto-organizante reactivo basado en impulso vehicular (SARIV). Otorga la preferencia a la calle que acumula una mayor cantidad de vehículos y tiempo de espera que un *umbral adaptable*. Utiliza el *sistema de detección reactivo* y el *sistema de control basado en impulso vehicular*.
- Semáforo auto-organizante deliberativo basado en impulso vehicular (SADIV). Otorga la preferencia a la calle que acumula una mayor cantidad de vehículos y tiempo de espera que un *umbral adaptable*. Utiliza el *sistema de detección deliberativo* y el *sistema de control basado en impulso vehicular*.

En este trabajo, los sistemas anteriormente mencionados coordinan distintos tipos de tránsito en las intersecciones: tránsito manual, tránsito autónomo y tránsito mixto. El tránsito manual contiene solo vehículos que son conducidos por humanos. El tránsito autónomo contiene solo vehículos autónomos que son conducidos por computadoras. El tránsito mixto está compuesto por vehículos manuales y vehículos autónomos.

7.1. Configuración de los experimentos

El desempeño de los sistemas OV, SAR, IBR, SARIV y SADIV fue evaluado con la regla 184 y el modelo LAI (Lárraga y Álvarez-Icaza) [20]. La ciudad donde se ejecutaron los experimentos estaba compuesta por 100×100 intersecciones, es decir, había 10,000 intersecciones. La distancia de todas las calles era de 3,300 celdas donde cada celda correspondía a $\Delta x = 2.5\text{m}$. Cada cuadra era una distancia de 32 celdas (sin la intersección).

Cada vehículo tenía una longitud $l_s = 2$ (5.0m). El paso de tiempo t se asignó a 1s. En el caso de la regla 184, la velocidad máxima de cada vehículo fue $v_{max} = 1$. En el caso del modelo LAI, la velocidad máxima de cada vehículo fue $v_{max} = 12$, la velocidad baja fue $v_s = 3$, el cambio en la aceleración fue $\Delta v = 1$ y la capacidad de frenar urgentemente fue $M = 2$. Los vehículos no tenían permitido incorporarse a otra calle en las intersecciones; es decir, siempre se desplazaban sobre la misma calle.

Los *vehículos manuales* son operados por humanos. Los valores de los parámetros que simulan el comportamiento de los *vehículos manuales* en el modelo LAI son: $R_d = 1.0$, $R_0 = 0.8$ y $R_s = 0.01$. Dichos valores fueron propuestos por Lárraga y Álvarez-Icaza [20].

Los *vehículos autónomos* son controlados de manera automatizada por una computadora. Los valores de los parámetros que simulan el comportamiento de los vehículos autónomos en el modelo LAI son $R_a = 1.0$ y $R_s = 0.0$. Los vehículos autónomos siempre reaccionan de manera eficiente con dichos valores. Es decir, no tienen retrasos al momento de acelerar y tampoco sobre frenan. Además, si los vehículos autónomos detectan vehículos detenidos a una distancia corta e después de la intersección, entonces se detienen antes de la intersección para evitar bloquearla. La conducción humana implica tiempos de reacción que afectan el flujo de vehicular negativamente. Aplicando automatización y sensores eficaces, el tiempo de reacción humana se puede eliminar y la distancia de seguridad entre vehículos se reduce considerablemente [33]. En este trabajo, no se modificó el modelo LAI para reducir las distancias de seguridad cuando se implementan vehículos autónomos.

Los experimentos consistieron en ejecutar la simulación en 50 densidades distintas para cada sistema de coordinación vehicular. En cada densidad se ejecutaron 20 simulaciones y los vehículos fueron colocados inicialmente en las calles de la ciudad de manera aleatoria. Cada simulación constó de dos fases. En la primera fase se simularon 5, 400 pasos de tiempo (hora y media) para relajar el sistema. En la segunda fase se simularon otros 5, 400 pasos de tiempo donde la velocidad se midió en cada paso de tiempo. Finalmente, para cada densidad se obtuvo el promedio de las 20 simulaciones respecto a la velocidad v y el flujo J .

Los parámetros usados para los sistemas SAR, SARIV y SADIV se muestran en la Tabla 7.1. En el sistema OV, el periodo de de tiempo $T = 64$ pasos de tiempo. Esto implica que las calles en dirección este u oeste tienen una luz verde de 32 pasos de tiempo (32s) y las calles hacia el sur o norte tienen una luz verde para los restantes 32 pasos de tiempo. En el sistema IBR, el valor del parámetro $N = 30$. Por esta razón, se aplica el método BATCH.

Parámetros	SAR	SARIV/SADIV
d	20 celdas (50m)	20 celdas (50m)
r	10 celdas (25m)	-
e	8 celdas (15m)	8 celdas (15m)
u	10 pasos (10s)	10 pasos (10s)
w	60 pasos (60s)	60 pasos (60s)
n	13 <i>veh · pasos</i>	-
m	2 vehículos	-
τ	-	32

Tabla 7.1: Parámetros usados por los sistemas auto-organizantes. El tamaño de la cuadra es de 32 celdas.

El simulador con el que realizamos nuestros experimentos fue desarrollado en el lenguaje C. El lector puede acceder al código en la URL: <https://github.com/Zapotecat1/Traffic-Light>.

7.2. Regla 184

Los resultados de los experimentos en términos del flujo vehicular de los sistemas OV, SAR, SARIV y SADIV aplicando la regla 184 se muestran en la Figura 7.1. En la Tabla 7.2 se muestran los resultados del flujo promedio y máximo de dichos sistemas.

