



UNIVERSIDAD POLITÉCNICA SALESIANA
UNIDAD DE POSGRADOS
MAESTRÍA EN CONTROL Y
AUTOMATIZACIÓN INDUSTRIALES

Proyecto de investigación
y desarrollo previo a la
obtención del Grado de Magister
en Control y Automatización
Industriales

MODELADO DE TRÁFICO VEHICULAR EN AVENIDAS
A PARTIR DE DATOS ESTADÍSTICOS

Autor:
Ing. Ismael Campoverde

Dirigido por:
Ing. Julio Viola Ph. D.

**MODELADO DE TRÁFICO VEHICULAR EN
AVENIDAS A PARTIR DE DATOS
ESTADÍSTICOS**

MODELADO DE TRÁFICO VEHICULAR EN AVENIDAS A PARTIR DE DATOS ESTADÍSTICOS

Autor:

ISMAEL SANTIAGO CAMPOVERDE BORJA Ingeniero
Electrónico
Egresado de la Maestría en Control y Automatización Industriales
Unidad de Postgrados
Universidad Politécnica Salesiana

Dirigido por:

JULIO CÉSAR VIOLA
Ingeniero Electrónico
Doctorado en Ingeniería
Investigador Proyecto Prometeo - SENESCYT



Cuenca - Ecuador



ISMAEL SANTIAGO CAMPOVERDE BORJA

“MODELADO DE TRÁFICO VEHICULAR EN AVENIDAS A PARTIR DE DATOS ESTADÍSTICOS”

Universidad Politécnica Salesiana, Cuenca - Ecuador, 2017

MAESTRÍA EN CONTROL Y AUTOMATIZACIÓN INDUSTRIALES

Formato 170 x 240 mm

Páginas: 59

Breve reseña del autor e información de contacto:



ISMAEL SANTIAGO CAMPOVERDE BORJA

Ingeniero Electrónico

Egresado de la Maestría de Control y Automatización Industriales

ismel19@hotmail.com

Dirigido por:



JULIO CÉSAR VIOLA

Ingeniero Electrónico

Doctor en Ingeniería

Investigador Proyecto Prometeo - SENESCYT

jcviola@gmail.com

Todos los derechos reservados

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos o investigativos por cualquier medio, con la debida notificación a los autores

DERECHOS RESERVADOS

©2017 Universidad Politécnica Salesiana

CUENCA - ECUADOR - SUDAMÉRICA

CAMPOVERDE BORJA ISMAEL S.

Modelado de tráfico vehicular en avenidas a partir de datos estadísticos.

IMPRESO EN ECUADOR - PRINTED IN ECUADOR

Índice general

1. MARCO TEÓRICO	1
1.1. Principios del tránsito vehicular	1
1.1.1. Definición de tráfico	1
1.1.2. Congestión vehicular	1
1.1.3. Conceptos básicos	2
1.2. Enfoques del flujo vehicular	2
1.2.1. Enfoque microscópico	2
1.2.2. Enfoque macroscópico	3
1.2.3. Enfoque mesoscópico	3
1.3. Modelos del flujo vehicular	3
1.3.1. Modelo lineal de Greenshields:	4
1.3.2. Modelo logarítmico de Greenbergs:	6
1.3.3. Modelo exponencial de Underwood:	8
1.4. Aplicaciones del flujo vehicular	9
1.4.1. Control por tiempos fijos	9
1.4.2. Control sensible al tráfico	9
2. MODELADO DEL TRÁFICO VEHICULAR	10
2.1. Fundamentos de lógica difusa	10
2.1.1. Sistemas funcionales Takagi-Sugeno	13
2.1.2. Método del gradiente descendente	14
2.2. Variables de entrada y salida	14
2.3. Herramientas para la obtención del modelo	16
2.3.1. Datos estadísticos:	16
2.3.2. Lenguaje de cálculo técnico	16
2.3.3. Toolbox de MATLAB:	16
2.4. Procedimiento para la obtención del modelo del flujo vehicular	19
2.4.1. Procesamiento de datos	19
2.4.2. Construcción del modelo ANFIS	21
3. MODELADO DEL TRÁFICO VEHICULAR DE LA AVENIDA HÉROES DE VERDELOMA	24

3.1. Antecedentes	24
3.2. Modelado del flujo vehicular en la intersección Mariano Cueva	25
3.2.1. Procesamiento de datos	25
3.2.2. Construcción del modelo ANFIS	28
3.3. Modelado del flujo vehicular en la intersección Luis Cordero	31
3.3.1. Procesamiento de datos	31
3.3.2. Construcción del modelo ANFIS	33
3.4. Modelado del flujo vehicular en la intersección Nicanor Merchán	34
3.4.1. Procesamiento de datos	35
3.4.2. Construcción del modelo ANFIS	35
4. MODELADO PREDICTIVO DE UNA INTERSECCIÓN SEMAFÓRICA Y SUS RESULTADOS	40
4.1. Antecedentes	40
4.2. Modelado predictivo del flujo vehicular en la intersección Luis Cordero	41
4.2.1. Procesamiento de datos	41
4.2.2. Construcción del modelo ANFIS	42
4.3. Modelado predictivo del flujo vehicular en la intersección Nicanor Merchán	43
4.3.1. Procesamiento de datos	45
4.3.2. Construcción del modelo ANFIS	45
Anexos	50
Anexos	50
A. Aforo Vehicular - Av. Héroes de Verdeloma	51
A.1. Intersección Av. Héroes de Verdeloma - Mariano Cueva	51
A.2. Intersección Av. Héroes de Verdeloma - Luis Cordero	52
A.3. Intersección Av. Héroes de Verdeloma - Nicanor Merchán	54
B. Código de MATLAB para la Obtención del Modelo	56
B.1. Procesamiento de Datos	56
B.2. Remuestreo de Datos	56
B.3. Construcción del Modelo Anfis	57
B.4. Procesamiento y Remuestreo de Datos	57
B.5. Cálculo del FIS	58

Índice de figuras

1.1. Densidad vs velocidad	4
1.2. Densidad vs Flujo	5
1.3. Modelo logarítmico de Greenbergs	8
2.1. Esquema del FIS	11
2.2. Funciones de pertenencia	12
2.3. Ruido Aleatorio	20
2.4. Procesamiento de datos	20
3.1. Avenida Héroes de Verdeloma	25
3.2. Intersecciones semafóricas	25
3.3. Datos vs tiempo en la intersección Mariano Cueva	26
3.4. Datos de entrada Procesados	27
3.5. Matriz de datos de <i>training</i>	28
3.6. Datos de la salida del modelo ANFIS vs. Tiempo	29
3.7. Errores de <i>training</i> y <i>checking</i> vs. Repeticiones de entrenamiento	30
3.8. Errores de predicción	31
3.9. Datos del flujo vehicular vs Tiempo, para la intersección Luis Cordero .	32
3.10. Datos de entrada procesados, intersección Luis Cordero	32
3.11. Datos de entrada y Modelo ANFIS vs. Tiempo	33
3.12. Errores de <i>training</i> y <i>checking</i> vs. Repeticiones de entrenamiento	34
3.13. Errores de predicción	35
3.14. Datos vs Tiempo, intersección Nicanor Merchán	36
3.15. Datos de entrada procesados, intersección Nicanor Merchán	36
3.16. Datos de entrada y Modelo ANFIS vs. Tiempo	37
3.17. Errores de <i>training</i> y <i>checking</i> vs. Repeticiones de entrenamiento	38
3.18. Errores de predicción	38
4.1. Avenida Héroes de Verdeloma	41
4.2. Datos de entrada y Modelo ANFIS vs. Tiempo	43
4.3. Errores de <i>training</i> y <i>checking</i> vs. Repeticiones de entrenamiento	44
4.4. Errores de predicción	44

4.5. Datos de entrada y Modelo ANFIS vs. Tiempo	45
4.6. Errores de <i>training</i> y checking vs. Repeticiones de entrenamiento	46
4.7. Errores de predicción	47

DEDICATORIA

*Dedicado:
A mis padres, hermanos, sobrinos
y seres queridos por su apoyo incondicional,
en especial a mi hijo Ismael Alejandro,
por ser mi fuerza y mis ganas para seguir adelante.*

PREFACIO

Actualmente la cantidad de vehículos existentes en las zonas urbanas de las ciudades ha incrementado, lo que produce largas filas de vehículos en las vías, que son altamente dependientes de la sincronización de encendido-apagado de los semáforos existentes en las intersecciones. En la zona céntrica de la ciudad de Cuenca, se encuentra implementado un Sistema de Control de Tráfico Urbano Adaptativo (SCATS), mientras que en las partes exteriores los semáforos se encuentran funcionando con tiempos fijos previamente establecidos.

En la Avenida Héroes de Verdeloma los semáforos se encuentran funcionando bajo el método de tiempos fijos, los cuales son establecidos en base a la toma de datos estadísticos. Para determinar los tiempos fijos en las intersecciones o implementar algún sistema inteligente de semáforos, es necesario modelar el tráfico existente, para luego realizar cualquier tipo de control. El modelo del sistema no existe actualmente y su desarrollo es propuesto como tema central de este trabajo de investigación.

En la Ciudad de Cuenca existen varias avenidas importantes que carecen del control de flujo vehicular, por esta razón, se ha visto necesario hallar el modelado del tráfico existente en una avenida de alta circulación de vehículos, para luego usar el mismo principio en diferentes zonas de la ciudad.

PRÓLOGO

La presente investigación se realizó en la avenida Héroes de Verdeloma, ubicada en la ciudad de Cuenca - Ecuador. Se consideró necesario analizar el flujo vehicular, modelando su comportamiento en diferentes intersecciones, las cuales se encuentran a lo largo de la avenida antes mencionada.

Este proyecto centra su estudio en las intersecciones: Mariano Cueva, Luis Cordero y Nicanor Merchán, de las que se tienen datos experimentales. Para cada una se analizó el flujo vehicular.

