

UNIVERSIDADE DA CORUÑA

DEPARTAMENTO DE ECONOMÍA APLICADA II

TESIS DOCTORAL

**MODELOS Y ALGORITMOS
SOLUCIÓN PARA UN PROBLEMA
DE CONTROL ÓPTIMO DE
SEMÁFOROS. APLICACIÓN A
CRUCES CON 2, 3, 4 o 6 FASES EN
LA CIUDAD DE A CORUÑA**

CARMEN SOCORRO LEMA FERNÁNDEZ

DIRECTOR: LUÍS PEDRO PEDREIRA ANDRADE

A Coruña, 2012

Don Luís Pedro Pedreira Andrade, catedrático del Departamento de Economía Aplicada II de la Universidade da Coruña, informa que la memoria titulada:

MODELOS Y ALGORITMOS SOLUCIÓN PARA UN PROBLEMA DE CONTROL ÓPTIMO DE SEMÁFOROS. APLICACIÓN A CRUCES CON 2, 3, 4 o 6 FASES EN LA CIUDAD DE A CORUÑA

fue realizada bajo su dirección por Doña Carmen Socorro Lema Fernández, estimando que la interesada se encuentra en condiciones de optar al grado de Doctor por la Universidade da Coruña, por lo que solicita que sea admitida a trámite para su lectura y defensa pública.

En A Coruña, a 3 de mayo de 2012.

El Director:

Prof. Dr. Luís Pedro Pedreira Andrade

La doctoranda: Carmen Socorro Lema Fernández

Agradecimientos

La elaboración de este proyecto fue un largo proceso que seguramente no hubiera podido realizar sin la participación de personas e instituciones que han facilitado las cosas para que este trabajo llegue a feliz término. Por ello, es para mí un verdadero placer utilizar este espacio para ser consecuente con ellas expresándoles mi agradecimiento.

En primer lugar, quiero dar las gracias al director de esta tesis, profesor Luís Pedro Pedreira Andrade. Su apoyo y confianza en mi trabajo y su capacidad para guiar mis ideas han sido un aporte fundamental e insustituible, no solamente en el desarrollo de este proyecto, sino también en mi formación como investigadora. Siempre me dio muchísimo ánimo, al igual que lo hizo su esposa Maribel (gracias Maribel), por lo que me facilitó enormemente el trabajo.

En segundo lugar quiero agradecerle a las profesoras del departamento de Matemáticas y Computación de la Universidad de La Habana, Sira Allende Alonso y Gemayqzel Bouza Allende, su apoyo y ayuda incondicional en la elaboración de esta tesis, tanto durante sus estancias en la Universidade da Coruña (gracias también a esta institución por haberlas financiado) como a través de e-mail. Yo las considero codirectoras de este trabajo, aunque por razones burocráticas, oficialmente no lo sean.

Mi gratitud a todos mis compañeros de la sección de Matemáticas, por todos los consejos y orientaciones que me han resultado de gran provecho, por el buen ambiente en el que se trabaja y por las veces que han realizado algunas de mis tareas docentes para que mi dedicación a la investigación fuese mayor. Igualmente mi agradecimiento a los demás compañeros del departamento de Economía Aplicada II y de la facultad de Economía y Empresa que mostraron su cariño e interés durante este tiempo.

También quiero agradecer a los miembros del tribunal el tiempo dedicado a leer y valorar este trabajo.

Y por supuesto, el agradecimiento más profundo y sentido va por mi familia, mi es-

poso José Luís, mis hijos Lucía, Marcos, Cristina y Miguel, sin su apoyo, colaboración, paciencia e inspiración habría sido imposible llevar a cabo esta dura empresa. También agradezco a mi hermana Loli, todas las horas que con cariño, dedicación y paciencia ha dedicado al cuidado de mis hijos. Un recuerdo para mis padres, ya fallecidos, Antonio y Lola, a quien dedico este trabajo, por su esmerada dedicación a la educación y cuidado de sus hijas y nietos.

Finalmente doy las gracias a la Universidade da Coruña, institución a la que estoy orgullosa de pertenecer como P.D.I. desde su fundación, por haberme liberado de mis tareas docentes durante un cuatrimestre (a través del programa de promoción de profesores Titulares de Escuela Universitaria y Colaboradores) para poder dedicarme íntegramente a la elaboración de esta tesis.

Resumo

O obxectivo deste traballo é resolver un problema de control óptimo que consiste en asignar tempos a luces de semáforos que regulan cruces de dúas rúas cos dous sentidos de circulación, e aplicar os resultados a cruzamentos (con conxestións frecuentes) da cidade da Coruña. O criterio de comparación exprésase nunha función obxectivo a minimizar que pode ser: lonxitude media da cola no carril con maiores colas, lonxitude da cola máis longa, tempo medio de espera no carril con maior tempo de espera, unha combinación das anteriores, entre outras posibilidades. Desta forma obtense un modelo tipo problema de optimización con restricións de complementariedade linear. Analizaremos: o caso estático, é dicir, o modelo no cal os tempos das luces son variables, pero só dependen dos fluxos de chegada e o caso dinámico, é dicir, considérase que no semáforo hai sensores que permiten coñecer a cantidade de autos en cada carril e, entón, pódese decidir se, dado o tráfico, mantense o estado actual, por canto tempo, ou se debe cambiar a luz. Para a súa solución, propoñemos un método híbrido que combina unha heurística, tipo recocido simulado, para calcular unha boa aproximación á solución, que logo será mellorada cun algoritmo quasi-Newton para problemas de optimización non suaves. Exporemos os resultados numéricos obtidos a partir da programación dos algoritmos anteriores en MATLAB.

Resumen

El objetivo de este trabajo es resolver un problema de control óptimo que consiste en asignar tiempos a luces de semáforos que regulan cruces de dos calles con los dos sentidos de circulación, y aplicar los resultados a cruces (con congestiones frecuentes) de la ciudad de A Coruña. El criterio de comparación se expresa en una función objetivo a minimizar que puede ser: longitud media de la cola en el carril con mayores colas, longitud de la cola más larga, tiempo medio de espera en el carril con mayor tiempo de espera, una combinación de las anteriores, entre otras posibilidades. De esta forma se obtiene un modelo tipo problema de optimización con restricciones de complementariedad lineal. Analizaremos: el caso estático, es decir, el modelo en el cual los tiempos de las luces son variables, pero sólo dependen de los flujos de llegada y el caso dinámico, es decir, se considera que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz. Para su solución, proponemos un método híbrido que combina una heurística, tipo recocido simulado, para calcular una buena aproximación a la solución, que luego será mejorada con un algoritmo quasi-Newton para problemas de optimización no suaves. Expondremos los resultados numéricos obtenidos a partir de la programación de los algoritmos anteriores en MATLAB.

Abstract

The objective of this thesis is to solve the problem of finding the switch times of traffic lights that regulate two two-way traffic intersections, and to then apply these results to intersections in the city of Corunna which suffer from frequent congestions. The criterion used for comparison is shown by minimizing the objective function which can be by: the average length of the queue in the lane with the worst queues, the length of the longest queue, the average wait time in the lane with the longest wait time, a combination of all of the above. The resulting model is an optimization problem with linear complementarity constraints. We will analyze the following: the static case, that is, the model in which the times of the lights are variable, but only depend on the arrival flows, and then the dynamic case, that is, there are sensors at all traffic lights that allow us to know the number of cars in each lane and therefore decide if, given the traffic, maintain the present state and for how long or decide if the light should change. In order to solve this problem, we propose a hybrid solution strategy by which an approximation to the solution is computed by means of a simulated annealing algorithm and then it is improved by a quasi-Newton method for non-smooth optimization problems. We will also present some numerical results obtained from the programming of the previous algorithms in MATLAB

Índice general

Introducción	1
1. Preliminares	5
1.1. Estado del arte	5
1.1.1. Servicios avanzados de información a los conductores (ATIS)	6
1.1.2. Sistemas avanzados de gestión del tráfico (ATMS)	7
1.1.3. Sistemas avanzados de optimización del tráfico (ATOS)	8
1.1.3.1. Optimización de una sola intersección	9
1.1.3.2. Lógica difusa	9
1.1.3.3. Redes de Petri	10
1.1.3.4. Redes Neuronales	10
1.1.3.5. Aproximaciones híbridas	11
1.2. Problema a investigar y metodología utilizada	14
2. Fundamentos teóricos	17
2.1. Introducción	17
2.2. Los problemas de complementariedad lineal	19
2.2.1. Introducción. De KKT a los LCP	19
2.2.2. El problema de complementariedad lineal y algunas generaliza- ciones	20

2.2.3.	El problema de complementariedad lineal extendido (ELCP)	21
2.3.	Método del gradiente discreto	29
2.4.	Métodos heurísticos	32
2.4.1.	Ventajas e inconvenientes de la utilización de métodos heurísticos	32
2.5.	Metaheurísticas: Definición y clasificación	34
2.6.	Recocido simulado (Simulated annealing)	36
2.6.1.	Metaheurística basada en Recocido Simulado	37
2.6.1.1.	Generación de la solución inicial	38
2.6.1.2.	Determinación de la temperatura inicial	39
2.6.1.3.	Condición de equilibrio y esquema de enfriamiento	39
2.6.1.4.	El criterio de parada (criterio de congelación)	40
2.6.1.5.	Definición del entorno de solución	40
3.	Modelos para cruce simple (2 fases)	43
3.1.	Antecedentes	43
3.1.1.	Introducción	43
3.1.2.	El ejemplo y su modelización	44
3.1.2.1.	El ejemplo	44
3.1.2.2.	El modelo exacto	46
3.1.2.3.	Un modelo simplificado	47
3.1.3.	Control óptimo	49
3.1.3.1.	Exposición del problema	49
3.1.3.2.	El problema aproximado relajado (PAR) y las soluciones subóptimas	55
3.1.4.	Modelo con restricciones de complementariedad. Propuesta de un método híbrido de solución	60
3.2.	Modelos estático y dinámico	62

3.2.1. Introducción	62
3.2.2. Modelo estático	62
3.2.3. Modelo dinámico	67
3.2.4. Algoritmos solución	68
3.2.5. Resultados computacionales	71
4. Modelos para cruce de tres fases	85
4.1. Modelo estático	85
4.1.1. Algoritmo solución	91
4.1.2. Aplicación al cruce de la avenida de Finisterre con las calles Pa- lomar y Paseo de los Puentes	92
4.1.3. Resultados computacionales	93
4.2. Modelo dinámico	99
5. Modelos para cruce de cuatro fases	101
5.1. Modelo estático para cuatro fases con cuatro semáforos	101
5.1.1. Algoritmo solución	108
5.1.2. Aplicación al cruce de la avenida de Finisterre con la Ronda de Nelle	109
5.1.3. Resultados computacionales	111
5.2. Modelos estático y dinámico para cuatro fases con ocho semáforos . . .	115
5.2.1. Modelo estático	115
5.2.2. Modelo dinámico	122
5.2.3. Resultados computacionales	123
6. Modelos para cruce de seis fases	143
6.1. Modelos estático y dinámico para seis fases con ocho semáforos	143
6.1.1. Modelo estático	143

6.1.2. Modelo dinámico	153
6.1.3. Resultados computacionales	154
6.1.4. Aplicación al cruce de la avenida de Arteixo con la Ronda de Outeiro	157
6.1.5. Resultados computacionales	160
Conclusiones	163
Bibliografía	167

Índice de figuras

3.1. Cruce simple	45
3.2. 1ª fase del modelo de dos fases	63
3.3. 2ª fase del modelo de dos fases	63
4.1. 1ª fase del modelo de tres fases	86
4.2. 2ª fase del modelo de tres fases	86
4.3. 3ª fase del modelo de tres fases	87
4.4. Cruce de la avda. de Finisterre con calle Palomar y Paseo de los Puentes	92
4.5. Modelo estático de tres fases: Resultados en el carril uno	97
4.6. Modelo estático de tres fases: Resultados en el carril dos	98
4.7. Modelo estático de tres fases: Resultados en el carril tres	98
4.8. Modelo estático de tres fases: Resultados en el carril cuatro	98
5.1. 1ª fase del modelo de cuatro fases con cuatro semáforos	102
5.2. 2ª fase del modelo de cuatro fases con cuatro semáforos	103
5.3. 3ª fase del modelo de cuatro fases con cuatro semáforos	103
5.4. 4ª fase del modelo de cuatro fases con cuatro semáforos	104
5.5. Cruce de la avda. de Finisterre con la Ronda de Nelle	110
5.6. 1ª fase del modelo de cuatro fases con ocho semáforos	116
5.7. 2ª fase del modelo de cuatro fases con ocho semáforos	116
5.8. 3ª fase del modelo de cuatro fases con ocho semáforos	117

5.9.	4ª fase del modelo de cuatro fases con ocho semáforos	117
6.1.	1ª fase del modelo de seis fases con ocho semáforos	145
6.2.	2ª fase del modelo de seis fases con ocho semáforos	145
6.3.	3ª fase del modelo de seis fases con ocho semáforos	146
6.4.	4ª fase del modelo de seis fases con ocho semáforos	146
6.5.	5ª fase del modelo de seis fases con ocho semáforos	147
6.6.	6ª fase del modelo de seis fases con ocho semáforos	147
6.7.	Cruce de la avda. de Arteixo con la Ronda de Outeiro	158

Índice de tablas

2.1. Simulación termodinámica y optimización combinatoria	37
3.1. Esquema de cambios (switching) en los semáforos	45
3.2. Parámetros usados en el ejemplo del modelo de dos fases	71
3.3. Con $J=J1$. Duración de la luz verde en cada ciclo	72
3.4. Con $J=J1$. Promedio de autos en el momento del cambio de luz	72
3.5. Duración de la luz verde en cada ciclo. Modelo dinámico	73
3.6. Promedio de autos en el momento del cambio de luz. Modelo dinámico	73
3.7. Con $J=J2$. Duración de la luz verde en cada ciclo	74
3.8. Con $J=J2$. Promedio de autos en el momento del cambio de luz	74
3.9. Duración de la luz verde en cada ciclo. Modelo dinámico	74
3.10. Promedio de autos en el momento del cambio de luz. Modelo dinámico	75
3.11. Con $J=J3$. Duración de la luz verde en cada ciclo	75
3.12. Con $J=J3$. Promedio de autos en el momento del cambio de luz	76
3.13. Duración de la luz verde en cada ciclo. Modelo dinámico	76
3.14. Promedio de autos en el momento del cambio de luz. Modelo dinámico	77
3.15. Con $J=J4$. Duración de la luz verde en cada ciclo	77
3.16. Con $J=J4$. Promedio de autos en el momento del cambio de luz	78
3.17. Duración de la luz verde en cada ciclo. Modelo dinámico	78
3.18. Promedio de autos en el momento del cambio de luz. Modelo dinámico	79

3.19. Con J=J5. Duración de la luz verde en cada ciclo	79
3.20. Con J=J5. Promedio de autos en el momento del cambio de luz	80
3.21. Duración de la luz verde en cada ciclo. Modelo dinámico	80
3.22. Promedio de autos en el momento del cambio de luz. Modelo dinámico	81
3.23. Con J=J6. Duración de la luz verde en cada ciclo	81
3.24. Con J=J6. Promedio de autos en el momento del cambio de luz	82
3.25. Duración de la luz verde en cada ciclo. Modelo dinámico	82
3.26. Promedio de autos en el momento del cambio de luz. Modelo dinámico	83
3.27. Tabla comparativa. Resultados modelos estático y dinámico	84
4.1. Promedio de autos en el momento del cambio de luz en ciclo fijo-3 fases	94
4.2. Promedio de autos en el momento del cambio de luz con heurística-3 fases	96
5.1. Promedio de autos en el momento del cambio de luz en ciclo fijo-4 fases-4 semáforos	112
5.2. Promedio de autos en el momento del cambio de luz con heurística-4 fases-4 semáforos	113
5.3. Parámetros usados en el ejemplo del modelo de cuatro fases con ocho semáforos	124
5.4. Con J=J1. Duración de la luz verde en cada ciclo	124
5.5. Con J=J1. Promedio de autos en el momento del cambio de luz	125
5.6. Promedio de autos en el momento del cambio de luz. Modelo dinámico	126
5.7. Duración de la luz verde en cada ciclo. Modelo dinámico	126
5.8. Con J=J2. Promedio de autos en el momento del cambio de luz	127
5.9. Con J=J2. Duración de la luz verde en cada ciclo	128
5.10. Duración de la luz verde en cada ciclo. Modelo dinámico	128
5.11. Promedio de autos en el momento del cambio de luz. Modelo dinámico	129
5.12. Con J=J3. Promedio de autos en el momento del cambio de luz	130

5.13. Con J=J3. Duración de la luz verde en cada ciclo	131
5.14. Duración de la luz verde en cada ciclo. Modelo dinámico	131
5.15. Promedio de autos en el momento del cambio de luz. Modelo dinámico	132
5.16. Con J=J4. Promedio de autos en el momento del cambio de luz	133
5.17. Con J=J4. Duración de la luz verde en cada ciclo	134
5.18. Duración de la luz verde en cada ciclo. Modelo dinámico	134
5.19. Promedio de autos en el momento del cambio de luz. Modelo dinámico	135
5.20. Con J=J5. Promedio de autos en el momento del cambio de luz	136
5.21. Con J=J5. Duración de la luz verde en cada ciclo	137
5.22. Duración de la luz verde en cada ciclo. Modelo dinámico	137
5.23. Promedio de autos en el momento del cambio de luz. Modelo dinámico	138
5.24. Con J=J6. Promedio de autos en el momento del cambio de luz	139
5.25. Con J=J6. Duración de la luz verde en cada ciclo	140
5.26. Duración de la luz verde en cada ciclo. Modelo dinámico	140
5.27. Promedio de autos en el momento del cambio de luz. Modelo dinámico	141
5.28. Tabla comparativa. Resultados modelos estático y dinámico	142
6.1. Parámetros usados en el modelo de seis fases	155
6.2. Tabla comparativa de los resultados para diferentes algoritmos en el modelo de 6 fases (1ª corrida)	156
6.3. Tabla comparativa de los resultados para diferentes algoritmos en el modelo de 6 fases (2ª corrida)	156
6.4. Parámetros usados en la aplicación del modelo de seis fases	161
6.5. Promedio de autos en el momento del cambio de luz. Ejemplo del modelo de seis fases	162

Introducción

Como el número de vehículos y las necesidades de transporte crecen, las ciudades del mundo muestran serios problemas de congestión de tráfico en sus carreteras. Los costes incluyen pérdida de tiempo (ya sea tiempo libre o de trabajo), aumento del consumo de combustible, contaminación, problemas de salud, stress y malestar, en general. Además, la congestión retarda el movimiento de mercancías y de servicios, de modo que, aumenta el precio de los productos y reduce la competitividad de los negocios. Si el problema no se trata correctamente, puede incluso paralizar el crecimiento económico y desarrollo de nuestras ciudades y contribuir significativamente al indeseable efecto invernadero y al cambio climático. Incluso cuando la sobresaturación de vehículos dura poco y ocurre en una zona pequeña o en una esquina en particular, su efecto negativo en el flujo del tráfico puede prolongarse por bastante tiempo. La sobresaturación se define como la situación en la cual los vehículos no pueden moverse libremente, debido a la cantidad de autos que están en una intersección o a las colas que se han formado en las calles por las cuales se puede salir de la intersección. Por otro lado, cuando no hay congestión queremos también que los vehículos atraviesen distintas zonas de la ciudad, o circulen por las vías principales en el menor tiempo posible. Por tanto, es necesario conseguir un sistema eficaz de transporte que facilite la movilidad de personas y mercancías. Podemos afirmar que esa movilidad es un factor importante que contribuye al crecimiento económico sostenible y a la generación de empleo. Para ello no sólo es necesario generar una infraestructura física adecuada, sino minimizar los impactos medioambientales que el transporte urbano y otros generan sobre el entorno. En efecto, la movilidad origina una serie de costes medioambientales, debido a sus implicaciones nocivas sobre el cambio climático (actualmente las emisiones de CO_2 son superiores en más de un 30 % a las producidas en 1.990) y el transporte de todo tipo es el *"único sector de la economía respecto del que se pronostican incrementos de las emisiones"* [17]. Aunque ha mejorado la eficiencia energética de los turismos mediante la aplicación de nuevas tecnologías y el peso de los mismos se ha reducido, lo que ha repercutido en

un menor consumo de combustible, el número de automóviles ha crecido. Pero además el ruido, la congestión del tráfico, cuya *"disminución contribuirá significativamente a la reducción de las emisiones de CO_2 "* [18], los accidentes y los efectos sobre la salud acarrearán una serie de costes económicos y sociales que el Libro Blanco sobre Crecimiento, Competitividad y Empleo [25], denominó: *"pérdidas de bienestar significativas, aunque ocultas"*, que pese a la dificultad de cuantificarlas la Unión Europea las valora en torno al 4 % del PIB.

A partir de la década de los 80 del siglo anterior, los responsables de los departamentos de tráfico intuyeron que las soluciones tradicionales que se estaban aplicando no serían capaces de resolver esta problemática. Estas soluciones típicamente implicaban la construcción de nuevas infraestructuras o la ampliación de las ya existentes, pero en muchos casos no eran soluciones viables debido principalmente a los elevados costes, a la carencia de espacio disponible y al inevitable impacto medioambiental. Por tanto, los departamentos de tráfico de todo el mundo están muy interesados en la optimización de las infraestructuras existentes, con el objeto de mejorar al máximo sus prestaciones. La Unión Europea tiene diferentes medios, a través de los cuales pretende mitigar los indeseables impactos citados con anterioridad. Desde el establecimiento de normas de comportamiento para los turismos nuevos, con el fin de reducir las emisiones de CO_2 [1], hasta el uso de una normativa para disminuir el ruido de vehículos de motor, pasando por las medidas ordenadoras del tráfico, como pueden ser, entre otras, el funcionamiento de los semáforos que regulan una intersección vial.

La función principal de los semáforos en el control de una intersección es dar el paso alternativamente a los distintos grupos de vehículos, peatones, bicicletas y/u otro sistema de transporte, de tal manera que éstos pasen a través de la intersección con un mínimo de problemas, riesgos y demoras. Los objetivos del diseño de una intersección regulada por semáforos son:

- Reducir y prevenir cierto tipo de accidentes en la intersección y en las intersecciones aledañas.
- Reducir las demoras que experimentan los peatones y los vehículos al intentar cruzar la intersección, y al mismo tiempo evitar los obstáculos en las intersecciones más cercanas causados por colas largas.
- Reducir el consumo de combustible de los vehículos en la intersección.
- Reducir la emisión de contaminantes al aire (CO_2), que afectará positivamente

al efecto invernadero, así como a la disminución de ruidos.

- Por último, mejorar la calidad de vida de los ciudadanos y su salud.

Para que dichos objetivos se cumplan, el elemento fundamental es sincronizar y optimizar el ciclo de los semáforos, es decir, sincronizar y optimizar la secuencia de estados (rojo, verde, etc.) por la que pasan los semáforos y que se repite de manera cíclica. Cada ciclo está asociado a una intersección, o sea, puede regir a varios semáforos. Cada fase del ciclo (o stage) se compone de los estados en los que están todos los semáforos de esa intersección durante un tiempo determinado, y que permiten uno o varios movimientos simultáneos a través de la intersección.

Este ciclo de los semáforos se debe estudiar desde tres puntos de vista distintos:

- La longitud en tiempo del ciclo completo de los semáforos, es decir, el tiempo necesario para que se dé una sucesión completa de indicaciones en los semáforos conectados a un mismo regulador. Independientemente de lo que resulte de los cálculos, la duración del ciclo tiene que estar forzosamente comprendida entre los límites que fija la psicología del conductor. La práctica indica que ciclos menores de 35 segundos o mayores de 120, se acomodan difícilmente a la mentalidad del usuario de la vía pública [66].
- El porcentaje del tiempo dedicado para cada una de las fases.
- Las transiciones entre los semáforos consecutivos. Un ejemplo típico de esta circunstancia es la presencia de dos semáforos consecutivos en una avenida. En este caso se define el tiempo de transición (offset time) como el tiempo desde que un semáforo pasa a estado verde, hasta que el siguiente semáforo de la avenida pasa también a verde.

Debido a que la buena coordinación de la red de semáforos de la ciudad y la optimización de los ciclos y fases de los mismos son herramientas fundamentales para tratar de resolver los problemas de congestión de tráfico y sobresaturación de vehículos, como se demuestra en numerosos trabajos (y entre ellos podemos citar [34], [55] y [51]), es por lo que se ha diseñado un nuevo modelo, que es la principal aportación de la presente tesis, para la resolución de dichos problemas.

Capítulo 1

Preliminares

1.1. Estado del arte

En esta sección se quiere dar una visión de los diferentes trabajos de investigación relacionados con el modelo que sostiene esta tesis. Se han clasificado los trabajos en tres grupos. En primer lugar se tienen los trabajos agrupados dentro de las siglas ATIS (Advanced Traveler Information Services) que son aquellos trabajos centrados en los servicios avanzados de información a los conductores. Por otra parte, se pueden agrupar otros trabajos en un segundo grupo etiquetado con las siglas ATMS (Advanced Traffic Management Systems) que significa sistemas avanzados de gestión del tráfico. Finalmente, otros trabajos de investigación concernientes al tráfico pueden agruparse en un tercer subconjunto que se puede denominar ATOS (Advanced Traffic Optimization Systems) y donde aglutinarían aquellas propuestas de sistemas avanzados de optimización del tráfico, en los cuales el tráfico no sólo es gestionado, sino que también se intenta su optimización de manera automática sin la ayuda de la experiencia de expertos humanos. Tales propuestas se han organizado en: optimización de una sola intersección, lógica difusa, redes de Petri, redes Neuronales o aproximaciones híbridas. Esta estructuración no es estática, ya que por su naturaleza, algunos trabajos podrían incluirse en más de un apartado.

Esta clasificación puede ser de utilidad para adquirir una ordenación mental inicial del campo de investigación en el cual se mueve la presente tesis.

1.1.1. Servicios avanzados de información a los conductores (ATIS)

Los servicios avanzados de información a los conductores son esos servicios que pueden potencialmente ayudarles a tomar decisiones para así realizar el viaje en las mejores condiciones de eficiencia y seguridad. Entre las principales funciones que realizan los ATIS, podemos resaltar:

- Planificación de viajes multimodales: proporcionan información regional global y asistencia al viajero de transporte particular (automóviles) y de transporte público.
- Servicios de información de guía para la ruta: proveen de una guía en línea autónoma o dinámica (información en tiempo real) que permite programar la ruta y propone instrucciones de viraje durante el viaje.
- Funciones de consulta y soporte: avisos y sugerencias que pueden incluir advertencias de incidentes, avisos de retardos, tiempos de viaje (estimado en tiempo real), aviso de condiciones meteorológicas adversas, restricciones a vehículos comerciales (altura y peso), información y estado de los estacionamientos, próximos peajes, etc...

Hay muchas iniciativas en este área. Aquí mostramos algunos ejemplos.

En la tesis defendida por Florian ([26]), se realiza un estudio empírico del impacto de los ATIS en la calidad de la red de los servicios de transporte mediante la aplicación del modelo DynaMIT (asignación dinámica de la red para la gestión de la información a los conductores). Los resultados principales son que el guiado dinámico de ruta puede mejorar los tiempos de los trayectos de los conductores, tanto de los guiados como de los no guiados, y también mejora el rendimiento de la infraestructura de tráfico existente.

En [31] se aplica a un sistema de información de tráfico en una autopista sin peaje, un modelo autómatas celular de alta resolución para simulación de tráfico. El sistema proporciona una simulación para el estado actual del tráfico sin usar sensores, y también realiza estimaciones de la situación futura del tráfico (en 30 y 60 minutos). Esta información a los usuarios de la red es proporcionada por un applet de java en una página web.

1.1.2. Sistemas avanzados de gestión del tráfico (ATMS)

Los sistemas avanzados de gestión del tráfico son los sistemas que facilitan la labor de los técnicos que gestionan el tráfico. Hay muchos trabajos acerca de esta materia, la mayor parte de ellos se centran en la simulación del tráfico. Los modelos de simulación de tráfico se pueden dividir en dos grandes categorías: microscópica y macroscópica, dependiendo del nivel de abstracción que utilicen. En los modelos microscópicos o de micro simulación se modela el tráfico a nivel de cada vehículo. En los modelos macroscópicos se abstrae hasta ver el comportamiento de todo el sistema, permitiendo así tener una visión global del comportamiento. Algunos ejemplos son los siguientes:

En [59] se propone el uso de señales de tráfico sensibles (Responsive Signals) combinado con la capacidad de la red (en vez del coste total del viaje) como criterio de control. La capacidad de la red se maximiza si las señales funcionan para equilibrar la densidad de tráfico en las zonas más ocupadas de la red.

En el trabajo [53] se utilizan supercomputadores vectoriales para simular el tráfico utilizando modelos basados en autómatas celulares en el menor tiempo posible.

El modelo AIMSUN2 se ha utilizado para simular las Rondas en Barcelona ([11]). Utiliza computadores paralelos para reducir el tiempo de ejecución.

El modelo MITSIM ([74]) se ha utilizado para evaluar aspectos tanto del sistema de control de tráfico como de las configuraciones de las vías de acceso en el proyecto área central/túnel en Boston (EE.UU.). Incorpora explícitamente la predicción del tráfico, la información del tráfico variable con el tiempo y la elección de ruta dinámica.

El modelo INTEGRATION se ha utilizado para simular el tráfico del Salt Lake Metropolitan Área (EE.UU.)([56]). El objetivo de este artículo es triple. Primero se presenta la viabilidad de modelar una red a gran escala con un nivel de detalle microscópico. En segundo lugar, se explica como han logrado la superación de los inconvenientes que supone la ingente colección de datos necesaria para construir y calibrar una simulación microscópica con una red de una escala tan grande. Por último, se describen las oportunidades y aplicaciones del uso de una herramienta de simulación microscópica en oposición a una macroscópica.

El modelo INTELSIM se usa en [6] y [7]. En esos trabajos se ha desarrollado un modelo de aceleración lineal de los llamados modelos de autos siguiéndose (car-following) o modelos de seguir al líder (los cuales modelan el comportamiento de cada vehículo individual dependiendo del vehículo que lo antecede) para la simulación realista del

flujo de tráfico orientado a aplicaciones a sistemas inteligentes de transporte (ITS). Se simulan los tiempos de reacción de freno y los tiempos de reacción en cadena de los conductores. Los autores argumentan que el modelo propuesto facilita perfiles de aceleración continuos en lugar de los perfiles escalonados que se usan comúnmente. Además en [7] los mismos autores comparan varios métodos de autos siguiéndose con el modelo que ellos proponen y con datos de campo.

En [42] se presenta un sistema experto para gestionar la congestión de tráfico. El modelo propuesto abarca un algoritmo de fusión de datos para el análisis de la congestión que permite representar e interpretar diferentes tipos de datos con distintos niveles de fiabilidad e incertidumbre junto con un algoritmo eficiente para la selección de planes de control que determina respuestas alternativas al control del tráfico. Se describe el plan propuesto con una explicación del proceso de razonamiento para ayudar a las decisiones de los operadores del tráfico y se muestra un ejemplo de validación para exhibir la capacidad del sistema para reducir la congestión.

En [68] se explica que un tiempo de ciclo común (common cycle time) a todos los semáforos de un conjunto de intersecciones es una aproximación peor que ciclos independientes para cada intersección.

En [13] se propone un modelo de alta fidelidad (high fidelity) para simular un gran volumen de tráfico a nivel regional. Su modelo usa conceptos de los modelos de autómatas celulares y de los de autos siguiéndose. (En él, el universo está dado por las calles de la ciudad, o por las carreteras a nivel regional y los elementos son los vehículos que se mueven siguiendo un conjunto de reglas. El modelo se puede extender fácilmente para representar otros elementos, por ejemplo conductores con distintos perfiles psicológicos, distintos tipos de vehículos, distintas representaciones de cruces de calles o semáforos). Proponen el concepto de la ocupación del espacio (Space Occupancy-SOC) y lo usan para medir la congestión de tráfico. Su objetivo es simular grandes volúmenes de tráfico con un tiempo de ejecución más corto usando algoritmos eficientes en un ordenador personal.

1.1.3. Sistemas avanzados de optimización del tráfico (ATOS)

Los sistemas avanzados de optimización del tráfico son los sistemas donde el tráfico se optimiza -o intenta ser optimizado- de una manera automática, sin la interacción humana. A continuación se presentan algunos ejemplos notables.

1.1.3.1. Optimización de una sola intersección

En [21] se analiza un problema de control óptimo de semáforos para un cruce de dos carreteras con los dos sentidos de circulación. Se expone primero, un modelo exacto y acíclico que describe la evolución de las longitudes de las colas en cada carril como una función del tiempo y de los instantes de cambio ("switching"). También se expone la forma de determinar una sucesión switching sub-óptima para este sistema. Este artículo se puede considerar la base de la presente Tesis.

En [70] se controla una intersección aislada aplicando técnicas basadas en la teoría de control de sistemas lineales para resolver el problema del modelo lineal de tráfico. La contribución principal de esta investigación es el desarrollo de una metodología, que utiliza la tecnología disponible, para aliviar la congestión recurrente de una intersección aislada, causada por la alta demanda de transporte.

Se publica un trabajo muy interesante en [71]. En este artículo, se piensa el tráfico como formado por un conjunto de cruces que son optimizados de forma individual. Proponen usar algoritmos de aprendizaje por refuerzo para optimizar lo que ellos consideran un problema de decisión multiagente.

En [73] para un cruce simple con dos direcciones los autores proponen un nuevo método basado por un lado en nuevas tecnologías para el reconocimiento de vehículos y semáforos, que les permiten tratar el movimiento de cada vehículo de forma individual y distinguir diferentes tipos de vehículos como, vehículos públicos o de emergencia y por otro basado en la utilización de una aproximación branch and bound para el control de semáforos con el objetivo de minimizar el tiempo de evacuación de la intersección.

1.1.3.2. Lógica difusa

En [76] los autores presentan un sistema de control difuso para ampliar o reducir el ciclo de los semáforos. Mediante el uso de semáforos con sensores se puede ampliar el ciclo del semáforo cuando éste detecta que están pasando muchos vehículos en la carretera o reducir el ciclo si hay pocos vehículos pasando. En este trabajo se simulan los resultados observándose un incremento en la eficiencia mediante el uso de estos semáforos. Se realiza una adaptación local para un solo semáforo en vez de una optimización global.

En [28] se propone una técnica de optimización local en tiempo real de una intersección basada en lógica difusa (fuzzy logic).

En [44] se establecen como variables de entrada la densidad de tráfico y la velocidad de los vehículos durante el estado verde del semáforo y como salida el tiempo que debe durar cada uno de los estados dentro del ciclo del semáforo.

El objetivo del trabajo presentado en [32] es optimizar el flujo de tráfico de un conjunto de intersecciones mediante el uso de controladores locales fuzzy y un controlador para un caso especial. Cada controlador local fuzzy controla el flujo de tráfico en el cruce asignado, de acuerdo con el flujo de tráfico de su cruce y de los cruces adyacentes. Cuando debido al estado del tráfico, los controladores locales no son suficientes, se activa el controlador para el caso especial, el cual, usando un algoritmo tipo recocido simulado optimiza el tiempo verde de todos los cruces de la red.

1.1.3.3. Redes de Petri

En [22] se aplican las redes de Petri para proporcionar una representación modular de las redes de tráfico urbano regulado por semáforos. Con el fin de describir minuciosamente el movimiento de vehículos en la red de tráfico, se adopta una representación microscópica realizada vía redes de Petri. Una característica interesante del modelo diseñado consiste en la posibilidad de representar los retardos (offset-time), entre los ciclos de los diferentes semáforos de una manera empotrada en el mismo modelo. Aunque es un trabajo muy interesante, los autores optimizan solamente la coordinación entre ciclos de diferentes semáforos.

Otro trabajo interesante usando redes de Petri es [40], donde se utilizan para controlar una sola intersección por medio de controladores lógicos programables (Programmable Logic Controllers-PLCs). Comparan tres métodos para modelar los semáforos en una intersección y encuentran que el más apropiado es el que combina las redes de Petri con los PLCs.

En [41] los autores usan las redes de Petri para modelar y controlar intersecciones semaforizadas. En particular aplican las redes de Petri a un controlador de semáforos de ocho fases.

1.1.3.4. Redes Neuronales

En [60] los autores presentan un enfoque con redes neuronales para optimizar los ciclos de los semáforos. Se utiliza una red neuronal (NN) para implementar la función de

control de los semáforos. El proceso de desarrollo de la NN se alimenta exclusivamente con datos reales. Debido a ello, este método sólo es útil en un sistema que tenga un módulo de adquisición de datos on-line instalado. Sin embargo, tales sistemas no son hasta ahora comunes en absoluto.

En [45] se optimiza el offset-time entre dos semáforos usando redes neuronales artificiales (ANNs). Aunque la metodología que se propone no trata explícitamente el parámetro de desfase, se enfoca la optimización del tráfico de una manera mucho más flexible.

En [37] los autores presentan un método de control inteligente de semáforos basado en la teoría de “extension neural network” (ENN) para una intersección. Primero miden (en la calle principal y en la secundaria de un cruce seleccionado) el número de vehículos que circulan y el tiempo de paso de un vehículo en el período de luz verde. Seguidamente los datos de tal medición se usan para construir un método de estimación basado en ENN para la detección del flujo de tráfico de un cruce estándar. Los resultados computacionales indican que el método de estimación propuesto puede discriminar de forma rápida y exacta el flujo de tráfico de un cruce.

1.1.3.5. Aproximaciones híbridas

El proyecto TRANSIMS (Transportation Analysis and simulation system) utilizó modelos de autómatas celulares para simular el tráfico para la ciudad de Fort Worth en Dallas (EE.UU.) usando computación paralela ([52]). Incluye módulos de simulación microscópica, planificación de rutas y estimación de la emisión de gases.

En [77] se propone el concepto de tiempo óptimo en verde (optimal green time), que reduce el tiempo de espera medio de los vehículos, al tiempo que se mejora la media de velocidad del vehículo usando reglas difusas (fuzzy) y redes neuronales. A través de la simulación por ordenador este método ha demostrado ser mucho más eficiente que usar tiempos fijos para los ciclos de los semáforos. La red neuronal difusa mejora de manera consistente el tiempo de espera, la velocidad de los vehículos, y el consumo de combustible. Este trabajo sólo tiene en cuenta una pequeña cantidad de semáforos (dos intersecciones cercanas). Se coincide con los autores en la poca eficiencia de los ciclos fijos de los semáforos.

En [57] se utiliza una arquitectura “ad hoc” para optimizar una red de tráfico de nueve intersecciones. Utiliza algoritmos genéticos (GA) como técnica de optimización

sobre una sola máquina. Se usa el modelo CORSIM (Corridor Traffic Simulation Model) ([33]) en la función de evaluación del GA.

Una combinación interesante de algoritmos y simulación de tráfico se publica en [67]. En este trabajo se presenta un sistema de programación de rutas y planificación horaria para vehículos de transporte de carga. Usan algoritmos genéticos (GA) como técnica de optimización. El objetivo de los GA es la minimización del coste del viaje. También se propone y prueba en una red real un algoritmo dinámico de programación de rutas para vehículos. El modelo de simulación de tráfico implementado es macroscópico.

Otro trabajo muy interesante se presenta en [69], en él se formula un modelo de asignación dinámica tipo Dynamic System-optimal (DSO) para una red urbana congestionada con un determinado número de cruces regulados por semáforos. También combinan simulación de tráfico con algoritmos genéticos. Lo fundamental de este trabajo es asignar a cualquier viajero una ruta. Se utiliza un algoritmo genético para minimizar el tiempo total de viaje de los usuarios. Se usa un modelo macroscópico para la estimación de los retardos de tráfico. El problema DSO se resuelve primero con tiempos fijos de semáforos, y también optimizando los tiempos de los semáforos.

En [61] se presenta una metodología para la optimización de los ciclos de un semáforo en una red de tráfico. Se diseña un modelo de tráfico microscópico y se usa además un autómata celular. Con esta teoría los vehículos se consideran como entidades unidimensionales, las calles se ejemplifican como un conjunto de puntos; en cada punto sólo puede haber un vehículo en cada instante de tiempo. Para alcanzar una solución del problema se representa el estado de los semáforos mediante un cromosoma, que tiene longitud variable en función del período de tiempo que se quiera optimizar, aunque se supone que ese período de tiempo representa un ciclo base que se repite indefinidamente. Este cromosoma está compuesto por valores enteros que representan qué semáforo está abierto en el cruce en cada momento. Los resultados obtenidos con esta codificación y el simulador descrito muestran que aparte de ser una forma válida, permite optimizar varias intersecciones al mismo tiempo, lo que reduce los tiempos de cálculo con respecto a otras soluciones.

En [63] se comparan dos versiones de un simulador microscópico de tráfico: un simulador de tráfico estocástico con uno determinístico. Hay tres diferencias entre la versión estocástica y la determinística: el orden de actualización de las celdas, el tiempo de creación del nuevo vehículo y la probabilidad de aceleración. En este trabajo se

concluye que el simulador estocástico es un proceso estadístico (convergente) adecuado de comparación y se demuestra que las salidas del simulador determinístico tienen una alta correlación lineal con las estocásticas. Por lo tanto, el simulador determinístico puede organizar la clasificación de la población en orden a la función fitness al menos como el simulador estocástico, pero con un tiempo de cálculo notablemente más bajo.

En [62] se describe la diferencia entre dos tipos de codificación: el rendimiento entre diferentes cruces y las estrategias de mutación. El logro principal de este trabajo fue demostrar (por medio de un amplio conjunto de tests) que, al menos para el caso particular de un cruce de pequeño nivel, si se combina con una probabilidad variable para la mutación se obtiene un gran ahorro de tiempo de cálculo. Además se expone cómo esa elección permite al algoritmo cubrir más rápidamente el espacio de soluciones debido a la mayor variabilidad de genes entre generaciones. Esa combinación parece evitar una convergencia prematura.

En [64] se introduce un estudio considerando tres criterios candidatos a ser el primer paso para la extensión de la función fitness hacia una multicriterio. En ese criterio se relaciona el número total de vehículos que abandonan la red, la ocupación de la red y la emisión de gases de efecto invernadero. Se realiza un estudio de la correlación entre esos parámetros.

En [65] usando una metodología basada en tres pilares:

1. Un multicomputador MIMD tipo Cluster Beowulf
2. Un algoritmo genético como técnica de optimización no determinística
3. Un simulador de tráfico microscópico basado en los autómatas celulares

se optimiza una red de tráfico situada en Santa Cruz de Tenerife (España).

En [39] se desarrolla un problema de regulación del tráfico a través del ajuste de los ciclos de los semáforos (problema en el que existen dos fases en cada ciclo), proponiéndose para su solución un método híbrido que combina una metaheurística tipo recocido simulado para calcular una buena aproximación de la solución, que luego será mejorada con un algoritmo quasi-Newton para problemas de optimización no suaves.

1.2. Problema a investigar y metodología utilizada

El objetivo de la investigación es resolver un problema de control óptimo consistente en asignar tiempos a luces de semáforos en intersecciones de distintos tipos y aplicar los resultados a cruces (con congestiones frecuentes) de la ciudad de A Coruña. El planteamiento es el siguiente: conocidas las tasas de llegada y de salida de vehículos del cruce, para un entero N dado, y un tiempo inicial t_0 , se pretende calcular la mejor sucesión t_1, t_2, \dots, t_N de instantes de cambio de las luces de los semáforos (tiempos switching). El criterio de comparación a considerar se expresa en una función objetivo a minimizar que puede ser:

- Suma (ponderada) de las longitudes medias de las colas en todos los carriles
- Longitud (ponderada) media de la cola en el carril con mayores colas
- Longitud (ponderada) de la cola más larga
- Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles
- Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera
- Una combinación de las anteriores

Una vez identificadas zonas neurálgicas en la ciudad a partir de distintos parámetros de llegada de coches y tiempos estimados de salida, se estudian cruces con características análogas, como por ejemplo:

- (a) Cruces de dos calles con los dos sentidos de circulación en los que solamente se puede circular de frente o girar a la derecha (llamados cruces con dos fases)
- (b) Cruces de dos calles con los dos sentidos de circulación en los que en una dirección se podrá circular de frente, girar a la derecha o a la izquierda, pero que en la otra dirección sólo está permitido circular de frente o girar a la derecha (llamados cruces con tres fases)
- (c) Cruces de dos calles con los dos sentidos de circulación en los que en ambas direcciones se podrá circular de frente, girar a la derecha o a la izquierda (llamados cruces con cuatro fases)

- (d) Cruces de dos calles con los dos sentidos de circulación y con dos carriles en cada sentido (con 8 semáforos en el cruce) en los que en ambas direcciones se podrá circular de frente, girar a la derecha o a la izquierda (llamados cruces con seis fases)

Nos proponemos lo siguiente:

1. Estudiar teóricamente y modelar el problema de optimización para un cruce del tipo (a), en donde las restricciones del programa se interpretan como un problema de complementariedad lineal extendido (ELCP). Analizaremos:
 - Caso estático, es decir, el modelo en el cual los tiempos de las luces son variables, pero sólo dependen de los flujos de llegada.
 - Caso dinámico, es decir, se considera que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz.
2. Resolver el problema modelado mediante tres tipos de algoritmos:
 - Tipo quasi-Newton (método del gradiente discreto)
 - Heurístico (recocido simulado)
 - Híbrido (que combina ambas estrategias)
3. Generalizar el modelo a cruces de los tipos (b) (c) y (d)
4. Obtener resultados numéricos a partir de la programación de los algoritmos anteriores en MATLAB.
5. Aplicar los métodos descritos a cruces de zonas neurálgicas de la ciudad de A Coruña, que correspondan a cada uno de los modelos estudiados teóricamente, usando datos reales o estimados de ocupación en distintas franjas horarias y diferentes días de la semana.
6. Derivar actuaciones que permitan una mayor fluidez del tráfico rodado en las zonas estudiadas.
7. Relatar y esbozar otras líneas de investigación, como por ejemplo: uso de otros métodos heurísticos: colonias de hormigas, algoritmos genéticos, etc. o extensión de la teoría propuesta al caso de las rotondas o de cruces concatenados.

Capítulo 2

Fundamentos teóricos

2.1. Introducción

En este capítulo se exponen los conceptos teóricos, en los que se fundamentan los métodos usados para tratar de resolver los diferentes problemas de optimización que se presentan en los modelos que se exponen en la presente tesis para optimizar el flujo de tráfico en un cruce regulado por semáforos, en el que existen 2, 3, 4 o 6 fases.

En primer lugar se define el problema de complementariedad lineal extendido y se exponen y demuestran las condiciones bajo las cuales este problema tiene solución única, ya que en el modelo presentado en la sección 3.1 las restricciones del programa se pueden expresar como un problema de complementariedad lineal extendido. Como éste es un problema NP- duro en el que aparecen (cuando se trabaja con un número importante de ciclos) un gran número de variables, restricciones suaves y no suaves, se propone para su solución un método híbrido consistente en una heurística y un algoritmo para resolver un problema de complementariedad lineal.