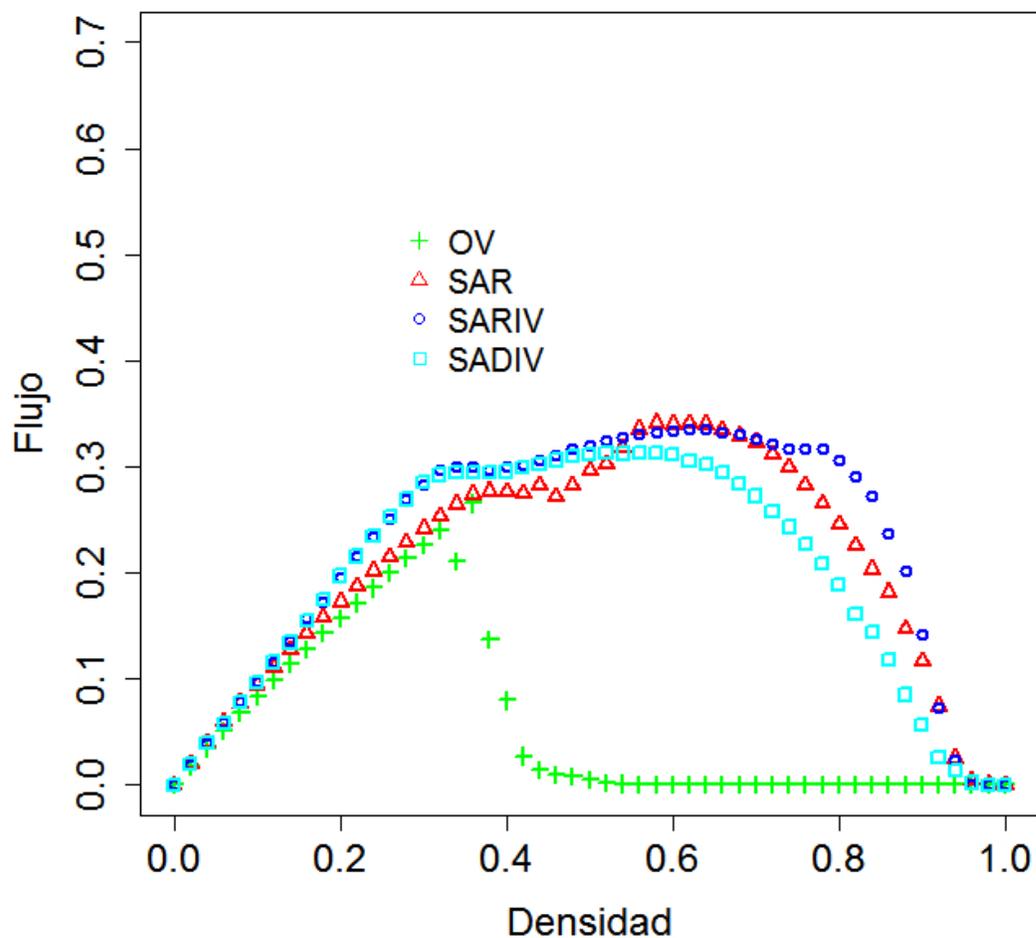


Figura 7.1: Modelo 184 y tránsito manual.

Tabla 7.2: Modelo 184 y tránsito manual.

	Promedio	Máximo	Tiempo en luz verde	Zona de detección
OV	0.056	0.266	32s	0m (0 celdas)
SAR	0.206	0.341	10s-60s	72.5m (29 celdas)
SARIV	0.225	0.334	10s-60s	72.5m (29 celdas)
SADIV	0.197	0.313	10s-60s	2.5m (1 celda)

A partir de la Tabla 7.2 concluimos lo siguiente:

- El OV obtuvo el menor desempeño de todos los sistemas porque tiene un tiempo fijo en luz verde de 32s. El OV no utiliza sensores, por esta razón, la zona de detección es cero.
- El SAR obtuvo el segundo mejor desempeño con $J \approx 0.206$. Los tiempos de luz verde están garantizados entre 10s y 60s. Sin embargo, requiere de una zona de detección de 72.5m.
- El SARIV obtuvo el mejor desempeño con $J \approx 0.225$. Los tiempos de luz verde están garantizados entre 10s y 60s. Sin embargo, al igual que en el SAR, requiere de una zona de detección de 72.5m.
- El SADIV obtuvo el tercer desempeño con $J \approx 0.197$. Sin embargo, el desempeño es competitivo, los tiempos de luz verde están garantizados entre 10s y 60s. Además, requiere de una zona de detección de 2.5m que es la menor de todos los sistemas.

7.3. Modelo LAI

En esta sección los experimentos consistieron en comparar los sistemas OV, SAR, IBR, SARIV y SADIV con tránsito *manual* y *autónomo*.

7.3.1. Sistemas de coordinación con vehículos manuales

Los resultados de los experimentos en términos del flujo vehicular de los sistemas OV, SAR, SARIV y SADIV aplicando el modelo LAI y tránsito manual se muestran en la Figura 7.2. En la Tabla 7.3 se muestran los resultados del flujo promedio y máximo de dichos sistemas.

Tabla 7.3: Modelo LAI y tránsito manual.

	Promedio	Máximo	Tiempo en luz verde	Zona de detección
OV	0.047	0.306	32s	0m (0 celdas)
SAR	0.344	0.513	0s-60s	72.5m (29 celdas)
SARIV	0.350	0.532	10s-60s	72.5m (29 celdas)
SADIV	0.305	0.514	10s-60s	2.5m (1 celda)

A partir de la Tabla 7.3 concluimos lo siguiente:

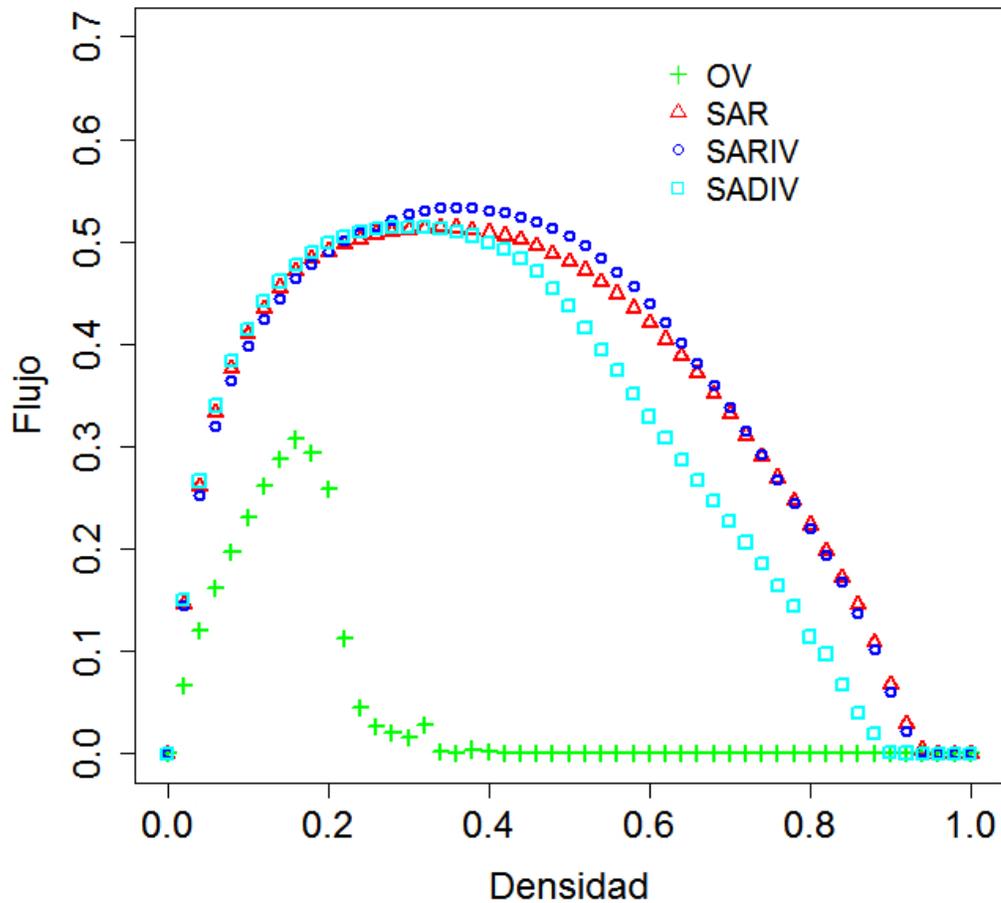


Figura 7.2: Modelo LAI y tránsito manual.

- El OV obtuvo el menor desempeño de todos los sistemas porque tiene un tiempo fijo en luz verde de 32s. Sin embargo, la zona de detección es la menor porque no utiliza sensores. Por esta razón, la zona de detección es cero.
- El SAR obtuvo el segundo desempeño con $J \approx 0.344$. Sin embargo, la zona de detección es la mayor con 72.5m.
- El SARIV obtuvo el mejor desempeño con $J \approx 0.350$. Los tiempos de luz verde están garantizados entre 10s y 60s. Sin embargo, la zona de detección, al igual que en el SAR, es la mayor con 72.5m.

- El SADIV obtuvo el tercer desempeño con $J \approx 0.305$. El desempeño es competitivo en comparación con el SAR y SARIV. Los tiempos de luz verde están garantizados entre 10s y 60s. Además, la zona de detección es la menor con 2.5m.

7.3.2. Sistemas de semáforos con vehículos autónomos

Los resultados de los experimentos en términos del flujo vehicular de los sistemas OV, SAR, IBR, SARIV y SADIV aplicando el modelo LAI y tránsito autónomo se muestran en la Figura 7.3. En la Tabla 7.4 se muestran los resultados del flujo promedio y máximo de los sistemas OV, SAR, IBR, SARIV y SADIV.

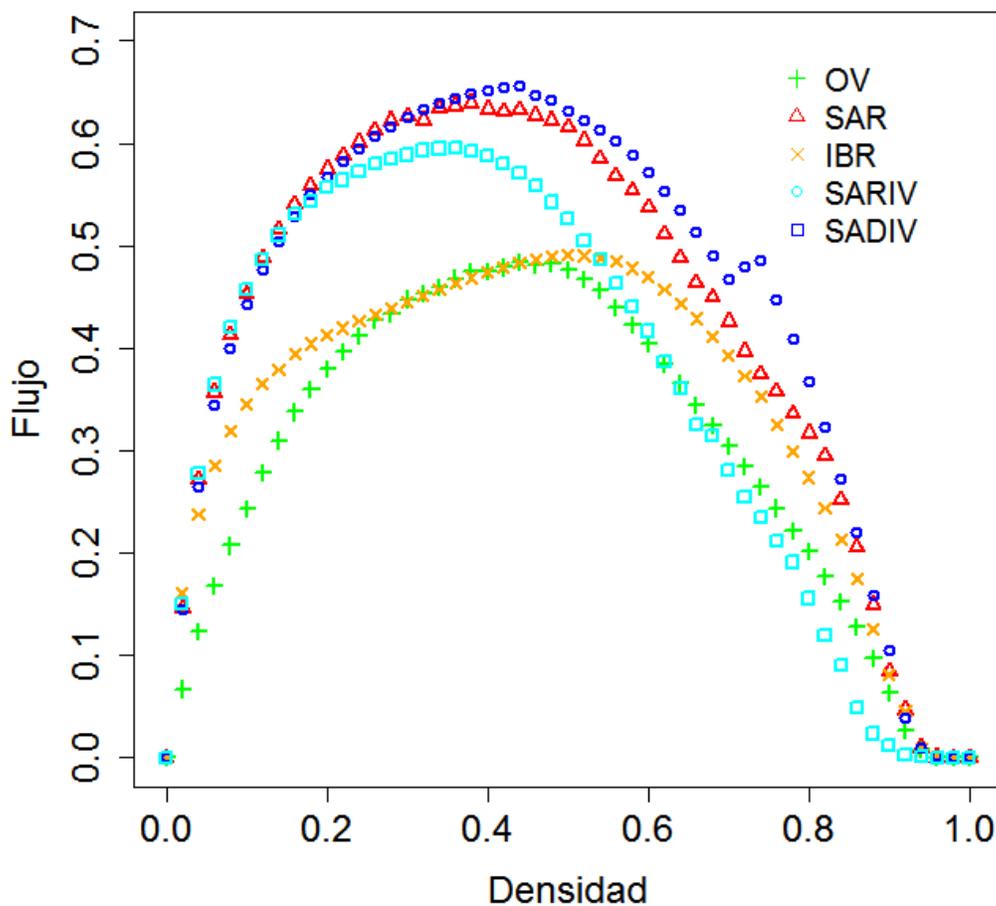


Figura 7.3: Modelo LAI y tránsito autónomo.

Tabla 7.4: Modelo LAI y tránsito autónomo.