El modelo del sistema se obtuvo a partir de: datos estadísticos conocidos, el uso del software MATLAB y aplicando principios de lógica difusa.

Luego de obtener el modelado del sistema, se efectuaron las comprobaciones necesarias para reconocerlo como válido. Al final se analizaron los resultados y se obtuvieron las conclusiones.

AGRADECIMIENTO

Expreso mi agradecimiento a la Universidad Politécnica Salesiana, a la Dirección de Postgrados, a todos los colaboradores y cordiales amigos; de manera especial al director el Dr. Julio Viola por su apoyo en el transcurso de mi Formación Académica, y por brindarme su amistad desinteresada. Hago extensivo este agradecimiento a la Municipalidad de la ciudad de Cuenca, por brindarme la confianza y oportunidad de desarrollar este proyecto de investigación en una avenida de la ciudad.

Capítulo 1

MARCO TEÓRICO

Introducción

En esta primera parte del proyecto de investigación, se trata todo lo relacionado a la teoría del flujo vehicular, los conceptos básicos y sus principios, además se explican los enfoques en los cuales se puede dividir el análisis del tránsito vehicular. Se presenta, también, una descripción de los modelos matemáticos que actualmente han sido desarrollados y por último se dan a conocer algunas aplicaciones que involucran el uso del concepto de flujo vehicular.

1.1. Principios del tránsito vehicular

1.1.1. Definición de tráfico

El tráfico, también llamado tránsito vehicular, es aquel que se produce al existir un flujo de vehículos en una vía, calle o autopista. Este flujo puede potencialmente producir un congestionamiento vehicular, el cual puede ser corregido mediante el uso de semáforos.[1]

1.1.2. Congestión vehicular

La congestión vehicular es aquella que se presenta cuando existe una saturación producida por el exceso de vehículos en las vías, lo que produce un incremento en los tiempos de viaje e inconvenientes para circular. Por lo general, la congestión vehicular se presenta en las denominadas horas pico, en las cuales la mayoría de personas circulan en sus vehículos hacia destinos comunes.[2]

El congestionamiento vehicular se produce, cuando los vehículos presentes ocupan todo el espacio de circulación disponible en las vías. Esto puede ser ocasionado por un accidente en la carretera, el daño de algún vehículo que circulaba por la vía, e incluso por el clima, pues en días lluviosos se produce mayor congestión que en un día despejado. En las zonas céntricas de las ciudades es muy frecuente la presencia del congestionamiento vehicular, sobre todo en ocasiones especiales como pueden ser las fechas festivas.

1.1.3. Conceptos básicos

A continuación se definen varios conceptos comunes, utilizados por distintos autores en la descripción de modelos de flujo vehicular [1], [2], entre los principales se encuentran:

Flujo: Es el número de vehículos por unidad de tiempo.

Velocidad: Es la relación existente entre la distancia recorrida y el tiempo que se tarda un vehículo en recorrer esa distancia. Esta variable es importante pues con ella se puede medir el nivel de congestión vehicular existente.

Cuando ningún vehículo circula sobre la vía, se considera a la velocidad de circulación como la velocidad máxima permitida.

Conforme circulen vehículos por la vía, la velocidad de circulación estimada para recorrer la calle, será la misma que el resto de vehículos que circulan por la carretera.

Tiempo: Es cuánto se demora un vehículo en recorrer alguna vía establecida, para calcularlo se debe realizar la división entre la distancia de la vía y la velocidad de circulación del vehículo.

Ocupación: Es el porcentaje de tiempo que un tramo del camino es ocupado por vehículos.

Densidad: Es el número de vehículos por unidad de distancia, en un instante de tiempo determinado.

1.2. Enfoques del flujo vehicular

1.2.1. Enfoque microscópico

En el estudio microscópico se analiza el flujo vehicular puntualmente, es decir se sigue la trayectoria de cada vehículo y se analizan las variables que lo describen: distancia del vehículo, densidad del flujo, velocidad a la que circula, aceleración, distancia existente entre vehículos, etc. La obtención del modelo dependerá del número de variables que se obtengan para encontrar el mismo.[3]

El enfoque microscópico describe en detalle el comportamiento del flujo vehicular, a tal punto que debe ser posible analizar el comportamiento del conductor y del vehículo como entidades individuales que interactúan entre sí, es decir cada vehículo es analizado por separado.

1.2.2. Enfoque macroscópico

En el estudio macroscópico se considera al flujo vehicular como un flujo continuo (similar al de un fluido o gas). Para esto se considera una gran cantidad de vehículos de manera que las variables como la densidad, velocidad y otras, sean una medida representativa del flujo, es decir, se analiza el flujo vehicular a gran escala. Existen varios tipos de modelos, pero todos dependen del número de variables que se analicen para obtenerlos. En este enfoque es necesario conocer la cantidad de vehículos que circulan por la avenida.[4], [5]

Las ventajas del enfoque macroscópico son:[3]

- Sus resultados brindan una buena aproximación cualitativa a la realidad del flujo vehicular
- Es posible analizar el flujo en varios carriles
- Son mas eficientes que los modelos microscópicos, desde el punto de vista computacional

Definir o conocer cuáles son exactamente las variables que intervienen para la obtención de un modelo de flujo vehicular es algo difícil, debido a que todo dependerá de los datos conocidos y del nivel de detalle al cual se quiere llegar con la aplicación del modelo. [3]

1.2.3. Enfoque mesoscópico

El enfoque mesoscópico o también llamado enfoque cinético, involucra algunas características del enfoque macroscópico y otras del enfoque microscópico. Los vehículos son identificados individualmente, pero su comportamiento se rige por probabilidad, es decir, analiza la probabilidad de que un vehículo a cierta velocidad se encuentre en determinado tiempo y en alguna posición. Este enfoque esta basado en la ecuación cinética de los gases. [6]

1.3. Modelos del flujo vehicular

Para determinar las relaciones existentes entre los parámetros del tráfico, se ha investigado sobre los trabajos elaborados en las últimas décadas,[7], [8], Los resultados de

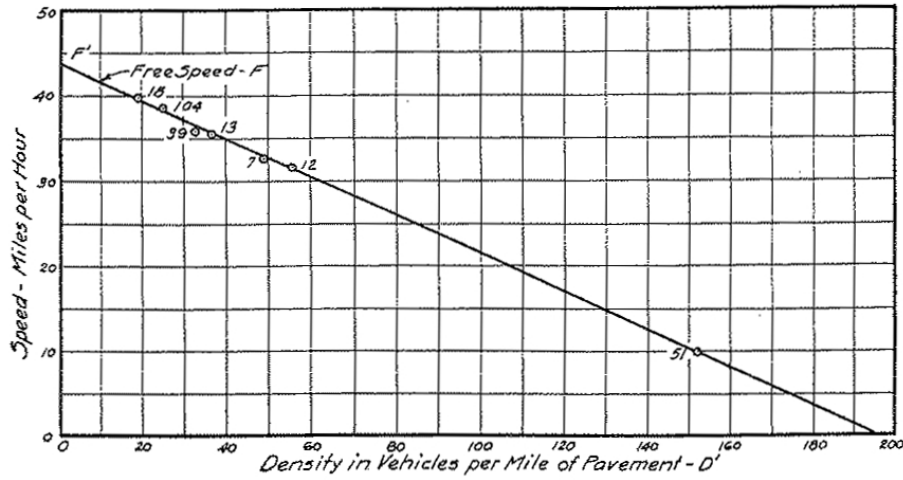


Figura 1.1: Densidad vs velocidad

[7]

estas investigaciones produjeron algunos modelos matemáticos, los cuales contienen las principales variables que intervienen en el cálculo del flujo vehicular, las cuales son: flujo (q), densidad (k) y velocidad (v). Estas variables se encuentran relacionadas en la ecuación general:

$$q = vk \quad (1.1)$$

A continuación se detallan algunos modelos propuestos por distintos autores y que hacen uso de los enfoques descritos previamente.

1.3.1. Modelo lineal de Greenshields:

Greenshields propone un modelo de tráfico ininterrumpido, que predice y explica las tendencias que se observan en un flujo de tráfico real, aunque este modelo no es perfecto, es muy preciso y simple [1].

El autor utiliza un conjunto de datos (k , v) bajo diferentes condiciones de tránsito, luego plantea una relación lineal entre la velocidad y la densidad, como se puede observar en la figura 1.1.

Plantea además la siguiente ecuación utilizando el método de mínimos cuadrados:

$$V = V_f - \frac{V_f}{k_c} k \quad (1.2)$$

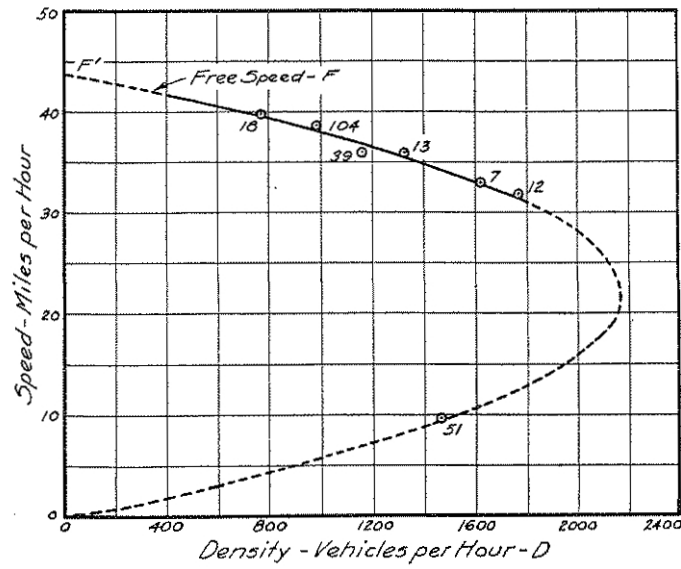


Figura 1.2: Densidad vs Flujo
[7]

Donde:

- V = Velocidad (km/hora)
- V_f = Velocidad a flujo libre (km/hora)¹
- k_c = Densidad de congestionamiento (vehículo/km/carril)²
- k = Densidad (vehículo/km/carril)

