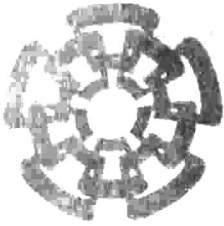




XX(138684.1)





Centro de Investigación y de Estudios Avanzados del I.P.N.  
Unidad Guadalajara

# **Un esquema multiagentes para la coordinación de tráfico urbano**

**CINVESTAV  
IPN  
ADQUISICION  
DE LIBROS**

Tesis que presenta:

**Zazhil-ha Herena Ulloa**  
para obtener el grado de:

**Maestro en Ciencias**  
en la especialidad de:

**Ingeniería Eléctrica**  
Directores de Tesis

**Dr. Luis Ernesto López Mellado**  
**Dr. Antonio Ramírez Treviño**

**CINVESTAV IPN**  
USB INFORMACION Y DOCUMENTACION  
SERVICIO DOCUMENTAL

Guadalajara, Jalisco, Octubre de 2007.

CLASIF.: TR115.68 H48 2007  
ADQUIS.: SSI-473  
FECHA: 2-VI-2008  
PROCED.: DON-2008  
\$ \_\_\_\_\_

I. D. 137710 - 2001

# **Un esquema multiagentes para la coordinación de tráfico urbano**

**Tesis de Maestría en Ciencias  
Ingeniería Eléctrica**

Por:

**Zazhil-ha Herena Ulloa**

Ingeniera en Sistemas Computacionales  
Instituto Tecnológico de Tepic 2000-2005

Becario de CONACYT, expediente no. 199608

Directores de Tesis

**Dr. Luis Ernesto López Mellado**

**Dr. Antonio Ramírez Treviño**

CINVESTAV del IPN Unidad Guadalajara, Octubre de 2007.



## **Agradecimientos:**

*A mi padre, Jaime Herena por representar un motivo en mi vida.*

*A mis madre, Ana María Ulloa y a mi tía, Micaela Ulloa por su amor infinito, por su apoyo incondicional y por ser un ejemplo a seguir.*

*A mis hermanos, Chava, Anna y Werner por estar siempre dispuestos a ayudarme.*

*A todos mis compañeros y amigos por su ayuda, por sus consejos y por los buenos momentos que compartimos.*

*A mis asesores, Dr. Ernesto López y Dr. Antonio Ramírez por el tiempo dedicado, por la paciencia y por el voto de confianza.*

*A mis profesores por compartir sus conocimientos conmigo.*

*A Dios por permitirme estar aquí.*

*Al **CONACYT** por su apoyo.*

# **Un Esquema Multiagentes para la Coordinación de Tráfico Urbano**

## **Resumen**

Se presenta un esquema basado en sistemas multiagentes para la coordinación de tráfico urbano, donde cada intersección es controlada por un agente. La operación del controlador es en dos fases: a) control local y b) negociación. En la fase de control local se obtiene una asignación de tiempos de acuerdo a las condiciones de tráfico de la intersección. La decisión final sobre la temporización a implementar es negociada con los agentes de intersecciones adyacentes. Para el control local se proponen dos técnicas: una basada en un sistema de inferencia difuso otra basada en un enfoque de programación lineal. Para la negociación entre agentes se propone un algoritmo de interacción basado en la teoría de negociación de Nash. El esquema de coordinación fue probado en un simulador de tráfico urbano orientado a eventos.

# **A Multiagent Based Scheme for Urban Traffic Coordination**

## **Abstract**

A Multiagent Based Scheme for Urban traffic Coordination is presented. In this scheme every intersection is controlled by an agent. The controller operates in two stages: a) local control, and b) negotiation. During the local control stage a timing assignation is determined from the traffic conditions on the intersection streets. The final decision on the timing to be implemented is negotiated among the agent from other intersections. For the local control stage two techniques are proposed: one based on a fuzzy inference approach, and other one based on a linear programming approach. For the negotiation an interaction algorithm based on Nash's negotiation theory is proposed. The coordination scheme has been tested in an event oriented urban traffic simulator.



# Índice

<i>Índice de Figuras</i> .....	<i>V</i>
<i>Índice de Tablas</i> .....	<i>VIII</i>
<i>Introducción</i> .....	<i>1</i>
<b>1 Sistemas de Control de Tráfico Urbano</b> .....	<b>3</b>
<b>1.1. Sistemas de Control de Tráfico Urbano.</b> .....	<b>3</b>
1.1.1. Nociones sobre Sistemas de Control de Tráfico Urbano. ....	3
1.1.2. El problema de Control de Tráfico Urbano.....	5
1.1.3. Técnicas actuales de Control de Tráfico .....	6
<b>1.2. Enfoques de Control de Sistemas de Tráfico Urbano</b> .....	<b>7</b>
1.2.1. Control Óptimo .....	7
1.2.2. Control Adaptable .....	10
1.2.3. Coordinación con el concepto de reputación .....	11
1.2.4. Control con Aprendizaje .....	12
1.2.5. Sistemas Multi-Agentes .....	14
1.2.6. Predicción de Tráfico Urbano .....	17
<b>1.3. Modelado de Sistemas de Tráfico Urbano</b> .....	<b>17</b>
1.3.1. Redes de Petri Coloreadas.....	17
1.3.2. Redes de Petri Híbridas .....	18
<b>1.4. Conclusiones</b> .....	<b>20</b>
<b>2 Un Esquema Multiagentes para la Coordinación de Tráfico Urbano</b> .....	<b>21</b>
<b>2.1. Simulación de Tráfico Urbano</b> .....	<b>21</b>
2.1.1. Arquitectura del Simulador de Tráfico Urbano.....	21
2.1.2. Generador de Vehículos .....	23
2.1.3. Comunicación entre diferentes módulos del simulador. ....	24
<b>2.2. Integración del esquema de Control de Tráfico al simulador</b> .....	<b>25</b>

2.2.1.	Interacción entre Simulador y Agentes de Control .....	26
2.2.2.	Requerimientos del Módulo de Control .....	27
<b>2.3.</b>	<b>Conclusiones.....</b>	<b>27</b>
<b>3</b>	<b><i>Un Esquema Multiagentes para el Control de Tráfico Urbano.</i> .....</b>	<b>28</b>
<b>3.1.</b>	<b>Agente de Control.....</b>	<b>28</b>
3.1.1.	Descripción del Agente de Control .....	29
3.1.2.	Ambiente del Agente de Control.....	29
3.1.3.	Arquitectura Interna del Agente de Control.....	30
<b>3.2.</b>	<b>Control basado en Lógica Difusa .....</b>	<b>33</b>
3.2.1.	Variables Lingüísticas .....	33
3.2.2.	Funciones de Membresía.....	34
3.2.3.	Estructura del Esquema Difuso .....	36
3.2.4.	Base de Reglas .....	38
<b>3.3.</b>	<b>Enfoque de Programación Lineal .....</b>	<b>40</b>
<b>3.4.</b>	<b>Implementación del esquema de control .....</b>	<b>43</b>
3.4.1.	Estrategia de Implementación .....	43
3.4.2.	Comunicación entre Módulo de Control y Matlab.....	45
<b>3.5.</b>	<b>Conclusiones.....</b>	<b>45</b>
<b>4</b>	<b><i>Coordinación de Semáforos</i> .....</b>	<b>46</b>
<b>4.1.</b>	<b>Estrategia general de la Coordinación de Tráfico Urbano.....</b>	<b>46</b>
4.1.1.	Planteamiento del Problema .....	46
4.1.2.	Análisis de problemas .....	47
<b>4.2.</b>	<b>Teoría de Negociación de Nash.....</b>	<b>49</b>
4.2.1.	Dominio orientado a valor de la negociación.....	50
4.2.2.	Modelos de cooperación y conflicto .....	50
4.2.3.	Escenario de Negociación y solución de Nash.....	51
<b>4.3.</b>	<b>Protocolo de Negociación .....</b>	<b>51</b>
<b>4.4.</b>	<b>Conclusiones.....</b>	<b>53</b>

<b>5 Pruebas y Análisis de Resultados</b> .....	<b>54</b>
<b>5.1. Caso de Estudio</b> .....	<b>54</b>
<b>5.2. Simulaciones realizadas</b> .....	<b>55</b>
<b>5.3. Resultados Obtenidos</b> .....	<b>56</b>
5.3.1. Escenario Alto .....	58
5.3.2. Escenario Medio.....	60
5.3.3. Escenario Bajo.....	66
<b>5.4. Conclusiones</b> .....	<b>70</b>
<b>Conclusiones</b> .....	<b>71</b>
<b>A. Apéndice A</b> .....	<b>73</b>
<b>A.1. Lógica Difusa</b> .....	<b>73</b>
A.1.1. Conjuntos Difusos y Funciones de Membresía .....	73
A.1.2. Variables Lingüísticas y Operaciones con lógica difusa.....	73
A.1.3. Modelo Takagi-Sugeno .....	74
A.1.4. Control Difuso.....	75
<b>A.2. Teoría de Juegos en Sistemas Multi-Agentes</b> .....	<b>75</b>
A.2.1. Encuentros Multiagentes .....	75
A.2.2. Teoría de Juegos.....	76
<b>B. Resultados obtenidos de las simulaciones</b> .....	<b>79</b>
<b>B.1. Caso de Estudio</b> .....	<b>79</b>
<b>B.2. Escenario Alto</b> .....	<b>80</b>
B.2.1. Llegadas de coches .....	80
B.2.2. Flujo de Salida sin negociación.....	84
B.2.3. Tiempos de verde sin negociación .....	85
B.2.4. Flujo de Salida con negociación.....	87
B.2.5. Tiempos de verde con negociación .....	90
<b>B.3. Escenario Medio</b> .....	<b>92</b>
B.3.1. Llegadas de coches.....	92



B.3.2.	Flujo de Salida sin negociación.....	95
B.3.3.	Tiempos de verde sin negociación .....	97
B.3.4.	Flujo de Salida con negociación.....	99
B.3.5.	Tiempos de verde con negociación .....	101
<b>B.4.</b>	<b>Escenario Bajo .....</b>	<b>103</b>
B.4.1.	Llegadas de coches.....	103
B.4.2.	Flujo de Salida sin negociación.....	106
B.4.3.	Tiempos de verde sin negociación .....	107
B.4.4.	Flujo de Salida con negociación.....	109
B.4.5.	Tiempos de verde con negociación .....	111
<b>REFERENCIAS.....</b>		<b>113</b>

# Índice de Figuras

Figura 1.1 Procedimiento de los Agentes para resolver el problema de Control [Xiao, 2004].	13
Figura 1.2 Sistema con coordinación centralizada y descentralizada [Ossowski, 1998].	14
Figura 1.3 Control anfitrión para tráfico [Wang, 2005].	16
Figura 1.4 Arquitectura de red neuronal utilizada como individuo [Annunziato, 2003].	17
Figura 1.5 Intersección con 5 movimientos permitidos [Dotoli, 2005].	18
Figura 1.6 Modelo de una intersección con RPC [Dotoli, 2005].	18
Figura 1.7 Red de Petri discreta modelando el semáforo en una intersección [Júlvez, 2005].	19
Figura 1.8 Red de Petri híbrida modelando una intersección. [Júlvez, 2005].	19
Figura 2.1 Algoritmo de Control de la simulación [Neri, 2005a].	24
Figura 2.2 Ambiente distribuido del simulador	25
Figura 2.3 Interacción entre los agentes de control y la máquina de simulación.	26
Figura 3.1 Arquitectura del Agente de Control	31
Figura 3.2 Actividades del agente de control.	33
Figura 3.3 Funciones de Membresía para Errori.	35
Figura 3.4 Funciones de membresía para Cambioi.	35
Figura 3.5 Funciones de membresía para Sig_semi.	35
Figura 3.6 Esquema de Control Difuso	37
Figura 3.7 Ejemplo de una intersección controlada por un agente	42
Figura 3.8 Diagrama de Arquitectura de Software del esquema de de control.	44
Figura 4.1 Roles del agente de control	47
Figura 4.2 Protocolo de Negociación modelado con una RPC	53
Figura 5.1 Caso de Estudio.	54
Figura 5.2 Flujo de Entrada Calle C1 y C11	56
Figura 5.3 Fila en la calle C1.	57
Figura 5.4 Fila en la calle C11.	57
Figura 5.5 Entrada real de coches calle C1.	58
Figura 5.6 Flujos en escenario Alto	60
Figura 5.7 Tiempos de verde calles C1 y C11, esquema PL.	60
Figura 5.8 Tiempos de verde calles C1 y C11, esquema difuso.	61
Figura 5.9 Flujo calle C11.	61

Figura 5.10 Flujo calle C2, esquema PL.....	61
Figura 5.11 Flujo calle C3, esquema PL.....	62
Figura 5.12 Flujo calle C2, esquema difuso.....	62
Figura 5.13 Flujo calle C3, esquema difuso.....	62
Figura 5.14 Tiempos de verde calle C2, esquema difuso.....	63
Figura 5.15 Tiempos de verde calle C2, esquema PL.....	63
Figura 5.16 Flujos en escenario Medio.....	64
Figura 5.17 Tiempos de verde calles C1 y C11, esquema PL.....	65
Figura 5.18 Tiempos de verde calles C1 y C11, esquema difuso.....	65
Figura 5.19 Flujo calle C2, esquema PL.....	66
Figura 5.20 Flujo calle C2, esquema difuso.....	66
Figura 5.21 Flujos en escenario Bajo.....	67
Figura 5.22 Flujo calle C2, esquema PL.....	68
Figura 5.23 Flujo calle C3, esquema PL.....	68
Figura 5.24 Flujo calle C2, esquema difuso.....	68
Figura 5.25 Flujo calle C3, esquema difuso.....	69
Figura 5.26 Tiempos de verde calle C2, esquema difuso.....	69
Figura 5.27 Tiempos de verde calle C2, esquema PL.....	69
Figura A.1 Variable Lingüística <i>Temperatura</i> .....	74
Figura A.2 Estructura de un Controlador Difuso.....	75
Figura A.3 Región alcanzable con su frontera Pareto.....	78
Figura B.1 Caso de Estudio.....	79
Figura B.2 Flujos de Entrada Calles C1, C11, C14, C15.....	81
Figura B.3 Flujos de Entrada calles C18, C6.....	82
Figura B.4 Flujo de Entrada calle C1.....	82
Figura B.5 Flujos de Entrada calles C2, C3, C4.....	83
Figura B.6 Flujos de Salida calles C1, C2, C3, C4.....	84
Figura B.7 Tiempos de verde intersecciones 1, 2, 3, esquema difuso.....	85
Figura B.8 Tiempos de verde intersección 4, esquema difuso.....	86
Figura B.9 Tiempos de verde intersecciones 1, 2, esquema PL.....	86
Figura B.10 Tiempos de verde intersecciones 3, 4, esquema PL.....	87
Figura B.11 Flujo de Salida calle C1.....	87
Figura B.12 Flujos de Salida calles C2, C3, C4.....	88
Figura B.13 Flujos de Salida calles C1, C2, C3, C4.....	89
Figura B.14 Tiempos de verde antes y después de negociar, intersecciones 1, 2, 3.....	90
Figura B.15 Tiempos de verde antes y después de negociar, intersección 4.....	91



Figura B.16 Tiempos de verde antes y después de negociar, intersecciones 1, 2. ....	91
Figura B.17 Tiempos de verde antes y después de negociar, intersecciones 3, 4. ....	92
Figura B.18 Flujo de Entrada calle C1. ....	92
Figura B.19 Flujos de Entrada calles C6, C11, C14, 15. ....	93
Figura B.20 Flujo de Entrada calle C18. ....	94
Figura B.21 Flujos de Entrada calles C1, C2. ....	94
Figura B.22 Flujos de Entrada calles C3, C4. ....	95
Figura B.23 Flujo de Salida calle C1. ....	95
Figura B.24 Flujos de Salida calle C2, C3, C4. ....	96
Figura B.25 Tiempos de verde intersecciones 1, 2, 3, 4, esquema difuso. ....	97
Figura B.26 Tiempos de verde intersecciones 1, 2, 3, 4, esquema PL. ....	98
Figura B.27 Flujos de Salida calles C1, C2, C3, C4. ....	99
Figura B.28 Flujos de Salida calles C1, C2, C3, C4. ....	100
Figura B.29 Tiempos de verde antes y después de negociar, int. 1, 2, 3, 4, esquema difuso. ....	101
Figura B.30 Tiempos de verde antes y después de negociar int. 1, 2, 3, 4, esquema PL. ....	102
Figura B.31 Flujos de Entrada calles C1, C6, C11. ....	103
Figura B.32 Flujos de Entrada calles C12, C15, C18. ....	104
Figura B.33 Flujos de Entrada calles C1, C2, C3, C4. ....	105
Figura B.34 Flujos de Salida calles C1, C2, C3, C4. ....	106
Figura B.35 Tiempos de verde intersecciones 1, 2, 3, 4, esquema difuso. ....	107
Figura B.36 Tiempos de verde intersecciones 1, 2, 3, 4, esquema PL. ....	108
Figura B.37 Flujos de Salida calles C1, C2, C3, C4. ....	109
Figura B.38 Flujos de Salida calles C1, C2, C3, C4. ....	110
Figura B.39 Tiempos de verde antes y después de negociar int. 1, 2, 3, 4, esquema difuso. ....	111
Figura B.40 Tiempos de verde antes y después de negociar int. 1, 2, 3, 4, esquema PL. ....	112

# Índice de Tablas

Tabla 2.1 Eventos manejados por el simulador.....	22
Tabla 3.1 Variables Lingüísticas de entrada .....	34
Tabla 3.2 Variables de Salida.....	34
Tabla 3.3 Funciones de membresía .....	36
Tabla 3.4 Reglas etapa local fase no. 1 .....	38
Tabla 3.5 Reglas etapa local fase no. 2 .....	38
Tabla 3.6 Reglas etapa no egoísta fase no. 1 y fase no. 2 .....	39
Tabla 3.7 Reglas de decisión etapa de decisión .....	40
Tabla 5.1 Variaciones en la tasa de llegada para el escenario Alto. ....	55
Tabla 5.2 Variaciones en la tasa de llegada para el escenario Medio. ....	55
Tabla 5.3 Variaciones en la tasa de llegada para el escenario Bajo. ....	56
Tabla B.1 Variaciones en la tasa de llegada para el escenario Alto.....	80
Tabla B.2 Variaciones en la tasa de llegada para el escenario Medio. ....	80
Tabla B.3 Variaciones en la tasa de llegada para el escenario Bajo. ....	80

# Introducción

El aumento de la cantidad de automóviles que circulan día con día en las ciudades ha provocado congestionamientos vehiculares recurrentes. Este problema da como resultado pérdidas económicas, debido por un lado, a retrasos, generando un incremento en los costos de transporte de bienes, y por otro lado, a la baja productividad en conductores fatigados y molestos [Vogel, 2000].

La mejora del flujo vehicular puede lograrse de varias maneras: incrementando la infraestructura vial, mejorando el servicio de transporte público e incentivando el uso de éste, o bien implementado nuevas estrategias de control de semáforos. Esta última solución parece la menos costosa pues la inversión requerida se enfoca a complementar la infraestructura de señalización ya existente [Hewage, 2004].

Un Sistema de Control de Tráfico Urbano debe proponer estrategias de control adecuadas a las condiciones de tráfico que se presenten a lo largo del día. Estas estrategias deben ser cuidadosamente evaluadas antes de ser implementados, debido al impacto que causarán en el flujo de tráfico. Las simulaciones por computadora son un método efectivo para realizar evaluaciones de cambios en las políticas de control y observar el comportamiento de los sistemas de control propuestos [Ehlert, 2001].

La optimización de los tiempos para los semáforos es un problema complejo, ya que el estado de un semáforo afecta el flujo de tráfico hacia muchas otras intersecciones. Si el flujo vehicular enviado por una intersección (cruce) es mayor al flujo vehicular que pueden soportar las calles de las intersecciones vecinas, entonces se generan problemas de saturación. Por esta razón es importante encontrar un esquema de coordinación efectivo donde se logre que los cambios en los tiempos de verde en una intersección no causen problemas en las intersecciones adyacentes.

En este contexto, el objetivo planteado para esta tesis, ha sido proponer una estrategia de control-coordinación que permita mejorar el flujo vehicular, adaptándose a las variaciones en las condiciones de tráfico.

Para ello, en esta tesis se propone un esquema basado en sistemas multiagentes para la coordinación de tráfico urbano, donde cada intersección es controlada por un agente; de esta forma cada agente se puede comunicar con los agentes de las intersecciones adyacentes. Cada

agente es capaz de realizar el control local de los tiempos de verde de su intersección; éste puede decidir si lleva a cabo una negociación con los agentes vecinos.

Para el control local se proponen dos técnicas: una basada en un sistema de inferencia difuso otra basada en un enfoque de programación lineal. Para la coordinación entre agentes se propone un algoritmo de interacción basado en la teoría de negociación de Nash. Esta propuesta fue desarrollada y probada en un simulador de tráfico urbano orientado a eventos, trabajo de investigación activo al cual se le agregaron algunas funcionalidades para lograr escenarios de control mas reales, así como para que cumpliera con los requerimientos del control propuesto.

El presente documento está organizado como sigue. En el capítulo 1 se presentan las nociones básicas de un sistema de control de tráfico urbano, así como el estado del arte del control de tráfico urbano y las metodologías existentes.

En el capítulo 2 se describe brevemente la arquitectura del simulador de tráfico urbano y el módulo de generación de vehículos.

En el capítulo 3 se presenta la arquitectura del esquema multiagentes y las dos técnicas de control propuestas para la asignación de tiempos de verde de las intersecciones, así como los componentes del agente de control.

En el capítulo 4 se presenta un esquema multiagentes para la coordinación de tráfico urbano, la forma en que estos interactúan con el ambiente y entre ellos mismos.

En el capítulo 5 se presenta el caso de estudio donde se muestra la forma en que nuestro sistema de control es capaz de adaptarse a la demanda de tráfico por medio de pruebas realizadas bajo diferentes condiciones de tráfico.

Finalmente se incluyen dos apéndices: el apéndice A que presenta los conceptos básicos de lógica difusa y la teoría de negociación de Nash; el apéndice B presenta resultados de simulación adicionales.



# Capítulo 1

## Sistemas de Control de Tráfico Urbano

**Resumen:** En este capítulo se presentan los conceptos básicos sobre Sistemas de Control de Tráfico Urbano. Se incluye también una revisión de los diferentes enfoques actuales de estos sistemas y de las técnicas de modelado empleadas.

### 1.1. Sistemas de Control de Tráfico Urbano.

La congestión en las calles actualmente es parte de la vida diaria en las ciudades, cuando la población crece, la congestión de tráfico empeora y construir más calles no siempre ayuda mucho, o resulta ser una solución muy costosa.

Los sistemas de control de tráfico urbano ubicados en las intersecciones de las ciudades, son los encargados de conceder el derecho de paso a los vehículos que pretenden cruzar la intersección. Los semáforos son utilizados para indicar a los conductores si tienen o no derecho de paso. La congestión, accidentes, y retraso en el viaje son algunos de los problemas que se presentan si una intersección no está bien controlada, y pueden ser minimizados si el sistema de control propone las estrategias de control adecuadas. La optimización en los cambios de luces de semáforos incrementa el flujo de vehículos y además puede prevenir las congestiones de tráfico, [Wiering, 2004].

#### 1.1.1. Nociones sobre Sistemas de Control de Tráfico Urbano.

El control de tráfico facilita la seguridad de las calles al asegurar el movimiento ordenado, eficiente y predecible de todo el tráfico en la red. [Chee, 2002].

Los Sistemas de Transporte Inteligentes, ITS, se encargan de investigar sobre sistemas de seguridad dentro de los autos, simulación de efectos por cambios en la infraestructura, planeación de rutas, optimización de transportes, etc. Sus principales objetivos son mejorar la seguridad,

minimizar el tiempo de viaje e incrementar la capacidad de las infraestructuras. Tales objetivos traen beneficios a la salud, la economía y el ambiente.

La calidad de los sistemas de control de tráfico se determina por la relación del esquema de control con los patrones de tráfico actuales. Si los patrones de tráfico cambian, la efectividad se determina por la forma en que el sistema se adapta a estos cambios. Cuando esta habilidad para adaptarse se vuelve una parte integral del sistema de control, éste puede reaccionar mejor a los cambios de las condiciones de tráfico [Roosmond, 2000].

Se puede decir que un sistema de control de tráfico es más eficiente si para una densidad dada de autos, logra incrementar la velocidad promedio de los vehículos [Gershenson, 2006].

Una intersección es el área de cruce común entre un conjunto de calles.

El tiempo verde es el lapso de tiempo durante el cual se permite el paso a los vehículos que se encuentran en la dirección hacia donde se despliega la luz verde en el semáforo.

Una fase es el conjunto de movimientos en la intersección que tienen permiso de cruzarla durante un determinado tiempo. Una secuencia de fases es una secuencia de los conjuntos de movimientos permitidos.

Un ciclo es el lapso de tiempo en el cual todas las calles que envían flujo de vehículos hacia la intersección obtienen un tiempo verde.

El offset de un ciclo define el tiempo en que inicia el ciclo de una intersección con relación al inicio del ciclo de la intersección de la cual recibe flujo de vehículos. El offset puede ser ajustado para permitir que varios semáforos cooperen, y crear por ejemplo una oleada verde, la cual se define como un pelotón que cruza varias intersecciones sin necesidad de detenerse en ninguna de ellas. Un pelotón es un número de carros que viajan a alta velocidad, con una pequeña distancia entre ellos. Cada conductor se asegura de que está siguiendo al líder.

La sobrecapacidad ocurre cuando filas de vehículos llenan completamente la capacidad de la calle, interfiriendo con el desempeño de las calles adyacentes, [Girianna, 2004].

En control de tráfico, las estrategias de control usualmente consisten en cambiar la secuencia de fases en la intersección, el tiempo verde de cada fase, el tiempo de ciclo de la intersección y/o el offset entre intersecciones sucesivas.

El control de tipo fijo o pre-temporizado se caracteriza por el uso ineficiente de la capacidad de la calle, debido que no se puede ajustar a las variaciones en el flujo de tráfico y la demanda actual de tráfico. En la práctica la mayoría de los semáforos son controlados a través de un tiempo fijo. Los controladores fijos se deben adaptar a situaciones específicas para comportarse adecuadamente. A menudo se utiliza una tabla de especificaciones de acuerdo al día y la hora para adaptarse a eventos recurrentes como horas pico.

Un controlador adaptable mejora al fijo al operar las señales de acuerdo a la demanda de tráfico, registrada a través de detectores de vehículos. El tiempo verde puede variar entre una

longitud mínima y una longitud máxima dependiendo de los flujos. Las longitudes de ciclos y fases son ajustadas de acuerdo a los datos obtenidos por los detectores. También se toman en cuenta parámetros como la hora, el día, la temporada, el clima y situaciones impredecibles como accidentes o actividades de construcción.

Mientras más fases haya en una intersección, la utilización de un control adaptable puede volverse muy complicado y el control de la intersección será efectivo solo si se tiene coordinación con las calles adyacentes.

La coordinación es un proceso en el cual las intersecciones involucradas están formulando sus acciones de control de acuerdo a su propia información y a la información pasada por el resto de las intersecciones, buscando una acción de control que afecte lo mínimo posible a todas las intersecciones involucradas.

En las grandes ciudades, los centros de control de tráfico (TCC) están a cargo de la administración del tráfico urbano, así como de mantener y restaurar el flujo de los carros. Dentro de un TCC, los ingenieros de tráfico hacen uso de fuentes de información proporcionadas por cámaras de video o sensores para identificar problemas de tráfico y utilizan los dispositivos de control de tráfico para superarlos [ Ossowski, 1998]

### **1.1.2. El problema de Control de Tráfico Urbano**

El incesante incremento del número de automóviles ha causado un aumento en la densidad de tráfico en las calles, especialmente en las áreas urbanas, donde el volumen de tráfico es enorme en las horas pico y pérdidas económicas significativas se han producido debido a congestiones recurrentes.

La congestión de tráfico ha alcanzado niveles tan altos que se han incrementado las muertes en accidentes y las emisiones de contaminantes por parte de los vehículos. Los retrasos en los viajes son una fuente constante de tensión, frustración e insatisfacción para los automovilistas. También incrementan los costos de operación en el movimiento de bienes, lo que lleva a costos más altos para los consumidores. Se espera que la cantidad de automovilistas se incremente en el futuro, agravando estos problemas, por lo que aún modestas mejoras en la disminución de la congestión de tráfico pueden traer grandes beneficios. [Vogel, 2000].

Los automóviles en el tráfico urbano pueden experimentar largos tiempos de viaje debido a un control de tráfico ineficiente. Por lo tanto, un control óptimo de los semáforos utilizando sensores y algoritmos inteligentes puede traer muchos beneficios.

La optimización de los tiempos para los semáforos es un problema complejo. Aún para intersecciones aisladas no existe una solución óptima obvia. Con múltiples intersecciones el problema se vuelve mas complejo, porque el estado de un semáforo afecta el flujo de tráfico



hacia muchas otras intersecciones. Otra complicación es que el flujo de tráfico cambia constantemente, dependiendo de la hora, el día de la semana y la época del año.

Debido a que las acciones locales tomadas en una parte del sistema de tráfico afectan la efectividad de otras, es necesario lograr un cierto nivel de coordinación para obtener un conjunto de decisiones coherente para el sistema. De esta forma, se previene que la red de tráfico se degenera a un estado en el cual el flujo de vehículos disminuya o se detenga completamente.[Chee, 2003]

### 1.1.3. Técnicas actuales de Control de Tráfico

Todos los sistemas de control de tráfico adaptables utilizados en la actualidad, a menudo operan con base a la adaptación a los tiempos verdes de las fases. Estos métodos aún no son óptimos cuando las demandas de tráfico cambian rápidamente en un intervalo de tiempo. La idea para lograr un control óptimo es que las herramientas que generan políticas de control de semáforos tomen decisiones racionales acerca del plan bajo condiciones de tráfico variables. Sin embargo, casi ninguna de las herramientas disponibles actualmente se comportan proactivamente o tienen reglas que puedan cambiar el comportamiento del controlador del sistema. A continuación se presentan los sistemas de control de tráfico urbano que actualmente se utilizan en diversas partes del mundo.

El sistema TRYS ha sido desarrollado para proveer soporte en las decisiones de tiempo real para los controladores de tráfico. TRYS se basa en un conjunto de agentes de control de tráfico, cada uno responsable de la administración de tráfico en su área local. Con base en los datos de los sensores, cada agente presenta propuestas de políticas de control para los semáforos. Los conflictos potenciales entre agentes se resuelven por un agente especial coordinador, que recibe los planes de señales de cada uno de los agentes y los armoniza para obtener planes globales consistentes. Estos planes de señales son presentados al operador quien finalmente decide si utilizarlos o modificarlos. TRYS ha sido instalado y evaluado en un sitio de prueba de Barcelona [Cuenca, 1996].

SCOOT es una herramienta para administrar y controlar las señales de tráfico en áreas urbanas. Es un sistema adaptable que responde automáticamente a los cambios en el flujo de tráfico a través del uso de detectores instalados en la calle. El envío de las instrucciones de control para los dispositivos lo hace a través de líneas telefónicas dedicadas. Ha sido implementado en mas de 200 ciudades, incluyendo Pekin, Londres, Chicago, Sao Paulo, Toronto, entre otras. SCOOT opera en cuatro áreas clave: comunicaciones, control de congestión, prioridad a autobuses y servicios a peatones [Breherton, 2004].

SCATS (Sydney Coordinated Adaptive Traffic System) recopila información acerca de los flujos de tráfico en tiempo real en cada intersección, si el espacio entre cada vehículo se incrementa, esto indica que se está desperdiciando tiempo verde, entonces SCATS busca reasignar ese tiempo verde a la calle donde la demanda es mas grande. Cada computadora regional envía instrucciones de cambios a las políticas de control a los semáforos de 120 intersecciones. SCATS está siendo utilizado en áreas urbanas como Hong Kong, Sydney, Jakarta, Detroit, Manila y Dublín [SCATS, 2007].

TRANSYT es un programa que se ejecuta fuera de línea para determinar y estudiar los tiempos de ciclo óptimos para las señales de tráfico urbano, en cualquier red de tráfico donde los flujos de tráfico promedio son conocidos; TRANSYT es el más famoso de los modelos para determinar los tiempos de verde óptimos. El programa puede ser utilizado para dar prioridad a vehículos de emergencia o autobuses [TRANSYT, 2007].

## **1.2. Enfoques de Control de Sistemas de Tráfico Urbano**

Ahora se presentan los principales enfoques estudiados para establecer el control en sistemas de tráfico urbano; los enfoques están organizados de acuerdo a la forma en que atacan el problema y se analizan además los beneficios y costos que puede traer su utilización.

### **1.2.1. Control Óptimo**

Se analizan ahora los enfoques que resuelven el problema tratándolo como un problema de optimización.

#### **Simulación**

En [Hewage, 2004] se presenta una herramienta de simulación de propósito especial para optimizar los tiempos de verde de los semáforos. También provee los tiempos de verde para ciertos períodos de tiempo de acuerdo a la demanda de tráfico. El usuario puede dibujar la red de tráfico que desee simular limitándose a un máximo de cuatro intersecciones. La herramienta checa la demanda de tráfico frecuentemente y calcula el tiempo de verde óptimo basado en la demanda instantánea. Como beneficios tiene que el usuario puede cambiar los patrones de tráfico de acuerdo a sus predicciones y así poder reconocer posibles futuras congestiones de tráfico, además la simulación puede ser ejecutada por cualquier período de tiempo. Las principales desventajas son que no puede manejar mas de 4 intersecciones, supone que los peatones cruzan la calle únicamente cuando la luz está en rojo y además que los buenos resultados de la simulación se garantizan de acuerdo a la precisión de los datos de entrada.

En [Tavladakis, 1999] se utiliza la simulación para asignar los tiempos verdes óptimos de la intersección, simulando 5 posibles políticas de control, una exactamente a la actualmente utilizada, otras 2 segundos menos y más de la actualmente utilizada y otras 5 segundos menos y más de la actualmente utilizada. Este enfoque está muy limitado a las posibles elecciones de adaptación de políticas de control.

### Algoritmos Genéticos

En [Sun, 2003] se presenta la aplicación de un NSGA-II, un algoritmo genético de segunda generación que se basa en varias capas de clasificación de los individuos. Los individuos se clasifican y seleccionan para resolver el problema multiobjetivo de la optimización de tiempos de los semáforos. Como resultado se obtienen tiempos de verde para las intersecciones. El problema de optimización se formula de la siguiente manera:

$$\begin{aligned} \text{Min} \quad & F(G) = [f_1(G), f_2(G)] \\ \text{Sujeto a} \quad & g'_i \leq G_i \leq g''_i \quad \text{para } k = 1, 2 \end{aligned}$$

Donde:

- $G_i$ : Es el vector de tiempo verde efectivo para cada fase  $i$ .
- $g'_i$ : Es el límite inferior del tiempo efectivo verde para la fase  $i$ .
- $g''_i$ : Es el límite superior del tiempo efectivo verde para la fase  $i$ .
- $f_1(G)$ : Es la primera función objetivo con respecto al retraso.
- $f_2(G)$ : Es la segunda función objetivo con respecto al número de paradas.

Como los algoritmos genéticos buscan las soluciones óptimas basados en poblaciones de puntos, se pueden encontrar soluciones optimas Pareto múltiples en una sola corrida. Entonces este conjunto de soluciones se localiza a lo largo de cierta línea recta dentro de un espacio alcanzable de soluciones, de donde los encargados pueden seleccionar la mas apropiada para las condiciones actuales.

En [Girianna, 2004], se presenta un algoritmo genético para generar tiempo de verde de semáforos óptimos, dependiendo de las variaciones en las demandas de tráfico. Ahora veamos el planteamiento del problema de optimización:

- $G = (N, L, P)$  : Denota la red de tráfico de señales.
- $N$ : Un conjunto de señales.
- $L$ : Un conjunto de calles.
- $L_p$ : Un conjunto de calles a lo largo de una ruta.



La función objetivo se divide en dos partes, la primera es el número de vehículos que libera la intersección multiplicado por la razón de la distancia de la intersección  $i$  a la  $j$  y la máxima longitud de la calle. La segunda parte representa un factor de descuento que penaliza la ocurrencia de una cola al final del tiempo verde.

$$\text{Max } Z = \sum_t^T \sum_{(i,j) \in L} \sum_h^H \frac{d_{i,j}}{d_{\max}} D_{i,j}^h(t) \quad (1\text{er. parte})$$

$$- \sum_k^K \sum_{(i,j) \in L_p} \delta_{i,j(k)} \max(q_{i,j}^{h^*}(k) - q_{i,j}^{\max}) \quad (2\text{da. parte})$$

*Sujeto a*  $\delta_{i,j(k)}, d_{i,j}, d_{i,j} > 0$

Donde:

$q_{i,j}^{h^*}(k)$ : Número de vehículos en la cola aproximándose a la señal  $j$  que vienen de la señal  $i$ , al inicio de la fase siguiente a  $h$ .

$\delta_{i,j(k)}$ : Factor no- negativo de descuento.

La función de descuento asegura que la cola sea menor que la capacidad de la calle. A esta función se le agregan restricciones para asegurar offsets ideales y para que el tiempo verde se mantenga dentro de un mínimo y un máximo.

Esta función objetivo se extiende de la siguiente manera para definir el desempeño de los individuos como :

$$\text{fitness}_i = C_{\min} + (Z_i - \sum_{j=1}^m \mu_j H_j)$$

Donde:

$m_j$ : Coeficiente de penalización para la restricción  $j$ .

$m$ : Número de restricciones implícitas.

$H_j$ : Función de restricción  $j$ .

$C_{\min}$ : Coeficiente de entrada, el valor absoluto del peor valor posible de la función objetivo aumentada  $Z_i - \sum_j \mu_j H_j$

Esta extensión soluciona el problema que ocurre durante las primeras generaciones donde las restricciones son violadas. Entonces funciona el algoritmo genético realizando selección, mutación y un cruce como operador genético, hasta alcanzar un número máximo de generaciones.

### 1.2.2. Control Adaptable

Se analizan ahora los enfoques que resuelven el problema adaptando los tiempos de verde asignados actualmente a las condiciones de tráfico identificadas por los detectores de tráfico.

#### Lógica Difusa

En [Sanal, 1999] se presenta un sistema de control de tráfico urbano basado en lógica difusa. El sistema difuso se encarga de determinar el incremento de tiempo de verde del semáforo, y a través de la evaluación de una función costo, se determina si se aplica este incremento. El sistema cuenta con sensores para recolectar los parámetros de control. Esta metodología permite tomar decisiones de acuerdo a como lo haría un operador humano. Los conjuntos difusos que presenta son: Tiempo desde el ultimo cambio(TLC), Número de Llegadas hacia la luz verde (ATG), Número de vehículos esperando en la cola (VWQ) y tiempo faltante para que las luces del semáforo cambien (ETC).

La decisión de extender el tiempo verde está basada en:

$$Q_A = G_A(\delta_{A_i}) - L_B(n_{B_j})$$

Donde:

$Q_A$ : Ganancia obtenida por la fase A.

$L_B$ : Correspondiente pérdida para B por haber hecho mas grande la fase A.

$\delta_{A_i}$ : Número de coches adicionales que pueden pasar la interseccion si el tiempo verde se incrementa.

$n_{B_j}$ : Número de coches en la cola de la fase B que sufrirán un incremento de retraso del tiempo incrementado a la fase A.

$$G_A = r_A[a_i \delta_{A_i}] + b_v \delta_{A_i}$$

$$L_B = \Delta T[a_i n_{B_j}] + b_v n_{B_j}$$

$r_A$  es el intervalo de tiempo entre el rojo y el verde para la fase A,  $a_i$  es el costo de retraso por segundo para los coches de la fase i y  $b_v$  son los costos de operación por detener un coche completamente y después llevarlo de nuevo a una velocidad normal.

La decisión de extender la fase A se hace con la siguiente regla:

$$Q_A = \begin{cases} \geq 0, & \text{extensión a la fase activa A} \\ < 0, & \text{activar fase B} \end{cases}$$

La principal desventaja de este enfoque es la dificultad que presentan los sistemas difusos para elegir las funciones de membresía adecuadas, además en este caso particular, se está considerando una intersección con únicamente 2 calles que la cruzan. Además el número de reglas a evaluar es muy elevada.

En [Khalid, 2004] se propone un control difuso de intersecciones que se pueden comunicar, administra las secuencias de fases y la longitud adaptable de fases. El control está dividido en 3 módulos. El primero que selecciona la fase necesaria para el siguiente período de tiempo verde y su urgencia; un segundo módulo que da como salida la posibilidad de que se extienda el tiempo verde de la fase activa; un tercer módulo de decisión donde se determina si se cambia la fase actual o se extiende el tiempo verde de la fase activa. El principal problema que se encuentra es que se pudieran tener cambios de secuencia de fases muy seguidos, ya que las mismas reglas se están utilizando para todas las fases, lo que puede llevar a que la red de tráfico se degenere a un estado en el cual el flujo de los vehículos sea muy lento o se detenga completamente en intersecciones vecinas.

### **Auto-Organización**

En [Gershenson, 2006] se presenta una alternativa donde los semáforos se auto-organizan para mejorar los flujos de tráfico sin necesidad de comunicarse unos con otros. El artículo presenta 3 métodos de control de semáforos adaptables. En el primer método, Sotl-request, se mantiene la cuenta del número de autos por unidad de tiempo que se aproximan a la luz roja, si este número alcanza un umbral, la luz se cambia a verde y la luz opuesta se cambia a rojo. En el segundo método, Sotl-phase, la luz del semáforo no cambiará si el tiempo que ha pasado es menor que un mínimo, regla que previene el cambio inmediato de luces en los semáforos. En el tercer método, Sotl-platoon, la luz roja no cambia a verde si hay pelotones que están cruzando la intersección; esta regla no se utiliza si hay un cierto número de autos acercándose a la intersección. La principal ventaja es que no se tiene que trabajar en métodos de comunicación; la desventaja es la elección de los parámetros adecuados para utilizarse en las reglas de los métodos, además de que no toma en cuenta la sobresaturación, con lo que se pueden alcanzar estados de bloqueo.

### **1.2.3. Coordinación con el concepto de reputación**

En [Bitting, 2004] se propone un control de intersecciones de tráfico con agentes que se encargan de estimar el tiempo de verde de la intersección de la que están a cargo. Cada uno tiene mecanismos para implementar racionalidad social. Cada agente tiene la capacidad de modelar estadísticamente la reputación de sus vecinos, la reputación está basada en 2 medidas, la primera es el flujo de tráfico que viene de ese vecino y la segunda es el historial de su voluntad para cooperar. La cooperación lleva a alta reputación, mientras una negativa a la cooperación lleva a una reputación baja. Con esto se da prioridad a los vecinos cuyos niveles de tráfico está incrementando. Existe una utilidad individual que resulta si se elige la propuesta del agente como la óptima, y una utilidad social, que es un resultado del incremento en el flujo de tráfico por



cooperar con un vecino, y del riesgo involucrado por el hecho de comprometerse a cooperar con un vecino. Los agentes utilizan una función de decisión que evalúa la utilidad al sumar sus utilidades individuales y sociales; en base a este cálculo deciden si cooperan.

#### 1.2.4. Control con Aprendizaje

Ahora se presentan los enfoques que resuelven el problema tratándolo como una situación de la cual se puede aprender para lograr mejores resultados.

##### **Sistemas Neuro-Difusos**

En [Chee, 2002] se presenta un sistema difuso-neural evolucionario con aprendizaje por refuerzo en línea para la administración y control inteligente del tráfico urbano. Existen 3 tipos de agentes en diferentes niveles de jerarquía, uno para cada intersección, otros de zona y de región. Los refuerzos se derivan de una estimación difusa de los cambio de estado entre intervalos de tiempo, estado positivo o negativo. Los ajustes de los pesos del sistema difuso-neural se lleva a cabo con la retro propagación del refuerzo. El agente utiliza reglas difusas para formular una política de control apropiada para la intersección que le toca controlar. El principal problema es la cantidad enorme de reglas a evaluar, que además se evalúa una vez por cada tipo de agente; además se hicieron pruebas únicamente en condiciones de tráfico fijas, simulando horas pico.