Para los modelos expuestos en las secciones 3.2, 4.1, 5.1, 5.2 y 6.1 se proponen tres tipos de algoritmos:

- Tipo quasi-Newton: Por tanto en la sección 2.3 se trata el método del gradiente discreto.
- Metaheurístico basado en la heurística de recocido simulado, de ahí que en la sección 2.4 se estudian los métodos heurísticos y en particular el de recocido simulado.

- Un híbrido en el que, usando las ventajas de ambos algoritmos, primero se busca una buena aproximación a la solución mediante el método de recocido simulado y luego se mejora el comportamiento en una vecindad del punto mediante el método del gradiente discreto.

2.2. Los problemas de complementariedad lineal

2.2.1. Introducción. De KKT a los LCP

La teoría de los programas matemáticos con restricciones de desigualdad se desarrolló a partir de 1951 por H. W. Kuhn y A. W. Tucker ([38]). Ellos enuncian un teorema que ya había sido demostrado con anterioridad por William Karush en su tesis doctoral ([36]), pero que nunca había publicado. El teorema establece condiciones necesarias de optimalidad local para programas del tipo:

$$\begin{aligned} & \text{minimizar } f(x_1, x_2, \dots, x_n) \\ & \text{sujeto a : } h_1(x_1, x_2, \dots, x_n) \geq 0 \\ & \quad h_2(x_1, x_2, \dots, x_n) \geq 0 \\ & \quad \dots\dots\dots \\ & \quad h_m(x_1, x_2, \dots, x_n) \geq 0 \end{aligned}$$

donde $f : D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$, $h : D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$, $h = (h_1, h_2, \dots, h_m)$ son diferenciables. Tales condiciones (KKT) vienen dadas, para puntos que cumplen una cierta condición de regularidad, por:

1. $\nabla f(x) - \sum_{i=1}^m \lambda_i \nabla h_i(x) = 0$
2. $h_i(x) \geq 0, \quad i = 1, \dots, m.$
3. $\lambda_i h_i(x) = 0, \quad i = 1, \dots, m.$
4. $\lambda_i \geq 0, \quad i = 1, \dots, m.$

donde $\lambda = (\lambda_1, \dots, \lambda_m) \subseteq \mathbb{R}^m$

En la búsqueda de métodos para resolver este sistema, se abordó el caso cuadrático

$$\min_{x \geq 0} \left\{ \frac{1}{2} x^T M x + q x \right\}$$

donde M es una matriz cuadrada simétrica de orden n y $q \in \mathbb{R}^n$. En este caso las condiciones son:

1. $x_i \geq 0, \quad i = 1, \dots, n$
2. $(Mx + q)_i \geq 0, \quad \text{si } x_i = 0$

$$3. (Mx + q)_i = 0, \quad \text{si } x_i > 0$$

es decir, se trata de encontrar x en \mathbb{R}^n tal que $Mx + q \geq 0$, $x \geq 0$, $x^T(Mx + q) = 0$. Este es el origen de la definición del problema de complementariedad lineal (LCP) ([16]).

2.2.2. El problema de complementariedad lineal y algunas generalizaciones

Definición 2.2.1. *El problema de complementariedad lineal (LCP) está definido así: Dada una matriz $M \in \mathbb{R}^{n \times n}$ y un vector $q \in \mathbb{R}^n$, hallar dos vectores $x, y \in \mathbb{R}_+^n$, tales que:*

$$\begin{aligned} y &= q + Mx \\ x^T y &= 0 \end{aligned}$$

Denotamos este LCP así: $LCP(M, q)$.

El nombre proviene de la condición $x^T y = 0$, que se llama *condición de complementariedad* ya que requiere que al menos una de las variables del par (x_j, y_j) sea igual a cero en la solución del problema, para $j = 1, \dots, n$.

Teniendo en cuenta el origen de la definición del problema de complementariedad lineal, se ve claramente que éste tiene asociado el siguiente programa cuadrático:

$$\begin{aligned} \text{minimizar} \quad & x^T(q + Mx) \\ \text{sujeto a:} \quad & x \geq 0 \\ & q + Mx \geq 0 \end{aligned} \tag{2.1}$$

Algunas generalizaciones del problema de complementariedad lineal son: el problema de complementariedad lineal horizontal (HLCP), el problema de complementariedad lineal vertical (VLCP), el problema de complementariedad lineal mixta (MLCP) y los que a continuación se definen:

Definición 2.2.2. *El problema de complementariedad lineal generalizado (GLCP), [que fue introducido por Y. Ye en [75]] está definido así: Dadas matrices $A, B \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{m \times k}$ y un vector $q \in \mathbb{R}^m$, encontrar vectores $x, y \in \mathbb{R}^n$, $z \in \mathbb{R}^k$ tales que:*

$$\begin{aligned} Ax + By + Cz &= q \\ (x, y, z) &\geq 0 \\ x^T y &= 0 \end{aligned}$$

Definición 2.2.3. El problema de complementariedad lineal extendido (ELCP) [que fue introducido por O.L. Mangasarian y J.S. Pang en [48]] está definido así: Dadas dos matrices $M, N \in \mathbb{R}^{m \times n}$ y dado K un conjunto poliédrico de \mathbb{R}^m , hallar dos vectores $x, y \in \mathbb{R}_+^n$ tales que:

$$\begin{aligned} Mx - Ny &\in K \\ x^T y &= 0 \end{aligned}$$

Denotamos este ELCP así: $ELCP(M, N, K)$. Cuando $m=n$, $N=I$ y $K=\{-q\}$, este problema se reduce al $LCP(M, q)$.

2.2.3. El problema de complementariedad lineal extendido (ELCP)

Teorema 2.2.1. El GLCP y el ELCP son equivalentes ([30])

DEMOSTRACIÓN. Para $A, B \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{m \times k}$ y $q \in \mathbb{R}^m$ el problema GLCP se convierte en $ELCP(A, -B, K)$, si definimos $K = \{q - Cz/z \geq 0\}$. Por otro lado, si tenemos un $ELCP(M, N, K)$, como K es un conjunto poliédrico, para alguna matriz $L \in \mathbb{R}^{k \times m}$ y algún vector $b \in \mathbb{R}^k$, podemos escribir $K = \{u \in \mathbb{R}^m / Lu \geq b\}$. Entonces el $ELCP(M, N, K)$ se convierte en un GLCP si consideramos $m=k$, $A=LM$, $B=-LN$, $C=-I$ y $q=b$. \square

Definición 2.2.4. Asociado al $ELCP(M, N, K)$ podemos definir un programa bilineal

$$\begin{aligned} &\text{minimizar } x^T y \\ &\text{sujeto a: } x \geq 0 \\ &\quad y \geq 0 \\ &\quad Mx - Ny \in K \end{aligned}$$

Denotamos este problema por: $BLP(M, N, K)$

Observación 2.2.1. En 2.1 habíamos expuesto el programa cuadrático asociado al $LCP(M, q)$; éste está definido sólo para la variable x , mientras que el $BLP(M, N, K)$ involucra al par (x, y) . Sin embargo el $BLP(M, N, K)$ juega un papel similar en el estudio del $ELCP(M, N, K)$ como el que jugaba 2.1 en el tratamiento del $LCP(M, q)$. Como la función objetivo del $BLP(M, N, K)$ es claramente no negativa sobre la región factible del $ELCP(M, N, K)$, es decir, sobre $\{(x, y) \in \mathbb{R}_+^{2n} / Mx - Ny \in K\}$, el $ELCP(M, N, K)$ es

equivalente al $BLP(M,N,K)$ en el sentido de que un par de vectores (x,y) resuelve el $ELCP(M,N,K)$ si y sólo si (x,y) es una solución óptima global del $BLP(M,N,K)$ con un valor de la función objetivo igual a cero. Además por el teorema de Frank-Wolfe de la programación cuadrática ([27]), el $BLP(M,N,K)$ siempre tiene una solución óptima si es factible (es decir si el conjunto $\{(x,y) \in \mathbb{R}_+^{2n}/Mx - Ny \in K\}$ es no vacío). Lo que no se verifica, en general, es que en la solución óptima del $BLP(M,N,K)$ el valor de la función objetivo sea cero.

En lo que sigue (basándonos en [30]) estableceremos varios resultados que refuerzan la relación entre el $ELCP$ y su BLP asociado.

Definición 2.2.5. Dadas matrices $M, N \in \mathbb{R}^{m \times n}$ y dado K un conjunto poliédrico de \mathbb{R}^m decimos que $\{M, N\}$ tiene la E -propiedad de suficiencia en columnas con respecto a K si se verifica:

$$\text{Si } [Mu - Nv \in K - K, u * v \leq 0] \Rightarrow u * v = 0$$

donde $*$ denota el producto de Hadamard de dos vectores, es decir $u*v$ es el vector cuyas componentes son los productos de las componentes correspondientes de los vectores u y v .

Observación 2.2.2. En el contexto de los $LCP(M,q)$ M tiene la propiedad de suficiencia en columnas si $x * Mx \leq 0 \Rightarrow x * Mx = 0$

Teorema 2.2.2. El par $\{M, N\}$ tiene la E -propiedad de suficiencia en columnas con respecto a K si y sólo si para cada $p \in \mathbb{R}^m$, el conjunto solución del $ELCP(M,N,K+p)$ si es no vacío es un conjunto convexo.

DEMOSTRACIÓN.

\Leftarrow Supongamos que la propiedad de suficiencia en columnas falla. Entonces podemos encontrar dos vectores u y v tales que $Mu - Nv \in K - K$, $u * v \leq 0$ y $u_i v_i < 0$ para algún índice i . Sean $Mu - Nv = a - b$ con $a, b \in K$ y $p = Mu^+ - Nv^+ - a = Mu^- - Nv^- - b$, donde u^+ denota el vector que tiene por i -ésima componente ($\forall i$) el valor máximo calculado entre las i -ésimas componentes de los vectores u y 0 , mientras que $u^- = u^+ - u$. Entonces (u^+, v^+) y (u^-, v^-) son soluciones del $ELCP(M,N,K+p)$ ya que: por definición $u^+ \geq 0$, $v^+ \geq 0$, $u^- \geq 0$, $v^- \geq 0$; como se verifica $u * v \leq 0$ entonces $u^{+T} v^+ = 0$, $u^{-T} v^- = 0$; $Mu^+ - Nv^+ = a + p \in K + p$, $Mu^- - Nv^- = b + p \in K + p$. Por tanto el conjunto solución del $ELCP(M,N,K+p)$ es no vacío, y sin embargo no es convexo, ya que una combinación convexa de esas dos soluciones no es una solución; efectivamente,

sea t entre 0 y 1, $t(u^+, v^+) + (1-t)(u^-, v^-) = (tu^+ + (1-t)u^-, tv^+ + (1-t)v^-)$ no es una solución, pues si lo fuera $(tu^+ + (1-t)u^-)^T (tv^+ + (1-t)v^-)$ tendría que ser cero y no lo es pues $u^+ * v^- + u^- * v^+ \neq 0$ debido a que $u_i v_i < 0$ para algún índice i .

\Rightarrow Supongamos que se verifica la propiedad de suficiencia en columnas. Si $(x, y), (\bar{x}, \bar{y})$ son dos soluciones del ELCP(M,N,K+p), entonces $(u, v) = (x - \bar{x}, y - \bar{y})$ satisfará que $Mu - Nv \in K - K, u * v \leq 0$ y por tanto $u * v = 0$. Debido a la no negatividad de los vectores x, y, \bar{x}, \bar{y} , tenemos que $x * \bar{y} = 0 = y * \bar{x}$, de donde se sigue que para cualquier t entre 0 y 1, $(tx + (1-t)\bar{x}) * (ty + (1-t)\bar{y}) = 0$, por tanto cualquier combinación convexa de $(x, y), (\bar{x}, \bar{y})$ es solución del ELCP(M,N,K+p), efectivamente: sea t entre 0 y 1, $t(x, y) + (1-t)(\bar{x}, \bar{y}) = (tx + (1-t)\bar{x}, ty + (1-t)\bar{y})$ es solución del ELCP(M,N,K+p), pues debido a la no negatividad de los vectores x, y, \bar{x}, \bar{y} entonces $tx + (1-t)\bar{x} \geq 0, ty + (1-t)\bar{y} \geq 0$; $(tx + (1-t)\bar{x})^T (ty + (1-t)\bar{y}) = 0$ como consecuencia de que $(tx + (1-t)\bar{x}) * (ty + (1-t)\bar{y}) = 0$ y por último $M(tx + (1-t)\bar{x}) - N(ty + (1-t)\bar{y}) \in K + p$ ya que $Mx - Ny \in K + p, M\bar{x} - N\bar{y} \in K + p$. \square

Definición 2.2.6. Dadas matrices $M, N \in \mathbb{R}^{m \times n}$ y dado K un conjunto poliédrico de \mathbb{R}^m decimos que $\{M, N\}$ tiene la E -propiedad de suficiencia en filas con respecto a K si se verifica:

$$\text{Si } [(M^T u) * (N^T u) \leq 0, u \in (0^+K)^*] \Rightarrow (M^T u) * (N^T u) = 0$$

donde si $K = \{u \in \mathbb{R}^m / Lu \geq b\}$ para alguna matriz $L \in \mathbb{R}^{n \times m}$ y algún vector $b \in \mathbb{R}^n$, 0^+K denota su cono recesión, es decir $0^+K = \{u \in \mathbb{R}^m / Lu \geq 0\}$ y $(0^+K)^*$ denota el dual de este cono recesión, es decir $(0^+K)^* = \{v \in \mathbb{R}^m / v = L^T \lambda \text{ para algún } \lambda \in \mathbb{R}_+^n\}$.

Observación 2.2.3. En el contexto de los LCP(M,q), una matriz cuadrada se dice que tiene la propiedad de suficiencia en filas si M^T tiene la propiedad de suficiencia en columnas.

Definición 2.2.7. Dadas matrices $M, N \in \mathbb{R}^{m \times n}$, decimos que $\{\bar{M}, \bar{N}\}$ es un cambio en columnas de $\{M, N\}$ si para cada índice i , $\bar{M}_i = M_i$ y $\bar{N}_i = N_i$, o $\bar{M}_i = N_i$ y $\bar{N}_i = M_i$; donde cada subíndice i se refiere a la columna correspondiente.

Teorema 2.2.3. El par $\{M, N\}$ tiene la E -propiedad de suficiencia en filas con respecto a K si y sólo si para cada $p \in \mathbb{R}^m$ y para cada cambio en las columnas $\{\bar{M}, \bar{N}\}$ de

$\{M, N\}$, cada vector de KKT del programa bilineal $BLP(\overline{M}, \overline{N}, K + p)$

$$\begin{aligned} & \text{minimizar } x^t y \\ & \text{sujeto a : } x \geq 0 \\ & \quad y \geq 0 \\ & \quad \overline{M}x - \overline{N}y \in K + p \end{aligned}$$

es solución del $ELCP(\overline{M}, \overline{N}, K + p)$.

DEMOSTRACIÓN.

\Rightarrow Supongamos que $\{M, N\}$ tiene la E-propiedad de suficiencia en filas, como esta propiedad se conserva si cambiamos $\{M, N\}$ por alguno de sus cambios en columnas y K se cambia por $K + p$, probamos esta implicación sólo para el $ELCP(M, N, K)$.

Sea (x, y) un vector de KKT del problema $BLP(M, N, K)$

$$\begin{aligned} & \text{minimizar } x^T y \\ & \text{sujeto a : } x \geq 0 \\ & \quad y \geq 0 \\ & \quad Mx - Ny \in K \quad (\Leftrightarrow LMx - LNy - b \geq 0) \end{aligned}$$

entonces existen vectores r, s y λ tales que:

$$y - r - (LM)^T \lambda = y - r - M^T L^T \lambda = 0 \Rightarrow y = r + M^T L^T \lambda$$

$$x - s + (LN)^T \lambda = x - s + N^T L^T \lambda = 0 \Rightarrow x = s - N^T L^T \lambda$$

$$x \geq 0, r \geq 0, x^T r = 0$$

$$y \geq 0, s \geq 0, y^T s = 0$$

$$LMx - LNy - b \geq 0, \lambda \geq 0, (LMx - LNy - b)^T \lambda = 0$$

Usando estas propiedades tenemos: $(M^T L^T \lambda) * (N^T L^T \lambda) = (y - r) * (s - x) \leq 0$, pero como $L^T \lambda \in (0^+ K)^*$, usando la hipótesis concluimos que: $(M^T L^T \lambda) * (N^T L^T \lambda) = 0$ y como consecuencia que $x * y = 0$ y por tanto que el par (x, y) es solución del $ELCP(M, N, K)$.

\Leftarrow Supongamos que no se verifica la E-propiedad de suficiencia en filas, por tanto existe un vector $u \in \mathbb{R}^m$ tal que:

$$\text{Si } [(M^T u) * (N^T u) \leq 0, u \in (0^+ K)^*] \Rightarrow (M^T u) * (N^T u) \neq 0$$

entonces para algún i $(M^T u)_i (N^T u)_i < 0$. Podemos suponer que $(M^T u)_i > 0$ y $(N^T u)_i < 0$; en otro caso intercambiamos las columnas M_i y N_i y trabajamos con un cambio en columnas de $\{M, N\}$. Definimos vectores no negativos:

$x^* := (N^T u)^-$, $s^* := (N^T u)^+$, $y^* := (M^T u)^+$, $r^* := (M^T u)^-$; se verifica:
 $x^* = -N^T u + s^*$, $y^* = M^T u + r^*$, $(x^*)^T r^* = 0$, $(y^*)^T s^* = 0$.

Vamos a probar que para un vector p^* apropiado el par (x^*, y^*) es un vector de KKT para el programa bilineal $\text{BLP}(M, N, K + p^*)$

$$\begin{aligned} & \text{minimizar} && x^T y \\ & \text{sujeto a :} && x \geq 0 \\ & && y \geq 0 \\ & && Mx - Ny \in K + p^* (\Leftrightarrow LMx - LNy \geq b + Lp^*) \end{aligned}$$

y no es solución del $\text{ELCP}(M, N, K + p^*)$. Con este fin sea $c := L(Mx^* - Ny^*) - b$. Se verifica que el conjunto $\{p \in \mathbb{R}^m / Lp \leq c\}$ es igual a $(Mx^* - Ny^*) - K$ y es no vacío (p^* es de dicho conjunto). Como $u \in (0^+ K)^*$, el conjunto $\{\lambda \in \mathbb{R}^n / L^T \lambda = u, \lambda \geq 0\}$ es no vacío. Por la dualidad de la programación lineal, los problemas

$$\begin{aligned} & \text{minimizar} && \lambda^T c \\ & \text{sujeto a :} && L^T \lambda = u \quad y \\ & && \lambda \geq 0 \end{aligned} \quad \begin{aligned} & \text{maximizar} && p^T u \\ & \text{sujeto a :} && Lp \leq c \end{aligned}$$

tienen soluciones óptimas λ^* y p^* tales que $(\lambda^*)^T c = (p^*)^T u$.

La desigualdad $Lp^* \leq c$ demuestra que el par (x^*, y^*) es una solución factible para el programa $\text{BLP}(M, N, K + p^*)$. Como $(\lambda^*)^T (LMx^* - LNy^* - b - Lp^*) = (\lambda^*)^T (c - Lp^*) = (\lambda^*)^T c - (p^*)^T u = 0$ vemos que el par (x^*, y^*) es un vector de KKT del programa $\text{BLP}(M, N, K + p^*)$ con multiplicadores r^* , s^* , λ^* , sin embargo no es una solución del $\text{ELCP}(M, N, K + p^*)$ debido a que $(x^*)^T y^* \neq 0$ pues habíamos supuesto que para algún i $(M^T u)_i > 0$ y $(N^T u)_i < 0$, luego $x_i^* = (N^T u)_i^- > 0$, $y_i^* = (M^T u)_i^+ > 0$. \square

Observación 2.2.4. En la prueba de la implicación \Leftarrow del teorema anterior, se usó un cambio en columnas de $\{M, N\}$ para hacer $(M^T u)_i > 0$ y $(N^T u)_i < 0$. Se puede evitar este cambio si $(0^+ K)^*$ es un subespacio: podemos simplificar el trabajo con $-u$ en lugar de u . Como se verifica que $(0^+ K)^*$ es un subespacio si y sólo si $0^+ K$ es un subespacio, podemos enunciar el siguiente:

Teorema 2.2.4. Supongamos que $0^+ K$ es un subespacio de \mathbb{R}^m . Entonces $\{M, N\}$ posee la E -propiedad de suficiencia en filas con respecto a K si y sólo si para cada $p^* \in \mathbb{R}^m$, cada vector de KKT del programa bilineal $\text{BLP}(M, N, K + p^*)$ es una solución del $\text{ELCP}(M, N, K + p^*)$.

Definición 2.2.8. Decimos que el $ELCP(M,N,K)$ es factible si el conjunto de soluciones factibles del $BLP(M,N,K)$ es no vacío, es decir el conjunto [también llamado región factible del $ELCP(M,N,K)$] $\{(x,y) \in \mathbb{R}_+^{2n}/Mx - Ny \in K\}$ es no vacío.

Corolario 2.2.1. Supongamos que $\{M,N\}$ tiene la E-propiedad de suficiencia en filas con respecto a K . Entonces para cada p , la factibilidad del $ELCP(M,N,K+p)$ implica su solucionabilidad (es decir que el conjunto $\{(x,y) \in \mathbb{R}_+^{2n}/Mx - Ny \in K+p, x^T y = 0\}$ sea no vacío).

DEMOSTRACIÓN. Como el $ELCP(M,N,K+p)$ es factible, por definición el $BLP(M,N,K+p)$ lo es, luego ha de tener una solución óptima que debe ser de complementariedad, ya que como $\{M,N\}$ tiene la E-propiedad de suficiencia en filas con respecto a K , sabemos por el teorema 2.2.3, que cada vector de KKT del $BLP(\overline{M}, \overline{N}, K+p)$ (con $\overline{M} = M$, $\overline{N} = N$) resuelve el $ELCP(M,N,K+p)$, como consecuencia se verifica la solucionabilidad del $ELCP(M,N,K+p)$. \square

Definición 2.2.9. Dadas matrices $M, N \in \mathbb{R}^{m \times n}$ y dado K un conjunto poliédrico de \mathbb{R}^m decimos que $\{M,N\}$ tiene la P-propiedad con respecto a K si se verifican las siguientes condiciones:

$$(P_1) \quad \text{Si } [Mu - Nv \in K - K, u * v \leq 0] \Rightarrow (u, v) = (0, 0)$$

$$(P_2) \quad \text{Si } [(M^T u) * (N^T u) \leq 0, u \in (0^+ K)^*] \Rightarrow u = 0$$

Observación 2.2.5. En el contexto de $LCP(M,q)$, M tiene la P-propiedad si $x * Mx \leq 0 \Rightarrow x = 0$ (es decir M es una P-matriz).

Teorema 2.2.5. El par $\{M,N\}$ tiene la P-propiedad con respecto a K si y sólo si para cada $p \in \mathbb{R}^m$ y para cada cambio en columnas $\{\overline{M}, \overline{N}\}$ de $\{M,N\}$, el $ELCP(\overline{M}, \overline{N}, K+p)$ tiene solución única.

DEMOSTRACIÓN.

\Rightarrow Supongamos que se verifican las condiciones (P_1) y (P_2) de la definición 2.2.9. Como éstas se siguen verificando cuando K se reemplaza por $K+p$, y cuando (M,N) se reemplaza por cualquier cambio en las columnas $\{\overline{M}, \overline{N}\}$ de $\{M,N\}$, probaremos la existencia y unicidad de solución para el $ELCP(M,N,K)$. Si (x^*, y^*) y (\bar{x}, \bar{y}) son dos soluciones cualesquiera de este problema, entonces $(u, v) = (x^* - \bar{x}, y^* - \bar{y}) \in K - K$ y $u * v \leq 0$, por tanto por verificarse la condición (P_1) de la definición 2.2.9, $(u,v)=(0,0)$.

Esto prueba la unicidad de la solución. Probamos ahora la existencia de solución. Como la condición (P_2) de la definición 2.2.9 implica la E-propiedad de suficiencia en filas de $\{M, N\}$ con respecto a K , usando el corolario 2.2.1 solamente tenemos que probar la factibilidad del $\text{ELCP}(M, N, K)$. Supongamos que el $\text{ELCP}(M, N, K)$ no es factible, por tanto los conjuntos cerrados y convexos K y $M(\mathbb{R}_+^n) - N(\mathbb{R}_+^n)$ son disjuntos. Por un teorema de separación en [47] existe un vector no cero $d \in \mathbb{R}^m$ tal que: $d^T r \geq 0 \geq d^T (Ms - Nt)$, para todo $r \in K$, $s, t \in \mathbb{R}_+^n$.

Denotamos por $\text{ext}(K)$ el conjunto (finito) de todos los puntos extremos de K . Como $K = \text{conv}(\text{ext}(K)) + 0^+ K$ (donde “conv” se refiere a la envolvente convexa), vemos que $0 \neq d \in (0^+ K)^*$, $M^T d \leq 0$, $N^T d \geq 0$, lo cual contradice (P_2) de la definición 2.2.9.

\Leftarrow Supongamos que no se verifica (P_2) de la definición 2.2.9, entonces existe un vector no cero $u \in (0^+ K)^*$ tal que: $(M^T u) * (N^T u) \leq 0$. Por un cambio en columnas $\{\bar{M}, \bar{N}\}$ de $\{M, N\}$ podemos suponer que $\bar{M}u \leq 0$ y $\bar{N}u \geq 0$. Entonces tenemos: $0 \geq u^T(\bar{M}x - \bar{N}y)$ para todos $x, y \in \mathbb{R}_+^n$. Ahora podemos encontrar un vector p tal que $u^T(e + p) > 0$ para todo $e \in \text{ext}(K)$. Como $K = \text{conv}(\text{ext}(K)) + 0^+ K$ y $u \in (0^+ K)^*$, vemos que $u^T(k + p) > 0$ para todo $k \in K$. Pero esto significa que para x e y no negativos en \mathbb{R}^n , $\bar{M}x - \bar{N}y$ no puede estar en $K + p$. Esto contradice la hipótesis de que el $\text{ELCP}(\bar{M}, \bar{N}, K + p)$ tiene una (única) solución. Por tanto se cumple la condición (P_2) de la definición 2.2.9. \square

Teorema 2.2.6. Sean $M, N \in \mathbb{R}^{m \times n}$ y $K = \{u \in \mathbb{R}^m / Lu = b\}$ donde $L \in \mathbb{R}^{k \times m}$ tiene (sin pérdida de generalidad) rango k , y $b \in \mathbb{R}^k$. Entonces son equivalentes:

- I. $\{M, N\}$ tiene la P-propiedad con respecto a K .
- II. Para cada $p \in \mathbb{R}^m$, $\text{ELCP}(M, N, K + p)$ tiene solución única.
- III. LM y LN son cuadradas y $(LM)^{-1}(LN)$ es una P-matriz.

DEMOSTRACIÓN. Vemos que $K - K = 0^+ K = \{u / Lu = 0\}$ y $(0^+ K)^* = \text{Im}(L^T)$. Entonces las condiciones (P_1) y (P_2) de la definición 2.2.9 se convierten en:

$$(P'_1) \quad \text{Si } [LMu - LNv = 0, \quad u * v \leq 0] \Rightarrow (u, v) = (0, 0)$$

$$(P'_2) \quad \text{Si } [(M^T L^T x) * (N^T L^T x) \leq 0, \quad x \in \mathbb{R}^k] \Rightarrow x = 0$$

La implicación $\text{I} \Rightarrow \text{II}$ se sigue del teorema anterior. Cuando se verifica III, (P'_1) y

(P'_2) se verifican ya que una matriz cuadrada es una P-matriz si y sólo si su traspuesta es una P-matriz. Luego hemos probado $\text{III} \Rightarrow \text{I}$. Solamente nos falta probar $\text{II} \Rightarrow \text{III}$. Para simplificar, usamos la siguiente notación $A=LM$, $B=LN$. Entonces las condiciones (P'_1) y (P'_2) se convierten en:

$$(P''_1) \quad \text{Si } [Au - Bv = 0, \quad u * v \leq 0] \Rightarrow (u, v) = (0, 0)$$

$$(P''_2) \quad \text{Si } [(A^T x) * (B^T x) \leq 0, \quad x \in \mathbb{R}^k] \Rightarrow x = 0$$

Se puede probar $\text{II} \Rightarrow \text{III}$ basándonos en que $\text{ELCP}(M, N, K+p)$ es equivalente al problema de complementariedad lineal horizontal $\text{HLCP}(A, B, b+Lp)$, donde $b+Lp \in \mathbb{R}^k$ y para los HLCP se verifica que: Si para cada $q \in \mathbb{R}^k$, $\text{HLCP}(A, B, q)$ tiene solución única, entonces A y B son matrices cuadradas y además $A^{-1}B$ es una P-matriz. \square

2.3. Método del gradiente discreto

En la literatura, por ejemplo en [12, 43] podemos encontrar diferentes métodos tipo quasi-Newton para resolver problemas de optimización no suaves, pero se optó por usar el introducido en [8, 9] denominado método del gradiente discreto, pues se ajusta adecuadamente al problema que se trata de resolver en este trabajo. Comenzamos con la definición del gradiente discreto.

Sea f una función continua localmente Lipschitziana definida sobre \mathbb{R}^n . Sean $S_1 = \{g \in \mathbb{R}^n / \|g\| = 1\}$, $P = \{z/z(\lambda) \in \mathbb{R}^1, z(\lambda) > 0, \lambda > 0, \lambda^{-1}z(\lambda) \rightarrow 0, \lambda \rightarrow 0\}$. P es el conjunto de las funciones infinitesimales univariantes positivas. Sea $G = \{e \in \mathbb{R}^n / e = (e_1, e_2, \dots, e_n), |e_j| = 1, j = 1, \dots, n\}$ el conjunto de todos los vértices del hipercubo unidad en \mathbb{R}^n . Tomamos $g \in S_1$, $e \in G$, un número positivo $\alpha \in (0, 1]$ y calculamos $i = \operatorname{argmax}\{|g_k|, k = 1, \dots, n\}$, definimos vectores $e^j = e^j(\alpha)$, $j = 1, \dots, n$ de la siguiente forma: $e^j = (\alpha e_1, \alpha^2 e_2, \dots, \alpha^j e_j, 0, \dots, 0)$ y consideramos los puntos $x^0 = x + \lambda g$, $x^j = x^0 + z(\lambda)e^j(\alpha)$, $j = 1, \dots, n$.

Definición 2.3.1. *El gradiente discreto de la función f en el punto $x \in \mathbb{R}^n$ es el vector $\Gamma^i(x, g, e, z, \lambda, \alpha) = (\Gamma_1^i, \dots, \Gamma_n^i) \in \mathbb{R}^n$, $g \in S_1$ con las siguientes coordenadas:*

$$\Gamma_j^i = [z(\lambda)\alpha^j e_j]^{-1} [f(x^j) - f(x^{j-1})], \quad j = 1, \dots, n, \quad j \neq i,$$

$$\Gamma_i^i = (\lambda g_i)^{-1} \left[f(x + \lambda g) - f(x) - \lambda \sum_{j=1, j \neq i}^n \Gamma_j^i g_j \right].$$

Nota 2.3.1. *De la definición anterior se sigue que:*

$$f(x + \lambda g) - f(x) = \lambda \langle \Gamma^i(x, g, e, z, \lambda, \alpha), g \rangle, \quad \forall g \in S_1, e \in G, z \in P, \lambda > 0, \alpha > 0.$$

Se expone a continuación un algoritmo para el cálculo de la dirección de descenso. Sean $z \in P$, $\lambda > 0$, $\alpha \in (0, 1]$, el número $c \in (0, 1)$ y una tolerancia $\delta > 0$ dados.

Algoritmo 2.3.1. *(Algoritmo para el cálculo de la dirección de descenso)*

1. Escoger un $g^1 \in S_1$, $e \in G$, calcular $i = \operatorname{argmax}\{|g_j|, j = 1, \dots, n\}$ y un gradiente discreto $v^1 = \Gamma^i(x, g^1, e, z, \lambda, \alpha)$. Sea el conjunto $\bar{D}_1(x) = \{v^1\}$ y $k=1$.
2. Calcular el vector $\|w^k\|^2 = \min\{\|w\|^2 / w \in \bar{D}_k(x)\}$. Si $\|w^k\| \leq \delta$, entonces se para. En otro caso se va al paso 3.
3. Calcular la dirección de búsqueda mediante $g^{k+1} = -\|w^k\|^{-1} w^k$

4. Si $f(x + \lambda g^{k+1}) - f(x) \leq -c\lambda\|w^k\|$ entonces se para. En otro caso se va al paso 5.
5. Calcular $i = \operatorname{argmax}\{|g_j^{k+1}|, j = 1, \dots, n\}$ y un gradiente discreto $v^{k+1} = \Gamma^i(x, g^{k+1}, e, z, \lambda, \alpha)$, construir el conjunto $\overline{D}_{k+1}(x) = \operatorname{co}\{\overline{D}_k(x) \cup \{v^{k+1}\}\}$, $k=k+1$ e ir al paso 2.

Nota 2.3.2. En el paso 1 del algoritmo anterior se calcula el gradiente discreto con respecto a una dirección inicial $g^1 \in \mathbb{R}^n$. La distancia entre el cono convexo $\overline{D}_k(x)$ de todos los gradientes discretos calculados y el origen se halla en el paso 2. Este problema se resuelve usando el algoritmo de [72]. Si la distancia es menor que la tolerancia $\delta > 0$ entonces se acepta el punto x como punto estacionario aproximado (paso 2), en otro caso se calcula otra dirección de búsqueda en el paso 3. En el paso 4 se comprueba si esta dirección es una dirección de descenso. Si lo es, se para y ya tenemos calculada la dirección de descenso, en otro caso se calcula otro gradiente discreto en esta dirección en el paso 5 y se actualiza el conjunto $\overline{D}_k(x)$.

Nota 2.3.3. Se puede probar que el algoritmo anterior termina después de un número finito de iteraciones [10].

Sean sucesiones $\delta_k > 0$, $z_k \in P$, $\lambda_k > 0$, $\delta_k \rightarrow 0^+$, $z_k \rightarrow 0^+$, $\lambda_k \rightarrow 0^+$, $k \rightarrow \infty$, un número suficientemente pequeño $\alpha > 0$ y números $c_1 \in (0, 1)$, $c_2 \in (0, c_1]$ dados.

Algoritmo 2.3.2. (Método del gradiente discreto)

1. Se escoge un punto de inicio $x^0 \in \mathbb{R}^n$ y sea $k=0$.
2. Sea $s=0$ y $x_s^k = x^k$.
3. Aplicar el algoritmo 2.3.1 para el cálculo de la dirección de descenso en $x = x_s^k$, $\delta = \delta_k$, $z = z_k$, $\lambda = \lambda_k$, $c = c_1$. Este algoritmo termina después de un número finito de iteraciones $l > 0$. Como resultado, se consigue el conjunto $\overline{D}_l(x_s^k)$ y un elemento v_s^k tal que $\|v_s^k\|^2 = \min\{\|v\|^2 / v \in \overline{D}_l(x_s^k)\}$. Además, o $\|v_s^k\| \leq \delta_k$ o para la dirección de búsqueda $g_s^k = -\|v_s^k\|^{-1}v_s^k$

$$f(x_s^k + \lambda_k g_s^k) - f(x_s^k) \leq -c_1 \lambda_k \|v_s^k\|$$

4. Si $\|v_s^k\| \leq \delta_k$, entonces $x^{k+1} = x_s^k$, $k=k+1$ y vamos al paso 2. En otro caso vamos al paso 5.

5. Se calcula $x_{s+1}^k = x_s^k + \sigma_s g_s^k$, donde σ_s está definido así:

$$\sigma_s = \operatorname{argmax} \{ \sigma \geq 0 / f(x_s^k + \sigma g_s^k) - f(x_s^k) \leq -c_2 \sigma \|v_s^k\| \}.$$

6. $s=s+1$ y vamos al paso 3.

2.4. Métodos heurísticos

Como se verá en el capítulo siguiente, el problema de regulación del tráfico a través del ajuste de los ciclos de los semáforos, es intrínsecamente difícil de resolver de manera óptima, como alternativa surgen los procedimientos heurísticos que proporcionan soluciones aceptablemente buenas, aunque no necesariamente óptimas, en tiempos de cálculo razonables. Se podrían encontrar diversas definiciones para el concepto heurístico, de las que presentamos tres a continuación:

“Procedimientos simples, a menudo basados en el sentido común, que tienden a ofrecer una buena solución (aunque no necesariamente la óptima) a problemas difíciles, de un modo fácil y rápido” [78].

“Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución” [2].

“Técnica que explora buenas soluciones, cercanas al óptimo, en un tiempo computacional razonable sin poder garantizar ni factibilidad ni optimalidad y en muchos casos incapaz de indicar la distancia a la que se queda del óptimo” [4].

2.4.1. Ventajas e inconvenientes de la utilización de métodos heurísticos

Para aplicar algún método de optimización a problemas del mundo real, es necesario construir un modelo de la realidad que será lo que se tratará de resolver en vez del problema en sí. Los métodos heurísticos suelen ser más flexibles a la hora de representar funciones objetivo (o multiobjetivo) de los problemas del mundo real que los métodos exactos (no requieren especificación matemática completa del sistema). Aunque la solución obtenida no sea exacta, muchas veces no se puede probar que sea peor que la que se obtendría mediante un método exacto, puesto que el modelo utilizado para solucionar éste será probablemente más lejano al mundo real. Por ello siempre nos quedará la pregunta ¿preferimos una solución exacta a un modelo aproximado a una solución aproximada a un modelo exacto? [50].

Además de la reflexión anterior, las razones para la utilización de métodos heurísticos pueden ser muy diversas:

- Para muchos de los problemas de optimización más comunes no existe un método exacto de resolución o el tiempo de cálculo requerido es desorbitado.
- Existen casos de problemas en los que la solución óptima no es indispensable; el objetivo es simplemente buscar una solución de calidad aceptable. En estos casos puede que no merezca la pena el esfuerzo de encontrarla.
- Además de los requerimientos de tiempo, los métodos exactos a veces tienen fuertes necesidades de memoria y puede ocurrir que en el entorno en que se esté no se disponga del mismo.
- Los métodos heurísticos resultan a menudo muy útiles como paso intermedio en la utilización de otro algoritmo.
- Como se puede intuir, estos métodos permiten mayor flexibilidad en el manejo de las características del problema y son más explicativos que los métodos exactos.

Por otra parte, no puede quedar sin mencionar que no todo son ventajas, puesto que, así como con los métodos exactos se puede asegurar que la solución hallada es la mejor solución que se puede obtener para el modelo resuelto, al utilizar métodos heurísticos no se puede conocer la calidad de la solución obtenida porque no conoceremos el óptimo. Así que, en caso de que exista un método exacto factible y queramos asegurar que la solución obtenida es óptima, será preferible utilizarlo.

2.5. Metaheurísticas: Definición y clasificación

Tradicionalmente para resolver un problema dado, se diseñaba un algoritmo específico. Hoy en día, el interés primordial de los investigadores es el de diseñar métodos generales que sirvan para resolver clases o categorías de problemas. Dado que estos métodos generales sirven para construir o guiar el diseño de métodos que resuelvan problemas específicos, se les ha dado el nombre de metaheurísticas. La característica general que comparten las técnicas metaheurísticas es la de guiar una heurística subordinada de acuerdo con algún concepto derivado de la inteligencia artificial, la biología, las matemáticas o la física, para mejorar su rendimiento.

Osman ([54]) proporciona una definición acertada de metaheurística: “Dentro de la clase denominada metaheurísticas se incluyen todos aquellos procedimientos que en un proceso iterativo, guían a una heurística subordinada combinando inteligentemente diferentes conceptos tomados de analogías de la naturaleza, y exploran el espacio de soluciones utilizando estrategias de aprendizaje para estructurar la información, con el objeto de encontrar eficientemente soluciones cercanas al óptimo”.

Las metaheurísticas pueden clasificarse en los siguientes grupos según González García, J. L. ([29]):

- **De relajación:** utilizan modelos relajados del original para la solución del problema. Se entiende por relajación la modificación, flexibilización o eliminación de restricciones del problema original.
- **Constructivas:** tratan de obtener soluciones al problema a partir del análisis y selección paulatina de las componentes que las forman. Se parte de una estructura de solución vacía a la que se incorporan iterativamente sus diferentes elementos. De este estilo es por ejemplo GRASP (Greedy randomized adaptative search procedures), que modifica una estrategia *greedy* incluyendo pasos aleatorios para seleccionar los elementos a incluir en una solución.
- **De búsqueda:** procedimientos que emplean transformaciones denominadas *movimientos* para recorrer el espacio de búsqueda cercano a una solución (vecindario de la solución actual). Pueden distinguirse varios subtipos:
 - *De búsqueda local:* establecen pautas de selección iterativa de las soluciones vecinas a la actual que dan lugar a búsquedas heurísticas de alto rendimiento.

- *De búsqueda global:* emplean diversos métodos para escapar de óptimos locales. Tres de ellos son:
 - Re-arrancar la búsqueda desde otra posición del espacio de soluciones (metaheurísticas de arranque múltiple o *Multistart*).
 - Modificar la estructura del entorno de una solución (búsqueda por entornos variables o *variable neighborhood search*).
 - Permitir soluciones que no mejoran el valor de la función objetivo de la solución actual con cierta probabilidad.
- *De búsqueda basada en poblaciones:* en lugar de recorrer el espacio de soluciones partiendo de un único punto, es una población de soluciones iniciales la que recorre el espacio de manera simultánea e interactuando entre ellas. Contemplan mecanismos (operadores) para generar nuevas soluciones a partir de las ya existentes. A este grupo pertenecen por ejemplo los algoritmos meméticos, la búsqueda dispersa (*scatter search*), o los algoritmos de hormigas.
- **Evolutivas:** en éstas, un conjunto de soluciones evoluciona en el espacio de búsqueda estableciendo procedimientos de guiado del proceso. Se distinguen por la forma en que combinan la información de las diferentes soluciones para obtener otras nuevas. A este grupo pertenecen por ejemplo los algoritmos genéticos.
- **De memoria a largo plazo:** metaheurísticas de aprendizaje que emplean la memoria del proceso de búsqueda para mejorar su rendimiento. El paradigma de este tipo de algoritmos es la búsqueda tabú.

2.6. Recocido simulado (Simulated annealing)

La idea básica de Recocido Simulado es la búsqueda global que permite movimientos ascendentes para evitar quedar atrapado permaturamente en un óptimo local y fue introducida por Metrópolis ([49]) en un algoritmo para simular el proceso de enfriamiento de un material.

El método del *recocido* se utiliza en la industria para obtener materiales más resistentes o más cristalinos, en general, para mejorar las cualidades de un material. El proceso consiste en *derretir* el material (calentarlo a temperatura muy alta). En esa situación los átomos adquieren una distribución *azarosa* dentro de la estructura del material y la energía del sistema es máxima. Luego se hace descender la temperatura muy lentamente por etapas, dejando que en cada una de esas etapas los átomos queden en equilibrio (es decir, que los átomos alcancen una configuración óptima para esa temperatura). Al final del proceso, los átomos forman una estructura cristalina altamente regular, el material alcanza así una máxima resistencia y la energía del sistema es mínima. Experimentalmente se comprueba que si se hace descender la temperatura bruscamente, o si no se espera suficiente tiempo en cada etapa, al final la estructura del material no es la óptima.

Metrópolis modeló el proceso de recocido, simulando los cambios energéticos en un sistema de partículas conforme decrece la temperatura, hasta que converge a un estado estable (congelado). Las leyes de la termodinámica dicen que a una temperatura T la probabilidad de un incremento energético de magnitud ΔE se puede aproximar por

$$P(\Delta E) = e^{\frac{-\Delta E}{K_B T}} \quad (2.2)$$

donde K_B es una constante física denominada de Boltzmann y T es la temperatura del sistema. En el algoritmo de Metrópolis se genera una perturbación aleatoria en el sistema y se calculan los cambios de energía resultantes: si hay una caída energética, el cambio se acepta automáticamente; por el contrario, si se produce un incremento energético, el cambio será aceptado con una probabilidad dada por 2.2. El proceso se repite durante un número predefinido de iteraciones en series decrecientes de temperaturas, hasta que el sistema esté "frío". Conforme disminuye la temperatura disminuye la probabilidad de aceptar configuraciones que supongan un incremento de la energía del sistema. Así pues, a temperaturas altas son las características del sistema que más afectan a su estado energético las que guían el proceso, mientras que al disminuir la temperatura son los detalles de la configuración los que predominan en la evolución.

2.6.1. Metaheurística basada en Recocido Simulado

Kirkpatrick, Gelatt y Vecchi ([58]) en sus publicaciones sobre circuitos VLSI, y Cerny ([14]) para el TSP independientemente, sugirieron que el tipo de simulación descrito en el apartado anterior, se podía utilizar para encontrar soluciones factibles de un problema de optimización, con el objetivo de llegar a una solución óptima, asociando conceptos clave del proceso original de simulación, con elementos de optimización combinatoria según se indica en la tabla 2.1

Simulación termodinámica	Optimización combinatoria
Estados del sistema	Soluciones factibles
Energía	Coste
Cambio de estado	Solución en el entorno
Temperatura	Parámetro de control
Estado congelado	Solución heurística

Tabla 2.1: Simulación termodinámica y optimización combinatoria

La exploración del conjunto de posibles soluciones se realiza teniendo en cuenta una estructura de vecindades pre-establecida, de modo que la evaluación de la diferencia entre las evaluaciones de la función objetivo entre dos soluciones consecutivas sea calculable de modo eficiente. La técnica de recocido simulado posee una elevada capacidad para escapar de los mínimos locales gracias al empleo del criterio Metrópolis de aceptación de configuraciones de mayor energía, es decir un modo de evitar que la búsqueda finalice en un óptimo local, que no tiene por qué ser forzosamente el global, es permitir que algunos movimientos sean hacia soluciones peores. Por tanto, cualquier implementación de búsqueda local-global puede convertirse en una implementación recocido simulado al elegir elementos del entorno de modo aleatorio, aceptar automáticamente todos los movimientos hacia una mejor solución, y aceptar los movimientos a una solución peor de acuerdo con una probabilidad dada por 2.2. La constante de Boltzmann K_B , en general no se considera, debido a que no tiene significado en los problemas de optimización. Por tanto, basándonos en Dowsland ([24]) podemos definir un algoritmo básico de recocido simulado para problemas de optimización de la manera siguiente:

Sea F la función objetivo, $F(s)$ el coste de la solución s y $N(s)$ su entorno.

Seleccionar una solución inicial s_0 ;

Seleccionar una temperatura inicial $T_0 > 0$;
 Seleccionar una función de reducción de la temperatura α ;
 Seleccionar un número de iteraciones “nrep” para cada valor de la temperatura;
 Seleccionar un criterio de parada (valor final de la temperatura);
 Seleccionar una estructura $N(s_0)$ de soluciones vecinas a s_0 .