	Promedio	Máximo	Tiempo en luz verde	Zona de detección
OV	0.296	0.483	32s	0m (0 celdas)
SAR	0.424	0.639	10s-60s	72.5m (29 celdas)
IBR	0.338	0.490	-	-
SARIV	0.442	0.656	10s-60s	72.5m (29 celdas)
SADIV	0.357	0.595	10s-60s	2.5m (1 celda)

A partir de la Tabla 7.4 concluimos lo siguiente:

- El OV obtuvo el menor desempeño de todos los sistemas porque tiene un tiempo fijo en luz verde de 32s. Sin embargo, la zona de detección es la menor porque no utiliza sensores. Por esta razón, la zona de detección es cero. A diferencia de los vehículos manuales, los vehículos autónomos comienzan a detenerse antes del semáforo si hay vehículos detenidos después de la intersección a una corta distancia e para prevenir bloquear la intersección. Por esta razón, el desempeño de los vehículos autónomos es mayor que los vehículos manuales.
- El SAR obtuvo el segundo mejor desempeño con $J \approx 0.424$. Sin embargo, la zona de detección es la mayor con 72.5m.
- El IBR obtuvo el cuarto desempeño de flujo promedio. Los tiempos de luz no están garantizados dentro de un rango. El sistema IBR está diseñado para que los vehículos autónomos se comuniquen mediante una red inalámbrica con el administrador de la intersección; es decir, no utilizan sensores. Por esta razón, no hay zona de detección.
- El SARIV obtuvo el mejor desempeño con $J \approx 0.442$. Los tiempos de luz verde están garantizados entre 10s y 60s. Sin embargo, la zona de detección, al igual que en el SAR, es la mayor con 72.5m.
- El SADIV obtuvo el tercer desempeño con $J \approx 0.357$. Sin embargo, el desempeño es competitivo, los tiempos de luz verde están garantizados entre 10s y 60s. Además, la zona de detección es la menor con 2.5m.

7.4. Errores en los sensores

En el sistema de detección *reactivo* la función **HayVehículo** se ejecuta k celdas para verificar si hay un vehículo en la zona de detección. En el sistema de detección *deliberativo* la función **HayVehículo** se ejecuta 1 vez. Para cada celda, el sistema de detección (reactivo y deliberativo) tiene asociada una probabilidad P que representa la precisión para detectar vehículos. Por ejemplo: si $P = 0.9$, entonces el sensor tiene una precisión del 90% de detectar vehículos. Dicha precisión está implementada dentro de la función **HayVehículo**. Además, un vehículo permanecerá imperceptible durante toda su trayectoria en la zona de detección desde momento que el sensor no lo detectó.

Los experimentos de esta sección consistieron en ejecutar la simulación en una precisión de $P = 0.9$ para evaluar la robustez de los sistemas SAR, SARIV y SADIV ante errores de detección.

Los resultados aplicando la regla 184 se muestran en la Figura 7.4 y en la Tabla 7.5. El SADIV obtuvo el mejor desempeño con $J \approx 0.078$.

Tabla 7.5: Precisión del sensor $P = 0.9$, modelo 184 y tránsito manual.

	Promedio	Máximo	Zona de detección
SAR	0.058	0.222	72.5m (29 celdas)
SARIV	0.055	0.211	72.5m (29 celdas)
SADIV	0.080	0.286	2.5m (1 celda)

Los resultados aplicando el modelo LAI con tránsito manual se muestran en la Figura 7.5 y en la Tabla 7.6. El SADIV obtuvo el mejor desempeño con $J \approx 0.108$.

Tabla 7.6: Precisión del sensor $P = 0.9$, modelo LAI y tránsito manual.

	Promedio	Máximo	Zona de detección
SAR	0.090	0.450	72.5m (29 celdas)
SARIV	0.086	0.437	72.5m (29 celdas)
SADIV	0.127	0.495	2.5m (1 celda)

Los resultados aplicando el modelo LAI se muestran en la Figura 7.6 y en la Tabla 7.7. El SADIV obtuvo el mejor desempeño con $J \approx 0.267$. Los vehículos autónomos comienzan a detenerse antes del semáforo si hay vehículos detenidos después de la intersección a una corta distancia e para prevenir bloquear la intersección.

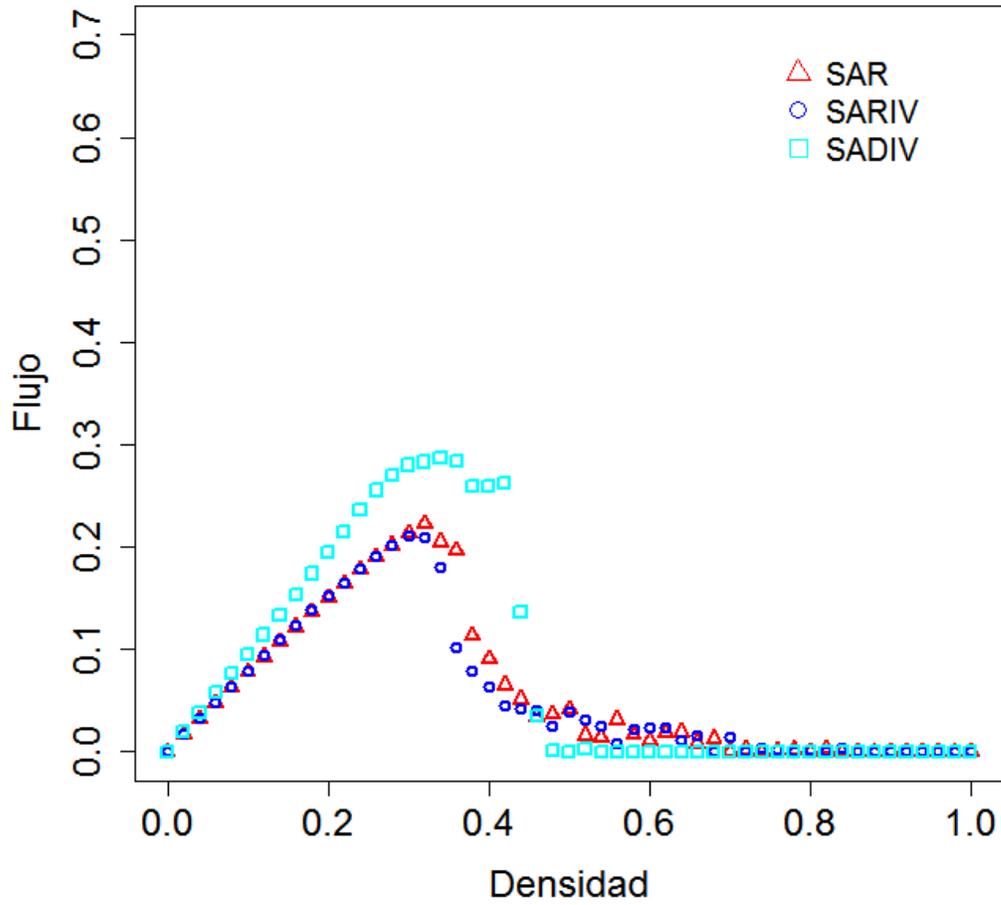


Figura 7.4: Precisión del sensor $P = 0.9$, regla 184.