Reemplazando (1.2) en la ecuación general del flujo (1.1), se tiene:

$$q = (V_f - \frac{V_f}{k_c}k)k = V_fk - \frac{V_f}{k_c}k^2 \quad (1.3)$$

Donde:

- q = Flujo (vehículo/hora)

La ecuación (1.3), representa gráficamente una parábola como se muestra en la figura 1.2

¹ **Velocidad a flujo libre:** Es la velocidad a la cual circula un vehículo, cuando la vía está sin obstáculos u otros vehículos

² **Densidad de congestionamiento:** Es el número máximo de vehículos por unidad de distancia, en un instante de tiempo determinado, que se encuentran en una misma vía o carril

Esta relación entre flujo y densidad, permitirá encontrar la densidad a la cual el flujo se maximiza, para esto se procede a encontrar el máximo valor de (1.3), derivando de la siguiente manera:

$$\frac{dq}{dk} = 0 = V_f - 2 \frac{V_f}{k_c} k \quad (1.4)$$

para obtener:

$$k = \frac{k_c}{2} \quad (1.5)$$

Para encontrar el flujo maximizado es necesario sustituir (1.5) en (1.2), de la siguiente forma:

$$V = V_f - \frac{V_f k_c}{k_c} \frac{k}{2} = \frac{V_f}{2} \quad (1.6)$$

Esto significa que el máximo flujo se presenta cuando el tráfico está fluyendo a la mitad de la velocidad del flujo libre. Sustituyendo luego la velocidad óptima y densidad en (1.1) es posible obtener el flujo máximo, así:

$$q = \frac{V_f}{2} \frac{k_c}{2} = \frac{V_f k_c}{4} \quad (1.7)$$

De esta manera se puede concluir que según el modelo de Greenshields:

- Cuando la densidad es cero el flujo también es cero porque no hay vehículos en el carril
- A medida que aumenta la densidad, el flujo también aumenta hacia condiciones de flujo máximo
- Cuando la densidad alcanza un máximo (densidad de congestionamiento), el flujo debe ser cero porque los vehículos tienden a alinearse y no pueden circular

1.3.2. Modelo logarítmico de Greenbergs:

Greenbergs asemeja al flujo vehicular con la teoría de fluidos (hidrodinámica)[1], realizando una combinación de las ecuaciones de movimiento y continuidad de fluidos compresibles, obteniendo una relación logarítmica entre la velocidad y densidad

$$V = V_f \ln \frac{k_c}{k} \quad (1.8)$$

y la relación con el flujo:

$$q = V_f k \ln \frac{k_c}{k} \quad (1.9)$$

Este modelo es muy útil para flujos congestionados pero no sirve para bajas densidades, esto es debido a que si k tiende a cero la velocidad puede ser infinita, es decir se llega a la condición de flujo libre, resultando:

$$V = V_f$$

$$k = k_{max}$$

Sustituyendo V_f y k en (1.8), se obtiene:

$$V_f = V_f \ln \frac{k_c}{k_{max}} \quad (1.10)$$

$$\ln \frac{k_c}{k_{max}} = 1$$

De donde:

$$k_{max} = \frac{k_c}{e} \quad (1.11)$$

siendo:

$$e = 2,718282$$

Por lo tanto, de la ecuación general del flujo (1.1), el flujo máximo se representa por:

$$q_{max} = V_f k_{max} = \frac{V_f k_c}{e} \quad (1.12)$$

La representación gráfica de la relación velocidad - densidad planteada por Greenbergs se puede observar en la línea punteada de la figura 1.3.

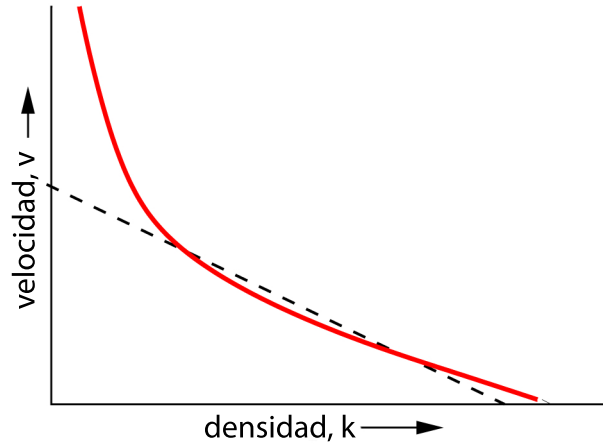


Figura 1.3: Modelo logarítmico de Greenbergs
[9]

1.3.3. Modelo exponencial de Underwood:

Al contrario que Greenbergs, Underwood se interesó en desarrollar un modelo para flujos no congestionados; es decir, que sea usado para autopistas, avenidas o carreteras urbanas [1]. En este modelo no se presenta la condición de congestionamiento, es decir la velocidad no es igual a cero para altas densidades de flujo.

Underwood formula las siguientes ecuaciones:

$$V = V_f e^{\frac{-k}{k_{max}}} \quad (1.13)$$

$$q = V_f k e^{\frac{-k}{k_{max}}} \quad (1.14)$$

Sustituyendo las condiciones de flujo máximo (k_{max} V_{max}) en (1.13) se obtiene:

$$V_{max} = V_f e^{\frac{-k_{max}}{k_{max}}} \quad (1.15)$$

Simplificando:

$$V_{max} = \frac{V_f}{e} \quad (1.16)$$

por lo tanto la ecuación del flujo máximo es:

$$q_{max} = V_{max}k_{max} = \frac{V_f K_{max}}{e} \quad (1.17)$$

1.4. Aplicaciones del flujo vehicular

Actualmente el flujo vehicular incrementa en varias ciudades metrópolis del mundo, este fenómeno causa problemas de congestión vehicular. Para solventar este inconveniente se intenta impulsar el uso de varios medios de transporte: buses, trenes, tranvías, bicicletas; también se construyen pasos a desnivel, puentes, vías alternas, etc. Sin embargo no es suficiente para controlar el alto nivel de flujo vehicular, para esto se crean varios aplicativos que involucran la medición del flujo vehicular y su control.[2], [5]

Las aplicaciones que se usan generalmente para el control del flujo vehicular se pueden dividir en dos grupos: control por tiempos fijos y control sensible al tráfico.

1.4.1. Control por tiempos fijos

Este control se realiza mediante registros históricos de flujo y densidad vehicular, para esto es necesario ajustar a valores constantes, los parámetros que intervienen en el desarrollo del flujo vehicular.

Algunos ejemplos de este tipo de control son: Traffic Signal Settings (SIGSET), The Traffic Capacity of Signal (SIGCAP), Setting Signals on Arteries and Triangular Networks (MAXBAND) y Urban Traffic Control System (UTCS). En las zonas exteriores de la ciudad de Cuenca, los semáforos se encuentran funcionando con este tipo de control, es decir, se establecen tiempos fijos para controlar su cambio de estado.

1.4.2. Control sensible al tráfico

A diferencia del control por tiempos fijos, las mediciones del flujo vehicular se realizan en tiempo real, mediante el uso de sensores, tarjetas de control o cámaras de detección vehicular.

Los ejemplos de este tipo de control son: Microprocessor Optimised Vehicle Actuation (MOVA) y Sydney Coordinated Adaptive Traffic System (SCATS).

Capítulo 2

MODELADO DEL TRÁFICO VEHICULAR

Introducción

En esta segunda parte del proyecto de investigación, se trata de encontrar el modelo del tráfico vehicular, que permita predecir el comportamiento del flujo vehicular a partir de datos experimentales. Tal como se mencionó, se desea obtener un modelo que, partiendo de mediciones disponibles de flujo vehicular en instantes de tiempo $(t-1, t-2, \dots, t-n)$, permita obtener el comportamiento para instantes de tiempo $(t+1, t+2, \dots, t+n)$. Antes se realiza una breve descripción de los conceptos básicos de lógica difusa, para luego explicar cuales son las variables de entrada y salida que intervienen en el proceso. Se presentan también las herramientas necesarias para la obtención del modelo, y al final se explica el procedimiento que se debe seguir para la obtención del Sistema de Inferencia neuro-difuso Adaptativo (ANFIS)

2.1. Fundamentos de lógica difusa

La *lógica difusa* es un sistema lógico que formaliza el razonamiento aproximado, es decir, procura describir el razonamiento humano. La teoría de conjuntos difusos es muy usada para el modelado de sistemas, llevándose a cabo mediante un sistema de inferencia difuso (FIS).

Los *sistemas de inferencia difusos* son aproximadores universales, esto significa que son capaces de aproximar cualquier función continua dentro de un dominio con un cierto nivel de precisión [10]. En conclusión, mediante un FIS se puede obtener un mapeo no lineal entre las entradas y salidas de un sistema basándose en la teoría de la lógica difusa. En la figura 2.1 se puede observar un esquema general del FIS, en el cual se muestran los instantes de tiempo en los cuales han sido adquiridas las entradas y los instantes de tiempo de las salidas que desean ser estimadas.

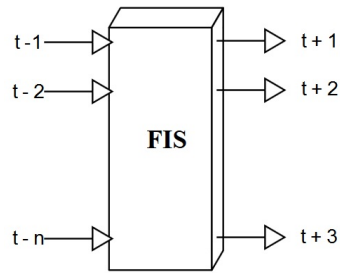


Figura 2.1: Esquema del FIS

Existen dos tipos de modelos que son los más usados para el modelado de sistemas de control: Mandami y Takagi-Sugeno. Los modelos Mandami utilizan proposiciones difusas tanto en los antecedentes y en las consecuencias y son usados para sistemas estáticos [11], mientras que los modelos Takagi-Sugeno utilizan proposiciones difusas solo en los antecedentes y son usados para sistemas de tipo dinámico [12].

Cualquier función puede ser válida para definir un conjunto difuso, sin embargo, algunas son más utilizadas por su simplicidad matemática y se las llama *Funciones de membresía o pertenencia*. Puede ser de tipo triangular, trapezoidal, gaussiana, sigmoidal, etc.