##### **Aprendizaje por Refuerzo**

En [Wiering, 2004] se presenta un algoritmo de aprendizaje por refuerzo para un sistema multiagentes para el control de los semáforos. El objetivo es minimizar la suma de los tiempos de espera de todos los autos que esperan en un semáforo. Cada auto estima el tiempo de espera hasta llegar a su destino si la luz del semáforo está en verde y también si está en rojo. Entonces se combinan las predicciones de todos los autos para tomar la decisión acerca del tiempo verde del semáforo. Para optimizar los ajustes del semáforo, se suman los tiempos de espera de todos los autos que esperan en una intersección y se toma la decisión que minimice el tiempo total de espera para toda la intersección. Para lograrlo se necesita haber aprendido una función valor que estima el tiempo de espera de cada auto hasta que llegue a su destino. Cada auto está en un nodo específico de tráfico (node), una dirección en ese nodo (dir), una posición en la cola (place), y tiene una dirección de destino (des). Para denotar la función valor cuando el semáforo tiene determinada luz se escribe  $Q([n, d, p, des], L)$ , y cuando no se conoce el color del semáforo se escribe  $V([node, dir, place, destination])$ .

Dada la situación actual de tráfico se toma la decisión de la intersección de la siguiente forma:

$$A_j^{opt} = \max \sum_{i \in A_j} \sum_{(n,d,p,des) \in queue} Q([n,d,p,des], red) - Q([n,d,p,des], green)$$

Para calcular las funciones Q y V se utilizan probabilidades de transiciones de estados, que se encuentran en una tabla que consta de  $P([node, dir, place, des], L, [new node, new dir, new place])$  donde L denota si la luz del semáforo para (node, dir) es verde o roja. También se calculan las probabilidades  $P(L|[node, dir, place, destination])$  que indican la probabilidad de que la luz del semáforo sea roja o verde para un carro que espera en (node, dir, place) con un destino particular. Finalmente se utiliza una función de recompensa  $R([node, dir, place], L, [node, dir, place]) = 1$ , si un carro permanece en el mismo lugar, de lo contrario  $R=0$ . Las desventajas que ofrece este enfoque es que instalar la tecnología necesaria para que los autos calcularan su tiempo de espera aproximado hasta su destino es muy cara, y por lo tanto muy difícil para lograr su implementación, además sería necesario que todos los autos conocieran su dirección destino.

### Razonamiento Basado en Casos

En [Xiao, 2004] se presenta un sistema de control de tráfico basado en sistemas multiagentes, que adopta la tecnología de Razonamiento basado en casos. Permite a los agentes aprender de los eventos pasados, encuentra una solución al nuevo problema al analizar los problemas previamente resueltos (casos), o adaptando viejas soluciones a nuevas demandas. El proceso se muestra en la Figura 1.1.

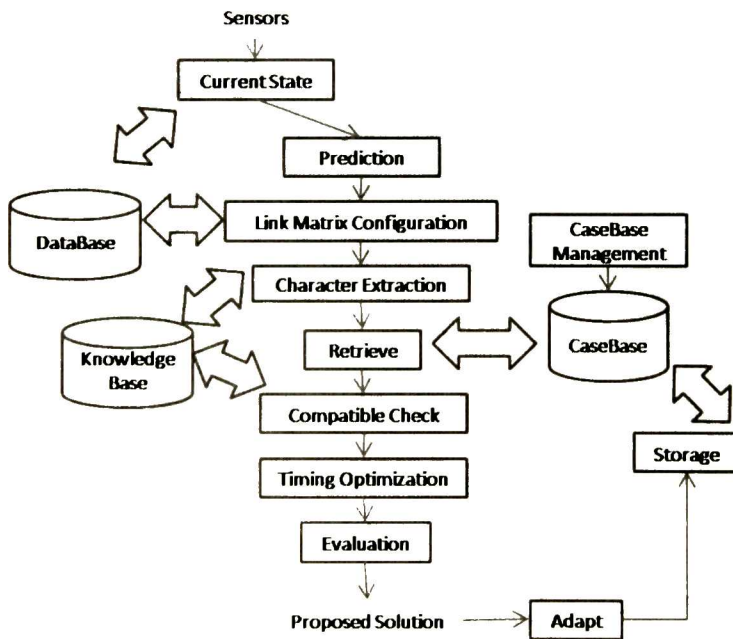


Figura 1.1 Procedimiento de los Agentes para resolver el problema de Control [Xiao, 2004].

A través de los sensores, los agentes se enteran de las condiciones de tráfico actuales. En base a éstas realiza una predicción del estado futuro del sistema. Después forma la matriz de enlaces donde indica las relaciones que tiene con otros agentes y el grado de saturación de las calles que éstos controlan. A continuación realiza una extracción de caracteres principales de las condiciones actuales, con los cuales intenta encontrar en la base de conocimientos un caso que sea muy parecido a las condiciones actuales y le hace una evaluación para saber si dará buenos resultados a través de simulación, si los resultados son buenos la solución se toma, de lo contrario se adapta la nueva solución y se almacena en la base de casos para su futura utilización.

### 1.2.5. Sistemas Multi-Agentes

Se presentan ahora los enfoques que resuelven el problema utilizando un sistema multi-agentes.

#### Solución Distribuida

En [Ossowski, 1998] se presenta una arquitectura basada en agentes para atender el problema de control de tráfico urbano, donde se resuelve el problema que se presenta cuando se intenta llevar a cabo la coordinación entre los agentes que controlan las intersecciones. Este problema va creciendo conforme crece el número de agentes. La propuesta elimina el agente coordinador y logra una coordinación emergente, como se muestra en la Figura 1.2.

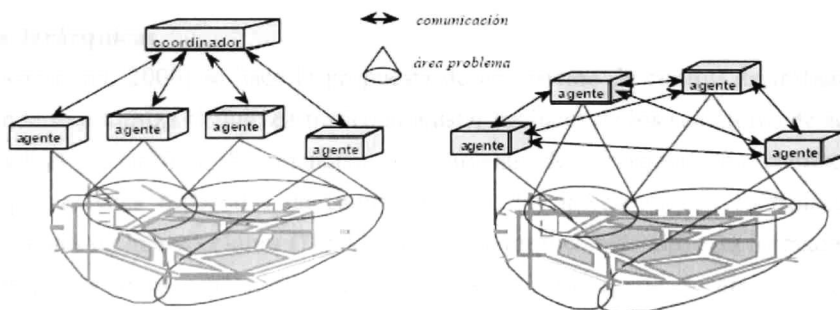


Figura 1.2 Sistema con coordinación centralizada y descentralizada [Ossowski, 1998].

La solución del problema y la coordinación se basan en un mecanismo llamado Cooperación Estructural, que ha sido desarrollado para lograr coordinación dentro de sociedades de agentes autónomos, donde la funcionalidad del agente se determina por los objetivos locales del agente, las relaciones de dependencia que su ambiente implica así como prescripciones normativas. El proceso para resolver el problema consta de tres etapas:

*Etapa Individual.* Los agentes generan sus propios objetivos locales de acuerdo a sus motivaciones, y cuentan con mecanismos de solución de problemas para lograr alcanzar estos objetivos.

*Etapa Social.* Se determinan las posibles interacciones entre las acciones para resolver el problema de los agentes. Entonces se busca un equilibrio de Nash.

*Etapa normativa.* El equilibrio que resulta de la etapa anterior no siempre corresponde a un comportamiento funcional a nivel sociedad; así que se diseñan prescripciones normativas que lleven el resultado de la interacción de los agentes a una dirección determinada.

### **Sistema de Reservación**

En [Dresner, 2004] se propone un sistema donde la intersección no cuenta con un semáforo. Los autos solicitan y reciben un lugar del cruce en un espacio de tiempo durante el cual pueden ocuparlo, es decir, lo reservan. Los automóviles pueden reservar los espacios que estarán necesitando en el momento en el que calculan que alcanzarán la intersección, la intersección está dividida como una celda de tamaño  $n \times n$ , donde  $n$  es la granularidad del sistema de reservación. Para poder realizar una reservación el auto envía un mensaje con los siguientes parámetros: la hora en que el auto llegará a la intersección, la velocidad con la cual llegará, su máxima velocidad, su máxima y mínima aceleración y sus dimensiones. De nueva cuenta la principal desventaja yace en la tecnología para tener el sistema de comunicación entre los autos y la intersección.

### **Arquitecturas Jerárquicas**

En [Roosmond, 2000] se hace la propuesta de un sistema de control de tráfico urbano inteligente, donde se optimiza el flujo de tráfico al ajustar las luces de los semáforos de acuerdo a la demanda actual de tráfico. Se manejan tres niveles de jerarquía: los caminos están representados por agentes especiales de camino (RSA), las intersecciones controladas están representadas por agentes de intersecciones (ITSA). El nivel jerárquico más alto está formado por dos tipos de agentes, para áreas específicas bien definidas está un agente de área y para rutas específicas puede haber agentes de ruta, que abarcan varios caminos.

El agente ITSA, se encarga de la distribución de datos a otros ITSA's y de la recolección de datos a través de los RSA's. También analiza la información y hace el cálculo de los tiempos verdes óptimos. Además opera las señales de acuerdo a la actual política de control. El agente ITSA también cuenta con un modelo de predicción que se encarga de predecir los futuros estados de las condiciones de tráfico. Con estos resultados tomará las decisiones acerca de la estrategia de control a seguir. Una desventaja es que el éxito del proceso depende en gran parte del modelo de predicción, así que si no se cuenta con una buena herramienta, las estrategias de control elegidas pueden ser inadecuadas.



En [Wang, 2005] se propone un controlador de tráfico donde un agente de tráfico es un anfitrión de diferentes agentes que residen en el controlador durante diferentes períodos de tiempo en respuesta a las diferentes demandas de tráfico. Entonces muchos agentes con diferentes tareas se encargan de la operación del controlador y se encuentran distribuidos en una red. Los DCA, agentes de control por default, residen en el controlador para asegurar su funcionamiento básico si la conectividad a la red no está disponible. El controlador hospeda a varios agentes ejecutores de control que manejan al controlador de acuerdo a las condiciones de tráfico.

Esto lo logran con un proceso que consiste de:

a). Análisis de la situación, donde se clasifican las condiciones de tráfico en casos predefinidos en los cuales los agentes pueden tomar decisiones. Todos los agentes ejecutores participan por el derecho a que sea su propuesta la que se despliegue en el semáforo.

b). Selección de propuesta, un árbitro determina cual agente gana, si se eligen varios agentes se llama a un algoritmo de fusión que combina todas las decisiones y forma una sola.

El árbitro también decide si los agentes ejecutores deben ser reemplazados por otros agentes. Los nuevos agentes de tráfico residen fuera del controlador en un RTOC, centro de operación de tráfico remoto. Este mecanismo se muestra en la Figura 1.3.

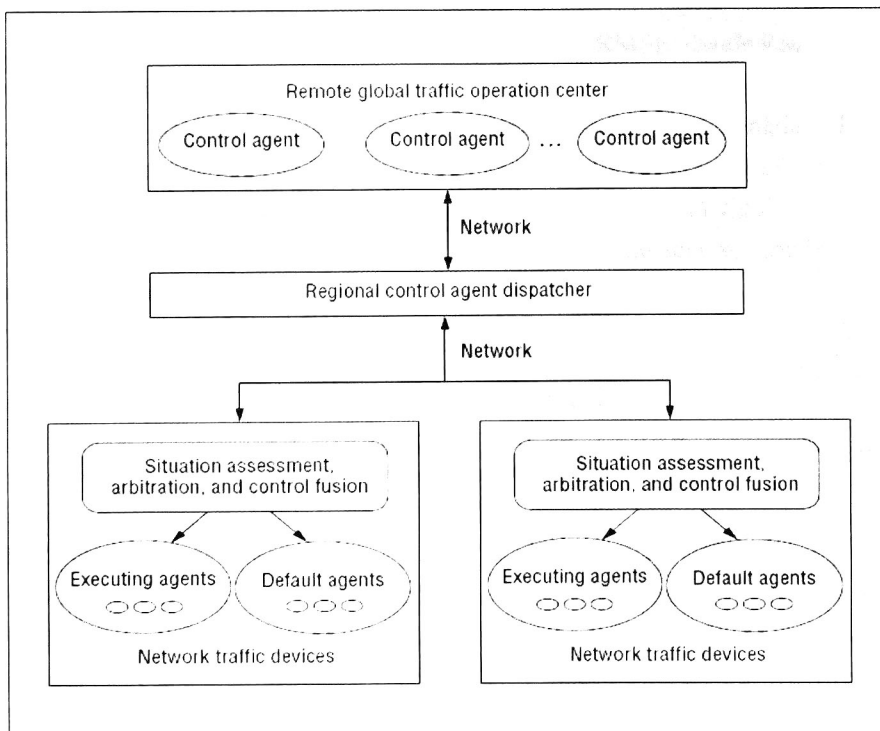


Figura 1.3 Control anfitrión para tráfico [Wang, 2005].

### 1.2.6. Predicción de Tráfico Urbano

En [Annunziato, 2003] se presentan algoritmos evolucionarios para optimizar en línea los pesos de una red neuronal de tipo feed-forward aplicada a la predicción corta de tráfico urbano. El sistema combina computación evolucionaria y redes neuronales. La idea básica es utilizar  $n$  muestras de datos de las condiciones de tráfico de la intersección como entrada a la red, de la forma  $x(t), x(t-T), \dots, x(t-nT)$ . En la Figura 1.4 se muestra la red que se utilizó como individuo en el algoritmo genético.

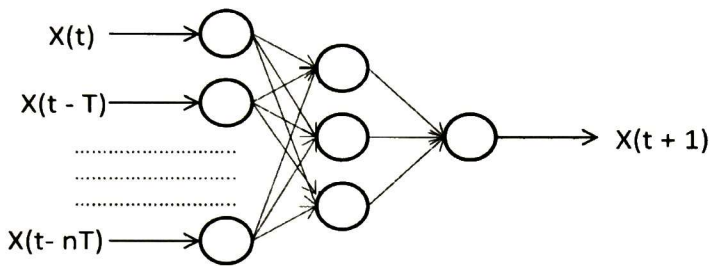


Figura 1.4 Arquitectura de red neuronal utilizada como individuo [Annunziato, 2003].

La aptitud se mide con la fórmula:  $\text{Fitness} = 1 - \text{RMSE}$ , donde RMSE es el clásico Error Cuadrático Medio.

En este trabajo se presenta también un nuevo algoritmo genético llamado Población Caótica, donde la selección se reemplaza por el concepto de “conocerse”. En cada iteración se elige un individuo de la población. La probabilidad de que dos individuos se conozcan se define con la densidad de población, entonces comienza la interacción, donde los individuos se reproducen y mutan.

## 1.3. Modelado de Sistemas de Tráfico Urbano

En esta sección se presentan los enfoques de modelado de sistemas de tráfico estudiados.

### 1.3.1. Redes de Petri Coloreadas

En [Dotoli, 2005] se presenta una forma de modelar sistemas de tráfico con Redes de Petri coloreadas, la finalidad de utilizar los colores es identificar los automóviles que viajan en determinada dirección, en la Figura 1.5 se muestra una intersección que después se modela con una Red de Petri coloreada de 5 colores, uno por cada uno de los movimientos permitidos en la intersección, de  $a_1$  a  $a_5$ . La Figura 1.6 muestra el modelo.



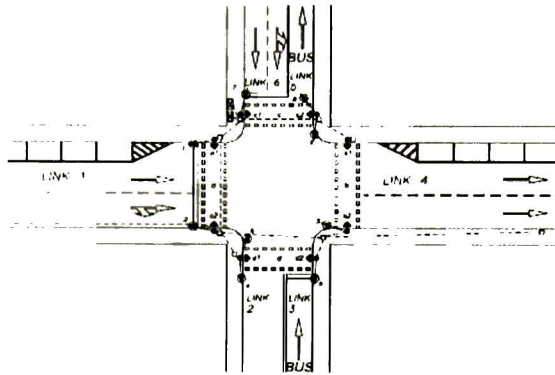


Figura 1.5 Intersección con 5 movimientos permitidos [Dotoli, 2005].

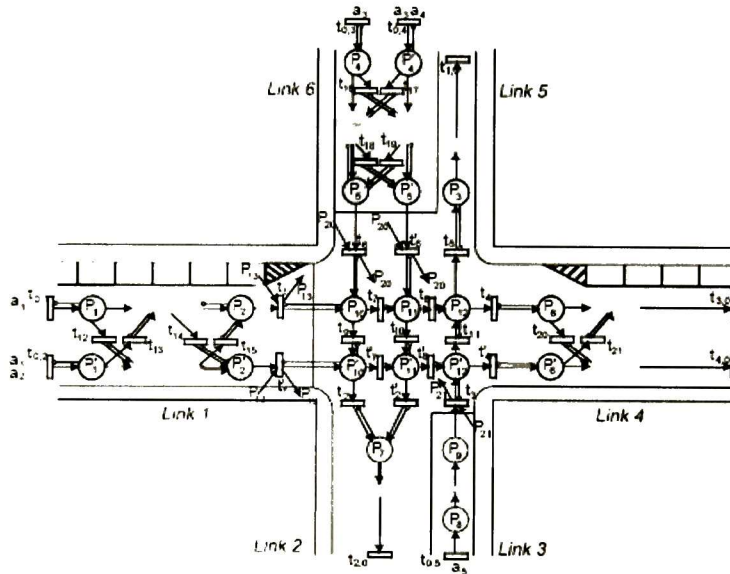


Figura 1.6 Modelo de una intersección con RPC [Dotoli, 2005].

La principal desventaja que tiene este modelo es que cada lugar discreto que se encuentra en la calle representa un espacio que puede ocupar un auto, por lo tanto tenemos una gran cantidad de lugares.

### 1.3.2. Redes de Petri Híbridas

En [Julvez, 2005] se presenta una forma de modelar sistemas de tráfico con Redes de Petri híbridas. El trabajo incluye un modelo de una intersección con 2 calles. Cada calle se modela con una Red de Petri continua y ambas calles se interconectan a través de una red de Petri discreta

temporizada que representa el semáforo, la intersección cuenta con 2 fases, que se modelan en la red de Petri discreta que se muestra en la Figura 1.7, donde:

- ggg: Carros en L1 cruzan la intersección.
- ggr: Carros en L1 se detienen.
- grr: Carros en L2 inician su avance.
- rrr: Carros en L2 cruzan la intersección.
- rrg: Carros en L2 se detienen.
- rgg: Carros en L2 inician su avance.

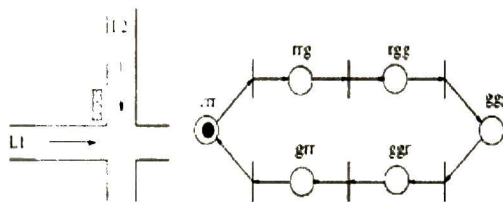


Figura 1.7 Red de Petri discreta modelando el semáforo en una intersección [Júlvez, 2005].

El modelo general de una intersección se muestra en la Figura 1.8.

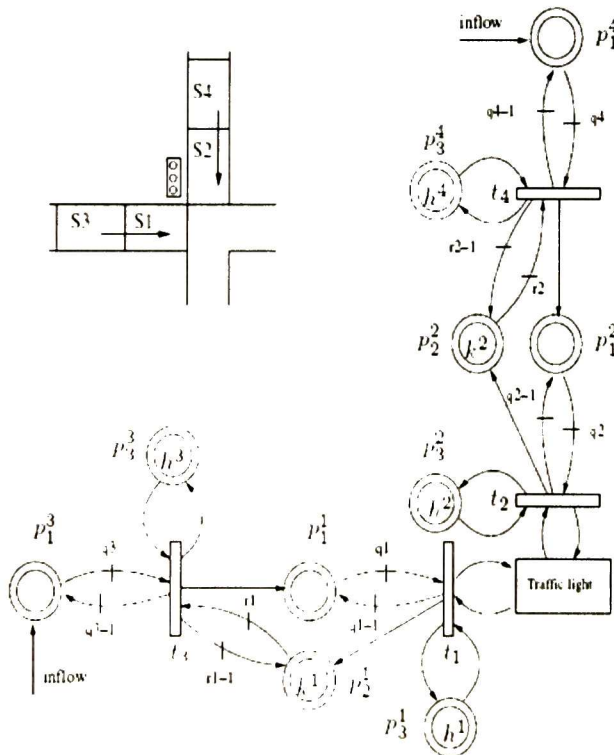


Figura 1.8 Red de Petri híbrida modelando una intersección. [Júlvez, 2005].

La principal ventaja de hacer el modelo del sistema de tráfico con una red híbrida es que se puede modelar el flujo de automóviles de una manera continua, aproximando lo que sucede en un sistema de tráfico real.

#### **1.4. Conclusiones**

De la revisión de la literatura aquí presentada, podemos hacer notar que el enfoque difuso para el control óptimo de los tiempos de verde es el más utilizado para el control de intersecciones. En cambio para asegurar un mejor desempeño de los controles de intersecciones a lo largo de las avenidas, la coordinación basada en Sistemas Multiagentes parece ser más apropiada. En este trabajo se adoptará el enfoque de Sistemas Multiagentes para la Coordinación de Tráfico Urbano. Se aplicarán dos técnicas para el control de intersecciones: Lógica Difusa y Programación Lineal. Para lograr la coordinación entre las diferentes intersecciones se utilizará la Teoría de Negociación de Nash.

# Capítulo 2

## Un Esquema Multiagentes para la Coordinación de Tráfico Urbano.

Resumen: En este capítulo se presenta brevemente la arquitectura del simulador de tráfico urbano presentado en [Neri, 2005a], al que se agregan los módulos de control y generación de vehículos, desarrollados en esta tesis.

### 2.1. Simulación de Tráfico Urbano

Describiremos el Simulador de Tráfico Urbano utilizado para evaluar el comportamiento del Control de Tráfico Urbano propuesto en esta tesis. Es un simulador que se encuentra actualmente en desarrollo, y que forma parte de un proyecto de Micro-simulación de Tráfico Urbano.

#### 2.1.1. Arquitectura del Simulador de Tráfico Urbano

El simulador actualmente en desarrollo es un modelo de simulación microscópico y utiliza la técnica de evento siguiente en la cual se tiene una lista de eventos futuros (FEL) donde se almacenan los eventos que habrán de ocurrir en la simulación. La implementación del simulador es realizada en el lenguaje de programación Java.

#### Características

La especificación del simulador cuenta con las siguientes características:

- a) Micro-simulador.
- b) Discreto.
- c) Estocástico.
- d) Basado en multiagente.
- e) Usa la técnica de avance de tiempo de evento siguiente.

Los cambios en el estado del sistema se hacen de forma discreta, entonces el simulador está especificado como un Sistema de Eventos Discretos. El modelo del sistema está basado en la metodología de modelado 3-LNS.

Cada vehículo es representado por un agente que se mueve a través de la red y es capaz de comportarse diferente de otros vehículos, a través del control de su velocidad y movimientos, estos de acuerdo a un plan de viaje. Para cumplir con los requerimientos de la micro-simulación, la red de tráfico está descompuesta en una colección de calles e intersecciones. Existen también otros agentes que administran los segmentos de calle y las intersecciones, estos agentes interactúan con los vehículos y con los semáforos.

### Funcionamiento general

Todos los tipos de eventos (llegadas de autos, cruce de una intersección, cambio de luz en el semáforo, etc.) son almacenados en una lista de eventos futuros (FEL). Cuando un evento ocurre, nuevos eventos son generados, afectando las diferentes calles e intersecciones. Estos eventos son introducidos ordenadamente en la FEL y la simulación continúa.

En la Tabla 2.1 se presenta una lista de los eventos que actualmente maneja el simulador y una breve descripción de cada uno de ellos.

Tipo de Evento	Descripción
AE	Llega un nuevo coche a la red a través del Módulo Generador de Autos.
SE	El vehículo realiza un alto total.
CE	El vehículo deja una calle o intersección.
LCE	Cambio de luz en el semáforo enviado por el Módulo de Control.
ALE	El vehículo llega a una calle o intersección después de haber realizado un CE.
BE	El vehículo arranca después de haber realizado un SE.
WSE	El vehículo realiza un alto de advertencia pues hay otros vehículos delante de él.
WBE	El vehículo arranca después de haber realizado un WSE.
BCE	Indica el final del tiempo de ciclo de una intersección y es recibido del Módulo de Control.

Tabla 2.1 Eventos manejados por el simulador

En la Figura 2.1 se presenta el algoritmo de control de la simulación, donde se calendarizan y ejecutan los eventos del sistema.

### 2.1.2. Generador de Vehículos

Como aportación inicial de esta tesis, se presenta un módulo Generador de Vehículos, encargado de mandar eventos de tipo AE al simulador. Los eventos pueden ser enviados a cualquiera de las calles fuente de la red de tráfico manejada por el simulador con una distribución de probabilidad de tipo Poisson.

Si  $X$  tienen una función de probabilidad de masa

$$p(k) = e^{-\lambda} \frac{\lambda^k}{k!}, k = 0, 1, \dots$$

Se dice que tiene una distribución de Poisson con parámetro  $\lambda > 0$ , y se escribe  $X \sim Poi(\lambda)$ . [Oloffson, 2005].

La distribución de Poisson tiende a presentarse cuando se cuenta el número de ocurrencias de algún evento impredecible en un período de tiempo. Ejemplos típicos son temblores, accidentes de autos, llamadas entrantes de celular, fallas de impresión en un periódico [Oloffson, 2005].

Si  $X \sim Poi(\lambda)$ , entonces

$$E[X] = \lambda \text{ y } Var[X] = \lambda.$$

donde:

$E[X]$  : Valor medio de  $X$

$Var[X]$  : Varianza de  $X$  [Oloffson, 2005].

El Módulo Generador de Vehículos puede realizar variaciones en la  $\lambda$ , tantas variaciones como se soliciten para el período de simulación. Para iniciar una simulación, se ejecuta primero la Máquina de Simulación y enseguida el Módulo Generador de Vehículos, que programa en la FEL todos los eventos de llegadas de autos a la red de tráfico de acuerdo a la  $\lambda$  especificada, con el tipo de evento AE. La simulación inicia hasta que todos los eventos de tipo AE han sido almacenados en la FEL. Con esto se logra la entrada de autos a la red de tráfico con un distribución de tipo Poisson.



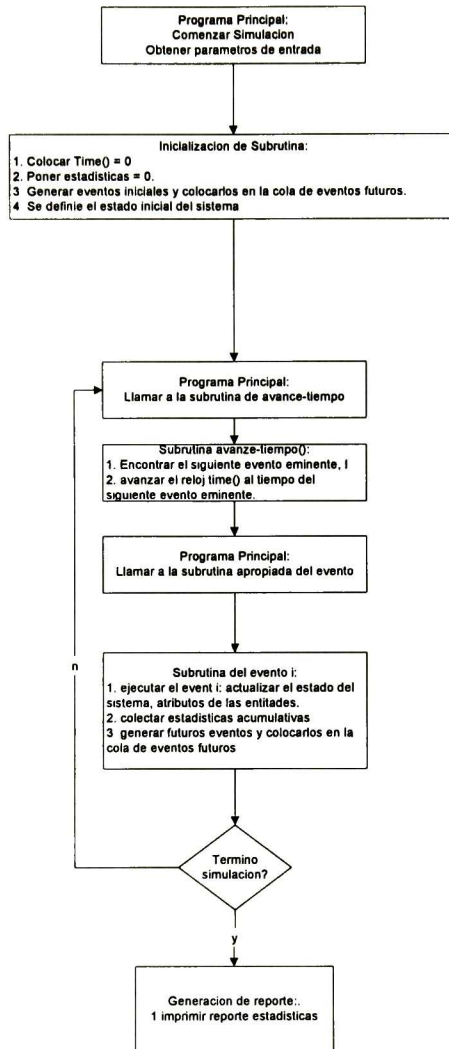


Figura 2.1 Algoritmo de Control de la simulación [Neri, 2005a]

### 2.1.3. Comunicación entre diferentes módulos del simulador.

El simulador tiene como objetivo representar un sistema de tráfico de tal forma que pueda simular el comportamiento del tráfico en forma real. Dado que un solo procesador resulta insuficiente para representar tanto el flujo vehicular como el control de tráfico, es necesario asignar a uno o más procesadores la tarea de simular el tráfico de una red y a otro procesador o más la tarea de controlar el flujo vehicular en la red. Con la finalidad de que los módulos puedan comunicarse entre sí y de que las distintas tareas se realicen en paralelo se ha establecido el

requerimiento de que se utilice un middleware para la comunicación entre los módulos implicados en la simulación [Trejo, 2006]. En la Figura 2.2 se observa el ambiente distribuido del simulador.

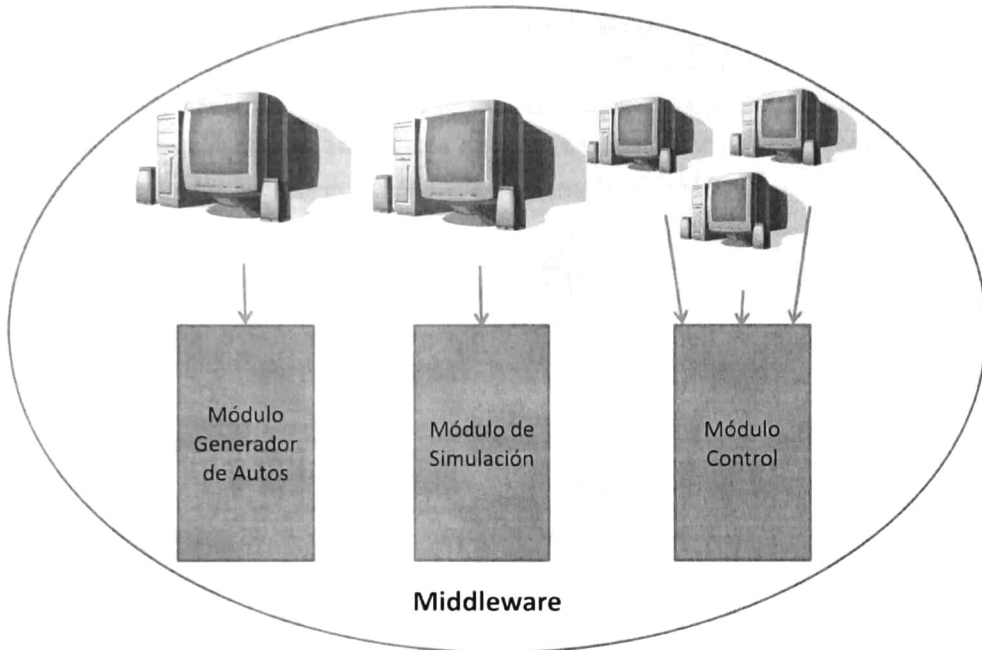


Figura 2.2 Ambiente distribuido del simulador

El control de tráfico propuesto en esta tesis puede ejecutarse en diferentes hosts, entonces la distribución de los módulos que se comunican con el simulador se realiza a través del middleware Proactive, que permite la comunicación eficiente entre estos.

El middleware ProActive es un conjunto de bibliotecas en Java para computación paralela, distribuida y concurrente; también se caracteriza por proveer movilidad y seguridad en un marco uniforme. Con un conjunto reducido de primitivas, ProActive permite programar aplicaciones que son distribuidas en una red de área local, en un cluster de ordenadores o en grids de Internet. ProActive está hecho solo con clases estándar de Java, y no requiere cambios en la máquina virtual [Trejo, 2006].

## 2.2. Integración del esquema de Control de Tráfico al simulador

El simulador de tráfico recibe los eventos de cambios de luz para las intersecciones del Módulo de Control propuesto en esta tesis, con lo que se pueden realizar pruebas del comportamiento del control propuesto con diferentes condiciones de tráfico, así como realizar comparaciones entre diferentes esquemas de control.

### 2.2.1. Interacción entre Simulador y Agentes de Control

El Módulo de Control consta de Agentes de Control encargados de enviar eventos de cambio de luz a los semáforos de la intersección de la que están a cargo. El simulador envía a cada uno de estos agentes la información necesaria acerca de las condiciones de tráfico de las calles. Una vez recibida esta información, cada agente de control realiza su proceso para decidir los tiempos de verde mas adecuados para cada una de las luces de los semáforos que tiene a cargo. Entonces calendariza los eventos de tipo LCE en la FEL. En la Figura 2.3 se observa la interacción entre los agentes de control y la máquina de simulación.

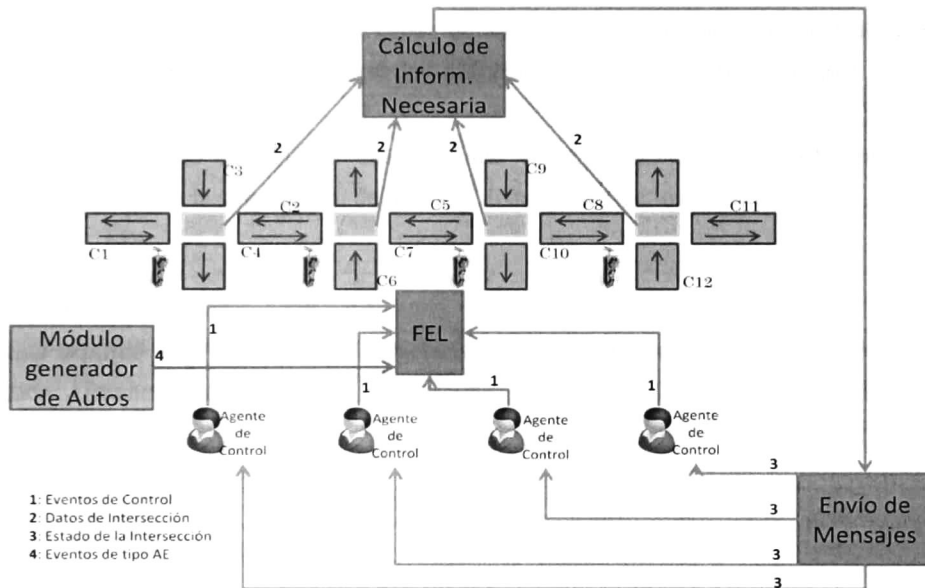


Figura 2.3 Interacción entre los agentes de control y la máquina de simulación

El simulador está diseñado de tal forma que la máquina de simulación es un despachador (dispatcher) al que se conectan los agentes de control, los cuales pueden estar en diferentes hosts. Inicialmente la máquina de simulación espera a que los agentes de control se conecten a ella; una vez que los agentes de control se han conectado, estos envían los primeros eventos de cambio de luz para que la simulación inicie [Trejo, 2006].

El intercambio de mensajes entre los agentes de control y la máquina de simulación se realiza a través de una primitiva propia de Proactive que se basa en el protocolo RMI (Remote Method Invocation).

### **2.2.2. Requerimientos del Módulo de Control**

Con la finalidad de evaluar los esquemas de control propuestos en esta tesis, se definen los siguientes requerimientos para la Máquina de Simulación:

- Contar con una plataforma de simulación que soporte comportamientos complejos (multiagente), que permita evaluar los esquemas de control propuestos.
- Poder implementar diferentes tipos de comportamientos de conductores de vehículos.
- Las pruebas deben realizarse en diferentes escenarios.
- La máquina de simulación será capaz de simular el comportamiento real de una red de tráfico de forma real, incluyendo las políticas de control establecidas por el agente de control.
- Existe un medio que brinda una eficiente conectividad entre los diferentes módulos involucrados en la simulación, debido a la distribución de los mismos.

### **2.3. Conclusiones**

En este capítulo se presentaron las características y el funcionamiento general del Simulador de Tráfico Urbano utilizado para realizar las pruebas de desempeño del Esquema de Coordinación presentado en esta tesis. Se presentó también la primera aportación de esta tesis, el Módulo Generador de Vehículos.

# Capítulo 3

## Un Esquema Multiagentes para el Control de Tráfico Urbano.

Resumen: En este capítulo se presenta un esquema multiagente para el control de tráfico urbano. En este esquema cada intersección es controlada por un agente de control; se describen todos los componentes de los agentes de control y la forma en que estos interactúan con el ambiente y entre ellos mismos. Se presentan de forma detallada los dos tipos de control que el agente puede realizar.

### 3.1. Agente de Control

En el esquema propuesto en esta tesis, cada intersección de la red de tráfico es controlada por un agente que determina los tiempos de verde de las fases de acuerdo a las condiciones de tráfico, con el fin de mejorar el flujo vehicular.

Cada agente se comunica con el simulador a través del paso de mensajes, recibiendo las condiciones de tráfico del simulador y enviando en respuesta eventos de cambios de luz para el semáforo que controla, estos eventos son ordenados dentro de la FEL.

Se presentan dos estrategias de control, uno utilizando lógica difusa y otro utilizando programación lineal.

Cada agente de control es capaz de negociar con el resto de los agentes para lograr un acuerdo en cuanto a los movimientos en los tiempos de verde para evitar causar problemas, especialmente de sobresaturación a las intersecciones vecinas. La negociación se logra utilizando la Teoría de Negociación de Nash, como se verá en el siguiente capítulo.

Las actualizaciones a los tiempos de verde las fases de la política de control se realizan tomando en cuenta un tiempo de verde máximo para cada calle, que será mencionado como



TVMAX. Este TVMAX representa el tiempo en el que pasa el número de autos que causa sobresaturación en la calle de abajo. De esta forma se evita que los agentes asignen tiempos de verde que causen sobresaturación.

### 3.1.1. Descripción del Agente de Control

Cada agente de control tiene un modelo de la política de control de la intersección especificado por una red de Petri temporizada en lugares.

El agente de control es un agente inteligente, pues tiene las características sugeridas por [Wooldridge, 2002] para serlo. Estas características son las siguientes:

**Reactivo.** El agente es capaz de percibir su ambiente y responder a los cambios que ocurren en el para poder satisfacer sus objetivos. El agente de control recibe información acerca del estado de la red de tráfico y entonces toma las acciones necesarias.

**Proactivo.** El agente es capaz de mostrar comportamiento dirigido hacia sus objetivos, al tomar la iniciativa para alcanzarlos. El agente de control lo hace al elegir los tiempos de verde adecuados de acuerdo a las condiciones de tráfico.

**Habilidad social.** El agente es capaz de interactuar con otros agentes para lograr sus objetivos. El agente de control negocia con los agentes vecinos para hacer cambios de tiempo de verde que no afecten las intersecciones vecinas.

El controlador es un agente de Proactive que opera de acuerdo a la situación actual del ambiente en la simulación, y ha sido desarrollado utilizando Java. El esquema propuesto provee un control descentralizado, ya que cada agente está encargado de realizar el control sobre la intersección a la cual está asignado.

### 3.1.2. Ambiente del Agente de Control

La red de tráfico se divide en secciones, la sección en la que está contenida la intersección que controla cada agente, representa su ambiente. El agente de control realiza dos tipos de control en su ambiente:

**Esquema de control fijo.** Asigna los tiempos a las fases de la política de control de acuerdo a tiempos predeterminados. El intervalo de tiempo asignado a cada fase en el esquema de control fijo se basa en patrones históricos de tráfico.

**Esquema de control adaptable.** Los tiempos asignados a las fases son establecidos de acuerdo a la demanda de tráfico. El agente de control observa las condiciones de tráfico y realiza los cambios necesarios a los tiempos de verde de las calles que tiene bajo su control.

### 3.1.3. Arquitectura Interna del Agente de Control

El agente de control está compuesto por varios módulos, cada uno realiza funciones específicas. Antes de iniciar la simulación se carga el Módulo Generador de Vehículos y después cada uno de los agentes de control. Inicialmente cada agente de control envía al simulador eventos de tipo LCE siguiendo un esquema de control fijo. Una vez que todos los cambios de luz han sido programados en la FEL, el simulador puede iniciar a ejecutar la simulación. Durante la simulación, el simulador ejecuta los eventos contenidos en la FEL; la ejecución de estos eventos provoca cambios en el estado de la red de tráfico. El simulador se encarga de enviar información acerca del estado actual de la red a los agentes de control, información como flujo, densidad, llegada promedio de coches, etc. Entonces el agente de control decide que tiempos de verde establecer de acuerdo a las condiciones de tráfico, generando nuevos eventos de tipo LCE que almacena en la FEL para su próxima ejecución. El agente de control negocia con otros agentes por medio del envío de mensajes para acordar tiempos de verde que no afecten a intersecciones vecinas. La Figura 3.1 muestra los componentes del agente de control.

La arquitectura propuesta consta de tres capas, la primera es la *capa reactiva*, que se encarga de implementar las políticas recibidas por parte de las capas de más alto nivel, monitorea sus acciones y pasa la información del estado del ambiente a las capas superiores. La segunda capa es la *capa de planeación local*, que decide que acciones se van a realizar, esto dentro de un ámbito local. La tercera es la *capa de planeación cooperativa*, que también decide sobre las acciones a realizar pero realizando interacciones con otros agentes.

Esta arquitectura es una extensión a la propuesta en [Trejo, 2006]. Se mantienen tres componentes: la Base de Datos, el Modelo de la política de control y el Operador de Semáforo. Se inicia la descripción de los componentes con ellos.

**Base de datos.** Contiene información sobre la estructura de la red de tráfico, tales como el número de calles involucradas, el número de carriles, los semáforos involucrados en la intersección, entre otras.

**Modelo de la política de control.** Debido a que las políticas de control pueden representarse como un sistema de eventos discretos, el modelo de las mismas puede ser representado por redes de Petri. El modelo que describe la política de control actual, el cual determina la operación de los semáforos en la intersección, está representado por medio de una red de Petri temporizada en lugares que especifica todos los movimientos que tienen derecho de paso en cada fase de la política de control, y la duración de cada fase.

**Operador del semáforo.** El operador del semáforo ejecuta el modelo de la política de control actual. La secuencia de eventos (disparo de transiciones) y los instantes de tiempo en que son disparados son determinados por la ecuación de estados de la red de Petri. Estos eventos son

traducidos en eventos significativos para el simulador, los cuales son agregados a la lista de eventos futuros.

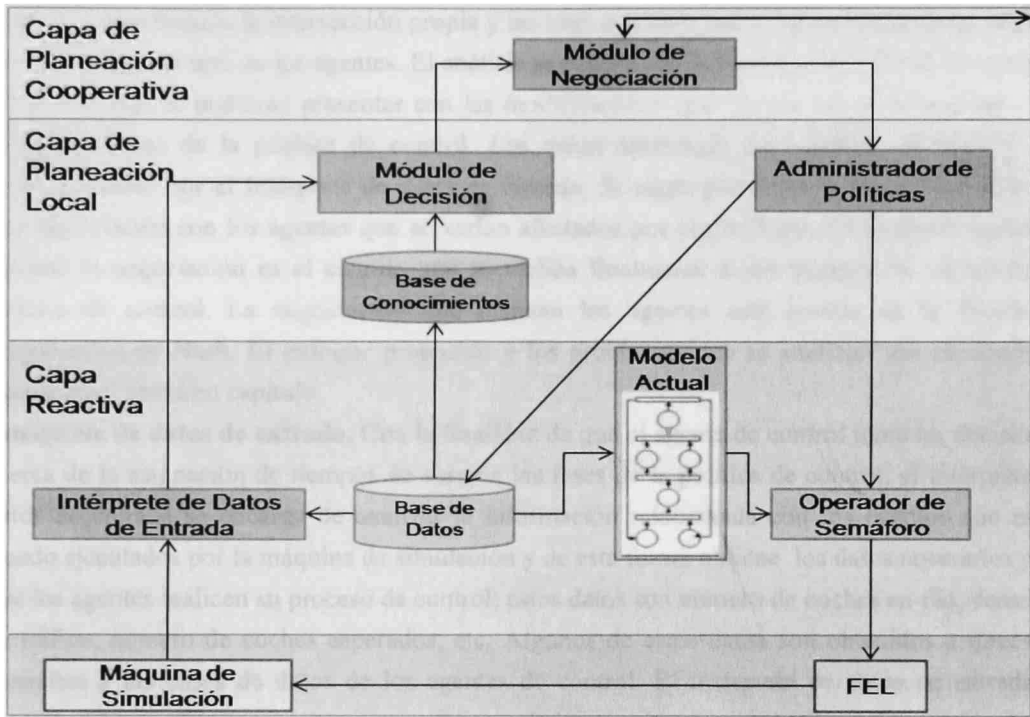


Figura 3.1 Arquitectura del Agente de Control

**Base de conocimientos.** Como se mencionó, se presentan dos tipos de control para los agentes. La Base de Conocimientos contienen las reglas que rigen las decisiones que tomará el agente de control acerca de los tiempos de verde asignados a la política de control. Estas reglas son de dos tipos, el primero contiene las reglas para el controlador difuso y el segundo contiene la definición del problema de programación lineal.