REPETIR

 REPETIR

 Seleccionar aleatoriamente una solución $s \in N(s_0)$;

 Sea $\Delta F = F(s) - F(s_0)$;

 Si $\Delta F < 0$ entonces $s_0 = s$

 SI NO

 Generar aleatoriamente $u \in U(0, 1)$;

 Si $u < e^{\frac{-\Delta F}{T}}$ entonces $s_0 = s$;

 FIN

 SI NO

HASTA QUE cuenta iteraciones = nrep

$T = \alpha(T)$

HASTA QUE criterio de parada = CIERTO

La mejor solución visitada será la solución heurística dada por el algoritmo.

2.6.1.1. Generación de la solución inicial

Asensio Cuesta ([5]) indica que para lograr que la evolución del algoritmo sea independiente de la solución inicial empleada (característica primordial de cualquier heurística), las primeras iteraciones se realizan a una temperatura muy elevada. De esta manera se consigue que en sus primeros pasos se recorran soluciones muy alejadas entre sí. La forma de generar la solución inicial s_0 es, habitualmente, aleatoria. Dado que la elevada temperatura de comienzo permitirá la aceptación de soluciones peores que la inicial, no es necesaria la búsqueda de una buena solución de partida mediante alguna heurística preliminar. En algunos casos se inicia el algoritmo desde varios puntos del espacio para

lograr una exploración más exhaustiva, y en otros se toma como punto de partida la mejor solución de una muestra aleatoria.

2.6.1.2. Determinación de la temperatura inicial

Para que la solución final del algoritmo sea independiente de la solución inicial de la que se parte, la temperatura inicial debería ser independiente de la solución inicial y lo suficientemente alta como para aceptar casi libremente las soluciones del entorno. En aquellos problemas que están bien estructurados, por lo general hay suficiente información como para estimar este valor. Por ejemplo, si el mayor incremento entre soluciones vecinas es conocido, es habitual escoger T_0 de tal forma que la probabilidad de aceptación sea $P = e^{\frac{-\Delta F}{T_0}} \approx 1$. Cuando esto no sea posible, la razón entre movimientos aceptados y rechazados que dé lugar a un estado inicial aceptablemente volátil puede ser fijado de antemano, y entonces el sistema podría ser calentado rápidamente hasta que esa razón alcance el valor deseado. Ese sería entonces el valor elegido T_0 para la temperatura inicial, a partir de la cual comenzaría el proceso de recocido.

2.6.1.3. Condición de equilibrio y esquema de enfriamiento

La velocidad a la que se produce el enfriamiento es otro factor clave en el éxito de la estrategia. Viene determinado, por una parte, por el número de iteraciones “nrep” que se ejecutarán a cada temperatura, y por otra, por la velocidad α a la que se realizará el enfriamiento. La teoría sugiere que se debería permitir que el sistema esté cerca de su estado estacionario correspondiente a la temperatura actual, antes de reducir ésta, y que además la temperatura vaya gradualmente acercándose al valor 0. En la definición de α debe llegarse a un compromiso entre una disminución rápida, que puede producir una convergencia prematura del algoritmo, y una evolución lenta en exceso que prolonga demasiado el tiempo de cálculo.

El esquema de enfriamiento más sencillo es una expresión de tipo geométrico [23] como: $T_M = \alpha T_{M-1}$ en la que α es un número real perteneciente a (0,1), lo habitual es que oscile entre 0,5 y 0,99.

Otro programa de enfriamiento es el propuesto por Lundy y Mees ([46]), en el que se ejecuta una sola iteración para cada temperatura, pero ésta se reduce a una velocidad muy lenta, según la fórmula $T_M = \frac{T_{M-1}}{1+\beta T_{M-1}}$, siendo β un valor muy pequeño.

La evidencia empírica y los resultados teóricos dan a entender que los detalles concretos del programa de enfriamiento no son tan importantes como la velocidad a la que se reduce la temperatura, y que tanto el programa geométrico como el de Lundy y Mees darán resultados similares cuando el enfriamiento se produzca para el mismo rango de temperaturas y para un número total de iteraciones similares.

2.6.1.4. El criterio de parada (criterio de congelación)

Existen diferentes criterios en base a los cuales el algoritmo detendrá su evolución. El más sencillo de todos es predefinir un número de iteraciones a completar antes de la detención. Otro es definir una temperatura de congelación, alcanzada la cuál, el algoritmo se detiene. Teóricamente T debería reducirse hasta 0, pero en la práctica la búsqueda converge, por lo general, a su óptimo local final bastante antes de ese valor nulo de la temperatura. Por tanto, si la temperatura de parada se fija muy baja, gastaremos mucho tiempo de búsqueda en las fases finales, que seguramente sería mejor aprovechado en temperaturas superiores. Al contrario, si la temperatura final se fija en un valor muy alto, es posible que la búsqueda no consiga alcanzar ningún óptimo local.

En [35] se propone definir una constante k , que indica un número de etapas consecutivas, tras la cuál se detendrá el proceso si no se produce una mejora de la última solución obtenida en un determinado porcentaje.

2.6.1.5. Definición del entorno de solución

El simulated annealing es un proceso de búsqueda por entornos. Se define el entorno N de una solución s_0 , como todas aquellas soluciones accesibles directamente desde s_0 .

Algunas de las primeras teorías sobre simulated annealing se basaban en entornos uniformes y simétricos, es decir, todos los entornos eran del mismo tamaño y si $i \in N(j) \Rightarrow j \in N(i)$. Sin embargo, hoy se sabe que es suficiente con exigir el cumplimiento de una condición más suave, que exige que cualquier solución pueda alcanzarse desde cualquier otra a través de una cadena de movimientos válidos, usando los entornos. Es decir, es importante garantizar esta condición de *alcanzabilidad* (o *ergodicidad*) al definir la estructura del espacio de soluciones y de los entornos.

Los algoritmos básicos emplean un método probabilista de exploración del entorno,

lo cuál puede provocar la aparición de ciclos exploratorios visitando sucesivamente las mismas soluciones. Implementaciones más elaboradas del algoritmo emplean técnicas para evitar la aparición de ciclos, como el empleo de listas de soluciones ya visitadas. En otros casos se realiza una exploración ordenada del entorno de las soluciones, resultando este método más eficiente que el enfoque probabilista clásico [15].

Capítulo 3

Modelos para cruce simple (2 fases)

3.1. Antecedentes

3.1.1. Introducción

En esta sección se describe el modelo para un cruce simple (2 fases) expuesto por Schutter-Moor en [21]. Se presenta primero, un modelo exacto que describe la evolución de las longitudes de las colas como función del tiempo y de los instantes de cambio ("switching"). Los datos de entrada para el modelo son las tasas de entrada y salida de los vehículos en el cruce. Esas cantidades se pueden determinar mediante detectores de inducción eléctricos, sensores de ultrasonidos o sistemas de visión, en combinación con datos históricos. Es obvio que en la práctica esta infraestructura de sensores debería tener un coste bajo, alto rendimiento y fiabilidad. Seguidamente y con el objetivo de obtener un modelo simplificado (más manejable para el análisis matemático) que será la base de los métodos eficientes para calcular esquemas óptimos para los semáforos, se hacen algunas hipótesis simplificadoras, explicándose por qué y cuando la mayoría de esas hipótesis son razonables y como se pueden tener en cuenta desviaciones de dichas hipótesis. Una vez expuesto el modelo simplificado, se pretende calcular una sucesión temporal switching para semáforos que minimice un criterio, tal como la longitud media de la cola sobre todas las colas, la longitud de la cola en el peor caso, el tiempo medio de espera, etc. lo que lleva a un aumento del flujo de tráfico y a una disminución de los efectos de la congestión de tráfico. Se interpretan las restricciones del problema como un problema de complementariedad lineal extendido, por tanto para determinar la sucesión temporal switching óptima, se debe optimizar la función objetivo en la solución del

problema de complementariedad lineal extendido, como éste es un problema NP-duro y además como los algoritmos para resolverlo (como el propuesto en [20]) requieren tiempos de ejecución exponenciales, no son factibles cuando el número de ciclos es grande, por lo que se discuten varias técnicas para calcular eficientemente subóptimos y aproximar las sucesiones temporales switching. Finalmente se propone un método híbrido de solución [3] que consiste en una heurística combinada con un algoritmo eficiente para resolver un LCP.

3.1.2. El ejemplo y su modelización

3.1.2.1. El ejemplo

Sea un cruce de dos carreteras con los dos sentidos de circulación, en el que confluyen cuatro carriles L_1, L_2, L_3 y L_4 y en el que en cada esquina hay un semáforo (T_1, T_2, T_3 y T_4), (Ver figura 3.1).

Para cada semáforo y en cada ciclo hay tres estados consecutivos: verde, ámbar y rojo. La tasa de llegada de coches en el carril L_i en el tiempo t es $\lambda_i(t)$ para $i=1,2,3,4$. Cuando el semáforo está verde (respectivamente ámbar) la tasa de salida en el carril L_i en el tiempo t es $\mu_i(t)$ (respectivamente $\kappa_i(t)$) para $i=1,2,3,4$. Se supone que la luz ámbar permanece encendida durante un espacio de tiempo fijo e igual a δ_{amb} (este tiempo se determina según normas establecidas, hay distintos tipos de norma: americana, europea, etc; en general fluctúa entre los 3 y los 4 segundos). La luz verde permanece encendida durante un espacio de tiempo variable, que se calcula según: la velocidad admitida en las vías, el número de carriles, la afluencia de vehículos, etc.; fluctúa entre los 20 y los 30 segundos. Se considera que la luz roja de un semáforo está encendida durante un espacio de tiempo igual a la suma de los tiempos en que están encendidas las luces verde y ámbar de los semáforos situados en las vías opuestas (Ver tabla 3.1).

Sea un intervalo de tiempo $[t_0, t_N]$ y un número N de cambios de señales previstos. Sean t_0, t_2, t_4, \dots los instantes de tiempo en los que los semáforos T_1 y T_3 cambian de ámbar a rojo (y T_2 y T_4 cambian de rojo a verde). Sean t_1, t_3, t_5, \dots los instantes de tiempo en los cuales los semáforos T_1 y T_3 cambian de rojo a verde (y T_2 y T_4 cambian de ámbar a rojo) (Ver tabla 3.1).

Para modelizar estas situaciones se consideran: $G_1 = G_3 = \{1, 3, \dots\}$ y $G_2 = G_4 = \{0, 2, \dots\}$. Así si $k \in G_i$ entonces el semáforo T_i cambia de rojo a verde en el instante

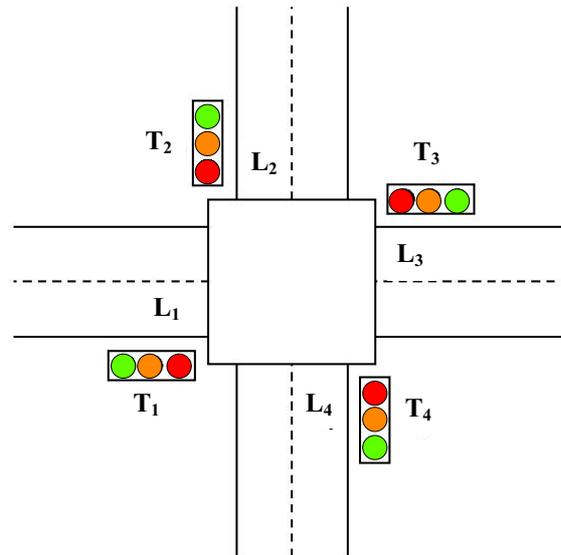


Figura 3.1: Cruce simple

Período	T_1	T_2	T_3	T_4
$[t_0, t_1 - \delta_{amb})$	rojo	verde	rojo	verde
$[t_1 - \delta_{amb}, t_1)$	rojo	ámbar	rojo	ámbar
$[t_1, t_2 - \delta_{amb})$	verde	rojo	verde	rojo
$[t_2 - \delta_{amb}, t_2)$	ámbar	rojo	ámbar	rojo
$[t_2, t_3 - \delta_{amb})$	rojo	verde	rojo	verde
$[t_3 - \delta_{amb}, t_3)$	rojo	ámbar	rojo	ámbar
\vdots	\vdots	\vdots	\vdots	\vdots

Tabla 3.1: Esquema de cambios (switching) en los semáforos

t_k y está verde en el período $(t_k, t_{k+1} - \delta_{amb})$. Es decir en este modelo aparecen dos tipos de situaciones diferentes (2 fases) que se van repitiendo alternadamente: una en la cual los semáforos T_1 y T_3 están en rojo y por tanto no pueden rebasar el cruce los vehículos que circulan por los carriles L_1 y L_3 , mientras que los semáforos T_2 y T_4 están en verde en el período inicial $[t_{2k}, t_{2k+1} - \delta_{amb})$ y en ámbar en el período final $[t_{2k+1} - \delta_{amb}, t_{2k+1})$, $k = 0, 1, 2, \dots, \lfloor \frac{N-1}{2} \rfloor$, y entonces si pueden rebasar el cruce los coches que circulan por los carriles L_2 y L_4 ; y otra en la cual los semáforos T_1 y T_3 están en verde en el período inicial $[t_{2k+1}, t_{2k+2} - \delta_{amb})$ y en ámbar en el período final $[t_{2k+2} - \delta_{amb}, t_{2k+2})$, $k = 0, 1, 2, \dots, \lfloor \frac{N}{2} \rfloor - 1$ por tanto pueden rebasar el cruce los vehículos que circulan por los carriles L_1 y L_3 , y se detienen los que circulan por L_2 y L_4 , pues durante estos períodos los semáforos T_2 y T_4 se encuentran en rojo. Definimos $\delta_k = t_{k+1} - t_k$, $k = 0, 1, 2, \dots, N - 1$. Es obvio que tenemos $0 \leq \delta_{amb} < \delta_k$ para todo k . Además $\lambda_i(t)$, $\mu_i(t)$, $\kappa_i(t) \geq 0$ para todos i, t y $t_k < t_{k+1}$ para todo k .

Sea $l_i(t)$ la longitud de la cola (es decir el número de coches esperando) en el carril L_i en el instante de tiempo t . Está claro que $l_i(t) \geq 0$ para todos i, t .

3.1.2.2. El modelo exacto

Sea, por ejemplo, el carril L_1 . Cuando el semáforo T_1 está rojo, hay llegadas al carril L_1 (caracterizadas por la función tasa de llegada λ_1) y no hay salidas. Cuando el semáforo T_1 está verde (o ámbar), hay llegadas y salidas en el carril L_1 . En ese caso la tasa neta de crecimiento de la cola en el tiempo t es $\lambda_1(t) - \mu_1(t)$, (respectivamente $\lambda_1(t) - \kappa_1(t)$). Además, la evolución de la longitud de cola en el carril L_1 está dada por:

$$\frac{dl_1(t)}{dt} = \begin{cases} \lambda_1(t) & \text{si } t \in (t_{2k}, t_{2k+1}) \\ \lambda_1(t) - \mu_1(t) & \text{si } t \in (t_{2k+1}, t_{2k+2} - \delta_{amb}) \\ \lambda_1(t) - \kappa_1(t) & \text{si } t \in (t_{2k+2} - \delta_{amb}, t_{2k+2}) \end{cases} \quad (3.1)$$

para $k=0,1,2,\dots$. Esto implica que la relación entre las longitudes de las colas y los instantes de tiempo switching viene dada por:

$$l_1(t_{2k+1}) = l_1(t_{2k}) + \int_{t_{2k}}^{t_{2k+1}} \lambda_1(t) dt \quad (3.2)$$

$$l_1(t_{2k+2}) = l_1(t_{2k+1}) + \int_{t_{2k+1}}^{t_{2k+2} - \delta_{amb}} (\lambda_1(t) - \mu_1(t)) dt + \int_{t_{2k+2} - \delta_{amb}}^{t_{2k+2}} (\lambda_1(t) - \kappa_1(t)) dt \quad (3.3)$$

para $k=0,1,2,\dots$. Podemos escribir ecuaciones análogas para la evolución de las longitudes de cola en los otros carriles.

3.1.2.3. Un modelo simplificado

El modelo descrito en el apartado anterior no es manejable para el análisis matemático, por tanto se introducen algunas hipótesis con las que se obtiene un modelo (aproximado) más simple que se puede analizar muy fácilmente y para el cual se puede calcular una sucesión temporal switching óptima para los semáforos.

Desde ahora se introducen las siguientes hipótesis:

- Las longitudes de las colas son variables continuas.
- La tasa media de llegadas y salidas de coches es constante (o varía muy despacio en el tiempo).
- Para cada carril, la tasa media de salidas durante la fase verde es mayor o igual que la tasa media de salidas durante la fase ámbar.

Las dos primeras hipótesis merecen algunas consideraciones:

- El propósito principal del modelo que se expone es calcular esquemas óptimos de control de semáforos. El diseño de sucesiones temporales switching óptimas para semáforos es sólo útil si las tasas de llegada y de salida de vehículos en el cruce son altas. En ese caso, la aproximación de las longitudes de cola mediante variables continuas sólo introduce pequeños errores. Además, también hay cierta incertidumbre y variación en el tiempo en las tasas de salida y de llegada, lo cual hace que, en general, calcular una sucesión exacta switching óptima para semáforos sea una utopía. Por otro lado, en la práctica interesa más, obtener rápidamente una buena aproximación de la sucesión temporal switching óptima para semáforos, que gastar una gran cantidad de tiempo en obtener la sucesión exacta switching óptima para semáforos.
- Si se tiene en cuenta que el propósito principal del modelo que se obtiene es el diseño de una sucesión switching óptima para semáforos, entonces la hipótesis de que la tasa media de llegadas y salidas sea constante no es una restricción seria, teniendo en cuenta que usamos una estrategia de horizonte móvil: es decir, se calcula el esquema switching óptimo para semáforos por ejemplo los próximos 10 ciclos, basándonos en una predicción de las tasas medias de llegadas y salidas (usando datos medidos durante los ciclos anteriores) y se aplica este esquema

durante el primero de los 10 ciclos, mientras tanto se actualizan las estimaciones de las tasas de llegada y de salida y se calcula un nuevo esquema óptimo para los 10 ciclos siguientes y así sucesivamente.

Sea $\bar{\lambda}_i$ la tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo). Sea $\bar{\mu}_i$ (respectivamente $\bar{\kappa}_i$) la tasa media de salida en el carril L_i cuando el semáforo está verde (respectivamente ámbar) y la longitud de cola es mayor que 0 (es decir, cuando hay coches esperando o llegando al carril L_i). Así a partir de ahora suponemos que:

$$\lambda_i(t) = \bar{\lambda}_i \text{ para todo } t$$

$$\mu_i(t) = \begin{cases} \bar{\mu}_i & \text{si } t \in \bigcup_{k \in G_i} (t_k, t_{k+1} - \delta_{amb}) \text{ y } l_i(t) > 0 \\ 0 & \text{en otro caso} \end{cases}$$

$$\kappa_i(t) = \begin{cases} \bar{\kappa}_i & \text{si } t \in \bigcup_{k \in G_i} (t_{k+1} - \delta_{amb}, t_{k+1}) \text{ y } l_i(t) > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Nótese que $\bar{\lambda}_i$, $\bar{\mu}_i$ y $\bar{\kappa}_i$ son constantes (posiblemente sobre un período limitado de tiempo si aplicamos una estrategia de horizonte móvil). Además $\bar{\lambda}_i \geq 0$, $\bar{\mu}_i \geq 0$, $\bar{\kappa}_i \geq 0$ y $\bar{\kappa}_i \leq \bar{\mu}_i$.

Hemos de tener en cuenta que si se consideran (3.1) y (3.2)-(3.3), entonces la longitud de cola nunca es negativa. Sin embargo, si se usan las tasas medias de llegada y de salida en la descripción de la evolución de las longitudes de cola, entonces hemos de incluir explícitamente la condición de no negatividad cuando escribamos más adelante las ecuaciones (aproximadas) de evolución de las longitudes de cola. Como consecuencia se tiene:

$$l_1(t_{2k+1}) = l_1(t_{2k}) + \bar{\lambda}_1 \delta_{2k}$$

$$l_1(t_{2k+2} - \delta_{amb}) = \max(l_1(t_{2k+1}) + (\bar{\lambda}_1 - \bar{\mu}_1)(\delta_{2k+1} - \delta_{amb}), 0) \quad (3.4)$$

$$l_1(t_{2k+2}) = \max(l_1(t_{2k+2} - \delta_{amb}) + (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{amb}, 0) = \quad (3.5)$$

$$= \max(l_1(t_{2k+1}) + (\bar{\lambda}_1 - \bar{\mu}_1)\delta_{2k+1} + (\bar{\mu}_1 - \bar{\kappa}_1)\delta_{amb}, (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{amb}, 0)$$

para $k=0,1,2,\dots$. Tengamos en cuenta que también

$$l_1(t_{2k+1}) = \max(l_1(t_{2k}) + \bar{\lambda}_1 \delta_{2k}, 0) \text{ para } k = 0, 1, 2, \dots, \text{ ademas } l_i(t) \geq 0, \forall t. \quad (3.6)$$

Se pueden escribir ecuaciones similares para $l_2(t_k)$, $l_3(t_k)$ y $l_4(t_k)$. Si se definen

$$x_k = \begin{bmatrix} l_1(t_k) \\ l_2(t_k) \\ l_3(t_k) \\ l_4(t_k) \end{bmatrix}, b_1 = \begin{bmatrix} \bar{\lambda}_1 \\ \bar{\lambda}_2 - \bar{\mu}_2 \\ \bar{\lambda}_3 \\ \bar{\lambda}_4 - \bar{\mu}_4 \end{bmatrix}, b_2 = \begin{bmatrix} \bar{\lambda}_1 - \bar{\mu}_1 \\ \bar{\lambda}_2 \\ \bar{\lambda}_3 - \bar{\mu}_3 \\ \bar{\lambda}_4 \end{bmatrix}, b_3 = \begin{bmatrix} 0 \\ (\bar{\mu}_2 - \bar{\kappa}_2)\delta_{amb} \\ 0 \\ (\bar{\mu}_4 - \bar{\kappa}_4)\delta_{amb} \end{bmatrix},$$

$$b_4 = \begin{bmatrix} (\bar{\mu}_1 - \bar{\kappa}_1)\delta_{amb} \\ 0 \\ (\bar{\mu}_3 - \bar{\kappa}_3)\delta_{amb} \\ 0 \end{bmatrix}, b_5 = \begin{bmatrix} 0 \\ \max((\bar{\lambda}_2 - \bar{\kappa}_2)\delta_{amb}, 0) \\ 0 \\ \max((\bar{\lambda}_4 - \bar{\kappa}_4)\delta_{amb}, 0) \end{bmatrix}, b_6 = \begin{bmatrix} \max((\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{amb}, 0) \\ 0 \\ \max((\bar{\lambda}_3 - \bar{\kappa}_3)\delta_{amb}, 0) \\ 0 \end{bmatrix},$$

Entonces se tiene:

$$x_{2k+1} = \max(x_{2k} + b_1\delta_{2k} + b_3, b_5) \quad (3.7)$$

$$x_{2k+2} = \max(x_{2k+1} + b_2\delta_{2k+1} + b_4, b_6) \quad (3.8)$$

para $k = 0, 1, 2, \dots$

Definición 3.1.1. Se dice que las sucesiones $\{x_k\}_{k=0,1,2,\dots,N}$ y $\{\delta_k\}_{k=0,1,2,\dots,N-1}$ son compatibles (para δ_{amb} , $\bar{\lambda}'_i$ s, $\bar{\mu}'_i$ s y $\bar{\kappa}'_i$ s dados) si satisfacen las ecuaciones de recurrencia (3.7) y (3.8) para todo k .

Nota 3.1.1. El modelo que se obtuvo es diferente de los modelos usados por la mayoría de los investigadores, debido a que consideramos que las longitudes de los ciclos verde-ámbar-rojo pueden variar de ciclo a ciclo, es decir se ha obtenido un modelo acíclico. Además se han considerado intersecciones no saturadas, es decir, aceptamos que las longitudes de cola puedan ser iguales a 0 durante la fase verde.

3.1.3. Control óptimo

3.1.3.1. Exposición del problema

Desde ahora se supone que conocemos las tasas de llegada y de salida. Para un entero N dado, y un tiempo inicial t_0 , se quiere calcular una sucesión óptima t_1, t_2, \dots, t_N de tiempos de cambio (switching) que minimice un criterio dado J . Definimos:

$G_i(N) = G_i \cap \{0, 1, 2, \dots, N-1\}$. Las posibles funciones objetivo son:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{i=1}^4 w_i \frac{\int_{t_0}^{t_N} l_i(t) dt}{t_N - t_0} \quad (3.9)$$

- **Longitud (ponderada) media de la cola en el carril con mayores colas**

$$J_2 = \max_i \left(w_i \frac{\int_{t_0}^{t_N} l_i(t) dt}{t_N - t_0} \right) \quad (3.10)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,t} (w_i l_i(t)) \quad (3.11)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{i=1}^4 w_i \frac{\int_{t_0}^{t_N} l_i(t) dt}{\lambda_i (t_N - t_0)} \quad (3.12)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_i \left(w_i \frac{\int_{t_0}^{t_N} l_i(t) dt}{\lambda_i (t_N - t_0)} \right) \quad (3.13)$$

donde $w_i > 0$ para todo i . Los factores peso w_i se pueden usar para dar una importancia más alta o peso a algunos carriles, por ejemplo escogemos $w_1 = w_3 = 2$ y $w_2 = w_4 = 1$ para dar una importancia 100 más alta a los coches que circulan por la calle formada por los carriles L_1 y L_3 .

La razón para introducir el factor $t_N - t_0$ en J_1 , J_2 , J_4 y J_5 es que en nuestra aproximación el horizonte temporal no es fijo. El usar criterios basados en valores medios temporales tiene la ventaja de que se mantienen valores finitos para las funciones objetivo, incluso si N o t_N tienden a ∞ (a condición de que las longitudes de las colas sigan siendo finitas). Nótese que J_1 y J_4 son de hecho equivalentes, en el sentido de que para cualquier vector peso w para J_1 , existe un vector peso \tilde{w} para J_4 tal que J_1 y J_4 son iguales. Esto también se verifica para J_2 y J_5 .

Se pueden imponer algunas condiciones extras, tales como las duraciones mínimas y máximas para los tiempos verde y rojo (si el tiempo rojo es demasiado largo, los conductores tienden a creer que los semáforos están estropeados) o longitudes máximas

para las colas (esto podría corresponder a una cota superior del espacio disponible debido a la distancia con la incorporación precedente o por las características del cruce). Esto nos lleva al problema **P** siguiente:

$$\text{Minimizar } J \quad (3.14)$$

sujeto a:

$$\delta_{min,verde,1} \leq \delta_{2k+1} - \delta_{amb} \leq \delta_{max,verde,1} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.15)$$

$$\delta_{min,verde,2} \leq \delta_{2k} - \delta_{amb} \leq \delta_{max,verde,2} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.16)$$

$$x_k \leq x_{max} \quad \text{para } k = 1, 2, \dots, N \quad (3.17)$$

$$x_{2k+1} = \max(x_{2k} + b_1\delta_{2k} + b_3, b_5) \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.18)$$

$$x_{2k+2} = \max(x_{2k+1} + b_2\delta_{2k+1} + b_4, b_6) \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.19)$$

donde $\delta_{min,verde,i}$ (respectivamente $\delta_{max,verde,i}$) es el mínimo (respectivamente el máximo) tiempo verde en el carril L_i , y $(x_{max})_i$ es la máxima longitud de cola en el carril L_i . Nótese que también podríamos considerar que $\delta_{min,verde,i}$, $\delta_{max,verde,i}$ y x_{max} dependen de k . En todo lo que sigue suponemos que $x_0 \leq x_{max}$.

A continuación se discuten algunos métodos para resolver el problema (3.14)-(3.19).

Primeramente -siguiendo el trabajo presentado por Schutter-Moor en [21]- se interpretan las restricciones (3.15)-(3.19) como un problema de complementariedad lineal extendido (ELCP).

Para construir el ELCP, primero se considera (3.18) para un índice arbitrario k . Esta ecuación se puede reescribir como sigue:

$$x_{2k+1} \geq x_{2k} + b_1\delta_{2k} + b_3 \quad (3.20)$$

$$x_{2k+1} \geq b_5 \quad (3.21)$$

$$(x_{2k+1})_i = (x_{2k} + b_1\delta_{2k} + b_3)_i \quad \text{o} \quad (x_{2k+1})_i = (b_5)_i \quad \text{para } i = 1, 2, 3, 4. \quad (3.22)$$

o equivalentemente:

$$x_{2k+1} - x_{2k} - b_1\delta_{2k} - b_3 \geq 0$$

$$x_{2k+1} - b_5 \geq 0$$

$$(x_{2k+1} - x_{2k} - b_1\delta_{2k} - b_3)_i (x_{2k+1} - b_5)_i = 0 \quad \text{para } i = 1, 2, 3, 4.$$

Como una suma de números no negativos es igual a 0 si y sólo si todos los números son iguales a 0, este sistema de ecuaciones es equivalente a:

$$x_{2k+1} - x_{2k} - b_1\delta_{2k} - b_3 \geq 0 \quad (3.23)$$

$$x_{2k+1} - b_5 \geq 0 \quad (3.24)$$

$$\sum_{i=1}^4 (x_{2k+1} - x_{2k} - b_1\delta_{2k} - b_3)_i (x_{2k+1} - b_5)_i = 0 \quad (3.25)$$

Se puede repetir este razonamiento para (3.19) y para cada k. Así si se define:

$$x^* = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \quad y \quad \delta^* = \begin{bmatrix} \delta_0 \\ \delta_1 \\ \vdots \\ \delta_{N-1} \end{bmatrix}$$

es fácil comprobar que finalmente se tiene un problema **P** de la forma

$$\text{Minimizar } J \quad (3.26)$$

sujeto a:

$$Ax^* + B\delta^* + c \geq 0 \quad (3.27)$$

$$x^* + d \geq 0 \quad (3.28)$$

$$Ex^* + F\delta^* + g \geq 0 \quad (3.29)$$

$$(Ax^* + B\delta^* + c)^T (x^* + d) = 0 \quad (3.30)$$

con matrices A, B, E, F y vectores c,d,g definidos de la siguiente forma:

La matriz A es una matriz cuadrada de orden 4N ya que tiene N filas (=columnas) de bloques y cada bloque tiene tantas filas (=columnas) como número de semáforos (en el ejemplo, cuatro). Es triangular inferior con bloques identidad (I_4) en la diagonal principal y con bloques -identidad ($-I_4$) a la izquierda de la diagonal principal, el resto son bloques cero (0_4). Por tanto sus menores principales dominantes son positivos, luego A es una P-matriz.

$$A = \left(\begin{array}{c|c|c|c|c|c} I_4 & 0_4 & 0_4 & \dots & 0_4 & 0_4 \\ \hline -I_4 & I_4 & 0_4 & \dots & 0_4 & 0_4 \\ \hline 0_4 & -I_4 & I_4 & \dots & 0_4 & 0_4 \\ \hline \dots & \dots & \dots & \dots & \dots & \dots \\ \hline 0_4 & 0_4 & 0_4 & \dots & I_4 & 0_4 \\ \hline 0_4 & 0_4 & 0_4 & \dots & -I_4 & I_4 \end{array} \right)$$

La matriz B es de orden $4N \times N$ ya que tiene N filas (=columnas) de bloques y cada bloque es un bloque columna con tantas filas como numero de semáforos. En la diagonal se alternan los bloques $(-b_1)$, $(-b_2)$ y el resto son bloques cero ($0_{4 \times 1}$), el bloque del extremo inferior derecho será $(-b_1)$ si N es impar y $(-b_2)$ si N es par.

$$B = \left(\begin{array}{c|c|c|c|c} -b_1 & 0_{4 \times 1} & 0_{4 \times 1} & \dots & 0_{4 \times 1} \\ \hline 0_{4 \times 1} & -b_2 & 0_{4 \times 1} & \dots & 0_{4 \times 1} \\ \hline 0_{4 \times 1} & 0_{4 \times 1} & -b_1 & \dots & 0_{4 \times 1} \\ \hline \dots & \dots & \dots & \dots & \dots \\ \hline 0_{4 \times 1} & 0_{4 \times 1} & 0_{4 \times 1} & \dots & -b_i \end{array} \right) i = \begin{cases} 1 & \text{si } N \text{ es impar} \\ 2 & \text{si } N \text{ es par} \end{cases}$$

La matriz E es una matriz cuadrada de orden $4N$, la matriz F es de orden $4N \times N$, corresponden a las restricciones de acotación de los delta y de las longitudes de cola.

Los vectores c,d,g son vectores columna de orden $4N \times 1$ formados por N bloques columna y cada bloque tiene tantas filas como número de semáforos. En el vector c los bloques se van alternado $(-b_3)$, $(-b_4)$ y el último será $(-b_3)$ si N es impar y $(-b_4)$ si N es par. En el vector d los bloques se van alternando $(-b_5)$, $(-b_6)$ y el último será $(-b_5)$ si N es impar y $(-b_6)$ si N es par. En el vector g se van alternando $(b_3 - b_5 + b_6)$ y $(-b_4 + b_5 - b_6)$ y el último será $(b_3 - b_5 + b_6)$ si N es impar y $(-b_4 + b_5 - b_6)$ si N es par.

$$c = \left(\begin{array}{c} -b_3 \\ -b_4 \\ -b_3 \\ \vdots \\ -b_i \end{array} \right) i = \begin{cases} 3 & \text{si } N \text{ es impar} \\ 4 & \text{si } N \text{ es par} \end{cases} \quad d = \left(\begin{array}{c} -b_5 \\ -b_6 \\ -b_5 \\ \vdots \\ -b_j \end{array} \right) j = \begin{cases} 5 & \text{si } N \text{ es impar} \\ 6 & \text{si } N \text{ es par} \end{cases}$$

$$g = \left(\begin{array}{c} b_3 - b_5 + b_6 \\ -b_4 + b_5 - b_6 \\ b_3 - b_5 + b_6 \\ \vdots \\ (-1)^{i-1}b_i + (-1)^i b_5 + (-1)^{i-1}b_6 \end{array} \right) i = \begin{cases} 3 & \text{si } N \text{ es impar} \\ 4 & \text{si } N \text{ es par} \end{cases}$$

Las ecuaciones (3.27), (3.28) y (3.30) corresponden a (3.3), (3.24) y (3.25) respectivamente, y el sistema de desigualdades lineales (3.29) contiene a las condiciones (3.15)-(3.17).

Definición 3.1.2. Se dice que los vectores x^* y δ^* son compatibles para un x_0 dado si las sucesiones correspondientes $\{x_k\}_{k=0,1,2,\dots,N}$ y $\{\delta_k\}_{k=0,1,2,\dots,N-1}$ son compatibles.

En lo que sigue un vector $x^* \in (R^+)^{4N}$ se corresponderá con una sucesión $\{x_k\}_{k=1,\dots,N}$ y viceversa, con $(x^*)_{4(k-1)+j} = (x_k)_j$ para $k = 1, 2, \dots, N$ y $j=1,2,3,4$. Un vector $\delta^* \in (R_0^+)^N$ se corresponderá con una sucesión $\{\delta_k\}_{k=0,1,2,\dots,N-1}$, y viceversa, con $(\delta^*)_k = \delta_{k-1}$ para $k = 1, 2, \dots, N$

Como se ha indicado antes, el sistema (3.27)-(3.30) es un caso especial de un problema de complementariedad lineal extendido (ELCP). Para determinar la sucesión temporal switching óptima, se debe optimizar la función objetivo J en la solución del ELCP (3.27)-(3.30). En [20] se desarrolla un algoritmo para encontrar todas las soluciones de un ELCP. Si se supone que x^* y δ^* están acotados superiormente (es decir $\delta_{max,verde,i}$, $\delta_{max,rojo,i}$ y las componentes de x_{max} están definidas y son distintas de ∞), entonces el conjunto solución del sistema (3.27)-(3.30) está caracterizado por un conjunto de vectores $V = \left\{ \begin{bmatrix} x_i^* \\ \delta_i^* \end{bmatrix} / i = 1, 2, \dots, r \right\}$ y un conjunto de conjuntos de índices $\Lambda = \{\phi_j / j = 1, 2, \dots, p\}$ con $\phi_j \subseteq \{1, 2, \dots, r\}$ para todo j , tal que para cualquier j , cualquier combinación convexa de la forma

$$\sum_{i \in \phi_j} \nu_i \begin{bmatrix} x_i^* \\ \delta_i^* \end{bmatrix} \quad \text{con } \nu_i \geq 0 \text{ y } \sum_{i \in \phi_j} \nu_i = 1,$$

es la solución de (3.27)-(3.30). Los vectores de V corresponden a los vértices del poliedro P definido por el sistema de ecuaciones e inecuaciones lineales (3.27)-(3.29), y cada conjunto de índices ϕ_j corresponde a una cara de este poliedro y por tanto los puntos de la cara pueden ser expresados como combinaciones convexas de los vértices. Para cada conjunto de índices ϕ_j podríamos determinar para que combinación de los ν_i 's la función objetivo J alcanza un mínimo global sobre la cara y después seleccionar el mínimo total.

Para calcular la solución de un ELCP, el algoritmo en [20] requiere tiempos de ejecución exponenciales. Esto implica que el ELCP no es factible si el número de ciclos switching N es grande. En ese caso se considera un número más pequeño N_s de ciclos switching, se calcula la estrategia switching óptima usando el método ELCP, implementando el primer paso (o pasos) de esta estrategia, después se calcula la estrategia switching óptima para los ciclos switching siguientes, y así sucesivamente. Nótese que aunque el último acercamiento es factible, solamente nos dará una solución subóptima.

Esta aproximación se llama aproximación subóptimal multi-ELCP. La solución subóptima puede usarse como punto inicial de una rutina de minimización local aplicada al problema original total.

Se podría también aplicar una estrategia de horizonte móvil si la tasa media de llegada $\bar{\lambda}_i$ y las tasas medias de salida $\bar{\mu}_i$ y $\bar{\kappa}_i$ varían despacio en el tiempo; se estiman las tasas medias de llegada y de salida sobre un cierto período (usando medidas obtenidas de cámaras o sensores), se calcula la estrategia de control óptimo y se aplica para un cierto número de ciclos switching. Mientras tanto se actualizan las estimaciones de los $\bar{\lambda}'_i$, $\bar{\mu}'_i$ y $\bar{\kappa}'_i$ y luego se calcula una nueva estrategia de control óptimo para los valores actualizados de las tasas medias de llegada y salida, y así sucesivamente. Esta aproximación se llama modelo basado en el control predictivo o control predictivo generalizado.

Como se había mencionado anteriormente, ya se introduce una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo en las tasas de llegada y de salida, lo cual hace que, en general, el cálculo de la sucesión temporal switching óptima para semáforos sea una utopía. Además en la práctica, interesa más, obtener rápidamente una buena aproximación de la sucesión temporal switching óptima, que gastar una gran cantidad de tiempo para obtener la exacta. Por tanto en [21] se hacen algunas aproximaciones extra, lo que lleva a algoritmos muy eficientes para calcular sucesiones temporales switching subóptimas, que aproximan el óptimo exacto bastante bien, así se propone la resolución de un problema que podríamos llamar aproximado relajado **PAR**.

3.1.3.2. El problema aproximado relajado (PAR) y las soluciones subóptimas

En [21] primero se introducen unas funciones objetivo aproximadas que dependen explícitamente de x^* y δ^* , de la siguiente forma: dados x_0 y t_0 , definen la función $\tilde{l}_i(\cdot, x^*, \delta^*)$, -o para abreviar $\tilde{l}_i(\cdot)$ - como la función lineal a trozos que interpola en los puntos $(t_k, l_i(t_k))$ para $k = 0, 1, \dots, N$. Las funciones objetivo aproximadas \tilde{J}_l para $l=1,2,3,4,5$ están definidas como en (3.9)-(3.13) pero reemplazando l_i por \tilde{l}_i .

Se probará que los valores de J_3 y \tilde{J}_3 coinciden. Sea $l \in \{1, 2, 4, 5\}$, el valor de las funciones objetivo J_l y \tilde{J}_l está determinada por la superficie bajo las funciones l_i y

\tilde{l}_i , respectivamente. Si la duración de la luz ámbar es cero y si las longitudes de cola nunca son cero durante el tiempo verde, entonces las funciones l_i y \tilde{l}_i y los valores de J_l y \tilde{J}_l coinciden. En la práctica, la duración de la luz ámbar será pequeña comparada con la de la verde o de la roja. Además, una sucesión temporal switching óptima para semáforos implica la ausencia de largos períodos en los cuales no espera ningún coche en un carril, mientras que en otros carriles las longitudes de cola aumentan. Así si tenemos una sucesión temporal switching óptima para semáforos, entonces los períodos durante los cuales la longitud de cola en algún carril es igual a cero son en general cortos. Como consecuencia, para esquemas switching para semáforos en un entorno, el esquema óptimo \tilde{J}_l será una buena aproximación de J_l .

Se define el problema aproximado (**PA**) de la misma forma que se definió el problema **P** pero usando como funciones objetivo las funciones \tilde{J}_l en lugar de las J_l para $l=1,2,3,4,5$.

$$\text{Minimizar } \tilde{J} \quad (3.31)$$

sujeto a:

$$\delta_{min,verde,1} \leq \delta_{2k+1} - \delta_{amb} \leq \delta_{max,verde,1} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.32)$$

$$\delta_{min,verde,2} \leq \delta_{2k} - \delta_{amb} \leq \delta_{max,verde,2} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.33)$$

$$x_k \leq x_{max} \quad \text{para } k = 1, 2, \dots, N \quad (3.34)$$

$$x_{2k+1} = \max(x_{2k} + b_1\delta_{2k} + b_3, b_5) \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.35)$$

$$x_{2k+2} = \max(x_{2k+1} + b_2\delta_{2k+1} + b_4, b_6) \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.36)$$

Se define el problema relajado (**PR**) correspondiente al problema original **P** como sigue:

$$\text{Minimizar } J \quad (3.37)$$

sujeto a:

$$\delta_{min,verde,1} \leq \delta_{2k+1} - \delta_{amb} \leq \delta_{max,verde,1} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.38)$$

$$\delta_{min,verde,2} \leq \delta_{2k} - \delta_{amb} \leq \delta_{max,verde,2} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.39)$$

$$x_k \leq x_{max} \quad \text{para } k = 1, 2, \dots, N \quad (3.40)$$

$$x_{2k+1} \geq x_{2k} + b_1 \delta_{2k} + b_3 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.41)$$

$$x_{2k+1} \geq b_5 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.42)$$

$$x_{2k+2} \geq x_{2k+1} + b_2 \delta_{2k+1} + b_4 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.43)$$

$$x_{2k+2} \geq b_6 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.44)$$

Si se compara este problema con el original, se han reemplazado (3.18) y (3.19) por ecuaciones relajadas de la forma (3.20) y (3.21) sin tener en cuenta (3.2).

En general es más fácil resolver el problema relajado (**PR**) ya que el conjunto de soluciones factibles de este problema es un conjunto convexo, mientras que el conjunto de soluciones factibles de **P** es en general no convexo, ya que consiste en la unión de caras del poliedro definido por el sistema de desigualdades (4.1)-(3.44).

Se define el problema aproximado relajado (**PAR**) así:

$$\text{Minimizar } \tilde{J} \quad (3.45)$$

sujeto a:

$$\delta_{min,verde,1} \leq \delta_{2k+1} - \delta_{amb} \leq \delta_{max,verde,1} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.46)$$

$$\delta_{min,verde,2} \leq \delta_{2k} - \delta_{amb} \leq \delta_{max,verde,2} \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.47)$$

$$x_k \leq x_{max} \quad \text{para } k = 1, 2, \dots, N \quad (3.48)$$

$$x_{2k+1} \geq x_{2k} + b_1 \delta_{2k} + b_3 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.49)$$

$$x_{2k+1} \geq b_5 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \quad (3.50)$$

$$x_{2k+2} \geq x_{2k+1} + b_2 \delta_{2k+1} + b_4 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.51)$$

$$x_{2k+2} \geq b_6 \quad \text{para } k = 0, 1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \quad (3.52)$$

Se verifican las siguientes propiedades:

Propiedad 3.1.1. Sean $x_0 \in (\mathbb{R}^+)^4$, $x^* \in (\mathbb{R}^+)^{4N}$ y $\delta^* \in (\mathbb{R}_0^+)^N$. Si x^* y δ^* son compatibles para x_0 entonces se tiene que: $J_3(\delta^*) = \tilde{J}_3(x^*, \delta^*)$ y $J_l(\delta^*) \leq \tilde{J}_l(x^*, \delta^*)$ para $l=1,2,4,5$.

DEMOSTRACIÓN. Sean x^* y δ^* compatibles para x_0 . Se supone que $\bar{\kappa}_i \leq \bar{\mu}_i$, por tanto $\bar{\lambda}_i - \bar{\kappa}_i \geq \bar{\lambda}_i - \bar{\mu}_i, \forall i$. Esto implica que no es posible una situación en la que $\bar{\lambda}_i - \bar{\mu}_i > 0$ y $\bar{\lambda}_i - \bar{\kappa}_i < 0$. Como consecuencia, el máximo de las funciones lineales a trozos $l_i(\cdot, \delta^*)$ y $\tilde{l}_i(\cdot, x^*, \delta^*)$ coincide. Esto implica que $J_3(\delta^*) = \tilde{J}_3(x^*, \delta^*)$.

Sea $k \in G_i(N)$, por tanto T_i está verde en el intervalo $(t_k, t_{k+1} - \delta_{amb})$ y ámbar en $(t_{k+1} - \delta_{amb}, t_{k+1})$. Es fácil ver que $l_i(t, \delta^*) \leq \tilde{l}_i(t, x^*, \delta^*), \forall t \in (t_k, t_{k+1})$ ya que $\bar{\kappa}_i \leq \bar{\mu}_i$. Si $k \leq N - 2$ entonces T_i está rojo en el intervalo (t_{k+1}, t_{k+2}) y en este intervalo l_i y \tilde{l}_i coinciden. Esto implica que $l_i(t, \delta^*) \leq \tilde{l}_i(t, x^*, \delta^*), \forall t \in [t_0, t_N]$. Por tanto

$$\int_{t_0}^{t_N} l_i(t, \delta^*) dt \leq \int_{t_0}^{t_N} \tilde{l}_i(t, x^*, \delta^*) dt$$

Como consecuencia se tiene que $J_l(\delta^*) \leq \tilde{J}_l(x^*, \delta^*)$ para $l = 1,2,4,5$. \square

Propiedad 3.1.2. Si J es una función estrictamente monótona de x^* (es decir si para cada δ^* con componentes positivas y para todos \tilde{x}^*, \hat{x}^* , con $0 \leq \tilde{x}^* \leq \hat{x}^*$, y con $\tilde{x}_j^* < \hat{x}_j^*$ para al menos un índice j , tenemos que $J(\tilde{x}^*, \delta^*) < J(\hat{x}^*, \delta^*)$) entonces cada solución del problema relajado **PR** también es solución del problema original **P**.