Un *conjunto difuso* es un conjunto cuya función de membresía describe el grado de pertenencia que tiene una variable, con las propiedades o el significado del conjunto.

Un ejemplo práctico puede ser el control del calentamiento manual de una piscina, en donde un operador controla la temperatura de forma manual activando o desactivando una válvula que permite el paso de agua caliente, manteniendo una temperatura aproximada de 25°C. Lo que se busca es automatizar el proceso haciendo uso de la experiencia del operador y aplicando lógica difusa.

En primer lugar se determinan las posibles temperaturas que intervienen: CALIENTE, TIBIO y FRÍO.

En lógica difusa se toma en cuenta que conceptos como: alto, bajo, grande, pequeño, ruidoso, dulce, ancho, delgado, etc. son interpretados de manera distinta por cada persona. Por ejemplo para una persona que trabaja en un proceso de fundición el concepto CALIENTE puede significar temperaturas de más de 300°C, mientras que para una persona que trabaja en un invernadero, CALIENTE pueden ser temperaturas por encima de 30°C. Por este motivo los conjuntos CALIENTE, TIBIO y FRÍO son llamados *conjuntos difusos*. Entonces se puede decir que un conjunto difuso es una agrupación con límites borrosos o no muy bien definidos.

Una vez determinados los conjuntos que representan las temperaturas del proceso, se les asignan valores a cada uno. Para realizar esto se tomará en cuenta la experiencia del operador.

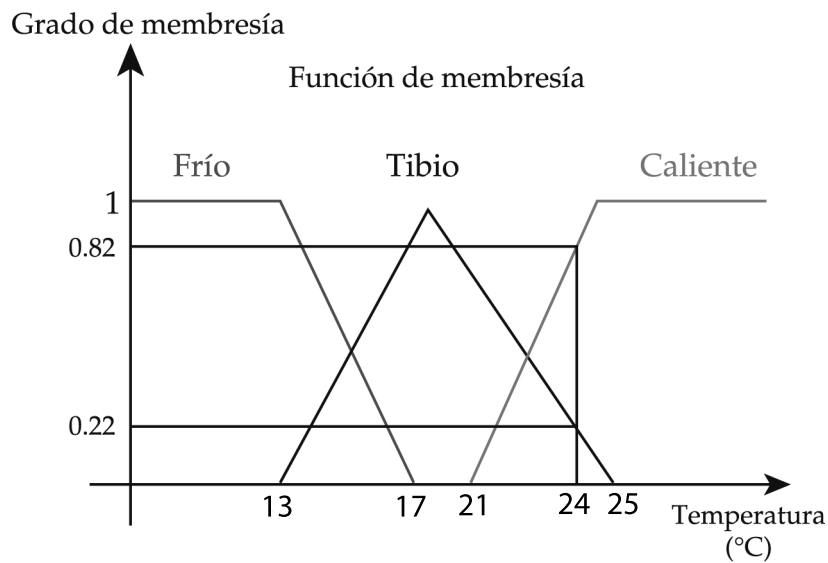


Figura 2.2: Funciones de pertenencia

Suponiendo que se mide la temperatura de la piscina y la medición es de $24,9^{\circ}\text{C}$, la temperatura pertenece al conjunto TIBIO, pero está únicamente a 0,1 grados de ser CALIENTE. Se puede decir que la temperatura es casi CALIENTE, pero según el rango pertenece al conjunto TIBIO, por esta razón es necesario definir un rango donde $24,9^{\circ}\text{C}$ tenga cierta pertenencia al conjunto CALIENTE.

La Lógica difusa asigna a la medición un porcentaje de pertenencia al conjunto, debido a que está cerca del mismo valor, por ejemplo $24,9^{\circ}\text{C}$ tiene un alto porcentaje de pertenencia al conjunto CALIENTE, a este concepto se le conoce como *Grado de membresía*, y puede tomar los valores entre 0 y 1, donde 1 representa pertenencia total al conjunto y 0 ninguna pertenencia. Esta forma de razonar se asemeja a la que utilizamos los humanos y se podría expresar verbalmente como: “Está poco menos que CALIENTE” o “Está demasiado FRÍO” o “Está medio TIBIO”, etc.

De esta manera se definen entonces los intervalos de los conjuntos difusos como se muestra en la figura 2.2, la cual se denomina *función de membresía*. La forma de la función se elige de acuerdo a la aplicación y tomando en cuenta la experiencia del operador. A esta traducción de los valores del mundo real a lógica difusa, a través de funciones de membresía, se le llama *fusificación*.

En la figura 2.2 se puede observar:

- El eje y que es el grado de pertenencia, que describe cuantitativamente la función de membresía
- El eje x que es la temperatura

- La etiquetas CALIENTE, TIBIO, y FRÍO son las variables lingüísticas que describen cualitativamente la función de membresía
- Se puede observar que para una medición de temperatura de 24°C, se asocia un grado de pertenencia tanto en CALIENTE como en TIBIO. Lo mismo sucederá con otras temperaturas que tendrán pertenencias parciales a más de un conjunto

El grado de membresía asociado, dependiendo de la función de membresía es llamado *grado de pertenencia* (GP), de la siguiente forma:

$$GP(24^{\circ}\text{C}, \text{CALIENTE}) = 0,82$$

$$GP(24^{\circ}\text{C}, \text{TIBIO}) = 0,22$$

$$GP(24^{\circ}\text{C}, \text{FRÍO}) = 0$$

El operador que controla la temperatura de la piscina, realiza la misma función aplicando su razonamiento y su experiencia en el proceso y establece las reglas:

FRÍO = Abrir válvula de agua caliente

TIBIO = Entre-abrir la válvula

CALIENTE = Cerrar la válvula de agua caliente

A este conjunto de reglas se las conoce como *reglas difusas* y se representan de la forma SI...ENTONCES, por ejemplo:

- SI **FRÍO** ENTONCES Abrir válvula
- SI **TIBIO** ENTONCES Abre y cierra válvula
- SI **CALIENTE** ENTONCES Cerrar válvula

A partir de esta información, se toman las decisiones de abrir o cerrar la válvula para realizar el control de temperatura de la piscina, a este proceso se le llama *proceso de inferencia*, en el cual se determina hasta que punto las reglas son relevantes y por último se obtienen las conclusiones finales.

Para terminar, se defusifican las conclusiones finales, es decir, se determina un valor numérico de salida a partir de los conjuntos inferidos llevando nuevamente al mundo real. Esta información indica cuánto se debe abrir o cerrar la válvula.

2.1.1. Sistemas funcionales Takagi-Sugeno

Esta arquitectura consiste en una serie de reglas de la forma:

Regla: SI X es A y Y es B ENTONCES Z es $z = f(x, y)$

Donde A y B son conjuntos difusos en el antecedente, $z = f(x, y)$ es una función en la consecuencia, y $f(x, y)$ es un polinomio en las variables de entrada, pero puede ser

cualquier función, siempre y cuando esta describa adecuadamente la salida del modelo especificada por las reglas difusas.

En este tipo de reglas, el consecuente es en realidad una función, de ahí el nombre de sistema funcional. Estos sistemas no tienen funciones de membresía de salida.

2.1.2. Método del gradiente descendente

Existen algunos métodos que son usados para construir modelos difusos, los cuales realizan una réplica del comportamiento de una función determinada. La información sobre la función se presenta en forma de datos de entrada y salida, lo que significa que un conjunto de puntos sobre el dominio de la entrada se selecciona y luego se evalúa en la salida.

La construcción de modelos difusos involucra la selección de algunos parámetros tales como: posición, forma y distribución de las funciones de membresía, base de reglas para la construcción, selección de las operaciones lógicas, consecuencias de las reglas, etc. los cuales deben ser configurados para la obtención de los modelos. Para esto se debe ajustar parámetros y realizar algunas estrategias, entre ellas se encuentra el *Método del gradiente descendente*.

Este método se basa en calcular los parámetros en los antecedentes y las consecuencias del sistema de inferencia difusa, exige al usuario la definición previa de las funciones de membresía de los antecedentes, puede combinarse con un cálculo de las consecuencias mediante el uso de mínimos cuadrados ¹. En este proyecto, el método es conocido como el esquema *anfis*. Este método presenta una convergencia rápida, especialmente para los modelos Takagi-Sugeno.

2.2. Variables de entrada y salida

Para obtener el modelo del flujo vehicular, es necesario observar de manera general el tipo de sistema a tratar, para luego determinar las variables de entrada y salida que intervienen en el proceso. El análisis del flujo para este proyecto de investigación será realizado sobre la Avenida Héroes de Verdeloma de la Ciudad de Cuenca, para lo cual se ha solicitado a la Alcaldía de la ciudad resultados de un aforo realizado sobre esta avenida. A partir de estos datos conocidos se podrán determinar las variables y el enfoque sobre el cual se modelará el sistema.

¹**Método de mínimos cuadrados:** Permite adaptar, cualquier aproximación de datos que se quiera analizar con una función determinada, a una muestra de datos u observaciones denotadas por las variables x y y , es decir se desea encontrar los valores de los parámetros de una función que minimizan la suma de los errores al cuadrado.

A continuación se muestra una pequeña cantidad de datos que contienen información del aforo antes mencionado.

Período	Vehículos
6:00:00	13
6:15:00	12
6:30:00	37
6:45:00	66
7:00:00	76
.	.
.	.
.	.
12:00:00	132
12:15:00	108
12:30:00	114
12:45:00	113
13:00:00	111
.	.
.	.
.	.
19:00:00	116
19:15:00	133
19:30:00	104
19:45:00	114

Tabla 2.1: *Aforo Vehicular*

En la tabla 2.1 se pueden observar dos columnas: Período y Vehículos, la primera muestra la hora a la cual ha sido medido el aforo vehicular, manteniendo un período de adquisición de 15 minutos, desde las 6:00 am hasta las 20:00 y la segunda columna indica la cantidad de vehículos que han circulado por una intersección dada en ese lapso de tiempo.