**Administrador de políticas.** Es el módulo encargado de realizar el cambio a las políticas de control, en este caso, a la temporización de los lugares de la Red de Petri. Una vez que los cambios de los tiempos de verde son realizados, se programan de nueva cuenta eventos en la FEL de la máquina de simulación para su próxima ejecución.

**Módulo de Decisión.** En el módulo de decisión, el agente de control se encarga de tomar las decisiones para actualizar los tiempos de verde de las políticas de control de la intersección que tiene a cargo. Se presentan dos formas en las que el agente puede tomar las decisiones; la primera, utilizando un sistema de inferencia difuso y la segunda, resolviendo un problema de programación lineal. El planteamiento de cada una de estas propuestas se presentan más adelante.



**Módulo de Negociación.** Una vez que el agente de control ha tomado la decisión acerca de la modificación de la temporización en la política de control, realiza un análisis del estado de la red de tráfico, que incluye la intersección propia y las intersecciones hacia las cuales se dirige el flujo de tráfico de cada uno de los agentes. El análisis se realiza con la finalidad de identificar posibles problemas que se pudieran presentar con las modificaciones que se han decidido realizar a las temporizaciones de la política de control. Los datos necesarios para realizar el análisis son proporcionados por el Intérprete de datos de entrada. Si algún problema es detectado, se inicia una negociación con los agentes que se verían afectados por el problema. El acuerdo alcanzado durante la negociación es el cambio que se realiza finalmente a los tiempos de verde de la política de control. La negociación que realizan los agentes está basada en la Teoría de Negociación de Nash. El enfoque propuesto y los problemas que se analizan son mostrados a detalle en el próximo capítulo.

**Intérprete de datos de entrada.** Con la finalidad de que el agente de control tome las decisiones acerca de la asignación de tiempos de verde a las fases de la política de control, el intérprete de datos de entrada se encarga de analizar la información relacionada con los eventos que están siendo ejecutados por la máquina de simulación y de esta forma obtiene los datos necesarios para que los agentes realicen su proceso de control; estos datos son número de coches en fila, densidad de tráfico, número de coches esperados, etc. Algunos de estos datos son obtenidos a través de consultas a las bases de datos de los agentes de control. El intérprete de datos de entrada se presenta en esta tesis como un componente más de la máquina de simulación. La ejecución de los eventos inscritos en la FEL representa el estado en el que se encuentra la red de tráfico durante un lapso de tiempo determinado. Periódicamente el simulador le envía al agente de control los datos obtenidos por el intérprete de datos de entrada, por lo que juega un papel esencial para el agente de control.

La estrategia del agente de control para cumplir sus objetivos se presenta en el Diagrama de Flujo de la Figura 3.2. Inicialmente, el agente de control programa en la FEL dos ciclos de Tiempo de Verde para la intersección. Una vez que la máquina de simulación ejecuta estos dos ciclos de control, envía a los agentes de control información relacionada con el estado de la red de tráfico. El módulo de decisión del agente de control decide que tiempos de verde son necesarios para aumentar el flujo de la red. Al terminar realiza un análisis de problemas, que se describe en el capítulo 4. Si el agente de control detecta algún problema, entonces el módulo de negociación inicia interacción con los agentes involucrados. Cuando la negociación finaliza se programa un ciclo de tiempo de verde para la intersección en la FEL. En caso de que el agente de control no haya detectado ningún problema, entonces programa un ciclo de tiempo de verde para la intersección en la FEL.



### 3.2. Control basado en Lógica Difusa

La principal tarea de un controlador es encontrar las acciones convenientes que puedan forzar al sistema a alcanzar un estado deseado. Un controlador difuso maneja problemas complejos de control a través de la lógica difusa. Las implementaciones de controles difusos son, en un sentido, imitaciones de las leyes de control que usan los humanos.

El esquema de control difuso presentado arroja como resultado aumentos de tiempo de verde necesarios para satisfacer las condiciones de tráfico actuales para cada una de las fases de la política de control.

#### 3.2.1. Variables Lingüísticas

Las variables lingüísticas de entrada a los sistemas de inferencia se muestran en la Tabla 3.1. Las variables de salida de cada una de las etapas de inferencia se muestran en la Tabla 3.2. En ambas tablas se detallan el significado, la forma en que se obtienen y los rangos de cada una de ellas. Los rangos presentados son válidos para el caso de estudio presentado en el capítulo 5.

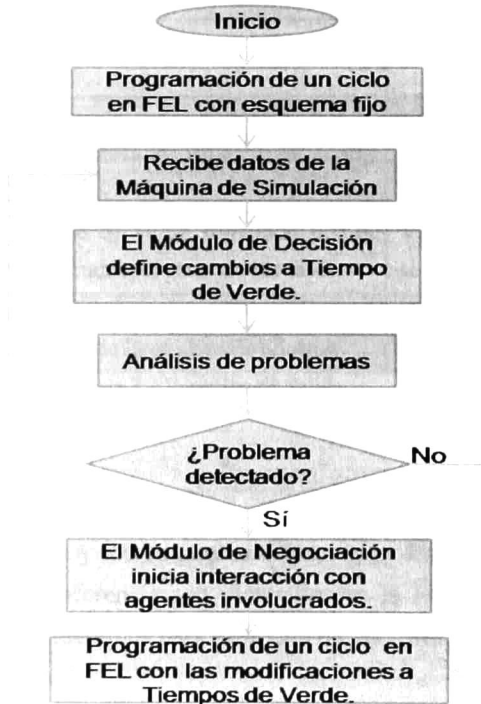


Figura 3.2 Actividades del agente de control.

Variable Lingüística	Significado	Cálculo	Rango
<b>Errori</b>	Diferencia entre la densidad actual de la fase i y un valor de referencia objetivo elegido, de acuerdo a la importancia de la fase.	$\text{Errori} = \text{Densidad}[i] - r_i$	$[-r_i, 50]$ , para todas las fases.
<b>Cambioi</b>	Diferencia entre el error actual y el error calculado en la etapa anterior de control.	$\text{Cambioi} = \text{Errori\_act} - \text{Errori\_ant}$	$[-50, 50]$ , para todas las fases.
<b>%TVi</b>	Porcentaje de tiempo de verde asignado a la fase i del tiempo de ciclo.	$\%TVi = (T_{vi} * 100) / \text{TCICLO}$	$[55, 85]$ $i = 1$ $[15,45]$ $i = 2$
<b>Sig_semi</b>	Densidad de las calles hacia las que se dirige el flujo de la fase i.	$\text{Sig\_Semi} = \max(\text{Densidad}[j])$ $j = \text{Índice de la calle hacia la que se dirige el flujo de la fase i.}$	$[0, 50]$ , para todas las fases.

Tabla 3.1 Variables Lingüísticas de entrada

Variable Lingüística	Significado	Cálculo
<b>Deltai</b>	Aumento de tiempo de verde solicitado por la fase i sin tomar en cuenta las calles hacia donde se dirige su flujo.	Variable de salida de la primera etapa de inferencia.
<b>Deltai'</b>	Aumento de tiempo de verde solicitado por la fase i sin tomar en cuenta el resto de las fases.	Variable de salida de la segunda etapa de inferencia.
<b>Deltai_def</b>	Aumento de tiempo de verde definitivo para cada una de las fases de la política de control.	Variable de salida de la tercera etapa de inferencia.

Tabla 3.2 Variables de Salida

### 3.2.2. Funciones de Membresía

Las funciones de membresía y los conjuntos difusos para las variables Errori y Cambioi utilizadas en la primera etapa de inferencia son mostrados en la Figura 3.3 y en la Figura 3.4 respectivamente. Estos grados son idénticos para cada uno de las fases de la política de control.

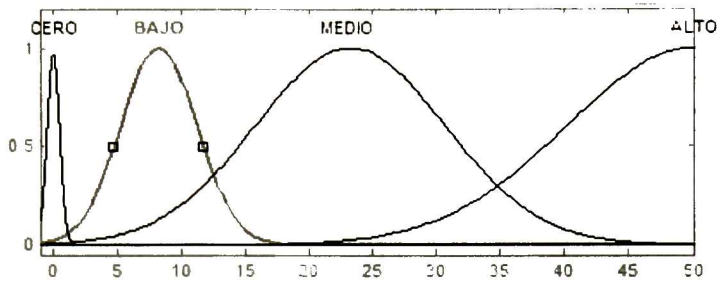


Figura 3.3 Funciones de Membresía para Errori.

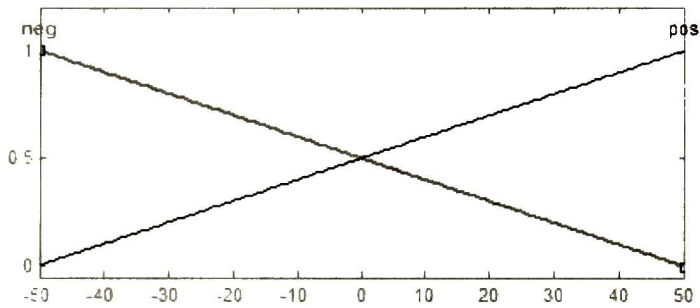


Figura 3.4 Funciones de membresía para Cambioi.

Las funciones de membresía y los conjuntos difusos para la variable Sig\_semi utilizada en la segunda etapa de inferencia son mostrados en la Figura 3.5. Estos grados son idénticos para cada uno de las fases de la política de control.

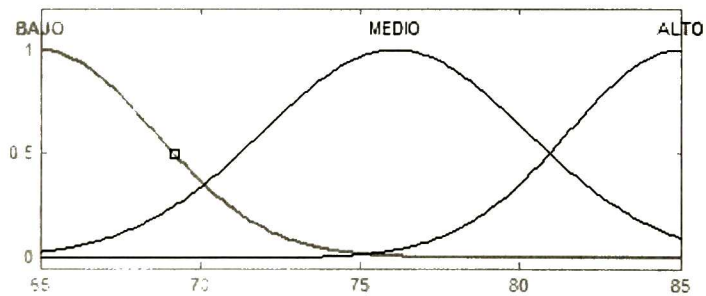


Figura 3.5 Funciones de membresía para Sig\_semi

Las funciones de membresía y los conjuntos difusos para el resto de las variables cambian de acuerdo a fase a la que hacen referencia, por lo que son mostrados en la Tabla 3.3. Estos valores son válidos para el caso de estudio presentado en el capítulo 5.

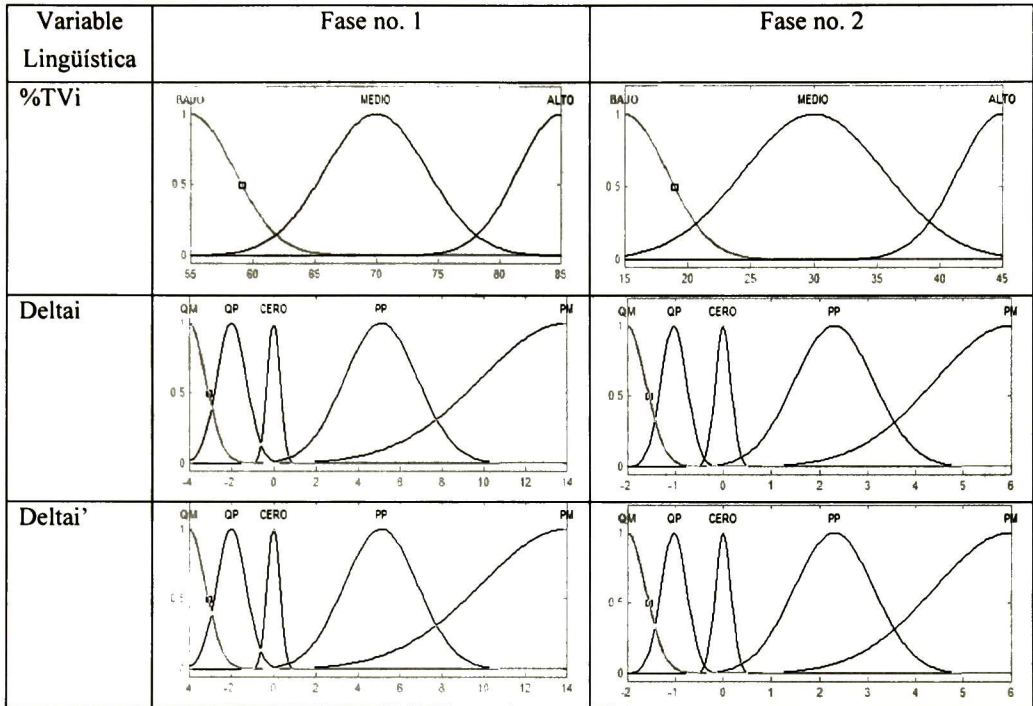


Tabla 3.3 Funciones de membresía

### 3.2.3. Estructura del Esquema Difuso

Se propone un esquema a tres niveles, el cual utiliza un sistema de inferencia difuso por cada fase de la política de control en cada una de los niveles. Los resultados de la inferencia del primer y segundo niveles son tratados por otro sistema de inferencia difuso para tomar la decisión final de los cambios en los tiempos del semáforo. Estos cambios son los que se toman en cuenta para analizar los problemas en la etapa previa a la Negociación.

El esquema propuesto está dividido en tres etapas para disminuir la cardinalidad de la complejidad del problema (Figura 3.6). En lugar de realizar un solo sistema de inferencia que evalúe todas las variables de entrada, un sistema de inferencia resuelve primero el problema localmente (por fase). Después otro sistema, tomando en cuenta las características de las calles hacia las que se dirige el flujo de la fase. En la última etapa se toman en cuenta las peticiones de todas las fases de la política de control.

Las salidas de cada uno de los sistemas de inferencia representan aumentos en tiempo de verde que las fases solicitan para poder alcanzar los valores de referencia establecidos. A continuación se detallan cada una de las etapas del esquema de control propuesto:



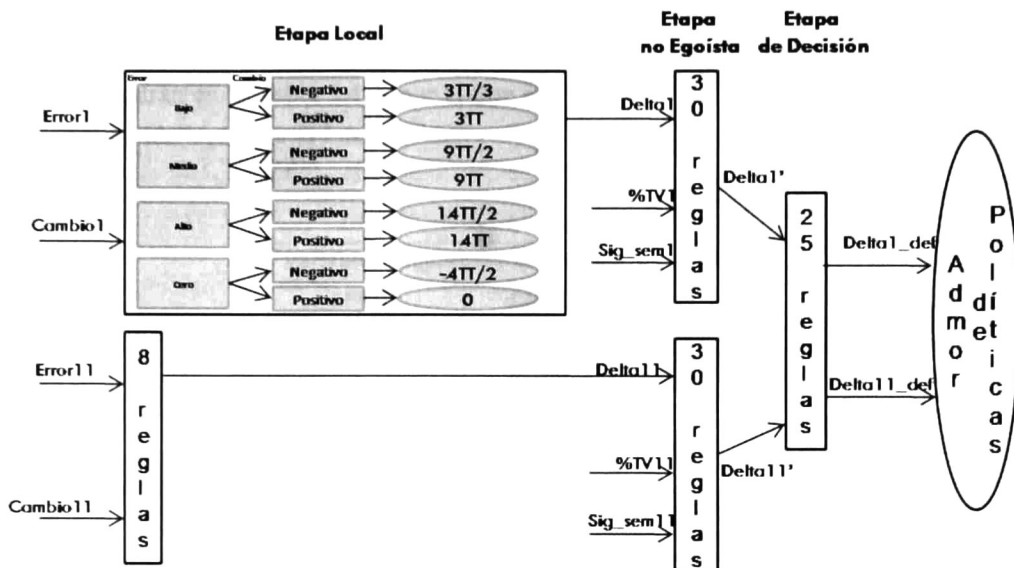


Figura 3.6 Esquema de Control Difuso

**Etapa Local.** En esta etapa se trabaja con un modelo Takagi-Sugeno y con un total de 8 reglas. Su operación es local pues el sistema de inferencia sólo toma en cuenta el estado ideal que desea alcanzar y el grado de aproximación a ese valor desde la última vez que se realizó el proceso de control. La salida de esta etapa representa el aumento en tiempo de verde que la fase desea obtener sin pensar en el resto de la red de tráfico.

**Etapa no Egoísta.** En esta etapa se tienen un total de 30 reglas. Esta etapa no es egoísta pues se toman como variables de entrada, el aumento de tiempo de verde deseado, el porcentaje de tiempo de verde del tiempo de ciclo que tiene asignada la fase y la densidad de las calles hacia las que se dirige el flujo de la fase. Con estos datos, no sólo se piensa en la fase local, sino que también se toman en cuenta las necesidades de las calles siguientes evitando sobresaturación; se evita que la fase local intente recibir la mayor parte del tiempo de verde del tiempo de ciclo.

**Etapa de Decisión.** En esta etapa se cuenta con un total de 25 reglas. Es la etapa final del esquema de control, pues tiene como variables de entrada las peticiones no egoístas de cada una de las fases de la política de control. La salida de esta etapa son los aumentos de tiempo de verde para cada una de las fases, cambios que son realizados a través del Administrador de políticas de control al modelo actual y a la base de datos.

### 3.2.4. Base de Reglas

A continuación se presentan las reglas de inferencia utilizadas por el agente de control para actualizar los tiempo de verde de las fases. Las reglas utilizadas en la etapa local por la fase no. 1 se muestran en la Tabla 3.4, y por la fase no. 2 en la Tabla 3.5. Se llama TT al tiempo que tarda un vehículo en cruzar la intersección.

Variables Lingüísticas de Entrada		Salida
ERROR	CAMBIO	
Bajo	Negativo	3TT/3
	Positivo	3TT
Medio	Negativo	9TT/2
	Positivo	9TT
Alto	Negativo	14TT/2
	Positivo	14TT
	Negativo	-4TT
	Positivo	0

Tabla 3.4 Reglas etapa local fase no. 1

Variables Lingüísticas de Entrada		Salida
ERROR	CAMBIO	DELTA
Bajo	Negativo	3TT/3
	Positivo	3TT * 0.6
Medio	Negativo	9TT/2 * 0.6
	Positivo	9TT * 0.6
Alto	Negativo	14TT/2 * 0.6
	Positivo	14TT * 0.6
	Negativo	-2TT
	Positivo	0

Tabla 3.5 Reglas etapa local fase no. 2

Las reglas utilizadas en la etapa no egoísta por la fase no. 1 y para la fase no. 2 se muestran en la Tabla 3.6. Las reglas son las mismas para ambas fases, la diferencia se encuentra en los conjuntos difusos presentados previamente.

Variables Lingüísticas de Entrada			Salida
%TV	SIG_SEM	DELTA	DELTA'
Bajo	Baja	QM	QP
		QP	QP
		C	C
		PP	PP
		PM	PM
	Alta	QM	QP
		QP	QP
		C	C
		PP	PP
		PM	PP
Medio	Baja	QM	QM
		QP	QP
		C	C
		PP	PP
		PM	PM
	Alta	QM	QM
		QP	QP
		C	C
		PP	C
		PM	C
Alto	Baja	QM	QM
		QP	QP
		C	C
		PP	PP
		PM	PP
	Alta	QM	QM
		QP	QM
		C	QP
		PP	QP
		PM	QP

Tabla 3.6 Reglas etapa no egoísta fase no. 1 y fase no. 2

Las reglas utilizadas en la etapa de decisión se muestran en la Tabla 3.7. En la definición de estas reglas se utilizó la función  $M$  tal que:

$$M : L \times C \times F \rightarrow \mathbb{R}$$

donde:

$L$  : Conjunto de variables lingüísticas.

$C$  : Conjunto que contiene a los conjuntos difusos de una variable lingüística.

$F$  : Conjunto de índices de las fases de la política de control.

La función  $M(l, c, f)$  devuelve el valor para el cual el conjunto difuso  $c$  de la variable lingüística  $l$ , obtiene un grado de pertenencia = 1 para la fase de control  $f$ .

Variables Lingüísticas de Entrada		Salidas	
DELTA1'	DELTA11'	DELTA_DEF1	DELTA_DEF11
QM	QM	0	0
	QP	0	0
	CERO	0	0
	PP	$M(QM,1)$	$-M(QM,1)$
	PM	$M(QM,1)$	$-M(QM,1)$
QP	QM	0	0
	QP	0	0
	CERO	0	0
	PP	$M(QP,1)$	$-M(QP,1)$
	PM	$M(QP,1)$	$-M(QP,1)$
CERO	QM	0	0
	QP	0	0
	CERO	0	0
	PP	$M(QP,1)$	$-M(QP,1)$
	PM	$M(QP,1)$	$-M(QP,1)$
PP	QM	$-M(QM,2)*2$	$M(QM,2)*2$
	QP	$-M(QM,2)*2$	$M(QM,2)*2$
	CERO	$-M(QM,2)$	$M(QM,2)$
	PP	$-M(QM,2)/2$	$M(QM,2)/2$
	PM	$-M(QM,2)/2$	$M(QM,2)/2$
PM	QM	$-M(QM,2)*3$	$M(QM,2)*3$
	QP	$-M(QM,2)*3$	$M(QM,2)*3$
	CERO	$-M(QM,2)*2$	$M(QM,2)*2$
	PP	$-M(QM,2)$	$M(QM,2)$
	PM	$-M(QM,2)$	$M(QM,2)$

Tabla 3.7 Reglas de decisión etapa de decisión

### 3.3. Enfoque de Programación Lineal

La programación matemática es una potente técnica usada en el proceso de toma de decisiones [Castillo, 2001]. La programación lineal se aplica a la optimización de una función



lineal mientras satisface un conjunto de restricciones lineales de igualdad/desigualdad [Bazaraa, 1977].

En esta sección se presenta el problema de actualización de tiempos de verde de las fases de una intersección como un problema de programación lineal. El agente de control resuelve el problema utilizando el Método Simplex. Como resultado se tienen aumentos de tiempo de verde necesarios para satisfacer las condiciones de tráfico actuales para cada una de las fases de la política de control de su intersección.

### Estructura del problema

El problema de programación lineal se plantea estableciendo los conjuntos de datos, variables y restricciones, así como la función a optimizar llamada función objetivo. A continuación presentaremos la estructura del problema de asignación de tiempo de verde a las fases.

- *Conjunto de datos:*

$F$  = Conjunto de índices de fases de la política de control.

$n$  = Número de fases en la política de control.

$d_j$  = Grado de importancia de la fase  $j$ .

$e_j$  = Número de coches que pueden dejar la intersección de la fase  $j$  en un segundo.

$q_j$  = Número de coches que están haciendo fila en la fase  $j$ .

$error_j$  = Diferencia entre densidad actual actual y deseada de la fase  $j$ .

$esp\_sig_j$  = Espacios disponibles en la calle hacia la que se dirige el flujo de  $j$ .

- *Conjunto de variables:*

$t_j$  = Número de segundos que se agregan a la fase  $j$                        $j = 1, \dots, n$ .

$t_j \geq 0$ .

$u_j$  = Número de segundos que se disminuyen a la fase  $j$ .                       $j = 1, \dots, n$ .

$u_j \geq 0$ .

- *Conjunto de restricciones:*

La primer restricción indica que los segundos que se aumenten al tiempo de verde de cualquier fase, debe ser igual al número de segundos que se disminuirán en el resto de las fases de la política de control.

$$t_j - \sum_{k \in F, k \neq j} u_k = 0 \quad j = 1, \dots, n.$$

La segunda restricción indica que el número de coches extra que podrán dejar la intersección debido al aumento del tiempo de verde, no debe exceder del error.

$$e_j t_j \leq error_j \quad j = 1, \dots, n.$$

La tercer restricción indica que el número de coches extra que podrán dejar la intersección debido al aumento del tiempo de verde, no debe exceder del número de espacios libres que se tienen en la calle siguiente.

$$e_j t_j \leq \text{esp\_sig}_j \quad j = 1, \dots, n.$$

• *Función objetivo:*

La función a optimizar es maximizar la diferencia entre el total de autos extra que podrán dejar todas las calles debido a los aumentos en tiempos de verde ( $\sum_{j=1}^{j=n} e_j t_j$ ) y los coches que perderán la oportunidad de dejar la intersección, por la correspondiente disminución de tiempo de verde ( $\sum_{k \in F, k \neq j} e_k u_k$ ). En otras palabras se busca maximizar el número de coches que dejarán las intersecciones. Para tomar en cuenta el grado de importancia de las fases y para que los aumentos de tiempo de verde se asignen en proporción a los coches que están haciendo fila de cada fase, estos valores son agregados a la función objetivo, quedando como se muestra a continuación:

$$\text{Max } z = \sum_{j=1}^{j=n} e_j d_j q_j t_j - \sum_{k \in F, k \neq j} e_k d_k q_k u_k$$

**Ejemplo**

Ahora se muestra el planteamiento del problema para una intersección, la cual forma parte de un caso de estudio más grande, el cual se presentará en el capítulo 5. El problema de programación será resuelto utilizando el Método Simplex.

La Figura 3.7 representa una intersección controlada por una política de control de dos fases, la primera de Norte a Sur, y la segunda, con movimientos Este-Oeste y Oeste-Este. La calle C1, representa la calle que tiene mayor flujo vehicular y por tanto, la fase que permite el flujo para esta, es la más importante. Revisemos como sería el planteamiento del problema de programación lineal para esta intersección.

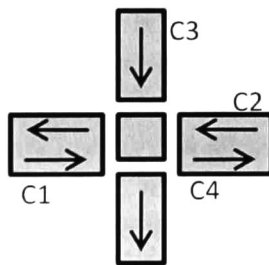


Figura 3.7 Ejemplo de una intersección controlada por un agente

*Conjunto de datos:*

$$F = \{1,2\}$$

$$n = 2$$

$$d_1 = 1 \text{ (fase más importante).}$$

$$d_2 = 0.5$$

$$e_1 = e_2 = 0.5 \text{ s.}$$

$q_j$  = Datos obtenidos durante la simulación.  
 $error_j$  = Datos obtenidos durante la simulación.  
 $esp\_sig_j$  = Datos obtenidos durante la simulación.

*Conjunto de restricciones:*

$$t_1, t_2, u_1, u_2 \geq 0$$

$$t_1 - u_2 = 0$$

$$t_2 - u_1 = 0$$

$$e_1 t_1 \leq error_1$$

$$e_2 t_2 \leq error_2$$

$$e_1 t_1 \leq esp\_sig_1$$

$$e_2 t_2 \leq esp\_sig_2$$

*Función objetivo:*

$$Max z = 0.5q_1 t_1 - 0.25q_2 u_2 + 0.25q_2 t_2 - 0.5q_1 u_1$$

### 3.4. Implementación del esquema de control

Se implementó un sistema multiagentes para el control de tráfico urbano, en donde cada agente está encargado de realizar el control de una intersección que le fue asignada. La Figura 3.8 presenta la arquitectura de software del esquema de control. El sistema no solo provee el control individual de cada intersección, sino que también permite que los agentes se coordinen para mejorar la fluidez del tráfico, como se verá en el capítulo 4. El agente de control se registra en el despachador, y después crea la Red de Petri que modela el control de los semáforos de la intersección. El agente ejecuta dos ciclos de la Red de Petri y envía los eventos a la FEL del simulador. Una vez que el simulador ejecuta un ciclo de la Red de Petri, entonces envía los datos del estado de la red de tráfico al agente. El agente realiza el control local, ayudándose de Matlab para realizar la inferencia difusa y para resolver el problema de programación lineal. El agente de control realiza los cambios arrojados a la temporización de la Red de Petri e inicia la identificación de problemas. Si el agente detecta algún problema inicia la interacción con otros agentes para negociar. El lenguaje de programación utilizado es Java, por ser un lenguaje de programación multiplataforma que permite la programación concurrente y distribuida a través de hilos, indispensable para la comunicación entre los agentes durante la negociación.

#### 3.4.1. Estrategia de Implementación

Con el fin de establecer la comunicación entre los módulos involucrados en la simulación se utilizó el middleware ProActive, el cual proporciona los protocolos necesarios para la

comunicación entre los agentes de control. La información almacenada en la base de datos debe ser administrada por un controlador de base de datos, el manejador de base de datos utilizado en la implementación es Microsoft Access; sin embargo, la migración a cualquier otro controlador de base de datos es posible.

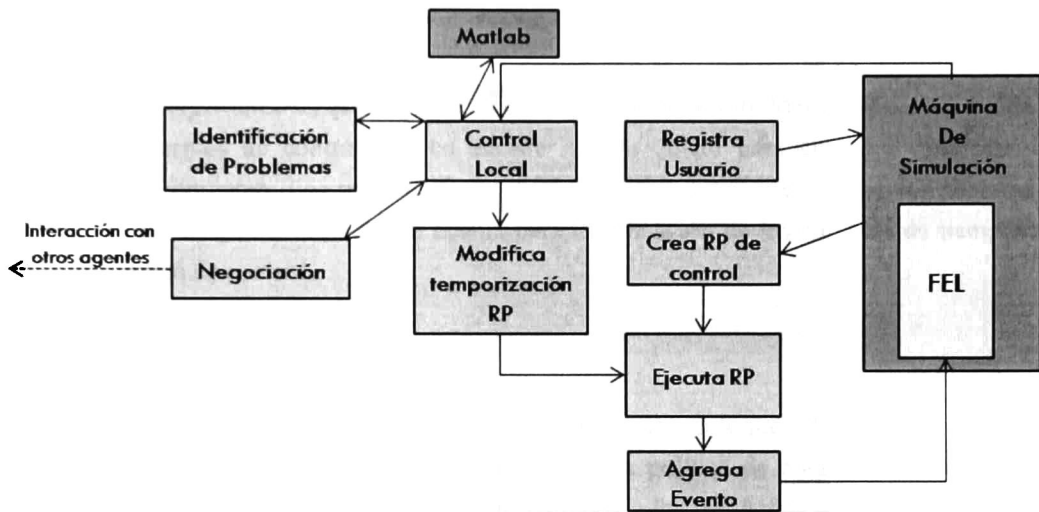


Figura 3.8 Diagrama de Arquitectura de Software del esquema de de control

La interacción entre la máquina de simulación y los agentes de control está activa durante todo el tiempo de simulación. La máquina de simulación envía a los agentes de control información referente a las condiciones de tráfico, y los agentes de control envían información referente a los cambios de luz que deben ser programados en la FEL. Inicialmente la máquina de simulación espera a que los agentes de control se conecten a ella; una vez que los agentes de control se han conectado, estos envían los primeros eventos de cambio de luz para que la simulación inicie.

Los mensajes que envía el agente de control hacia la máquina de simulación son enviados a través de la primitiva `addEvent()` propia de ProActive que se basa en el protocolo RMI (Remote Method Invocation). Los mensajes enviados por el agente de control contienen la estructura del evento de cambio de luz. En cuanto a los mensajes intercambiados entre los agentes durante la negociación, también son enviados a través de ProActive utilizando el protocolo RMI.

Debido a que durante el proceso de negociación, un agente es capaz de atender negociaciones en todas las direcciones hacia las que se dirige su flujo y además en las direcciones de las que recibe flujo vehicular, se utilizaron hilos para lograr que cada agente pudiera jugar el rol de Solicitante y Colaborador simultáneamente.



### **3.4.2. Comunicación entre Módulo de Control y Matlab**

Con la finalidad de que el agente de control realice los procesos de inferencia difusa y resuelva el problema de programación lineal, se hizo una interacción entre el lenguaje de programación Java y Matlab, utilizando la librería JMatLink. JMatLink permite utilizar el poder computacional de Matlab dentro de las aplicaciones de Java, utilizando las instrucciones nativas de Matlab.

Al inicio de la simulación se crea una conexión con Matlab a través de la creación de un objeto llamado CoreJMatLink, que crea una interfaz hacia el motor de Matlab. Una vez creada la conexión, los agentes de control pueden acceder a este objeto para enviar a ejecución las operaciones que requieren. Los resultados que arroja Matlab a estas operaciones son capturados en un buffer para luego ser tomados en cuenta para la aplicación de los cambios de tiempos de verde en la política de control.

### **3.5. Conclusiones**

Se presentó la arquitectura del agente de control, y dos enfoques de control para tomar decisiones acerca de la actualización a los tiempos de la política de control de acuerdo a las condiciones de tráfico. Ambos enfoques fueron definidos y los resultados de su implementación se muestran en el capítulo 5.

# Capítulo 4

## Coordinación de Semáforos

Resumen: En este capítulo se presenta la forma en que los agentes logran una coordinación en los semáforos que controlan. Se muestra el planteamiento del problema el protocolo de negociación entre los agentes.

### 4.1. Estrategia general de la Coordinación de Tráfico Urbano

El agente de control tiene la capacidad de comunicarse con otros agentes asignados en otras intersecciones. Cada vez que un agente realiza su proceso de control local efectúa un análisis de los posibles problemas que se pudieran presentar con esta nueva asignación de tiempos. Si el agente detecta un problema, inicia una negociación con los agentes que se verían afectados con estas nuevas asignaciones de tiempos. Una vez terminada la negociación se realizan los cambios al modelo de las políticas de control.

#### 4.1.1. Planteamiento del Problema

Se tiene un conjunto de intersecciones donde el control de semáforos de cada una de ellas es realizada por un agente de control. Después de analizar el estado de la red de tráfico, si el agente detecta un problema inicia una negociación con el agente de la calle de abajo.

La detección de los problemas se realiza en todas las direcciones de las calles de la intersección que controla el agente y la comunicación se realiza directamente con los agentes encargados de las intersecciones vecinas. Llamamos intersecciones vecinas, a las intersecciones que se encuentran a un segmento de calle de distancia en todas las direcciones. El agente puede iniciar entonces dos negociaciones en cada una de las direcciones de su flujo, como se muestra en la Figura 4.1.

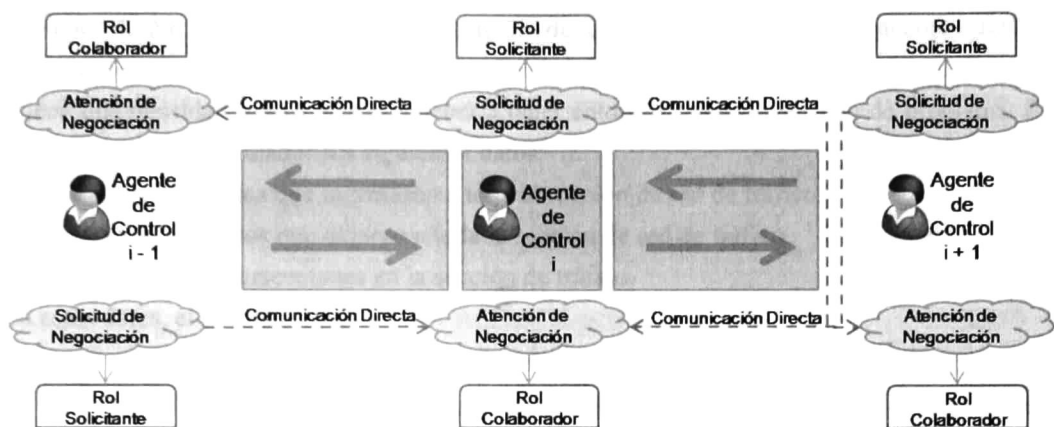


Figura 4.1 Roles del agente de control

Un agente no puede atender dos negociaciones en la misma dirección al mismo tiempo, es decir, no puede jugar el mismo rol con más de un agente al mismo tiempo. Si un agente A desea negociar con un agente B, que está participando en una negociación, entonces tendrá que esperar a que B termine su negociación. En caso de que el agente haya realizado más de una negociación, y que estas negociaciones afecten a la misma fase de la política de control, entonces el cambio que se realiza al modelo actual con el menor aumento de todos los acordados.

Los acuerdos alcanzados son válidos hasta que los agentes involucrados participen en nuevas negociaciones. Para evitar que la actualización de tiempos de verde en las políticas de control afecten a otras intersecciones, todas las decisiones tomadas durante la negociación son basadas en los TVMAX propios. El escenario de negociación es orientado a valor y será presentado en la sección siguiente.

#### 4.1.2. Análisis de problemas

El agente realiza el análisis de problemas en todas las direcciones hacia las que se dirige su flujo vehicular. El análisis lo realiza a través del estado de la red de tráfico para detectar tres tipos de problemas que a continuación se detallan. El primer tipo de problema se resuelve localmente, el resto de los tipos de problema, en caso de ser detectados llevan al inicio de una negociación. Para los tres tipos de problema denotaremos:

A = Agente que está realizando el análisis.

B = Agente encargado de la intersección de la calle de abajo (downstream).

C = Agente encargado de la intersección de la calle de arriba (upstream).

**Problema 1. No se mantiene la distribución de coches.** En este caso el agente calcula el promedio de autos que debería tener en su calle de acuerdo al total de autos en la red de tráfico. Si tiene una densidad mayor a la que debería tener entonces detecta este tipo de problema. Para lograrlo, recibe del simulador los siguientes datos:

a = Total de autos que ingresaron a toda la sección de red de tráfico.

b = Total de autos que dejaron a toda la sección de red de tráfico.

c = Total de intersecciones en la sección de tráfico.

Con estos datos, el agente calcula:

$$e = (a - b) / c$$

$$S = d - e$$

donde:

d = Es la densidad local después de su tiempo de verde.

S = Es el número de autos extra que la intersección tiene con respecto a los que debería tener si la densidad vehicular estuviera uniformemente distribuida.

Si S es positivo, y el aumento de S segundos a la fase respectiva en el modelo actual no sobrepasa el tiempo de verde máximo TVMAX, entonces se actualiza el modelo actual. En caso de que se sobrepase el TVMAX, entonces el modelo actual se actualiza al TVMAX.

**Problema 2. Sobresaturación.** Una vez que el agente ha determinado las modificaciones que realizará a los tiempos de verde de la política de control, verifica que el número de autos que podrá retirar durante su tiempo de verde no cause sobresaturación en las calles hacia las que se dirige su flujo vehicular. Para lograrlo, recibe del simulador los siguientes datos:

f = Número de autos promedio en la calle de abajo (B).

g = Número de autos que la calle de abajo (B) podrá retirar durante su TV.

Con estos datos, el agente determina:

$$S = g - (h + f)$$

donde:

h = No. de coches que la intersección A puede retirar durante su TV en la dirección que está analizando.

S = Representa el número de autos que las intersecciones vecinas enviarán B, y que éste no podrá liberar durante su tiempo de verde.

Si S es negativo, entonces el A inicia una negociación con B, donde le solicita que aumente su TV en la cantidad necesaria para retirar a |S| autos.

**Problema 3. El no. de autos que dejan la calle es menor a no. de autos que llegan.** En este tipo de problema, el agente compara el número de autos que dejan su calle y el número de autos que entran. Para lograrlo, recibe del simulador los siguientes datos:

m = Número de autos que no están en fila.

n = Número de autos que entraron a la intersección.

o = Número de autos que dejaron la intersección.

t = Espacio de tiempo de verde durante el cual no salió ningún auto.

Con estos datos, y en el caso de que t sea positivo, el agente determina:

$$S = (n - m) - o$$

Si S es positivo, entonces tenemos un problema pues la calle de A tiene tiempo de verde suficiente, mas no se han logrado retirar todos los coches debido probablemente a sobrecargas en las calles vecinas. Entonces A inicia una negociación con el agente B, donde le solicita que aumente su TV en los segundos necesarios para retirar a S autos.

En el caso de que el agente detecte los tipos de problema 2 y 3 en la misma dirección, realiza únicamente una negociación con el tipo de problema que arrojó la S con valor más alto.

## 4.2. Teoría de Negociación de Nash

Cuando es necesario manejar interdependencias entre las actividades de los agentes, es necesario utilizar un mecanismo de coordinación. La negociación se utiliza para alcanzar acuerdos en asuntos de interés mutuo [Wooldridge, 2002]; se trata de una forma de proceso de toma de decisión donde dos o más partes conjuntamente buscan un espacio de posible solución, con el objetivo de llegar a un acuerdo [Rosenschein, 1994]. Cada vez que un agente de control requiere actualizar los tiempos de verde de la política de control, verifica antes de realizar los cambios si esta decisión no causa ningún problema a las intersecciones vecinas. De ser así trata de alcanzar un acuerdo con los agentes vecinos para mantener un equilibrio en la red de tráfico. Como los agentes pueden no tener los mismos objetivos y tienen que llegar a un acuerdo, estamos hablando de un escenario de Negociación. Como propuesta de esta tesis se presenta un esquema de coordinación de tráfico urbano como un escenario de Negociación.

Una vez que el agente ha detectado un problema, entonces inicia una negociación. El protocolo de negociación propuesto es descentralizado, es decir, no existe un coordinador central que administra los acuerdos conforme a un objetivo común grupal. En el mecanismo propuesto, los acuerdos son alcanzados con las interacciones que realizan los agentes, el resultado es un compromiso. El mecanismo está basado en la Teoría de Negociación de Nash. A continuación se presenta el escenario de Negociación.



#### 4.2.1. Dominio orientado a valor de la negociación

El dominio de la negociación es orientado a valor, pues los objetivos de los agentes son especificados al definir una función valor que asigna una utilidad a los posibles estados alcanzables del ambiente. El objetivo de los agentes es llegar al estado del ambiente que tenga el mayor valor.

A continuación se presenta el escenario de negociación con dominio orientado a valor utilizando teoría de juegos. Tenemos un juego cooperativo donde si los agentes pueden obtener beneficio de la cooperación, no dudarán en hacerlo. El dominio orientado a valor presentado a continuación presenta una variación con respecto al definido en [Rosenschein, 1994].

Formalmente el dominio orientado a valor utilizado es una 5-tupla:

$$\langle E, Ag, D, c, w \rangle$$

donde:

E: Conjunto de posibles estados del ambiente.

Ag: Conjunto de posibles jugadores.

D: Conjunto de posibles deseos de los jugadores, expresados en estados del ambiente.

c:  $E \times Ag \times D \rightarrow \mathbb{R}$ ;  $c(e, i, d)$ , Costo para el jugador  $i$  de llegar al estado  $e$  cuando desea  $d$ .

w:  $E \times Ag \times D \rightarrow \mathbb{R}$ ;  $w(e, i, d)$ , Valor del estado  $e$  para el jugador  $i$ , utilizada cuando desea  $d$ .

#### 4.2.2. Modelos de cooperación y conflicto

Utilizando la función valor  $w$ , cada agente asigna un valor al estado del ambiente indicado durante la negociación. Asignamos una utilidad máxima para todos los agentes  $U_{max} = 10$ .

$$w(e, i, d) = U_{max} - c(e, i, d)$$

Utilizando la función costo  $c$ , cada agente puede asignar costos a los estados alcanzables de acuerdo a sus deseos, y se expresa por la distancia entre el estado indicado y el estado que el agente desea alcanzar. Los deseos de los agentes cambian conforme avanza la simulación.

$$c(e, i, d) = d(e, d)$$

Tomando en cuenta las funciones anteriores, se puede formar un vector de utilidades que contenga el deseo del agente solicitante de la negociación, otro que contenga el deseo del agente colaborador de la negociación y un tercer caso donde no se aplican cambios. El conjunto formado por los vectores de utilidad de todos los agentes es  $U(J)$ .

Se presentó el modelo para expresar las preferencias de los agentes hacia los estados del ambiente de acuerdo a sus deseos. Los agentes coordinan sus estrategias y seleccionan como

resultado un vector de utilidades correspondiente, los valores contenidos en el vector seleccionado son precisamente los que esperan obtener. Aún así, es posible que los agentes no estén interesados en cooperar, entonces ahora se modela la utilidad en caso de que los agentes decidan no cooperar, esto es, la utilidad de conflicto.