Tabla 7.7: Modelo LAI con tránsito autónomo.

	Promedio	Máximo	Zona de detección
SAR	0.117	0.551	72.5m (29 celdas)
SARIV	0.105	0.515	72.5m (29 celdas)
SADIV	0.267	0.610	2.5m (1 celda)

El SADIV tiene un desempeño mayor que el SAR y SARIV. Es decir, el SADIV es más robusto ante los errores de detección. Por un lado, el SADIV mantiene el estado del tránsito en memoria y el error de detección solo afecta a una celda. El SADIV mediante el entorno virtual adquiere información

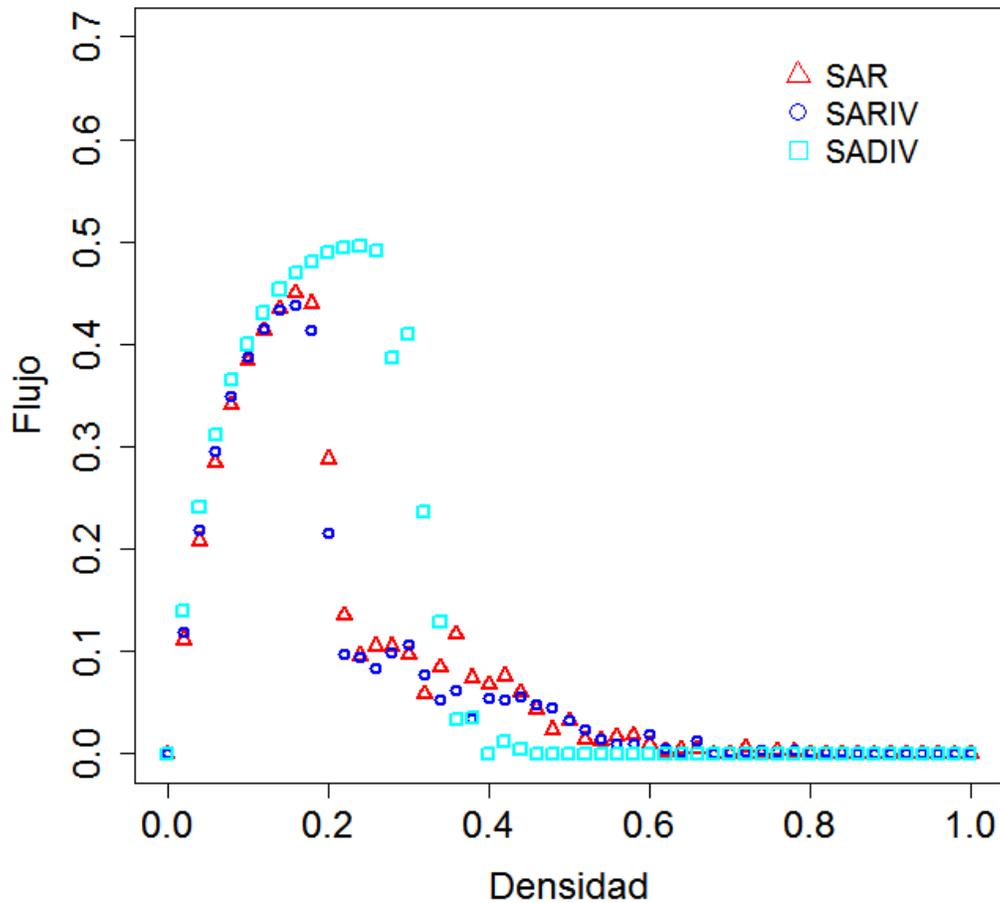


Figura 7.5: Precisión del sensor $P = 0.9$, modelo LAI y tránsito manual.

de donde se encuentran posiblemente los vehículos. Por otro lado, con el SAR y SARIV el estado del tránsito no se mantiene en memoria y el error se distribuye en todas las celdas porque es inherente a cada celda de la zona de detección. Por esta razón, el sistema reactivo no tiene forma de recuperarse.

7.5. Ciudad de Manhattan con tránsito mixto

En esta sección, se presentan los resultados de la simulación obtenidos cuando los vehículos autónomos coexisten con vehículos manuales (operados por humanos). El experimento consiste en reemplazar gradualmente vehículos autónomos por vehículos manuales. El conjunto de experimentos es