El objetivo de este proyecto de investigación, es encontrar un modelo predictivo, es decir, a partir de datos estadísticos conocidos, se predice cual será el dato futuro, para poder reconstruir el comportamiento del mismo y encontrar el modelo del sistema. Para este caso las variables de entrada son: el tiempo y la cantidad de vehículos, siendo la salida el número de vehículos que habrá en un instante de tiempo futuro en una determinada intersección. La tabla 2.2 muestra un ejemplo donde los datos de número de vehículos tomados a las 12:00 y 12:15 se utilizan como entradas y el dato de número de vehículos a las 12:30 será la predicción que el modelo tendrá como salida.

Período	Vehículos	Tipo
12:00:00	132	ENTRADA
12:15:00	108	ENTRADA
12:30:00	114	SALIDA

Tabla 2.2: Ejemplo de variables de entrada y salida del sistema

2.3. Herramientas para la obtención del modelo

Una vez conocidas cuáles son las variables que intervienen en el sistema, se procede a describir las herramientas necesarias para la obtención del modelo.

2.3.1. Datos estadísticos:

Es la información que contiene el aforo vehicular facilitado por la Alcaldía de la ciudad de Cuenca

2.3.2. Lenguaje de cálculo técnico

Para obtener el modelo del sistema se ha utilizado MATLAB, el cual es usado a nivel mundial por millones de ingenieros y científicos para analizar y diseñar sistemas. Los gráficos integrados le brindan al usuario una facilidad para elaborar cálculos computacionales a nivel profesional permitiendo además elaborar simulaciones de sistemas complejos

2.3.3. Toolbox de MATLAB:

MATLAB contiene una amplia variedad de conjuntos de funciones (comunmente conocidos como *toolboxes*) para cada tipo de aplicación. Para este proyecto de investigación se usará el “Fuzzy Logic Toolbox” utilizando las siguientes herramientas: `genfis1` y `anfis`. Cada una de ellas se detalla a continuación:

Herramienta GENFIS1

Esta herramienta genera la estructura de un Sistema de Inferencia Difuso a partir de un conjunto de datos de entrenamiento (*training data*), con el número y tipo de funciones de membresía de entrada y salida explícitamente especificadas.

Sintaxis: `fismat = genfis1(data,numMFs,inmftype,outmftype)`

Los argumentos de entrada de la herramienta `genfis1` son:

`data`: Es la matriz de datos de entrenamiento, la cual debe ser ingresada en columnas, las primeras contienen los datos de entrada y la última los datos de salida.

`numMFs`: Es un vector en el cual las coordenadas especifican el número de funciones de membresía asociadas con cada entrada

`inmftype`: Indica el tipo de funciones de membresía que son asociadas con cada entrada

`outmftype`: Es una cadena que especifica el tipo de funciones de membresía asociadas con la salida. Genera un sistema de inferencia difuso tipo Sugeno de una sola salida. El tipo de funciones de membresía de salida puede ser lineal o constante, el número de funciones de membresía asociadas con la salida es el mismo que el número de reglas generadas por `genfis1`.

Herramienta ANFIS

Esta herramienta es la principal rutina de entrenamiento para los sistemas de inferencia difusos y utiliza un algoritmo de aprendizaje híbrido para identificar parámetros de los sistemas de inferencia difuso tipo Sugeno. Mediante una combinación del método de mínimos cuadrados y el método del gradiente descendente con retropropagación, logra adaptar los parámetros de las funciones de membresía del FIS, emulando el comportamiento de un conjunto de datos de entrenamiento dado. Esta herramienta permite usar un argumento opcional para la validación del modelo. El tipo de validación que se lleva a cabo con esta opción es una comprobación y el argumento es un conjunto de datos denominado conjunto de datos de comprobación (*checking data*).

La herramienta `anfis` solamente se usa para sistemas de tipo Sugeno y deben cumplir con las siguientes propiedades:

- Los sistemas tipo Sugeno deben ser de primer orden o de orden cero
- Tener una sola salida, usando defusificación promedio ponderada. Todas las funciones de membresía de salida deben ser del mismo tipo y ser lineales o constantes
- No tener reglas compartidas. Reglas diferentes no pueden compartir las mismas funciones de membresía, es decir el número de funciones de membresía de salida debería ser igual al número de reglas
- Tener un peso unitario para cada regla

Sintaxis: `[fis,error,stepsize,chkFis,chkErr] = anfis(trnData,initFIS,trnOpt,dispOpt,chkData)`

Los argumentos de entrada de la herramienta `anfis` son:

`trnData`: Llamado conjunto de datos de entrenamiento, esta matriz contiene todos los datos de entrada en las primeras columnas mientras que la última columna contiene un simple vector de datos de salida.

`initFis`: Es el nombre de un sistema de inferencia difuso usado para proporcionar a `anfis` un conjunto inicial de funciones de membresía de entrenamiento

`nuMFs`: Es el número de funciones de membresía

`trnOpt`: Es un vector de opciones de entrenamiento

`dispOpt`: Es un vector de opciones de pantalla, es decir mediante la configuración de opciones, permite visualizar diferentes tipos de información en la pantalla de comandos de MATLAB

`chkData`: Es el nombre de un conjunto de datos de comprobación que se utiliza para validar el modelo. Este conjunto de datos representa una matriz en el mismo formato que el conjunto de *training data*

`optMethod`: Es un método opcional de optimización, usado en el entrenamiento de los parámetros de la función de membresía. El valor 1 selecciona el método híbrido y 0 el método de retropropagación.

Los argumentos de salida de la herramienta `anfis` son:

`fis`: Es la estructura FIS cuyos parámetros se establecen de acuerdo con un criterio de error de entrenamiento mínimo.

`error` o `chkErr`: Es una matriz de errores cuadráticos medios, que representan el error de datos de entrenamiento y el error de datos de comprobación, respectivamente. La función sólo devuelve `chkErr` cuando se suministra `chkData` como un argumento de entrada.

`stepsize`: Es el tamaño de saltos de cada repetición (*epoch*) de entrenamiento y es una matriz de escalares

`chkFis`: Es la estructura FIS cuyos parámetros se establecen de acuerdo con un criterio de error de comprobación mínimo. La función sólo devuelve `chkFis` cuando se suministra `chkData` como un argumento de entrada.

2.4. Procedimiento para la obtención del modelo del flujo vehicular

El objetivo de este proyecto de investigación, es encontrar el modelo del sistema a partir de datos de entrada y salida conocidos. Luego se desea predecir su comportamiento en las siguientes intersecciones. A continuación se describe el procedimiento para la obtención del modelo predictivo del sistema.

2.4.1. Procesamiento de datos

Una vez obtenidos los datos conocidos, se procede a analizar y adecuar los mismos para su uso. Como se mencionó en el capítulo anterior, los datos constan de dos columnas: Tiempo y Cantidad de vehículos, donde cada muestra se encuentra adquirida con un período de muestreo de 15 minutos, desde las 6:00 hasta las 20:00, lo cual da como resultado 56 muestras conocidas. La herramienta `genfis1` de MATLAB es capaz de obtener un modelo del sistema a partir de estas 56 muestras, sin embargo, al ser una cantidad pequeña, resulta difícil obtener un modelo suficientemente general que opere correctamente para situaciones ligeramente distintas. Para solucionar esto se aplica una técnica de procesamiento de señales llamada Remuestreo de la señal. Lo que se intenta realizar con esta técnica es tener un número mayor de muestras que analizar, para lo cual se añade un ruido gaussiano cuyos valores se aproximen en amplitud a los valores de la señal real. Por último, se incrementa la señal repitiéndola varias veces a lo largo del eje del tiempo, es decir, si la señal tiene 56 muestras, al repetirla 4 veces a la misma frecuencia a lo largo del eje del tiempo, se tendrán 224 muestras que analizar. De esta manera las herramientas `genfis1` y `anfis` se ajustarán de manera apropiada para la obtención de un modelo más general.

Para este proyecto de investigación, se ha añadido un ruido aleatorio cuyos valores están en un rango de ± 2 respecto a los valores reales. En la figura 2.3, se puede visualizar de color azul la gráfica de los datos de entrada y de color rojo la gráfica del ruido aleatorio generado. Luego se concatenan 20 veces el conjunto de muestras a lo largo del tiempo, es decir, si se tenían 56 muestras al concatenar 20 veces se tiene una función con 1120 muestras, tal como se muestra en la figura 2.4

En la figura 2.4 se pueden observar 3 gráficos: el primero muestra los datos de entrada sin ningún procesamiento, el segundo muestra los datos de entrada con el ruido añadido y el tercer gráfico indica la función finalmente procesada, es decir, con ruido y repetida 20 veces.