Para encontrar el vector de conflicto se asume que el resto de los jugadores eligen el vector de utilidades que minimiza la utilidad del jugador conflictivo. El valor resultante de la selección de ese vector de utilidades para el agente conflictivo  $i$  es la utilidad de conflicto:  $U_{c_i}$ . Se forma entonces el vector de utilidades en caso de desacuerdo:

$$d = (U_{c_1}, \dots, U_{c_n}) \quad n = \text{Número de jugadores.}$$

#### 4.2.3. Escenario de Negociación y solución de Nash

Ahora se define un escenario de Negociación con los elementos previamente presentados. Para esto tomamos a el vector de utilidades en caso de desacuerdo,  $d$ , como una solución alcanzable, entonces se define un conjunto  $S$  de soluciones alcanzables tal que:

$$S = U(J) \cup d$$

El escenario de negociación  $B$ , asociado a nuestro problema de tráfico es el par:

$$B = (S, d)$$

Ahora que se tiene el problema de Negociación, es necesario encontrar la solución. Esto se logra utilizando la Teoría de Negociación de Nash, en concreto encontrando la solución a :

$$f = \max \prod_{i=1}^{i=n} (s_i - d_i)$$

Debido a las características que tiene la solución de Nash y que se presentan en el Apéndice A, esta solución garantiza que las utilidades de los agentes alcanzando un acuerdo son mayores que si decidieran entrar en conflicto, con lo que se obliga a todos los agentes a cooperar.

#### 4.3. Protocolo de Negociación

Inicialmente el agente realiza el proceso de control local y después analiza los problemas, como se ha definido en las secciones previas. Si el agente detecta un problema, solicita ayuda. El agente  $A$  que detecta un problema y por consecuencia inicia la negociación será llamado Solicitante, y el agente  $B$  que da respuesta a esa solicitud será llamado Colaborador, pues son estos los roles que jugarán durante la negociación. Cuando el Solicitante no recibe respuesta indicando disposición de ayuda por parte del Colaborador en más de una ocasión, entonces el agente  $A$  no vuelve a solicitarle ayuda hasta después de un cierto número de tiempos de ciclo. Si el colaborador no está ocupado, envía respuesta indicando disposición a ayudar al Solicitante,

quien le envía su caso ideal, (número de segundos que el agente A desea que el agente B aumente a su TV para solucionar el problema detectado), para la negociación. En respuesta, el colaborador responde a ese caso ideal y a un caso donde no se realiza ningún cambio con la función valor. Además crea su propio caso ideal, todos los valores son enviados al solicitante.

Cuando el colaborador no puede cumplir con las peticiones del solicitante, éste comienza a jugar un rol de Solicitante e intenta negociar con el agente a cargo de la intersección siguiente. Este es el caso donde se propaga la negociación hasta el final de la sección o hasta que haya un respuesta positiva. Entonces se hace una retro propagación de las negociaciones hasta llegar a enviar el caso ideal por parte del colaborador inicial al solicitante inicial.

Cuando el solicitante recibe el caso ideal, le asigna una utilidad y entonces resuelve el problema de Negociación. Si la solución al problema de Negociación genera cambios en el ambiente, entonces el agente solicitante envía los tiempos de verde a los cuales el agente colaborador debe actualizar sus políticas de control. El agente colaborador aplica los cambios, envía una confirmación al agente solicitante y termina la Negociación. En caso de que la solución al problema de Negociación no genere cambios, la negociación termina.

En la Figura 4.2 se presenta un modelo de el protocolo de negociación con una Red de Petri Coloreada (RPC). Las RPC son un lenguaje para el modelado y validación de de sistemas en los que la concurrencia, comunicación y sincronización juegan un papel muy importante [Jensen, 2007]. El modelo se realizó con la herramienta CPN Tools, que permite construir y analizar modelos de RPC. Además, permite mediante simulación observar el comportamiento del sistema modelado. El modelo presentado representa el comportamiento que tienen cada uno de los agentes de control.

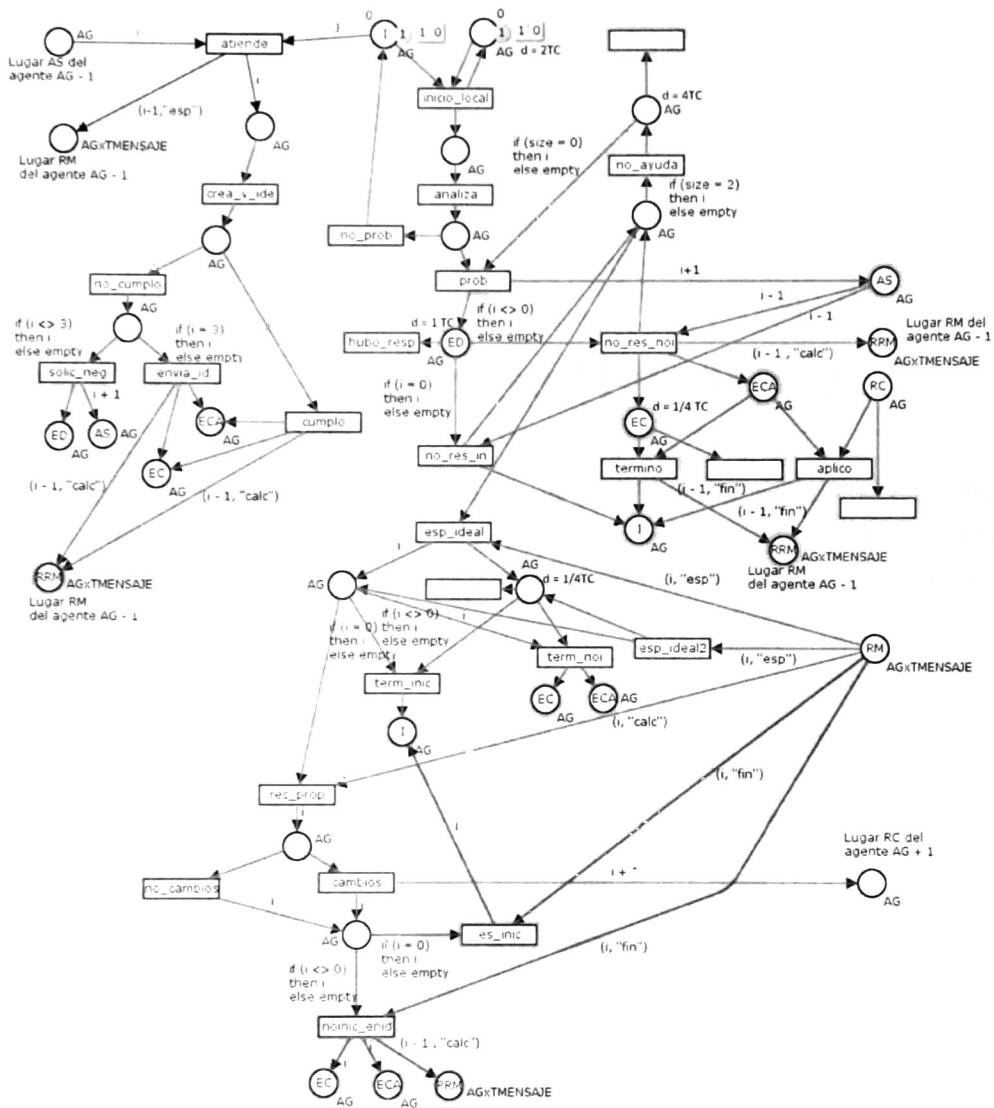


Figura 4.2 Protocolo de Negociación modelado con una RPC

#### 4.4. Conclusiones

Se presentó un esquema de coordinación de tráfico urbano basado en Sistemas MultiAgentes. Donde los agentes de control de cada intersección logran coordinación entre las intersecciones vecinas utilizando la Teoría de Negociación de Nash.

# Capítulo 5

## Pruebas y Análisis de Resultados.

**Resumen.** En este capítulo se presenta un caso de estudio donde se realizaron pruebas del comportamiento de los esquemas de control propuesto. Se presentan los resultados de las simulaciones y el análisis respectivo, con el que se comprueba la eficiencia de los esquemas de control.

### 5.1. Caso de Estudio

Con la finalidad de observar el comportamiento de los esquemas de control propuestos, se realizaron pruebas utilizando el simulador presentado en el Capítulo 2. Se definió un caso de estudio con cuatro intersecciones como se muestra en la Figura 5.1.

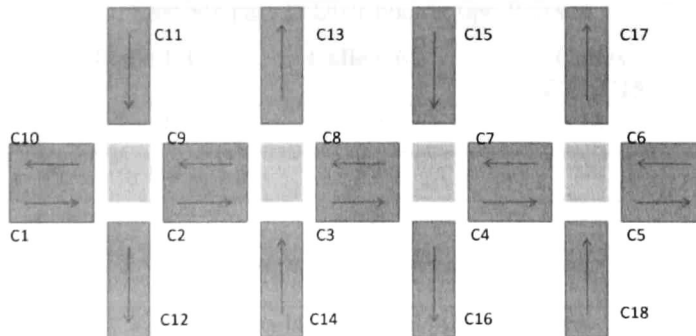


Figura 5.1 Caso de Estudio.

En el caso de estudio, la calle de doble sentido tiene mayor importancia; por lo tanto el flujo de llegada de coches es mayor que en las calles que la atraviesan. La política de control inicial, que utiliza un esquema de control fijo para las cuatro intersecciones tiene dos fases, una



de Norte a Sur o de Sur a Norte, según sea la intersección, con una duración de 10 s., y otra que incluye los movimientos de Este a Oeste y de Oeste a Este con una duración de 20 s.

Los segmentos de calle son de 100m. de longitud. La velocidad promedio de los autos es de 40km/h. La tasa de llegada de autos a los calles fuente se modifica en las diferentes simulaciones por medio del Módulo Generador de Autos, lo que permite probar el comportamiento de los esquemas de control en diferentes condiciones de tráfico.

## 5.2. Simulaciones realizadas

Durante cada simulación, inicialmente se realizan dos ciclos de la política de control con un esquema de control fijo. A partir de este momento, en cada ciclo se realiza un análisis de la situación actual de tráfico. El agente de control toma la decisión acerca de los cambios en los tiempos de verde de las fases de la política de control y realiza las negociaciones necesarias. Una vez tomada la decisión, se programa un nuevo ciclo de la política de control con los cambios y así sucesivamente hasta el final de la simulación.

Se simuló el caso de estudio con un esquema de control fijo y con los dos esquemas de control adaptable que puede realizar el agente de control. La simulación se ejecutó durante 2880 segundos para tres escenarios llamados Alto, Medio y Bajo, cada uno con diferentes tasas de llegada y con variaciones en las mismas, simulando el aumento de tráfico vehicular. Las variaciones en las tasas de llegada para los tres escenarios se muestran en la Tabla 5.1, Tabla 5.2 y Tabla 5.3 para los escenarios Alto, Medio y Bajo respectivamente.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	1	6	12	10
961-1920 s.	0.5	5	8	6
1921-2880 s.	1	6	12	10

Tabla 5.1 Variaciones en la tasa de llegada para el escenario Alto.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	1.6	6	12	10
961-1920 s.	1.3	5	8	6
1921-2880 s.	1.6	6	12	10

Tabla 5.2 Variaciones en la tasa de llegada para el escenario Medio.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	2	6	12	10
961-1920 s.	1	5	8	6
1921-2880 s.	2	6	12	10

Tabla 5.3 Variaciones en la tasa de llegada para el escenario Bajo.

La Figura 5.2 muestra la llegada de coches a la calle C1 y C11 para el primer experimento en el escenario Alto. El primer módulo que se conecta al simulador es el generador de autos, que almacena en archivo los flujos de entrada para cada una de las calles fuente de la red de tráfico. Estos archivos son utilizadas para simular la entrada de vehículos a la red de tráfico para las diferentes estrategias de control.

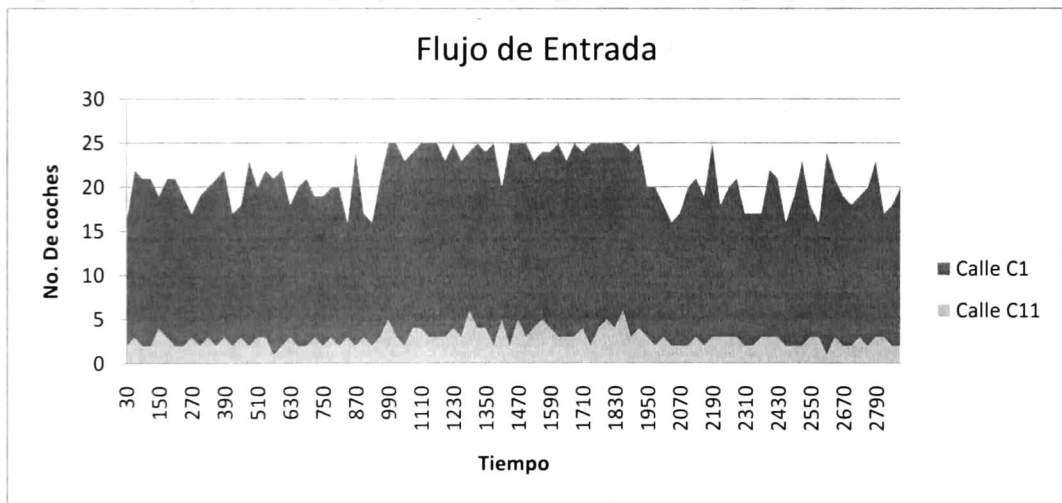


Figura 5.2 Flujo de Entrada Calle C1 y C11

### 5.3. Resultados Obtenidos

Los resultados presentados son obtenidos del promedio de dos simulaciones con cada esquema de control comparado. En principio, el hecho de que una calle reciba más tiempo de verde que otra, implica que las filas crezcan en la calle que está perdiendo tiempo de verde. Esto se muestra en la Figura 5.3 y en la Figura 5.4, donde la fila en la calle C1, que tiene mayor importancia, se disminuye visiblemente resolviendo el problema de programación lineal. Situación que afecta a la fila de la calle C11. Este comportamiento se ve repetido en todos los

escenarios, por lo que esta es la última vez que se mencionan filas como métrica de desempeño y en su lugar analizaremos flujo vehicular. Llamamos flujo vehicular al número de vehículos que dejan la intersección; en las gráficas se promedian los valores cada 180s.

A pesar de que la tasa de llegada para las calles fuente es la misma, la entrada a las calles de la red de tráfico que no son fuente varía de acuerdo a la estrategia de control que se esté utilizando, ya que ésta depende del número de coches que libera la calle de arriba. La Figura 5.5 muestra como el número de coches que entran a la calle C2 es mayor cuando se utilizan los controles adaptables presentados en esta tesis, que cuando se utiliza un esquema de control fijo. Este comportamiento se repite en todos los escenarios.

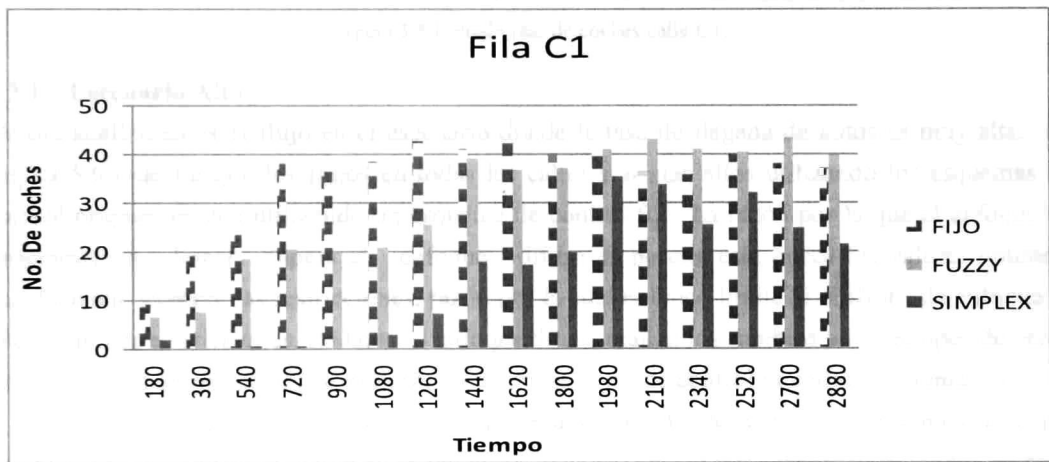


Figura 5.3 Fila en la calle C1.

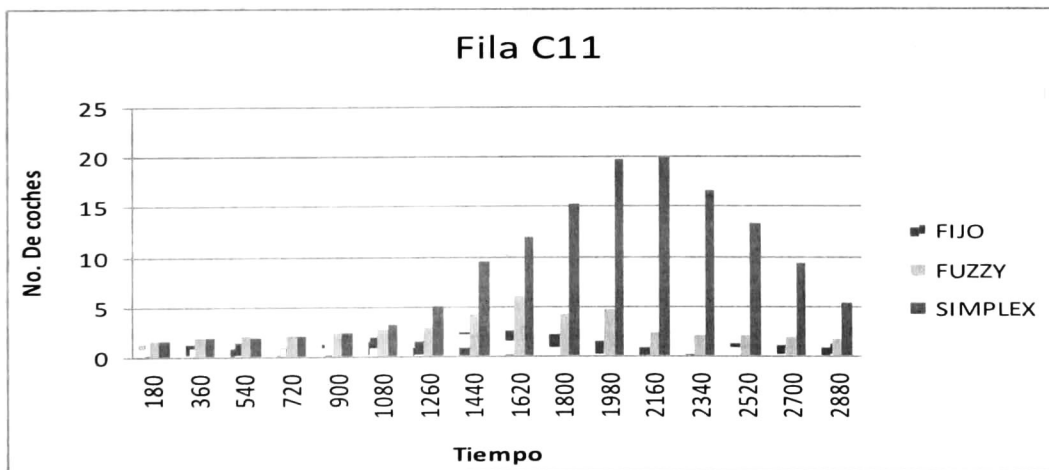


Figura 5.4 Fila en la calle C11.

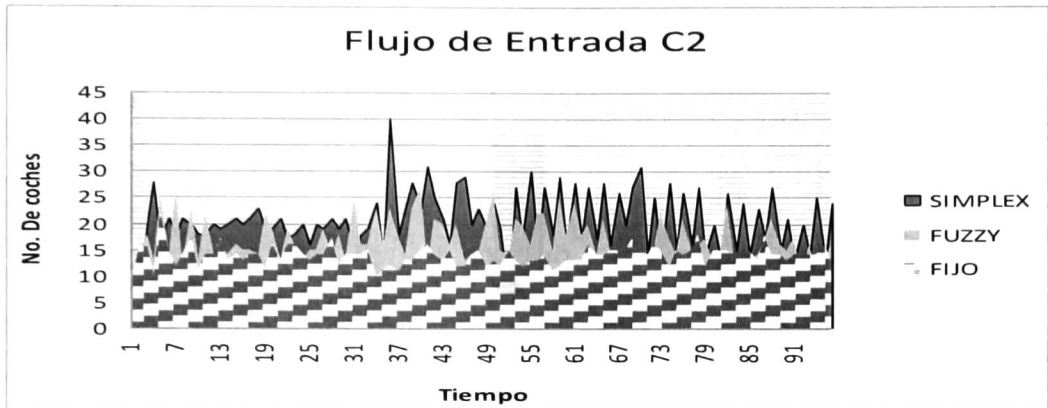


Figura 5.5 Entrada real de coches calle C1.

### 5.3.1. Escenario Alto

Ahora analizaremos el flujo en el escenario donde la tasa de llegada de autos es muy alta. La Figura 5.6 muestra que los flujos en todas las calles son más altos utilizando los esquemas de control propuestos que utilizando un esquema de control fijo. La razón por la que el enfoque de programación lineal supera al enfoque difuso es porque ésta busca los valores óptimos, satisfaciendo siempre las restricciones, razón por la cual el único límite al utilizar este enfoque es el conjunto de restricciones. En tanto el enfoque difuso realiza los cambios a los tiempos de verde realizando una aproximación al óptimo, imitando las leyes de control que usan los humanos.

Como el escenario Alto es el que tiene una tasa de llegada de coches muy alta; los cambios en los tiempos de verde alcanzan en su mayoría los valores máximos asignados en cada calle, esto se puede observar en la Figura 5.7 y en la Figura 5.8, donde los tiempos de verde asignados a la calle de mayor importancia alcanzan un límite durante la mayoría del tiempo de simulación. El hecho de que los cambios a tiempos de verde en el esquema difuso sean más suaves que utilizando programación lineal, se refleja en un flujo más elevado en la calle C11, que tiene menor importancia, esto se puede observar en la Figura 5.9. Este comportamiento se repite para cada una de las intersecciones, pues el flujo de entrada satura las calles.

Una vez observada la ventaja de los esquemas de control propuestos sobre el esquema fijo, ahora analizamos los cambios presentados cuando los agentes de control negocian entre ellos. En la Figura 5.10 y en la Figura 5.11, se muestran los flujos para las calles C2 y C3 respectivamente, se observa que los flujos se incrementan cuando los agentes negocian, utilizando programación lineal, lo mismo en la Figura 5.12 y en la Figura 5.13 pero con el esquema difuso. En cuanto a los tiempos de verde, en la Figura 5.14 y en la Figura 5.15, se observan los tiempos de verde asignados utilizando los esquemas difuso y programación lineal respectivamente, así como el aumento en segundos logrado al negociar con los agentes vecinos.

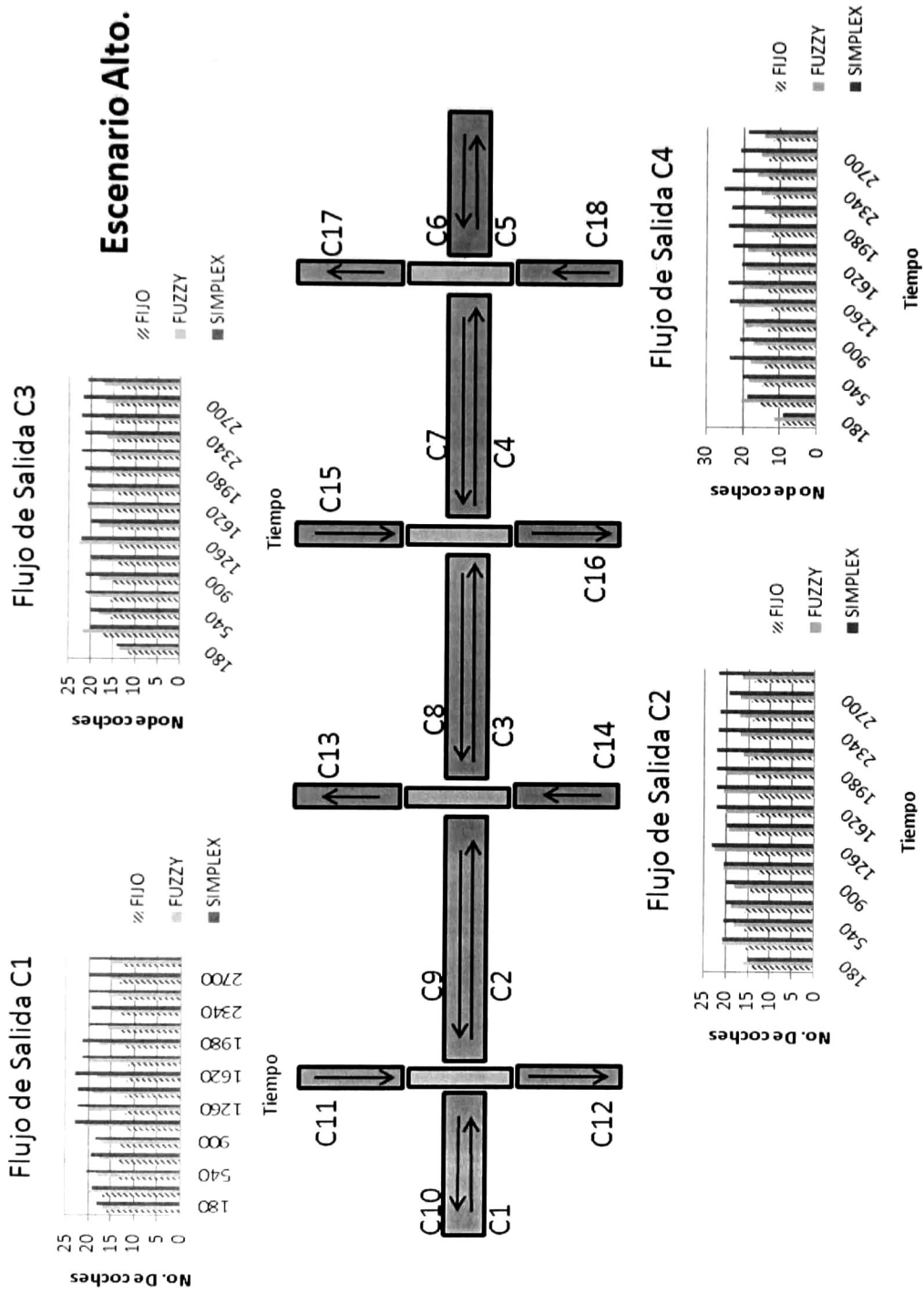


Figura 5.6. Flujos en escenario Alto.



### 5.3.2. Escenario Medio

Ahora analizaremos el flujo en el escenario donde la tasa de llegada de autos es más reducida. La Figura 5.16 muestra que los flujos en todas las calles son más altos utilizando los esquemas de control propuestos que utilizando un esquema de control fijo. Pero en este caso, los enfoques de programación lineal y difuso no muestran diferencias significativas, pues en promedio están dando salida al mismo número de coches. Lo que nos lleva a analizar los tiempos de verde. En la Figura 5.17 y en la Figura 5.18 , se observa que el enfoque de programación lineal alcanza un nivel alto de tiempo de verde mucho antes que en el enfoque de lógica difusa. Esta situación se compensa pues en el enfoque difuso, una vez que aumenta el nivel de tiempo de verde, lo hace hasta un punto más alto que en el enfoque de programación lineal. La diferencia entre flujos es menos marcada que en el escenario alto, pues el ritmo de llegada de coches es más lenta.

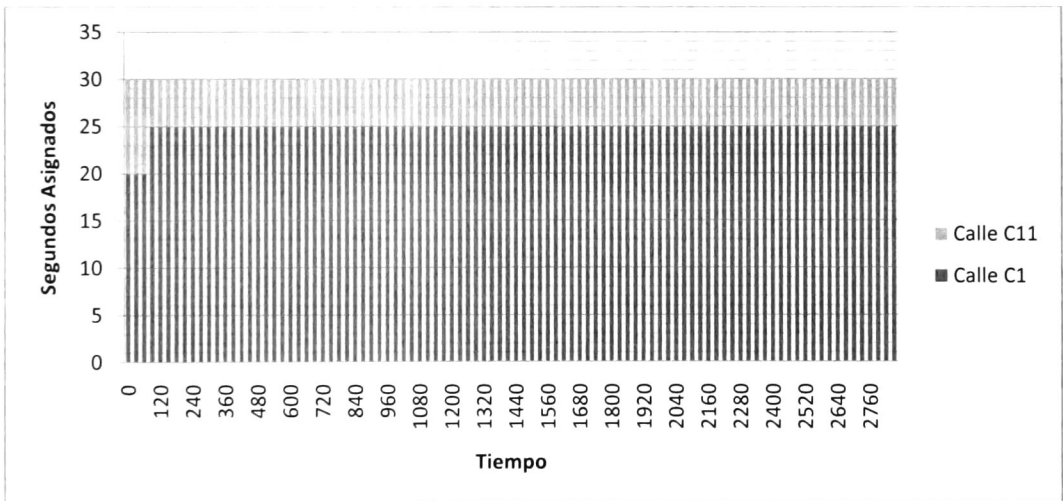


Figura 5.7 Tiempos de verde calles C1 y C11, esquema PL.

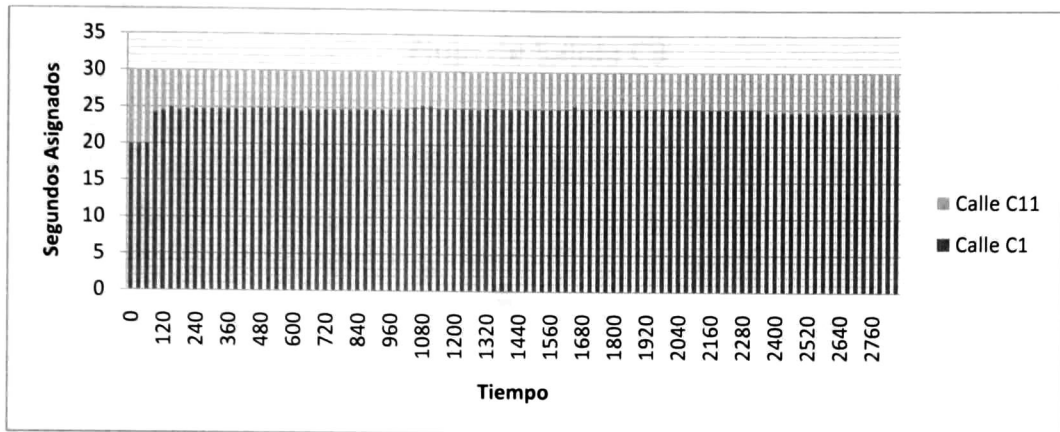


Figura 5.8 Tiempos de verde calles C1 y C11, esquema difuso.

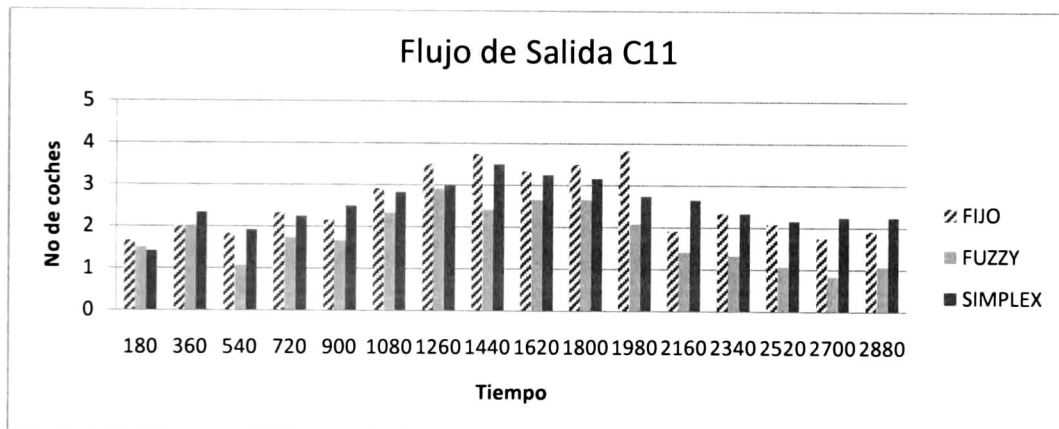


Figura 5.9 Flujo calle C11.

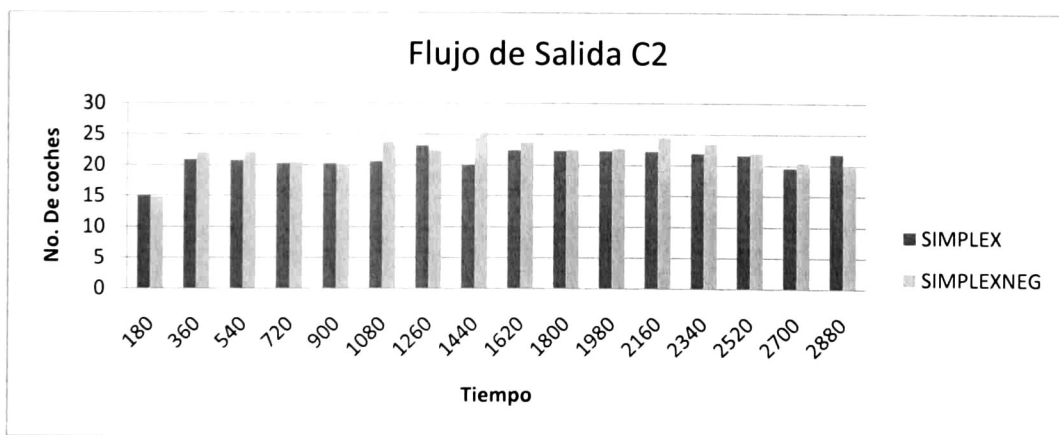


Figura 5.10 Flujo calle C2, esquema PL.

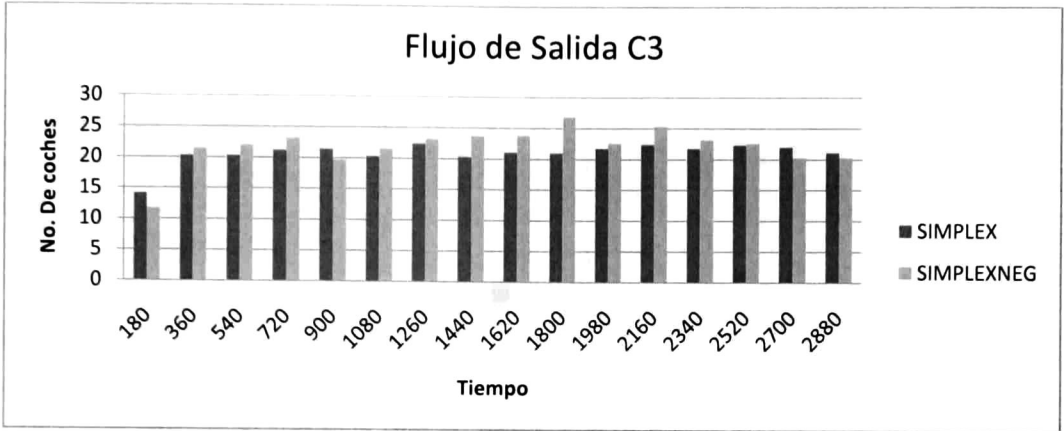


Figura 5.11 Flujo calle C3, esquema PL.

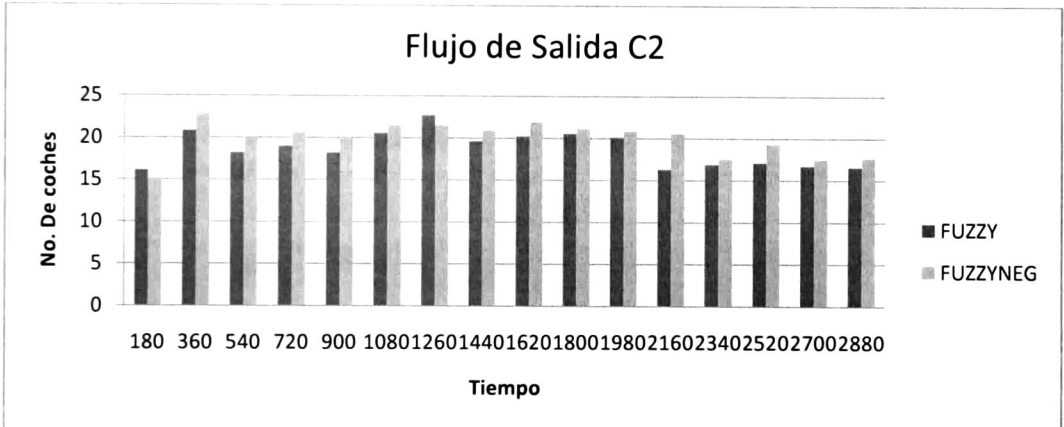


Figura 5.12 Flujo calle C2, esquema difuso.

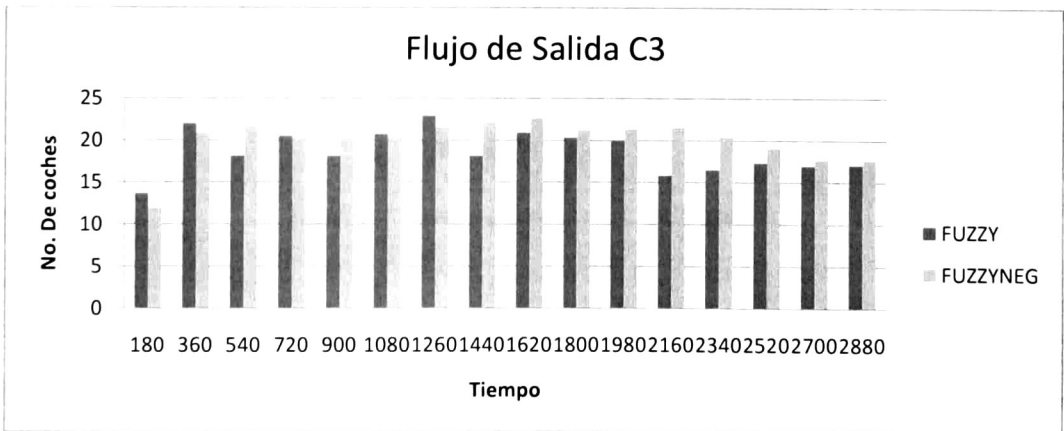


Figura 5.13 Flujo calle C3, esquema difuso.

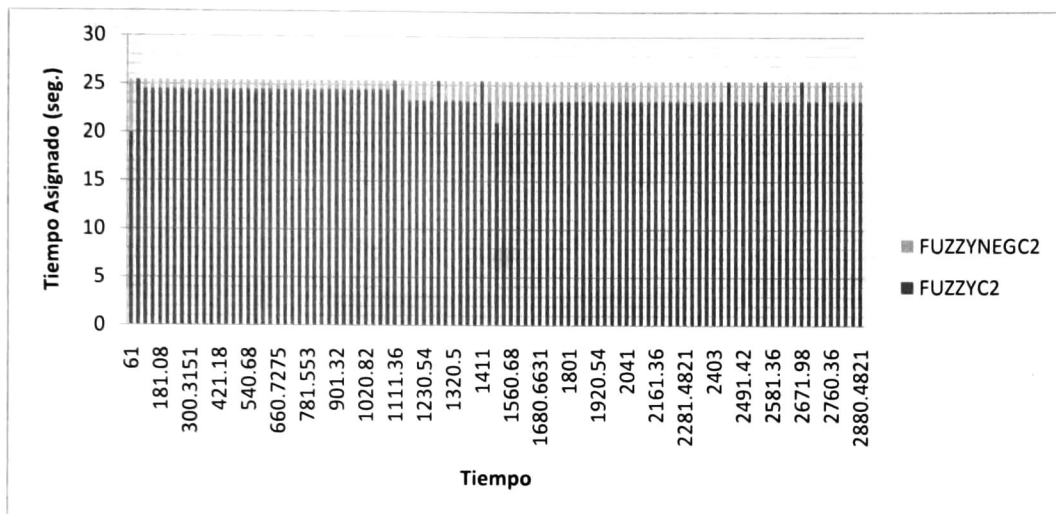


Figura 5.14 Tiempos de verde calle C2, esquema difuso.

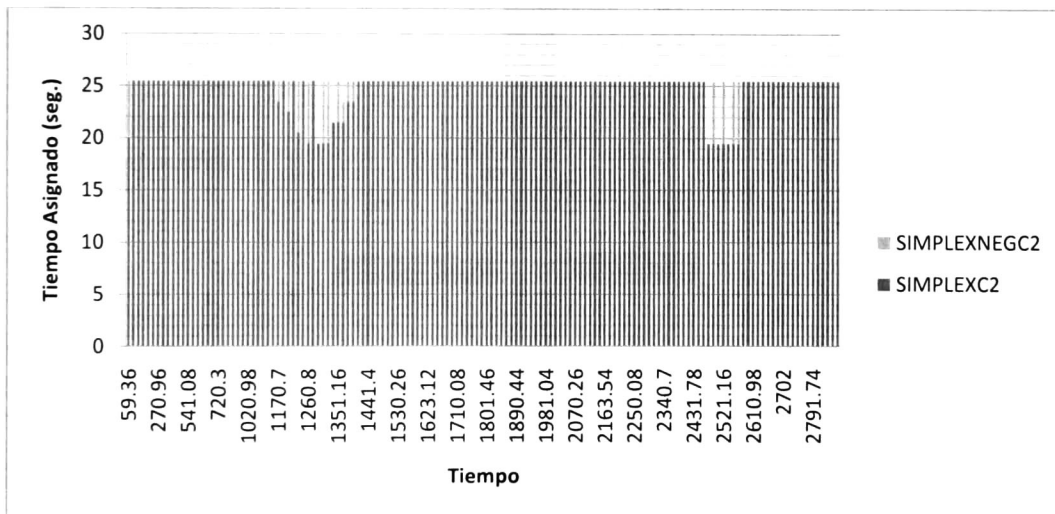
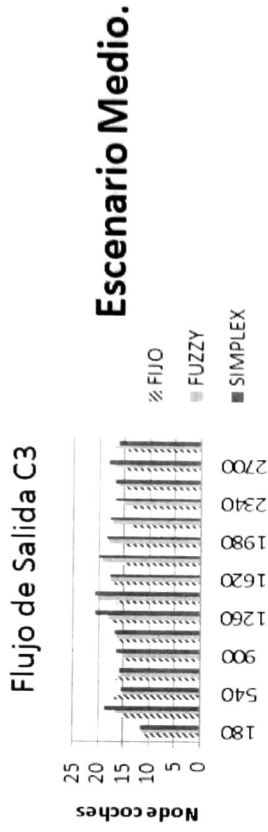
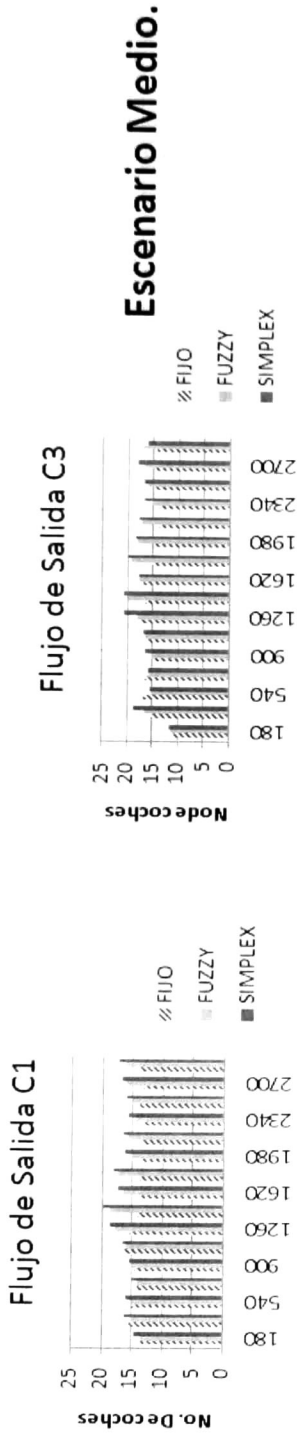


Figura 5.15 Tiempos de verde calle C2, esquema PL.