DEMOSTRACIÓN. Sean $\tilde{x}^*, \tilde{\delta}^*$ una solución óptima de **PR**. Ahora se demuestra por contradicción que si J es estrictamente monótona de x^* , entonces \tilde{x}^* también satisface (3.18)-(3.19). Supongamos que \tilde{x}^* no satisface (3.18)-(3.19).

- Primero supongamos que para algún índice k el vector \tilde{x}_{2k+1} no satisface (3.18). Sea l el índice más pequeño tal que

$$\tilde{x}_{2l+1} \geq \max(\tilde{x}_{2l} + b_1 \tilde{\delta}_{2l} + b_3, b_5) \text{ y } (\tilde{x}_{2l+1})_i \neq \max((\tilde{x}_{2l} + b_1 \tilde{\delta}_{2l} + b_3)_i, (b_5)_i)$$

para algún índice i .

Ahora se define $\hat{\delta}^* = \tilde{\delta}^*$ y

$$\hat{x}^* = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \vdots \\ \hat{x}_N \end{bmatrix}$$

con:

$$\begin{aligned}\widehat{x}_k &= \widetilde{x}_k && \text{para } k = 0, 1, \dots, 2l, \\ \widehat{x}_{2k+1} &= \max(\widehat{x}_{2k} + b_1 \widehat{\delta}_{2k} + b_3, b_5) && \text{para } k = l, l+1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor, \\ \widehat{x}_{2k+2} &= \max(\widehat{x}_{2k+1} + b_2 \widehat{\delta}_{2k+1} + b_4, b_6) && \text{para } k = l, l+1, \dots, \left\lfloor \frac{N}{2} \right\rfloor - 1,\end{aligned}$$

Trivialmente $\widehat{x}_{2l+1} \leq \widetilde{x}_{2l+1}$. Demostramos ahora que $\widehat{x}_{2l+2} \leq \widetilde{x}_{2l+2}$. Se tiene que:

$$\begin{aligned}\widehat{x}_{2l+2} &= \max(\widehat{x}_{2l+1} + b_2 \widehat{\delta}_{2l+1} + b_4, b_6) \\ &= \max(\widehat{x}_{2l+1} + b_2 \widetilde{\delta}_{2l+1} + b_4, b_6) \quad (\text{ya que } \widehat{\delta}^* = \widetilde{\delta}^*) \\ &\leq \max(\widetilde{x}_{2l+1} + b_2 \widetilde{\delta}_{2l+1} + b_4, b_6) \quad (\text{ya que } \widehat{x}_{2l+1} \leq \widetilde{x}_{2l+1}) \\ &\leq \widetilde{x}_{2l+2} \quad (\text{por (3.43)-(3.44)})\end{aligned}$$

De manera similar se puede probar que $\widehat{x}_k \leq \widetilde{x}_k$ para $k = 2l+3, 2l+4, \dots, N$. Por tanto, $\widehat{x}^* \leq \widetilde{x}^*$. Como $\widetilde{x}_k \leq x_{max} \forall k$, esto implica que \widehat{x}_k también satisface (4.3) $\forall k$. Además como $\widehat{\delta}^* = \widetilde{\delta}^*$, esto implica que \widehat{x}_k y $\widehat{\delta}_k$ satisfacen las desigualdades del sistema (4.1)-(3.44) $\forall k$.

- Si \widetilde{x}_{2k+1} satisface (3.18) $\forall k$, entonces tiene que existir un índice l tal que (3.19) no se cumple para \widetilde{x}_{2l+2} ya que se supone que \widetilde{x}^* no satisface (3.18)-(3.19). Ahora se puede usar un razonamiento similar al utilizado anteriormente para construir \widehat{x}^* y $\widehat{\delta}^*$ tal que $\widehat{x}^* \leq \widetilde{x}^*$, $\widehat{x}_j^* < \widetilde{x}_j^*$ para algún j y tal que \widehat{x}_k y $\widehat{\delta}_k$ satisfagan las desigualdades del sistema (4.1)-(3.44), $\forall k$.

Como $\widehat{x}^* \leq \widetilde{x}^*$ y $\widehat{x}_j^* < \widetilde{x}_j^*$ para al menos un índice j , se tiene que $J(\widehat{x}^*, \widehat{\delta}^*) < J(\widetilde{x}^*, \widetilde{\delta}^*)$ lo cual implica que $\widetilde{x}^*, \widetilde{\delta}^*$ no es una solución óptima del problema **PR**, lo cual es una contradicción, por tanto nuestra hipótesis inicial de que \widetilde{x}^* no satisface (3.18)-(3.19) es falsa. El conjunto solución de (3.15)-(3.19) es un subconjunto del conjunto solución de (4.1)-(3.44), esto implica que $\widetilde{x}^*, \widetilde{\delta}^*$ también es una solución óptima del problema original **P**. En resumen, cada solución óptima del problema relajado **PR** es también una solución óptima del problema original **P**. \square

Propiedad 3.1.3. Para $x_0, \delta_{amb}, \overline{\lambda}'_i s, \overline{\mu}'_i s, \overline{\kappa}'_i s$ y δ^* dados, las funciones \widetilde{J}_1 y \widetilde{J}_4 son funciones estrictamente monótonas de x^* .

DEMOSTRACIÓN. Sea $\delta^* \in (\mathbb{R}_0^+)^N$ y sean $\widetilde{x}^*, \widehat{x}^* \in (\mathbb{R}^+)^{4N}$. Si $\widetilde{x}^* \leq \widehat{x}^*$ y $\widetilde{x}_j^* < \widehat{x}_j^*$ para al menos un índice $j=4(k-1)+i$ entonces se tiene que $\widetilde{l}_i(t, \widetilde{x}^*, \delta^*) \leq \widehat{l}_i(t, \widehat{x}^*, \delta^*) \forall t \in [t_0, t_N]$,

y $\tilde{l}_i(t, \tilde{x}^*, \delta^*) < \tilde{l}_i(t, \hat{x}^*, \delta^*)$ en algún intervalo no vacío $(t_k - \eta, t_k + \eta) \cap [t_0, t_N]$ con $\eta > 0$. Entonces,

$$\int_{t_0}^{t_N} \tilde{l}_i(t, \tilde{x}^*, \delta^*) dt < \int_{t_0}^{t_N} \tilde{l}_i(t, \hat{x}^*, \delta^*) dt$$

lo cual implica que $\tilde{J}_1(\tilde{x}^*, \delta^*) < \tilde{J}_1(\hat{x}^*, \delta^*)$ y $\tilde{J}_4(\tilde{x}^*, \delta^*) < \tilde{J}_4(\hat{x}^*, \delta^*)$. \square

Como consecuencia al usar como función objetivo \tilde{J}_1 o \tilde{J}_4 (que son estrictamente monótonas de x^*) se ha probado que toda solución del problema aproximado relajado es una solución del problema aproximado y esta es una solución suficientemente buena para el problema **P**, según se justificó anteriormente. Sin embargo las funciones \tilde{J}_2 , \tilde{J}_3 y \tilde{J}_5 no son funciones estrictamente monótonas de x^* para un δ^* dado, por lo que no podemos obtener para ellas resultados análogos a los encontrados para \tilde{J}_1 o \tilde{J}_4 .

3.1.4. Modelo con restricciones de complementariedad. Propuesta de un método híbrido de solución

Como hemos visto en [21] se proponen algoritmos de solución de un problema aproximado basados en la relajación de éste. Como los autores señalan, estos algoritmos conducen, en general, a soluciones subóptimas. El problema ELCP es un problema NP-duro. Todo esto justifica la aplicación de un algoritmo heurístico de solución.

El método de solución que proponemos se basa en la propiedad que a continuación demostramos y en la descomposición del vector solución del problema en la forma $x^*(\delta^*)$.

Propiedad 3.1.4. *Para cada $\delta^* \in (\mathbb{R}_0^+)^N$, las restricciones de ELCP describen un problema de complementariedad lineal con solución única.*

DEMOSTRACIÓN. Al considerar el cambio de variable $x^* + d = y$ se obtiene el problema LCP con matriz A y término independiente $B\delta^* + c - Ad$. La matriz A es triangular inferior con elementos positivos (1) en la diagonal principal, por tanto P-matriz y como consecuencia [16], el problema $LCP(\delta^*)$ tiene solución única, esto quiere decir que dado δ^* la longitud de las colas está unívocamente determinada. \square

El método de solución consistirá en una heurística para fijar los valores de δ^* mas un algoritmo eficiente para resolver el problema $LCP(\delta^*)$:

1. $\hat{J} = M$ (valor suficientemente grande), $k=1$. Repetir q veces.
2. Mediante una heurística se explora el dominio acotado del vector δ^* . Construir un vector δ^* .
3. Fijado δ^* , se determina la solución factible única del problema de complementariedad lineal $LCP(\delta^*)$ correspondiente. Sea $x^*(\delta^*)$ la solución determinada.
4. Se evalúan las funciones J_l , según los distintos criterios.
5. Para cada criterio s : Si $J_s(x^*(\delta^*)) < \hat{J}_s$ entonces $\hat{J}_s := J_s(x^*(\delta^*))$ y $\hat{\delta}(s) = \delta^*$. $q=q+1$.

Este enfoque tiene las siguientes ventajas:

- Es sencillo, adaptable a variaciones en los datos y rápido en su ejecución.
- Fácilmente adaptable a cambios en el modelo como pueden ser: tiempos de fase proporcionales a un tiempo fijo por determinar, con constantes de proporcionalidad prefijadas o cotas pre-establecidas a la duración de las fases según el ciclo.
- Permite introducir un diálogo con el usuario a partir de los resultados y, según los criterios evaluados, ofrecer a éste la posibilidad del análisis de diversas alternativas de operación.
- Posibilidad de instrumentar distintos algoritmos heurísticos en el primer paso.
- El algoritmo propuesto por [21] trabaja con un problema relajado y caracterizando puntos estacionarios. Las heurísticas ofrecen la opción de explorar la región de soluciones “escapando” de extremos locales.
- Podemos considerar la minimización de la congestión medida por cualquiera de los criterios aún cuando la función no sea diferenciable.

3.2. Modelos estático y dinámico

3.2.1. Introducción

En esta sección se proponen y describen dos modelos para optimizar el flujo de tráfico en un cruce simple regulado por semáforos: un modelo estático y otro dinámico.

El modelo estático, aunque está basado en el de Schutter-Moor expuesto en la sección 3.1, es una simplificación del mismo, lo que facilita su descripción, comprensión, programación y resolución a través de un método híbrido que combina una metaheurística tipo recocido simulado con un algoritmo quasi-Newton para problemas de optimización no suaves.

Con el modelo dinámico lo que se intenta es eliminar situaciones indeseadas, como por ejemplo, aquella en la que no hay autos en el carril con luz verde, mientras en otros carriles hay vehículos esperando, para ello se considera que hay sensores que permiten conocer la cantidad de autos en cada vía. Usando el método de resolución empleado para el caso estático, los resultados computacionales reflejan una mejoría en el flujo de tráfico cuando se usa el modelo dinámico respecto al estático.

3.2.2. Modelo estático

En esta sección presentamos un modelo matemático para el problema de control de tráfico en un cruce simple. Se tiene una intersección a la que confluyen dos calles, ambas doble vía y en las cuales no es posible doblar hacia la izquierda. En cada esquina del cruce hay un semáforo. Se quiere hallar los períodos en que deben permanecer en verde los semáforos T_1 y T_3 , (rojo T_2 y T_4) y en cuales T_1 y T_3 muestran la luz roja (verde T_2 y T_4), para evitar la congestión de las vías, descrita por el número de vehículos en espera (longitud de la cola) en cada vía. Para evaluar la congestión de tráfico en las vías se consideran distintos criterios: la suma de las longitudes medias de las colas de cada vía, el tiempo medio de espera, largo de la cola más larga o una combinación de ellas. De modo que, el objetivo es calcular una sucesión temporal switching óptima para semáforos que minimice alguno de los criterios anteriores. Con este planteamiento, hemos de modelar y resolver un problema que en cada ciclo tiene dos fases: ver figuras 3.2 y 3.3

Consideramos dado el número de veces que los semáforos tienen la luz verde y lo

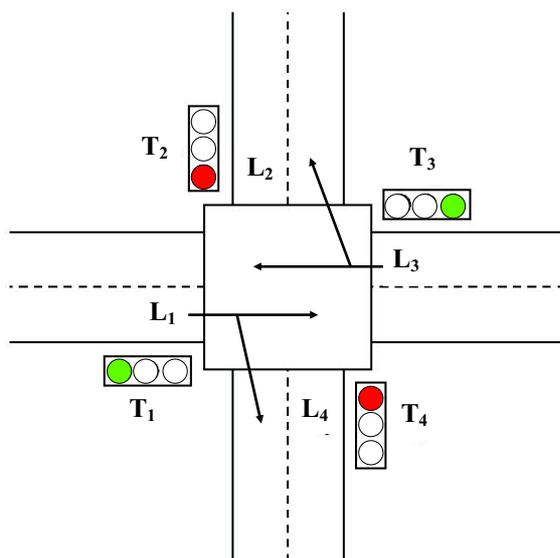


Figura 3.2: 1ª fase

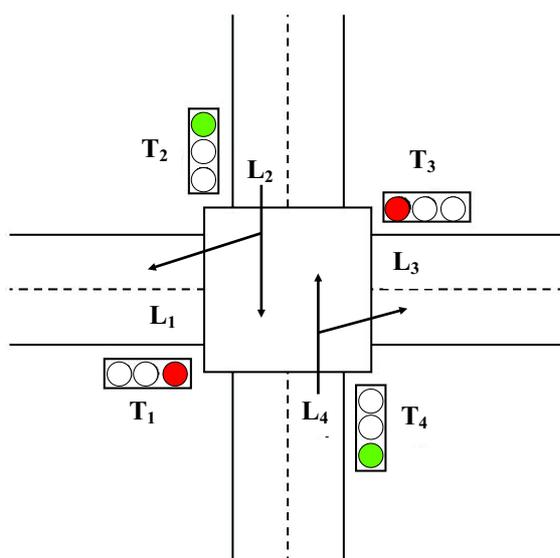


Figura 3.3: 2ª fase

denotamos por N . También se considera pre-establecido δ_{amb} que representa el tiempo de duración de la luz ámbar (en la práctica se considera una constante que fluctúa entre los 3 y los 4 segundos).

Como el tiempo de luz verde y el tiempo de luz ámbar de los carriles pares (resp. impares) determinan el tiempo de luz roja de los carriles impares (resp. pares), el problema consiste en determinar los períodos que deben permanecer en verde las luces de los semáforos T_1, T_2, T_3 y T_4 .

Describamos el problema *con las variables de control* δ_k , tiempo de duración de la luz verde en el k -ésimo cambio de luz (incluyendo el ámbar) y las *variables de estado* $x_k = (x_{k1}, x_{k2}, x_{k3}, x_{k4})^t$, cantidad promedio de vehículos en los carriles L_1, L_2, L_3, L_4 en el momento del k -ésimo cambio de luz.

Como simplificaciones del modelo trabajaremos con el comportamiento promedio por lo que asumimos que la cantidad de autos es continua. Para cada carril i se definen las tasas medias de llegada y salida de los vehículos bajo las luces verde y ámbar y se denotan:

λ_i : tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo).

μ_i : tasa media de salida en el carril L_i cuando el semáforo está en verde.

κ_i : tasa media de salida en el carril L_i cuando el semáforo está en ámbar.

Los vectores $b_i, i=1, \dots, 6$ representan el número de vehículos en cada carril teniendo en cuenta los movimientos asociados a cada fase:

$$b_1 = \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 \end{bmatrix}, b_2 = \begin{bmatrix} \lambda_1 \\ \lambda_2 - \mu_2 \\ \lambda_3 \\ \lambda_4 - \mu_4 \end{bmatrix}, b_3 = \begin{bmatrix} (\mu_1 - \kappa_1)\delta_{amb} \\ 0 \\ (\mu_3 - \kappa_3)\delta_{amb} \\ 0 \end{bmatrix},$$

$$b_4 = \begin{bmatrix} 0 \\ (\mu_2 - \kappa_2)\delta_{amb} \\ 0 \\ (\mu_4 - \kappa_4)\delta_{amb} \end{bmatrix}, b_5 = \begin{bmatrix} \max((\lambda_1 - \kappa_1)\delta_{amb}, 0) \\ 0 \\ \max((\lambda_3 - \kappa_3)\delta_{amb}, 0) \\ 0 \end{bmatrix}, b_6 = \begin{bmatrix} 0 \\ \max((\lambda_2 - \kappa_2)\delta_{amb}, 0) \\ 0 \\ \max((\lambda_4 - \kappa_4)\delta_{amb}, 0) \end{bmatrix},$$

Entonces:

$$x_{2k+1} = \max(x_{2k} + b_1\delta_{2k+1} + b_3, b_5) \quad \text{para } k = 0, 1, \dots, N-1,$$

$$x_{2k+2} = \max(x_{2k+1} + b_2\delta_{2k+2} + b_4, b_6) \quad \text{para } k = 0, 1, \dots, N-1,$$

Las características de la zona de tráfico puede imponer otras regulaciones, tales como establecer cotas a las duraciones mínimas y máximas para los tiempos verde y rojo, así:

$\delta_{min.verde,i}$: cota mínima de duración de la luz en semáforo $i=1,2$.

$\delta_{max.verde,i}$: cota máxima de duración de la luz en semáforo $i=1,2$.

El problema de regulación de los semáforos puede representarse a través del siguiente modelo:

$$\text{Minimizar } J \quad (3.53)$$

sujeto a:

$$\delta_{min.verde,1} \leq \delta_{2k+1} - \delta_{amb} \leq \delta_{max.verde,1} \quad \text{para } k = 0, 1, \dots, N - 1, \quad (3.54)$$

$$\delta_{min.verde,2} \leq \delta_{2k+2} - \delta_{amb} \leq \delta_{max.verde,2} \quad \text{para } k = 0, 1, \dots, N - 1, \quad (3.55)$$

$$x_{2k+1} = \max(x_{2k} + b_1\delta_{2k+1} + b_3, b_5) \quad \text{para } k = 0, 1, \dots, N - 1, \quad (3.56)$$

$$x_{2k+2} = \max(x_{2k+1} + b_2\delta_{2k+2} + b_4, b_6) \quad \text{para } k = 0, 1, \dots, N - 1, \quad (3.57)$$

Si los carriles difieren en importancia al evaluar el nivel de congestión de los mismos, se representa el nivel de importancia del carril, fijando diferentes pesos o ponderaciones $w_i, i = 1, \dots, 4$. La función J a minimizar puede ser:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{j=1}^4 w_j \frac{\sum_{i=1}^{2N} x_{ij} \delta_i}{\sum_{i=1}^{2N} \delta_i} \quad (3.58)$$

- **Longitud (ponderada) media de las colas en el carril con mayores colas**

$$J_2 = \max_j w_j \frac{\sum_{i=1}^{2N} x_{ij} \delta_i}{\sum_{i=1}^{2N} \delta_i} \quad (3.59)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,j} w_j x_{ij} \quad (3.60)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{j=1}^4 \frac{w_j \sum_{i=1}^{2N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{2N} \delta_i} \quad (3.61)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_j \frac{w_j \sum_{i=1}^{2N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{2N} \delta_i} \quad (3.62)$$

Se puede analizar una sexta opción que consiste en considerar una combinación de los criterios anteriores. Esto nos lleva a un modelo multi-objetivo que podemos resolver tomando una combinación positiva de las distintas funciones objetivo.

$$J_6 = \sum_{i=1}^5 \alpha_i J_i \quad (3.63)$$

Como consecuencia de lo anteriormente expuesto, conocido δ (vector de los tiempos de cada luz), la cantidad de autos en cada carril quede determinado mediante las ecuaciones (3.56)-(3.57). Denotamos por $x(\delta)$ la matriz de $2N$ filas y 4 columnas donde $x_{ij}(\delta)$ indica la cantidad de autos en el carril j en el momento del cambio de luz i .

Definimos:

$$\Delta_{ij} = \begin{cases} 1 & \text{si } i \equiv j(2) \\ 0 & \text{si } \text{no} \end{cases}$$

donde $i \equiv j(2)$ significa que i es congruente con j módulo 2, o sea, existe un entero n tal que $i=j+2n$.

Una forma más compacta de escribir el problema, sin tener que diferenciar entre índices pares (correspondientes al cambio de luz roja-verde de los semáforos 1 y 3) y los impares (donde cambia la luz de los semáforos 2 y 4) es la siguiente:

$$\text{Minimizar } J \quad (3.64)$$

sujeto a:

$$\Delta_{k1} \delta_{\min.verde,1} + \Delta_{k2} \delta_{\min.verde,2} \leq \delta_k - \delta_{amb} \leq \Delta_{k1} \delta_{\max.verde,1} + \Delta_{k2} \delta_{\max.verde,2} \quad (3.65)$$

para $k = 1, \dots, 2N$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj}\mu_j)\delta_k + \Delta_{kj}(\mu_j - \kappa_j)\delta_{amb}, \max(\Delta_{kj}(\lambda_j - \kappa_j)\delta_{amb}, 0)] \quad (3.66)$$

para $k = 1, \dots, 2N$; $j = 1, 2, 3, 4$

Como habíamos mencionado anteriormente, ya introducimos una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo de las tasas de llegada y de salida, lo que hace recomendable obtener una buena aproximación de la solución óptima que sea calculable en un tiempo corto y fácilmente ajustable a los cambios en los valores de los parámetros. De hecho, utilizando un razonamiento análogo al empleado en la sección 3.1.3.1, llegamos a que estamos ante un problema de optimización con restricciones de complementariedad lineal (problema NP-duro), que involucra a $10N$ variables, $4N$ restricciones suaves y $8N$ no suaves, lo que hace que sea muy costoso aplicar un algoritmo iterativo [19]. Es por eso que para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves ([39]).

3.2.3. Modelo dinámico

En este modelo consideramos que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz. Esta decisión permite menores tiempos de espera, pues elimina situaciones indeseadas como sería que no hubiera autos en el carril con la luz verde mientras otros carriles esperan.

Para la estrategia de cambio consideramos que si hay al menos r_2 vehículos esperando en carriles con luz roja, entonces es recomendable cambiar, así como si hay a lo más r_1 autos en los carriles a los que corresponde la luz verde. Dado que puede ser muy restrictivo asumir que los cambios de luces ocurren en estas situaciones, suavizamos el modelo planteando que se minimiza el error cuadrático de aproximar la cantidad de autos en los momentos de cambio en los carriles en espera por r_2 y en los carriles de servicio por r_1 .

O sea si la cantidad de autos en el carril al que corresponde la luz verde en el momento i es $\sum_j \Delta_{ij}x_{ij}$, debemos agregar a la función objetivo el cuadrado de la diferencia entre este valor y r_1 , pero solo si $\sum_j \Delta_{ij}x_{ij} > r_1$. Agregando la variable auxiliar ρ_i^{max} y

la restricción $\sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max}$ entonces al minimizar $(\rho_i^{max} - r_1^2)$ se cumple lo deseado. Realizando un razonamiento análogo para los autos que esperan por el cambio de luz para pasar y con la ayuda de la variable ρ_i^{min} , la penalidad quedaría

$$\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2$$

donde

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \quad \text{para } i = 1, \dots, 2N \quad \sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \quad \text{para } i = 1, \dots, 2N$$

Usando un parámetro c para controlar el peso que se le da a esta nueva función en el objeto a minimizar, el modelo final es:

$$\text{Minimizar } F = J + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right] \quad (3.67)$$

sujeto a:

$$\Delta_{k1} \delta_{min.verde,1} + \Delta_{k2} \delta_{min.verde,2} \leq \delta_k - \delta_{amb} \leq \Delta_{k1} \delta_{max.verde,1} + \Delta_{k2} \delta_{max.verde,2} \quad (3.68)$$

para $k = 1, \dots, 2N$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj} \mu_j) \delta_k + \Delta_{kj} (\mu_j - \kappa_j) \delta_{amb}, \max(\Delta_{kj} (\lambda_j - \kappa_j) \delta_{amb}, 0)] \quad (3.69)$$

para $k = 1, \dots, 2N$; $j = 1, 2, 3, 4$

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \quad \text{para } i = 1, \dots, 2N \quad (3.70)$$

$$\sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \quad \text{para } i = 1, \dots, 2N \quad (3.71)$$

Para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves.

3.2.4. Algoritmos solución

Según se indicó en la sección 2.6.1, con la metaheurística basada en recocido simulado, se trata de resolver un problema de optimización $\min F(x)$, $x \in M$, simulando el

proceso de enfriamiento en el que a medida que baja la temperatura el movimiento de los átomos es menos probable ([58]). Dado un candidato a solución Ω el algoritmo busca en una vecindad del mismo, un punto con mejor evaluación de la función objetivo. De no existir, se toma un punto con peor evaluación como nueva solución con una cierta probabilidad. Esta probabilidad que se calcula a partir de la ley de Boltzman, es proporcional a la temperatura del sistema e inversamente proporcional a la variación de los valores de la función objetivo al pasar al nuevo punto. Cada cierto número de iteraciones se disminuye la temperatura del sistema. Los distintos criterios de parada son: encontrar un punto con un valor adecuado de la función objetivo y alcanzar una temperatura suficientemente baja en el sistema.

En base a lo expuesto en la sección 2.6.1, se formuló el algoritmo, cuyo pseudo-código se presenta a continuación.

1. Fijar $t_0 > 0$ (temperatura inicial), $\alpha \in (0, 1)$ (parámetro del esquema de enfriamiento), Q (cantidad de pasos con igual temperatura), T (menor temperatura a alcanzar por el sistema). Escoger J_i , $i = 1, \dots, 6$, y construir F , función objetivo del problema.
2. Construir un vector δ_0 y la solución $x(\delta_0)$ asociada. $F_{mejor} = F(x(\delta_0), \delta_0)$, $\delta_{mejor} = \delta_0$ temperatura $t = t_0$, $i=0$, $F_{anterior} = +\infty$.
3. Mientras $t > T$,
 1. $q = 1$
 2. Mientras $q < Q$
 1. Tomar δ un punto vecino de δ_i y hallar $x(\delta)$. Si $F(x(\delta), \delta) < F_{mejor}$, $\delta_{mejor} = \delta$, $F_{mejor} = F(x(\delta), \delta)$. Si $F(x(\delta), \delta) < F_{anterior}$ $\delta_{i+1} = \delta$, $F_{anterior} = F(x(\delta), \delta)$ e ir a 3.2.4.
 2. Generar r , número aleatorio de acuerdo a la ley uniforme en $(0,1)$.
 3. Si $r < \exp\left(\frac{F_{anterior} - F(x(\delta), \delta)}{t}\right)$, $\delta_{i+1} = \delta$, si no, $\delta_{i+1} = \delta_i$, $\delta_{anterior} = \delta_{i+1}$ y $F_{anterior} = F(x(\delta_{i+1}), \delta_{i+1})$ y construir la solución $x(\delta_{i+1})$ asociada.
 4. $q=q+1$ e ir a 3.2.
 3. $t = t \cdot \alpha$ e ir a 3.
4. Mejor solución δ_{mejor} con valor de la función objetivo $F(x(\delta_{mejor}), \delta_{mejor})$. Fin.

Pueden considerarse diversos criterios para definir el sistema de vecindades para ejecutar el algoritmo formulado. Consideramos el sistema de vecindades definido por el criterio: δ es vecino de δ^* si difieren en solo una componente y $\|\delta - \delta^*\|=1$, es decir δ es vecino de δ^* si existe i tal que $\delta_i = \delta_i^* \pm 1$ y $\delta_j = \delta_j^*$ para todo $j \neq i$. En aras de mantener factibilidad se tiene en cuenta que el nuevo punto cumpla las restricciones en (3.68). Claramente las vecindades así definidas, son un subconjunto discreto del conjunto de soluciones factibles, pero posibilita la rápida exploración de soluciones factibles.

Teniendo en cuenta las restricciones (3.69) podemos reducir el modelo implícitamente y calcular los valores de las funciones involucradas y sus derivadas de forma iterativa. Así resolvemos el problema con un algoritmo determinístico tipo quasi-Newton para funciones no suaves ([10]). La estrategia es:

1. Escoger un vector inicial δ_0 y la solución $x(\delta_0)$ asociada y $k=0$. Fijar sucesiones $\alpha_k, \beta_k, \gamma_k, \zeta_k \rightarrow 0^+, k \rightarrow \infty$ y números $c_1 > c_2 > 0$.
2. Fijar $s=0, \delta_{k,s} = \delta_k$.
3. Buscar una dirección de descenso d , estimando ∇F por el método de gradiente discreto ([10]), con $(\alpha, \beta, \gamma, \zeta, c) = (\alpha_k, \beta_k, \gamma_k, \zeta_k, c_1)$.
4. Si $|d| \leq \alpha_k, \delta_{k+1} = \delta_{k,s}, k=k+1$, ir a 2.
5. Sea $\sigma_s = \operatorname{argmax}\{\sigma \geq 0, F(\delta_{k,s} + \sigma \frac{d}{|d|}) - F(\delta_{k,s}) \leq -c_2 \frac{\sigma}{|d|}\}$.
6. $\delta_{k,s+1} = \delta_{k,s} + \sigma_s \frac{d}{|d|}$
7. $s=s+1$. Ir a 3.

La dirección de descenso se calcula de la siguiente manera:

1. Fijar $g, |g_1| = 1$ y fijar e , vértice del cubo unitario en \mathbb{R}^n .
2. Calcular el gradiente discreto $v_1 = \Gamma^i(\delta, g_1, e, \beta, \gamma, \zeta)$. $D = \{v_1\}, |g_i| > \sqrt{\frac{1}{2n}} k=1$.
3. Hallar $w_k = \operatorname{argmin}\{|w|/w \in D\}$.
4. Si $|w_k| < \alpha$, fin.
5. Hallar $g_{k+1} = \frac{w_k}{|w_k|}$. Si $F(\delta + \gamma g_{k+1}) - F(\delta) \leq c\gamma|w_k|$, fin.

6. Calcular el gradiente discreto $v_{k+1} = \Gamma^i(\delta, g_{k+1}, e, \beta, \gamma, \zeta)$, $|g_i| > \sqrt{\frac{1}{2n}}$ $D=D \cup \{v_k\}$ e ir a 3.

El gradiente discreto $\Gamma^i(\delta, g_1, e, \beta, \gamma, \zeta)$ se calcula componente a componente mediante un esquema de diferencias divididas entre puntos consecutivos que difieren en $\zeta^j \beta$ unidades en la componente j , a lo que se agrega una combinación de las componentes $j \neq i$ para la i -ésima componente.

Este método arrojó en la mayoría de los casos mejores resultados que la heurística, pero a un coste computacional superior. Usando las ventajas de ambos algoritmos, proponemos buscar una buena aproximación a la solución mediante el método de recocido simulado. Luego se mejora el comportamiento en una vecindad del punto mediante el algoritmo quasi-Newton para funciones no suaves ya expuesto.

Presentamos ahora un ejemplo para ilustrar el comportamiento del algoritmo, así como los resultados obtenidos mediante la programación en MATLAB de los modelos y métodos de resolución propuestos y su implementación en un ordenador con procesador Intel Core i7, 950 que trabaja a 307 GHz.

3.2.5. Resultados computacionales

Datos: 5 ciclos, tiempo de ámbar 3 sg. Se usaron los parámetros que se indican en la tabla 3.2

	Carril 1	Carril 2	Carril 3	Carril 4
Tasa-llegada	0.3	0.3	0.2	0.25
Tasa-sal.ver.	0.7	0.7	1	0.5
Tasa-sal.ámb.	0.25	0.25	0.5	0.5

Tabla 3.2: Parámetros

En cuanto a las cotas del tiempo de las luces, se tomaron: $\delta_{min.verde,1} = \delta_{min.verde,2} = 5$ y $\delta_{max.verde,1} = \delta_{max.verde,2} = 30$.

Construcción de las funciones objetivo: Pesos $w = [1 \ 1 \ 1 \ 1]$, $\alpha_i = 1$, $\forall i = 1, \dots, 5$ (construcción de la función objetivo J_6), $r_1 = 2$, $r_2 = 8$ (construcción de las funciones objetivo del modelo dinámico)

Se obtuvo aleatoriamente un δ_0 inicial (para cada una de las 6 opciones) y se utilizó

para evaluar los dos modelos.

Temperatura inicial $t_0 = 100000000$

Función de reducción de temperatura $T_M = 0.5 T_{M-1}$

Número de iteraciones con igual temperatura $Q = 100$

Criterio de parada $T < 0.000001$

Veremos que en todos los casos, los resultados obtenidos con el modelo dinámico son mejores que los del modelo estático, aunque el tiempo computacional sea sensiblemente mayor.

Para $J = J_1$ obtuvimos los resultados indicados en las tablas 3.3 y 3.4

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	5	7.37	10.34	19.64	6.80
Semáforos 2-4	6.66	8.90	13.50	5.65	7.82

Tabla 3.3: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	1.5	0	1.25
Ciclo 1º-2ª fase	2.15	0.18	1.33	0
Ciclo 2º-1ª fase	0.55	2.39	0	1.84
Ciclo 2º-2ª fase	3.22	0.18	1.78	0
Ciclo 3º-1ª fase	0.43	3.29	0	2.58
Ciclo 3º-2ª fase	4.48	0.15	2.7	0
Ciclo 4º-1ª fase	0.15	6.04	0	4.91
Ciclo 4º-2ª fase	1.84	5.13	1.13	3.5
Ciclo 5º-1ª fase	0.47	7.17	0	5.2
Ciclo 5º-2ª fase	2.82	5.40	1.56	3.24

Tabla 3.4: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 8,265 autos. Peor cola: 7 autos. Tiempo computacional 0,7 sg.

$$\text{Para } F_1 = J_1 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	5	7.44	9.42	7	6.08
Semáforos 2-4	5.44	8.17	7.42	10.68	6.18

Tabla 3.5: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1°-1ª fase	0.15	1.5	0	1.25
Ciclo 1°-2ª fase	1.78	0.67	1.09	0
Ciclo 2°-1ª fase	0.15	2.91	0	1.86
Ciclo 2°-2ª fase	2.61	0.99	1.63	0
Ciclo 3°-1ª fase	0.19	3.81	0	2.36
Ciclo 3°-2ª fase	2.41	2.2	1.48	0.50
Ciclo 4°-1ª fase	0.96	4.30	0	2.25
Ciclo 4°-2ª fase	4.17	1.37	2.14	0
Ciclo 5°-1ª fase	3.09	3.20	0	1.52
Ciclo 5°-2ª fase	4.94	2.07	1.24	0

Tabla 3.6: Promedio de vehículos en el momento del cambio de luz

En este ejemplo hemos obtenido buenos resultados ya que las colas son cortas (peor cola aproximadamente 5 autos). Además observamos que se evitó la congestión con períodos largos sin cambio de luz cuando la cantidad de tráfico es grande y la situación no deseada de que haya largas colas de vehículos esperando, mientras que en la vía con luz verde sean muy pocos los que están pasando. El valor de la función objetivo es 7,8874 autos. Tiempo computacional 0,9 sg.

Para $J = J_2$ obtuvimos los resultados indicados en las tablas 3.7 y 3.8

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	5	6.70	6.10	7.52	23.69
Semáforos 2-4	5.69	6.11	11.21	8.09	11.82

Tabla 3.7: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	1.5	0	1.25
Ciclo 1º-2ª fase	1.85	0.57	1.13	0
Ciclo 2º-1ª fase	0.53	2.58	0	1.67
Ciclo 2º-2ª fase	2.36	1.49	1.22	0.15
Ciclo 3º-1ª fase	1.27	3.32	0	1.67
Ciclo 3º-2ª fase	4.63	0.18	2.24	0
Ciclo 4º-1ª fase	2.97	2.44	0	1.88
Ciclo 4º-2ª fase	5.38	0.58	1.60	0
Ciclo 5º-1ª fase	0.15	7.69	0	5.92
Ciclo 5º-2ª fase	3.70	4.31	2.36	2.97

Tabla 3.8: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 3,43 autos. Peor cola 7,69 autos. Tiempo computacional 0,8 sg.

$$Para F_2 = J_2 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	5	7.52	10.02	12.86	6.14
Semáforos 2-4	5.91	8.49	12.21	10.42	5.71

Tabla 3.9: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1 ^o -1 ^a fase	0.15	1.5	0	1.25
Ciclo 1 ^o -2 ^a fase	1.92	0.48	1.18	0
Ciclo 2 ^o -1 ^a fase	0.26	2.74	0	1.88
Ciclo 2 ^o -2 ^a fase	2.81	0.70	1.70	0
Ciclo 3 ^o -1 ^a fase	0.15	3.70	0	2.50
Ciclo 3 ^o -2 ^a fase	3.82	0.17	2.44	0
Ciclo 4 ^o -1 ^a fase	0.15	4.02	0	3.21
Ciclo 4 ^o -2 ^a fase	3.27	1.21	2.08	0.61
Ciclo 5 ^o -1 ^a fase	2.17	3.04	0	2.14
Ciclo 5 ^o -2 ^a fase	3.88	2.11	1.14	0.72

Tabla 3.10: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 3,29 autos. Peor cola 4 autos. Tiempo computacional 1,2 sg.

Para $J = J_3$ obtuvimos los resultados indicados en las tablas 3.11 y 3.12

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	21.95	10.98	17.97	16	17
Semáforos 2-4	16.86	12	18.97	12	12

Tabla 3.11: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	6.58	0	5.48
Ciclo 1º-2ª fase	5.21	1.19	3.37	1.27
Ciclo 2º-1ª fase	2.16	4.48	0	4.01
Ciclo 2º-2ª fase	5.77	1.02	2.4	1
Ciclo 3º-1ª fase	0.15	6.41	0	5.5
Ciclo 3º-2ª fase	5.84	0.17	3.79	0.76
Ciclo 4º-1ª fase	0.8	5	0	4.76
Ciclo 4º-2ª fase	4.39	1.52	2.4	1.76
Ciclo 5º-1ª fase	0.15	6.62	0	6
Ciclo 5º-2ª fase	3.74	3.17	2.4	3

Tabla 3.12: Promedio de vehículos en el momento del cambio de luz

La peor cola tuvo 6,62 autos. Tiempo computacional 0,84 sg.

$$Para F_3 = J_3 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	9.38	6.31	7.54	9.36	7.75
Semáforos 2-4	6.71	8.91	7.46	5	10.99

Tabla 3.13: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1 ^o -1 ^a fase	0.15	2.82	0	2.35
Ciclo 1 ^o -2 ^a fase	2.16	1.48	1.34	0.67
Ciclo 2 ^o -1 ^a fase	0.99	3.37	0	2.24
Ciclo 2 ^o -2 ^a fase	3.66	1.16	1.78	0.02
Ciclo 3 ^o -1 ^a fase	1.99	3.42	0	1.90
Ciclo 3 ^o -2 ^a fase	4.23	1.79	1.49	0.04
Ciclo 4 ^o -1 ^a fase	1.84	4.60	0	2.38
Ciclo 4 ^o -2 ^a fase	3.34	3.95	1	1.13
Ciclo 5 ^o -1 ^a fase	1.59	6.27	0	3.07
Ciclo 5 ^o -2 ^a fase	4.88	3.23	2.20	0.32

Tabla 3.14: Promedio de vehículos en el momento del cambio de luz

La peor cola tuvo 6,27 autos. Tiempo computacional 1,3 sg.

Para $J = J_4$ obtuvimos los resultados indicados en las tablas 3.15 y 3.16

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	13.98	13.11	17	16.22	15.14
Semáforos 2-4	13.87	12.3	17.7	14.9	19.9

Tabla 3.15: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	4.19	0	3.49
Ciclo 1º-2ª fase	4.31	0.15	2.77	0.03
Ciclo 2º-1ª fase	0.41	4.08	0	3.3
Ciclo 2º-2ª fase	4.1	0.51	2.46	0.23
Ciclo 3º-1ª fase	0.15	5.61	0	4.48
Ciclo 3º-2ª fase	5.46	0.15	3.54	0.05
Ciclo 4º-1ª fase	0.32	5.01	0	4.1
Ciclo 4º-2ª fase	4.78	0.41	2.97	0.39
Ciclo 5º-1ª fase	0.15	4.96	0	4.17
Ciclo 5º-2ª fase	6.12	0.15	3.98	0

Tabla 3.16: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 33,61 sg. Peor cola 6,1 autos. Tiempo computacional 0,7 sg.

$$Para F_4 = J_4 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	5	8.57	7.53	10.03	10.80
Semáforos 2-4	5.52	8.43	5.27	7.82	19.33

Tabla 3.17: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1 ^o -1 ^a fase	0.15	1.5	0	1.25
Ciclo 1 ^o -2 ^a fase	1.81	0.64	1.10	0
Ciclo 2 ^o -1 ^a fase	0.15	3.21	0	2.14
Ciclo 2 ^o -2 ^a fase	2.68	1.19	1.69	0
Ciclo 3 ^o -1 ^a fase	1.01	3.44	0	1.92
Ciclo 3 ^o -2 ^a fase	2.60	2.69	1.1	0.60
Ciclo 4 ^o -1 ^a fase	0.15	5.70	0	3.10
Ciclo 4 ^o -2 ^a fase	2.50	3.91	1.57	1.15
Ciclo 5 ^o -1 ^a fase	0.15	7.15	0	3.85
Ciclo 5 ^o -2 ^a fase	5.95	0.77	3.87	0

Tabla 3.18: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 30,88 sg. Peor cola 7 autos. Tiempo computacional 1,1 sg.

Para $J = J_5$ obtuvimos los resultados indicados en las tablas 3.19 y 3.20

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	6.99	6.92	7.69	16.21	7.14
Semáforos 2-4	5.03	7.25	5.35	5	11.22

Tabla 3.19: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	2.09	0	1.75
Ciclo 1º-2ª fase	1.66	1.43	1	0.49
Ciclo 2º-1ª fase	0.24	3.51	0	2.22
Ciclo 2º-2ª fase	2.41	1.96	1.45	0.41
Ciclo 3º-1ª fase	0.69	4.27	0	2.33
Ciclo 3º-2ª fase	2.30	3.48	1.1	0.99
Ciclo 4º-1ª fase	0.15	8.34	0	5.05
Ciclo 4º-2ª fase	1.65	7.69	1	3.79
Ciclo 5º-1ª fase	0.15	9.83	0	5.58
Ciclo 5º-2ª fase	3.51	6.70	2.24	2.78

Tabla 3.20: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 18,22 sg. Peor cola 9,83 autos. Tiempo computacional 0,8 sg.

$$Para F_5 = J_5 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	14.68	16.76	28.3	22.69	29.43
Semáforos 2-4	15.46	16.24	27.8	21.35	28.19

Tabla 3.21: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1 ^o -1 ^a fase	0.15	4.4	0	3.67
Ciclo 1 ^o -2 ^a fase	4.78	0.15	3.01	0
Ciclo 2 ^o -1 ^a fase	0.15	5.18	0	4.19
Ciclo 2 ^o -2 ^a fase	5.02	0.15	3.24	0.12
Ciclo 3 ^o -1 ^a fase	0.15	8.64	0	7.2
Ciclo 3 ^o -2 ^a fase	8.49	0.15	5.56	0.25
Ciclo 4 ^o -1 ^a fase	0.76	6.95	0	5.92
Ciclo 4 ^o -2 ^a fase	7.17	0.15	4.27	0.58
Ciclo 5 ^o -1 ^a fase	0.15	8.98	0	7.94
Ciclo 5 ^o -2 ^a fase	8.6	0.15	5.63	0.89

Tabla 3.22: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 13,47 sg. Peor cola 9 autos. Tiempo computacional 1,4 sg.

Culminamos con el problema multi-objetivo, modelado mediante la minimización de la función $J = J_6$, usando como vector de pesos (1 1 1 1 1) para no hacer diferencias entre los distintos criterios, los resultados están indicados en las tablas 3.23 y 3.24.

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	12.77	9.74	24.12	6.98	19.5
Semáforos 2-4	8.94	11.41	18.85	14.56	5.86

Tabla 3.23: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	3.83	0	3.19
Ciclo 1º-2ª fase	2.83	1.6	1.79	0.95
Ciclo 2º-1ª fase	0.28	4.52	0	3.39
Ciclo 2º-2ª fase	3.7	1.31	2.28	0.54
Ciclo 3º-1ª fase	0.15	8.55	0	6.57
Ciclo 3º-2ª fase	5.8	2.36	3.77	1.86
Ciclo 4º-1ª fase	4.36	4.45	0	3.6
Ciclo 4º-2ª fase	8.73	0.15	2.91	0
Ciclo 5º-1ª fase	2.28	6	0	4.87
Ciclo 5º-2ª fase	4	5	1.17	3.41

Tabla 3.24: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 81,5. Peor cola 8,73 autos. Tiempo computacional 1,7 sg.

$$Para F_6 = J_6 + c \left[\sum_{i=1}^{2N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{2N} (\rho_i^{min} - r_2)^2 \right]$$

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-3	11.07	12.08	10.13	13.50	11.33
Semáforos 2-4	11.97	15.53	7.92	5.31	5

Tabla 3.25: Duración de la luz verde en cada ciclo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.15	3.32	0	2.77
Ciclo 1º-2ª fase	3.73	0.15	2.39	0
Ciclo 2º-1ª fase	0.25	3.77	0	3.02
Ciclo 2º-2ª fase	4.91	0.15	3.11	0
Ciclo 3º-1ª fase	2.21	3.19	0	2.53
Ciclo 3º-2ª fase	4.58	1.37	1.58	0.55
Ciclo 4º-1ª fase	0.54	5.42	0	3.93
Ciclo 4º-2ª fase	2.13	4.65	1.06	2.60
Ciclo 5º-1ª fase	0.15	8	0	5.43
Ciclo 5º-2ª fase	1.65	7.40	1	4.18

Tabla 3.26: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo 58,12. Peor cola 8 autos. Tiempo computacional 1,9 sg.

Como se observa en la tabla 3.27, el modelo dinámico funciona mucho mejor que el estático ya que, en todos los casos (con las diferentes funciones objetivo) el valor de la función objetivo y la longitud de la peor cola es menor, y el tiempo computacional es solamente sensiblemente superior.

	Valor función objetivo	Longitud peor cola	Tiempo computacional
J_1	8.26 autos	7 autos	0.7 sg.
F_1	7.88 autos	5 autos	0.9 sg.
J_2	3.43 autos	7.69 autos	0.8 sg.
F_2	3.29 autos	4 autos	1.2 sg.
J_3	6.62 autos	6.62 autos	0.84 sg.
F_3	6.27 autos	6.27 autos	1.3 sg.
J_4	33.61 sg.	6.1 autos	0.7 sg.
F_4	30.88 sg.	7 autos	1.1 sg.
J_5	18.22 sg.	9.83 autos	0.8 sg.
F_5	13.47 sg.	9 autos	1.4 sg.
J_6	81.5	8.73 autos	1.7 sg.
F_6	58.12	8 autos	1.9 sg.