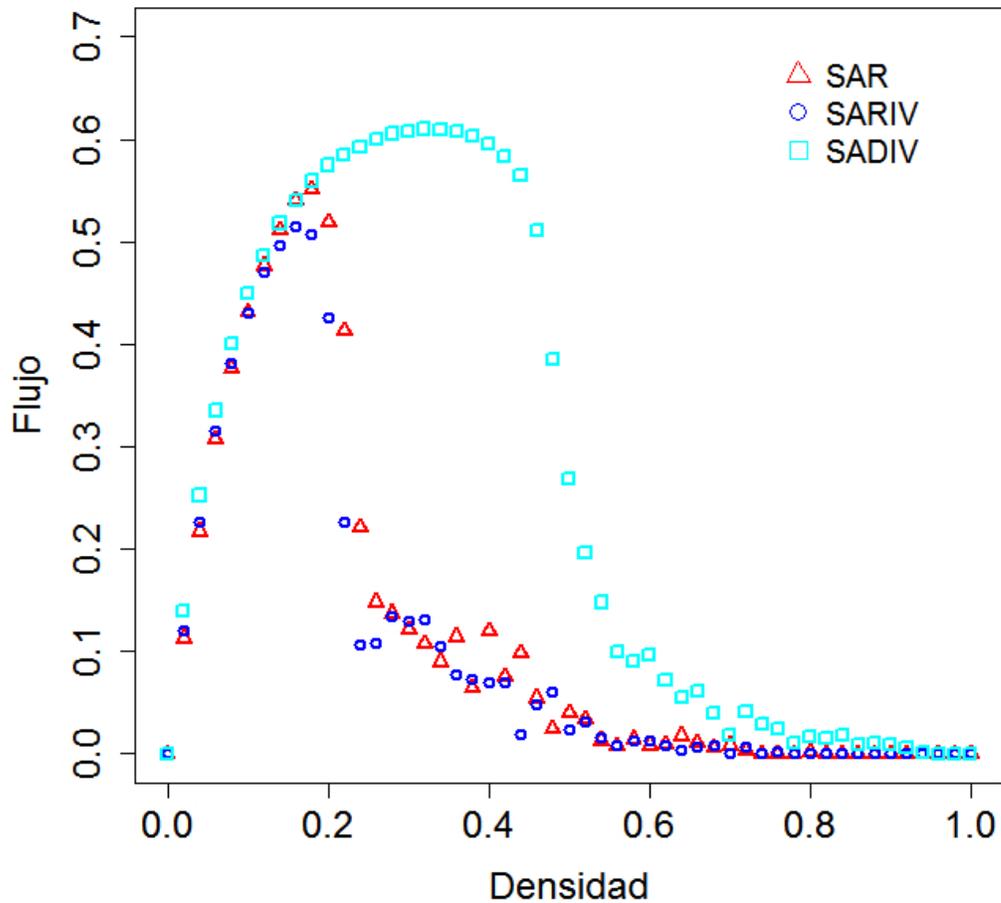


Figura 7.6: Precisión del sensor $P = 0.9$, modelo LAI y tránsito autónomo.

obtenido al variar la densidad y el porcentaje de vehículos autónomos. Las densidades son variadas en 50 diferentes valores $\rho = 0, \rho = 0.02, \rho = 0.04, \dots, \rho = 1.0$ y en la misma manera, el porcentaje de vehículos autónomos es variado en 50 diferentes valores $\phi = 0\%, \phi = 2\%, \phi = 4\%, \phi = 6\%, \dots, \phi = 100\%$. En cada densidad, diez corridas fueron ejecutadas y los resultados se promediaron.

Por ejemplo, en el primer conjunto de experimentos, en ninguna de las 50 diferentes densidades se ingresan vehículos autónomos. En el segundo conjunto de experimentos, en cada una de las 50 densidades, hay un 2% de vehículos autónomos y un 98% de vehículos manuales. En el tercer con-

junto de experimentos, hay un 4 % de vehículos autónomos y un 96 % de vehículos manuales, y así sucesivamente. En el conjunto de experimentos finales, el 100 % de los vehículos son autónomos.

Para el OV, en la Figura 7.7 se muestra el efecto que tendría el flujo vehicular y la velocidad al momento de remplazar gradualmente vehículos manuales por vehículos autónomos que aplican la regla que busca evitar bloquear la intersección. Como se puede apreciar, el flujo vehicular aumenta en la medida que se agregan más vehículos autónomos.

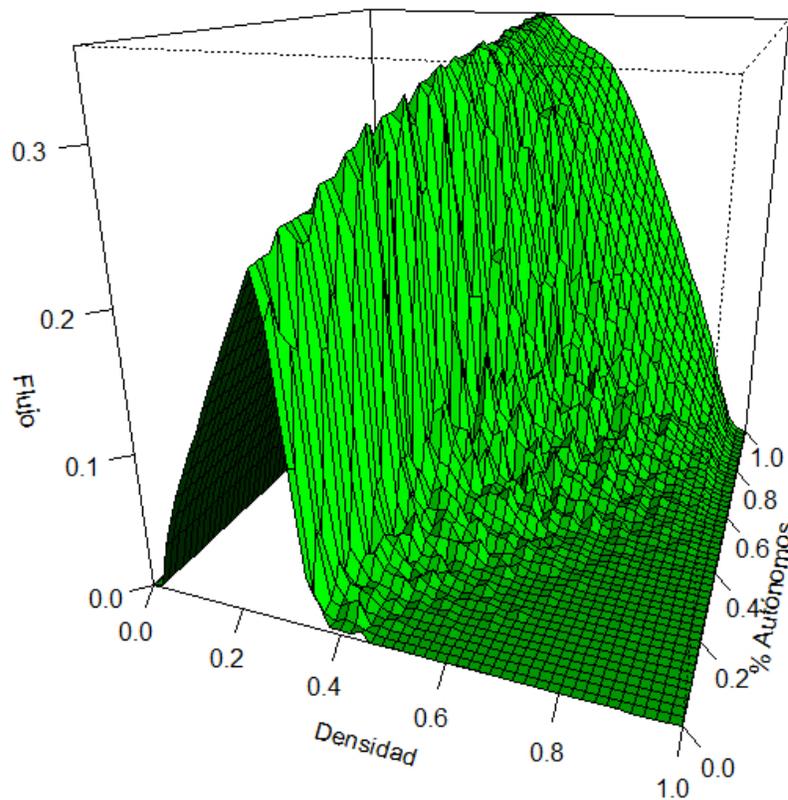


Figura 7.7: El eje X representa la densidad, el eje Y el porcentaje de vehículos autónomos y el eje Z el flujo.

Para el SAR, en la Figura 7.8 se muestra el efecto que tendría el flujo vehicular y la velocidad al momento de reemplazar gradualmente vehículos manuales por vehículos autónomos. Como se puede apreciar, el flujo vehicular aumenta en la medida que se agregan más vehículos autónomos. El SAR es superior al OV en todas las densidades.