**Escenario Medio.**

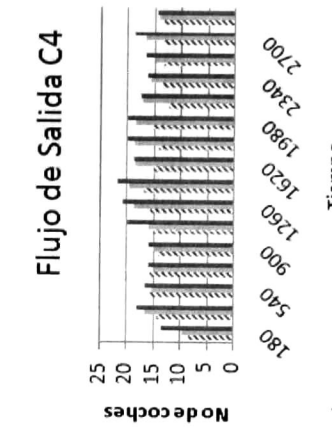
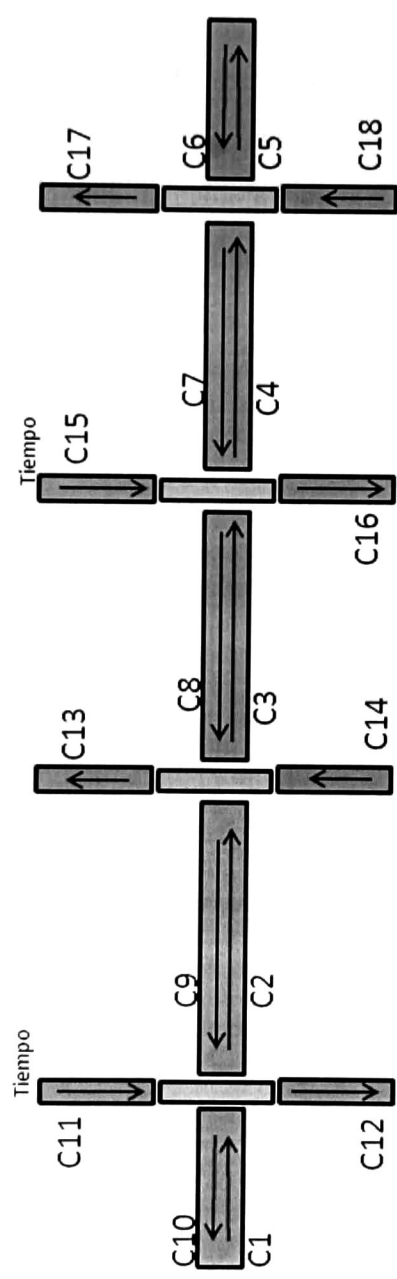


Figura 5.16. Flujos en escenario Medio



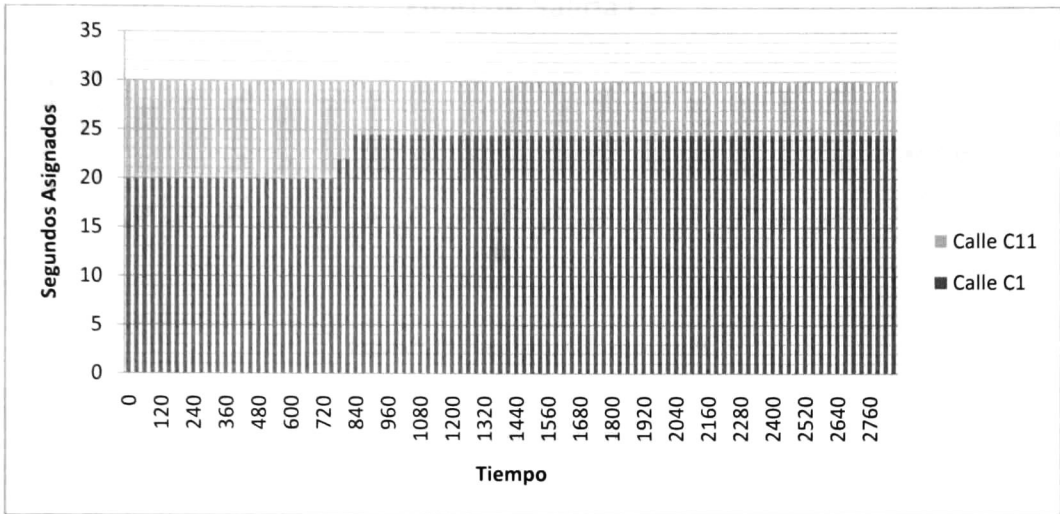


Figura 5.17 Tiempos de verde calles C1 y C11, esquema PL.

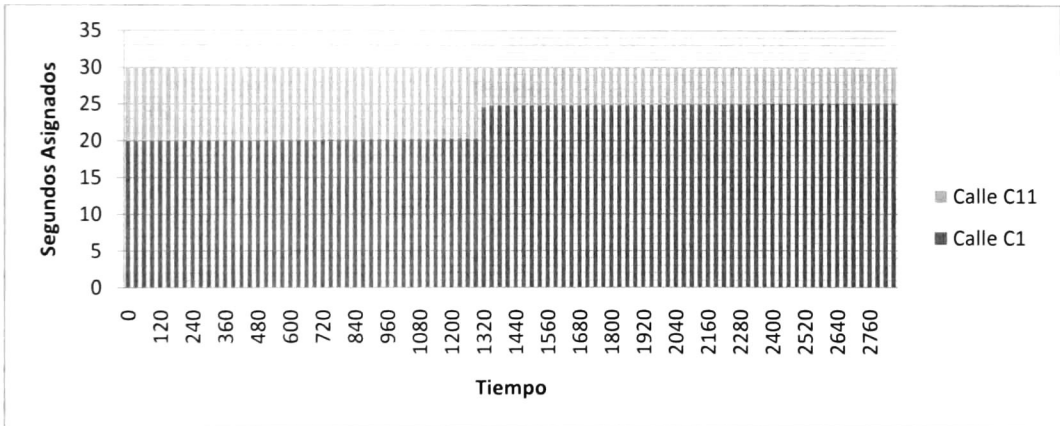


Figura 5.18 Tiempos de verde calles C1 y C11, esquema difuso.

Ahora analizamos los cambios presentados cuando los agentes de control negocian entre ellos. En la Figura 5.19 y en la Figura 5.20, se muestran los flujos para la calle C2, se observa que los flujos se incrementan cuando los agentes negocian, en los dos esquemas propuestos.

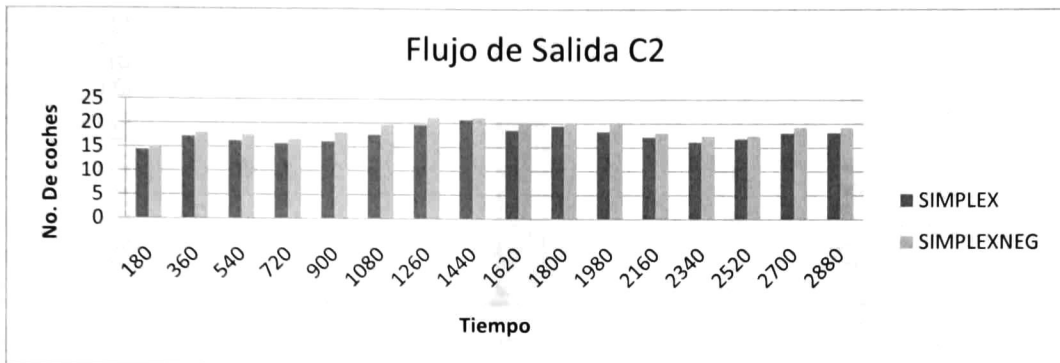


Figura 5.19 Flujo calle C2, esquema PL.

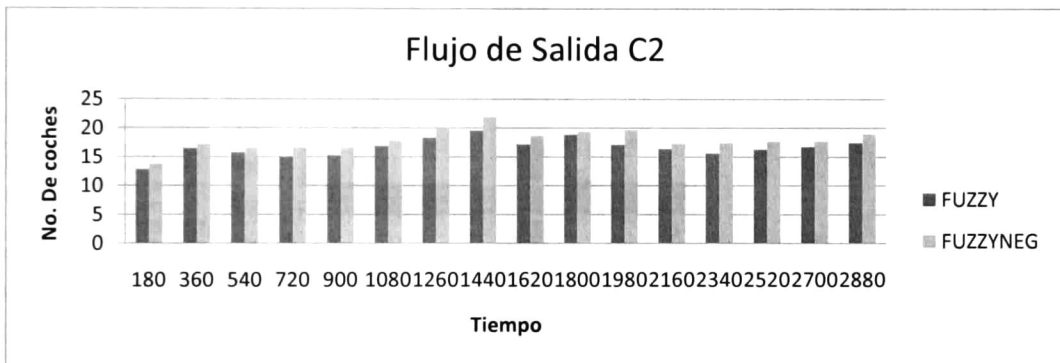


Figura 5.20 Flujo calle C2, esquema difuso.

### 5.3.3. Escenario Bajo

Ahora analizaremos el flujo en el escenario donde la tasa de llegada de autos es más baja. La Figura 5.21 muestra que los flujos en todas las calles son más altos utilizando los esquemas de control propuestos que utilizando un esquema de control fijo. Esta ventaja es muy ligera y en promedio, la salida total de coches es mayor utilizando el esquema de programación lineal que utilizando el esquema difuso.

Como la tasa de llegada de coches no es muy alta esto se refleja en los tiempos de verde asignados, pues no se alcanzan los límites. En la Figura 5.22 y en la Figura 5.23 se muestra como el esquema de control con programación lineal se mejora una vez que los agentes negocian, aumentando el flujo vehicular. En la Figura 5.24 y en la Figura 5.25 se observa la misma situación pero empleando el esquema de control difuso. En la Figura 5.26 y en la Figura 5.27 se observa que aún cuando los agentes negocian entre sí, los tiempos de verde no alcanzan los límites máximos.

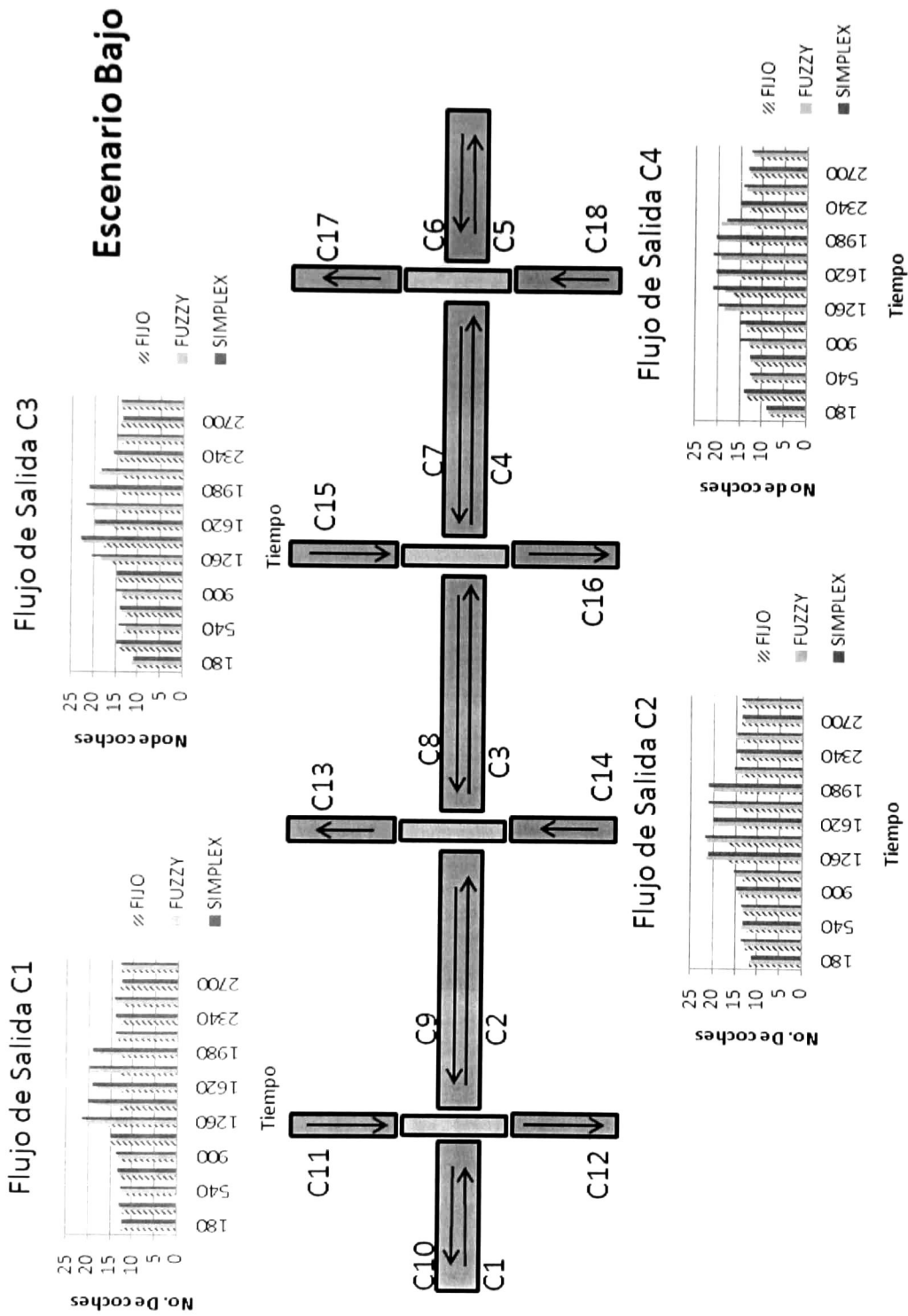


Figura 5.21. Flujos en escenario Bajo.

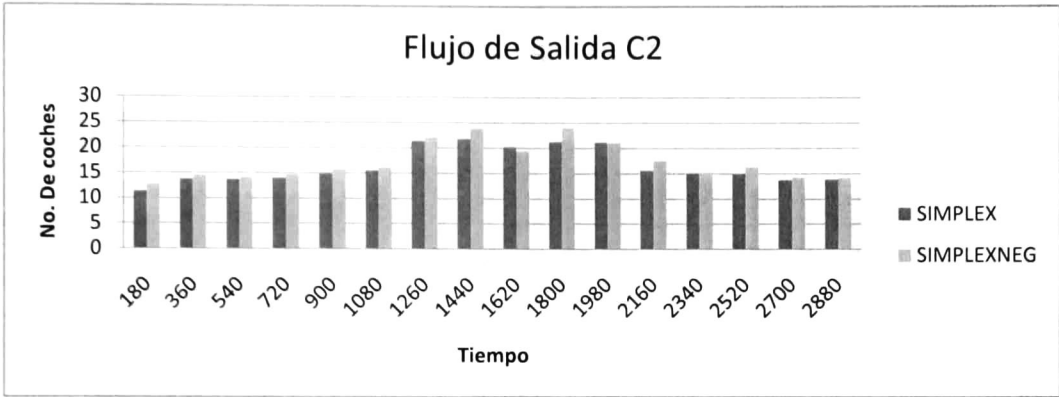


Figura 5.22 Flujo calle C2, esquema PL.

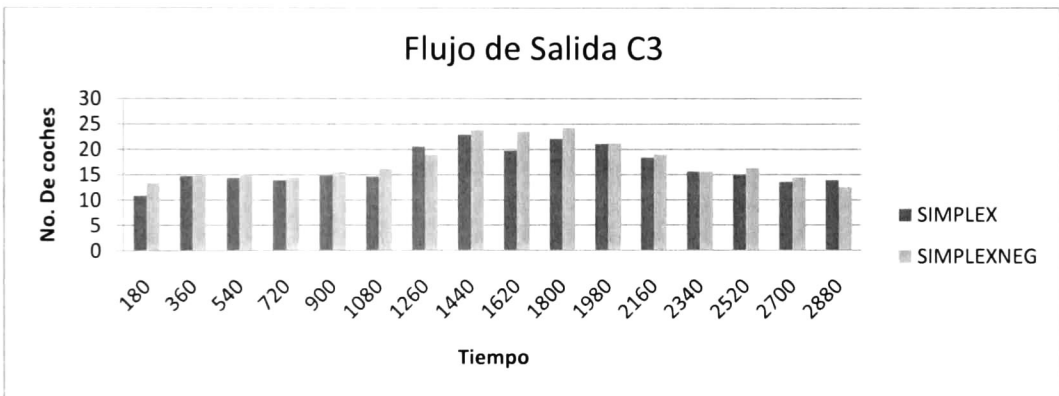


Figura 5.23 Flujo calle C3, esquema PL.

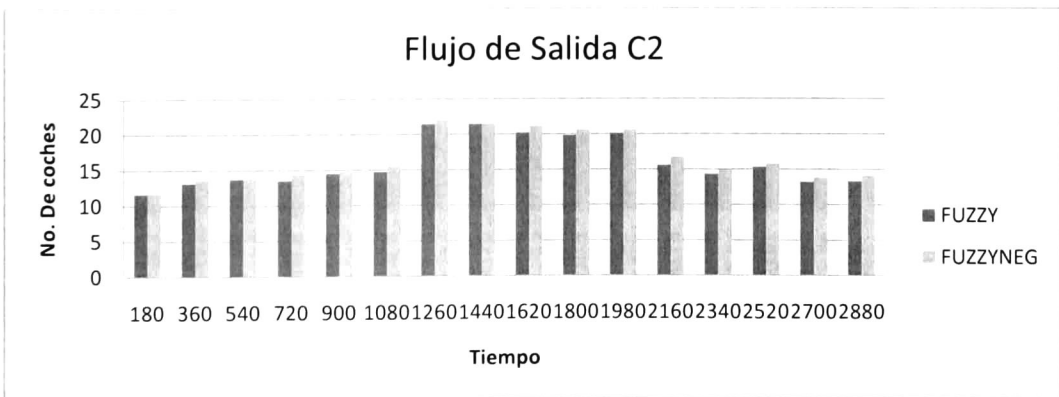


Figura 5.24 Flujo calle C2, esquema difuso.

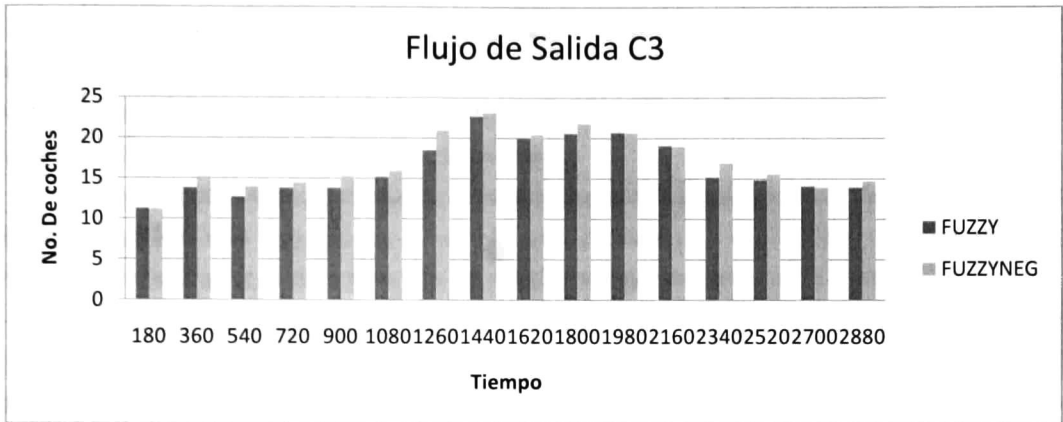


Figura 5.25 Flujo calle C3, esquema difuso.

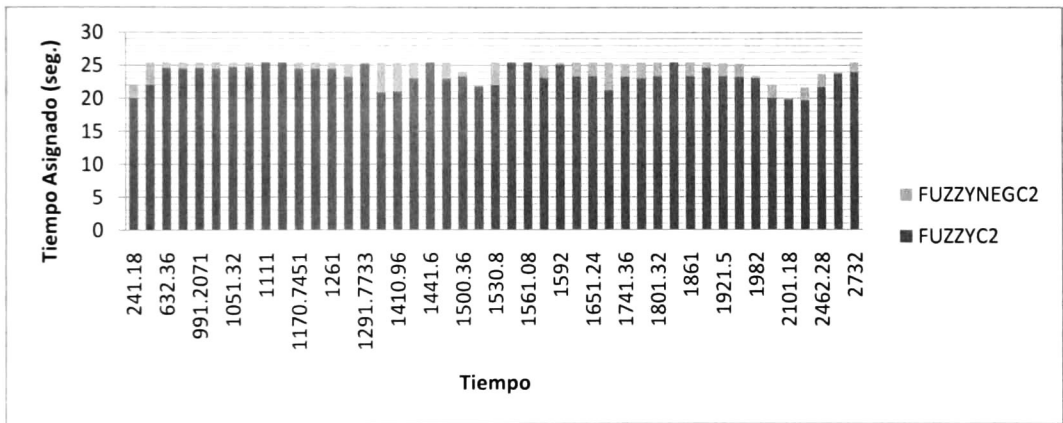


Figura 5.26 Tiempos de verde calle C2, esquema difuso.

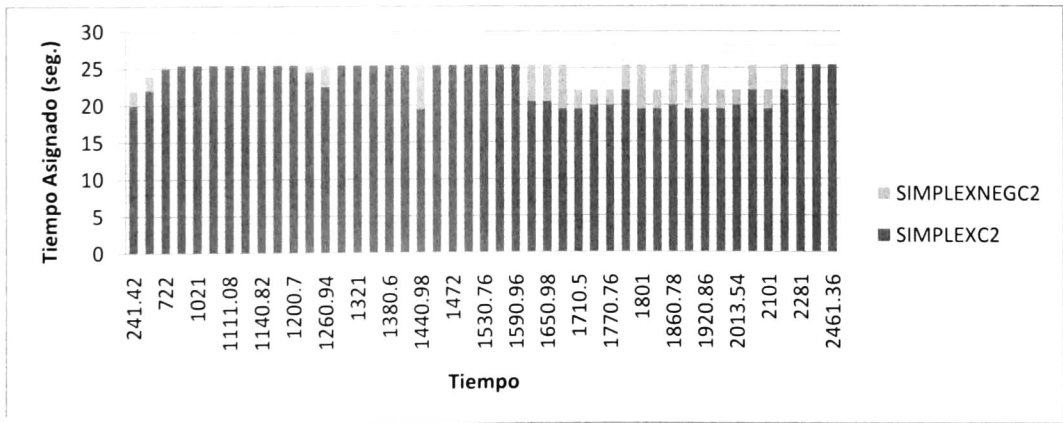


Figura 5.27 Tiempos de verde calle C2, esquema PL.



## **5.4. Conclusiones**

Se presentaron los experimentos realizados en el simulador de tráfico urbano. Se observó que el flujo vehicular se aumenta en todas las intersecciones cuando se utilizan los esquemas de control propuestos, ya que se adaptan a la demanda de tráfico. Se logró que un mayor número de vehículos utilicen la red sin que exista congestión.

Como puede notarse, durante las simulaciones realizadas las métricas de desempeño de la red de tráfico han sido mejorados considerablemente cuando los agentes de control realizan su trabajo, en comparación a una operación de semáforos bajo un esquema de control fijo.

El esquema de programación lineal supera ligeramente al esquema difuso, esto es debido a que la búsqueda de soluciones en este tipo de problemas se realiza en los límites de la región de posibles soluciones, obteniendo un valor óptimo.

En el caso del esquema difuso, se realiza una aproximación a las leyes de control humanas; esta aproximación se ve reflejada en los cambios a los tiempos de verde en los experimentos. Además, la negociación de los agentes incrementa el flujo vehicular, eliminando las posibilidades de congestión.

# Conclusiones

En esta tesis se presentó un esquema multiagentes para la coordinación de tráfico urbano. El controlador de cada intersección es un agente proactivo que determina los tiempos de verde de los semáforos de acuerdo a las condiciones de tráfico observadas. La operación del controlador es en dos fases: a) control local y b) negociación. En el control local se obtiene una temporización de acuerdo a las condiciones de tráfico de la intersección. La decisión final sobre la temporización a implementar es negociada con los agentes de intersecciones vecinas.

El control local fue realizado de dos formas diferentes basadas en dos enfoques. El primer enfoque utiliza un Sistema de Inferencia Difuso que evalúa las condiciones de tráfico variables. El esquema propuesto reduce considerablemente el número de reglas a evaluar comparado con propuestas de otros autores. Además se tomó en cuenta un grado de importancia de las calles en la redacción de las reglas difusas. Los resultados experimentales muestran que el esquema propuesto mejora en gran medida un esquema de control fijo.

El segundo enfoque resuelve un problema de programación lineal, que busca un equilibrio entre las pérdidas de tiempo de verde de las calles de la intersección, basado en grados de importancia de dichas calles. De esta forma se evita que las filas en las calles menos importantes crezcan demasiado por dar preferencia a las calles de mayor importancia.

Además, el agente de control, una vez que ha determinado los tiempos de verde adecuados para su intersección, realiza un proceso de identificación de problemas, donde evalúa las condiciones de tráfico de las intersecciones vecinas y en caso de detectar algún conflicto ocasionado por las política de control elegida, inicia una negociación con la intersección afectada antes de aplicar los cambios. De esta forma, se alcanza un acuerdo antes de aplicar la nueva temporización, evitando llevar a la red de tráfico a una situación de desequilibrio en los flujos de los tramos de una misma calle.

Como puede notarse, durante las simulaciones realizadas la evaluación del desempeño de la red de tráfico han sido mejorados considerablemente cuando los agentes de control realizan su trabajo, en comparación a una operación de semáforos bajo un esquema de control fijo. Aunque el desempeño de los esquemas adaptables es similar, el esquema de programación lineal supera ligeramente al esquema difuso, esto es debido a que la búsqueda de soluciones en este tipo de problemas se realiza en los límites de la región de posibles soluciones, obteniendo un valor óptimo. En el caso del esquema difuso, se realiza una aproximación a las leyes de control

humanas, esta aproximación se ve reflejada en los cambios a los tiempos de verde de los experimentos.

Los resultados obtenidos con el esquema multiagentes propuesto son prometedores. La fase de negociación complementa la decisión local amortiguando los efectos de un cambio drástico en la operación de una intersección. Sin embargo otras capacidades pueden ser adicionadas a los agentes, tales como el aprendizaje, las cuales permitirían tomar decisiones de tipo predictivas. De igual forma es interesante abordar casos de estudio con intersecciones que requieran políticas de control con más fases donde también se prevea responder a eventos externos, tales como peatones y vehículos con prioridad.

# Apéndice A

## Conceptos Básicos sobre Lógica Difusa y Teoría de Juegos

### A.1. Lógica Difusa.

La lógica difusa proporciona los fundamentos para el razonamiento aproximado utilizando proposiciones imprecisas basadas en la teoría de conjuntos difusos. [Chen, 2001]. Revisaremos aquí los conceptos más importantes.

#### A.1.1. Conjuntos Difusos y Funciones de Membresía

Contrario a la teoría clásica de conjuntos, un conjunto difuso no tiene límites bien definidos que separaran a los elementos que pertenecen al conjunto y los que no.

Un conjunto difuso  $A$  de un dominio  $X$  es una función  $A: X \rightarrow [0,1]$  de  $X$  a el intervalo  $[0, 1]$ . El valor de  $A(x)$  es conocido como el grado de membresía de  $x$  en  $A$ . Esta función es algunas veces llamada la función de membresía de  $A$ . El conjunto  $X$  es algunas veces llamado el universo de discurso [Nguyen, 2003]. El conjunto difuso  $A$ , está determinado por los pares

$$A = \{(x, A(x)) \mid x \in X\}$$

#### A.1.2. Variables Lingüísticas y Operaciones con lógica difusa

Las variables lingüísticas toman valores en un lenguaje natural y cada valor está representado por un conjunto difuso, con lo que se logra que las expresiones naturales tengan una interpretación matemática.

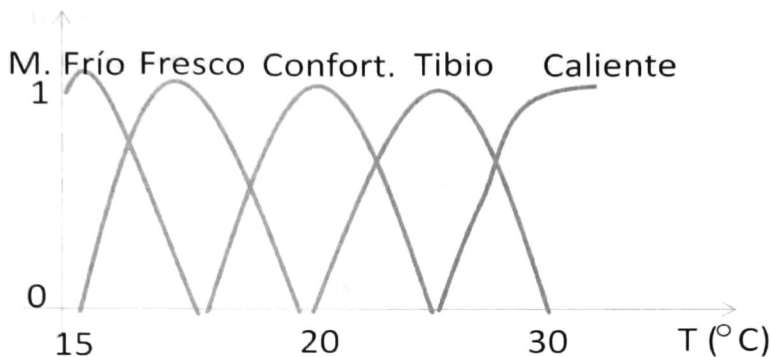


Figura A.1 Variable Lingüística *Temperatura*.

Ahora se muestran las operaciones con lógica difusa y sus evaluaciones en un conjunto difuso  $A$  con la función de membresía  $A(x)$ :

- $A = B$ , dos conjuntos difusos son iguales si y solo si:  $\forall x \in X : A(x) = B(x)$
- $A \subseteq B$ ,  $A$  es un subconjunto de  $B$ , si y solo si:  $\forall x \in X : A(x) \leq B(x)$
- $A \cup B$ ,  $A$  unión  $B$ , (OR),  $\forall x \in X : (A \cup B)(x) = \max(A(x), B(x))$
- $A \cap B$ ,  $A$  intersección  $B$ , (AND),  $\forall x \in X : (A \cap B)(x) = \min(A(x), B(x))$
- Complemento de  $A$ ,  $\forall x \in X : (X - A)(x) = 1 - A(x)$

### A.1.3. Modelo Takagi-Sugeno

Las reglas en un sistema basado en reglas son generalmente expresadas en una forma tal que “Si  $x$  es  $A$  entonces  $y$  es  $B$ ”, donde  $A$  y  $B$  son conjuntos difusos,  $x$  está en el dominio de  $A$  e  $y$  está en el dominio de  $B$ . Existen muchas generalizaciones de la operación implicación en lógica clásica a conjuntos difusos, pero la mayoría no son generalizaciones de la implicación en lógica clásica [Nguyen, 2003]. A continuación se presenta uno de ellos.

Para el modelo Takagi-Sugeno, las reglas son dadas en la forma:

$$R_i : \text{Si } x_1 \text{ es } A_{i1} \text{ y } x_2 \text{ es } A_{i2} \text{ y...y } x_k \text{ es } A_{ik} \text{ entonces } f_i(x_1, x_2, \dots, x_k), \quad i = 1, 2, \dots, n$$

La salida de este modelo para la regla  $R_i$ ,  $R(x)$ , se obtiene de la siguiente forma:

$$R(x) = \frac{A_1(x)f_1(x) + A_2(x)f_2(x) + \dots + A_n(x)f_n(x)}{A_1(x) + A_2(x) + \dots + A_n(x)}$$



#### A.1.4. Control Difuso

Las implementaciones de control difuso son de alguna manera, imitaciones de las leyes de control que utilizan los humanos. El crear máquinas que emulan la experiencia humana en control nos da una nueva forma de diseñar controladores para problemas complejas cuyos modelos no son fáciles de especificar. [Nguyen, 2003]. En la Figura A.2 se muestra la estructura de un controlador difuso.



Figura A.2 Estructura de un Controlador Difuso.

El módulo de *fuzzificación* transforma los valores de entrada en valores de pertenencia en conjuntos difusos. El módulo de *inferencia* contiene una base de datos y una base de reglas, la base de datos contiene las funciones de membresía y los universos de discurso. El módulo de *desfuzzificación* convierte todos los términos difusos creados por la base de reglas a valores numéricos.

## A.2. Teoría de Juegos en Sistemas Multi-Agentes

Los sistemas basados en agentes están situados en algún ambiente. Las acciones que un agente tiene a su disposición son no deterministas, por lo que se tiene un rango de posibles resultados. Cada agente tiene objetivos (posiblemente conflictivos) que trata de lograr, además puede haber otros agentes operando en el ambiente con los que el primer agente tiene que interactuar. Estos aspectos pueden ser manejados por la aplicación de teoría de juegos y teoría de decisión. [Parsons, 2006].

### A.2.1. Encuentros Multiagentes

Supongamos que tenemos dos agentes. Cada uno de los agentes es egoísta, es decir, cada agente tiene sus preferencias y deseos acerca de cómo debería ser el mundo. Estas preferencias se toman como utilidades. Se necesita introducir un modelo del ambiente en donde los agentes se van a desenvolver. La idea es que los dos agentes eligen acciones a realizar en el ambiente, y esto

generará un resultado que causará un cambio en el ambiente. Este resultado dependerá de la combinación de acciones realizadas.

Se capturan las preferencias de un agente a través de una función de utilidad, que está definida como:

$$u_i : \Omega \rightarrow \mathbb{R}$$

Una utilidad es un valor, el cual es asociado a un estado del mundo, que representa el valor que el agente da a ese estado del mundo.

La forma en que el ambiente se conforma está determinado por una función ambiente:

$$Env : Ac \times Ac \rightarrow \Omega$$

Ahora se pueden reunir la función ambiente con la función de utilidad que tiene un agente sobre la salida de la función ambiente, y de esta forma se pueden formar matrices de pagos. [Parsons, 2002].

### A.2.2. Teoría de Juegos

La teoría juegos es una rama de la economía que estudia las interacciones entre agentes egoístas. Esta teoría tiene sus raíces en el trabajo de von Neumann y Morgenstern. Se ha vuelto claro que las técnicas y resultados de la teoría de juegos puede ser aplicada a todas las interacciones que puedan ocurrir entre agentes egoístas. [Parsons, 2002].

Un juego está definido por quienes toman las decisiones, llamados jugadores: las decisiones que deben tomar, la forma en que las decisiones determinan el resultado, y sus preferencias sobre esos resultados representados por las funciones de utilidad. En los juegos, la maximización de los pagos es ambigua, porque los pagos para los jugadores dependen de las estrategias de otros jugadores [Crawford, 2000].

#### Juegos Cooperativos y no Cooperativos

En la teoría de juegos no cooperativos, se tiene un conjunto de jugadores que toman sus decisiones independientemente unos de otros, aunque conociendo sus oponentes y las posibles estrategias que estos tienen a su disposición. Los jugadores tratan de predecir lo que otros jugadores harán para trabajar entonces en conveniencia propia.

Un juego no cooperativo es una tupla  $\Gamma = \langle N, S^i, R \rangle$ , donde  $N = \{1, 2, \dots, n\}$  es el conjunto de jugadores,  $S^i = \{S_1^i, \dots, S_{|S^i|}^i\}$  es el conjunto de estrategias puras de un jugador  $i$ , y  $R = \{u^i | i \in N \text{ y } u^i : S \rightarrow \mathbb{R}\}$  es la función de pago de un jugador  $i$  [Herings, 2002].

Cuando todos los jugadores predicen correctamente las estrategias de sus oponentes, y juegan sus mejores respuestas a estas predicciones, la estrategia resultante es un Equilibrio de Nash.

En la teoría de juegos cooperativos, los jugadores formarán coaliciones siempre y cuando puedan obtener un beneficio de éstas. Los jugadores tienen información sobre cierta valoración a priori de las coaliciones, reconociendo cuales son menos valiosas y cuales son más valiosas [Monsalve, 2003].

### Estrategias Puras y Mixtas

Una estrategia pura define las acciones que los jugadores seguirán en cada posible momento del juego.

Una estrategia mixta para un jugador  $i$  es una distribución de probabilidad sobre el conjunto de estrategias puras del jugador. Un jugador  $i$  selecciona un vector de probabilidades sobre las columnas en la matriz de pagos que se relaciona a las estrategias puras que debe elegir. En general, cuando un jugador  $i$  tiene  $n$  posibles estrategias, necesita un vector:

$$c = [c_1, c_2, \dots, c_n]^T, \text{ donde: } \sum_k c_k = 1 \quad \text{y} \quad c_k \geq 0$$

Entonces el jugador  $i$  selecciona la estrategia  $k$  con la probabilidad  $c_k$ . Cualquier vector de probabilidades da como resultado una estrategia mixta. [Parsons, 2002].

### Equilibrio de Nash

Sea  $N = \{1, 2, \dots, n\}$  el conjunto de jugadores,  $E_i = \{s_{1,\dots,s_k}\}$  el conjunto de estrategias del agente  $i$ ,  $SS = E_1 \times E_2 \times \dots \times E_n$  el espacio de estrategias de todos los jugadores. Sea  $S = \{s_1, \dots, s_n\} \in SS$  el conjunto de estrategias elegidas por cada uno de los jugadores, y  $u_i : SS \rightarrow \mathbb{R}$  la función de pago para el un conjunto de estrategias, al elegir una estrategia. Sea  $S \oplus s'_i = \{s_1, \dots, s_{i-1}, s'_i, s_{i+1}, s_i, \dots, s_n\}$  el conjunto de estrategias obtenido si el agente  $i$  cambia la estrategia  $s_i$  por la estrategia  $s'_i$ . Se dice que  $S$  está en un punto de equilibrio de Nash si ningún agente obtiene un incentivo al cambiar de estrategia, es decir:

$$u_i(S) \geq u_i(S \oplus s'_i) \quad \forall s'_i \in E_i \quad [\text{Vetta, 2002}].$$

### Estrategia de Negociación de Nash

Dentro de los juegos cooperativos se encuentran los modelos de Negociación, donde John Nash realizó contribuciones fundamentales, donde supone que la negociación entre agentes racionales conduce a un único resultado. Y lo determina imponiéndole al modelo ciertas propiedades deseables. [Monsalve, 2003].

Un problema de Negociación es un par  $(S, d)$  donde  $S \subseteq \mathbb{R}^2$  es un conjunto compacto y convexo que contiene pares  $s = s_1^x \times s_2^x$ , que asignan una utilidad a cada jugador al elegir la

estrategia  $x$ .  $d \in S$  y existe  $s \in S$  con  $s_i \geq d_i$ ,  $\mathcal{B}$  es el conjunto de todos los posibles problemas de Negociación.

La solución de Negociación de Nash es la solución  $f : \mathcal{B} \rightarrow 2^{\mathbb{R}^2} \setminus \emptyset$  que para cada problema de Negociación  $(S, d)$  selecciona el par  $\{(s_1^*, s_2^*)\} \subseteq S$  que contiene el punto que satisface  $(s_1^* - d_1)(s_2^* - d_2) \geq (s_1 - d_1)(s_2 - d_2), \forall (s_1, s_2) \in (S, d)$ . [Dagan, 2002].

La solución tiene las siguientes propiedades:

*Simetría:*  $(s_1, s_2) \in f(S, d) \leftrightarrow (s_2, s_1) \in f(S, d)$

*Pareto óptimo:* No existe ninguna otra solución donde todos los jugadores simultáneamente ganen mas.

*Invariante a transformaciones:* Tenemos  $(S', d')$  que se obtiene de  $(S, d)$  por medio de la transformación:  $s_i \rightarrow \alpha_i s_i + \beta_i$

donde:  $\alpha_i \geq 0$  y  $\beta_i \in \mathbb{R}$ ,

entonces:  $f_i(S', d') = \alpha_i f_i(S, d) + \beta_i$

*Independiente de alternativas irrelevantes:*  $f(S', d) = f(S, d) \cap S'$

donde:  $S' \subseteq S$  y  $f(S, d) \cap S' \neq \emptyset$ .

En la Figura A.3, se muestra gráficamente el conjunto  $S$ . Los ejes  $u_i$  representan las utilidades para el jugador  $i$ .

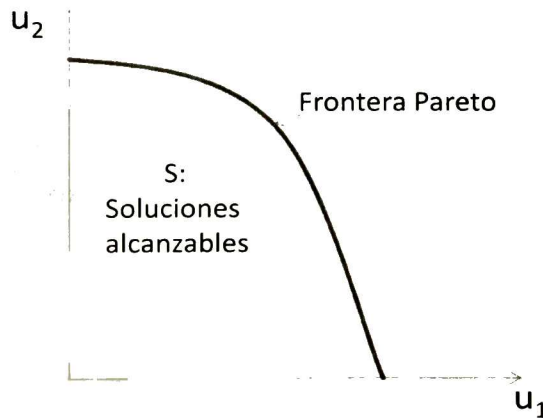


Figura A.3 Región alcanzable con su frontera Pareto.

# Apéndice B

## Resultados obtenidos de las simulaciones

### B.1. Caso de Estudio

Se presentan los resultados del comportamiento de los esquemas de control propuesto. Se definió un caso de estudio con cuatro intersecciones como se muestra en la Figura B.1.

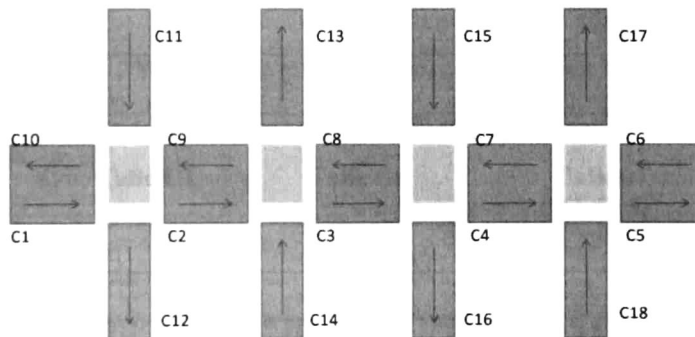


Figura B.1 Caso de Estudio

Las calles de mayor importancia son las de doble sentido. Se programan inicialmente dos ciclos con un esquema de control fijo de 20s. para las calles de mayor importancia y de 10 s. para el resto. A partir de este momento, en cada ciclo se realiza un análisis de la situación actual de tráfico. El agente de control toma la decisión acerca de los cambios en los tiempos de verde de las fases de la política de control y realiza las negociaciones necesarias. Después se programa un nuevo ciclo con los cambios y así sucesivamente hasta el final de la simulación.

Los segmentos de calle son de 100m. de longitud. La velocidad promedio de los autos es de 40km/h. La tasa de llegada de autos a los calles fuente se modifica en las diferentes simulaciones por medio del Módulo Generador de Autos, lo que permite probar el comportamiento de los esquemas de control en diferentes condiciones de tráfico.



La simulación se ejecutó durante 2880 segundos para tres escenarios llamados; Alto, Medio y Bajo. Las variaciones en las tasas de llegada para los tres escenarios se muestran en la Tabla B.1, Tabla B.2 y Tabla B.3 para los escenarios Alto, Medio y Bajo respectivamente.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	1	6	12	10
961-1920 s.	0.5	5	8	6
1921-2880 s.	1	6	12	10

Tabla B.1 Variaciones en la tasa de llegada para el escenario Alto.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	1.6	6	12	10
961-1920 s.	1.3	5	8	6
1921-2880 s.	1.6	6	12	10

Tabla B.2 Variaciones en la tasa de llegada para el escenario Medio.

Lambda para la Distribución tipo Poisson				
Tiempo/Calle	Calle C1	Calle C6	Calles C11,C15	Calles C14,C18
0-960 s.	2	6	12	10
961-1920 s.	1	5	8	6
1921-2880 s.	2	6	12	10

Tabla B.3 Variaciones en la tasa de llegada para el escenario Bajo.

## B.2. Escenario Alto

Resultados de simulaciones para el escenario alto.

### B.2.1. Llegadas de coches

Flujos de entrada desde archivo

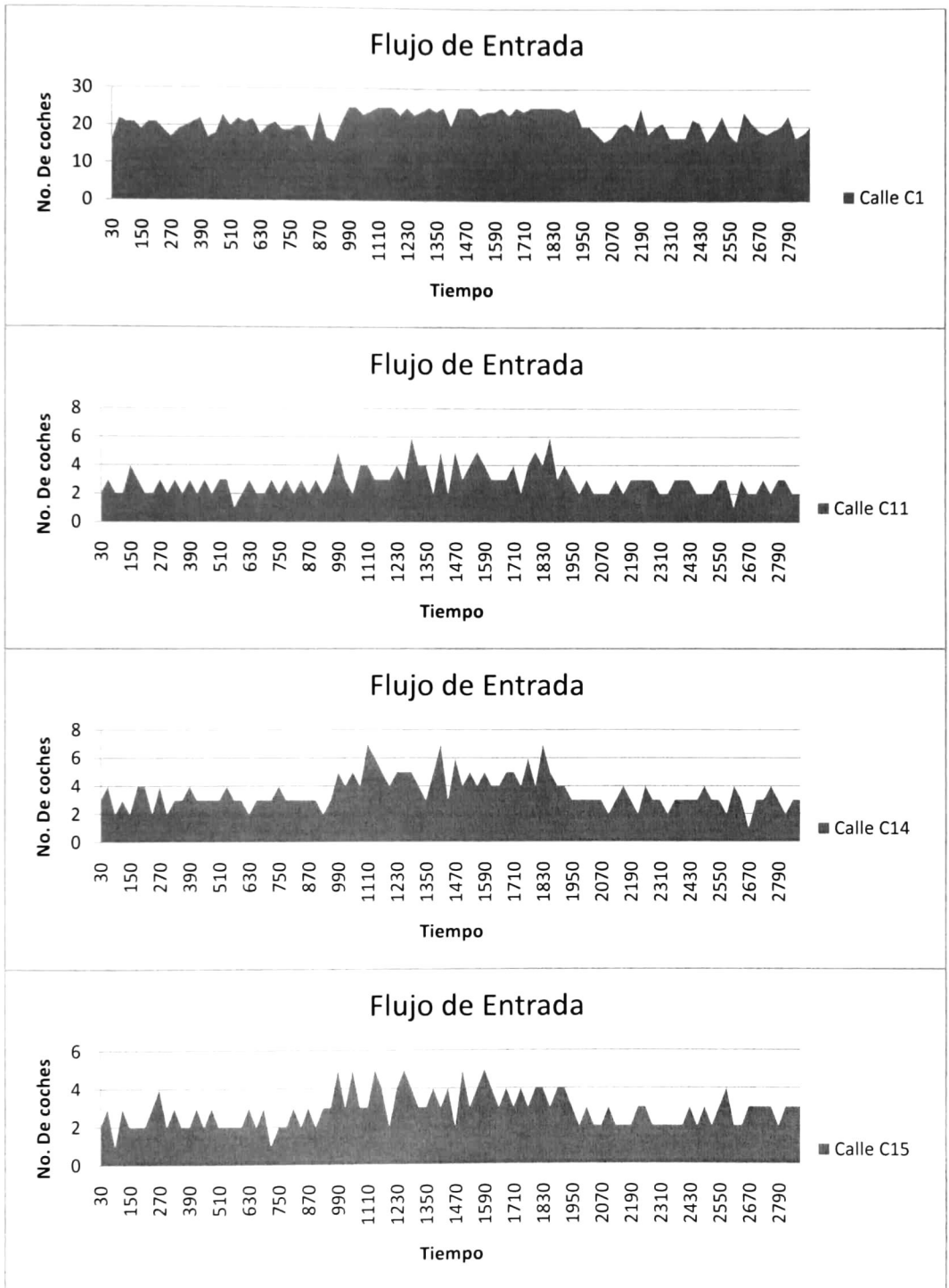


Figura B.2 Flujos de Entrada Calles C1, C11, C14, C15.