En la predicción de series de tiempo, es necesario utilizar los valores conocidos de la serie temporal, hasta un punto en el tiempo, t , para predecir los valores de algún punto en el futuro $t + \Delta$. El método estándar para este tipo de predicción es crear un mapeo

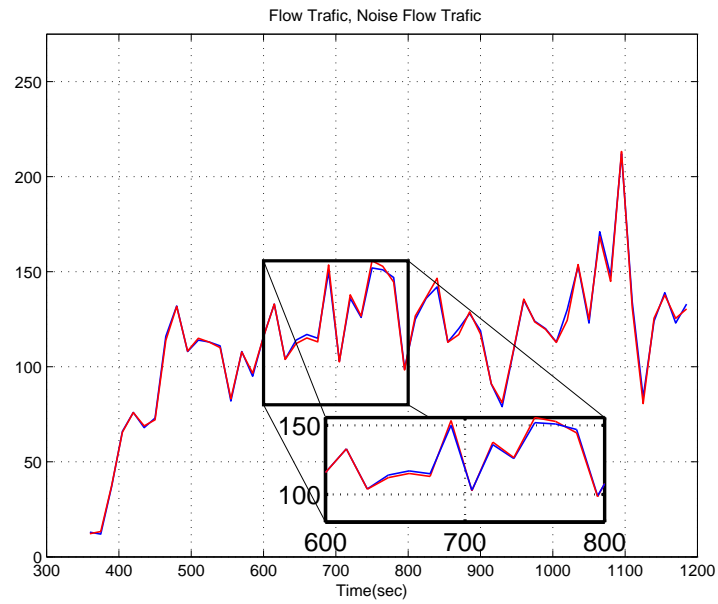


Figura 2.3: *Ruido Aleatorio*

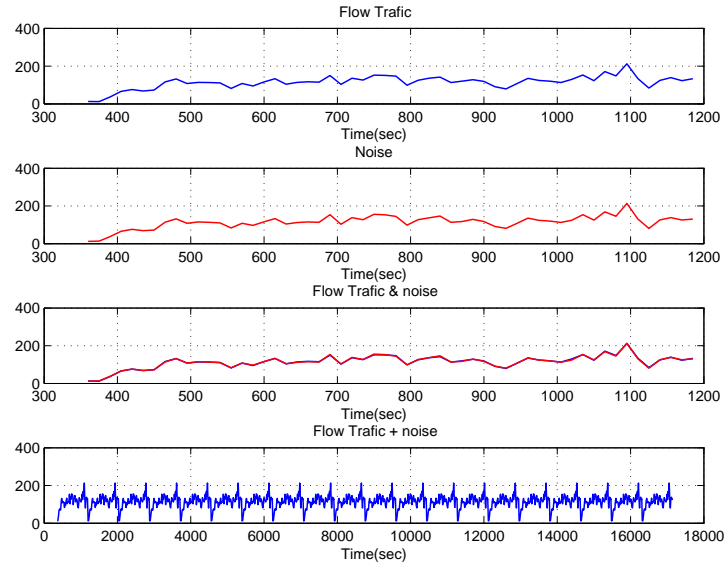


Figura 2.4: *Procesamiento de datos*

desde la muestra D de los puntos de datos, muestreando cada Δ unidades en el tiempo, $(x(t - (D - 1)\Delta), \dots, x(t - \Delta), x(t))$, para predecir un valor futuro $x = (t + \Delta)$, donde D representa el número de muestras y Δ la frecuencia de muestreo. Algunas configuraciones típicas para series de tiempo incluyen los valores $D = 4$, $\Delta = 1$ y $D = 4$, $\Delta = 2$, se analizará cual de estos valores brinda mejores resultados. Para cada t , los datos de entrenamiento de entrada para el `anfis` se representa con un vector de 4 columnas de la siguiente forma:

$$w(t) = [x(t - 3), x(t - 2), x(t - 1), x(t)]$$

Los datos de entrenamiento de salida corresponden a la predicción de la trayectoria

$$s(t) = x(t + 1)$$

Sin embargo al aplicar esta combinación de datos pasados, el modelo predecido no se ajustó correctamente por lo que se estableció $D = 4$ y $\Delta = 2$, entonces para cada t , los datos de entrenamiento de entrada para el `anfis` es un vector de 4 columnas de la siguiente forma:

$$w(t) = [x(t - 6), x(t - 4), x(t - 2), x(t)]$$

Los datos de entrenamiento de salida corresponden a la predicción de la trayectoria.

$$s(t) = x(t + 2)$$

Para cada t , que varía desde 8 hasta 1117, los datos de entrada/salida son una estructura cuyo primer componente es la entrada de cuatro dimensiones w , y cuyo segundo componente es la salida s . Para los 1110 valores de datos de entrada/salida los primeros 500 valores de datos son usados para el entrenamiento del `anfis`, mientras los otros valores son usados como datos de comprobación para la validación del modelo difuso identificado.

2.4.2. Construcción del modelo ANFIS

Para obtener el modelo `anfis` es necesario usar la herramienta `genfis1` para generar una matriz FIS inicial de los datos de entrenamiento. Este comando requiere solamente configurar los siguientes argumentos:

data: Se ingresa una matriz con los datos anteriormente procesados, para este caso es una matriz de 5 columnas en la cual las 4 primeras son las entradas y la quinta es la salida.

numMFS: Se han realizado varias pruebas para determinar el número de funciones de membresía que se asocian a cada entrada. El valor que entregó mejores resultados con respecto a otras pruebas realizadas fue de 3 funciones de membresía

inmfype: Para evitar que MATLAB consuma recursos al realizar cálculos complicados es necesario definir funciones sencillas, en este caso de tipo triangular, esto es útil para futuras implementaciones en sistemas basados en microprocesador.

outmfype: Para este caso no se tienen funciones de membresía de salida sino valores numéricos, por lo que se ha elegido la opción “linear”

Luego de haber generado un FIS no adaptado, se usa la herramienta *anfis* de MATLAB para obtener un FIS adaptado. Para hacer uso de esta herramienta es necesario ingresar como argumentos los *training data* y los *checking data*. Con los *training data* el sistema aprende como se comporta el flujo vehicular, una vez que el sistema se encuentra entrenado, realiza una validación del modelo haciendo uso de los *checking data*, creando de esta manera un modelo con datos de *training* y *checking*. La finalidad de la herramienta es que el error existente entre el modelo de *training* y *checking* sea el mínimo, con eso se garantiza que el modelo obtenido es suficientemente general. Los ajustes de los argumentos de la herramienta *anfis* son:

trnData: Es la matriz que tiene los datos de *training*, compuesta por 4 columnas de entrada y una de salida

fismat: Es la salida del sistema de inferencia difuso generado por la herramienta *genfis1*, haciendo uso de los *training data*.

epoch_n: Es el número de repeticiones de entrenamiento que se desean realizar, para este caso se ha configurado en 40.

dispOpt: Esta opción permite visualizar la siguiente información: acerca del *anfis* con la opción “1”, acerca del error en cada repetición (epoch) con la opción “2”, información del tamaño de pasos o muestras con la opción “3” y los resultados finales con la opción “4”. Por defecto se encuentra configurada la opción “1”, sin embargo cuando se configura la opción *NaN*, MATLAB lo interpreta como si estuviera determinada la opción por defecto, es decir 1, como en este caso.

chkData: Es la matriz que tiene los datos de *checking*, se usan para la comprobación del modelo

optMethod: Es el método que se usa para establecer el método de ajuste de los parámetros de las funciones de membresía, que puede ser el de mínimos cuadrados o el de gradiente descendiente con retropropagación. Este argumento se configura con la opción “1” para el método híbrido y “0” para el método de retropropagación. Para este caso se ha elegido la opción “1”.

```
sintaxis: [trn_fismat, trn_error, stepsize, chkFis, chk_error] =  
anfis(trn_data, fismat, epoch_n, dispOpt, chk_data, 1)
```

Hasta el momento, se presentó el procedimiento llevado adelante para la obtención del

modelo `anfis`, realizando para esto un procesamiento de datos de entrada y obteniéndose luego la construcción del modelo. Para esto se utilizaron muestras pasadas como datos de entrada y se ha obtenido una salida futura. En el siguiente capítulo se aplicará este procedimiento, para la obtención del modelo del flujo vehicular de la Avenida Héroes de Verdeloma en 3 intersecciones semafóricas consecutivas.

Capítulo 3

MODELADO DEL TRÁFICO VEHICULAR DE LA AVENIDA HÉROES DE VERDELOMA

Introducción

En esta tercera parte del proyecto de investigación, se explica la obtención del modelo del *anfis* a lo largo de la avenida Héroes de Verdeloma, cuyas intersecciones semafóricas son las calles: Mariano Cueva, Luis Cordero y Nicanor Merchán. Luego se analizan los errores de: entrenamiento, comprobación y predicción, para proceder con la validación del modelo obtenido

3.1. Antecedentes

En este proyecto de investigación, se pretende obtener el modelo del flujo vehicular que existe sobre la Avenida Héroes de Verdeloma de la Ciudad de Cuenca. Esta avenida tiene dos carriles en el sentido ESTE - OESTE (de ida) y dos carriles en el sentido OESTE - ESTE (de regreso). Esta vía inicia en la calle Gil Ramírez Dávalos y termina en la Avenida de las Américas, en todo este trayecto es intersectada por varias calles secundarias, entre ellas se encuentran: Barrial Blanco, Mariano Cueva, Luis Cordero, Juan Bautista Aguirre, General Torres, Tarqui, Nicanor Merchán y Estevez de Toral. Como se explicó en capítulos anteriores, la Alcaldía de la ciudad facilitó datos de un aforo vehicular elaborado a lo largo de esta avenida, para la elaboración de este proyecto. En el carril de sentido ESTE-OESTE, existen tres semáforos instalados, los cuales se encuentran en las intersecciones: Mariano Cueva, Luis Cordero y Nicanor Merchán. Se utilizarán los datos obtenidos del aforo vehicular en estas tres intersecciones semafóricas para la obtención del modelo.

En la figura 3.1, se pueden observar los semáforos ubicados en las intersecciones que fueron mencionadas en el párrafo anterior. En esta sección del proyecto de investigación

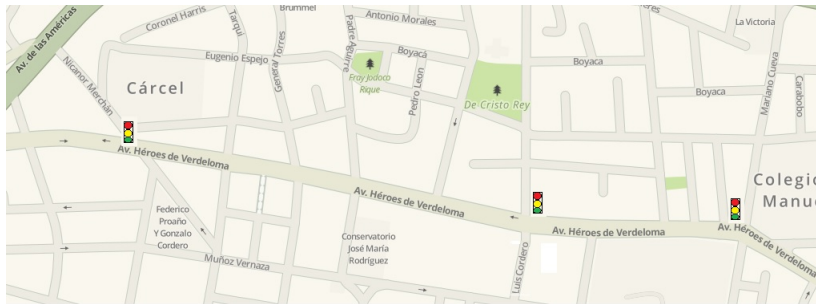


Figura 3.1: Avenida Héroes de Verdeloma

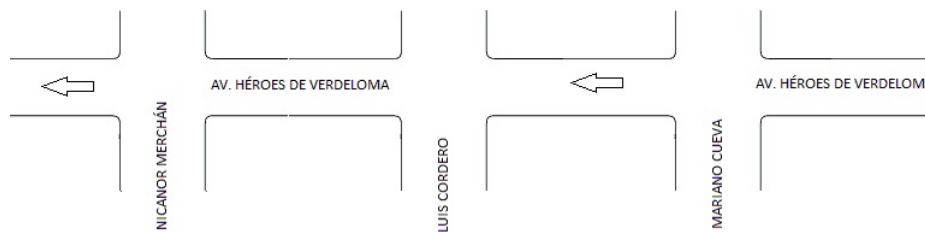


Figura 3.2: Intersecciones semafóricas

se obtendrá el modelo del flujo vehicular en cada intersección donde se encuentra un semáforo, prediciendo muestras futuras. En la figura 3.2, se puede observar el esquema general del sistema en el cual se evaluará el tránsito de vehículos.