Tabla 3.27: Tabla comparativa. Resultados modelos estático y dinámico

Capítulo 4

Modelos para cruce de tres fases

4.1. Modelo estático

En esta sección presentamos un modelo matemático para el problema de control de tráfico en una intersección a la que confluyen dos calles, ambas doble vía con las siguientes características: en la de dirección y sentidos norte-sur y sur-norte se puede circular de frente, girar a derecha e izquierda, mientras que en la de dirección y sentidos este-oeste y oeste-este sólo se puede circular de frente y girar a la derecha. En cada esquina del cruce hay un semáforo. Se quiere determinar los períodos en que deben permanecer en verde o rojo las luces de los semáforos T_1 , T_2 , T_3 y T_4 para evitar la congestión de las vías, descrita por el número de vehículos en espera (longitud de la cola) en cada vía. Para evaluar la congestión de tráfico en las vías se consideran distintos criterios: la suma de las longitudes medias de las colas de cada vía, el tiempo medio de espera, largo de la cola más larga o una combinación de ellas. De modo que, el objetivo es calcular una sucesión temporal switching óptima para semáforos que minimice alguno de los criterios anteriores. Con este planteamiento, hemos de modelar y resolver un problema que en cada ciclo tiene tres fases: ver figuras 4.1, 4.2 y 4.3

- En la **primera fase** el semáforo T_1 está verde y T_2 , T_3 y T_4 permanecen en rojo, por tanto los coches del carril L_1 rebasan el cruce, continuando de frente o girando a la derecha o a la izquierda.
- En la **segunda fase** los semáforos T_2 y T_4 que han cambiado de rojo a verde, permiten que los vehículos que circulan por los carriles L_2 y L_4 rebasen el cruce siguiendo de frente o girando a la derecha, (T_1 y T_3 están en rojo).

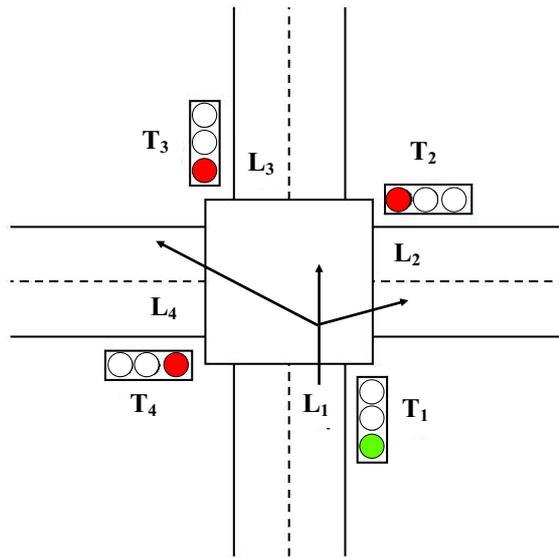


Figura 4.1: 1ª fase

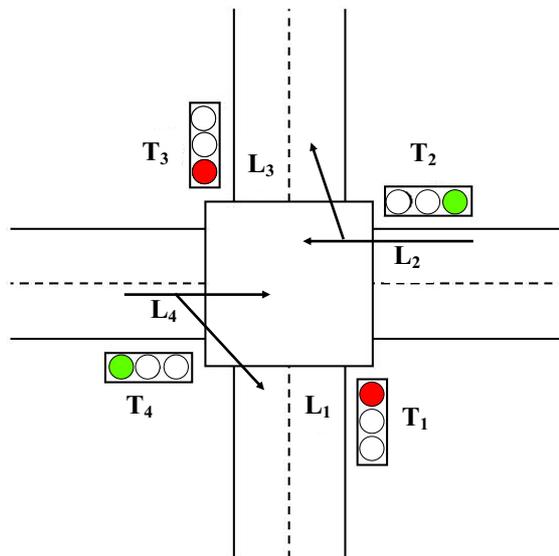


Figura 4.2: 2ª fase

- En la **tercera fase** T_3 cambia de rojo a verde, por tanto atraviesan el cruce los vehículos que circulan por el carril L_3 pudiendo continuar de frente o girar a la derecha o a la izquierda, (T_1 , T_2 y T_4 están en rojo).

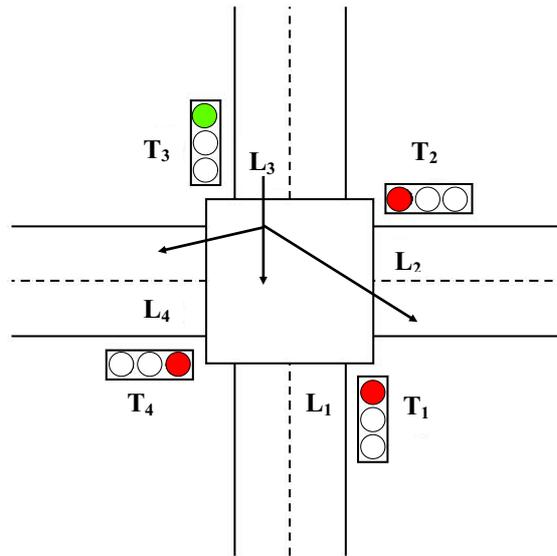


Figura 4.3: 3ª fase

Consideramos dado el número de veces que los semáforos tienen la luz verde y lo denotamos por N . También se considera pre-establecido δ_{amb} que representa el tiempo de duración de la luz ámbar (en la práctica se considera una constante que fluctúa entre los 3 y los 4 segundos).

Describimos el problema *con las variables de control* δ_k , tiempo de duración de la luz verde en el k -ésimo cambio de luz (incluyendo el ámbar) y las *variables de estado* $x_k = (x_{k1}, x_{k2}, x_{k3}, x_{k4})^t$, cantidad promedio de vehículos en los carriles L_1 , L_2 , L_3 , L_4 en el momento del k -ésimo cambio de luz.

Como simplificaciones del modelo trabajaremos con el comportamiento promedio por lo que asumimos que la cantidad de autos es continua. Para cada carril L_i se definen las tasas medias de llegada y salida de los vehículos bajo las luces verde y ámbar y se denotan:

λ_i : tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo).

μ_i : tasa media de salida en el carril L_i cuando el semáforo está en verde.

κ_i : tasa media de salida en el carril L_i cuando el semáforo está en ámbar.

Los vectores b_i , $i=1, \dots, 9$ representan el número de vehículos en cada carril teniendo en cuenta los movimientos asociados a cada fase:

$$\begin{aligned}
 b_1 &= \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix}, b_2 = \begin{bmatrix} \lambda_1 \\ \lambda_2 - \mu_2 \\ \lambda_3 \\ \lambda_4 - \mu_4 \end{bmatrix}, b_3 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 \end{bmatrix}, b_4 = \begin{bmatrix} (\mu_1 - \kappa_1)\delta_{amb} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
 b_5 &= \begin{bmatrix} 0 \\ (\mu_2 - \kappa_2)\delta_{amb} \\ 0 \\ (\mu_4 - \kappa_4)\delta_{amb} \end{bmatrix}, b_6 = \begin{bmatrix} 0 \\ 0 \\ (\mu_3 - \kappa_3)\delta_{amb} \\ 0 \end{bmatrix}, b_7 = \begin{bmatrix} \max((\lambda_1 - \kappa_1)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
 b_8 &= \begin{bmatrix} 0 \\ \max((\lambda_2 - \kappa_2)\delta_{amb}, 0) \\ 0 \\ \max((\lambda_4 - \kappa_4)\delta_{amb}, 0) \end{bmatrix}, b_9 = \begin{bmatrix} 0 \\ 0 \\ \max((\lambda_3 - \kappa_3)\delta_{amb}, 0) \\ 0 \end{bmatrix}.
 \end{aligned}$$

Entonces:

$$\begin{aligned}
 x_{3k+1} &= \max(x_{3k} + b_1\delta_{3k+1} + b_4, b_7) \quad \text{para } k = 0, 1, \dots, N-1, \\
 x_{3k+2} &= \max(x_{3k+1} + b_2\delta_{3k+2} + b_5, b_8) \quad \text{para } k = 0, 1, \dots, N-1, \\
 x_{3k+3} &= \max(x_{3k+2} + b_3\delta_{3k+3} + b_6, b_9) \quad \text{para } k = 0, 1, \dots, N-1,
 \end{aligned}$$

Las características de la zona de tráfico puede imponer otras regulaciones, tales como establecer cotas a las duraciones mínimas y máximas para los tiempos verde y rojo, así:

$\delta_{min.verde,i}$: cota mínima de duración de la luz en semáforo $i=1,2,3$.

$\delta_{max.verde,i}$: cota máxima de duración de la luz en semáforo $i=1,2,3$.

El problema de regulación de los semáforos puede representarse a través del siguiente modelo:

$$\text{Minimizar } J \quad (4.1)$$

sujeto a:

$$\delta_{min.verde,1} \leq \delta_{3k+1} - \delta_{amb} \leq \delta_{max.verde,1} \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.2)$$

$$\delta_{min.verde,2} \leq \delta_{3k+2} - \delta_{amb} \leq \delta_{max.verde,2} \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.3)$$

$$\delta_{min.verde,3} \leq \delta_{3k+3} - \delta_{amb} \leq \delta_{max.verde,3} \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.4)$$

$$x_{3k+1} = \max(x_{3k} + b_1\delta_{3k+1} + b_4, b_7) \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.5)$$

$$x_{3k+2} = \max(x_{3k+1} + b_2\delta_{3k+2} + b_5, b_8) \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.6)$$

$$x_{3k+3} = \max(x_{3k+2} + b_3\delta_{3k+3} + b_6, b_9) \quad \text{para } k = 0, 1, \dots, N-1, \quad (4.7)$$

Si los carriles difieren en importancia al evaluar el nivel de congestión de los mismos, se representa el nivel de importancia del carril, fijando diferentes pesos o ponderaciones $w_i, i = 1, \dots, 4$. La función J a minimizar puede ser:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{j=1}^4 w_j \frac{\sum_{i=1}^{3N} x_{ij} \delta_i}{\sum_{i=1}^{3N} \delta_i} \quad (4.8)$$

- **Longitud (ponderada) media de las colas en el carril con mayores colas**

$$J_2 = \max_j w_j \frac{\sum_{i=1}^{3N} x_{ij} \delta_i}{\sum_{i=1}^{3N} \delta_i} \quad (4.9)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,j} w_j x_{ij} \quad (4.10)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{j=1}^4 \frac{w_j}{\lambda_j} \frac{\sum_{i=1}^{3N} x_{ij} \delta_i}{\sum_{i=1}^{3N} \delta_i} \quad (4.11)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_j \frac{w_j \sum_{i=1}^{3N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{3N} \delta_i} \quad (4.12)$$

Se puede analizar una sexta opción que consiste en considerar una combinación de los criterios anteriores. Esto nos lleva a un modelo multi-objetivo que podemos resolver tomando una combinación positiva de las distintas funciones objetivo.

$$J_6 = \sum_{i=1}^5 \alpha_i J_i \quad (4.13)$$

donde los valores α_i corresponden a los pesos que se les dan a los distintos usuarios.

Como consecuencia de lo anteriormente expuesto, conocido δ (vector de los tiempos de cada luz), la cantidad de autos en cada carril quede determinado mediante las ecuaciones (4.5)-(4.6)-(4.7). Denotamos por $x(\delta)$ la matriz de $3N$ filas y 4 columnas donde $x_{ij}(\delta)$ indica la cantidad de autos en el carril j en el momento del cambio de luz i .

Definimos:

$$\Delta_{ij} = \begin{cases} 1 & \text{si } i \equiv j(3) \quad j = 1, 2, 3 \\ 0 & \text{si } i \equiv 2(3) \quad j = 4 \\ 0 & \text{si no} \end{cases}$$

es decir:

$\Delta_{i1} = 1$, cuando i es congruente con 1 módulo 3, en los demás casos $\Delta_{i1} = 0$.

$\Delta_{i2} = 1$, cuando i es congruente con 2 módulo 3, en los demás casos $\Delta_{i2} = 0$.

$\Delta_{i3} = 1$, cuando i es congruente con 3 módulo 3, en los demás casos $\Delta_{i3} = 0$.

$\Delta_{i4} = 1$, cuando i es congruente con 2 módulo 3, en los demás casos $\Delta_{i4} = 0$.

Una forma más compacta de escribir el problema, sin tener que diferenciar entre las tres fases es la siguiente:

$$\text{Minimizar } J \quad (4.14)$$

sujeto a:

$$\begin{aligned} \Delta_{k1} \delta_{min.verde,1} + \Delta_{k2} \delta_{min.verde,2} + \Delta_{k3} \delta_{min.verde,3} &\leq \delta_k - \delta_{amb} \leq \\ &\leq \Delta_{k1} \delta_{max.verde,1} + \Delta_{k2} \delta_{max.verde,2} + \Delta_{k3} \delta_{max.verde,3} \quad \text{para } k = 1, \dots, 3N \end{aligned} \quad (4.15)$$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj}\mu_j)\delta_k + \Delta_{kj}(\mu_j - \kappa_j)\delta_{amb}, \max(\Delta_{kj}(\lambda_j - \kappa_j)\delta_{amb}, 0)] \quad (4.16)$$

para $k = 1, \dots, 3N$; $j = 1, 2, 3, 4$

Como habíamos mencionado anteriormente, ya introducimos una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo de las tasas de llegada y de salida, lo que hace recomendable obtener una buena aproximación de la solución óptima que sea calculable en un tiempo corto y fácilmente ajustable a los cambios en los valores de los parámetros. De hecho, utilizando un razonamiento análogo al empleado en la sección 3.1.3.1, llegamos a que estamos ante un problema de optimización con restricciones de complementariedad lineal (problema NP-duro), que involucra a $15N$ variables, $6N$ restricciones suaves y $12N$ no suaves, lo que hace que sea muy costoso aplicar un algoritmo iterativo [19]. Es por eso que para su solución proponemos el algoritmo indicado en la siguiente sección.

4.1.1. Algoritmo solución

Usaremos el algoritmo heurístico inspirado en la metaheurística de recocido simulado utilizado en la sección 3.2.4, adaptado al modelo del presente capítulo, como método de resolución. Al igual que se hizo en el capítulo anterior, se podría combinar el método heurístico con el método quasi-Newton, pero en el ejemplo descrito en la siguiente sección, se comprueba que el método heurístico funciona adecuadamente en un tiempo computacional corto, permitiendo mejorar el flujo de tráfico en las horas punta en la intersección considerada, si lo comparamos con el flujo existente cuando la regulación de semáforos es a ciclo fijo, que es la que en la actualidad se usa en dicho cruce.

El algoritmo y el método de resolución propuestos han sido programados en MATLAB y se han implementado en un ordenador con procesador Intel Core i7, 950 que trabaja a 307 GHz. El programa permite variar los parámetros del algoritmo (temperatura inicial y final, número de repeticiones para una temperatura fija y estrategia de disminución de la temperatura), a fin de posibilitar el ajuste de sus valores en distintas situaciones de tráfico (por ejemplo, en cuanto a densidad de tráfico o número de ciclos).

4.1.2. Aplicación al cruce de la avenida de Finisterre con las calles Palomar y Paseo de los Puentes

La intersección semaforizada de la avenida de Finisterre con las calles Palomar y Paseo de los Puentes de la ciudad de A Coruña, (ver figura 4.4) es un ejemplo del tipo de cruce presentado y modelado en este capítulo, o sea, de cruce regulado por semáforos con las tres fases descritas en las figuras 4.1, 4.2 y 4.3 en cada ciclo.



Figura 4.4: Imagen del cruce

En la primera fase, el semáforo situado en la calle Palomar se encuentra en color verde, permitiendo que los coches que circulan por dicha calle puedan rebasar el cruce y continuar de frente, por el Paseo de los Puentes, o girar a la derecha o a la izquierda hacia la avenida de Finisterre.

En la segunda fase, los semáforos situados en la avenida de Finisterre cambian de rojo a verde, permitiendo la circulación por la mencionada avenida, o girar a la derecha hacia la calle Palomar, o girar a la derecha hacia el Paseo de los Puentes.

En la tercera fase, el semáforo situado en el Paseo de los Puentes cambia de rojo a verde, permitiendo a los vehículos rebasar el cruce, para continuar de frente hacia la calle Palomar, o girar a la derecha o a la izquierda hacia la avenida de Finisterre.

En las proximidades del cruce mencionado se encuentra un centro escolar. A las horas de entrada y salida del colegio, existe un problema importante de embotellamiento, provocado fundamentalmente por el número elevado de vehículos que acceden la zona para poder llevar o recoger a los alumnos.

Los semáforos están regulados mediante un ciclo fijo en todos los momentos del día. La duración del tiempo ámbar es de 3 sg. y la duración del verde es (incluido el ámbar) de 30 sg. para los semáforos de la calle Palomar y avenida de Finisterre y de 20 sg. para el del Paseo de los Puentes. Esta regulación, aunque se aproxima bastante a la óptima durante la mayor parte del día, resulta bastante mala en las horas punta de horario escolar, durante las cuales se pueden producir colas de hasta 20-22 coches.

A continuación, se analiza la validez del modelo propuesto para representar la situación del cruce mencionado. Se observa, a partir de datos reales de entrada y salida de vehículos en el cruce, que el número de vehículos que esperan en los semáforos según el modelo, es aproximadamente el mismo que los que en la realidad constituyen las colas en las horas punta de horario escolar. Luego, usando esos mismos datos y el método heurístico propuesto para la resolución del problema, se obtiene una regulación variable de los tiempos verde de los semáforos para llegar a optimizar las diferentes funciones objetivo, con lo cual la longitud de las colas se reduce considerablemente, lo que permite una circulación más fluida, una reducción del gasto de combustible, disminución de la contaminación ambiental y de los accidentes.

4.1.3. Resultados computacionales

Se consideran en este ejemplo diez ciclos ($N=10$). Por mediciones directas en observaciones de varios días durante el horario pico, se estimaron las tasas de llegada y salida en los carriles.

$$\lambda_1=0.16, \lambda_2=0.1, \lambda_3=0.12, \lambda_4=0.11 \text{ (vehículos por segundo)}$$

$$\mu_1=0.43, \mu_2=0.43, \mu_3=0.45, \mu_4=0.51 \text{ (vehículos por segundo)}$$

$$\kappa_1 = \kappa_2 = \kappa_3 = \kappa_4=0.1 \text{ (vehículos por segundo)}. \delta_{amb} = 3sg.$$

$$\delta_{min.verde,1} = \delta_{min.verde,2} = \delta_{min.verde,3} = 10sg.$$

$$\delta_{max.verde,1} = \delta_{max.verde,2} = 50sg., \delta_{max.verde,3} = 30sg.$$

Pesos $w=[1 \ 1 \ 1 \ 1]$. En ciclo fijo, el tiempo verde (incluido el ámbar) para los semáforos T_1, T_2 y T_4 es de 30 sg. y para el T_3 de 20 sg.

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1 ^o -1 ^a fase	0.18	3	3.6	3.3
Ciclo 1 ^o -2 ^a fase	4.98	0	7.2	0
Ciclo 1 ^o -3 ^a fase	8.18	2	1.65	2.2
Ciclo 2 ^o -1 ^a fase	1.07	5	5.25	5.5
Ciclo 2 ^o -2 ^a fase	5.87	0	8.85	0
Ciclo 2 ^o -3 ^a fase	9.07	2	3.3	2.2
Ciclo 3 ^o -1 ^a fase	1.96	5	6.9	5.5
Ciclo 3 ^o -2 ^a fase	6.76	0	10.5	0
Ciclo 3 ^o -3 ^a fase	9.96	2	4.95	2.2
Ciclo 4 ^o -1 ^a fase	2.85	5	8.55	5.5
Ciclo 4 ^o -2 ^a fase	7.65	0	12.15	0
Ciclo 4 ^o -3 ^a fase	10.85	2	6.6	2.2
Ciclo 5 ^o -1 ^a fase	3.74	5	10.2	5.5
Ciclo 5 ^o -2 ^a fase	8.54	0	13.8	0
Ciclo 5 ^o -3 ^a fase	11.74	2	8.25	2.2
Ciclo 6 ^o -1 ^a fase	4.63	5	11.85	5.5
Ciclo 6 ^o -2 ^a fase	9.43	0	15.45	0
Ciclo 6 ^o -3 ^a fase	12.63	2	9.9	2.2
Ciclo 7 ^o -1 ^a fase	5.52	5	13.5	5.5
Ciclo 7 ^o -2 ^a fase	10.32	0	17.1	0
Ciclo 7 ^o -3 ^a fase	13.52	2	11.55	2.2
Ciclo 8 ^o -1 ^a fase	6.41	5	15.15	5.5
Ciclo 8 ^o -2 ^a fase	11.21	0	18.75	0
Ciclo 8 ^o -3 ^a fase	14.41	2	13.2	2.2
Ciclo 9 ^o -1 ^a fase	7.3	5	16.8	5.5
Ciclo 9 ^o -2 ^a fase	12.1	0	20.4	0
Ciclo 9 ^o -3 ^a fase	15.3	2	14.85	2.2
Ciclo 10 ^o -1 ^a fase	8.19	5	18.45	5.5
Ciclo 10 ^o -2 ^a fase	12.99	0	22.05	0
Ciclo 10 ^o -3 ^a fase	16.19	2	16.5	2.2

Tabla 4.1: Promedio de autos en el momento del cambio de luz en ciclo fijo

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0.18	1.5	1.8	1.65
Ciclo 1º-2ª fase	2.42	0	3.48	0
Ciclo 1º-3ª fase	4.82	1.5	0	1.68
Ciclo 2º-1ª fase	0.41	3.5	2.46	3.88
Ciclo 2º-2ª fase	2.49	0.2	4.02	0
Ciclo 2º-3ª fase	5.21	1.9	0	1.9
Ciclo 3º-1ª fase	0.18	4.4	3.06	4.65
Ciclo 3º-2ª fase	2.58	0.44	4.86	0
Ciclo 3º-3ª fase	5.46	2.24	0	2.01
Ciclo 4º-1ª fase	0.18	4.74	3.06	4.76
Ciclo 4º-2ª fase	2.58	0.78	4.86	0
Ciclo 4º-3ª fase	5.46	2.58	0	2.01
Ciclo 5º-1ª fase	0.18	5.08	3.06	4.76
Ciclo 5º-2ª fase	2.74	0.79	4.98	0
Ciclo 5º-3ª fase	5.46	2.49	0.42	1.9
Ciclo 6º-1ª fase	0.51	4.69	3.06	4.32
Ciclo 6º-2ª fase	2.59	1.39	4.62	0.35
Ciclo 6º-3ª fase	5.31	3.09	0	2.22
Ciclo 7º-1ª fase	0.18	5.39	2.82	4.75
Ciclo 7º-2ª fase	3.06	0.44	4.98	0
Ciclo 7º-3ª fase	5.46	1.94	1.08	1.68
Ciclo 8º-1ª fase	0.18	4.44	4.08	4.43
Ciclo 8º-2ª fase	1.78	2.13	5.28	1.66
Ciclo 8º-3ª fase	4.02	3.53	1.71	3.2
Ciclo 9º-1ª fase	0.69	5.13	3.63	4.96
Ciclo 9º-2ª fase	2.61	2.16	5.07	1.39
Ciclo 9º-3ª fase	5.01	3.66	1.17	3.04
Ciclo 10º-1ª fase	1.41	5.36	3.21	4.91
Ciclo 10º-2ª fase	3.01	3.05	4.41	2.14
Ciclo 10º-3ª fase	5.25	4.45	0.85	3.68

Tabla 4.2: Promedio de autos en el momento del cambio de luz con heurística

Puede apreciarse que la peor cola es de 5.46 vehículos que se alcanza en la calle Palomar, en el Paseo de los Puentes es de 5.28 y en la avenida de Finisterre es aproximadamente de 5 vehículos. Por tanto los resultados son muchísimo mejores que los que se presentan en la realidad con el ciclo de semáforo de tiempo fijo. Ello es debido a que se evita que en la avenida de Finisterre existan momentos de luz verde sin que rebase el cruce ningún vehículo. Además el tiempo verde para la fase 1 está en torno a los 23-25 sg, (30 en ciclo fijo), en la fase 2 está en torno a los 15-16 sg. (30 en ciclo fijo) y en la fase 3 está en torno a 15-17 sg. (20 en ciclo fijo). Por tanto, como cabía esperar, el error mayor está en el tiempo que en ciclo fijo se le concede a la fase 2.

El tiempo computacional para la heurística es en torno a los 3,3 sg., tiempo que se puede considerar adecuado para que on-line se puedan ajustar de modo óptimo los ciclos de los semáforos que rigen el cruce mencionado, al menos en las horas punta.

Con los diagramas de barras 4.5, 4.6, 4.7 y 4.8 se pueden observar comparativamente, el número de vehículos en espera cuando el ciclo de los semáforos es fijo (color azul), con las colas que se obtienen después de aplicar el método heurístico (color rosa), durante cinco ciclos (fases 1-15) en cada uno de los cuatro carriles (Carril 1: calle Palomar, carriles 2 y 4: avenida de Finisterre, carril 3: Paseo de los Puentes).

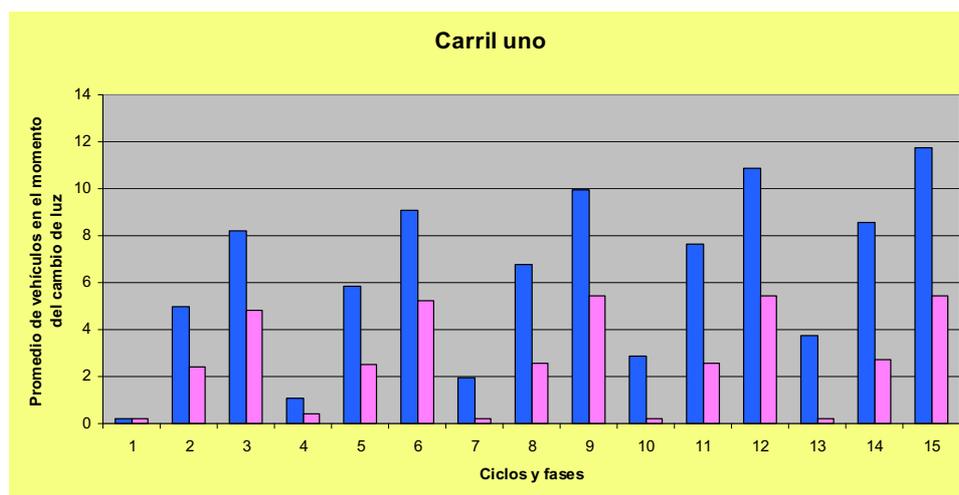


Figura 4.5: Resultados: Color azul-ciclo fijo, color rosa-heurística

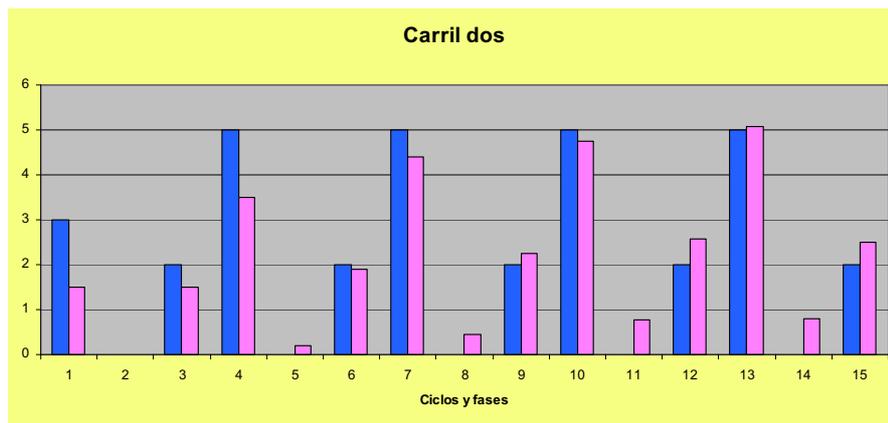


Figura 4.6: Promedio de vehículos en el momento del cambio de luz

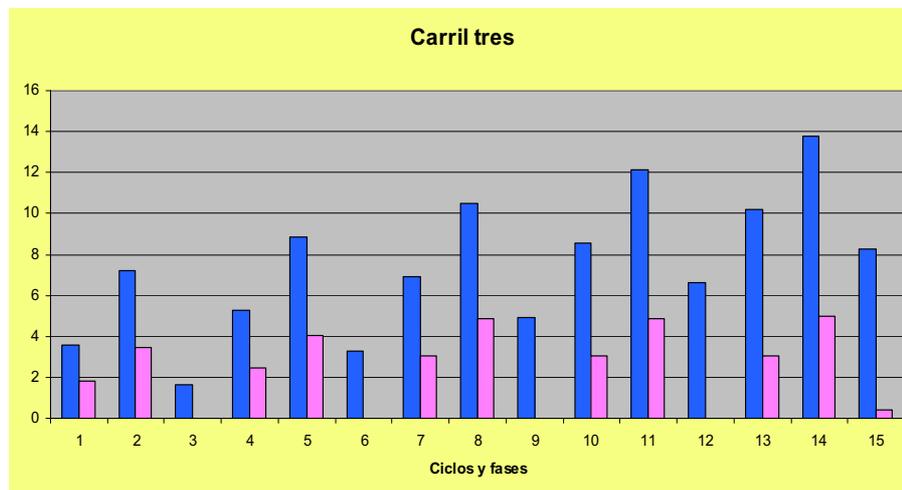


Figura 4.7: Promedio de vehículos en el momento del cambio de luz

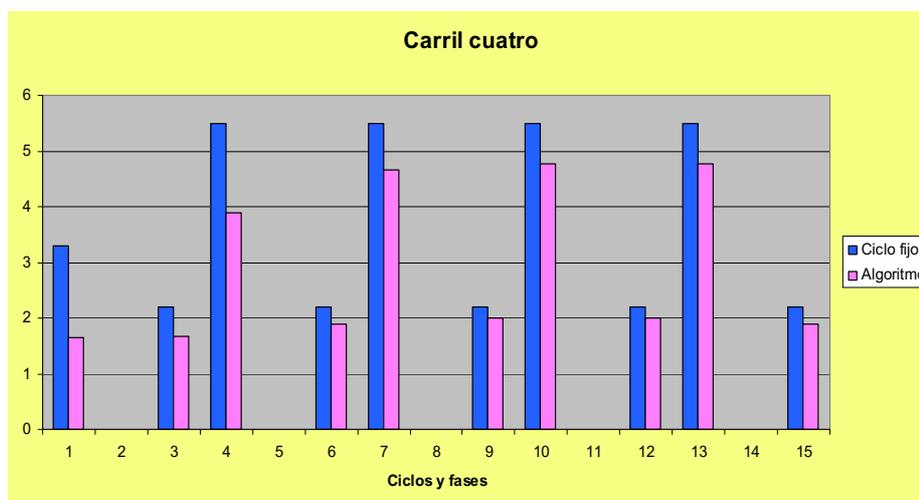


Figura 4.8: Promedio de vehículos en el momento del cambio de luz

4.2. Modelo dinámico

Si se considera la posibilidad de que en los semáforos haya sensores que permitan conocer la cantidad de autos en cada carril y, por ende, se pueda decidir si dado el tráfico, se mantiene el estado actual, por cuanto tiempo, o si debe cambiar la luz, podríamos diseñar, para un cruce regulado por semáforos con tres fases en cada ciclo, un modelo dinámico análogo al expuesto en la sección 3.2.3.

Con este modelo dinámico, usando como algoritmo solución el método heurístico combinado con el quasi-Newton, se obtendrían indudablemente resultados mejores que con el modelo estático, al que se le aplicó el método heurístico simplemente. Pero nos encontramos ante un problema de tipo económico (no matemático) consistente en el incremento de coste que supone la colocación de sensores en los semáforos que regulan la intersección vial.

Nos preguntamos ¿para el cruce de la avenida de Finisterre con las calles Palomar y Paseo de los Puentes, resultaría estrictamente necesario y rentable (en cuanto ahorro en el consumo de combustible, tiempos de espera, contaminación medioambiental) la colocación de sensores? o, por el contrario, a la vista de los buenos resultados obtenidos con el método heurístico como método de resolución aplicado al modelo estático, ¿resultaría suficiente la estimación o simulación de los datos de entrada y salida de vehículos en las horas pico, para obtener una regulación próxima a la óptima de los ciclos de los semáforos en las horas punta?. Pensamos que la respuesta a la segunda pregunta es afirmativa, de ahí que, hayamos optado en este capítulo y para este ejemplo concreto por la implementación exclusivamente del modelo estático.

Capítulo 5

Modelos para cruce de cuatro fases

5.1. Modelo estático para cuatro fases con cuatro semáforos

En esta sección presentamos un modelo matemático para el problema de control de tráfico en un cruce. Se tiene una intersección a la que confluyen dos calles, ambas doble vía en las que se puede circular de frente, girar a derecha e izquierda. En cada esquina del cruce hay un semáforo. Se quiere determinar los períodos en que deben permanecer en verde o rojo las luces de los semáforos T_1 , T_2 , T_3 y T_4 para evitar la congestión de las vías, descrita por el número de vehículos en espera (longitud de la cola) en cada vía. Para ello se consideran distintos criterios: la suma de las longitudes medias de las colas de cada vía, longitud de la cola más larga, tiempo medio de espera o una combinación de ellas. De modo que, el objetivo es calcular una sucesión temporal switching óptima para semáforos que minimice alguno de los criterios anteriores. Con este planteamiento, hemos de modelar y resolver un problema que en cada ciclo tiene cuatro fases: ver figuras 5.1, 5.2, 5.3 y 5.4.

- En la **primera fase** el semáforo T_1 está verde y T_2 , T_3 y T_4 permanecen en rojo, por tanto los coches del carril L_1 rebasan el cruce, continuando de frente o girando a la derecha o a la izquierda.

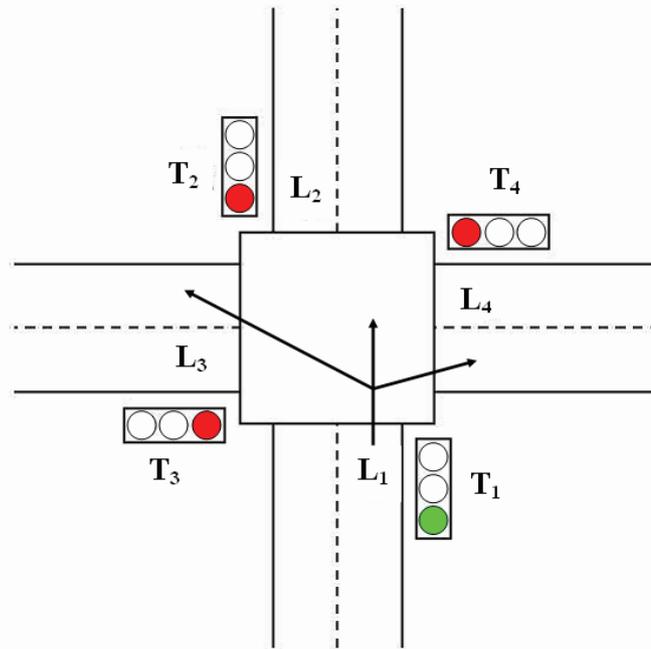


Figura 5.1: 1ª fase

- En la **segunda fase** el semáforo T_2 que ha cambiado de rojo a verde, permite que los vehículos que circulan por el carril L_2 rebasen el cruce siguiendo de frente o girando a la derecha o izquierda (T_1 , T_3 y T_4 están en rojo).
- En la **tercera fase** el semáforo T_3 que ha cambiado de rojo a verde, permite que los vehículos que circulan por el carril L_3 rebasen el cruce siguiendo de frente o girando a la derecha o izquierda (T_1 , T_2 y T_4 están en rojo).
- En la **cuarta fase** el semáforo T_4 que ha cambiado de rojo a verde, permite que los vehículos que circulan por el carril L_4 rebasen el cruce siguiendo de frente o girando a la derecha o izquierda (T_1 , T_2 y T_3 están en rojo).

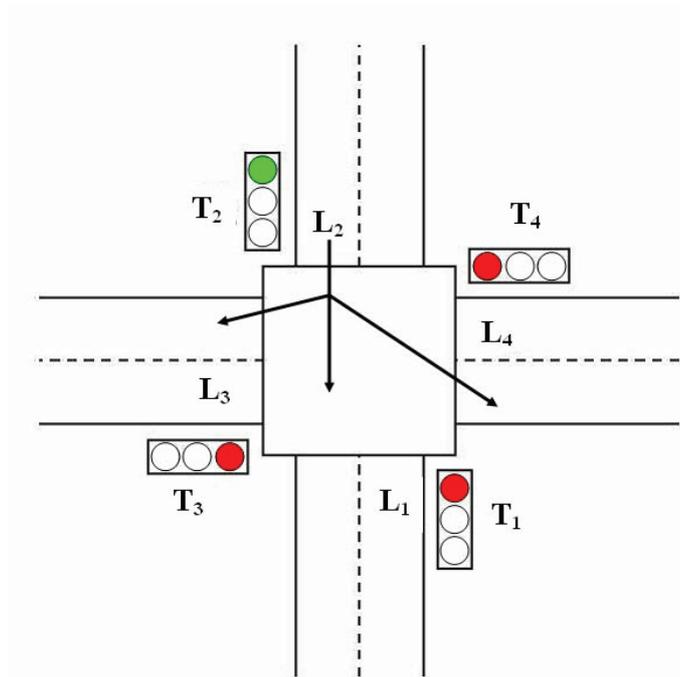


Figura 5.2: 2ª fase

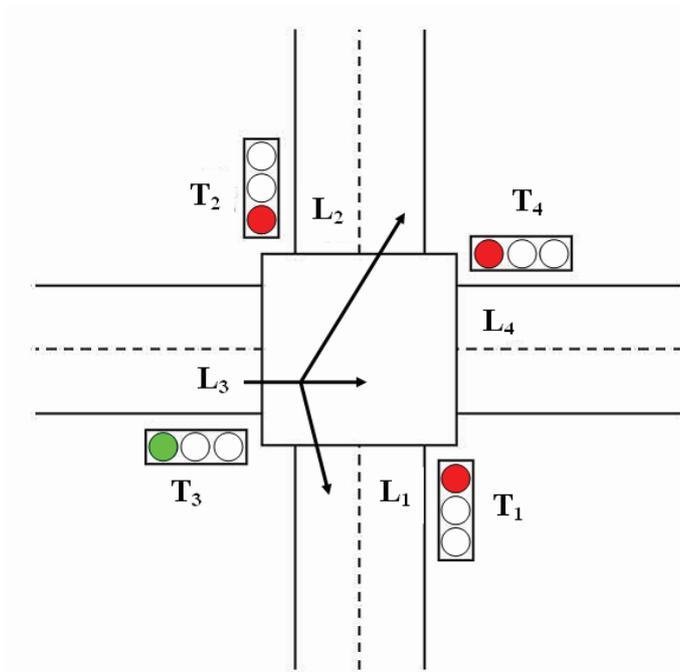


Figura 5.3: 3ª fase

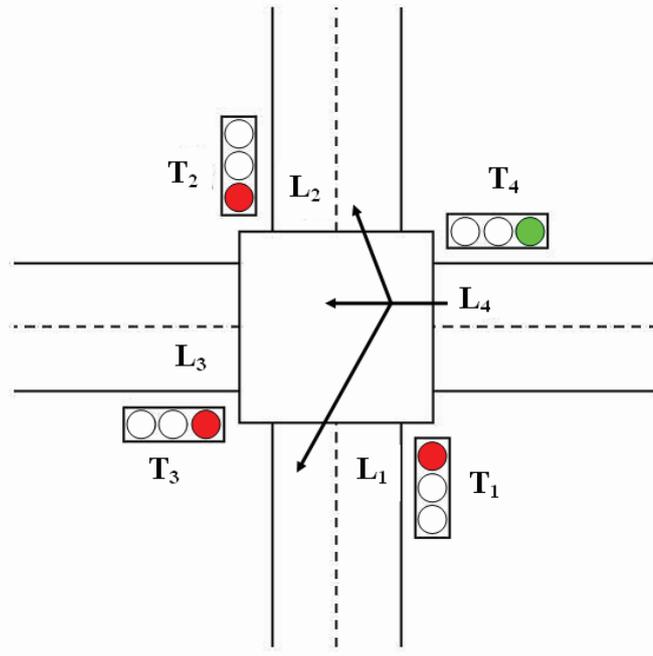


Figura 5.4: 4ª fase

Consideramos dado el número de veces que los semáforos tienen la luz verde y lo denotamos por N . También se considera pre-establecido δ_{amb} que representa el tiempo de duración de la luz ámbar (en la práctica se considera una constante que fluctúa entre los 3 y los 4 segundos).

Describimos el problema con las variables de control δ_k , tiempo de duración de la luz verde en el k -ésimo cambio de luz (incluyendo el ámbar) y las variables de estado $x_k = (x_{k1}, x_{k2}, x_{k3}, x_{k4})^t$, cantidad promedio de vehículos en los carriles L_1, L_2, L_3, L_4 en el momento del k -ésimo cambio de luz.

Como simplificaciones del modelo trabajaremos con el comportamiento promedio por lo que asumimos que la cantidad de autos es continua. Para cada carril L_i se definen las tasas medias de llegada y salida de los vehículos bajo las luces verde y ámbar y se denotan:

λ_i : tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo).

μ_i : tasa media de salida en el carril L_i cuando el semáforo está en verde.

κ_i : tasa media de salida en el carril L_i cuando el semáforo está en ámbar.

Los vectores b_i , $i=1, \dots, 12$ representan el número de vehículos en cada carril te-

niendo en cuenta los movimientos asociados a cada fase:

$$\begin{aligned}
 b_1 &= \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix}, b_2 = \begin{bmatrix} \lambda_1 \\ \lambda_2 - \mu_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix}, b_3 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 \end{bmatrix}, b_4 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 - \mu_4 \end{bmatrix}, \\
 b_5 &= \begin{bmatrix} (\mu_1 - \kappa_1)\delta_{amb} \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_6 = \begin{bmatrix} 0 \\ (\mu_2 - \kappa_2)\delta_{amb} \\ 0 \\ 0 \end{bmatrix}, b_7 = \begin{bmatrix} 0 \\ 0 \\ (\mu_3 - \kappa_3)\delta_{amb} \\ 0 \end{bmatrix}, \\
 b_8 &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ (\mu_4 - \kappa_4)\delta_{amb} \end{bmatrix}, b_9 = \begin{bmatrix} \max((\lambda_1 - \kappa_1)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{10} = \begin{bmatrix} 0 \\ \max((\lambda_2 - \kappa_2)\delta_{amb}, 0) \\ 0 \\ 0 \end{bmatrix}, \\
 b_{11} &= \begin{bmatrix} 0 \\ 0 \\ \max((\lambda_3 - \kappa_3)\delta_{amb}, 0) \\ 0 \end{bmatrix}, b_{12} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \max((\lambda_4 - \kappa_4)\delta_{amb}, 0) \end{bmatrix},
 \end{aligned}$$

Entonces:

$$\begin{aligned}
 x_{4k+1} &= \max(x_{4k} + b_1\delta_{4k+1} + b_5, b_9) & \text{para } k = 0, 1, \dots, N-1, \\
 x_{4k+2} &= \max(x_{4k+1} + b_2\delta_{4k+2} + b_6, b_{10}) & \text{para } k = 0, 1, \dots, N-1, \\
 x_{4k+3} &= \max(x_{4k+2} + b_3\delta_{4k+3} + b_7, b_{11}) & \text{para } k = 0, 1, \dots, N-1, \\
 x_{4k+4} &= \max(x_{4k+3} + b_4\delta_{4k+4} + b_8, b_{12}) & \text{para } k = 0, 1, \dots, N-1,
 \end{aligned}$$

Las características de la zona de tráfico puede imponer otras regulaciones, tales como establecer cotas a las duraciones mínimas y máximas para los tiempos verde y rojo, así:

$\delta_{min.verde,i}$: cota mínima de duración de la luz en semáforo $i=1,2,3,4$.

$\delta_{max.verde,i}$: cota máxima de duración de la luz en semáforo $i=1,2,3,4$.

El problema de regulación de los semáforos puede representarse a través del siguiente modelo:

$$\text{Minimizar } J \quad (5.1)$$

sujeto a:

$$\delta_{min.verde,1} \leq \delta_{4k+1} - \delta_{amb} \leq \delta_{max.verde,1} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.2)$$

$$\delta_{min.verde,2} \leq \delta_{4k+2} - \delta_{amb} \leq \delta_{max.verde,2} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.3)$$

$$\delta_{min.verde,3} \leq \delta_{4k+3} - \delta_{amb} \leq \delta_{max.verde,3} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.4)$$

$$\delta_{min.verde,4} \leq \delta_{4k+4} - \delta_{amb} \leq \delta_{max.verde,4} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.5)$$

$$x_{4k+1} = \max(x_{4k} + b_1\delta_{4k+1} + b_5, b_9) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.6)$$

$$x_{4k+2} = \max(x_{4k+1} + b_2\delta_{4k+2} + b_6, b_{10}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.7)$$

$$x_{4k+3} = \max(x_{4k+2} + b_3\delta_{4k+3} + b_7, b_{11}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.8)$$

$$x_{4k+4} = \max(x_{4k+3} + b_4\delta_{4k+4} + b_8, b_{12}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.9)$$

Si los carriles difieren en importancia al evaluar el nivel de congestión de los mismos, se representa el nivel de importancia del carril, fijando diferentes pesos o ponderaciones $w_i, i = 1, \dots, 4$. La función J a minimizar puede ser:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{j=1}^4 w_j \frac{\sum_{i=1}^{4N} x_{ij} \delta_i}{\sum_{i=1}^{4N} \delta_i} \quad (5.10)$$

- **Longitud (ponderada) media de las colas en el carril con mayores colas**

$$J_2 = \max_j w_j \frac{\sum_{i=1}^{4N} x_{ij} \delta_i}{\sum_{i=1}^{4N} \delta_i} \quad (5.11)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,j} w_j x_{ij} \quad (5.12)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{j=1}^4 \frac{w_j \sum_{i=1}^{4N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{4N} \delta_i} \quad (5.13)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_j \frac{w_j \sum_{i=1}^{4N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{4N} \delta_i} \quad (5.14)$$

Se puede analizar una sexta opción que consiste en considerar una combinación de los criterios anteriores. Esto nos lleva a un modelo multi-objetivo que podemos resolver tomando una combinación positiva de las distintas funciones objetivo.

$$J_6 = \sum_{i=1}^5 \alpha_i J_i \quad (5.15)$$

donde los valores α_i corresponden a los pesos que se les dan a los distintos usuarios.

Como consecuencia de lo anteriormente expuesto, conocido δ (vector de los tiempos de cada luz), la cantidad de autos en cada carril quede determinado mediante las ecuaciones (5.6)-(5.7)-(5.8)-(5.9). Denotamos por $x(\delta)$ la matriz de $4N$ filas y 4 columnas donde $x_{ij}(\delta)$ indica la cantidad de autos en el carril j en el momento del cambio de luz i .