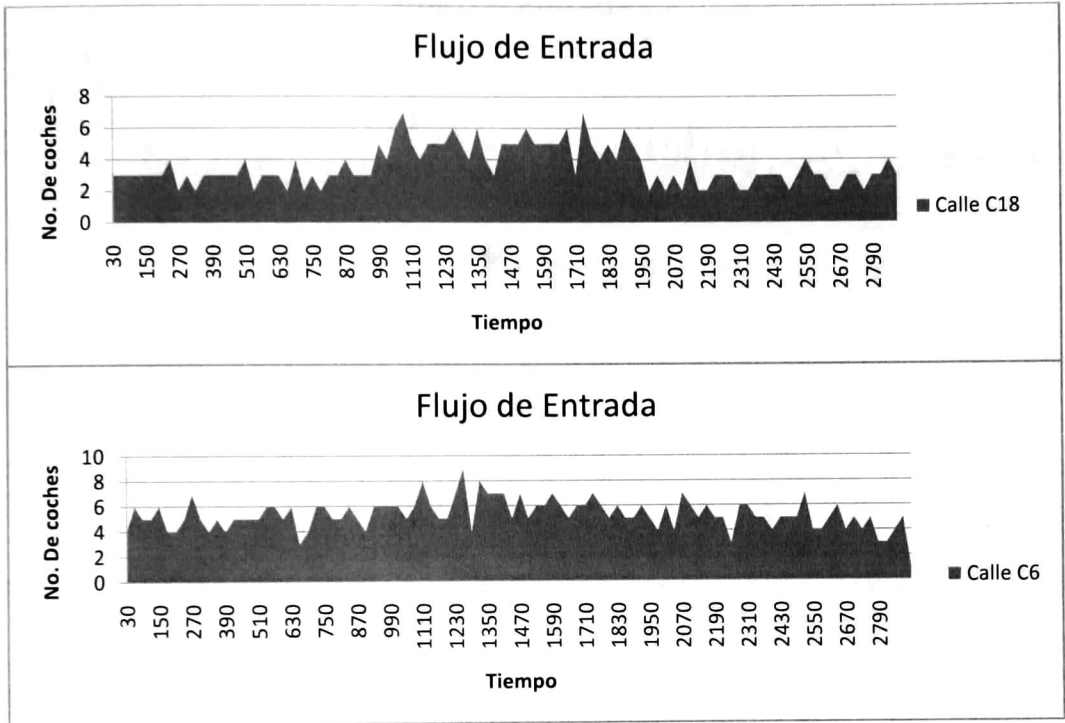


Figura B.3 Flujos de Entrada calles C18, C6.

Flujos de Entrada

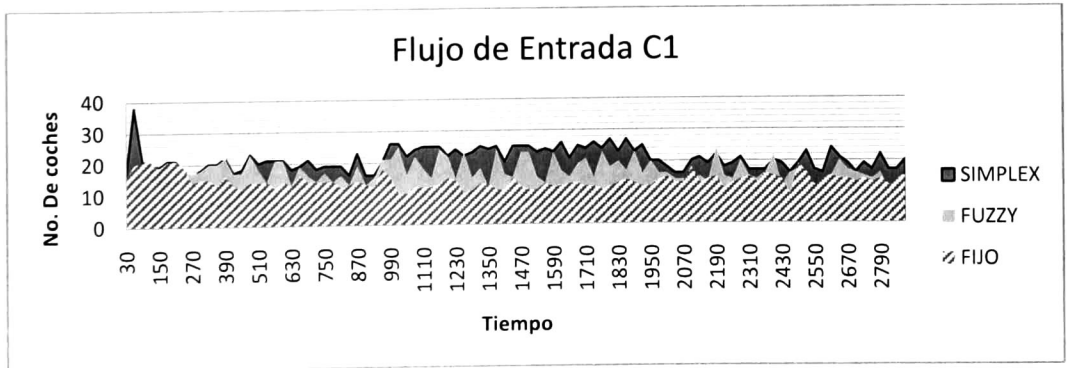


Figura B.4 Flujo de Entrada calle C1.

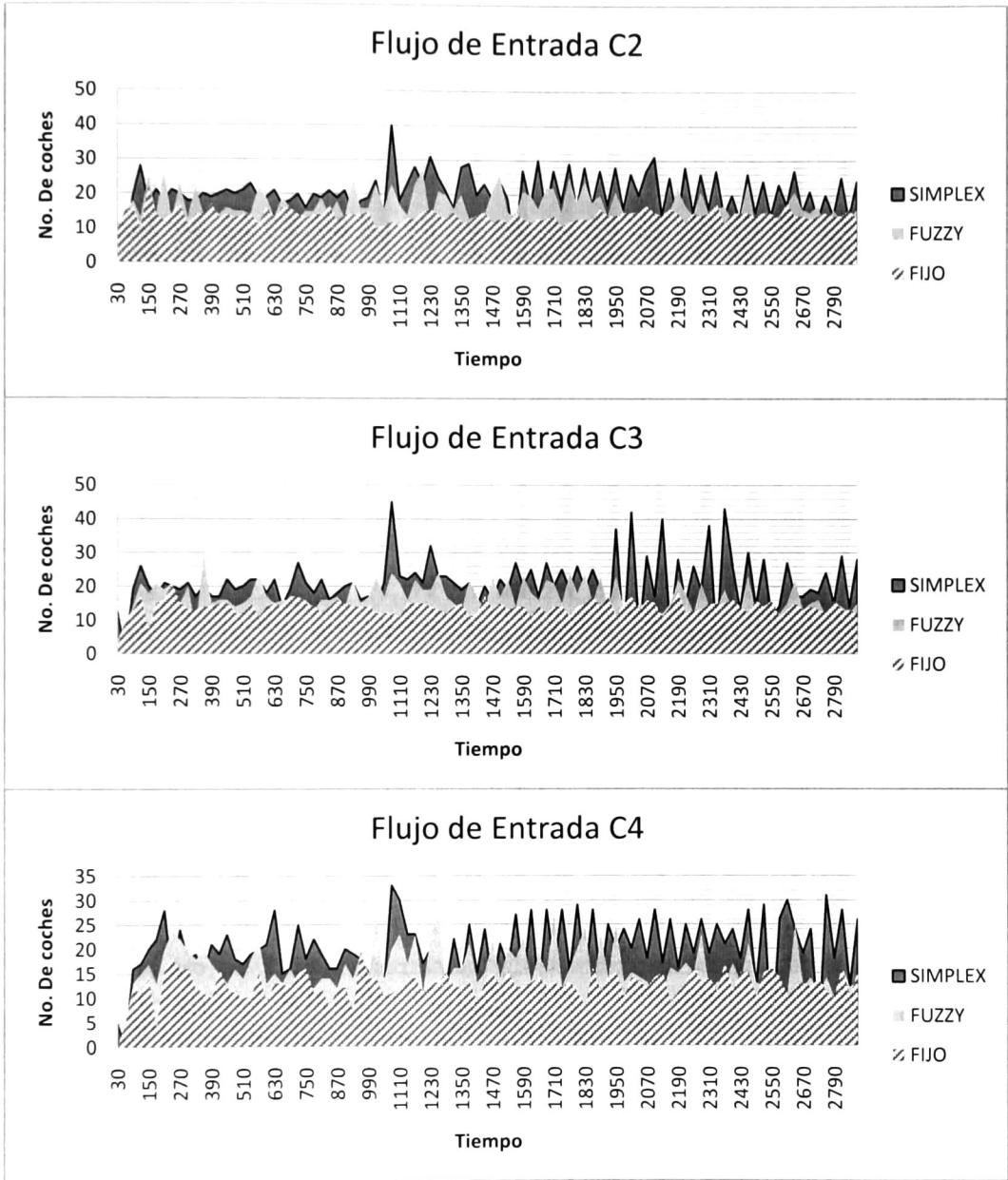


Figura B.5 Flujos de Entrada calles C2, C3, C4.

### B.2.2. Flujo de Salida sin negociación

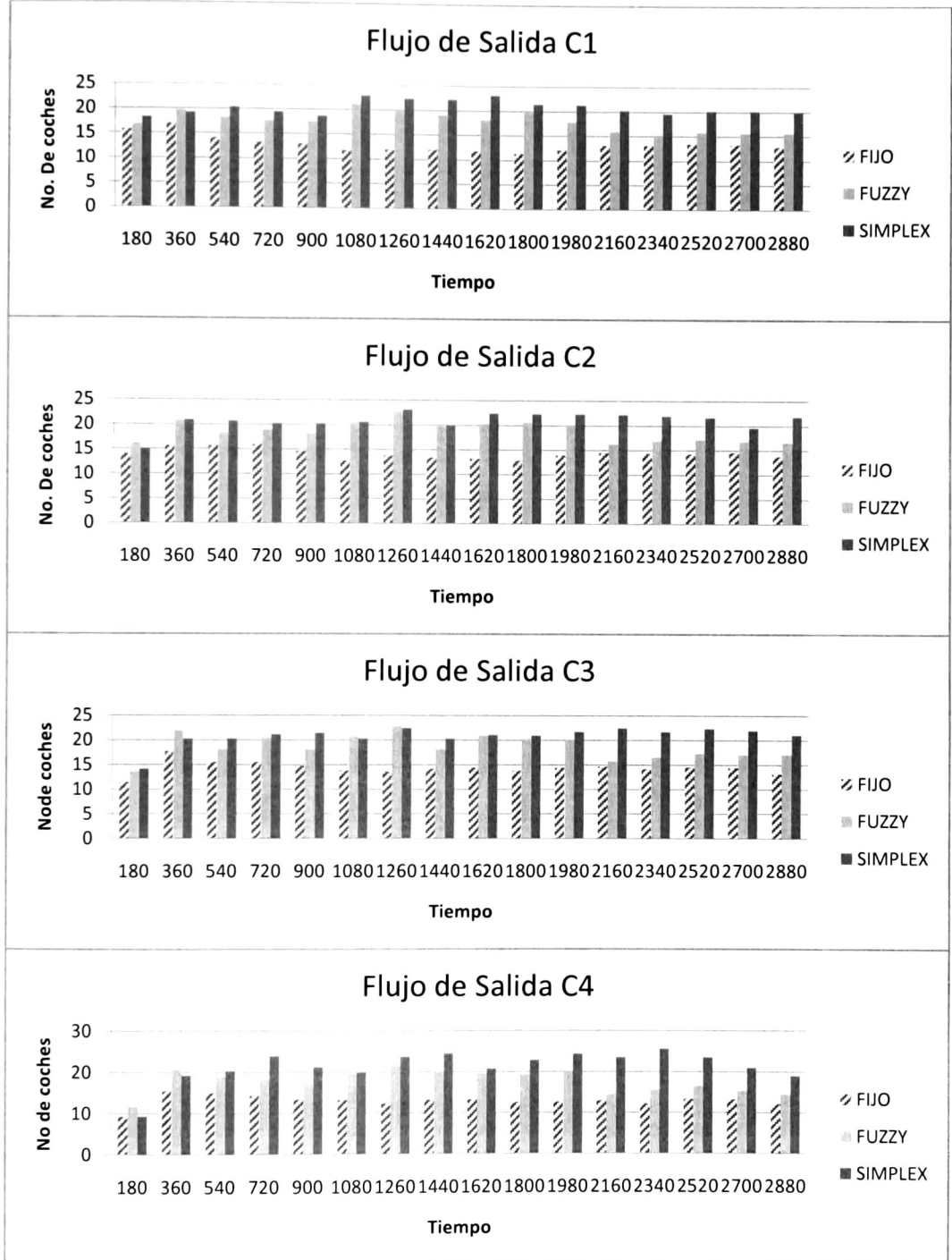


Figura B.6 Flujos de Salida calles C1, C2, C3, C4.



### B.2.3. Tiempos de verde sin negociación

Estas gráficas muestran la suma de los tiempos de verde asignados a cada una de las fases de las intersecciones.

#### Esquema Difuso

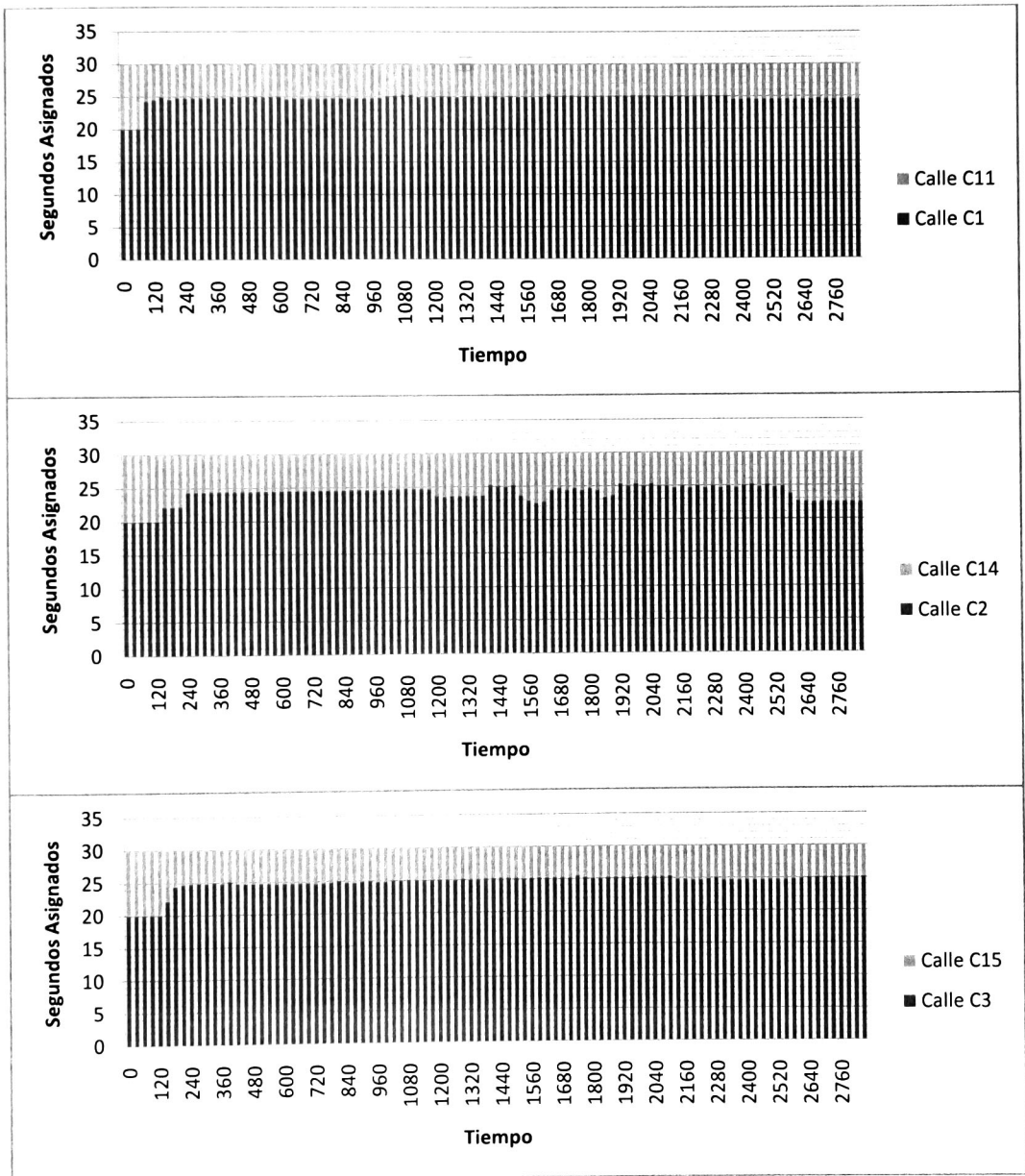


Figura B.7 Tiempos de verde intersecciones 1, 2, 3, esquema difuso.

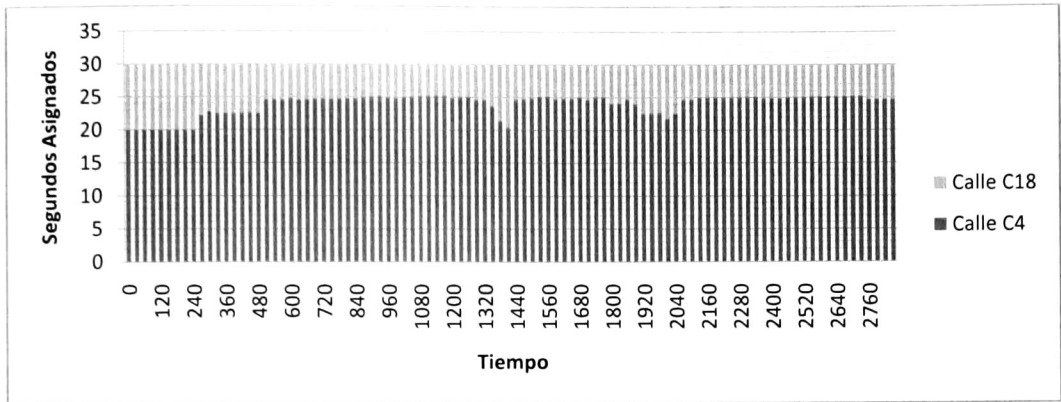


Figura B.8 Tiempos de verde intersección 4, esquema difuso.

### Esquema Programación Lineal

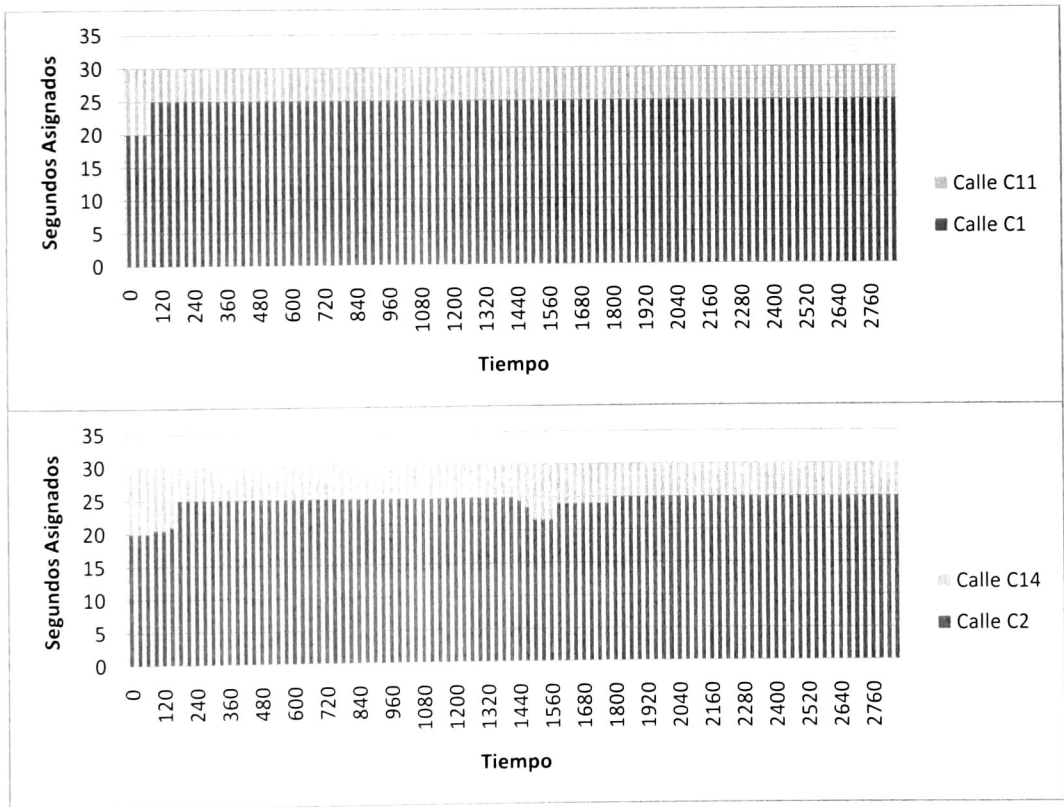


Figura B.9 Tiempos de verde intersecciones 1, 2, esquema PL.

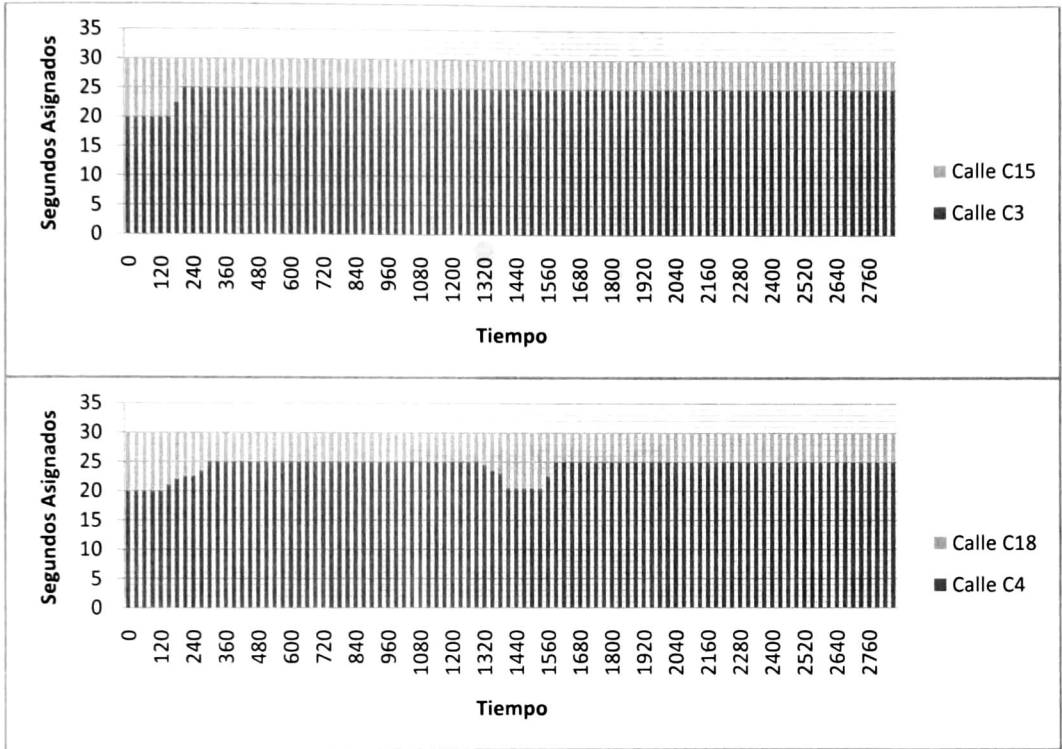


Figura B.10 Tiempos de verde intersecciones 3, 4, esquema PL.

### B.2.4. Flujo de Salida con negociación

#### Esquema Difuso

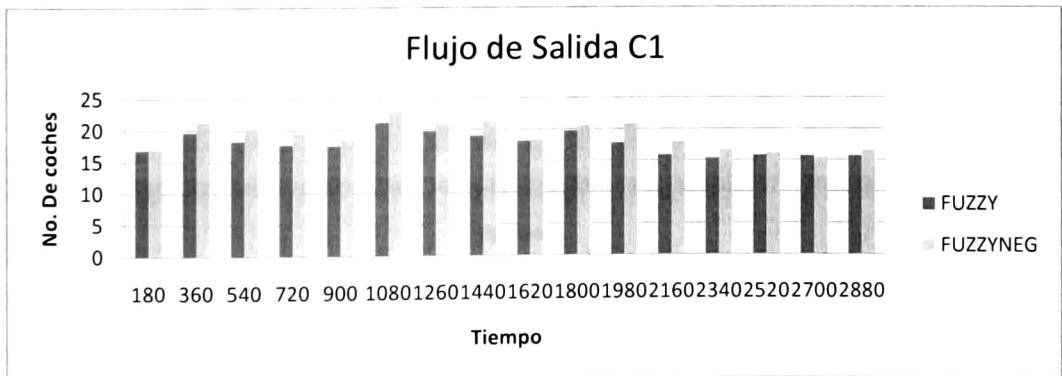


Figura B.11 Flujo de Salida calle C1.

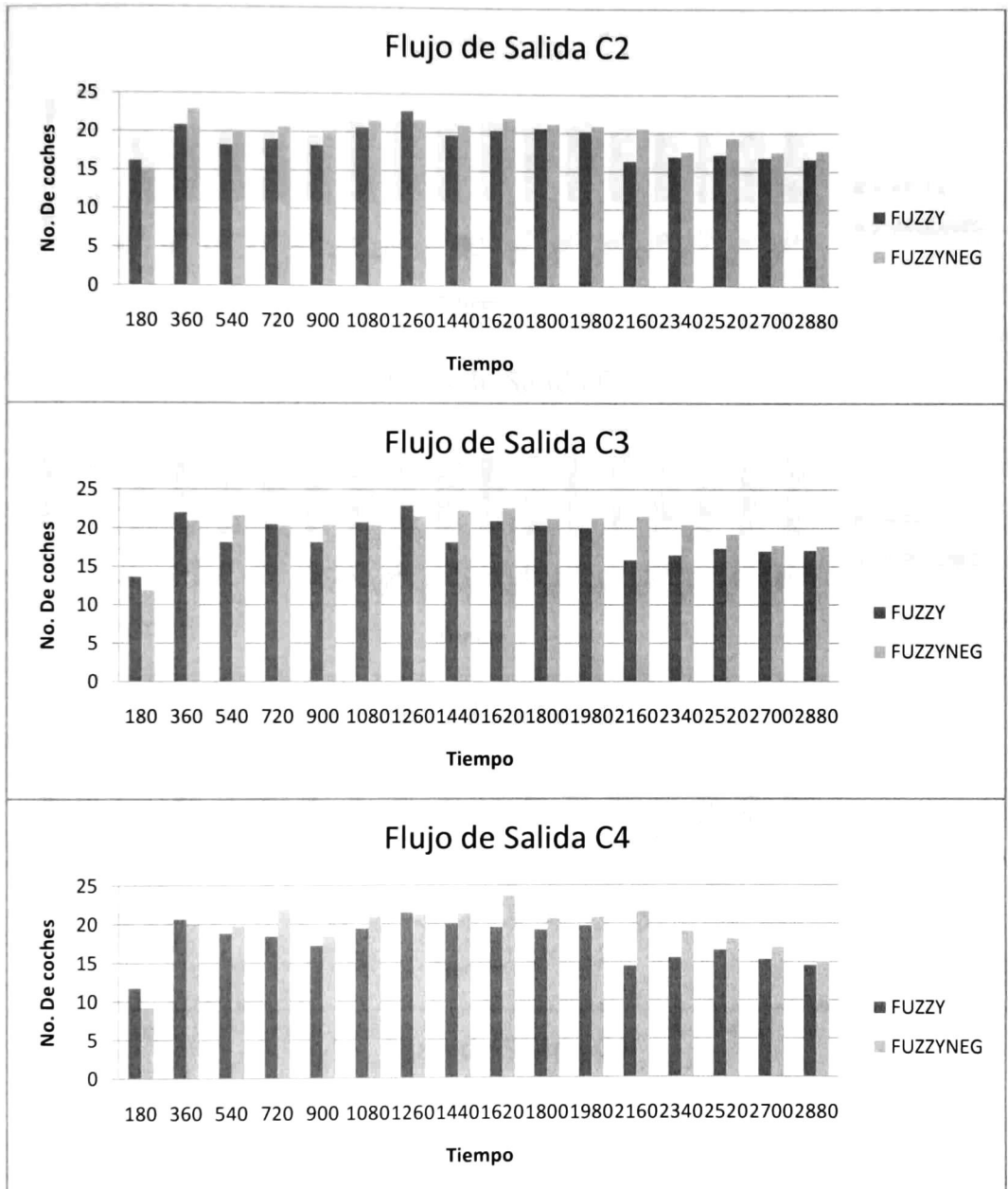


Figura B.12 Flujos de Salida calles C2, C3, C4.

## Esquema Programación Lineal

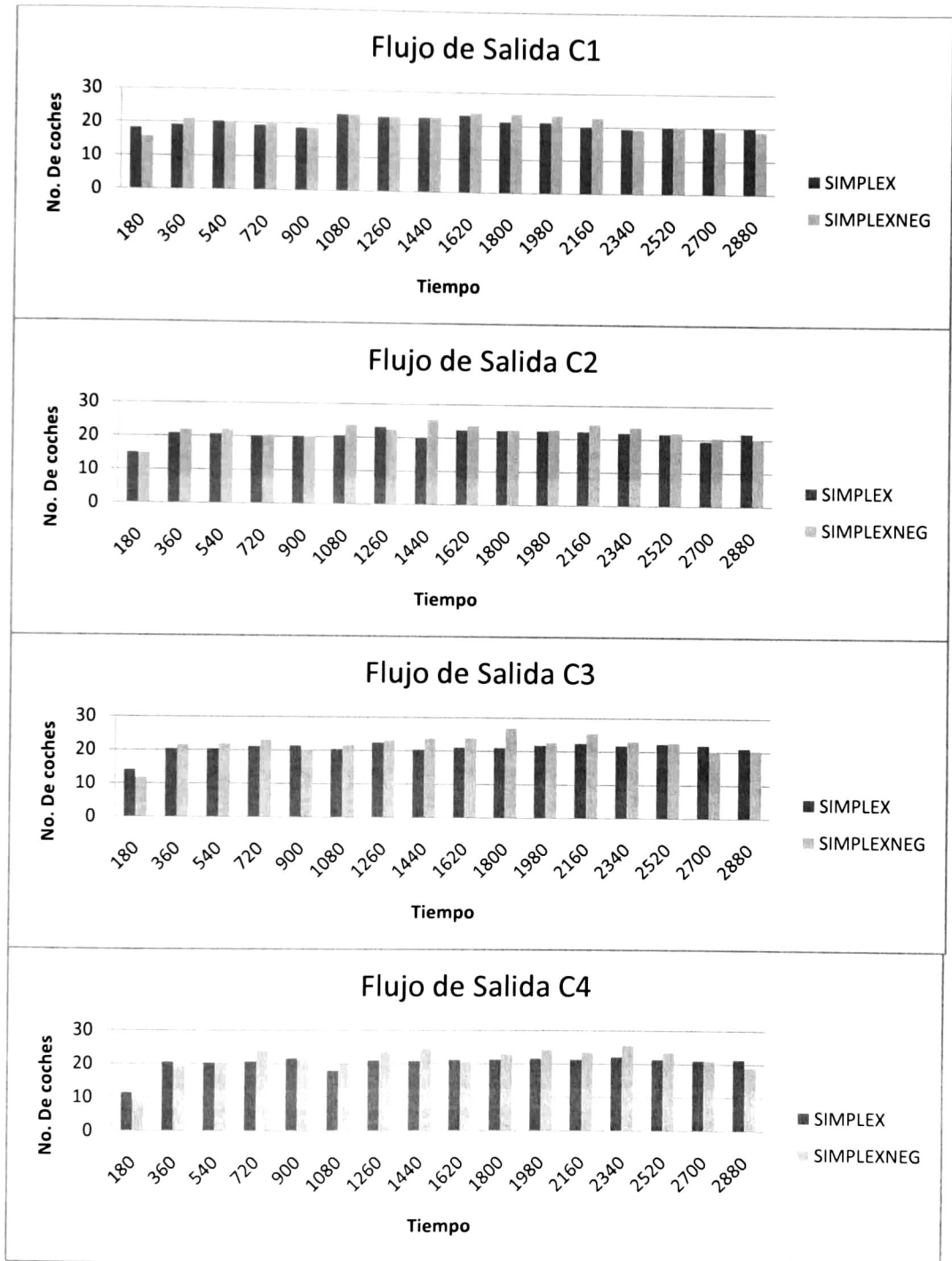


Figura B.13 Flujos de Salida calles C1, C2, C3, C4.



### B.2.5. Tiempos de verde con negociación

Estas gráficas muestran el tiempo de verde asignado a la fase de mayor importancia en el momento en que el agente de control detectó la necesidad de realizar una negociación. Y el tiempo de verde aumentado como resultado de la negociación.

#### Esquema difuso

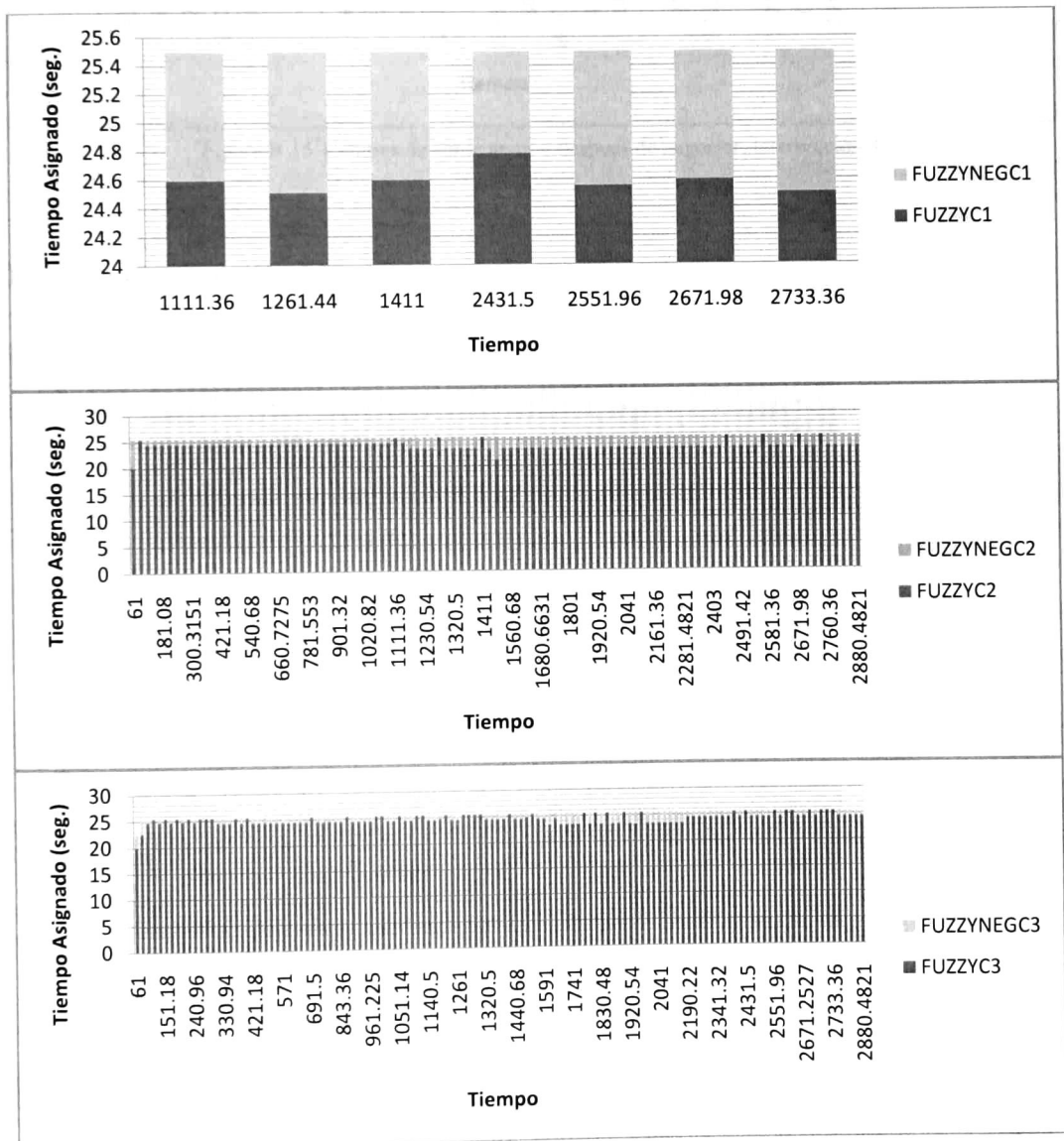


Figura B.14 Tiempos de verde antes y después de negociar, intersecciones 1, 2, 3.

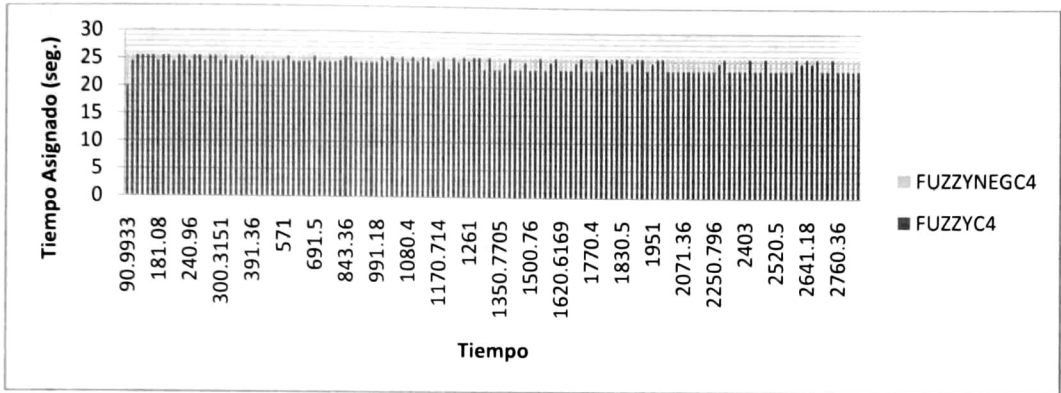


Figura B.15 Tiempos de verde antes y después de negociar, intersección 4.

### Esquema Programación Lineal

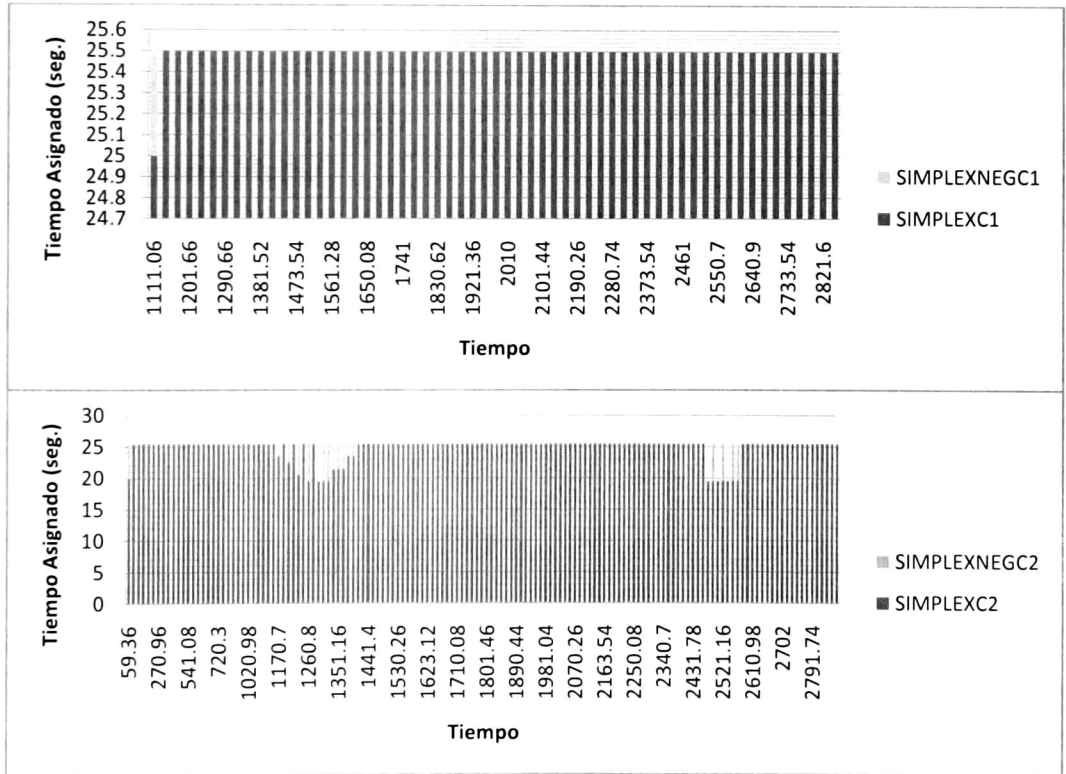


Figura B.16 Tiempos de verde antes y después de negociar, intersecciones 1, 2.

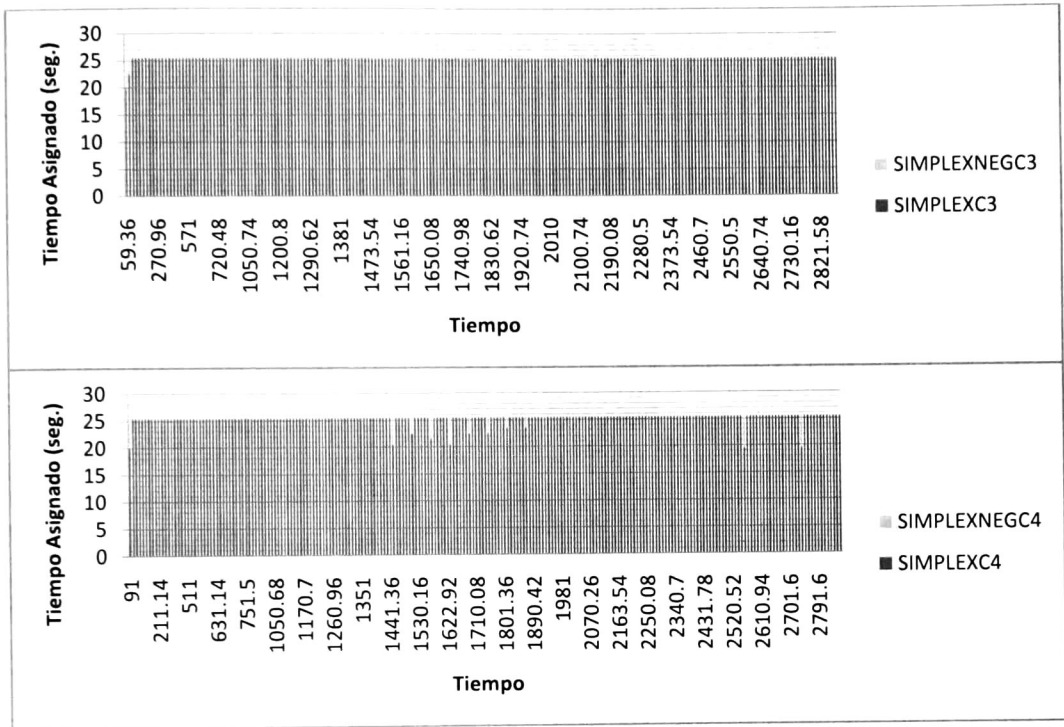


Figura B.17 Tiempos de verde antes y después de negociar, intersecciones 3, 4.

### B.3. Escenario Medio

Resultados de simulaciones para el escenario medio.