A continuación se explica la obtención del modelo del flujo vehicular, en cada intersección semafórica.

3.2. Modelado del flujo vehicular en la intersección Mariano Cueva

En el capítulo anterior se explicó el procedimiento que se debe seguir, para la obtención del modelo del sistema, a continuación se ejecutará el procedimiento con datos reales de entrada y salida en la Avenida Héroes de Verdeloma

3.2.1. Procesamiento de datos

Los datos del aforo vehicular que han sido facilitados por la Alcaldía de la Ciudad, se pueden observar en el Anexo A. Allí se muestran 56 pares de datos, desde las 6:00 hasta las 20:00, con un período de adquisición de 15 minutos. Estos pueden ser representados gráficamente mediante la herramienta `plot(x, y)` de MATLAB

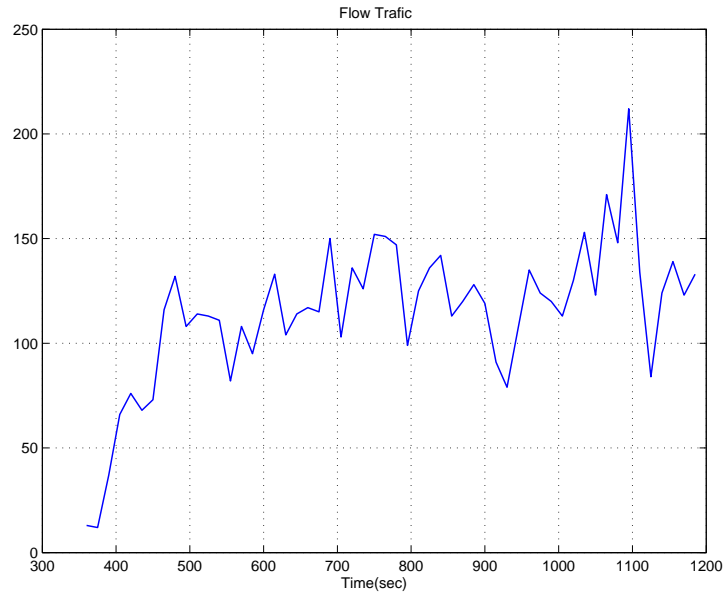


Figura 3.3: Datos vs tiempo en la intersección Mariano Cueva

En la figura 3.3, se muestra el flujo vehicular existente en la Avenida Héroes de Verdeloma y la primera intersección semafórica, la calle Mariano Cueva. Se puede observar la variable *tiempo* representada en segundos en el eje *x*, y en el eje *y*, la cantidad de vehículos que han circulado, en el lapso de tiempo antes especificado.

Una vez obtenidos los datos de entrada, se procede a realizar el remuestreo, añadiendo un ruido y repitiendo la función a lo largo del eje *x* 20 veces, como ya se explicó en el capítulo 2.

En el Anexo B.1, se puede observar que los datos del aforo vehicular se almacenan en los vectores *a* y *time*, donde: *a* contiene la cantidad de vehículos y *time* la hora. Luego se añade un ruido aleatorio, para lo cual se utiliza la función `random`, en la que se debe configurar: el tipo de distribución, la media y la desviación estándar con la que se desea trabajar. En este caso se utiliza: una distribución aleatoria de tipo normal, la media que se obtiene a partir de todos los valores del vector *a* y la desviación estándar a la cual se le asigna un valor de 2. Es decir, los valores reales de entrada son considerados como una referencia media y se les aplica una suma o resta aleatoria de ± 2 para generar el nuevo vector de datos de entrada *noise*.

Luego de añadir el ruido a los datos de entrada, se procede a aumentar el número de muestras obtenidas repitiéndolas 20 veces a lo largo del eje *x*, para lo cual se utiliza la función `vertcat`, que concatena arreglos de datos verticalmente. Se genera entonces el vector *Signal* que contiene al vector *noise* repetido 20 veces. En la figura 3.4, se pueden observar 4 gráficas, la primera representa los datos del aforo vehicular, la segunda

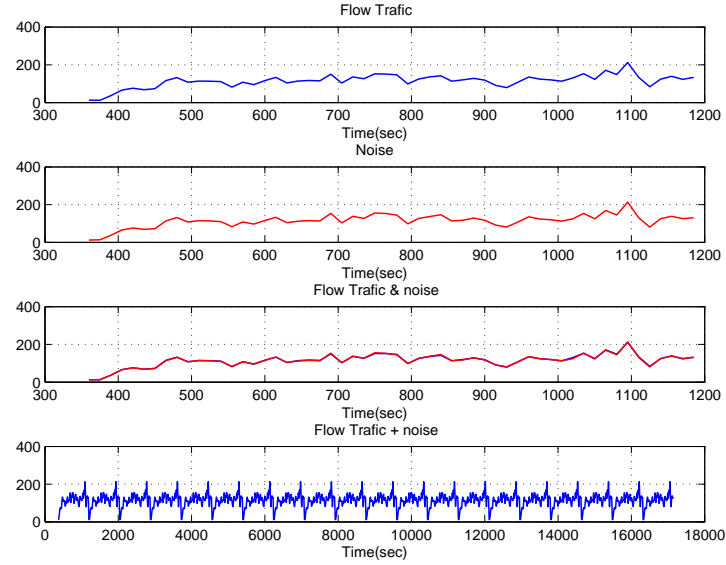


Figura 3.4: Datos de entrada Procesados

muestra el ruido generado a partir de los datos conocidos, en la tercera se puede observar el contraste entre la primera y la segunda gráfica y en la última se puede observar la función de entrada repetida 20 veces, para aumentar la cantidad de muestras a analizar.

Luego de procesar los valores de entrada, se establecen los datos de *training* y *checking*, los cuales son necesarios para usar las herramientas de MATLAB: `genfis1` y `anfis`

Como se explicó en el capítulo 2, se usará el 50 % de los datos para entrenamiento y el 50 % para comprobación, luego se pretende generar un `anfis` que pueda predecir la muestra $x(t+2)$, es decir 2 saltos en el futuro, para lo cual se usa la trama: $[x(t-6), x(t-4), x(t-2), x(t); x(t+2)]$, donde $x(t-6), x(t-4), x(t-2), x(t)$ serán los datos de entrada y $x(t+2)$ la salida.

En el Anexo B.2, se puede observar la función `zeros` que genera una matriz de ceros cuyo tamaño es de m filas y n columnas, en este caso se generan las matrices `trn_data` y `chk_data` cuyo tamaño es de 555 filas y 5 columnas. Esta operación se realiza para garantizar la inicialización de todos los valores de las matrices en cero

A continuación se ingresan los valores en las matrices, pudiéndose observar que se cumple la trama dispuesta, $[x(t-6), x(t-4), x(t-2), x(t); x(t+2)]$. La primera columna se llena con la muestra número 2 hasta la 556, la siguiente columna se llena con 2 saltos de muestras futuras, es decir, desde la muestra 4 hasta la 558, repitiendo este procedimiento hasta completar la matriz de 555 filas y 5 columnas.

En la figura 3.5 se muestra parte de la matriz de *training*, pudiéndose observar que consta

	1	2	3	4	5
1	15.6678	67.7243	65.3846	116.6852	113.5389
2	32.4823	76.6375	72.1328	139.1568	111.3002
3	67.7243	65.3846	116.6852	113.5389	119.0698
4	76.6375	72.1328	139.1568	111.3002	112.4508
5	65.3846	116.6852	113.5389	119.0698	81.8739
6	72.1328	139.1568	111.3002	112.4508	109.4295
7	116.6852	113.5389	119.0698	81.8739	94.5901
8	139.1568	111.3002	112.4508	109.4295	115.7517
9	113.5389	119.0698	81.8739	94.5901	135.9794
10	111.3002	112.4508	109.4295	115.7517	106.8181
11	119.0698	81.8739	94.5901	135.9794	116.8344
	555	84.1547	136.7730	136.0653	15.6678
					67.7243

Figura 3.5: Matriz de datos de training

de 5 columnas, las 4 primeras son los datos de entrada mientras que la columna 5 contiene los datos de salida. Se puede visualizar, que desde la columna 1 hasta la 5 se van prediciendo 2 saltos futuros hasta llegar a obtener la salida. De esta manera se interpretan los datos de entrada y salida en las funciones `genfis1` y `anfis`.

3.2.2. Construcción del modelo ANFIS

Después de haber procesado los datos y una vez generadas las variables, se procede a obtener el modelo `anfis`

Para generar el `anfis`, es necesario en primer lugar obtener el FIS no adaptado del sistema, el cual se genera mediante la herramienta `genfis1`, asignando el resultado a la variable `fismat`, como se puede observar en el Anexo B.3. Los datos de entrada para esta herramienta son los valores introducidos en la matriz `trn_data`, el FIS es generado aplicando 3 funciones de membresía de tipo triangular y una salida de tipo lineal.

A continuación, se procede con la obtención de un FIS adaptado, para lo cual se usa la herramienta `anfis` de MATLAB, como se muestra en el Anexo B.3. Los parámetros para la obtención del `anfis` son: los datos de entrenamiento, es decir la matriz `trn_data`, la variable `fismat` que contiene al FIS generado, el número de repeticiones de entrenamiento que se desea para la obtención del modelo ($epoch.n = 40$), los datos de comprobación que son ingresados en la matriz `chk_data`. Adicionalmente se configura la opción "1", para indicarle a la herramienta que ajuste los parámetros de las funciones de membresía mediante el método híbrido.