Definimos:

$$\Delta_{ij} = \begin{cases} 1 & \text{si } i \equiv j(4) \\ 0 & \text{si } \text{no} \end{cases}$$

Una forma más compacta de escribir el problema, sin tener que diferenciar entre las cuatro fases es la siguiente:

$$\text{Minimizar } J \quad (5.16)$$

sujeto a:

$$\begin{aligned} \Delta_{k1} \delta_{\min.verde,1} + \Delta_{k2} \delta_{\min.verde,2} + \Delta_{k3} \delta_{\min.verde,3} + \Delta_{k4} \delta_{\min.verde,4} &\leq \delta_k - \delta_{amb} \leq \\ &\leq \Delta_{k1} \delta_{\max.verde,1} + \Delta_{k2} \delta_{\max.verde,2} + \Delta_{k3} \delta_{\max.verde,3} + \Delta_{k4} \delta_{\max.verde,4} \end{aligned} \quad (5.17)$$

para $k = 1, \dots, 4N$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj}\mu_j)\delta_k + \Delta_{kj}(\mu_j - \kappa_j)\delta_{amb}, \max(\Delta_{kj}(\lambda_j - \kappa_j)\delta_{amb}, 0)] \quad (5.18)$$

para $k = 1, \dots, 4N$; $j = 1, 2, 3, 4$

Como habíamos mencionado anteriormente, ya introducimos una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo de las tasas de llegada y de salida, lo que hace recomendable obtener una buena aproximación de la solución óptima que sea calculable en un tiempo corto y fácilmente ajustable a los cambios en los valores de los parámetros. De hecho, utilizando un razonamiento análogo al empleado en la sección 3.1.3.1, llegamos a que estamos ante un problema de optimización con restricciones de complementariedad lineal (problema NP-duro), que involucra a $20N$ variables, $8N$ restricciones suaves y $16N$ no suaves, lo que hace que sea muy costoso aplicar un algoritmo iterativo [19]. Es por eso que para su solución proponemos el algoritmo indicado en la siguiente sección.

5.1.1. Algoritmo solución

Usaremos el algoritmo heurístico inspirado en la metaheurística de recocido simulado utilizado en la sección 3.2.4, adaptado al modelo de la sección 5.1, como método de resolución. Al igual que se hizo en el capítulo tres, se podría combinar el método heurístico con el método quasi-Newton, pero en el ejemplo descrito en la siguiente sección, se comprueba que el método heurístico funciona adecuadamente en un tiempo computacional corto, permitiendo mejorar el flujo de tráfico en las horas punta en la intersección considerada, si lo comparamos con el flujo existente cuando la regulación de semáforos es a ciclo fijo, que es la que en la actualidad se usa en dicho cruce.

El algoritmo y el método de resolución propuestos han sido programados en MATLAB y se han implementado en un ordenador con procesador Intel Core i7, 950 que trabaja a 307 GHz. El programa permite variar los parámetros del algoritmo (temperatura inicial y final, número de repeticiones para una temperatura fija y estrategia de disminución de la temperatura), a fin de posibilitar el ajuste de sus valores en distintas situaciones de tráfico (por ejemplo, en cuanto a densidad de tráfico o número de ciclos).

5.1.2. Aplicación al cruce de la avenida de Finisterre con la Ronda de Nelle

La intersección semaforizada de la avenida de Finisterre con la Ronda de Nelle en la ciudad de A Coruña, es un ejemplo del tipo de cruce presentado y modelado en la sección 5.1, o sea, de cruce regulado por semáforos con las cuatro fases descritas en las figuras 5.1, 5.2, 5.3 y 5.4 en cada ciclo.

En la primera fase, el semáforo situado en la Ronda de Nelle (T_1) se encuentra en color verde, permitiendo que los coches que circulan por dicha calle puedan rebasar el cruce y continuar de frente, por la Ronda de Nelle en sentido hacia el estadio de Riazor, o girar a la derecha o a la izquierda hacia la avenida de Finisterre.

En la segunda fase, el semáforo situado en la Ronda de Nelle en sentido hacia los Cuatro Caminos (T_2) cambia de rojo a verde, permitiendo la circulación en dicho sentido, o girar a la derecha o a la izquierda hacia la avenida de Finisterre.

En la tercera fase, el semáforo situado en la avenida de Finisterre (T_3) cambia de rojo a verde, permitiendo a los vehículos rebasar el cruce, para continuar de frente por dicha avenida en sentido hacia el centro de la ciudad, o girar a la derecha o a la izquierda hacia la Ronda de Nelle.

En la cuarta fase, el semáforo situado en la avenida de Finisterre (T_4) cambia de rojo a verde, permitiendo que los vehículos rebasen el cruce, para continuar de frente por dicha avenida en sentido hacia la salida de la ciudad, o girar a la derecha o a la izquierda hacia la Ronda de Nelle.

En el cruce mencionado hemos de tener en cuenta las siguientes consideraciones, algunas de las cuales pueden apreciarse en la figura 5.5

- Existe un paso elevado que permite la circulación en ambos sentidos por la Ronda de Nelle, de ahí que en las fases 1 y 2 sean pocos los vehículos que continúan de frente una vez que han rebasado el cruce.
- En el semáforo T_1 existe un anexo en donde, en cierto momento, se ilumina en ámbar una flecha que permite girar a la derecha, mientras que la luz de T_1 está en rojo. Esta situación dura el tiempo que permanece en verde la luz del semáforo T_4 y finaliza en el momento en que se pone en verde la luz de T_1 . En este trabajo, esta situación no la hemos contemplado, pues sólo tiene efecto en el tráfico si el coche que ocupa el primer lugar en la cola del carril L_1 gira a la derecha, y



Figura 5.5: Imagen del cruce

los siguientes también, ya que como la vía es muy estrecha, es prácticamente imposible que dos vehículos se sitúen a la par.

- En el semáforo T_2 también existe un anexo en donde, en cierto momento, se ilumina en ámbar una flecha que permite girar a la derecha, mientras que la luz de T_2 está en rojo. Esta situación dura el tiempo que permanece en verde la luz del semáforo T_3 finalizando en el momento en que se pone en verde la luz de T_4 . En este trabajo, esta situación no la hemos contemplado, pues sólo tiene efecto en el tráfico si el coche que ocupa el primer lugar (y los siguientes) en la cola del carril L_2 gira a la derecha, ya que como la vía es muy estrecha es imposible que dos vehículos se sitúen a la par.
- Teniendo en cuenta las consideraciones anteriores, es fácilmente comprensible que el tiempo que los responsables de tráfico han asignado para la luz verde de los semáforos T_1 y T_2 sea bajo, concretamente 14 sg. y 12 sg. respectivamente.
- En la avenida de Finisterre en sentido hacia la salida de la ciudad, antes de rebasar el cruce, el carril L_4 se convierte en doble carril, siendo uno para los coches que giran a la derecha hacia la Ronda de Nelle, y otro para los que continúan de frente o giran a la izquierda, de ahí que cuando la luz del semáforo T_4 está verde

(22 sg. en ciclo fijo) puedan rebasar el cruce un número elevado de vehículos (en torno a los 14) a pesar de que la calzada esté en rampa.

- El tiempo que los responsables del tráfico han asignado a la luz verde del semáforo T_3 en ciclo fijo es de 30 sg.
- La duración de un ciclo completo (en ciclo fijo) es de 78 sg. que se encuentra en el rango entre 35 sg. y 120 sg. que según [66] son los tiempos que en la práctica mejor se acomodan a la mentalidad del usuario de la vía pública.

A continuación, se analiza la validez del modelo propuesto para representar la situación del cruce mencionado. Se observa, a partir de datos reales de entrada y salida de vehículos en el cruce, que el número de vehículos que esperan en los semáforos según el modelo, es aproximadamente el mismo que los que en la realidad constituyen las colas en las horas punta. Luego, usando esos mismos datos y el método heurístico propuesto para la resolución del problema, se obtiene una regulación variable de los tiempos verde de los semáforos para llegar a optimizar las diferentes funciones objetivo, con lo cual se obtiene una reducción en la longitud de las colas, lo que permite una circulación más fluida, una reducción del gasto de combustible, disminución de la contaminación ambiental y de los accidentes.

5.1.3. Resultados computacionales

Se consideran en este ejemplo cinco ciclos ($N=5$). Por mediciones directas en observaciones de varios días durante el horario pico, se estimaron las tasas de llegada y salida en los carriles.

$$\lambda_1=0.09, \lambda_2=0.08, \lambda_3=0.19, \lambda_4=0.21 \text{ (vehículos por segundo)}$$

$$\mu_1=0.57, \mu_2=0.6, \mu_3=0.52, \mu_4=0.83 \text{ (vehículos por segundo)}$$

$$\kappa_1 = \kappa_2 = \kappa_3 = \kappa_4=0.1 \text{ (vehículos por segundo)}. \delta_{amb} = 3 \text{ sg.}$$

$$\delta_{min.verde,1} = \delta_{min.verde,2} = \delta_{min.verde,3} = \delta_{min.verde,4} = 8 \text{ sg.}$$

$$\delta_{max.verde,1} = 20 \text{ sg.}, \delta_{max.verde,2} = 16 \text{ sg.}, \delta_{max.verde,3} = 52 \text{ sg.}, \delta_{max.verde,4} = 36 \text{ sg.}$$

Pesos $w=[1 \ 1 \ 1 \ 1]$. En ciclo fijo, el tiempo verde (incluido el ámbar) para los semáforos es: para T_1 : 14 sg., para T_2 : 12 sg., para T_3 : 30 sg. y para T_4 : 22 sg.

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1º-1ª fase	0	1.12	2.66	2.94
Ciclo 1º-2ª fase	1.08	0	4.94	5.46
Ciclo 1º-3ª fase	3.78	2.4	0.27	11.76
Ciclo 1º-4ª fase	5.76	4.16	4.45	0.33
Ciclo 2º-1ª fase	0.45	5.28	7.11	3.27
Ciclo 2º-2ª fase	1.53	0.54	9.39	5.79
Ciclo 2º-3ª fase	4.23	2.94	0.75	12.09
Ciclo 2º-4ª fase	6.21	4.7	4.93	0.545
Ciclo 3º-1ª fase	0.9	5.82	7.59	3.485
Ciclo 3º-2ª fase	1.98	1.08	9.87	6
Ciclo 3º-3ª fase	4.68	3.48	1.23	12.3
Ciclo 3º-4ª fase	6.66	5.24	5.41	0.76
Ciclo 4º-1ª fase	1.35	6.36	8	3.7
Ciclo 4º-2ª fase	2.43	1.62	10.35	6.22
Ciclo 4º-3ª fase	5.13	4	1.71	12.52
Ciclo 4º-4ª fase	7.11	5.78	5.89	0.97
Ciclo 5º-1ª fase	1.8	6.9	8.55	3.91
Ciclo 5º-2ª fase	2.88	2.16	10.83	6.43
Ciclo 5º-3ª fase	5.58	4.56	2.19	12.73
Ciclo 5º-4ª fase	7.56	6.32	6.37	1.19

Tabla 5.1: Promedio de autos en el momento del cambio de luz en ciclo fijo

Se ve que la longitud de la peor cola es de 12,73 vehículos, que se alcanza en la avenida de Finisterre en sentido hacia la salida de la ciudad. A pesar de este dato, se observa que la regulación a ciclo fijo determinada por los responsables de tráfico es bastante buena ya que, prácticamente rebasan el cruce todos los vehículos que se encuentran en la cola, cuando se enciende la luz verde del semáforo correspondiente (como máximo quedan dos vehículos en espera).

Partiendo de la misma situación inicial, y de los datos anteriormente señalados, se aplicó el algoritmo propuesto con los siguientes parámetros:

Solución inicial: $\delta_0 = [14 \ 12 \ 30 \ 22 \ 14 \ 12 \ 30 \ 22 \ 14 \ 12 \ 30 \ 22 \ 14 \ 12 \ 30 \ 22 \ 14 \ 12 \ 30 \ 22]$

Temperatura inicial $t_0 = 100000000$

Función de reducción de temperatura $T_M = 0.5T_{M-1}$

Número de iteraciones con igual temperatura $Q = 100$

Criterio de parada $T < 0.000000001$

Los tiempos de luz verde obtenidos son dados por el vector:

$\delta = [19 \ 8 \ 16 \ 18 \ 8 \ 8 \ 24 \ 17 \ 13 \ 8 \ 21 \ 21 \ 8 \ 8 \ 27 \ 17 \ 14 \ 8 \ 18 \ 26]$

La siguiente tabla muestra el estado del sistema en las diferentes fases y ciclos asociadas a estos tiempos de luz verde.

	Carril 1	Carril 2	Carril 3	Carril 4
Ciclo 1°-1ª fase	0	1.52	3.61	3.99
Ciclo 1°-2ª fase	0.72	0	5.13	5.67
Ciclo 1°-3ª fase	2.16	1.28	1.11	9.03
Ciclo 1°-4ª fase	3.78	2.72	4.53	0.33
Ciclo 2°-1ª fase	1.83	3.28	5.86	1.8
Ciclo 2°-2ª fase	2.55	0.62	7.38	3.48
Ciclo 2°-3ª fase	4.71	2.54	0.72	8.52
Ciclo 2°-4ª fase	6.24	3.9	3.95	0.33
Ciclo 3°-1ª fase	1.41	4.94	6.42	3.06
Ciclo 3°-2ª fase	2.13	2.28	7.94	4.74
Ciclo 3°-3ª fase	4	3.96	2.27	9.15
Ciclo 3°-4ª fase	5.91	5.64	6.26	0.33
Ciclo 4°-1ª fase	4.44	6.12	7.4	1.59
Ciclo 4°-2ª fase	5.16	3.46	8.92	3.27
Ciclo 4°-3ª fase	7.59	5.62	1.27	8.94
Ciclo 4°-4ª fase	9.12	6.98	4.5	0.52
Ciclo 5°-1ª fase	3.81	8.1	7.16	3.46
Ciclo 5°-2ª fase	4.53	5.44	8.68	5.14
Ciclo 5°-3ª fase	6.15	6.88	4	8.92
Ciclo 5°-4ª fase	8.49	8.96	8.94	0.33

Tabla 5.2: Promedio de autos en el momento del cambio de luz con heurística

Se ve que la peor cola es de 9,15 vehículos que se alcanza en la avenida de Finisterre en sentido hacia la salida de la ciudad, mientras que en ciclo fijo en esa misma dirección

y sentido era de 12,73. Se aprecia, por tanto, la mejora obtenida con la heurística, a pesar de que se observa que pueden quedar incluso 5 vehículos sin rebasar el cruce durante la etapa de luz verde.

El tiempo computacional es de 1,8 sg. tiempo que se puede considerar adecuado para que on-line se puedan ajustar de modo óptimo los ciclos de los semáforos que rigen el cruce mencionado.

5.2. Modelos estático y dinámico para cuatro fases con ocho semáforos

5.2.1. Modelo estático

En esta sección presentamos un modelo matemático para el problema de control de tráfico en un cruce. Se tiene una intersección a la que confluyen dos calles, ambas doble vía (con dos carriles en cada sentido de circulación), en las que se puede circular de frente, girar a derecha e izquierda. En cada carril antes de rebasar el cruce hay un semáforo, por tanto es un cruce con ocho semáforos. Los semáforos T_1 , T_3 , T_5 y T_7 permiten seguir de frente o girar a la derecha, y los semáforos T_2 , T_4 , T_6 y T_8 sólo permiten girar a la izquierda. Se quiere hallar los períodos en que deben permanecer en verde o rojo las luces de los semáforos T_i , $i = 1, \dots, 8$ para evitar la congestión de las vías, descrita por el número de vehículos en espera (longitud de la cola) en cada vía. Para ello se consideran distintos criterios: la suma de las longitudes medias de las colas de cada vía, longitud de la cola más larga, tiempo medio de espera o una combinación de ellas. Con este planteamiento hemos de modelar y resolver un problema que en cada ciclo tiene cuatro fases: ver figuras 5.6, 5.7, 5.8 y 5.9.

- En la **primera fase** los semáforos T_1 y T_5 están en verde (los demás semáforos están en rojo), por tanto los vehículos que circulan por los carriles L_1 y L_5 rebasan el cruce, continuando de frente o girando a la derecha.
- En la **segunda fase** los semáforos T_2 y T_6 que han cambiado de rojo a verde, permiten que los vehículos que circulan por los carriles L_2 y L_6 rebasen el cruce para girar hacia la izquierda, (los demás semáforos están en rojo).
- En la **tercera fase** los semáforos T_3 y T_7 que han cambiado de rojo a verde permiten que los vehículos que circulan por los carriles L_3 y L_7 rebasen el cruce, continuando de frente o girando a la derecha, (los demás semáforos están en rojo).
- En la **cuarta fase** los semáforos T_4 y T_8 que han cambiado de rojo a verde permiten que los vehículos que circulan por los carriles L_4 y L_8 rebasen el cruce, para girar hacia la izquierda, (los demás semáforos están en rojo).

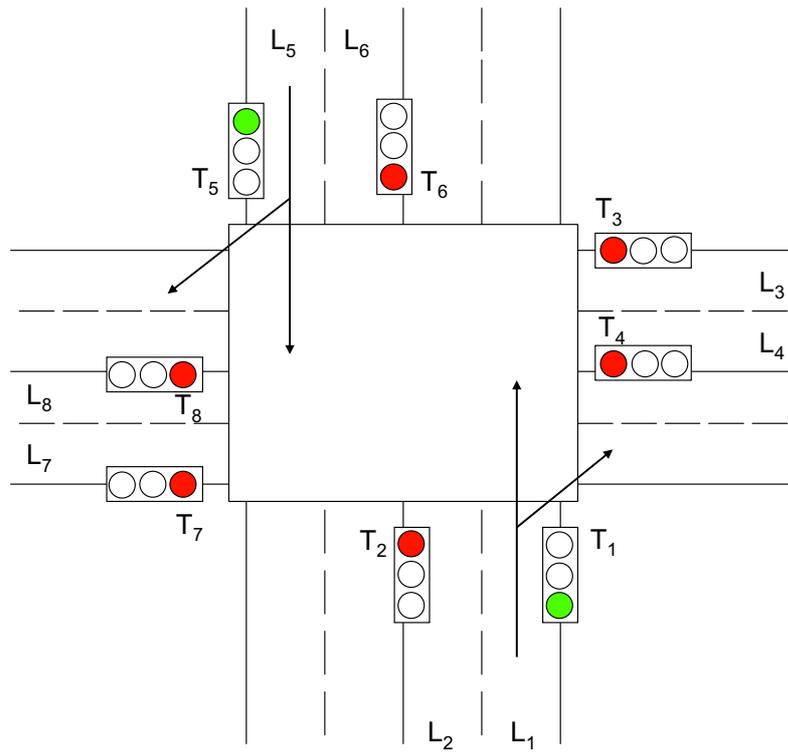


Figura 5.6: 1ª fase

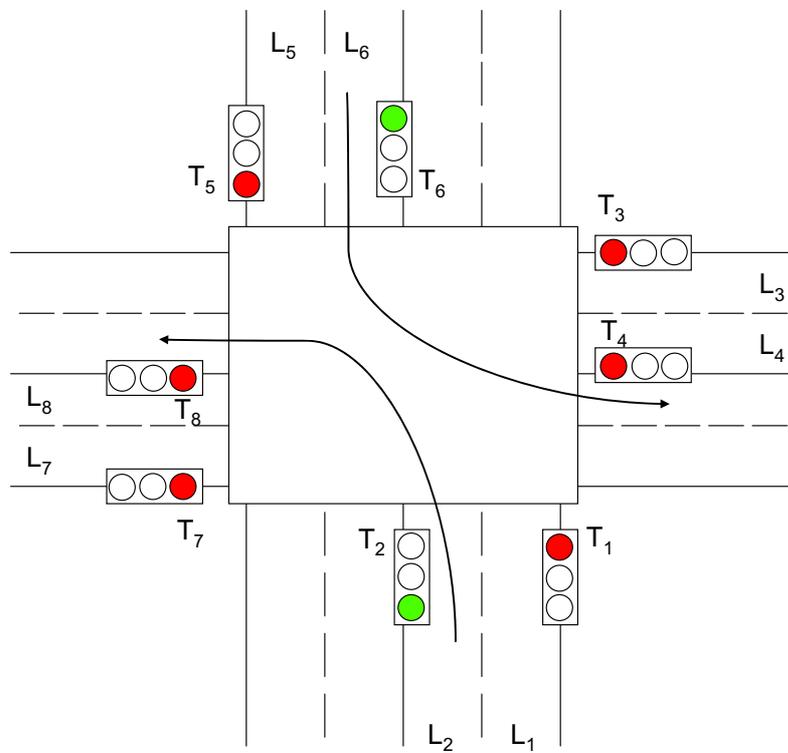


Figura 5.7: 2ª fase

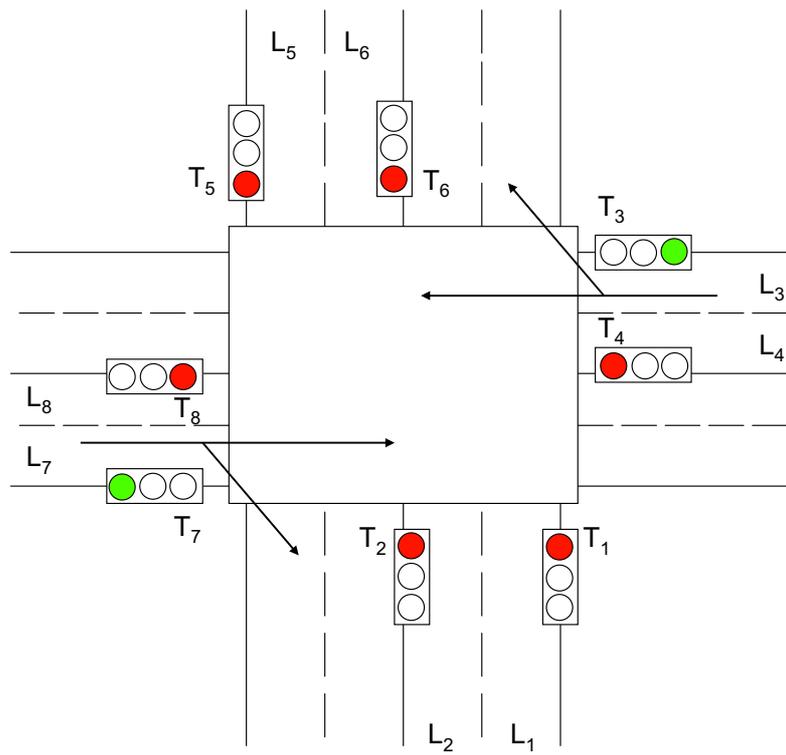


Figura 5.8: 3ª fase

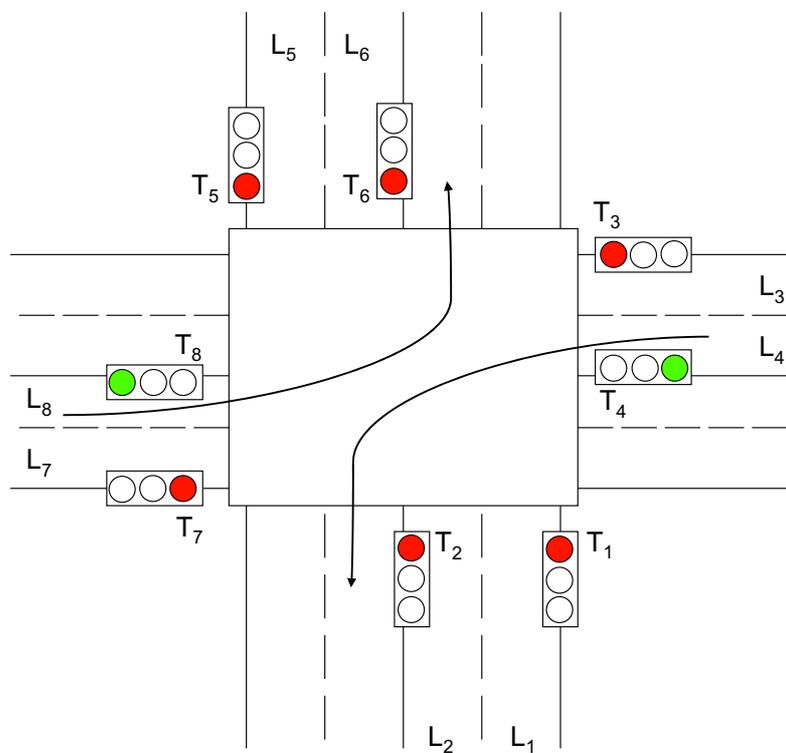


Figura 5.9: 4ª fase

Consideramos dado el número de veces que los semáforos tienen la luz verde y lo denotamos por N . También se considera pre-establecido δ_{amb} que representa el tiempo de duración de la luz ámbar (en la práctica se considera una constante que fluctúa entre los 3 y los 4 segundos).

Describimos el problema *con las variables de control* δ_k , tiempo de duración de la luz verde en el k -ésimo cambio de luz (incluyendo el ámbar) y las *variables de estado* $x_k = (x_{k1}, x_{k2}, x_{k3}, x_{k4}, x_{k5}, x_{k6}, x_{k7}, x_{k8})^t$, cantidad promedio de vehículos en los carriles L_i , $i = 1, \dots, 8$ en el momento del k -ésimo cambio de luz.

Como simplificaciones del modelo trabajaremos con el comportamiento promedio por lo que asumimos que la cantidad de autos es continua. Para cada carril L_i se definen las tasas medias de llegada y salida de los vehículos bajo las luces verde y ámbar y se denotan:

λ_i : tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo).

μ_i : tasa media de salida en el carril L_i cuando el semáforo está en verde.

κ_i : tasa media de salida en el carril L_i cuando el semáforo está en ámbar.

Los vectores b_i , $i=1, \dots, 12$ representan el número de vehículos en cada carril teniendo en cuenta los movimientos asociados a cada fase.

$$\begin{aligned}
 b_1 &= \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 - \mu_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix}, b_2 = \begin{bmatrix} \lambda_1 \\ \lambda_2 - \mu_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 - \mu_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix}, b_3 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 - \mu_7 \\ \lambda_8 \end{bmatrix}, b_4 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 - \mu_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 - \mu_8 \end{bmatrix}, \\
 b_5 &= \begin{bmatrix} (\mu_1 - \kappa_1)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ (\mu_5 - \kappa_5)\delta_{amb} \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_6 = \begin{bmatrix} 0 \\ (\mu_2 - \kappa_2)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ (\mu_6 - \kappa_6)\delta_{amb} \\ 0 \\ 0 \end{bmatrix}, b_7 = \begin{bmatrix} 0 \\ 0 \\ (\mu_3 - \kappa_3)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ (\mu_7 - \kappa_7)\delta_{amb} \\ 0 \end{bmatrix},
 \end{aligned}$$

$$\begin{aligned}
b_8 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ (\mu_4 - \kappa_4)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ (\mu_8 - \kappa_8)\delta_{amb} \end{bmatrix}, & b_9 = \begin{bmatrix} \max((\lambda_1 - \kappa_1)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ \max((\lambda_5 - \kappa_5)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \end{bmatrix}, & b_{10} = \begin{bmatrix} 0 \\ \max((\lambda_2 - \kappa_2)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ \max((\lambda_6 - \kappa_6)\delta_{amb}, 0) \\ 0 \\ 0 \end{bmatrix}, \\
b_{11} = \begin{bmatrix} 0 \\ 0 \\ \max((\lambda_3 - \kappa_3)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ \max((\lambda_7 - \kappa_7)\delta_{amb}, 0) \\ 0 \end{bmatrix}, & b_{12} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \max((\lambda_4 - \kappa_4)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ \max((\lambda_8 - \kappa_8)\delta_{amb}, 0) \end{bmatrix},
\end{aligned}$$

Entonces:

$$\begin{aligned}
x_{4k+1} &= \max(x_{4k} + b_1\delta_{4k+1} + b_5, b_9) & \text{para } k = 0, 1, \dots, N-1, \\
x_{4k+2} &= \max(x_{4k+1} + b_2\delta_{4k+2} + b_6, b_{10}) & \text{para } k = 0, 1, \dots, N-1, \\
x_{4k+3} &= \max(x_{4k+2} + b_3\delta_{4k+3} + b_7, b_{11}) & \text{para } k = 0, 1, \dots, N-1, \\
x_{4k+4} &= \max(x_{4k+3} + b_4\delta_{4k+4} + b_8, b_{12}) & \text{para } k = 0, 1, \dots, N-1,
\end{aligned}$$

Las características de la zona de tráfico puede imponer otras regulaciones, tales como establecer cotas a las duraciones mínimas y máximas para los tiempos verde y rojo, así:

$\delta_{min.verde,i}$: cota mínima de duración de la luz en semáforo $i=1,2,3,4$.

$\delta_{max.verde,i}$: cota máxima de duración de la luz en semáforo $i=1,2,3,4$.

El problema de regulación de los semáforos puede representarse a través del siguiente modelo:

$$\text{Minimizar } J \quad (5.19)$$

sujeto a:

$$\delta_{\min.verde,1} \leq \delta_{4k+1} - \delta_{amb} \leq \delta_{\max.verde,1} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.20)$$

$$\delta_{\min.verde,2} \leq \delta_{4k+2} - \delta_{amb} \leq \delta_{\max.verde,2} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.21)$$

$$\delta_{\min.verde,3} \leq \delta_{4k+3} - \delta_{amb} \leq \delta_{\max.verde,3} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.22)$$

$$\delta_{\min.verde,4} \leq \delta_{4k+4} - \delta_{amb} \leq \delta_{\max.verde,4} \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.23)$$

$$x_{4k+1} = \max(x_{4k} + b_1\delta_{4k+1} + b_5, b_9) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.24)$$

$$x_{4k+2} = \max(x_{4k+1} + b_2\delta_{4k+2} + b_6, b_{10}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.25)$$

$$x_{4k+3} = \max(x_{4k+2} + b_3\delta_{4k+3} + b_7, b_{11}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.26)$$

$$x_{4k+4} = \max(x_{4k+3} + b_4\delta_{4k+4} + b_8, b_{12}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (5.27)$$

Si los carriles difieren en importancia al evaluar el nivel de congestión de los mismos, se representa el nivel de importancia del carril, fijando diferentes pesos o ponderaciones (w_1, \dots, w_8). La función J a minimizar, puede ser:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{j=1}^8 w_j \frac{\sum_{i=1}^{4N} x_{ij} \delta_i}{\sum_{i=1}^{4N} \delta_i} \quad (5.28)$$

- **Longitud (ponderada) media de las colas en el carril con mayores colas**

$$J_2 = \max_j w_j \frac{\sum_{i=1}^{4N} x_{ij} \delta_i}{\sum_{i=1}^{4N} \delta_i} \quad (5.29)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,j} w_j x_{ij} \quad (5.30)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{j=1}^8 \frac{w_j \sum_{i=1}^{4N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{4N} \delta_i} \quad (5.31)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_j \frac{w_j \sum_{i=1}^{4N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{4N} \delta_i} \quad (5.32)$$

Se puede analizar una sexta opción que consiste en considerar una combinación de los criterios anteriores. Esto nos lleva a un modelo multi-objetivo que podemos resolver tomando una combinación positiva de las distintas funciones objetivo.

$$J_6 = \sum_{i=1}^5 \alpha_i J_i \quad (5.33)$$

donde los valores α_i corresponden a los pesos que se les dan a los distintos usuarios.

Como consecuencia de lo anteriormente expuesto, conocido δ (vector de los tiempos de cada luz), la cantidad de autos en cada carril quede determinado mediante las ecuaciones (5.24)-(5.25)-(5.26)-(5.27). Denotamos por $x(\delta)$ la matriz de $4N$ filas y 8 columnas donde $x_{ij}(\delta)$ indica la cantidad de autos en el carril j en el momento del cambio de luz i .

Definimos:

$$\Delta_{ij} = \begin{cases} 1 & \text{si } i \equiv j(4) \\ 0 & \text{si } \text{no} \end{cases}$$

Una forma más compacta de escribir el problema, sin tener que diferenciar entre las cuatro fases es la siguiente:

$$\text{Minimizar } J \quad (5.34)$$

sujeto a:

$$\begin{aligned} \Delta_{k1} \delta_{\min.verde,1} + \Delta_{k2} \delta_{\min.verde,2} + \Delta_{k3} \delta_{\min.verde,3} + \Delta_{k4} \delta_{\min.verde,4} &\leq \delta_k - \delta_{amb} \leq \\ &\leq \Delta_{k1} \delta_{\max.verde,1} + \Delta_{k2} \delta_{\max.verde,2} + \Delta_{k3} \delta_{\max.verde,3} + \Delta_{k4} \delta_{\max.verde,4} \end{aligned} \quad (5.35)$$

para $k = 1, \dots, 4N$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj}\mu_j)\delta_k + \Delta_{kj}(\mu_j - \kappa_j)\delta_{amb}, \max(\Delta_{kj}(\lambda_j - \kappa_j)\delta_{amb}, 0)] \quad (5.36)$$

para $k = 1, \dots, 4N$; $j = 1, \dots, 8$

Como habíamos mencionado anteriormente, ya introducimos una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo de las tasas de llegada y de salida, lo que hace recomendable obtener una buena aproximación de la solución óptima que sea calculable en un tiempo corto y fácilmente ajustable a los cambios en los valores de los parámetros. De hecho, utilizando un razonamiento análogo al empleado en la sección 3.1.3.1, llegamos a que estamos ante un problema de optimización con restricciones de complementariedad lineal (problema NP-duro), que involucra a $36N$ variables, $8N$ restricciones suaves y $32N$ no suaves, lo que hace que sea muy costoso aplicar un algoritmo iterativo [19]. Es por eso que para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves, como el descrito en la sección 3.2.4.

5.2.2. Modelo dinámico

En este modelo consideramos que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz. Esta decisión permite menores tiempos de espera, pues elimina situaciones indeseadas como sería que no hubiera autos en el carril con la luz verde mientras otros carriles esperan.

Para la estrategia de cambio consideramos que si hay al menos r_2 vehículos esperando en carriles con luz roja, entonces es recomendable cambiar, así como si hay a lo más r_1 autos en los carriles a los que corresponde la luz verde. Dado que puede ser muy restrictivo asumir que los cambios de luces ocurren en estas situaciones, suavizamos el modelo planteando que se minimiza el error cuadrático de aproximar la cantidad de autos en los momentos de cambio en los carriles en espera por r_2 y en los carriles de servicio por r_1 .

O sea si la cantidad de autos en el carril al que corresponde la luz verde en el momento i es $\sum_j \Delta_{ij}x_{ij}$, debemos agregar a la función objetivo el cuadrado de la diferencia entre este valor y r_1 , pero solo si $\sum_j \Delta_{ij}x_{ij} > r_1$. Agregando la variable auxiliar

ρ_i^{max} y la restricción $\sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max}$ entonces al minimizar $(\rho_i^{max} - r_1^2)$ se cumple lo deseado. Realizando un razonamiento análogo para los autos que esperan por el cambio de luz para pasar y con la ayuda de la variable ρ_i^{min} , la penalidad quedaría

$$\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2$$

donde

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \quad \text{para } i = 1, \dots, 4N \quad \sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \quad \text{para } i = 1, \dots, 4N$$

Usando un parámetro c para controlar el peso que se le da a esta nueva función en el objeto a minimizar, el modelo final es:

$$\text{Minimizar } F = J + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right] \quad (5.37)$$

sujeto a:

$$\begin{aligned} \Delta_{k1} \delta_{min.verde,1} + \Delta_{k2} \delta_{min.verde,2} + \Delta_{k3} \delta_{min.verde,3} + \Delta_{k4} \delta_{min.verde,4} &\leq \delta_k - \delta_{amb} \leq \\ &\leq \Delta_{k1} \delta_{max.verde,1} + \Delta_{k2} \delta_{max.verde,2} + \Delta_{k3} \delta_{max.verde,3} + \Delta_{k4} \delta_{max.verde,4} \end{aligned} \quad (5.38)$$

para $k = 1, \dots, 4N$

$$x_{kj} = \max[x_{k-1j} + (\lambda_j - \Delta_{kj} \mu_j) \delta_k + \Delta_{kj} (\mu_j - \kappa_j) \delta_{amb}, \max(\Delta_{kj} (\lambda_j - \kappa_j) \delta_{amb}, 0)] \quad (5.39)$$

para $k = 1, \dots, 4N$; $j = 1, \dots, 8$

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \quad \text{para } i = 1, \dots, 4N \quad (5.40)$$

$$\sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \quad \text{para } i = 1, \dots, 4N \quad (5.41)$$

Para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves, como el descrito en la sección 3.2.4.

5.2.3. Resultados computacionales

Datos: 5 ciclos ($N=5$), tiempo de ámbar 3 sg. ($\delta_{amb}=3$ sg.) Se usaron los parámetros que se indican en la tabla 5.3

Carriles	1	2	3	4	5	6	7	8
Tasa-llegada (λ)	0.18	0.12	0.18	0.15	0.18	0.12	0.18	0.15
Tasa-salida verde (μ)	0.9	0.6	0.9	0.7	0.9	0.6	0.9	0.7
Tasa-salida ámbar (κ)	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

Tabla 5.3: Parámetros

En cuanto a las cotas del tiempo de las luces, se tomaron:

$$\delta_{min.verde,1} = \delta_{min.verde,2} = \delta_{min.verde,3} = \delta_{min.verde,4} = 10 \text{ sg.}$$

$$\delta_{max.verde,1} = \delta_{max.verde,3} = 30 \text{ sg. y } \delta_{max.verde,2} = \delta_{max.verde,4} = 20 \text{ sg.}$$

Construcción de las funciones objetivo

Pesos $w = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$

$\alpha_i = 1, \forall i = 1, \dots, 5$ (construcción de la función objetivo J_6)

$r_1 = 2, r_2 = 8$ (construcción de las funciones objetivo del modelo dinámico)

Solución inicial: $\delta_0 = [20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15 \ 20 \ 15]$

Elementos para la construcción de la heurística

Temperatura inicial $t_0 = 100000000$

Función de reducción de temperatura $T_M = 0.5 T_{M-1}$

Número de iteraciones con igual temperatura $Q = 100$

Criterio de parada $T < 0.000000001$

Veremos que en todos los casos, los resultados obtenidos con el modelo dinámico son mejores que los del modelo estático, aunque el tiempo computacional sea algo mayor.

Para $J = J_1$ obtuvimos los resultados indicados en las tablas 5.4 y 5.5

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	20.45	18.48	20.16	19.1	16.9
Semáforos 2-6	10.78	15.69	16.08	16	13.01
Semáforos 3-7	15.06	15.68	18.62	18.19	19.65
Semáforos 4-8	16.71	15.95	18.7	20	16.73

Tabla 5.4: Duración de la luz verde en cada ciclo

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	2.45	3.68	3.06	0.24	2.45	3.68	3.06
Ciclo 1°-2ª fase	2.18	0.06	5.62	4.68	2.18	0.06	5.62	4.68
Ciclo 1°-3ª fase	4.89	1.86	0.24	6.94	4.89	1.86	0.24	6.94
Ciclo 1°-4ª fase	7.9	3.87	3.24	0.15	7.9	3.87	3.24	0.15
Ciclo 2°-1ª fase	0.24	6.09	6.57	2.92	0.24	6.09	6.57	2.92
Ciclo 2°-2ª fase	3.06	0.06	9.4	5.27	3.06	0.06	9.4	5.27
Ciclo 2°-3ª fase	5.88	1.94	0.5	7.63	5.88	1.94	0.5	7.63
Ciclo 2°-4ª fase	8.75	3.85	3.37	0.65	8.75	3.85	3.37	0.65
Ciclo 3°-1ª fase	0.24	6.27	7	3.68	0.24	6.27	7	3.68
Ciclo 3°-2ª fase	3.13	0.06	9.9	6.09	3.13	0.06	9.9	6.09
Ciclo 3°-3ª fase	6.48	2.29	0.24	8.9	6.48	2.29	0.24	8.9
Ciclo 3°-4ª fase	9.85	4.53	3.6	0.4	9.85	4.53	3.6	0.4
Ciclo 4°-1ª fase	0.24	6.83	7.04	3.26	0.24	6.83	7.04	3.26
Ciclo 4°-2ª fase	3.11	0.66	9.92	5.66	3.11	0.66	9.92	5.66
Ciclo 4°-3ª fase	6.39	2.84	0.24	8.39	6.39	2.84	0.24	8.39
Ciclo 4°-4ª fase	9.99	5.25	3.84	0.15	9.99	5.25	3.84	0.15
Ciclo 5°-1ª fase	0.24	7.28	6.89	2.68	0.24	7.28	6.89	2.68
Ciclo 5°-2ª fase	2.58	2.53	9.23	4.63	2.58	2.53	9.23	4.63
Ciclo 5°-3ª fase	6.12	4.89	0.24	7.58	6.12	4.89	0.24	7.58
Ciclo 5°-4ª fase	9.13	6.9	3.25	0.18	9.13	6.9	3.25	0.18

Tabla 5.5: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo = 33,9 autos. Peor cola \cong 10 autos. Tiempo computacional 1,2 sg.

$$Para \quad F_1 = J_1 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados indicados en las tablas 5.6 y 5.7

Valor de la función objetivo = 29,2 autos. Peor cola \cong 8,6 autos. Tiempo computacional 1,9 sg.

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	1.29	1.94	1.62	0.24	1.29	1.94	1.62
Ciclo 1°-2ª fase	3.32	0.06	5	4.18	3.32	0.06	5	4.18
Ciclo 1°-3ª fase	5.29	1.37	0.24	5.82	5.29	1.37	0.24	5.82
Ciclo 1°-4ª fase	7.67	2.96	2.62	0.34	7.67	2.96	2.62	0.34
Ciclo 2°-1ª fase	0.24	4.71	5.24	2.52	0.24	4.71	5.24	2.52
Ciclo 2°-2ª fase	2.69	0.06	7.69	4.56	2.69	0.06	7.69	4.56
Ciclo 2°-3ª fase	5.35	1.84	0.24	6.79	5.35	1.84	0.24	6.79
Ciclo 2°-4ª fase	8.05	3.63	2.93	0.36	8.05	3.63	2.93	0.36
Ciclo 3°-1ª fase	0.24	5.52	5.76	2.72	0.24	5.52	5.76	2.72
Ciclo 3°-2ª fase	2.85	0.06	8.38	4.9	2.85	0.06	8.38	4.9
Ciclo 3°-3ª fase	5.52	1.83	0.24	7.12	5.52	1.83	0.24	7.12
Ciclo 3°-4ª fase	8.6	3.89	3.32	0.15	8.6	3.89	3.32	0.15
Ciclo 4°-1ª fase	0.24	5.72	6.06	2.43	0.24	5.72	6.06	2.43
Ciclo 4°-2ª fase	2.42	1.4	8.24	4.25	2.42	1.4	8.24	4.25
Ciclo 4°-3ª fase	5.16	3.24	0.24	6.53	5.16	3.24	0.24	6.53
Ciclo 4°-4ª fase	8	5.13	3.1	0.15	8	5.13	3.1	0.15
Ciclo 5°-1ª fase	0.24	6.83	5.63	2.28	0.24	6.83	5.63	2.28
Ciclo 5°-2ª fase	2.91	1.2	8.3	4.5	2.91	1.2	8.3	4.5
Ciclo 5°-3ª fase	5.76	3.1	0.24	6.88	5.76	3.1	0.24	6.88
Ciclo 5°-4ª fase	8.53	4.95	3	0.2	8.53	4.95	3	0.2

Tabla 5.6: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	10.79	14.54	15.73	15.22	14.19
Semáforos 2-6	17.11	13.6	14.52	12.1	14.85
Semáforos 3-7	10.95	14.83	14.82	15.23	15.83
Semáforos 4-8	13.24	14.95	17.11	15.78	15.4

Tabla 5.7: Duración de la luz verde en cada ciclo

Para $J = J_2$ obtuvimos los resultados indicados en las tablas 5.8 y 5.9

Carriles	1	2	3	4	5	6	7	8
Ciclo 1 ^o -1 ^a fase	0.24	2.35	3.53	2.94	0.24	2.35	3.53	2.94
Ciclo 1 ^o -2 ^a fase	2.87	0.06	6.17	5.14	2.87	0.06	6.17	5.14
Ciclo 1 ^o -3 ^a fase	4.85	1.37	0.66	6.79	4.85	1.37	0.66	6.79
Ciclo 1 ^o -4 ^a fase	7.95	3.44	3.77	0.15	7.95	3.44	3.77	0.15
Ciclo 2 ^o -1 ^a fase	0.24	5.91	7.47	3.23	0.24	5.91	7.47	3.23
Ciclo 2 ^o -2 ^a fase	3.17	0.06	10.41	5.68	3.17	0.06	10.41	5.68
Ciclo 2 ^o -3 ^a fase	6.65	2.37	0.24	8.58	6.65	2.37	0.24	8.58
Ciclo 2 ^o -4 ^a fase	10.3	4.8	3.88	0.15	10.3	4.8	3.88	0.15
Ciclo 3 ^o -1 ^a fase	0.24	7.27	7.57	3.22	0.24	7.27	7.57	3.22
Ciclo 3 ^o -2 ^a fase	2.85	1.79	10.19	5.4	2.85	1.79	10.19	5.4
Ciclo 3 ^o -3 ^a fase	7.37	4.81	0.24	9.17	7.37	4.81	0.24	9.17
Ciclo 3 ^o -4 ^a fase	9.97	6.54	2.84	3	9.97	6.54	2.84	3
Ciclo 4 ^o -1 ^a fase	0.24	9.14	6.73	6.25	0.24	9.14	6.73	6.25
Ciclo 4 ^o -2 ^a fase	2.69	4.09	9.19	8.3	2.69	4.09	9.19	8.3
Ciclo 4 ^o -3 ^a fase	6.41	6.56	0.24	11.4	6.41	6.56	0.24	11.4
Ciclo 4 ^o -4 ^a fase	9.25	8.46	3.08	4.52	9.25	8.46	3.08	4.52
Ciclo 5 ^o -1 ^a fase	0.24	10.71	6.46	7.34	0.24	10.71	6.46	7.34
Ciclo 5 ^o -2 ^a fase	3.08	4.63	9.3	9.71	3.08	4.63	9.3	9.71
Ciclo 5 ^o -3 ^a fase	6.3	6.76	0.24	12.38	6.3	6.76	0.24	12.38
Ciclo 5 ^o -4 ^a fase	8.64	8.33	2.59	6.97	8.64	8.33	2.59	6.97

Tabla 5.8: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	19.64	20.58	20.5	21.63	18.79
Semáforos 2-6	14.65	16.32	14.52	13.65	15.8
Semáforos 3-7	11	19.31	25.1	20.63	17.79
Semáforos 4-8	17.22	20	14.47	15.77	13.1

Tabla 5.9: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 6,03 autos. Peor cola \cong 12,38 autos. Tiempo computacional 0,7 sg.

$$\text{Para } F_2 = J_2 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados reflejados en las tablas 5.10 y 5.11

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	15.94	16.91	20.73	18.92	22.96
Semáforos 2-6	12.95	13.13	17.47	19	13
Semáforos 3-7	27.9	13.92	18.95	24.93	20.69
Semáforos 4-8	15.12	18.06	20	18	15

Tabla 5.10: Duración de la luz verde en cada ciclo

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	1.91	2.87	2.39	0.24	1.91	2.87	2.39
Ciclo 1°-2ª fase	2.57	0.06	5.2	4.33	2.57	0.06	5.2	4.33
Ciclo 1°-3ª fase	7.59	3.4	0.24	8.5	7.59	3.4	0.24	8.5
Ciclo 1°-4ª fase	10.31	5.22	2.96	2	10.31	5.22	2.96	2
Ciclo 2°-1ª fase	0.53	7.25	6	4.54	0.53	7.25	6	4.54
Ciclo 2°-2ª fase	2.89	2.45	8.37	6.5	2.89	2.45	8.37	6.5
Ciclo 2°-3ª fase	5.4	4.1	0.74	8.59	5.4	4.1	0.74	8.59
Ciclo 2°-4ª fase	8.65	6.28	4	0.46	8.65	6.28	4	0.46
Ciclo 3°-1ª fase	0.24	8.77	7.72	3.57	0.24	8.77	7.72	3.57
Ciclo 3°-2ª fase	3.38	1.88	10.87	6.19	3.38	1.88	10.87	6.19
Ciclo 3°-3ª fase	6.8	4.16	0.24	9.03	6.8	4.16	0.24	9.03
Ciclo 3°-4ª fase	10.39	6.56	3.84	0.15	10.39	6.56	3.84	0.15
Ciclo 4°-1ª fase	0.24	8.83	7.24	2.99	0.24	8.83	7.24	2.99
Ciclo 4°-2ª fase	3.66	1.2	10.67	5.84	3.66	1.2	10.67	5.84
Ciclo 4°-3ª fase	8.15	4.18	0.24	9.58	8.15	4.18	0.24	9.58
Ciclo 4°-4ª fase	11.41	6.35	3.49	1.42	11.41	6.35	3.49	1.42
Ciclo 5°-1ª fase	0.24	9.11	7.63	4.87	0.24	9.11	7.63	4.87
Ciclo 5°-2ª fase	2.58	4.36	10	6.82	2.58	4.36	10	6.82
Ciclo 5°-3ª fase	6.3	6.85	0.24	9.93	6.3	6.85	0.24	9.93
Ciclo 5°-4ª fase	9	8.65	2.94	3.47	9	8.65	2.94	3.47

Tabla 5.11: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo = 5,23 autos. Peor cola \cong 11,4 autos. Tiempo computacional 0,8 sg.