#### B.3.1. Llegadas de coches

Flujos de Entrada desde archivos

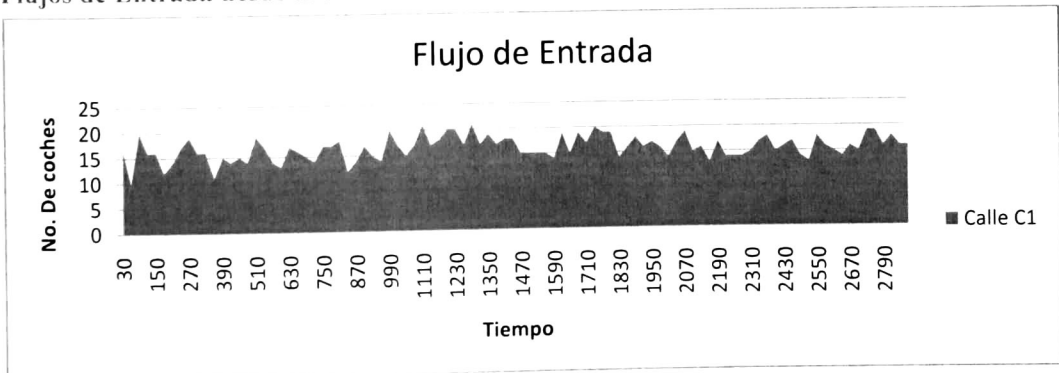


Figura B.18 Flujo de Entrada calle C1.

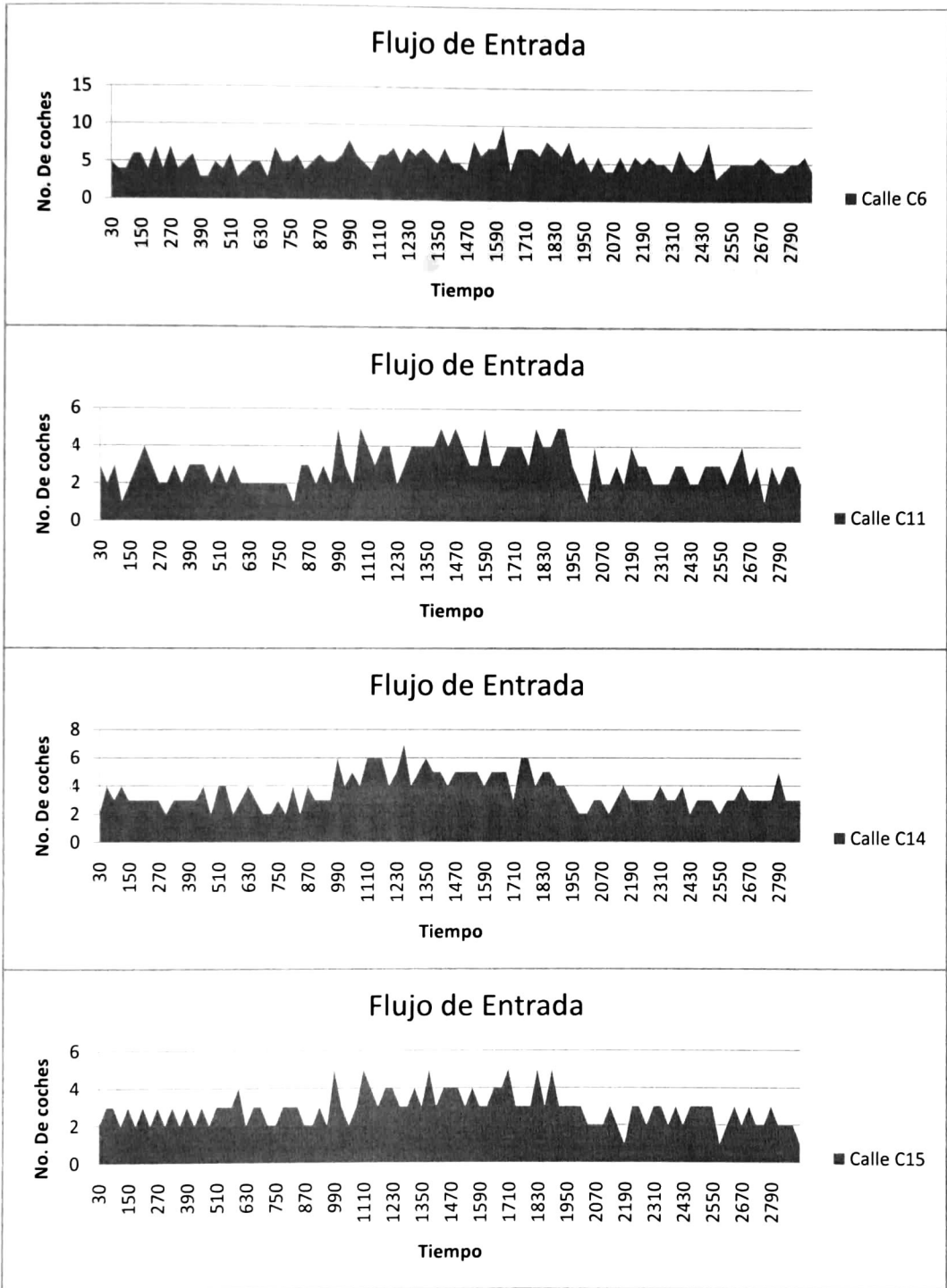


Figura B.19 Flujos de Entrada calles C6, C11, C14, 15.

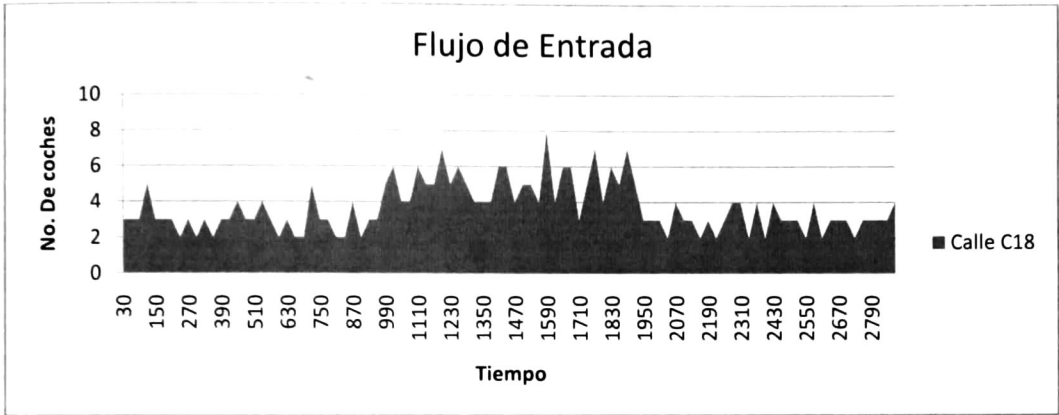


Figura B.20 Flujo de Entrada calle C18.

## Flujos de Entrada

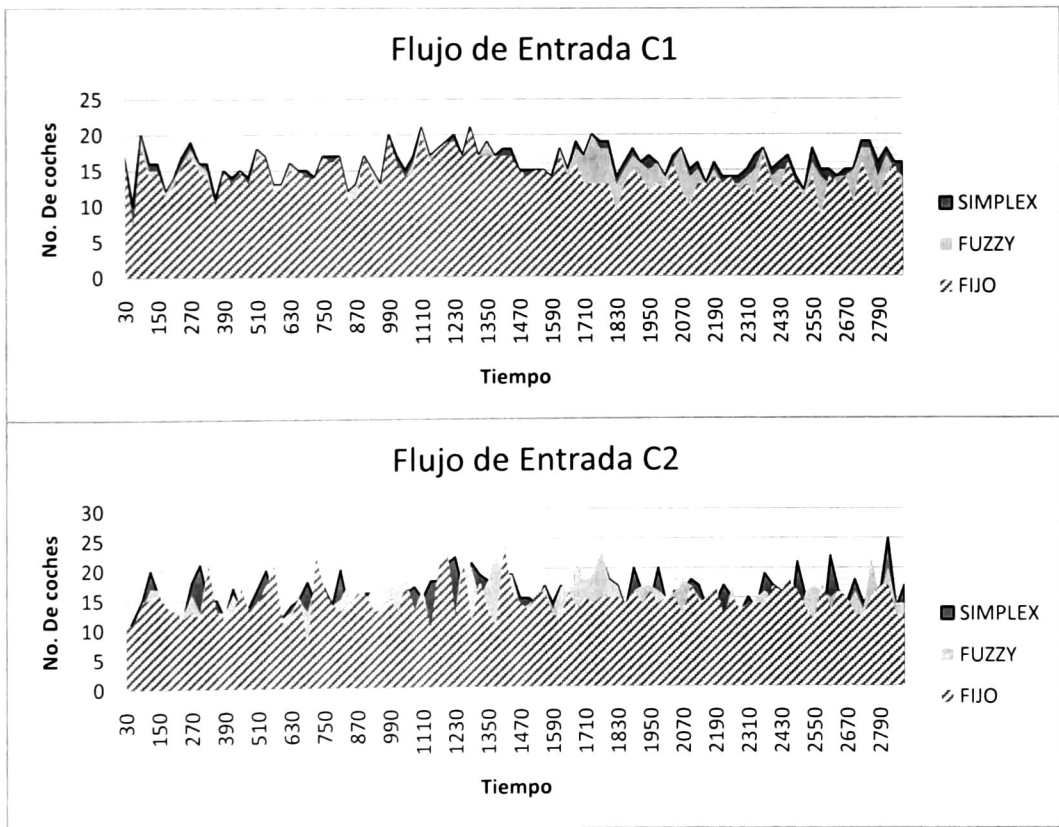


Figura B.21 Flujos de Entrada calles C1, C2.

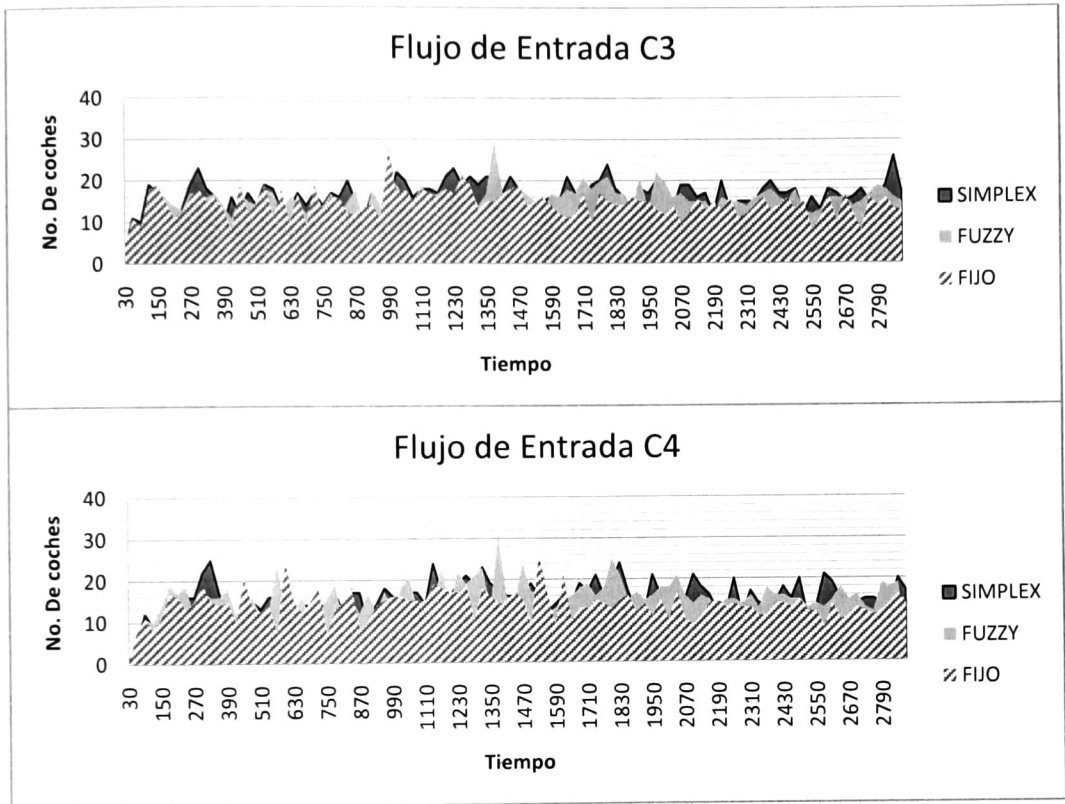


Figura B.22 Flujos de Entrada calles C3, C4.

**B.3.2. Flujo de Salida sin negociación**

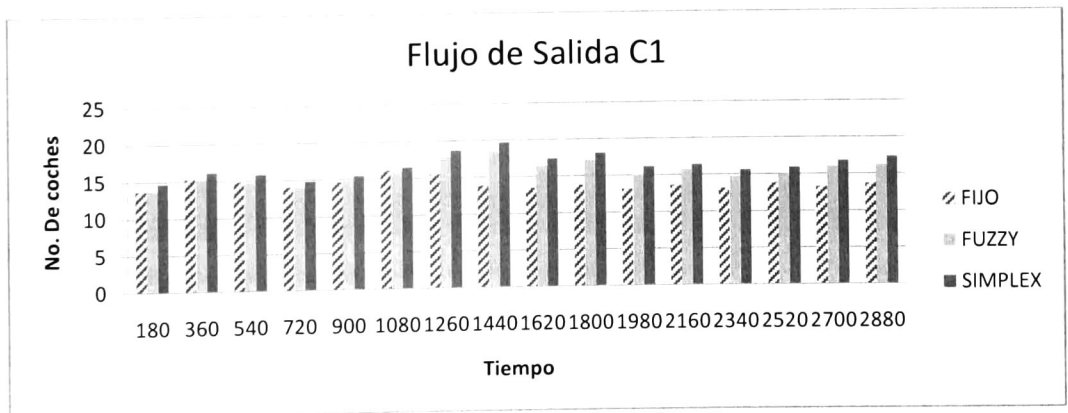


Figura B.23 Flujo de Salida calle C1.



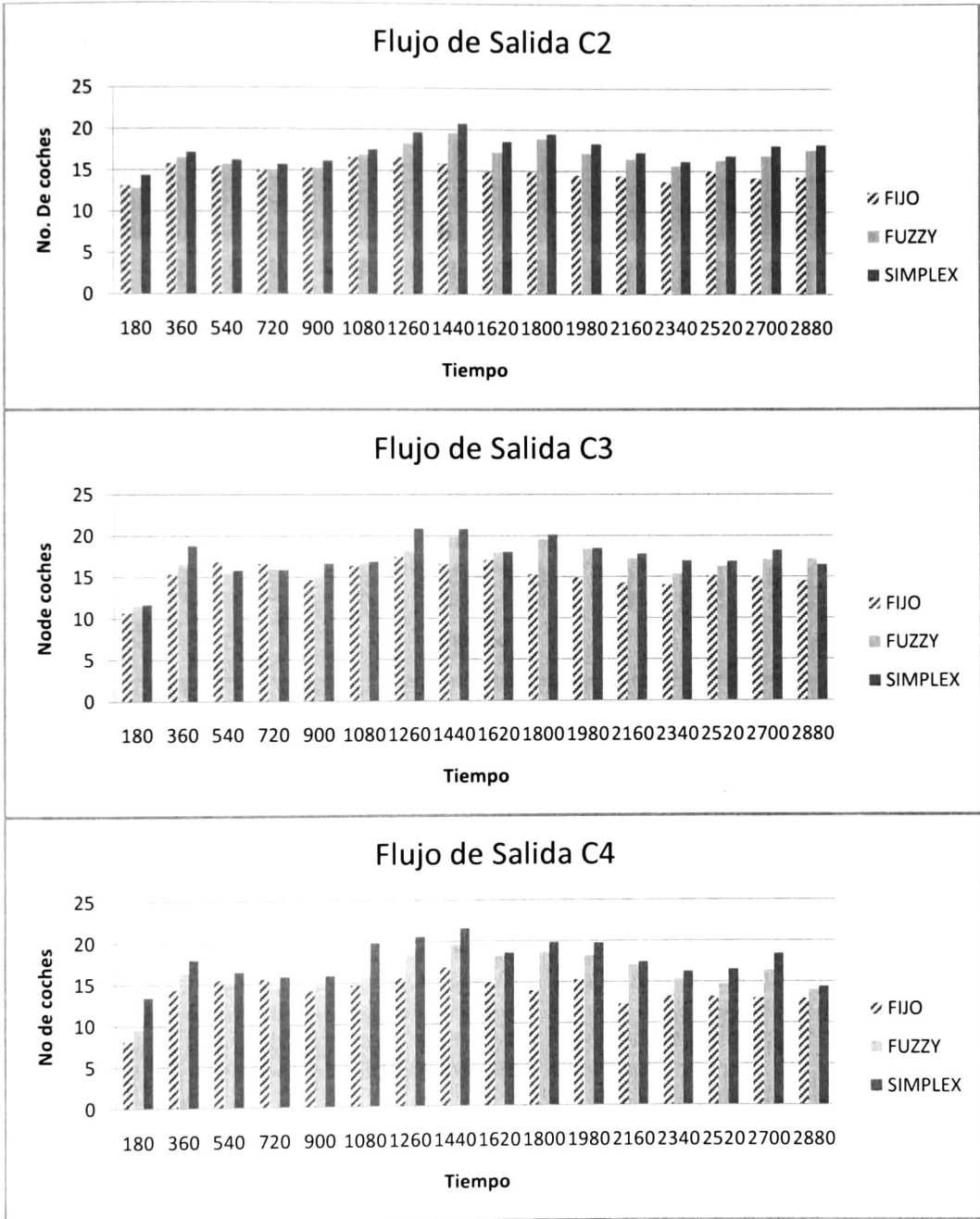


Figura B.24 Flujos de Salida calle C2, C3, C4.

### B.3.3. Tiempos de verde sin negociación

#### Esquema Difuso

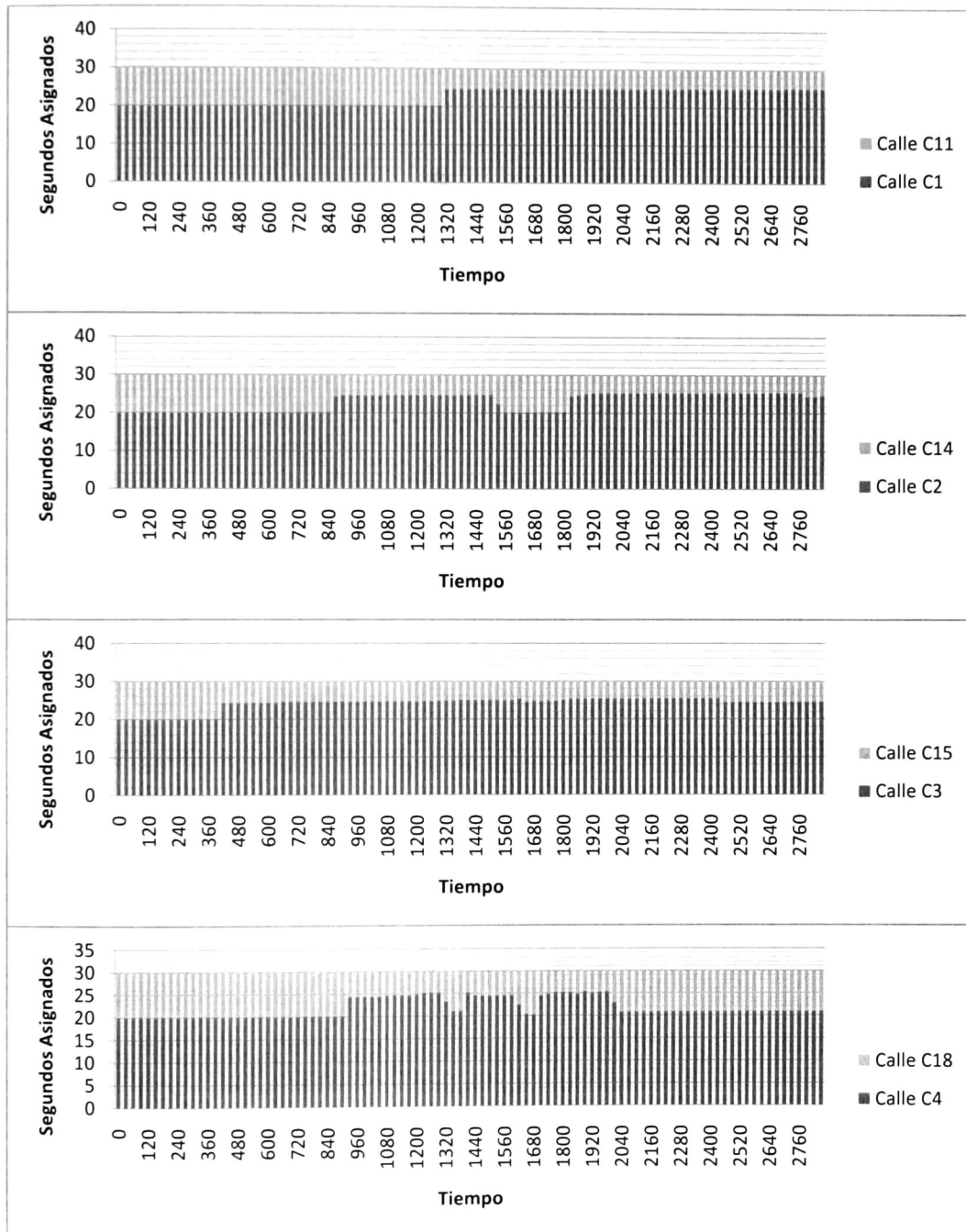


Figura B.25 Tiempos de verde intersecciones 1, 2, 3, 4, esquema difuso.

## Esquema Programación Lineal

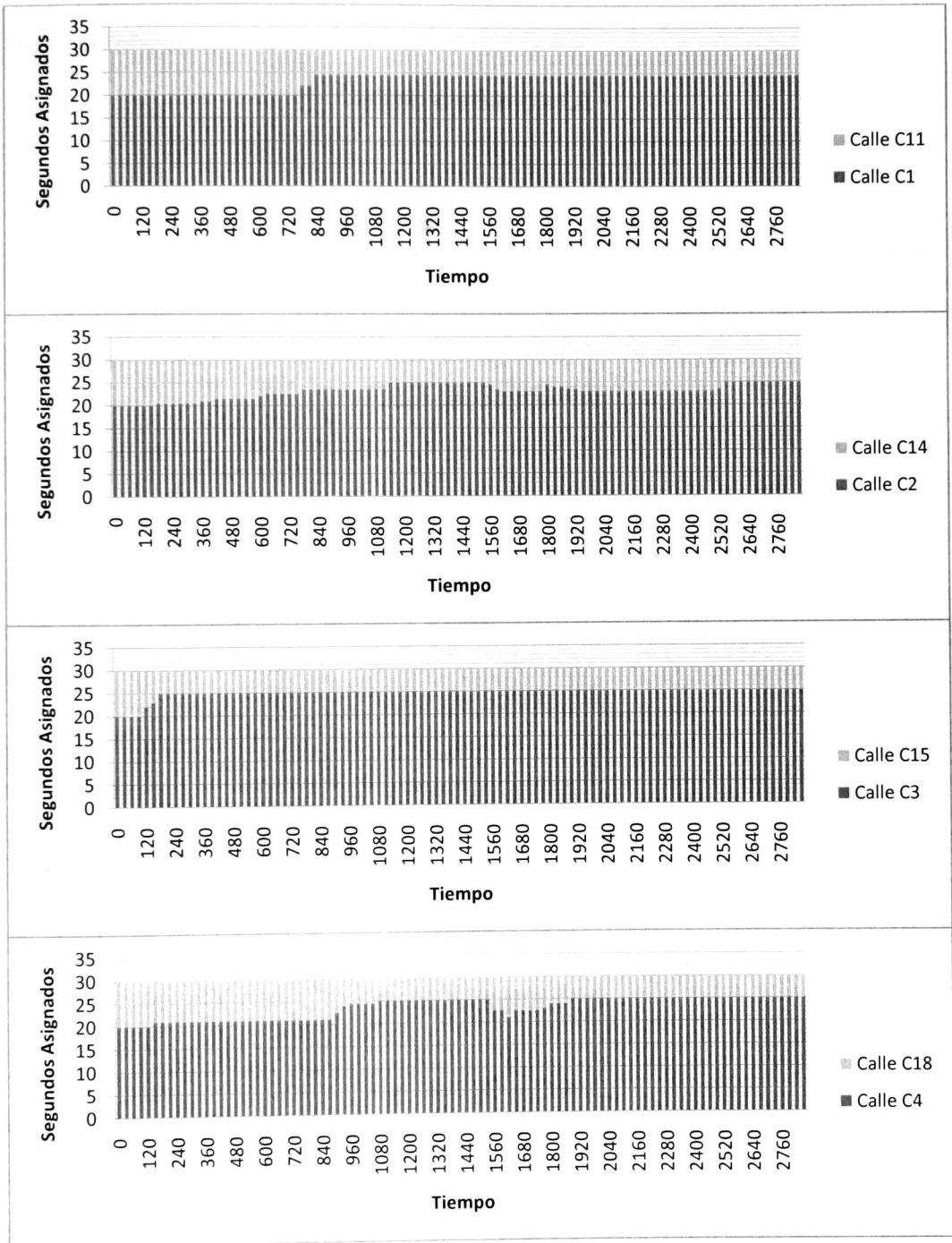


Figura B.26 Tiempos de verde intersecciones 1, 2, 3, 4, esquema PL.

### B.3.4. Flujo de Salida con negociación

#### Esquema Difuso

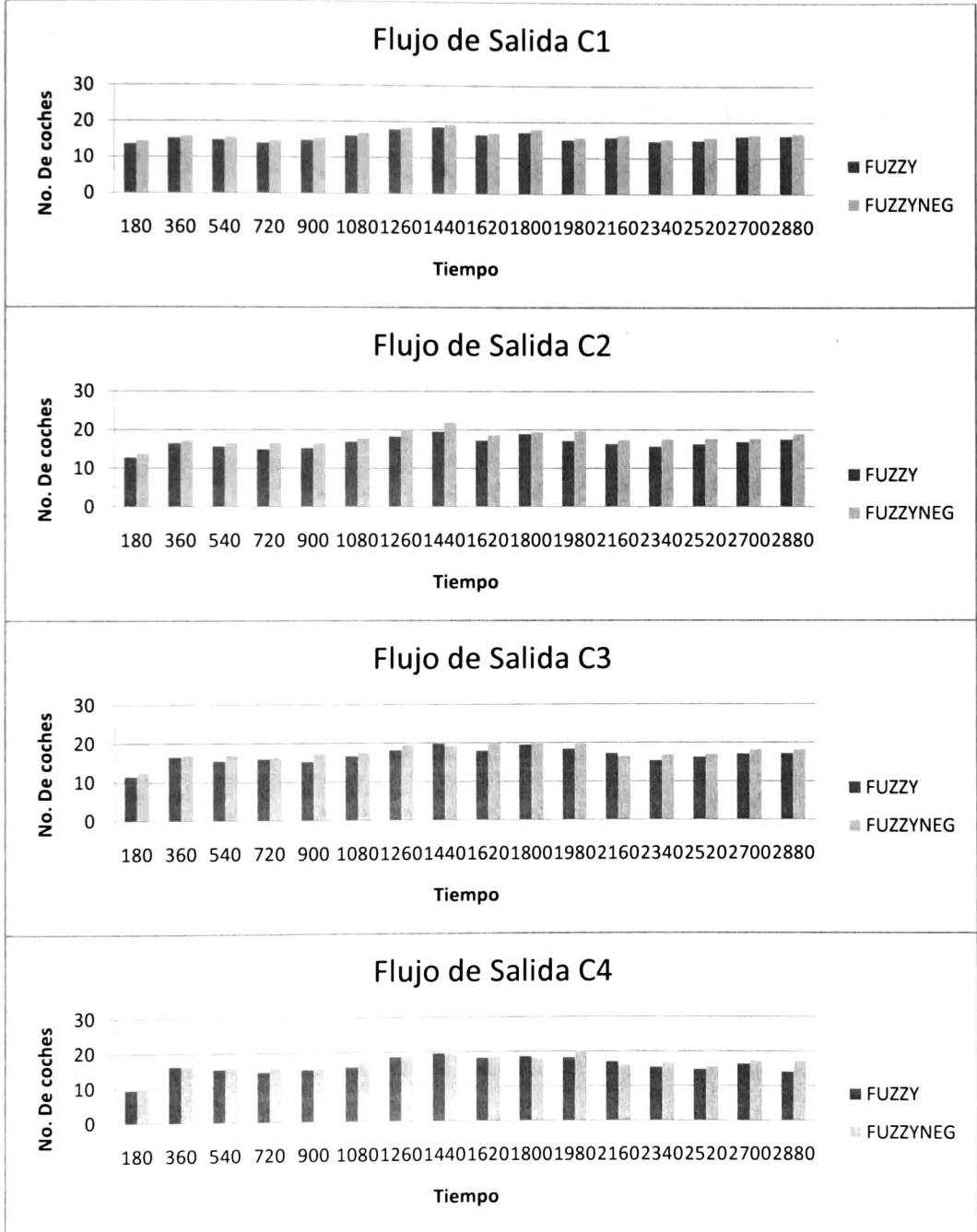


Figura B.27 Flujos de Salida calles C1, C2, C3, C4.

**Esquema Programación Lineal.**

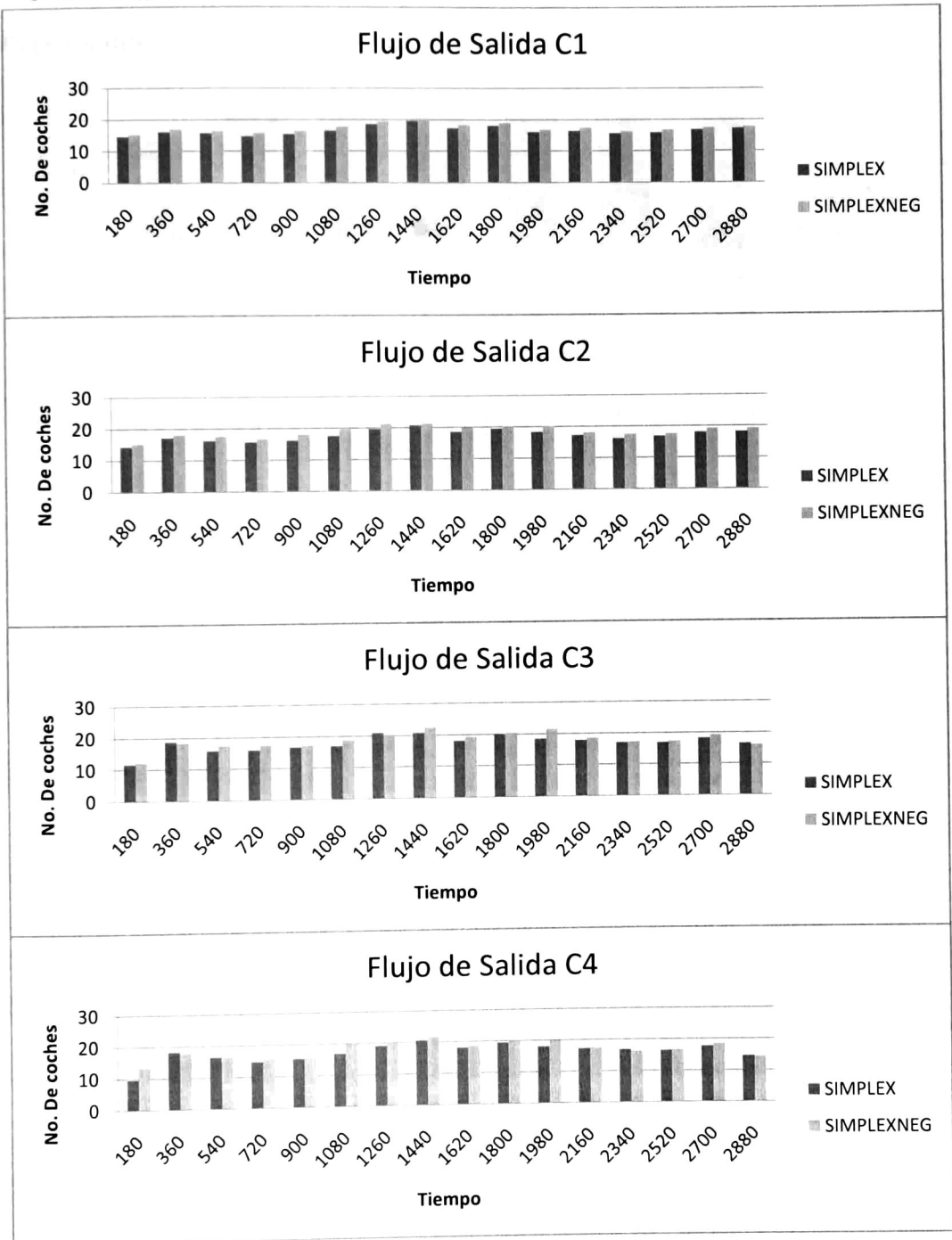


Figura B.28 Flujos de Salida calles C1, C2, C3, C4.

### B.3.5. Tiempos de verde con negociación

#### Esquema Difuso

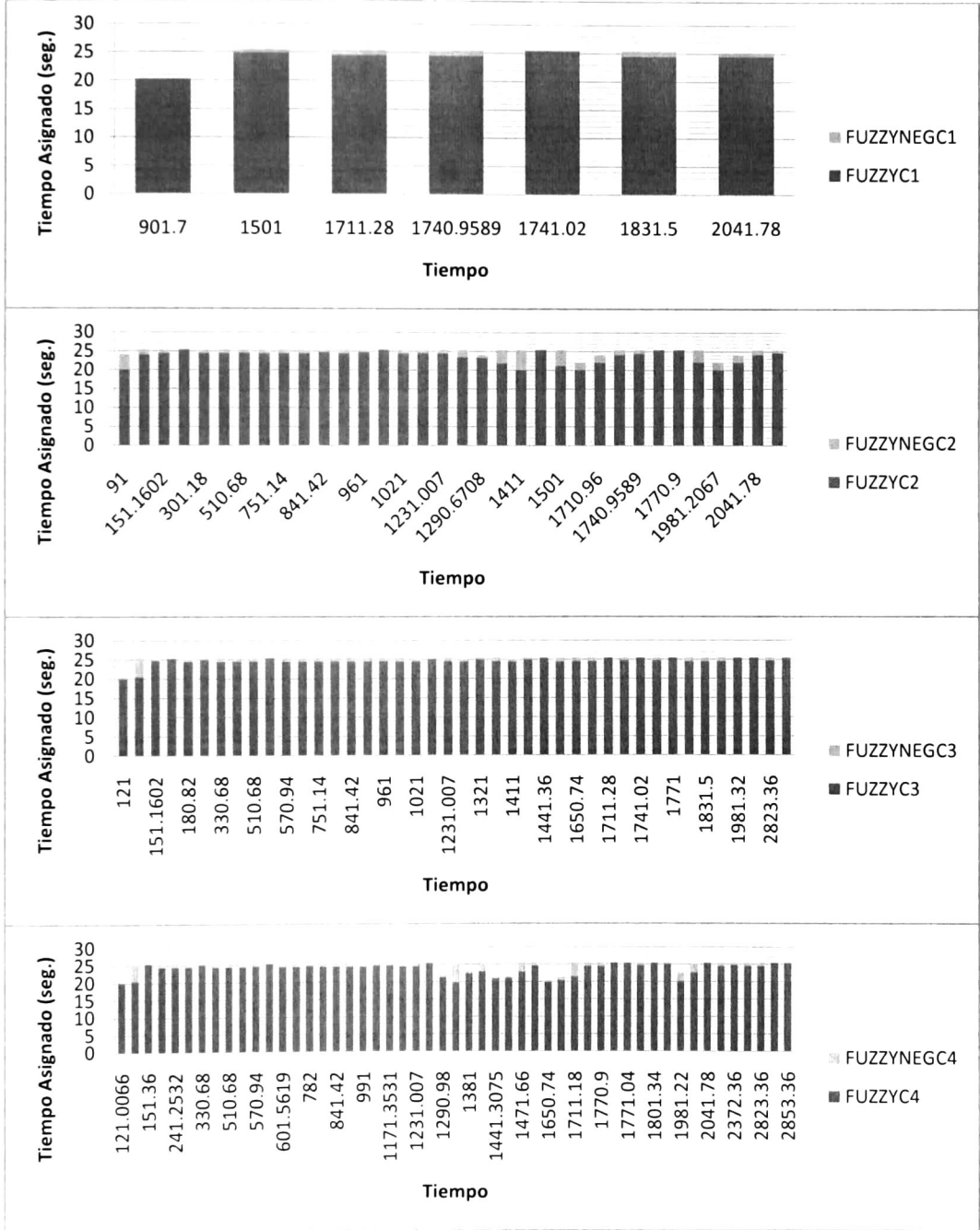


Figura B.29 Tiempos de verde antes y después de negociar, intersecciones 1, 2, 3, 4, esquema difuso.



### Esquema Programación Lineal

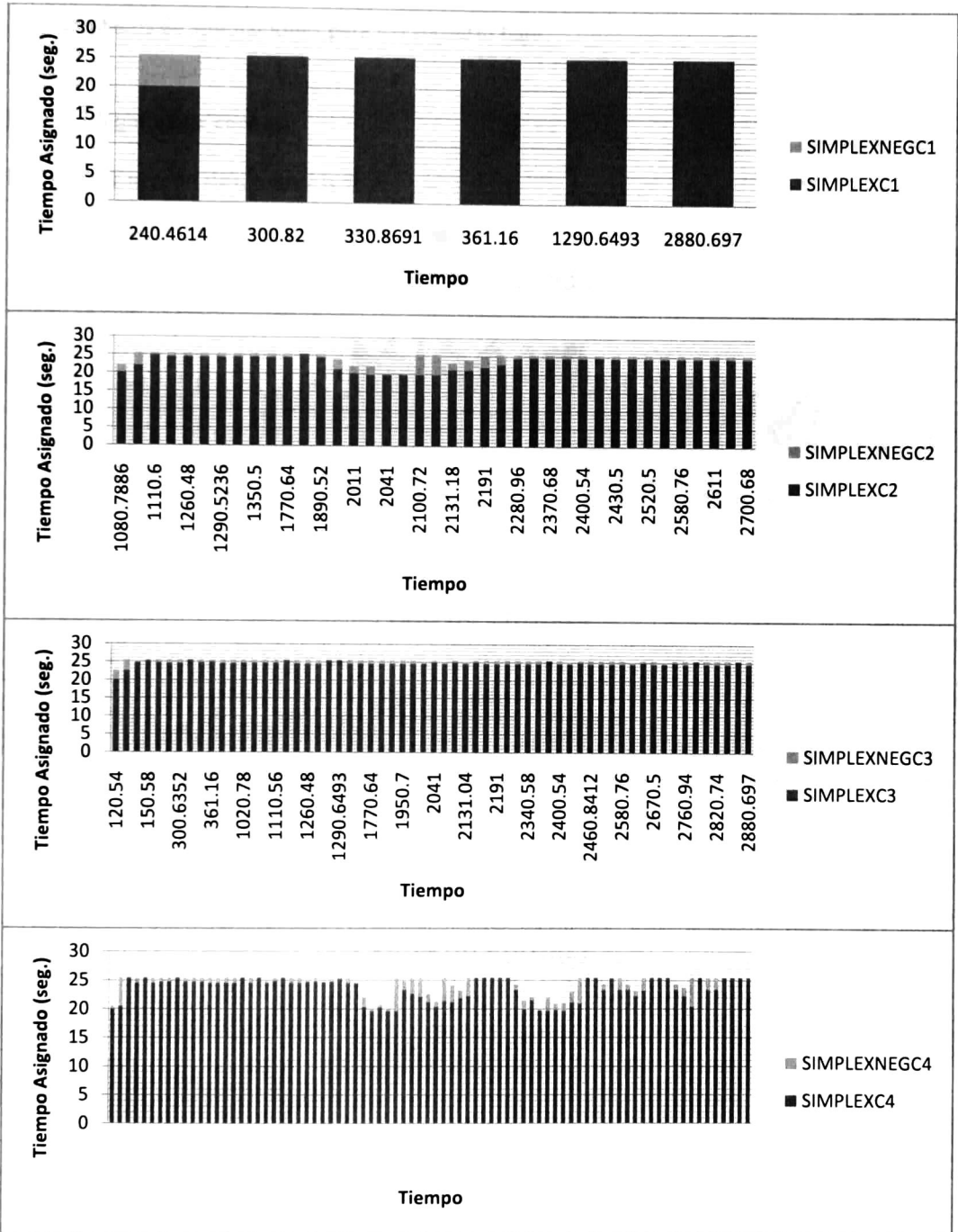


Figura B.30 Tiempos de verde antes y después de negociar intersecciones 1, 2, 3, 4, esquema PL.

## B.4. Escenario Bajo

Resultados de simulaciones para el escenario bajo.

### B.4.1. Llegadas de coches

Flujos de Entrada desde archivos

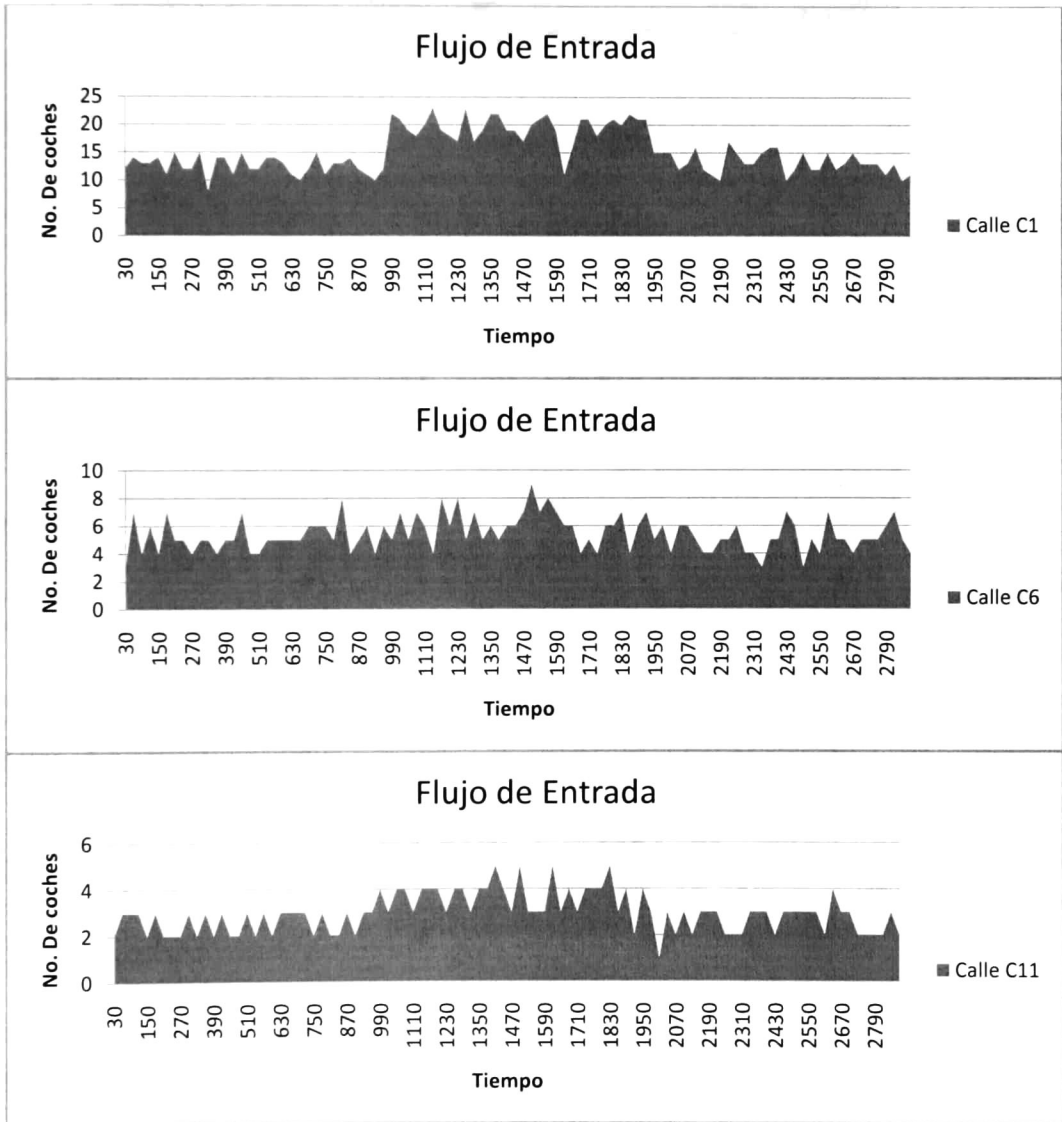


Figura B.31 Flujos de Entrada calles C1, C6, C11.