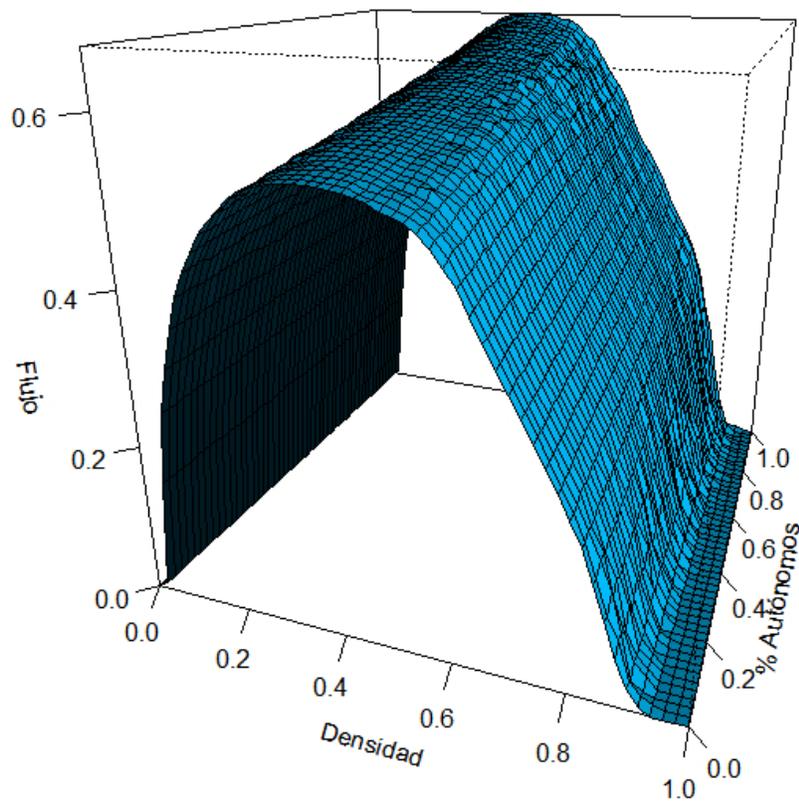


Figura 7.8: El eje *X* representa la densidad, el eje *Y* el porcentaje de vehículos autónomos y el eje *Z* el flujo.

Capítulo 8

Conclusiones y trabajo futuro

Todas las grandes ciudades sufren congestionamientos de tránsito. Las mejoras tecnológicas como las presentadas en este trabajo pueden aumentar la capacidad de la infraestructura urbana. Sin embargo, si la eficiencia de viajar en vehículos privados se incrementa, más personas estarán dispuestas a utilizar vehículos, aumentando la densidad y como consecuencia disminuirá el flujo vehicular. Esto significa que las mejoras en la infraestructura de tránsito pueden generar una situación peor que la que están tratando de resolver [34, 35]. Por esta razón, se requiere un plan de transporte integrado para equilibrar las demandas y mejoras en todos los modos de transporte.

Está claro que los algoritmos adaptativos superarán a los métodos estáticos tradicionales [36], como se mostró en este trabajo. Es importante desarrollar sistemas robustos y de bajo costo. El sistema de detección deliberativo tiene un desempeño competitivo en comparación con sistemas de detección tradicionales, pero a un costo económico potencialmente menor. Además, el sistema de detección deliberativo es más robusto si se presentan fallas de detección.

El sistema de control basado en impulso vehicular se adapta a las condiciones del tránsito vehicular mediante un umbral adaptable que indica los cambios de luz. El sistema toma en cuenta si un vehículo pertenece o no a un pelotón. Por un lado, el sistema promueve una estrategia de uno por uno en densidades bajas porque no retrasa los cambios de luz. Por otro lado, promueve la formación de pelotones en densidades medias porque retrasa los cambios de luz. El sistema de control tiene el mejor desempeño en términos de flujo vehicular que los sistemas OV, SAR e IBR.

Considerando los recientes avances en el desarrollo de vehículos autónomos [37], los sistemas descentralizados como los presentados aquí serán útiles en la coordinación de vehículos autónomos en las intersecciones. Al utilizar vehículos autónomos que tienen la capacidad de reaccionar de manera

eficiente en la aceleración y el frenado, el flujo vehicular se incrementa porque los congestionamientos fantasma se reducen. Además, si se aplica un sistema adaptativo en las intersecciones el desempeño del flujo vehicular mejora considerablemente.

Es importante considerar que antes de que los vehículos autónomos sean utilizados completamente en las ciudades, habrá un periodo de transición donde vehículos autónomos coexistirán con vehículos operados por humanos. En este trabajo mostramos que al reemplazar gradualmente la cantidad de vehículos autónomos en una ciudad donde los semáforos son controlados por los sistemas *ola verde* o *auto-organizante reactivo* el flujo vehicular se incrementa.

Este trabajo contribuye con el desarrollado de un simulador basado en autómatas celulares. El lector puede acceder al código en la URL: <https://github.com/Zapotecat1/Traffic-Light>.

La primera línea de continuación de este trabajo de investigación es desarrollar una ciudad tipo Manhattan que contenga varios carriles e intersecciones con múltiples calles. En dicha ciudad tipo Manhattan será necesario realizar las siguientes extensiones al modelo LAI y a los semáforos auto-organizantes:

- Los comportamientos de conducción de los vehículos tendrán que extenderse para soportar cambios de carril de manera segura en el modelo LAI.
- Los sistemas de detección y de control tendrán que extenderse para soportar múltiples calles conectadas a una intersección.

Bibliografía

- [1] Vincent, R. A. and Young, C. P. Self optimising traffic signal control using microprocessors—the TRRL MOVA strategy for isolated intersections. *Traffic Engineering and Control*, 27(7-8):385–387, 1986.
- [2] Mirchandani, P. and Head, L. A real-time traffic signal control system: Architecture, algorithms, and analysis. *Transportation Research Part C: Emerging Technologies*, 9 (6):415–432, 2001.
- [3] Smith, S. F., Barlow, G. J., Xie X. F., and Rubinstein, Z. B. Smart urban signal networks: Initial application of the SURTRAC adaptive traffic signal control system. In *Proceedings of the Twenty-Third International Conference on Automated Planning and Scheduling, ICAPS*, 2013.
- [4] Gershenson, C. and Rosenblueth, D. A. Adaptive self-organization vs static optimization: A qualitative comparison in traffic light coordination. *Kybernetes*, 41(3–4):386–403, 2012.
- [5] Campbell, M., Egerstedt, M., How, J. P., and Murray, R. M. Autonomous driving in urban environments: approaches, lessons and challenges. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 368(1928):4649–4672, 2010.
- [6] Prigogine, I. and Herman, R. *Kinetic Theory of Vehicular Traffic*. Elsevier, Nueva York, 1971.
- [7] Wolf, D. E., Schreckenberg, M., and Bachem. *Traffic and Granular Flow '95*. World Scientific, Singapur, 1996.
- [8] Helbing, D. and Huberman, B. A. Coherent moving states in highway traffic. *Nature*, 396:738–740, 1998.
- [9] Chowdhury, D., Santen, L., and Schadschneider, A. Statistical physics of vehicular traffic and some related systems. *Physics Reports*, 329 (4-6):199–329, 2000.