Para $J = J_3$ obtuvimos los resultados indicados en las tablas 5.12 y 5.13

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	3.48	5.22	4.35	0.24	3.48	5.22	4.35
Ciclo 1°-2ª fase	2.67	0.06	7.65	6.38	2.67	0.06	7.65	6.38
Ciclo 1°-3ª fase	6.03	2.29	0.24	9.17	6.03	2.29	0.24	9.17
Ciclo 1°-4ª fase	9.56	4.65	3.77	0.18	9.56	4.65	3.77	0.18
Ciclo 2°-1ª fase	0.24	7.35	7.83	3.56	0.24	7.35	7.83	3.56
Ciclo 2°-2ª fase	3.16	1.04	10.76	6	3.16	1.04	10.76	6
Ciclo 2°-3ª fase	6.45	3.24	0.24	8.74	6.45	3.24	0.24	8.74
Ciclo 2°-4ª fase	9.39	5.19	3.17	1.58	9.39	5.19	3.17	1.58
Ciclo 3°-1ª fase	3.17	6.63	5.32	3.37	3.17	6.63	5.32	3.37
Ciclo 3°-2ª fase	3.88	0.9	8	5.63	3.88	0.9	8	5.63
Ciclo 3°-3ª fase	8.07	2.36	1.68	7.46	8.07	2.36	1.68	7.46
Ciclo 3°-4ª fase	10.46	3.95	4.1	1.94	10.46	3.95	4.1	1.94
Ciclo 4°-1ª fase	0.24	6.27	7.55	4.83	0.24	6.27	7.55	4.83
Ciclo 4°-2ª fase	3.13	0.06	10.44	7.24	3.13	0.06	10.44	7.24
Ciclo 4°-3ª fase	6.95	2.61	0.24	10.43	6.95	2.61	0.24	10.43
Ciclo 4°-4ª fase	10.47	4.96	3.75	1.5	10.47	4.96	3.75	1.5
Ciclo 5°-1ª fase	1	7	6.72	4	1	7	6.72	4
Ciclo 5°-2ª fase	4.1	0.15	10	6.57	4.1	0.15	10	6.57
Ciclo 5°-3ª fase	7.75	2.58	0.24	9.61	7.75	2.58	0.24	9.61
Ciclo 5°-4ª fase	10.41	4.35	2.89	3.29	10.41	4.35	2.89	3.29

Tabla 5.12: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	29	22.55	12	19.27	16.5
Semáforos 2-6	13.5	16.27	15	16	17.27
Semáforos 3-7	18.64	18.27	12.15	21.27	20.27
Semáforos 4-8	19.63	16.29	13.29	19.51	14.75

Tabla 5.13: Duración de la luz verde en cada ciclo

Peor cola \cong 11 autos. Tiempo computacional 1,1 sg.

$$Para \quad F_3 = J_3 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados indicados en las tablas 5.14 y 5.15

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	19.06	16.69	18.69	17.72	19.72
Semáforos 2-6	14.06	15.37	14	14.05	19.05
Semáforos 3-7	10.93	15.06	18.72	18.72	17.06
Semáforos 4-8	16.06	17.06	19.1	19.1	15

Tabla 5.14: Duración de la luz verde en cada ciclo

Peor cola \cong 10,65 autos. Tiempo computacional 1,25 sg.

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	2.28	3.43	2.86	0.24	2.28	3.43	2.86
Ciclo 1°-2ª fase	2.77	0.06	5.96	4.96	2.77	0.06	5.96	4.96
Ciclo 1°-3ª fase	4.73	1.37	0.49	6.6	4.73	1.37	0.49	6.6
Ciclo 1°-4ª fase	7.63	3.3	3.38	0.15	7.63	3.3	3.38	0.15
Ciclo 2°-1ª fase	0.24	5.3	6.39	2.65	0.24	5.3	6.39	2.65
Ciclo 2°-2ª fase	3	0.06	9.15	4.96	3	0.06	9.15	4.96
Ciclo 2°-3ª fase	5.72	1.86	0.7	7.22	5.72	1.86	0.7	7.22
Ciclo 2°-4ª fase	8.79	3.91	3.78	0.15	8.79	3.91	3.78	0.15
Ciclo 3°-1ª fase	0.24	6.16	7.14	2.95	0.24	6.16	7.14	2.95
Ciclo 3°-2ª fase	2.75	0.95	9.66	5.04	2.75	0.95	9.66	5.04
Ciclo 3°-3ª fase	6.12	3.2	0.24	7.85	6.12	3.2	0.24	7.85
Ciclo 3°-4ª fase	9.56	5.49	3.67	0.15	9.56	5.49	3.67	0.15
Ciclo 4°-1ª fase	0.24	7.61	6.86	2.8	0.24	7.61	6.86	2.8
Ciclo 4°-2ª fase	2.76	2.37	9.39	4.91	2.76	2.37	9.39	4.91
Ciclo 4°-3ª fase	6.14	4.62	0.24	7.72	6.14	4.62	0.24	7.72
Ciclo 4°-4ª fase	9.5	6.91	3.67	0.15	9.5	6.91	3.67	0.15
Ciclo 5°-1ª fase	0.24	9.27	7.22	3.1	0.24	9.27	7.22	3.1
Ciclo 5°-2ª fase	3.69	1.63	10.65	6	3.69	1.63	10.65	6
Ciclo 5°-3ª fase	6.74	3.68	0.77	8.52	6.74	3.68	0.77	8.52
Ciclo 5°-4ª fase	9.44	5.48	3.47	2.07	9.44	5.48	3.47	2.07

Tabla 5.15: Promedio de vehículos en el momento del cambio de luz

Para $J = J_4$ obtuvimos los resultados indicados en las tablas 5.16 y 5.17

Carriles	1	2	3	4	5	6	7	8
Ciclo 1 ^o -1 ^a fase	0.24	2.33	3.49	2.91	0.24	2.33	3.49	2.91
Ciclo 1 ^o -2 ^a fase	2.04	0.06	5.29	4.41	2.04	0.06	5.29	4.41
Ciclo 1 ^o -3 ^a fase	3.87	1.28	0.36	5.94	3.87	1.28	0.36	5.94
Ciclo 1 ^o -4 ^a fase	6.62	3.11	3.11	0.15	6.62	3.11	3.11	0.15
Ciclo 2 ^o -1 ^a fase	0.27	4.58	5.3	1.97	0.27	4.58	5.3	1.97
Ciclo 2 ^o -2 ^a fase	3.21	0.06	8.24	4.42	3.21	0.06	8.24	4.42
Ciclo 2 ^o -3 ^a fase	5.5	1.59	1.45	6.34	5.5	1.59	1.45	6.34
Ciclo 2 ^o -4 ^a fase	7.81	3.12	3.76	1.09	7.81	3.12	3.76	1.09
Ciclo 3 ^o -1 ^a fase	1.12	4.64	6.03	2.98	1.12	4.64	6.03	2.98
Ciclo 3 ^o -2 ^a fase	3.85	0.06	8.75	5.26	3.85	0.06	8.75	5.26
Ciclo 3 ^o -3 ^a fase	6.8	2.02	0.24	7.71	6.8	2.02	0.24	7.71
Ciclo 3 ^o -4 ^a fase	9.92	4.1	3.35	0.15	9.92	4.1	3.35	0.15
Ciclo 4 ^o -1 ^a fase	1.91	5.83	5.95	2.31	1.91	5.83	5.95	2.31
Ciclo 4 ^o -2 ^a fase	4.64	0.06	8.68	4.59	4.64	0.06	8.68	4.59
Ciclo 4 ^o -3 ^a fase	7.31	1.84	0.39	6.82	7.31	1.84	0.39	6.82
Ciclo 4 ^o -4 ^a fase	10.16	3.74	3.24	0.15	10.16	3.74	3.24	0.15
Ciclo 5 ^o -1 ^a fase	1.8	5.54	5.93	2.39	1.8	5.54	5.93	2.39
Ciclo 5 ^o -2 ^a fase	4.4	0.1	8.53	4.56	4.4	0.1	8.53	4.56
Ciclo 5 ^o -3 ^a fase	7.38	2.09	0.24	7.04	7.38	2.09	0.24	7.04
Ciclo 5 ^o -4 ^a fase	10.22	4	3.08	0.16	10.22	4	3.08	0.16

Tabla 5.16: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	19.42	12.16	12.61	14.45	14.95
Semáforos 2-6	10	16.33	15.14	15.15	14.45
Semáforos 3-7	10.19	12.75	16.39	14.85	16.58
Semáforos 4-8	15.3	12.8	17.3	15.83	15.79

Tabla 5.17: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 193,18 sg. Peor cola \cong 10,23 autos. Tiempo computacional 2,95 sg.

$$\text{Para } F_4 = J_4 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados indicados en las tablas 5.18 y 5.19

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	26.1	12.95	15.49	14.89	14.42
Semáforos 2-6	10	13.71	15.17	14.42	15.23
Semáforos 3-7	11.79	15.77	15.15	14.61	14.21
Semáforos 4-8	17.37	16.36	15.8	15.27	11

Tabla 5.18: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 185 sg. Peor cola \cong 8,7 autos. Tiempo computacional 3,4 sg.

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	3.13	4.7	3.91	0.24	3.13	4.7	3.91
Ciclo 1°-2ª fase	2	0.06	6.5	5.41	2	0.06	6.5	5.41
Ciclo 1°-3ª fase	4.16	1.47	0.41	7.18	4.16	1.47	0.41	7.18
Ciclo 1°-4ª fase	7.29	3.56	3.53	0.15	7.29	3.56	3.53	0.15
Ciclo 2°-1ª fase	0.35	5.11	5.87	2.1	0.35	5.11	5.87	2.1
Ciclo 2°-2ª fase	2.82	0.06	8.34	4.15	2.82	0.06	8.34	4.15
Ciclo 2°-3ª fase	5.66	1.95	0.24	6.51	5.66	1.95	0.24	6.51
Ciclo 2°-4ª fase	8.61	3.91	3.18	0.15	8.61	3.91	3.18	0.15
Ciclo 3°-1ª fase	0.24	5.77	5.97	2.47	0.24	5.77	5.97	2.47
Ciclo 3°-2ª fase	2.97	0.06	8.7	4.75	2.97	0.06	8.7	4.75
Ciclo 3°-3ª fase	5.7	1.88	0.24	7	5.7	1.88	0.24	7
Ciclo 3°-4ª fase	8.54	3.77	3.08	0.15	8.54	3.77	3.08	0.15
Ciclo 4°-1ª fase	0.24	5.56	5.76	2.38	0.24	5.56	5.76	2.38
Ciclo 4°-2ª fase	2.83	0.13	8.36	4.55	2.83	0.13	8.36	4.55
Ciclo 4°-3ª fase	5.46	1.89	0.24	6.74	5.46	1.89	0.24	6.74
Ciclo 4°-4ª fase	8.22	3.72	3	0.15	8.22	3.72	3	0.15
Ciclo 5°-1ª fase	0.24	5.45	5.58	2.31	0.24	5.45	5.58	2.31
Ciclo 5°-2ª fase	2.98	0.06	8.32	4.59	2.98	0.06	8.32	4.59
Ciclo 5°-3ª fase	5.54	1.76	0.49	6.73	5.54	1.76	0.49	6.73
Ciclo 5°-4ª fase	7.52	3.09	2.48	2.45	7.52	3.09	2.48	2.45

Tabla 5.19: Promedio de vehículos en el momento del cambio de luz

Para $J = J_5$ obtuvimos los resultados indicados en las tablas 5.20 y 5.21

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	1.55	2.33	1.94	0.24	1.55	2.33	1.94
Ciclo 1°-2ª fase	2.59	0.06	4.68	3.9	2.59	0.06	4.68	3.9
Ciclo 1°-3ª fase	5.51	2	0.24	6.34	5.51	2	0.24	6.34
Ciclo 1°-4ª fase	8.77	4.18	3.5	0.15	8.77	4.18	3.5	0.15
Ciclo 2°-1ª fase	0.24	6	6.27	2.45	0.24	6	6.27	2.45
Ciclo 2°-2ª fase	3.71	0.06	9.75	5.35	3.71	0.06	9.75	5.35
Ciclo 2°-3ª fase	6.67	2	0.34	7.81	6.67	2	0.34	7.81
Ciclo 2°-4ª fase	10.05	4.28	3.72	0.15	10.05	4.28	3.72	0.15
Ciclo 3°-1ª fase	0.24	7.08	7.92	3.64	0.24	7.08	7.92	3.64
Ciclo 3°-2ª fase	3.04	1.09	10.73	6	3.04	1.09	10.73	6
Ciclo 3°-3ª fase	5.54	2.76	3.12	8	5.54	2.76	3.12	8
Ciclo 3°-4ª fase	8.82	4.94	6.4	0.15	8.82	4.94	6.4	0.15
Ciclo 4°-1ª fase	0.9	6.66	9	2.3	0.9	6.66	9	2.3
Ciclo 4°-2ª fase	4.21	0.06	12.29	5	4.21	0.06	12.29	5
Ciclo 4°-3ª fase	9.13	3.33	0.24	9.15	9.13	3.33	0.24	9.15
Ciclo 4°-4ª fase	12	5.22	3	2.28	12	5.22	3	2.28
Ciclo 5°-1ª fase	0.93	7.46	6.43	5	0.93	7.46	6.43	5
Ciclo 5°-2ª fase	3.53	2	9	7.24	3.53	2	9	7.24
Ciclo 5°-3ª fase	6.01	3.69	1.5	9.31	6.01	3.69	1.5	9.31
Ciclo 5°-4ª fase	8.78	5.54	4.27	2.65	8.78	5.54	4.27	2.65

Tabla 5.20: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	13	15.39	23.31	14.33	18.65
Semáforos 2-6	13	19.33	15.6	18.41	14.43
Semáforos 3-7	16.23	16.4	13.89	27.28	13.8
Semáforos 4-8	18.11	18.79	18.19	15.77	15.38

Tabla 5.21: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 30 sg. Peor cola \cong 12 autos. Tiempo computacional 1,7 sg.

$$Para F_5 = J_5 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados indicados en las tablas 5.22 y 5.23

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	10.6	15.24	16.28	16.7	16.35
Semáforos 2-6	14.12	15.77	15.64	16.63	14.28
Semáforos 3-7	10.46	15.04	17.17	20	18.71
Semáforos 4-8	19.68	18.9	19.28	15.13	15

Tabla 5.22: Duración de la luz verde en cada ciclo

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	1.27	1.9	1.59	0.24	1.27	1.9	1.59
Ciclo 1°-2ª fase	2.78	0.06	4.45	3.7	2.78	0.06	4.45	3.7
Ciclo 1°-3ª fase	4.66	1.31	0.24	5.27	4.66	1.31	0.24	5.27
Ciclo 1°-4ª fase	8.2	3.67	3.78	0.15	8.2	3.67	3.78	0.15
Ciclo 2°-1ª fase	0.24	5.5	6.52	2.43	0.24	5.5	6.52	2.43
Ciclo 2°-2ª fase	3.08	0.06	9.36	4.8	3.08	0.06	9.36	4.8
Ciclo 2°-3ª fase	5.78	1.86	0.93	7.05	5.78	1.86	0.93	7.05
Ciclo 2°-4ª fase	9.19	4.13	4.34	0.15	9.19	4.13	4.34	0.15
Ciclo 3°-1ª fase	0.24	6.08	7.27	2.59	0.24	6.08	7.27	2.59
Ciclo 3°-2ª fase	3.05	0.07	10.08	4.94	3.05	0.07	10.08	4.94
Ciclo 3°-3ª fase	6.14	2.13	0.24	7.51	6.14	2.13	0.24	7.51
Ciclo 3°-4ª fase	9.61	4.45	3.7	0.15	9.61	4.45	3.7	0.15
Ciclo 4°-1ª fase	0.24	6.45	6.71	2.65	0.24	6.45	6.71	2.65
Ciclo 4°-2ª fase	3.23	0.06	9.71	5.15	3.23	0.06	9.71	5.15
Ciclo 4°-3ª fase	6.82	2.45	0.24	8.14	6.82	2.45	0.24	8.14
Ciclo 4°-4ª fase	9.55	4.27	2.96	1.62	9.55	4.27	2.96	1.62
Ciclo 5°-1ª fase	0.24	6.23	5.9	4.07	0.24	6.23	5.9	4.07
Ciclo 5°-2ª fase	2.8	0.88	8.47	6.21	2.8	0.88	8.47	6.21
Ciclo 5°-3ª fase	6.17	3.12	0.24	9	6.17	3.12	0.24	9
Ciclo 5°-4ª fase	8.89	4.93	2.95	2.53	8.89	4.93	2.95	2.53

Tabla 5.23: Promedio de vehículos en el momento del cambio de luz

Valor de la función objetivo = 26,6 sg. Peor cola \cong 10 autos. Tiempo computacional 2,1 sg.

Para $J = J_6$ obtuvimos los resultados indicados en las tablas 5.24 y 5.25

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	1.32	2	1.66	0.24	1.32	2	1.66
Ciclo 1°-2ª fase	3.49	0.06	5.25	4.37	3.49	0.06	5.25	4.37
Ciclo 1°-3ª fase	5.29	1.26	0.45	5.87	5.29	1.26	0.45	5.87
Ciclo 1°-4ª fase	7.61	2.8	2.77	0.6	7.61	2.8	2.77	0.6
Ciclo 2°-1ª fase	2.32	4	4.69	2.19	2.32	4	4.69	2.19
Ciclo 2°-2ª fase	4.14	0.72	6.52	3.71	4.14	0.72	6.52	3.71
Ciclo 2°-3ª fase	7.5	3	0.24	6.5	7.5	3	0.24	6.5
Ciclo 2°-4ª fase	9.72	4.44	2.45	1.53	9.72	4.44	2.45	1.53
Ciclo 3°-1ª fase	2	6.11	5	3.62	2	6.11	5	3.62
Ciclo 3°-2ª fase	4.22	1.9	7.11	5.41	4.22	1.9	7.11	5.41
Ciclo 3°-3ª fase	6.89	3.68	0.24	7.64	6.89	3.68	0.24	7.64
Ciclo 3°-4ª fase	9.4	5.35	2.75	1.76	9.4	5.35	2.75	1.76
Ciclo 4°-1ª fase	2.83	6.85	5	3.63	2.83	6.85	5	3.63
Ciclo 4°-2ª fase	5.14	2.19	7.3	5.56	5.14	2.19	7.3	5.56
Ciclo 4°-3ª fase	8.24	4.26	0.24	8.14	8.24	4.26	0.24	8.14
Ciclo 4°-4ª fase	10.92	6.04	2.92	1.75	10.92	6.04	2.92	1.75
Ciclo 5°-1ª fase	2.71	7.8	5.57	4	2.71	7.8	5.57	4
Ciclo 5°-2ª fase	5.3	2.4	8.16	6.12	5.3	2.4	8.16	6.12
Ciclo 5°-3ª fase	8	4.2	0.24	8.38	8	4.2	0.24	8.38
Ciclo 5°-4ª fase	11.58	6.58	3.81	0.15	11.58	6.58	3.81	0.15

Tabla 5.24: Promedio de vehículos en el momento del cambio de luz

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	11	10.68	13.94	12.45	14.74
Semáforos 2-6	18.1	10.13	11.9	12.82	14.4
Semáforos 3-7	10	18.65	14.83	17.21	15
Semáforos 4-8	12.88	12.3	14	14.89	19.83

Tabla 5.25: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 316. Peor cola \cong 11,58 autos. Tiempo computacional 5,2 sg.

$$\text{Para } F_6 = J_6 + c \left[\sum_{i=1}^{4N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{4N} (\rho_i^{min} - r_2)^2 \right]$$

obtuvimos los resultados indicados en las tablas 5.26 y 5.27

	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 5
Semáforos 1-5	17.68	11.82	13.45	13.36	13.76
Semáforos 2-6	10	12	14.1	13.39	13.96
Semáforos 3-7	10	13.1	13.39	14	13.42
Semáforos 4-8	15.04	13.77	14.45	12.65	15

Tabla 5.26: Duración de la luz verde en cada ciclo

Valor de la función objetivo = 228. Peor cola \cong 7,9 autos. Tiempo computacional 6,3 sg.

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0.24	2.12	3.18	2.65	0.24	2.12	3.18	2.65
Ciclo 1°-2ª fase	2.04	0.06	5	4.15	2.04	0.06	5	4.15
Ciclo 1°-3ª fase	3.84	1.26	0.24	5.65	3.84	1.26	0.24	5.65
Ciclo 1°-4ª fase	6.54	3.06	2.94	0.15	6.54	3.06	2.94	0.15
Ciclo 2°-1ª fase	0.43	4.48	5.07	1.92	0.43	4.48	5.07	1.92
Ciclo 2°-2ª fase	2.6	0.2	7.24	3.73	2.6	0.2	7.24	3.73
Ciclo 2°-3ª fase	4.96	1.77	0.24	5.69	4.96	1.77	0.24	5.69
Ciclo 2°-4ª fase	7.44	3.42	2.72	0.15	7.44	3.42	2.72	0.15
Ciclo 3°-1ª fase	0.24	5	5.14	2.17	0.24	5	5.14	2.17
Ciclo 3°-2ª fase	2.77	0.06	7.68	4.28	2.77	0.06	7.68	4.28
Ciclo 3°-3ª fase	5.19	1.66	0.43	6.29	5.19	1.66	0.43	6.29
Ciclo 3°-4ª fase	7.79	3.4	3.04	0.15	7.79	3.4	3.04	0.15
Ciclo 4°-1ª fase	0.56	5	5.44	2.15	0.56	5	5.44	2.15
Ciclo 4°-2ª fase	3	0.07	7.85	4.16	3	0.07	7.85	4.16
Ciclo 4°-3ª fase	5.49	1.75	0.24	6.26	5.49	1.75	0.24	6.26
Ciclo 4°-4ª fase	7.77	3.27	2.51	1.1	7.77	3.27	2.51	1.1
Ciclo 5°-1ª fase	0.26	4.92	5	3.16	0.26	4.92	5	3.16
Ciclo 5°-2ª fase	2.78	0.06	7.5	5.26	2.78	0.06	7.5	5.26
Ciclo 5°-3ª fase	5.19	1.67	0.24	7.27	5.19	1.67	0.24	7.27
Ciclo 5°-4ª fase	7.9	3.47	2.94	0.82	7.89	3.47	2.94	0.82

Tabla 5.27: Promedio de vehículos en el momento del cambio de luz

En la tabla 5.28 se comparan los resultados obtenidos para las diferentes funciones objetivo (con los modelos estático y dinámico), indicando además la longitud de la cola más larga y el tiempo computacional en cada caso.

Se realizaron varias corridas de los programas para los modelos estático y dinámico, y se observó que, para todas las funciones objetivo, los mejores resultados para el modelo estático (en cuanto a valor de la función objetivo y longitud de la cola más larga) son peores que los peores resultados (en cuanto a valor de la función objetivo y longitud de la cola más larga) para el modelo dinámico, tal y como se puede apreciar en la tabla 5.28. El tiempo computacional es sensiblemente superior para el modelo dinámico que para el estático. Por tanto, a la vista de estos resultados obtenidos con tasas teóricas de llegada y salida de vehículos en el cruce, podemos concluir que, sería muy conveniente la instalación de sensores en los semáforos para facilitar la ordenación del tráfico. Ahora bien, corresponde a las Autoridades Públicas basarse en esta investigación para *tomar decisiones* en lo tocante a una buena regulación del tráfico.

	Valor función objetivo	Longitud peor cola	Tiempo computacional
J_1	33.9 autos	10 autos	1.2 sg.
F_1	29.2 autos	8.6 autos	1.9 sg.
J_2	6.03 autos	12.38 autos	0.7 sg.
F_2	5.23 autos	11.4 autos	0.8 sg.
J_3	11 autos	11 autos	1.1 sg.
F_3	10.65 autos	10.65 autos	1.25 sg.
J_4	193.18 sg.	10.23 autos	2.95 sg.
F_4	185 sg.	8.7 autos	3.4 sg.
J_5	30 sg.	12 autos	1.7 sg.
F_5	26.6 sg.	10 autos	2.1 sg.
J_6	316	11.58 autos	5.2 sg.
F_6	228	7.9 autos	6.3 sg.

Tabla 5.28: Tabla comparativa. Resultados modelos estático y dinámico

Capítulo 6

Modelos para cruce de seis fases

6.1. Modelos estático y dinámico para seis fases con ocho semáforos

6.1.1. Modelo estático

En esta sección presentamos un modelo matemático para el problema de control de tráfico en un cruce. Se tiene una intersección a la que confluyen dos calles, ambas doble vía, (con dos carriles en cada sentido de circulación), en las que se puede circular de frente, girar a derecha e izquierda. En cada carril, antes de rebasar el cruce hay un semáforo, por tanto es un cruce con ocho semáforos. Los semáforos T_1 , T_3 , T_5 y T_7 permiten seguir de frente o girar a la derecha, y los semáforos T_2 , T_4 , T_6 y T_8 sólo permiten girar a la izquierda. Se quiere hallar los períodos en que deben permanecer en verde o rojo las luces de los semáforos T_i , $i = 1, \dots, 8$ para evitar la congestión de las vías, descrita por el número de vehículos en espera (longitud de la cola) en cada vía. Para ello se consideran distintos criterios: la suma de las longitudes medias de las colas de cada vía, longitud de la cola más larga, tiempo medio de espera o una combinación de ellas. Con este planteamiento hemos de modelar y resolver un problema que en cada ciclo tiene seis fases: ver figuras 6.1, 6.2, 6.3, 6.4, 6.5 y 6.6.

- En la **primera fase** los semáforos T_1 y T_2 están en verde (los demás semáforos están en rojo), por tanto los vehículos que circulan por el carril L_1 rebasan el cruce siguiendo de frente, o giran a la derecha y los que circulan por L_2 rebasan el cruce girando a la izquierda.

- En la **segunda fase** el semáforo T_1 continúa en verde (T_2 ha cambiado de verde a rojo) y T_3 cambia de rojo a verde, por tanto continúan rebasando el cruce siguiendo de frente o continúan girando a la derecha los vehículos que circulan por el carril L_1 y rebasan el cruce continuando de frente o giran a la derecha los que circulan por L_3 .
- En la **tercera fase** el semáforo T_3 continúa en verde (T_1 ha cambiado de verde a rojo) y T_4 cambia de rojo a verde, por tanto continúan rebasando el cruce siguiendo de frente o continúan girando a la derecha los vehículos que circulan por el carril L_3 y los que circulan por L_4 rebasan el cruce girando a la izquierda.
- En la **cuarta fase** los semáforos T_5 y T_6 cambian de rojo a verde (T_3 y T_4 cambian de verde a rojo), por tanto rebasan el cruce siguiendo de frente, o giran a la derecha los vehículos que circulan por el carril L_5 y los que circulan por L_6 rebasan el cruce girando a la izquierda.
- En la **quinta fase** el semáforo T_5 continúa en verde (T_6 ha cambiado de verde a rojo) y T_7 cambia de rojo a verde, por tanto continúan rebasando el cruce siguiendo de frente o continúan girando a la derecha, los vehículos que circulan por el carril L_5 y rebasan el cruce siguiendo de frente o giran a la derecha los que circulan por L_7 .
- En la **sexta fase** el semáforo T_7 continúa en verde (T_5 ha cambiado de verde a rojo) y T_8 cambia de rojo a verde, por tanto continúan rebasando el cruce siguiendo de frente o continúan girando a la derecha los vehículos que circulan por el carril L_7 y los que circulan por L_8 rebasan el cruce girando a la izquierda.

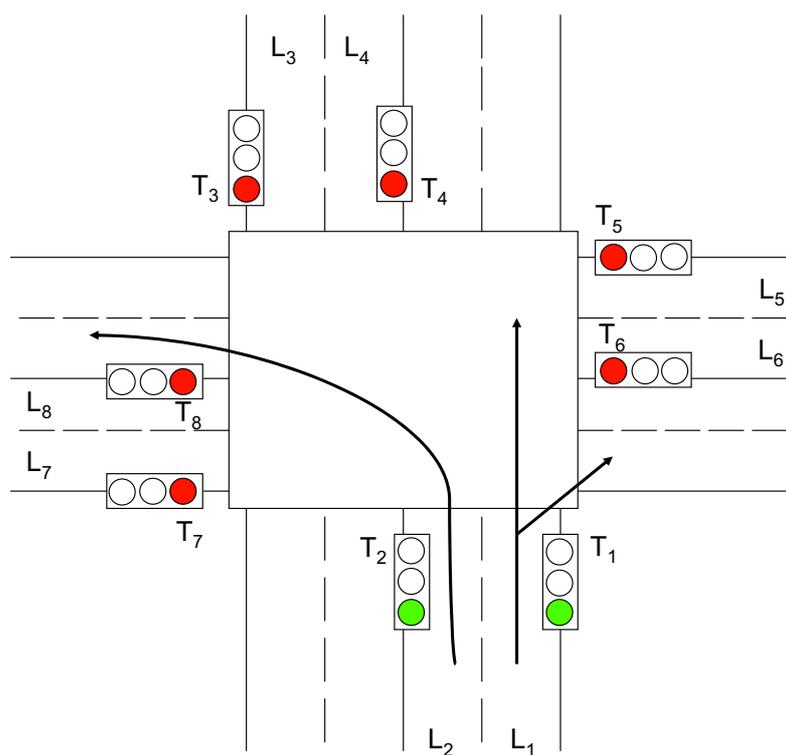


Figura 6.1: 1ª fase

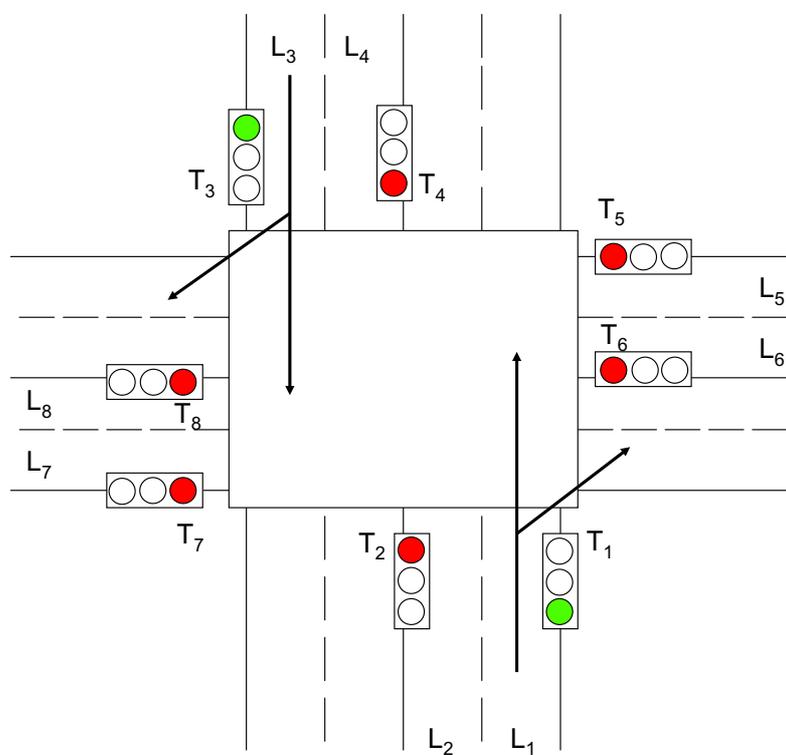


Figura 6.2: 2ª fase

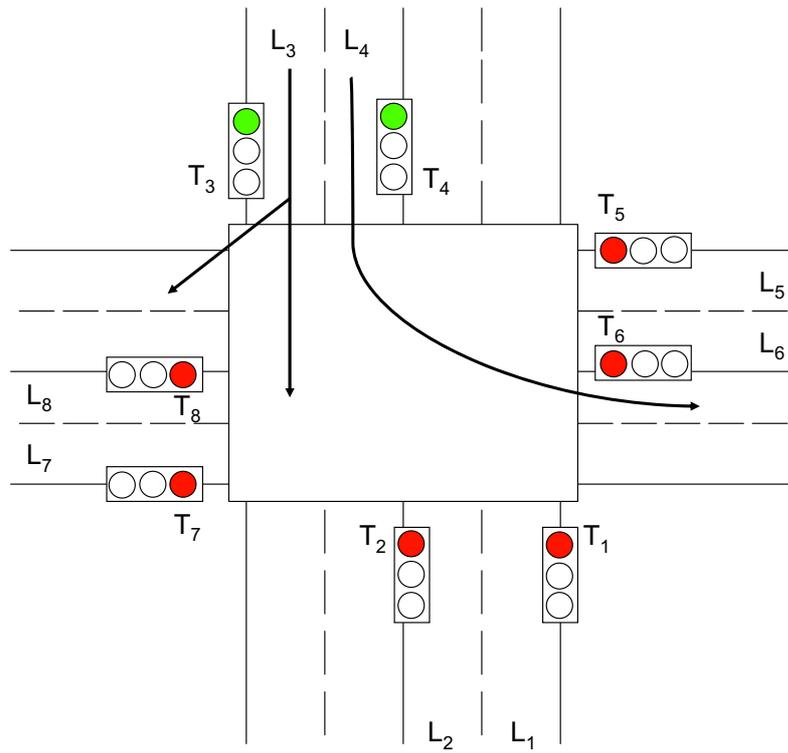


Figura 6.3: 3ª fase

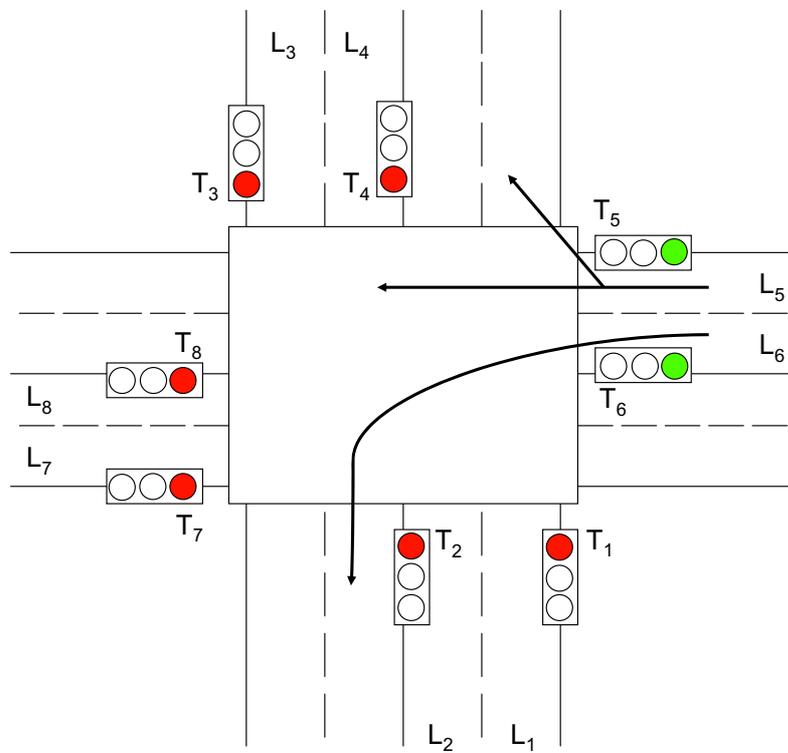


Figura 6.4: 4ª fase

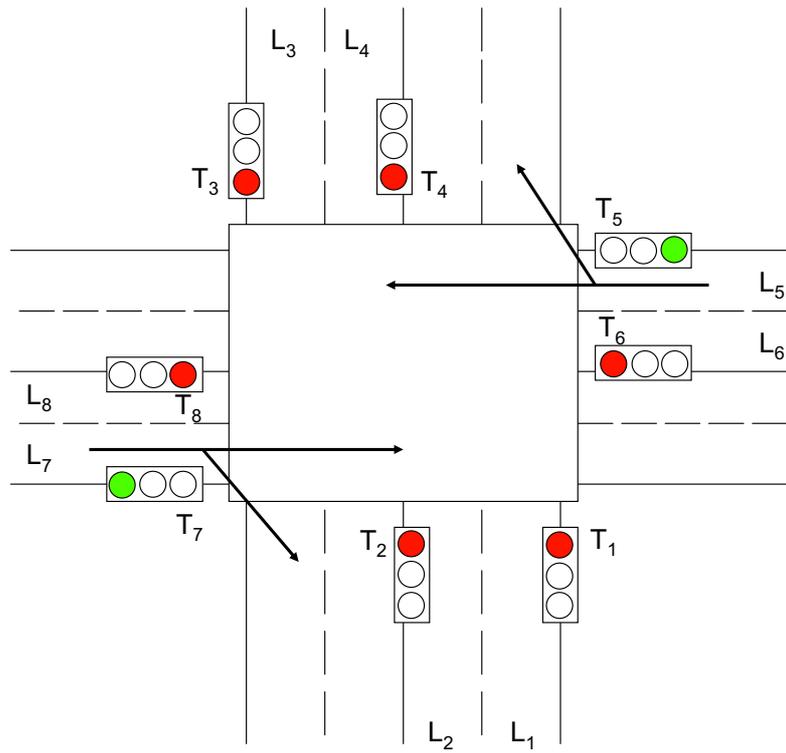


Figura 6.5: 5ª fase

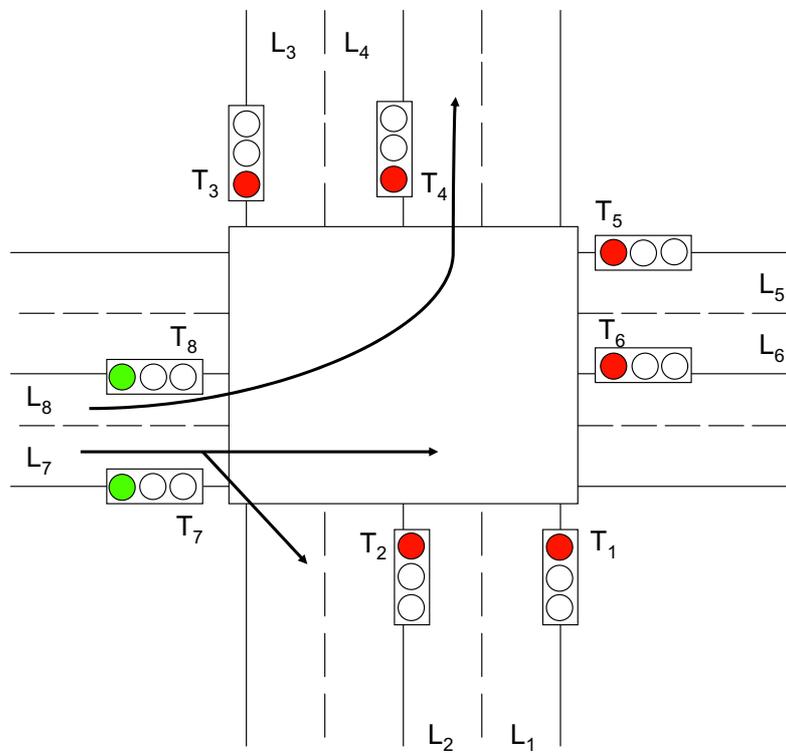


Figura 6.6: 6ª fase

Consideramos dado el número de veces que los semáforos tienen la luz verde y lo denotamos por N . También se considera pre-establecido δ_{amb} que representa el tiempo de duración de la luz ámbar (en la práctica se considera una constante que fluctúa entre los 3 y los 4 segundos).

Describimos el problema *con las variables de control* δ_k , tiempo de duración de la luz verde en el k -ésimo cambio de luz (incluyendo el ámbar) y las *variables de estado* $x_k = (x_{k1}, x_{k2}, x_{k3}, x_{k4}, x_{k5}, x_{k6}, x_{k7}, x_{k8})^t$, cantidad promedio de vehículos en los carriles L_i , $i = 1, \dots, 8$ en el momento del k -ésimo cambio de luz.

Como simplificaciones del modelo trabajaremos con el comportamiento promedio por lo que asumimos que la cantidad de autos es continua. Para cada carril L_i se definen las tasas medias de llegada y salida de los vehículos bajo las luces verde y ámbar y se denotan:

λ_i : tasa media de llegada de vehículos en el carril L_i (dada en vehículos por segundo).

μ_i : tasa media de salida en el carril L_i cuando el semáforo está en verde.

κ_i : tasa media de salida en el carril L_i cuando el semáforo está en ámbar.

Los vectores b_i , $i=1, \dots, 18$ representan el número de vehículos en cada carril teniendo en cuenta los movimientos asociados a cada fase.

$$b_1 = \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 - \mu_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix}, b_2 = \begin{bmatrix} \lambda_1 - \mu_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix}, b_3 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 - \mu_3 \\ \lambda_4 - \mu_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix}, b_4 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 - \mu_5 \\ \lambda_6 - \mu_6 \\ \lambda_7 \\ \lambda_8 \end{bmatrix},$$

$$\begin{aligned}
b_5 = & \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 - \mu_5 \\ \lambda_6 \\ \lambda_7 - \mu_7 \\ \lambda_8 \end{bmatrix}, b_6 = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 - \mu_7 \\ \lambda_8 - \mu_8 \end{bmatrix}, b_7 = \begin{bmatrix} 0 \\ (\mu_2 - \kappa_2)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_8 = \begin{bmatrix} (\mu_1 - \kappa_1)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
b_9 = & \begin{bmatrix} 0 \\ 0 \\ (\mu_3 - \kappa_3)\delta_{amb} \\ (\mu_4 - \kappa_4)\delta_{amb} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{10} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ (\mu_6 - \kappa_6)\delta_{amb} \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{11} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ (\mu_5 - \kappa_5)\delta_{amb} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
b_{12} = & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ (\mu_7 - \kappa_7)\delta_{amb} \\ (\mu_8 - \kappa_8)\delta_{amb} \end{bmatrix}, b_{13} = \begin{bmatrix} 0 \\ \max((\lambda_2 - \kappa_2)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{14} = \begin{bmatrix} \max((\lambda_1 - \kappa_1)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
b_{15} = & \begin{bmatrix} 0 \\ 0 \\ \max((\lambda_3 - \kappa_3)\delta_{amb}, 0) \\ \max((\lambda_4 - \kappa_4)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{16} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \max((\lambda_6 - \kappa_6)\delta_{amb}, 0) \\ 0 \\ 0 \end{bmatrix},
\end{aligned}$$

$$b_{17} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \max((\lambda_5 - \kappa_5)\delta_{amb}, 0) \\ 0 \\ 0 \\ 0 \end{bmatrix}, b_{18} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \max((\lambda_7 - \kappa_7)\delta_{amb}, 0) \\ \max((\lambda_8 - \kappa_8)\delta_{amb}, 0) \end{bmatrix}.$$

Entonces:

$$\begin{aligned} x_{6k+1} &= \max(x_{6k} + b_1\delta_{6k+1} + b_7, b_{13}) && \text{para } k = 0, 1, \dots, N-1, \\ x_{6k+2} &= \max(x_{6k+1} + b_2\delta_{6k+2} + b_8, b_{14}) && \text{para } k = 0, 1, \dots, N-1, \\ x_{6k+3} &= \max(x_{6k+2} + b_3\delta_{6k+3} + b_9, b_{15}) && \text{para } k = 0, 1, \dots, N-1, \\ x_{6k+4} &= \max(x_{6k+3} + b_4\delta_{6k+4} + b_{10}, b_{16}) && \text{para } k = 0, 1, \dots, N-1, \\ x_{6k+5} &= \max(x_{6k+4} + b_5\delta_{6k+5} + b_{11}, b_{17}) && \text{para } k = 0, 1, \dots, N-1, \\ x_{6k+6} &= \max(x_{6k+5} + b_6\delta_{6k+6} + b_{12}, b_{18}) && \text{para } k = 0, 1, \dots, N-1, \end{aligned}$$

Las características de la zona de tráfico puede imponer otras regulaciones, tales como establecer cotas a las duraciones mínimas y máximas para los tiempos verde y rojo, así:

$\delta_{min.verde,i}$: cota mínima de duración de la luz en semáforo $i = 1 \dots, 6$.

$\delta_{max.verde,i}$: cota máxima de duración de la luz en semáforo $i = 1, \dots, 6$.