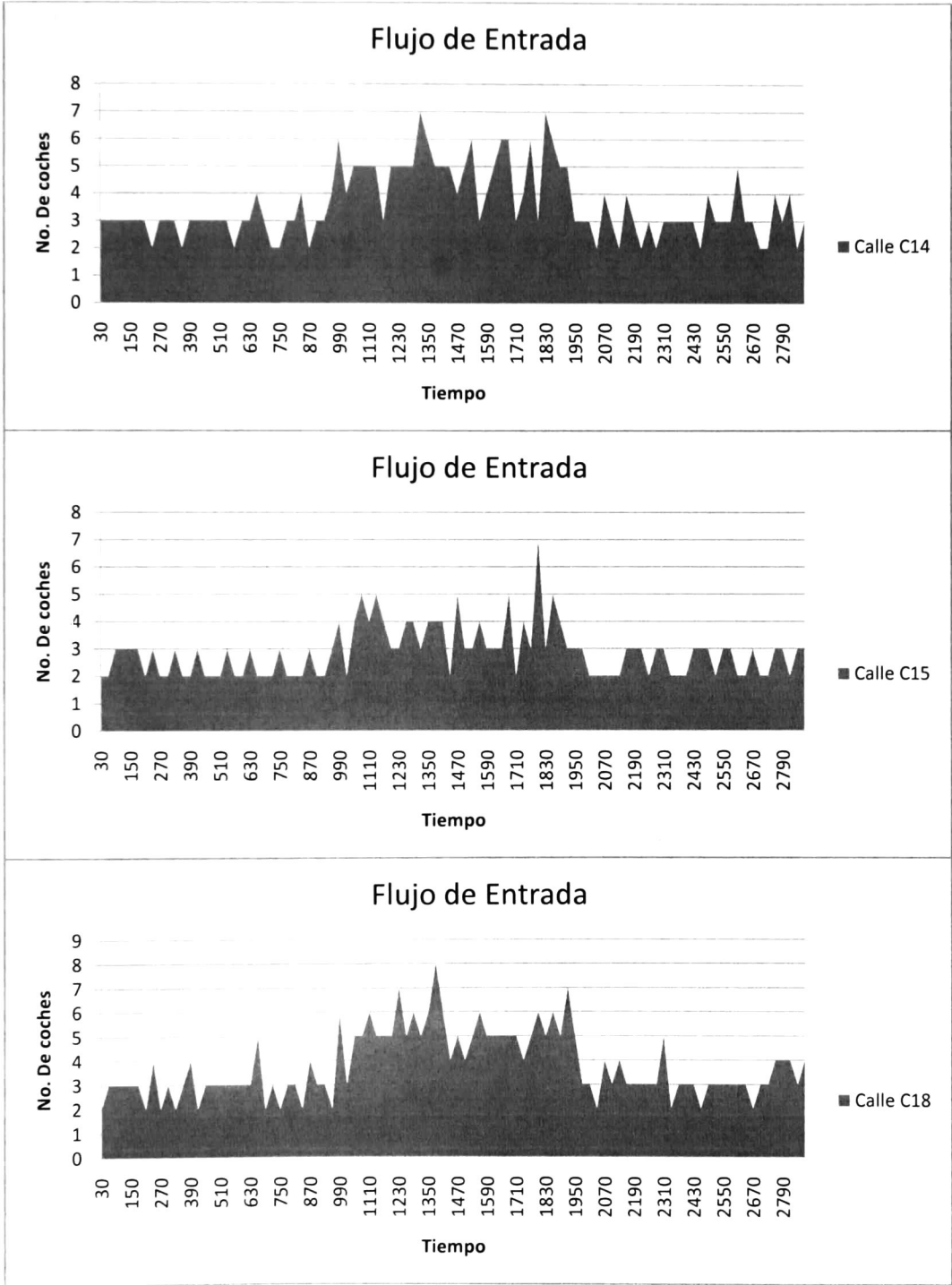


Figura B.32 Flujos de Entrada calles C12, C15, C18.

## Flujos De Entrada

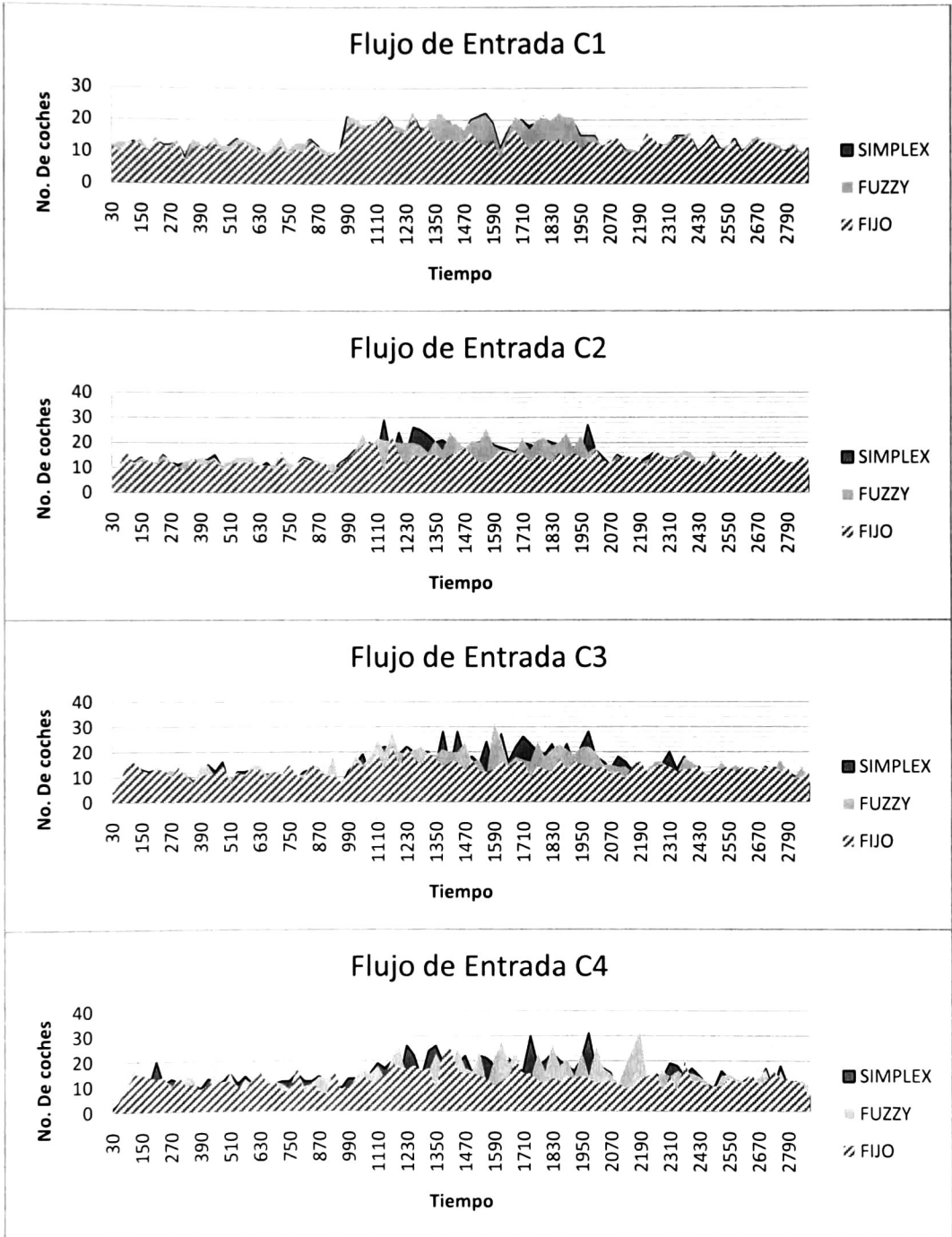


Figura B.33 Flujos de Entrada calles C1, C2, C3, C4.

**B.4.2. Flujo de Salida sin negociación**

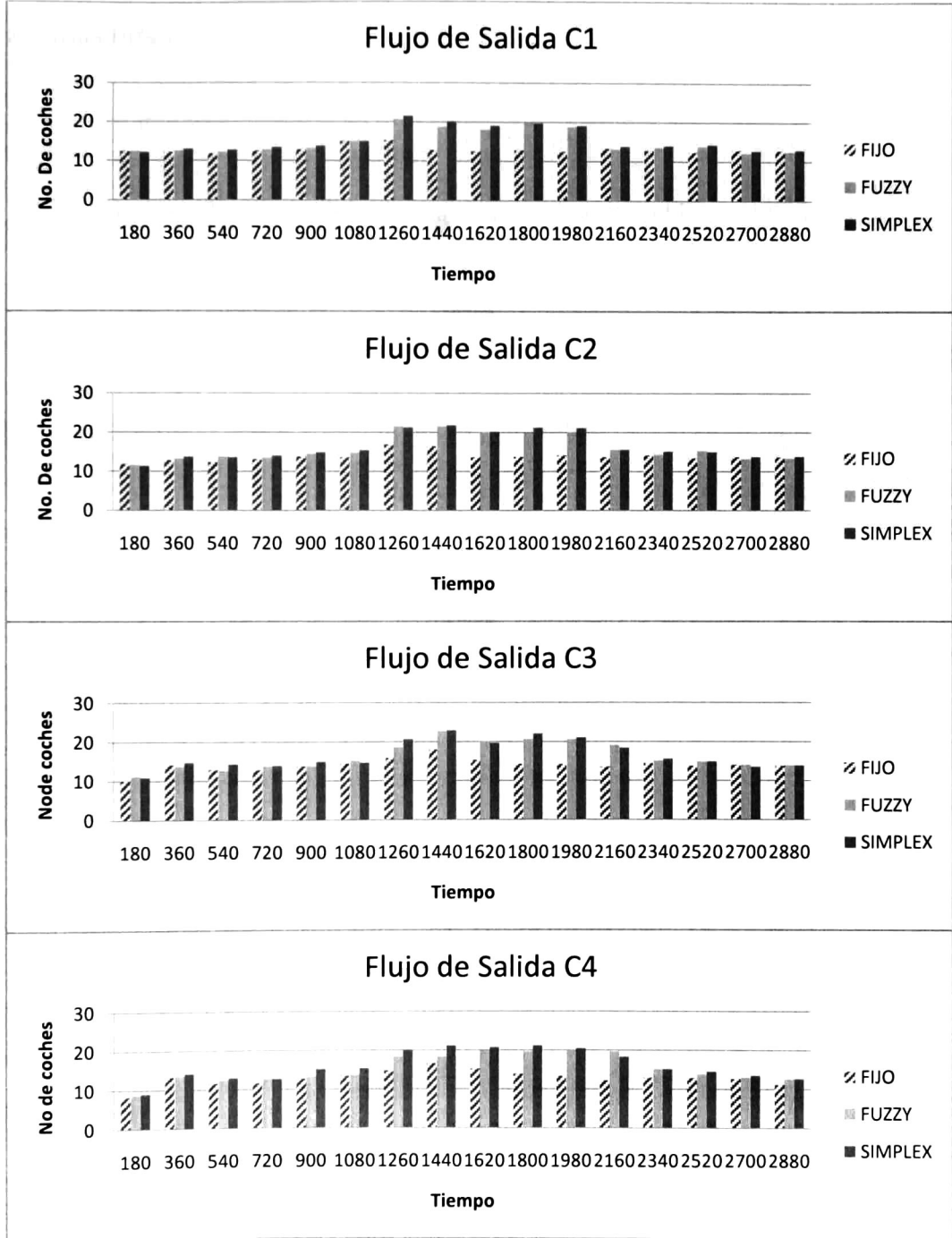


Figura B.34 Flujos de Salida calles C1, C2, C3, C4.

### B.4.3. Tiempos de verde sin negociación

#### Esquema Difuso

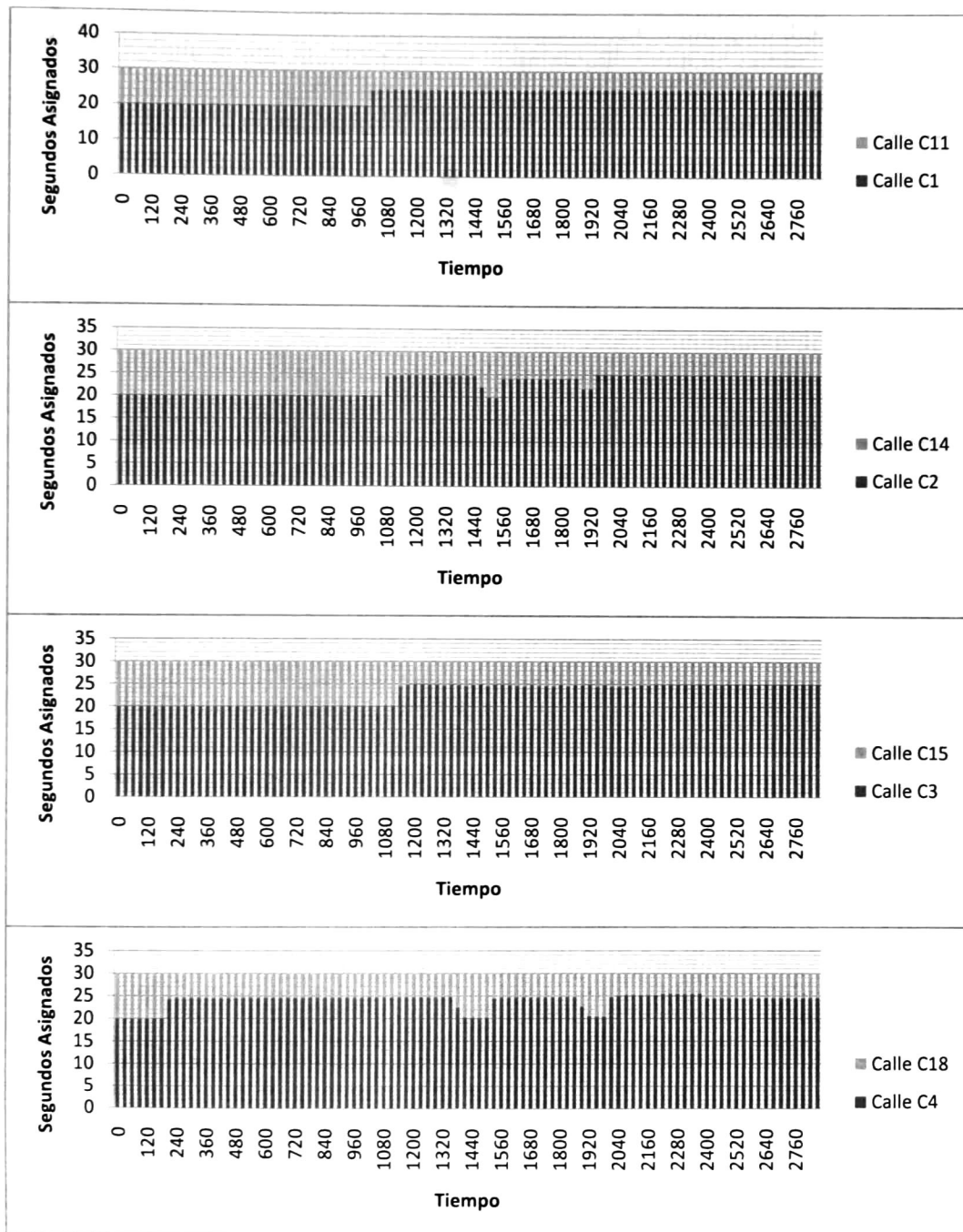


Figura B.35 Tiempos de verde intersecciones 1, 2, 3, 4, esquema difuso.



### Esquema Programación Lineal

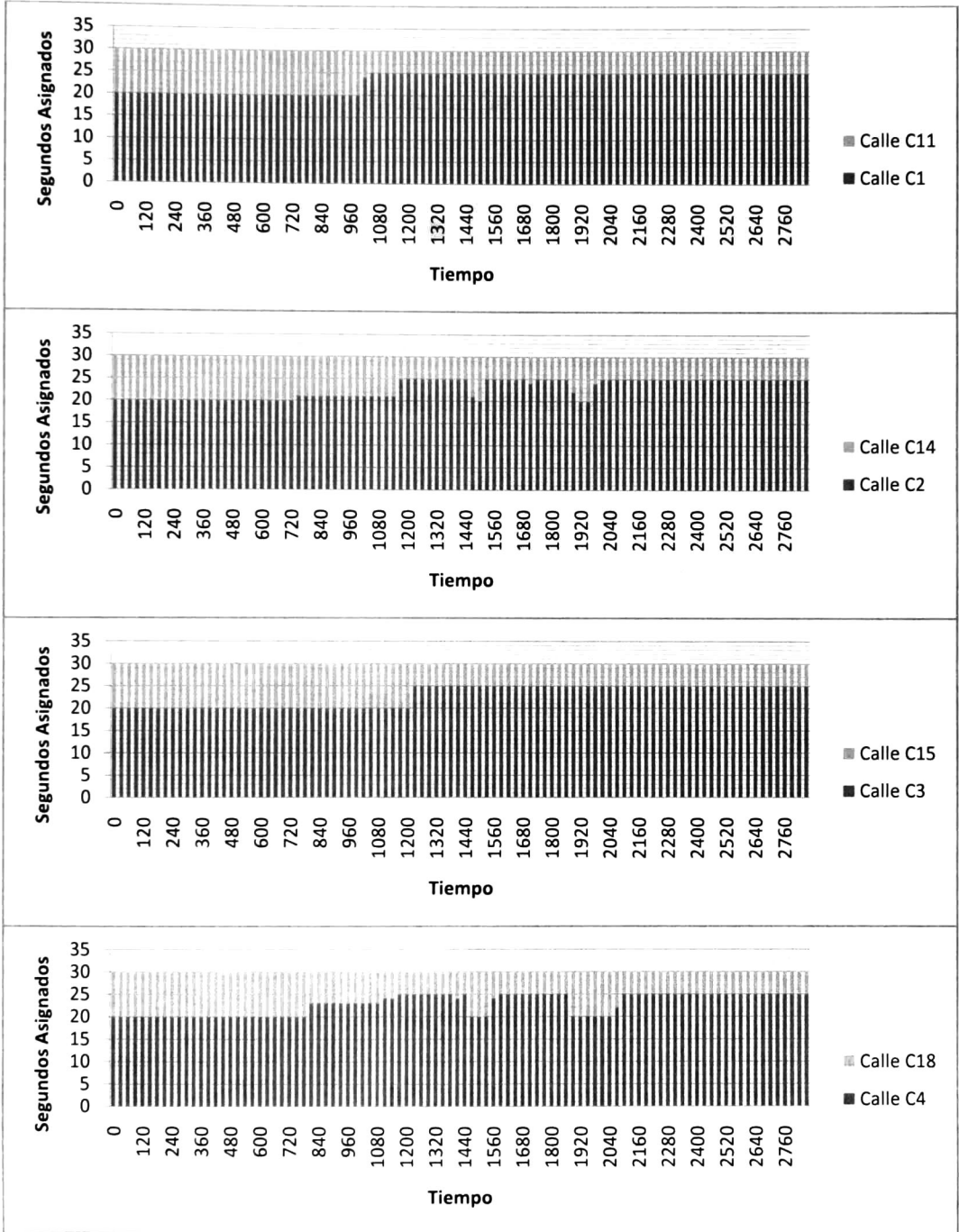


Figura B.36 Tiempos de verde intersecciones 1, 2, 3, 4, esquema PL.

### B.4.4. Flujo de Salida con negociación

#### Esquema Difuso

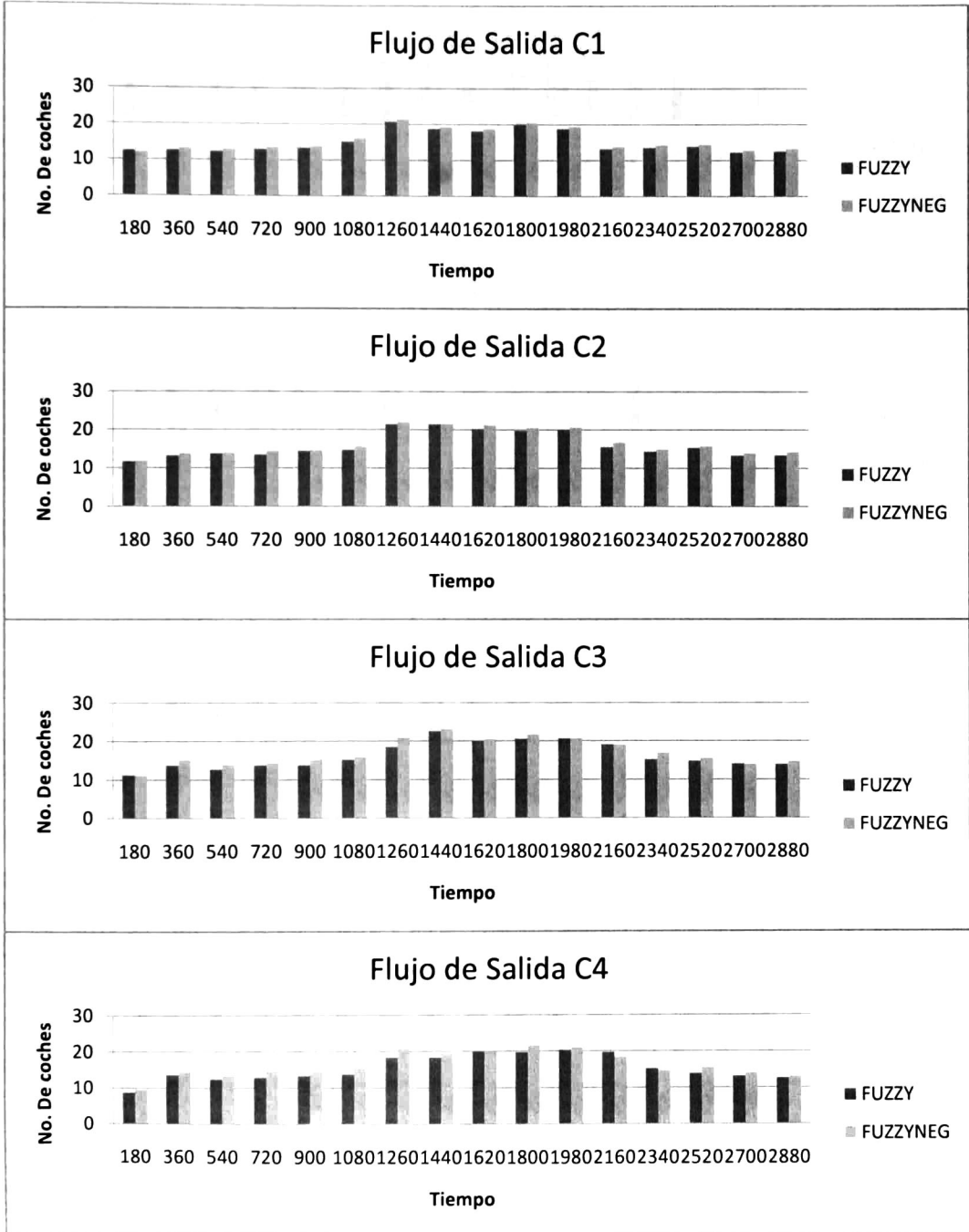


Figura B.37 Flujos de Salida calles C1, C2, C3, C4.

Esquema Programación Lineal

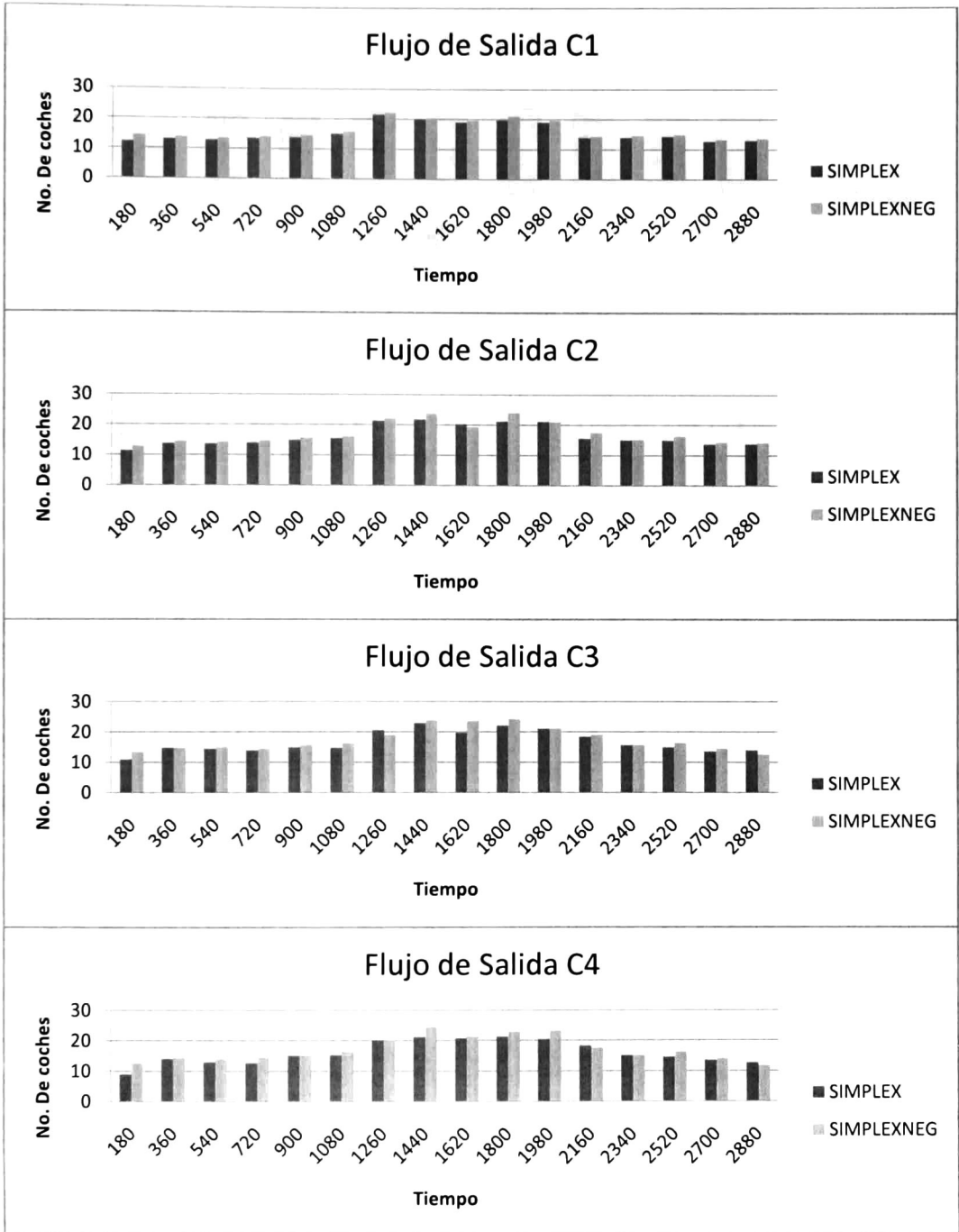


Figura B.38 Flujos de Salida calles C1, C2, C3, C4

### B.4.5. Tiempos de verde con negociación

#### Esquema Difuso

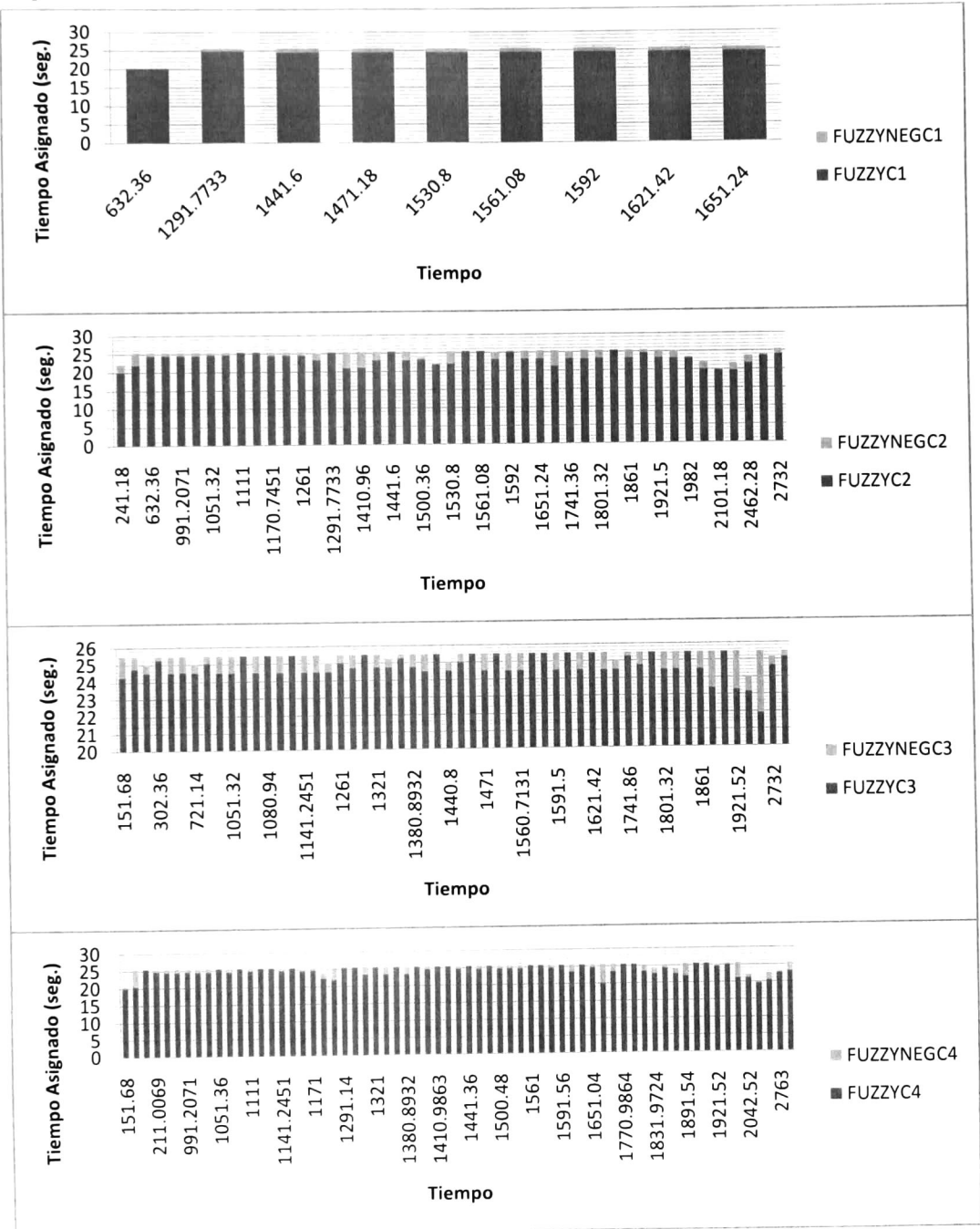


Figura B.39 Tiempos de verde antes y después de negociar intersecciones 1, 2, 3, 4, esquema difuso.

### Esquema Programación Lineal

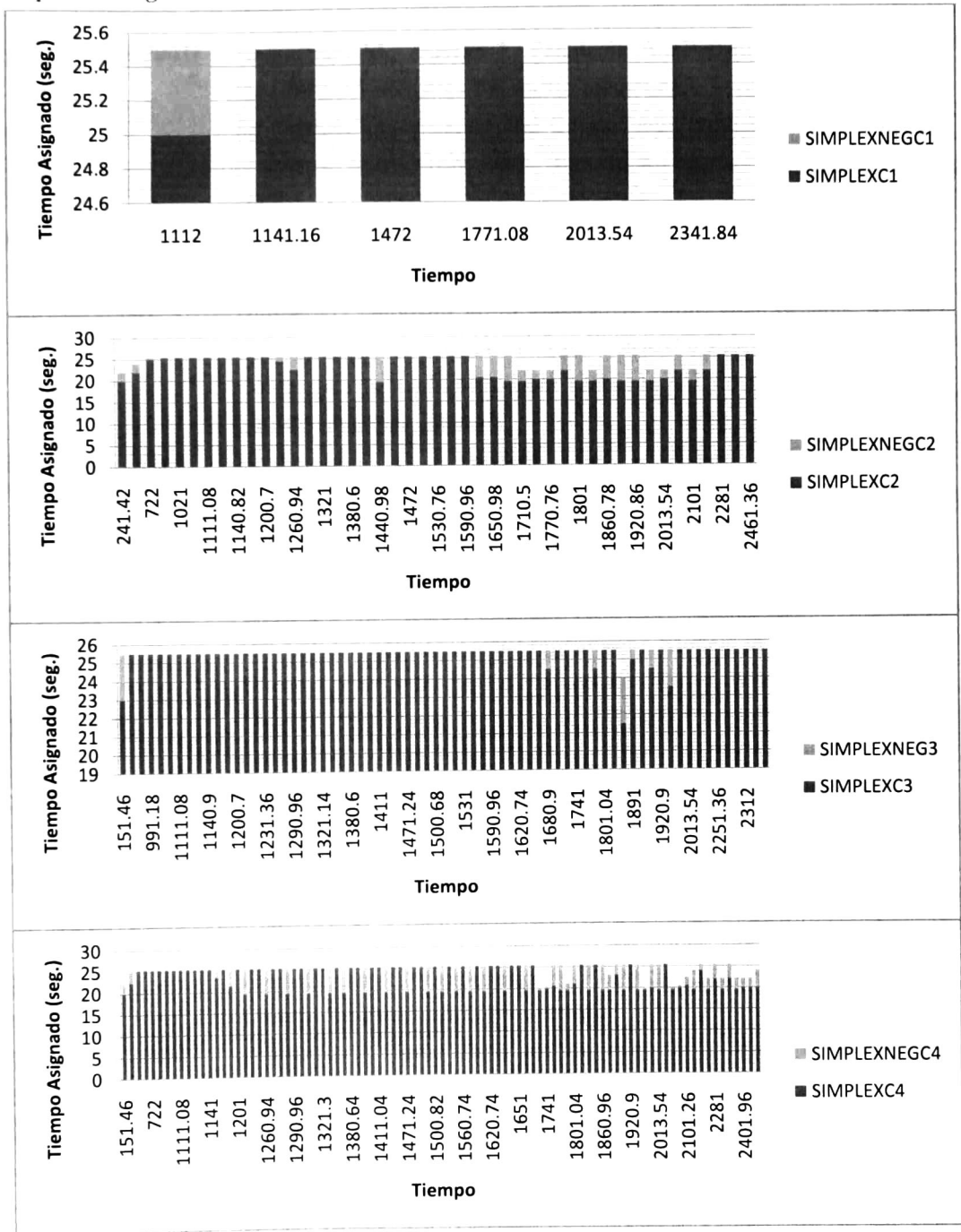


Figura B.40 Tiempos de verde antes y después de negociar intersecciones 1, 2, 3, 4, esquema PL.

## REFERENCIAS

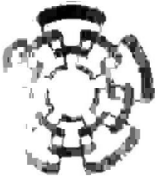
- [Annunziato, 2003] Annunziato M., Bertini I., Pannicelli A., Pizzuti S. "Evolutionary feed-forward networks for traffic prediction" Evolutionary Methods for Design, Optimization and Control with application to Industrial problems, EUROGEN 2003. Barcelona, Spain. 15-17 September, 2003.
- [Bazaraa, 1977] Bazaraa M., Jarvis J., Sherali H. "Linear Programming and Network Flows" John WILEY & SONS, 1977.
- [Bitting, 2004] Bitting E., Ghorbani A. "Cooperative Multiagent Systems for the Optimization of Urban Traffic," IEEE/WIC/ACM International Conference on Intelligent Agent Technology , IAT'04. Beijing, China. 20-24 September 2004.
- [Breherton, 2004] Bretherton, D., Bodger, M., Baber, N. "SCOOT the future" Road Transport Information and Control Conference, 2004. RTIC 2004. 12th IEE International Conference on Volume Issue , 20-22 April 2004 Pags.: 301 - 306
- [Castillo, 2002] Castillo E., Conejo A., Pedregal P., García R., Alguacil N. "Building and Solving Mathematical Programming Models in Engineering and Science", Pure and Applied Mathematics Series, Wiley. 2002.
- [Chee, 2002] Chee M. ,Srinivasan D., Long R.. "Hybrid Cooperative Agents with Online Reinforcement Learning for Traffic Control" IEEE International Conference on Fuzzy Systems, 2002. Honolulu, HI, USA. 12-17 May 2002.
- [Chee, 2003] Chee M.. "Cooperative, Hybrid Agent Architecture for Real-Time Traffic Signal Control" IEEE Transactions on Systems, Man and Cybernetics- Part A. Vol. 33, No. 5, September 2003.
- [Chen, 2001]. Chen G., Tat T. "Introduction to Fuzzy Sets, Fuzzy Logic and Fuzzy Control Systems" CRC Press 2001.
- [Crawford, 2000] Crawford V. "John Nash and the analysis of strategic behaviour" Discussion Paper. January, 2000.
- [Dagan, 2002] Dagan N., Volij O., Winter E. "A characterization of the Nash bargaining solution".Social Choice and Welfare, 2002 – Springer Page 1. Soc Choice Welfare 19: 811–823. 2002.
- [Dotoli, 2005] Dotoli M, Pia M., Iacobellis G. "Validation of an Urban Traffic Network Model using Colored Timed Petri Nets" Systems, Man and



- Cybernetics, 2005 IEEE International Conference on Waikoloa, Hawaii, USA. 10-12 Oct. 2005.
- [Dresner, 2004] Dresner K., Stone P. "Multiagent Traffic Management: An Improved Intersection Control Mechanism" AAMAS'05. Utrecht, Netherlands. July 25-29, 2005.
- [Ehlert, 2001] Ehlert P., Rothkrantz L. "Microscopic traffic simulation with reactive driving agents" 2001 IEEE Intelligent Transportation Systems Conference Proceedings – Oakland, CA, USA. 25-29 August, 2001.
- [Gershenson, 2006] Gershenson C. "Self-organizing traffic lights" Complex Systems 16, 1, pags. 29–53.
- [Ghanbari, 1997] Ghanbari M., Hughes C., Sinclair M., Eade J. "Principles of Performance Engineering for telecommunications and informations systems" IET, 1997.
- [Girianna, 2004] Girianna M., Benekohal R. "Using Genetic Algorithms to Design Signal Coordination for Oversaturated Networks" 11th World Congress and Exhibition on ITS. Nagoya, Japan. 18-22 October 2004.
- [Herings, 2002] Herings, P. Jean-Jacques P. "A Globally Convergent Algorithm to Compute All Nash Equilibria of n-Person Games" METEOR Working Paper. December 10, 2002.
- [Hewage, 2004] Hewage K., Ruwanpura J. "Optimization of traffic signal light timing using simulation" 2004 Winter Simulation Conference. Washington, D.C. 5-8 December 2004.
- [Jensen, 2007] Kurt, J., Kristensen, L., Wells, L. "Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems" In: International Journal on Software Tools for Technology Transfer 2007. pringer Berlin / Heidelberg ISSN1433-2779. .Pags. 213-254. 2007.
- [Julvez, 2003] Julvez J., Boel R. "Modelling and controlling traffic behaviour with continuous Petri nets". In: Proc. 16th IFAC World Congress. Prague, Czech Republic. July, 2005.
- [Khalid, 2004] Khalid M. "Control of a Complex Traffic Junction using Fuzzy Inference," in IEEE Transactions on Systems, Man and Cybernetics. Melbourne, Australia, 2004. Vol. 3. Pags. 1544- 1551.
- [Monsalve, 2003] Monsalve S. "John Nash y la Teoría de Juegos" Lecturas Matemáticas Volumen 24 (2003). Pags. 127-149.

- [Neri, 2005a] López E.. “Micro-Simulación de Tráfico Urbano basada en Sistemas de Agentes Móviles”. Tesis de Maestría, CINVESTAV Unidad Guadalajara. Agosto de 2005.
- [Nguyen, 2003] Nguyen H., Prasad N., Walker C., Walker E. “A First Course in Fuzzy and Neural Control”. CHAPMAN & HALL/CRC, 2003.
- [Oloffson, 2005] Olofsson P. “Probability, Statistics, and Stochastic Processes” WILEY-INTERSCIENCE, 2005.
- [ Ossowski, 1998] Ossowski S., Cuenca J., García-Serrano A. “A Case of Multiagent Decision Support: Using Autonomous Agents for Urban Traffic Control”. IBERAMIA 98, 6th Ibero-American Conference on AI. Lisbon, Portugal. October 5-9, 1998.
- [Parsons, 2002] Parsons S., Gmytrasiewicz P., Wooldridge M. “Game Theory and Decision Theory in Agent-Based Systems” Kluwer Academic Publishers, 2002.
- [Roozemon, 2000] Roozemon D., Rogier L. “Agent Controlled Traffic Lights”, European Symposium on Intelligent Techniques. Aachen, Germany. September 14-15 2000.
- [Rosenschein, 1994] Rosenschein J., Zlotkin G. “Rules of Encounter : Designing Conventions for Automated Negotiation Among Computers” MIT Press, Cambridge, Massachusetts, USA. 1994.
- [Sanal, 1999] Sanal R. “Intelligent Traffic Signals using Fuzzy Logic Control” Kerala Science Congress- Idukki, Kerala, India 1999.
- [SCATS, 2007] [http://www.powerhousemuseum.com/australia\\_innovates/?behaviour=view\\_article&Section\\_id=1080&article\\_id=10087](http://www.powerhousemuseum.com/australia_innovates/?behaviour=view_article&Section_id=1080&article_id=10087)
- [Tavladakis, 1999] Tavladakis K., Voulgaris N. “Development of an Autonomus Adaptive Traffic Control System” ESIT '99 The European Symposium on Intelligent Techniques, Crete, Greece. 03-04 June, 1999.
- [TRANSYT, 2007] <http://www.trlsoftware.co.uk/products/detail.asp?aid=4&c=2&pid=66>
- [Trejo, 2006] Trejo Sanchez Joel. “Control de Tráfico Urbano basado en Sistemas Multiagentes” Tesis de Maestría, CINVESTAV Unidad Guadalajara. Septiembre de 2006.
- [Vetta, 2002] Vetta, A. “Nash equilibria in competitive societies with applications to facility location, traffic routing and auctions” Proceedings of the 43rd Symposium on the Foundations of Computer Science, pp416-425, 2002.

- [Vogel, 2000] Vogel A., Goerick C., Von-Seelen W.. “Evolutionary Algorithms for Optimizing Traffic Signal Operation” ESIT2000, Aachen, Germany. 14-15 September, 2000.
- [Wang, 2005] Wang F “Agent-Based Control for Networked Traffic Management Systems” Published by the IEEE Computer Society, September-October 2005.
- [Wiering, 2004] Wiering M., Van Veenen J., Vreeken J., Koopman A. “Intelligent Traffic Light Control” Reporte Técnico, Institute of information and computing sciences, Utrecht University.
- [Wooldridge, 2002] Wooldridge M.. “An Introduction to Multiagent Systems” John Wiley & Sons, Ltd., 2002.
- [Xiao, 2004] Xiao W., Shu Y., Xue Z. “Architecture of Multi-agent System for Traffic Signal Control” 8th International Conference on Control, Automation, Robotics and Vision. Kunming, China, 06-09 December 2004.



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL I.P.N.  
UNIDAD GUADALAJARA**

El Jurado designado por la Unidad Guadalajara del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional aprobó la tesis

Un esquema multiagentes para la coordinación de tráfico urbano

del (la) C.

Zazhil-ha HERENA ULLOA

el día 18 de Octubre de 2007.

Dr. Félix Francisco Ramos Corchado  
Investigador CINVESTAV 3A  
CINVESTAV Unidad Guadalajara

Dr. Antonio Ramirez Treviño  
Investigador CINVESTAV 2C  
CINVESTAV Unidad Guadalajara

Dr. Mario Angel Siller González  
Pico  
Investigador CINVESTAV 2A  
CINVESTAV Unidad Guadalajara

Dra. María Elena Meda Campaña  
Profesor Titular  
Universidad de Guadalajara CUCEA





CINVESTAV  
BIBLIOTECA CENTRAL



SSIT000006230