- [10] Maerivoet, S. and Moor, B. Cellular automata models of road traffic. *Physics Reports*, 419 (1):1–64, Noviembre 2005.
- [11] Yukawa, S., Kikuchi, M., and Tadaki, S. Dynamical phase transition in one dimensional traffic flow model with blockage. *Journal of the Physical Society of Japan*, 63 (10):3609–3618, 1994.
- [12] Sokolowski, J. and Banks, C. *Principles of Modeling and Simulation: A Multidisciplinary Approach*. John Wiley and Sons, 2 2009. An optional note.
- [13] Nagel, K. and Schreckenberg, M. A cellular automaton model for freeway traffic. *Journal of Physics I France*, 2:2221–2229, 1992.
- [14] Nagel, K. and Paczuski, M. Emergent traffic jams. *Physical Review E*, 51 (4):2909–2918, 1995.
- [15] Biham, O., Middleton, A. A., and Levine, D. Self organization and a dynamical transition in traffic flow models. *Physical Review A*, 46 (10):R6124–R6217, 1992.
- [16] Simon, P. M. and Nagel, K. A simplified cellular automaton model for city traffic. Technical report, Tech. Rep. LA-UR 97-707, LANL, 1997.
- [17] Chowdhury, D. and Schadschneider, A. Self-organization of traffic jams in cities: Effects of stochastic dynamics and signal periods. *Physical Review E*, 59 (2):R1311–R1314, 1999.
- [18] Schadschneider, A., Chowdhury, D., Brockfeld, E., Klauck, K., Santen, L., and Zittartz, J. A new cellular automata model for city traffic. In *Proc. Traffic and Granular Flow '99: Social, Traffic, and Granular Dynamics*, pages 437–442, 1999.
- [19] Brockfeld, E., Barlovic, R., Schadschneider, A., and Schreckenberg, M. Optimizing traffic lights in a cellular automaton model for city traffic. *Physical Review E*, 64:056132, 2001.
- [20] Lárraga M. E. and Alvarez-Icaza L. Cellular automaton model for traffic flow based on safe driving policies and human reactions. *Physica A*, 389:5425–5438, 2010.
- [21] Lämmer, S. and Helbing, D. Self-control of traffic lights and vehicle flows in urban road networks. *Journal of Statistical Mechanics: Theory and Experiment*, (4), 2008.
- [22] Gier, J., Garoni, T. M., and Rojas, O. Traffic flow on realistic road networks with adaptive traffic lights. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(4), 2011.

- [23] Goel, S., Bush, S.F., and Ravindranathan, K. Self-organization of traffic lights for minimizing vehicle delay. In *Connected Vehicles and Expo (ICCVE), 2014 International Conference on*, pages 931–936, Noviembre 2014.
- [24] Cesme, B. and Furth, P. G. Self-organizing traffic signals using secondary extension and dynamic coordination. *Transportation Research Part C: Emerging Technologies*, 48:1–15, 2014.
- [25] Török, J. and Kertész, J. The green wave model of two-dimensional traffic: Transitions in the flow properties and in the geometry of the traffic jam. *Physica A*, 231 (4):515–533, 1996.
- [26] Dresner, K. and Stone, P. A multiagent approach to autonomous intersection management. *Journal of Artificial Intelligence Research*, 31:591–656, 2008.
- [27] Yao-Ming, Y., Rui, J., Mao-Bin, H., Qing-Song, W., and Ruili W. Traffic flow characteristics in a mixed traffic system consisting of ACC vehicles and manual vehicles: A hybrid modelling approach. *Physica A: Statistical Mechanics and its Applications*, 388(12):2483 – 2491, 2009.
- [28] Tsz-Chiu, A., Zhang, S., and Stone, P. Semi-autonomous intersection management. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems, AAMAS '14*, pages 1451–1452, Richland, SC, 2014. International Foundation for Autonomous Agents and Multiagent Systems.
- [29] Tachet, R., Santi, P., Sobolevsky, S., Reyes-Castro, L. I., Frazzoli, E., Helbing, D., and Ratti, C. Revisiting street intersections using slot-based systems. *PLoS ONE*, 11(3):1–9, Marzo 2016.
- [30] Wahle, J., Esser, J., Neubert, L., and Schreckenberg, M. *A Cellular Automaton Traffic Flow Model for Online-Simulation of Urban Traffic*, pages 185–193. Springer, Londres, 1998.
- [31] Wahle, J., Neubert, L., Esser, J., and Schreckenberg, M. A cellular automaton traffic flow model for online simulation of traffic. *Parallel Computing*, 27(5):719–735, 2001.
- [32] Hoare, C. A. R. An axiomatic basis for computer programming. *Commun. ACM*, 12(10):576–580, October 1969.
- [33] Petros, A. I. and Chien, C. V. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):1–9, 1993.

- [34] Braess, D., Nagurney, A., and Wakolbinger, T. On a paradox of traffic planning. *Transportation Science*, 39(4):446–450, Noviembre 2005.
- [35] Steinberg, R. and Zangwill, W. I. The prevalence of Braess' paradox. *Transportation Science*, 17(3):301–318, 1983.
- [36] Gershenson, C. Living in living cities. *Artificial Life*, 19(3 - 4):401–420, Verano/Otoño 2013.
- [37] R. Bishop. A survey of intelligent vehicle applications worldwide. In *Intelligent Vehicles Symposium IV, Proceedings of the IEEE*, pages 25–30, 2000.