El problema de regulación de los semáforos puede representarse a través del siguiente modelo:

$$\text{Minimizar } J \tag{6.1}$$

sujeto a:

$$\delta_{min.verde,1} \leq \delta_{6k+1} - \delta_{amb} \leq \delta_{max.verde,1} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.2}$$

$$\delta_{min.verde,2} \leq \delta_{6k+2} - \delta_{amb} \leq \delta_{max.verde,2} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.3}$$

$$\delta_{min.verde,3} \leq \delta_{6k+3} - \delta_{amb} \leq \delta_{max.verde,3} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.4}$$

$$\delta_{min.verde,4} \leq \delta_{6k+4} - \delta_{amb} \leq \delta_{max.verde,4} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.5}$$

$$\delta_{min.verde,5} \leq \delta_{6k+5} - \delta_{amb} \leq \delta_{max.verde,5} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.6}$$

$$\delta_{min.verde,6} \leq \delta_{6k+6} - \delta_{amb} \leq \delta_{max.verde,6} \quad \text{para } k = 0, 1, \dots, N-1, \tag{6.7}$$

$$x_{6k+1} = \max(x_{6k} + b_1\delta_{6k+1} + b_7, b_{13}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.8)$$

$$x_{6k+2} = \max(x_{6k+1} + b_2\delta_{6k+2} + b_8, b_{14}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.9)$$

$$x_{6k+3} = \max(x_{6k+2} + b_3\delta_{6k+3} + b_9, b_{15}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.10)$$

$$x_{6k+4} = \max(x_{6k+3} + b_4\delta_{6k+4} + b_{10}, b_{16}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.11)$$

$$x_{6k+5} = \max(x_{6k+4} + b_5\delta_{6k+5} + b_{11}, b_{17}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.12)$$

$$x_{6k+6} = \max(x_{6k+5} + b_6\delta_{6k+6} + b_{12}, b_{18}) \quad \text{para } k = 0, 1, \dots, N-1, \quad (6.13)$$

Si los carriles difieren en importancia al evaluar el nivel de congestión de los mismos, se representa el nivel de importancia del carril, fijando diferentes pesos o ponderaciones (w_1, \dots, w_8). La función J a minimizar, puede ser:

- **Suma (ponderada) de las longitudes medias de las colas en todos los carriles**

$$J_1 = \sum_{j=1}^8 w_j \frac{\sum_{i=1}^{6N} x_{ij} \delta_i}{\sum_{i=1}^{6N} \delta_i} \quad (6.14)$$

- **Longitud (ponderada) media de las colas en el carril con mayores colas**

$$J_2 = \max_j w_j \frac{\sum_{i=1}^{6N} x_{ij} \delta_i}{\sum_{i=1}^{6N} \delta_i} \quad (6.15)$$

- **Longitud (ponderada) de la cola más larga**

$$J_3 = \max_{i,j} w_j x_{ij} \quad (6.16)$$

- **Suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles**

$$J_4 = \sum_{j=1}^8 \frac{w_j}{\lambda_j} \frac{\sum_{i=1}^{6N} x_{ij} \delta_i}{\sum_{i=1}^{6N} \delta_i} \quad (6.17)$$

- **Tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera**

$$J_5 = \max_j \frac{w_j \sum_{i=1}^{6N} x_{ij} \delta_i}{\lambda_j \sum_{i=1}^{6N} \delta_i} \quad (6.18)$$

Se puede analizar una sexta opción que consiste en considerar una combinación de los criterios anteriores. Esto nos lleva a un modelo multi-objetivo que podemos resolver tomando una combinación positiva de las distintas funciones objetivo.

$$J_6 = \sum_{i=1}^5 \alpha_i J_i \quad (6.19)$$

donde los valores α_i corresponden a los pesos que se les dan a los distintos usuarios.

Como consecuencia de lo anteriormente expuesto, conocido δ (vector de los tiempos de cada luz), la cantidad de autos en cada carril quede determinado mediante las ecuaciones (6.8)-(6.9)-(6.10)-(6.11)-(6.12)-(6.13). Denotamos por $x(\delta)$ la matriz de $6N$ filas y 8 columnas donde $x_{ij}(\delta)$ indica la cantidad de autos en el carril j en el momento del cambio de luz i .

Sean los parámetros:

$$\Delta_{ij} = \begin{cases} 1 & \text{si} \\ 0 & \text{si no} \end{cases} \begin{cases} i \equiv 1(6) & j = 1, 2 \\ i \equiv 2(6) & j = 1, 3 \\ i \equiv 3(6) & j = 3, 4 \\ i \equiv 4(6) & j = 5, 6 \\ i \equiv 5(6) & j = 5, 7 \\ i \equiv 0(6) & j = 7, 8 \end{cases} \quad \tilde{\Delta}_{ij} = \begin{cases} 1 & \text{si} \\ 0 & \text{si no} \end{cases} \begin{cases} i \equiv 1(6) & j = 2 \\ i \equiv 2(6) & j = 1 \\ i \equiv 3(6) & j = 3, 4 \\ i \equiv 4(6) & j = 6 \\ i \equiv 5(6) & j = 5 \\ i \equiv 0(6) & j = 7, 8 \end{cases}$$

Una forma más compacta de escribir el problema, sin tener que diferenciar entre las seis fases es la siguiente:

$$\text{Minimizar } J \quad (6.20)$$

sujeto a:

$$\begin{aligned} & \Delta_{k1} \delta_{min.verde,1} + \Delta_{k2} \delta_{min.verde,2} + \Delta_{k3} \delta_{min.verde,3} + \Delta_{k4} \delta_{min.verde,4} + \Delta_{k5} \delta_{min.verde,5} + \\ & + \Delta_{k6} \delta_{min.verde,6} \leq \delta_k - \delta_{amb} \leq \Delta_{k1} \delta_{max.verde,1} + \Delta_{k2} \delta_{max.verde,2} + \Delta_{k3} \delta_{max.verde,3} + \\ & + \Delta_{k4} \delta_{max.verde,4} + \Delta_{k5} \delta_{max.verde,5} + \Delta_{k6} \delta_{max.verde,6} \quad \text{para } k = 1, \dots, 6N \end{aligned} \quad (6.21)$$

$$x_{k,j} = \max[x_{k-1,j} + (\lambda_j - \Delta_{kj}\mu_j)\delta_k + \tilde{\Delta}_{kj}(\mu_j - \kappa_j)\delta_{amb}, \max(\tilde{\Delta}_{kj}(\lambda_j - \kappa_j)\delta_{amb}, 0)] \quad (6.22)$$

para $k = 1, \dots, 6N$; $j = 1, \dots, 8$

Como habíamos mencionado anteriormente, ya introducimos una aproximación extra al considerar las longitudes de cola continuas. Además, en la práctica, hay también alguna incertidumbre y variación en el tiempo de las tasas de llegada y de salida, lo que hace recomendable obtener una buena aproximación de la solución óptima que sea calculable en un tiempo corto y fácilmente ajustable a los cambios en los valores de los parámetros. De hecho, utilizando un razonamiento análogo al empleado en la sección 3.1.3.1, llegamos a que estamos ante un problema de optimización con restricciones de complementariedad lineal (problema NP-duro), que involucra a $54N$ variables, $12N$ restricciones suaves y $48N$ no suaves, lo que hace que sea muy costoso aplicar un algoritmo iterativo [19]. Es por eso que para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves, como el descrito en la sección 3.2.4.

6.1.2. Modelo dinámico

En este modelo consideramos que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz. Esta decisión permite menores tiempos de espera, pues elimina situaciones indeseadas como sería que no hubiera autos en el carril con la luz verde mientras otros carriles esperan.

Para la estrategia de cambio consideramos que si hay al menos r_2 vehículos esperando en carriles con luz roja, entonces es recomendable cambiar, así como si hay a lo más r_1 autos en los carriles a los que corresponde la luz verde. Dado que puede ser muy restrictivo asumir que los cambios de luces ocurren en estas situaciones, suavizamos el modelo planteando que se minimiza el error cuadrático de aproximar la cantidad de autos en los momentos de cambio en los carriles en espera por r_2 y en los carriles de servicio por r_1 .

O sea si la cantidad de autos en el carril al que corresponde la luz verde en el momento i es $\sum_j \Delta_{ij}x_{ij}$, debemos agregar a la función objetivo el cuadrado de la diferencia entre este valor y r_1 , pero solo si $\sum_j \Delta_{ij}x_{ij} > r_1$. Agregando la variable auxiliar ρ_i^{max} y la restricción $\sum_j \Delta_{ij}x_{ij} \leq \rho_i^{max}$ entonces al minimizar $(\rho_i^{max} - r_1^2)$ se cumple lo deseado. Realizando un razonamiento análogo para los autos que esperan por el cambio

de luz para pasar y con la ayuda de la variable ρ_i^{min} , la penalidad quedaría

$$\sum_{i=1}^{6N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{6N} (\rho_i^{min} - r_2)^2$$

donde

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \text{ para } i = 1, \dots, 6N \quad \sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \text{ para } i = 1, \dots, 6N$$

Usando un parámetro c para controlar el peso que se le da a esta nueva función en el objeto a minimizar, el modelo final es:

$$\text{Minimizar } F = J + c \left[\sum_{i=1}^{6N} (\rho_i^{max} - r_1)^2 + \sum_{i=1}^{6N} (\rho_i^{min} - r_2)^2 \right] \quad (6.23)$$

sujeto a:

$$\begin{aligned} & \Delta_{k1} \delta_{min.verde,1} + \Delta_{k2} \delta_{min.verde,2} + \Delta_{k3} \delta_{min.verde,3} + \Delta_{k4} \delta_{min.verde,4} + \Delta_{k5} \delta_{min.verde,5} + \\ & + \Delta_{k6} \delta_{min.verde,6} \leq \delta_k - \delta_{amb} \leq \Delta_{k1} \delta_{max.verde,1} + \Delta_{k2} \delta_{max.verde,2} + \Delta_{k3} \delta_{max.verde,3} + \\ & + \Delta_{k4} \delta_{max.verde,4} + \Delta_{k5} \delta_{max.verde,5} + \Delta_{k6} \delta_{max.verde,6} \quad \text{para } k = 1, \dots, 6N \end{aligned} \quad (6.24)$$

$$x_{k,j} = \max[x_{k-1,j} + (\lambda_j - \Delta_{kj} \mu_j) \delta_k + \tilde{\Delta}_{kj} (\mu_j - \kappa_j) \delta_{amb}, \max(\tilde{\Delta}_{kj} (\lambda_j - \kappa_j) \delta_{amb}, 0)] \quad (6.25)$$

para $k = 1, \dots, 6N$; $j = 1, \dots, 8$

$$\sum_j (1 - \Delta_{ij}) x_{ij} \geq \rho_i^{min} \text{ para } i = 1, \dots, 6N \quad (6.26)$$

$$\sum_j \Delta_{ij} x_{ij} \leq \rho_i^{max} \text{ para } i = 1, \dots, 6N \quad (6.27)$$

Para su solución proponemos un método híbrido que combina una heurística, tipo recocido simulado, con un algoritmo quasi-Newton para problemas de optimización no suaves, como el descrito en la sección 3.2.4.

6.1.3. Resultados computacionales

Con el objetivo de verificar la validez del algoritmo propuesto en esta tesis, se programan en MATLAB y se implementan en un ordenador con procesador Intel Core i7, 950 que trabaja a 307 GHz., el algoritmo del método quasi-Newton (método del gradiente discreto), el del método heurístico basado en la metaheurística de recocido simulado, el

híbrido que combina el método heurístico y luego el quasi-Newton (que es el propuesto) y el híbrido que combina el método quasi-Newton y luego el método heurístico. Todos ellos se corren partiendo de los datos que a continuación se especifican, obteniéndose los resultados para las funciones objetivo de los modelos estático y dinámico.

Datos: 5 ciclos ($N=5$), tiempo de ámbar 3 sg. ($\delta_{amb}=3$ sg.).

Se usaron los parámetros que se indican en la tabla 6.1

Carriles	1	2	3	4	5	6	7	8
Tasa-llegada (λ)	0.1	0.15	0.2	0.15	0.2	0.1	0.15	0.2
Tasa-salida verde (μ)	0.7	0.7	2	1.5	0.7	0.7	1	1.5
Tasa-salida ámbar (κ)	0.25	0.25	0.45	0.2	0.25	0.25	0.45	0.2

Tabla 6.1: Parámetros

En cuanto a las cotas del tiempo de las luces, se tomaron:

$$\delta_{min.verde,i} = 5, \forall i = 1, \dots, 6, \quad \delta_{max.verde,i} = 30, \forall i = 1, \dots, 6$$

Solución inicial: δ_0 aleatorio

Construcción de las funciones objetivo

Pesos $w = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$

$\alpha_i = 1, \forall i = 1, \dots, 5$ (construcción de la función objetivo J_6)

$r_1 = 8, r_2 = 32$ (construcción de las funciones objetivo del modelo dinámico)

Elementos para la construcción de la heurística

Temperatura inicial $t_0 = 100000000$

Función de reducción de temperatura $T_M = 0.001 T_{M-1}$

Número de iteraciones con igual temperatura $Q = 20$

Criterio de parada $T < 0.0001$

Los resultados obtenidos aparecen en las tablas 6.2 y 6.3

Función objetivo	QuasiNewton	Heurístico	Heurístico+Quasi	Quasi+Heurístico
J_1	51.26 autos	55.72 autos	48.49 autos	50 autos
F_1	45.59 autos	55 autos	45 autos	45.57 autos
J_2	12 autos	14.51 autos	10.57 autos	12 autos
F_2	9.57 autos	15.88 autos	8.67 autos	9.57 autos
J_3	22.6 autos	32.4 autos	21.2 autos	22.6 autos
F_3	20.7 autos	31.4 autos	19.45 autos	20.7 autos
J_4	258.6 sg.	404.4 sg.	194.5 sg.	257.4 sg.
F_4	255.2 sg.	376.9 sg.	249.9 sg.	250.5 sg.
J_5	44.92 sg.	108.2 sg.	40.94 sg.	44.41 sg.
F_5	36.12 sg.	83.51 sg.	34.48 sg.	35.92 sg.
J_6	439.18	658.54	374.54	438.33
F_6	329.14	485.61	298.92	326.75

Tabla 6.2: Tabla comparativa (1ª corrida). Diferentes algoritmos

Función objetivo	QuasiNewton	Heurístico	Heurístico+Quasi	Quasi+Heurístico
J_1	61.33 autos	64.09 autos	58.29 autos	61.3 autos
F_1	38.2 autos	45.38 autos	37.05 autos	38.1 autos
J_2	13.58 autos	14.56 autos	10.95 autos	13.52 autos
F_2	11.43 autos	15.24 autos	10.33 autos	11.4 autos
J_3	25.2 autos	34.8 autos	22.8 autos	25.2 autos
F_3	22.9 autos	29.85 autos	20.8 autos	22.6 autos
J_4	220.5 sg.	379.7 sg.	210.9 sg.	219.08 sg.
F_4	199.8 sg.	339.6 sg.	185.6 sg.	198.9 sg.
J_5	49.82 sg.	81.78 sg.	38.57 sg.	49.32 sg.
F_5	41.85 sg.	99.35 sg.	36.95 sg.	41.18 sg.
J_6	447.06	630.34	415.1	445.35
F_6	328.23	487.76	313.14	324.47

Tabla 6.3: Tabla comparativa (2ª corrida). Diferentes algoritmos

A la vista de los resultados que se recogen en las tablas 6.2 y 6.3 llegamos a las siguientes conclusiones:

- El algoritmo que mejor se comporta (en cuanto a la obtención de los valores mínimos de las diferentes funciones objetivo) es el algoritmo híbrido consistente en buscar una aproximación a la solución por medio de recocido simulado, y luego mejorarla a través del método del gradiente discreto.
- En segundo lugar, también se podría usar el método híbrido que busca primero una solución por el método del gradiente discreto y, partiendo de ella, usa recocido simulado, ya que su comportamiento es bastante similar al del anterior.
- En cuanto a los métodos que se usan en solitario, va mejor el método quasi-Newton que la heurística. Quizás sería necesario replantearse las condiciones que se usan para construir la heurística, en cuanto a función de reducción de temperatura, número de pasos con igual temperatura, criterio de parada, etc., para que los resultados de ésta sean más ajustados a los de los métodos híbridos, lo que además permitiría que éstos pudiesen mejorar al mejorar la heurística.

6.1.4. Aplicación al cruce de la avenida de Arteixo con la Ronda de Outeiro

La intersección semaforizada de la avenida de Arteixo con la Ronda de Outeiro de la ciudad de A Coruña, (ver figura 6.7) es un ejemplo del tipo de cruce presentado y modelado en este capítulo, o sea, de cruce regulado por semáforos con las seis fases descritas en las figuras 6.1, 6.2, 6.3, 6.4, 6.5 y 6.6 en cada ciclo.

En la primera fase los semáforos T_1 y T_2 están en verde (los demás semáforos están en rojo), por tanto los vehículos que circulan por la Ronda de Outeiro en sentido hacia el interior de la ciudad, rebasan el cruce siguiendo por dicha Ronda, o giran a la derecha o a la izquierda hacia la avenida de Arteixo.

En la segunda fase el semáforo T_1 sigue en verde (T_2 ha cambiado de verde a rojo), por tanto los vehículos continúan circulando por la Ronda de Outeiro, o continúan girando a la derecha hacia la avenida de Arteixo, y T_3 cambia de rojo a verde, permitiendo a los vehículos iniciar la marcha por la Ronda de Outeiro en sentido hacia la salida de la ciudad o girar a la derecha hacia la avenida de Arteixo.



Figura 6.7: Imagen del cruce

En la tercera fase el semáforo T_3 sigue en verde (T_1 ha cambiado de verde a rojo), por tanto los vehículos continúan circulando por la Ronda de Outeiro, o continúan girando a la derecha hacia la avenida de Arteixo y T_4 cambia de rojo a verde, por tanto se puede girar a la izquierda hacia la avenida de Arteixo.

En la cuarta fase los semáforos T_5 y T_6 cambian de rojo a verde (T_3 y T_4 cambian de verde a rojo), permitiendo a los vehículos iniciar la marcha por la avenida de Arteixo en sentido hacia la salida de la ciudad, o girar a la derecha o a la izquierda hacia la Ronda de Outeiro.

En la quinta fase el semáforo T_5 sigue en verde (T_6 ha cambiado de verde a rojo), por tanto los vehículos continúan circulando por la avenida de Arteixo o continúan girando a la derecha hacia la Ronda de Outeiro y T_7 cambia de rojo a verde, permitiendo a los vehículos iniciar la marcha por la avenida de Arteixo en sentido hacia el centro de la ciudad, o girar a la derecha hacia la Ronda de Outeiro.

En la sexta fase el semáforo T_7 sigue en verde (T_5 ha cambiado de verde a rojo), por tanto los vehículos continúan circulando por la avenida de Arteixo, o continúan girando a la derecha hacia la Ronda de Outeiro y T_8 cambia de rojo a verde, por tanto se puede girar a la izquierda hacia la Ronda de Outeiro.

Una vez descritas las seis fases que constituyen cada uno de los ciclos de los semáforos, quisiera hacer los comentarios siguientes:

- En cada sentido de circulación, antes de rebasar el cruce, hay tres carriles: el de la izquierda para girar hacia la izquierda, el del centro para seguir de frente y el de la derecha para seguir de frente o girar hacia la derecha, excepto en la Ronda de Outeiro en dirección a la salida de la ciudad que hay cuatro, uno para girar a la izquierda, otro para girar a la derecha y dos para seguir de frente.
- En los semáforos T_1 , T_3 , T_5 y T_7 hay un anexo que, en ciertos momentos se pone en verde o en ámbar intermitente, permitiendo el giro hacia la derecha. Su funcionamiento es el siguiente:
 - En el anexo del semáforo T_1 sólo se enciende la luz verde a la vez y durante el mismo tiempo que la luz verde del semáforo principal, por tanto, su existencia sólo puede justificarse por el hecho de que permite una mejor información al conductor acerca de la posibilidad de girar a la derecha.
 - En el anexo del semáforo T_3 sólo se enciende la luz ámbar intermitente en dos períodos: en el primero de ellos, durante 10 sg. (tiempo que dura encendida la luz verde del semáforo T_8), mientras que la luz del semáforo principal está en rojo y en el segundo durante el tiempo que está en verde el semáforo principal.
 - En el anexo del semáforo T_5 sólo se enciende la luz ámbar intermitente. El tiempo de duración es la suma del tiempo de luz verde del semáforo T_4 -10 sg.- (luz roja en el semáforo principal) con el tiempo de luz verde del semáforo principal.
 - En el anexo del semáforo T_7 sólo se enciende la luz ámbar intermitente durante el tiempo que está en verde el semáforo principal y continúa durante 10 sg. más (tiempo que dura encendida la luz verde del semáforo T_2), una vez que la luz del semáforo principal cambió a rojo.
- El semáforo T_7 permite a los peatones cruzar la avenida de Arteixo durante el tiempo que el anexo del semáforo T_3 está en ámbar intermitente en su segundo período, lo que dificulta durante ese tiempo el giro a la derecha de los vehículos que circulan por el carril L_3 .
- El semáforo T_3 permite a los peatones cruzar la Ronda de Outeiro durante el tiempo en el que el anexo del semáforo T_5 está en ámbar intermitente a la vez

que está en verde el semáforo principal, lo que dificulta durante ese tiempo el giro a la derecha de los vehículos que circulan por el carril L_5 .

- El semáforo T_1 permite a los peatones cruzar la Ronda de Outeiro durante el tiempo en el que el anexo del semáforo T_7 está en ámbar intermitente a la vez que está en verde el semáforo principal, lo que dificulta durante ese tiempo el giro a la derecha de los vehículos que circulan por el carril L_7 .

A la vista de las consideraciones anteriores, en nuestro modelo no hemos tenido en cuenta los anexos de los semáforos T_1 , T_3 , T_5 y T_7 ya que, el del semáforo T_1 no afecta a la circulación y los de T_3 , T_5 y T_7 , aunque si afectan, no lo hacen en una medida significativa ya que, están puestos para compensar las dificultades en los giros a la derecha provocados por el cruce de peatones. Lo que si se ha considerado es la existencia de tres o cuatro carriles que, por supuesto, afecta a las tasas de llegada y de salida de vehículos del cruce.

Los tiempos de luz verde que rigen los semáforos de la intersección son: para los semáforos T_2 , T_4 , T_6 y T_8 que permiten el giro a la izquierda es de 10 sg., para los semáforos T_1 y T_3 que permiten la circulación por la Ronda de Outeiro o giro a la derecha, es de 40 sg. y para los semáforos T_5 y T_7 que permiten la circulación por la avenida de Arteixo o giro a la derecha es de 25 sg. Por tanto la duración de las fases es: primera fase 10 sg., segunda fase 30 sg., tercera fase 10 sg., cuarta fase 10 sg., quinta fase 15 sg. y sexta fase 10 sg.

También se ha observado que, como es un cruce con mucha densidad de circulación durante todo el día y fundamentalmente en horas punta, en ciertos intervalos de tiempo, se produce una pequeña variación en la duración de las fases, ya que, los vehículos que circulan por la Ronda de Outeiro lo hacen durante 50 sg. y los que circulan por la avenida de Arteixo, durante 40 sg. Por tanto la duración del ciclo completo sería en un caso de 85 sg. y en otro de 110 sg., tiempo que está según se indica en [66] dentro del rango que se acomoda a la mentalidad del conductor.

6.1.5. Resultados computacionales

Por mediciones directas en observaciones de varios días durante diferentes momentos del día y principalmente durante el horario pico, se estimaron las tasas de llegada y salida en los carriles, las cuales se recogen en la tabla 6.4

Carriles	1	2	3	4	5	6	7	8
Tasa-llegada (λ)	0.35	0.1	0.4	0.09	0.26	0.09	0.35	0.1
Tasa-salida verde (μ)	1.05	0.7	1.1	0.6	1	0.7	1	0.6
Tasa-salida ámbar (κ)	0.25	0.25	0.45	0.2	0.25	0.25	0.45	0.2

Tabla 6.4: Parámetros

Se implementó el algoritmo híbrido heurística+quasi-Newton con las tasas anteriores en cinco ciclos (ver los resultados en la tabla 6.5) y se observó que el largo de las colas de vehículos en el cruce en horario pico, es muy similar al que se da en la realidad; colas que no son muy deseables, debido a que en los carriles L_1 , L_3 , L_5 y L_7 están entre los 20 y 25 vehículos, y en los carriles L_2 , L_4 , L_6 y L_8 entre los 8 y 10 vehículos. Ahora bien, también se observa que hay fases en las que rebasan la intersección todos los vehículos que están esperando, por tanto, podemos afirmar que la regulación hecha tanto con nuestro algoritmo como la diseñada por las Autoridades de Tráfico, es bastante adecuada, teniendo en cuenta que este cruce absorbe el tráfico de entrada y salida de la ciudad en dos direcciones diferentes, y el tráfico de entrada y salida de un polígono industrial y de una zona en donde se encuentra una de las superficies comerciales más grandes de España.

Carriles	1	2	3	4	5	6	7	8
Ciclo 1°-1ª fase	0	0	2	0.45	1.3	0.45	1.75	0.5
Ciclo 1°-2ª fase	0.3	1	0	1.35	3.9	1.35	5.25	1.5
Ciclo 1°-3ª fase	3.45	1.9	0	0	6.24	2.16	8.4	2.4
Ciclo 1°-4ª fase	5.2	2.4	2	0.45	2.54	0.46	10.15	2.9
Ciclo 1°-5ª fase	8	3.2	5.2	1.17	0.03	1.18	4.95	3.7
Ciclo 1°-6ª fase	11.15	4.1	8.8	1.98	2.37	2	0.75	0.4
Ciclo 2°-1ª fase	7.65	2.45	10.8	2.43	3.67	2.44	2.5	0.9
Ciclo 2°-2ª fase	0.3	3.85	1	3.69	7.31	3.7	7.4	2.3
Ciclo 2°-3ª fase	2.4	4.45	0	1.83	8.87	4.24	9.5	2.9
Ciclo 2°-4ª fase	5.55	5.35	3.6	2.64	2.2	0.1	12.65	3.8
Ciclo 2°-5ª fase	8.7	6.25	7.2	3.45	0.03	0.91	6.8	4.7
Ciclo 2°-6ª fase	12.9	7.45	12	4.53	3.15	2	0.65	0
Ciclo 3°-1ª fase	2.4	0	18	5.88	7.05	3.34	5.9	1.5
Ciclo 3°-2ª fase	0.3	1.7	6.1	7.4	11.47	4.87	11.85	3.2
Ciclo 3°-3ª fase	4.15	2.8	0.35	3	14.33	5.86	15.7	4.3
Ciclo 3°-4ª fase	9.75	4.4	6.75	4.4	2.49	0	21.3	5.9
Ciclo 3°-5ª fase	16	6.2	14	6	0.03	1.62	9.6	7.7
Ciclo 3°-6ª fase	22	7.9	20.75	7.59	4.45	3.15	0.2	0.4
Ciclo 4°-1ª fase	13.6	2.05	25.5	8.67	7.57	4.23	4.4	1.6
Ciclo 4°-2ª fase	2.7	4	12.25	10.38	12.5	6	11	3.5
Ciclo 4°-3ª fase	7.6	5.35	4.4	4.4	16.15	7.2	16	4.9
Ciclo 4°-4ª fase	10	6	7.2	5	11	4.28	18.4	5.6
Ciclo 4°-5ª fase	13.9	7.15	11.6	6	5	5.27	11.25	6.7
Ciclo 4°-6ª fase	17.75	8.25	16	7	8	6.26	5.75	2.4
Ciclo 5°-1ª fase	8.65	1.8	21.2	8.22	11.3	7.43	10.3	3.7
Ciclo 5°-2ª fase	0.3	3.4	10	9.6	15.48	8.87	15.9	5.3
Ciclo 5°-3ª fase	5.2	4.8	2.15	3.72	19.12	10.1	20.8	6.7
Ciclo 5°-4ª fase	8.7	5.8	6.15	4.62	11.7	5.38	24.3	7.7
Ciclo 5°-5ª fase	12.55	6.9	10.5	5.6	5.83	6.37	17.1	8.8
Ciclo 5°-6ª fase	17.1	8.2	15.7	6.78	9.2	7.54	10.35	3.5

Tabla 6.5: Promedio de vehículos en el momento del cambio de luz

Conclusiones

El objetivo de este trabajo ha sido desarrollar modelos y algoritmos solución precisos y estables para problemas de control óptimo de semáforos. Como aplicación particular hemos considerado cruces con 2, 3, 4 o 6 fases en la ciudad de A Coruña. La metodología de trabajo y los resultados obtenidos han sido los siguientes:

- Partiendo del modelo para cruce simple (2 fases) propuesto por Schutter-Moor, (en el que las restricciones del modelo constituyen un problema de complementariedad lineal extendido, NP-duro, por ello, propone un algoritmo solución de un problema aproximado basado en la relajación de éste, lo que le lleva a soluciones subóptimas), se diseñan dos modelos, uno en el cual los tiempos de las luces son variables, pero sólo dependen de los flujos de llegada (caso estático) y otro en el que se considera que en el semáforo hay sensores que permiten conocer la cantidad de autos en cada carril y, por ende, se puede decidir si, dado el tráfico, se mantiene el estado actual, por cuánto tiempo, o si debe cambiar la luz (caso dinámico). Se consideraron diferentes funciones objetivo a minimizar: suma (ponderada) de las longitudes medias de las colas en todos los carriles, longitud (ponderada) media de las colas en el carril con mayores colas, longitud (ponderada) de la cola más larga, suma (ponderada) de los tiempos medios de espera considerando las colas de todos los carriles, tiempo (ponderado) medio de espera en el carril con mayor tiempo de espera, o una combinación de todas ellas. El problema, por tanto, consiste en encontrar los instantes de cambio *switching* de las luces de los semáforos, para los que se minimiza alguna de las funciones anteriores. Se propone un método híbrido de solución donde se combina la filosofía de recocido simulado y un método quasi-Newton adecuado. Con la heurística fuimos capaces de explorar la región de soluciones *escapando* de extremos locales, y mediante el algoritmo determinístico se mejoró considerablemente la solución obtenida a través de la heurística. Hemos implementado estos algoritmos mediante programas escritos en

MATLAB. Hemos presentado ejemplos numéricos para cinco ciclos, obteniéndose que el modelo dinámico da resultados mejores, en cuanto al valor de la función objetivo y al largo de las colas, que el modelo estático y el tiempo computacional es sólo sensiblemente superior.

- Hemos diseñado un modelo para un cruce regulado por semáforos en el que hay tres fases en cada ciclo. Para comprobar su validez, se implementó el modelo diseñado y se aplicó al cruce de las calles Palomar, Paseo de los Puentes y avenida de Finisterre, de la ciudad de A Coruña, comprobándose que las colas en los distintos carriles y los tiempos medios de espera se corresponden con los que se dan en la realidad en horas pico. Seguidamente se programó en MATLAB el método heurístico aplicado a este modelo y se implementó partiendo de los datos iniciales del cruce, comprobándose la bondad del algoritmo, ya que las colas se redujeron sensiblemente, pues al ser los tiempos de las luces variables, se evita que haya coches esperando cuando en otros carriles ninguno está pasando.
- A continuación se diseñan modelos para un cruce regulado por semáforos en el que hay cuatro fases en cada ciclo con cuatro u ocho semáforos. Se aplica el modelo diseñado al cruce de la avenida de Finisterre con la Ronda de Nelle en A Coruña (4 semáforos) y, al igual que en el caso de tres fases, se comprueba su validez y se programa en MATLAB el algoritmo heurístico, con el cual se obtienen resultados algo mejores que los que se dan en la realidad con los semáforos a ciclo fijo. En el caso de ocho semáforos, con datos de un cruce teórico, se programa en MATLAB el método híbrido propuesto en este trabajo, y se aplica a los modelos estático y dinámico usando todas las funciones objetivo que se consideran. Como sucedió para el cruce simple, el comportamiento del modelo dinámico es mejor, en cuanto al valor de las funciones objetivo y al largo de las colas, con un tiempo computacional ligeramente superior.
- Para finalizar se diseña un modelo para un cruce regulado por ocho semáforos con seis fases en cada ciclo. Se programan en MATLAB para este modelo, el método quasi-Newton, el método heurístico, y dos métodos híbridos: uno que empieza en la heurística y a continuación usa el quasi-Newton y otro que empieza por este método y luego acaba usando la heurística. Se implementaron estos métodos con datos teóricos, comprobando que con el que se obtienen los mejores resultados en cuanto al valor de las funciones objetivo de los modelos estático y dinámico, y en cuanto a la longitud de las colas, es con el método que empieza en la heurística

y luego mejora la solución con el determinístico, que es el que propusimos desde el principio de este trabajo. Seguidamente este método se aplica al cruce de la Ronda de Outeiro con la avenida de Arteixo en A Coruña usando datos reales del cruce en horas pico. La conclusión es que, aunque se mejora algo respecto a lo que sucede en la realidad, los resultados son bastante similares a los que se producen en el cruce, debido a que es un cruce con frecuentes embotellamientos, pues tiene una densidad de tráfico muy alta.

- Los modelos que hemos diseñado y aplicado a la intersección regulada, impregnan a nuestra investigación de un carácter *interdisciplinario* al incorporar no sólo referencias económicas y sociales, sino también los aspectos medioambientales que pretende resolver, como la congestión, la generación de accidentes, los ruidos y la contaminación atmosférica.
- Corresponde a las Autoridades Públicas basarse en esta investigación para *tomar decisiones* en lo tocante a una buena regulación del tráfico.

En el futuro se tratará de seguir trabajando en dichas intersecciones con datos reales de flujo de vehículos, incluyendo en nuestro estudio diferencias entre los distintos carriles, un mayor número de ciclos y en el caso del problema multiobjetivo, vectores de peso que ponderen más un objetivo que otro. También se tratará de relatar y esbozar otras líneas de investigación, como por ejemplo: uso de otros métodos heurísticos: colonias de hormigas, algoritmos genéticos, etc. o extensión de la teoría propuesta al caso de las rotondas o de cruces concatenados.

Bibliografía

- [1] Reglamento (CE) del Parlamento Europeo y del Consejo. (443), 2009.
- [2] B. Adenso Díaz, F. Glover, H. Ghaziri, J.M. González, M. Laguna, P. Moscato, and F. Tseng. *Optimización heurística y redes neuronales en dirección de operaciones e ingeniería*. Editorial Paraninfo. Madrid, 1996.
- [3] S. Allende, A. Blanco, C.S. Lema, and L.P. Pedreira. Modelo de optimización con restricciones de equilibrio para el control de semáforos. *Rect@, Actas 14*, 1:44.1–44.13, 2006.
- [4] O. Arbelaitz Gallego. Soluciones basadas en simulated annealing para el VRPTW. *Tesis. Donostia*, 2002.
- [5] S. Asensio Cuesta. Metodología para la generación de agendas de rotación de puestos de trabajo desde un enfoque ergonómico mediante algoritmos evolutivos. *Tesis. Universidad Politécnica de Valencia*, 2009.
- [6] M. F. Aycin and R. F. Benekohal. Linear acceleration car-following model development and validation. *Transportation research record*, (1644), pages 10–19, 1998.
- [7] M. F. Aycin and R. F. Benekohal. Comparison of car-following models for simulation. *Transportation research record*, (1678), pages 116–127, 1999.
- [8] A.M. Bagirov. Minimization methods for one class of nonsmooth functions and calculation of semiequilibrium prices. *In: Eberhard, A. et al. (eds). Progress in Optimization: Contribution from Australasia*, pages 147–175, 1999.
- [9] A.M. Bagirov. Continuous subdifferential approximations and their applications. *Journal of Mathematical Sciences*, 115:2567–2609, 2003.

-
- [10] A.M. Bagirov, B. Karasözen, and M. Sezer. Discrete gradient method: Derivative-free method for nonsmooth optimization. *Journal of Optimization Theory and Applications*, 137(2):317–334, 2008.
- [11] J. Barcelo, J.L. Ferrer, D. Garcia, M. Florian, and E. Le Saux. The parallelization of AIMSUN2 microscopic simulator for its applications. *In Proceedings of The 3rd World Congress on Intelligent Transportation Systems*, 1996.
- [12] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear programming. Theory and algorithms*. Wiley-Interscience, third edition, 2006.
- [13] Ghulam H. Bham and Rahim F. Benekohal. A high fidelity traffic simulation model based on cellular automata and car-following concepts. *Transportation Research Part C*, 12:1–32, 2004.
- [14] V. Cerny. A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm. *Journal of Optimization, Theory and Applications*, 45:41–55, 1985.
- [15] D.T. Connolly. An improved annealing schema for the QAP. *European Journal of Operational Research*, 46:93–100, 1990.
- [16] R.W. Cottle, J.S. Pang, and R.E. Stone. *The linear complementarity problem*. Academic Press. Boston, 1992.
- [17] Comisión de las Comunidades Europeas. Comunicación de la Comisión al Consejo y al Parlamento Europeo. Hacia un transporte más ecológico, p. 2. *COM(2008) 433*, 2008.
- [18] Comisión de las Comunidades Europeas. Comunicación de la Comisión al Consejo y al Parlamento Europeo. Hacia un transporte más ecológico, p. 8. *COM(2008) 433*, 2008.
- [19] B. De Schutter. Optimizing acyclic traffic signal switching sequences through an extended linear complementarity problem formulation. *European Journal of Operational Research*, 139(2):400–415, 2002.
- [20] B. De Schutter and B. De Moor. The extended linear complementarity problem. *Mathematical Programming*, 71(3):289–325, 1995.

-
- [21] B. De Schutter and B. De Moor. Optimal traffic light control for a single intersection. *European Journal of Control*, 4(3):260–273, 1998.
- [22] A. Di Febbraro, D. Giglio, and N. Sacco. On applying Petri nets to determine optimal offsets for coordinated traffic light timings. *In The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 773–778, 2002.
- [23] K.A Dowsland. *Simulated annealing*. Modern heuristics techniques for combinatorial problems. Blackwell scientific publisher. Oxford, 1993.
- [24] K.A. Dowsland and B. Adenso Díaz. Heuristic design anf fundamentals of the simulated annealing. *Revista iberoamericana de inteligencia artificial*, 19:93–102, 2003.
- [25] Comisión Europea. Libro Blanco sobre Crecimiento, Competitividad y Empleo. Condiciones para el crecimiento y para una mayor competitividad. *Boletín Económico de ICE*, (2406):805–825, 1994.
- [26] D.G. Florian. Simulation-based evaluation of Advanced Traveler Information Services. *Thesis*, 2004.
- [27] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.
- [28] L. GiYoung, K. JeongJin, and H. You-Sik. The optimization of traffic signal light using artificial intelligence. *In The 10th IEEE International Conference on Fuzzy Systems*, 3:1279–1282, 2001.
- [29] J.L. González García. Aportación a la optimización multiobjetivo de la distribución en planta. *Tesis. Universidad Politécnica de valencia*, 2005.
- [30] M. S. Gowda. On the extended linear complementarity problem. *Mathematical Programming*, 72:33–50, 1996.
- [31] S. F. Hafstein, R. Chrobok, A. Pottmeier, M. Schreckenber, and F.C. Mazur. A high-resolution cellular automata traffic simulation model with application in a freeway traffic information system. *Computer-Aided Civil and Infrastructure Engineering*, 19:338–350, 2004.
- [32] M. HaiBo, Y. JianNing, and L. LinZhong. Traffic signals control of urban traffic. *In Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, pages 763–767, 2010.

- [33] A. Halati, H. Lieu, and S. Walker. CORSIM - corridor traffic simulation model. *In The 76th Annual Meeting of the Transportation Research Board*, pages 570–576, 1997.
- [34] D.W. Huang and W.N. Huang. Optimization of traffic lights at crossroads. *International Journal of Modern Physics C, Physics and Computer*, 14(5):539–548, 2003.
- [35] D.S. Johnson, C.R. Aragon, L.A. McGeoch, and C. Schevon. Optimization by simulated annealing: An experimental evaluation; part i, graph coloring and number partitioning. *Operations research*, 37:865–892, 1989.
- [36] W. Karush. Minima of functions of several variables with inequalities as side conditions. *Thesis*, 1939.
- [37] Ch. Kuei-Hsiang, L. Ren-Hao, and W. Meng-Hui. An intelligent traffic light control based on extension neural network. *In Proceedings of the 12th International Conference on Knowledge-based Intelligent Information and Engineering Systems*, pages 17–24, 2008.
- [38] H.W. Kuhn and A.W. Tucker. Nonlinear programming. *Proc. 2nd. Berkeley Symposium on Mathematical, Statistics and Probability*, J. Neyman (Ed.), University of California Press, Berkeley, California, 1951.
- [39] C. S. Lema, L. P. Pedreira, G. Bouza, and S. Allende. Un método híbrido para resolver el problema de control óptimo de semáforos. *Rect@*, 10(1):285–298, 2009.
- [40] L. Li, N. Tang, X. Mu, and F. Shi. Implementation of traffic lights control based on Petri nets. *In The 7th IEEE International Conference on Intelligent Transportation Systems*, 3:1749–1752, 2004.
- [41] G.F. List and M. Cetin. Modelling traffic signal control using petri nets. *IEEE Transactions on Intelligent Transportation Systems*, 5(3):177–187, 2004.
- [42] F. Logi and S. G. Ritchie. Development and evaluation of a knowledge-based system for traffic congestion management and control. *Transportation Research Part C: Emerging Technologies*, 9(6):433–459, 2001.
- [43] D.G. Luenberger and Yinyu Ye. *Linear and nonlinear programming*. Springer. New York, 2008.

-
- [44] J. López, A. García, I. García Blanco, and F. Zúñiga. Traffic lights fuzzy control proposals to improve vehicular flow. *Fourth Congress of Electronics, Robotics and Automotive Mechanics*, pages 406–411, 2007.
- [45] S. López, P. Hernández, A. Hernández, and M. Garcia. Artificial neural networks as useful tools for the optimization of the relative offset between two consecutive sets of traffic lights. *LNCS. Foundations and Tools for Neural Modeling. Springer-Verlag*, pages 795–804, 1999.
- [46] Lundy M. and A. Mees. Convergence of an annealing algorithm. *Mathematical programming*, 34(1):111–124, 1986.
- [47] O.L. Mangasarian. *Nonlinear Programming*. McGraw-Hill. New York, 1969.
- [48] O.L. Mangasarian and J.S. Pang. The extended linear complementarity problem. *SIAM Journal on Matrix Analysis and Applications*, 16:359–368., 1995.
- [49] N. Metrópolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of state calculation by fast computing machines. *Journal of chemistry physics*, 21:1087–1091, 1953.
- [50] Z. Michalewicz and D.B. Fogel. *How to solve it: Modern heuristics*. Berlin. Heidelberg, 2000.
- [51] T. Nagatani. Dispersion and scaling of fluctuating vehicles through a sequence of traffic lights. *Physica A*, 361(2):619–629, 2006.
- [52] K. Nagel and C. Barret. Using microsimulation feedback for trip adaptation for realistic traffic in Dallas. Technical Report LA-UR 97-1334, 1997.
- [53] K. Nagel and A. Schleicher. Microscopic traffic modeling on parallel high performance computers. *Parallel Computing*, 20(1):125–146, 1994.
- [54] I.H. Osman. An introduction to meta-heuristics. *Operational Research*, Tutorial papers:92–122, 1995.
- [55] J. Poli and L.H.A. Monteiro. Improving vehicle flow with traffic lights. *Advances in complex systems*, 8(1):59–63, 2005.
- [56] H. Rakha, M. Van Aerde, L. Bloomberg, and X. Huang. Construction and calibration of a large scale microsimulation model of the Salt Lake Area. *Transportation Research Record*, 1644:93–102, 1998.

- [57] N. Roupail, B. Park, and J. Sacks. Direct signal timing optimization: Strategy development and results. *In XI Pan American Conference in Traffic and Transportation Engineering*, 2000.
- [58] S. S. Kirkpatrick, J. R. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.
- [59] M. J. Smith. Optimum network control using traffic signals. *In IEEE Colloquium on Developments in Road Traffic Signalling*, pages 8/1–8/3, 1988.
- [60] J. C. Spall and D. C. Chin. A model-free approach to optimal signal light timing for system-wide traffic control. *In 33rd IEEE Conference on Decision and Control*, pages 1868–1875, 1994.
- [61] J. Sánchez, M. Galán, and E. Rubio. Genetic algorithms and cellular automata: A new architecture for traffic light cycles optimization. *In Proceedings of The Congress on Evolutionary Computation 2004 (CEC2004)*, 2:1668–1674, 2004.
- [62] J. Sánchez, M. Galán, and E. Rubio. Bit level versus gene level crossover in a traffic modeling environment. *In International Conference on Computational Intelligence for Modelling control and Automation (CIMCA 2005)*, 1:1190–1195, 2005.
- [63] J. Sánchez, M. Galán, and E. Rubio. Stochastic vs deterministic traffic simulator. Comparative study for its use within a traffic light cycles optimization architecture. *In Proceedings of the International Workconference on the Interplay between Natural and Artificial Computation (IWINAC)*, 2:622–631, 2005.
- [64] J. Sánchez, M. Galán, and E. Rubio. Study of correlation among several traffic parameters using evolutionary algorithms: Traffic flow, greenhouse emissions and network occupancy. *In Proceedings of the EUROCAST 2007 conference*, pages 1134–1141, 2007.
- [65] J. Sánchez, M. Galán, and E. Rubio. Applying a traffic lights evolutionary optimization technique to a real case: Las Ramblas. Area in Santa Cruz de Tenerife. *IEEE Transactions on Evolutionary Computation*, 12(1):25–40, 2008.
- [66] J. Sánchez-Toscano Barbero. *Temario específico ESTT-OEP, tema 82*. 2005.
- [67] E. Taniguchi and H. Shimamoto. Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times. *Transportation Research Part C: Emerging Technologies*, 12(3-4):235–250, 2004.

-
- [68] O. Tveit. Common cycle time. A strength or barrier in traffic light signalling. *Traffic Engineering and Control (TEC) Magazine*, 1(44):19–21, 2003.
- [69] H. R. Varia and S. L. Dhingra. Dynamic optimal traffic assignment and signal time optimization using genetic algorithms. *Computer-Aided Civil and Infrastructure Engineering*, 19:260–273, 2004.
- [70] Wann-Ming Wey. Applications of linear systems controller to a cycle-based traffic signal control. *Intelligent Transportation Systems*, pages 179–184, 2001.
- [71] M. Wiering, J. Vreeken, J. van Veenen, and A. Koopman. Simulation and optimization of traffic in a city. *In Intelligent Vehicles Symposium, IEEE 2004*, pages 453–458, 2004.
- [72] P. H. Wolfe. Finding the nearest point in a polytope. *Mathematical Programming*, 11:128–149, 1976.
- [73] F. Yan, M. Dridi, and A. El Moudni. Control of traffic lights in intersection: A new branch and bound approach. *In International Conference on Service Systems and Service Managament*, pages 1–6, 2008.
- [74] Q. Yang. A simulation laboratory for evaluation of dynamic traffic management systems. *PhD Thesis, Massachusetts Institute of Technology*, 1997.
- [75] Y. Ye. A fully polynomial-time approximation algorithm for computing a stationary point of the general linear complementarity problem. *Mathematics of Operations Research*, 18:334–345, 1993.
- [76] H. You-Sik, J. Hyunsoo, and P. Chong-Kug. New electrosensitive traffic light using fuzzy neural network. *In IEEE Transactions on Fuzzy Systems*, VII(6):759–767, 1999.
- [77] H. You-Sik, K. JongSoo, K. Jeong, and P. Chong-Kug. Estimation of optimal green time simulation using fuzzy neural network. *In Fuzzy Systems Conference Proceedings, FUZZY-IEEE 99*, pages 761–766, 1999.
- [78] S.H. Zanakis and J.R. Evans. Heuristic optimization: why, when and how to use it. *Interfaces*, 11:84–90, 1981.