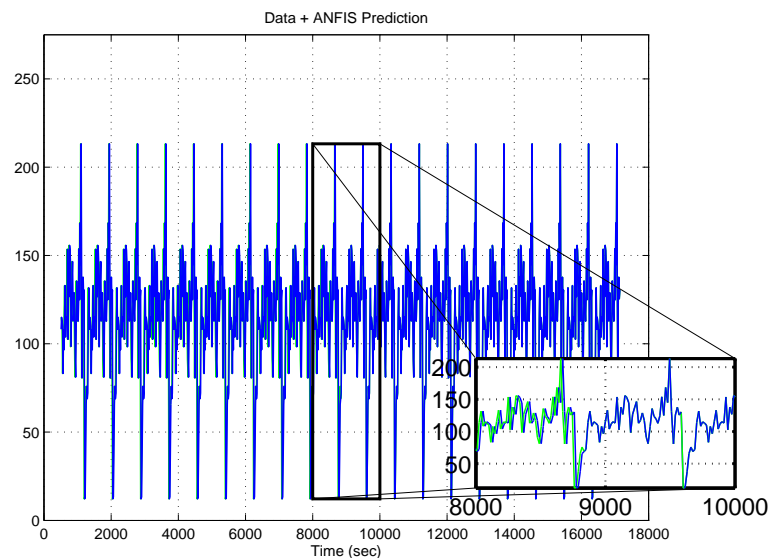


Figura 3.6: Datos de la salida del modelo ANFIS vs. Tiempo

Al generar el `anfis`, MATLAB crea las variables:

`trn_fismat`, `trn_error`, `stepsize`, `chkFis`, `chk_error`, que fueron explicadas en el capítulo 2, las cuales son usadas para obtener los resultados del modelado del sistema.

Luego de generar el `anfis`, se usa la función `evalfis`, que permite evaluar la salida del FIS generado para un conjunto de entradas determinado.

Sintaxis: `output = evalfis(input, fismat)`

Donde los argumentos de entrada son:

`input` : Es un valor numérico o una matriz que contiene los valores de entrada

`fismat` : Es la estructura FIS que se desea evaluar

Obteniendo el argumento de salida:

`output` : Es una matriz de tamaño $m \times l$, donde m representa el número de entradas, y l es el número de salidas para el FIS.

En el Anexo B.5 se puede observar el cálculo del FIS. Los valores de entrada son las 4 primeras columnas de las matrices `trn_data` y `chk_data` y la estructura que se desea evaluar es `chkFIS`

En la figura 3.6 se puede observar en color verde la gráfica de los datos de entrada vs. tiempo y en color azul los datos de la salida del modelo `anfis` vs tiempo. Se puede observar que existe un error entre las dos gráficas hasta la mitad del recorrido, pero

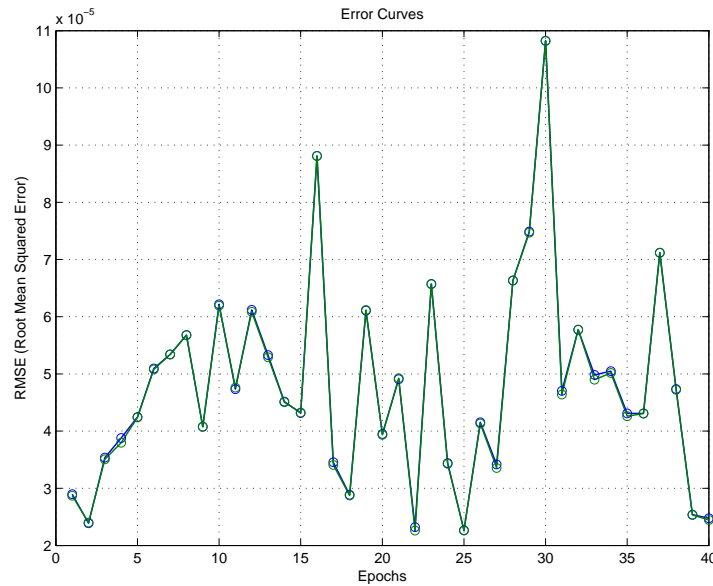


Figura 3.7: Errores de training y checking vs. Repeticiones de entrenamiento

luego las gráficas se ajustan de tal manera que ya no se puede divisar la gráfica de color verde, lo que indica que el modelo obtenido fue correctamente adaptado por el algoritmo híbrido

Al igual que `chkFis`, MATLAB devuelve las variables *trn_error* y *chk_error*, con las cuales se puede validar el modelo. Los valores de los errores de entrenamiento y de comprobación son los que determinan si el modelo obtenido es el apropiado. En la figura 3.7, se puede observar en color azul el error de *training* y en color verde el error de *checking*. Se puede observar también que los valores de los errores a lo largo del eje *x* (repeticiones de entrenamiento ó Epoch), son menores a 11×10^{-5} , lo que indica que el resultado del *anfis* obtenido con dos saltos futuros $x(t+2)$, se adaptó correctamente

Para terminar, se encuentra la diferencia existente entre los valores de los datos de entrada y los generados por el *anfis*, con esto se puede conocer cual es la desviación existente entre los datos conocidos y los obtenidos.

En la figura 3.8, se puede observar el resultado de restar los valores de entrada y los valores arrojados por la salida del modelo *anfis* obtenido. Se puede notar que durante la mitad del tiempo, existe un error con valores entre 70 y -120, lo cual pertenece al período de entrenamiento. Luego de eso el error es muy cercano a 0, lo que indica que el modelo *anfis* se adaptó correctamente y se obtuvieron resultados suficientemente generales.

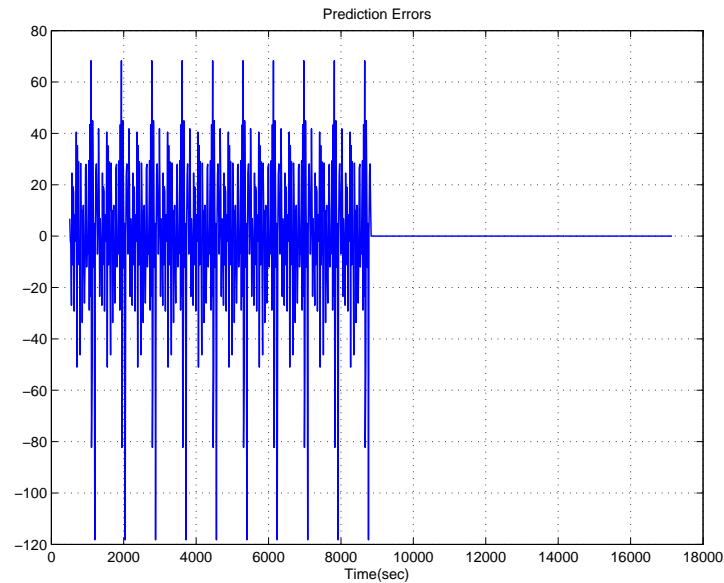


Figura 3.8: Errores de predicción

3.3. Modelado del flujo vehicular en la intersección Luis Cordero

La siguiente intersección semafórica que atraviesa a la avenida Héroes de Verdeloma es la calle Luis Cordero. Para hallar el modelo del flujo vehicular, se realiza un procedimiento idéntico al de la sección anterior.

3.3.1. Procesamiento de datos

Los datos del aforo vehicular para esta intersección, se pueden observar en el Anexo B. Al igual que en el caso anterior, se muestran 56 pares de datos, desde las 6:00 hasta las 20:00, con un período de adquisición de 15 minutos. La representación gráfica de los datos del aforo vehicular se muestra en la figura 3.9. En el eje x se muestra el tiempo en segundos, y en el eje y la cantidad de vehículos.

A continuación se realiza el remuestreo de los datos de ingreso, añadiendo un ruido aleatorio y concatenando el conjunto de muestras 20 veces a lo largo del eje x , usando el mismo método que se realizó en la intersección anterior. En la figura 3.10, se pueden observar 4 gráficas que muestran los datos del aforo vehicular, el ruido aleatorio generado, el contraste entre los datos del aforo con el ruido aleatorio y el conjunto de muestras concatenadas 20 veces, obteniendo de esta manera los datos que van a ser analizados

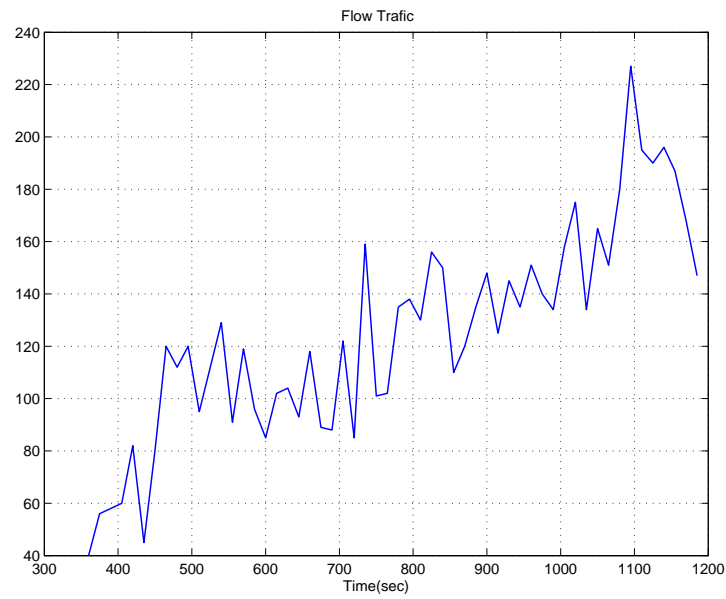


Figura 3.9: Datos del flujo vehicular vs Tiempo, para la intersección Luis Cordero

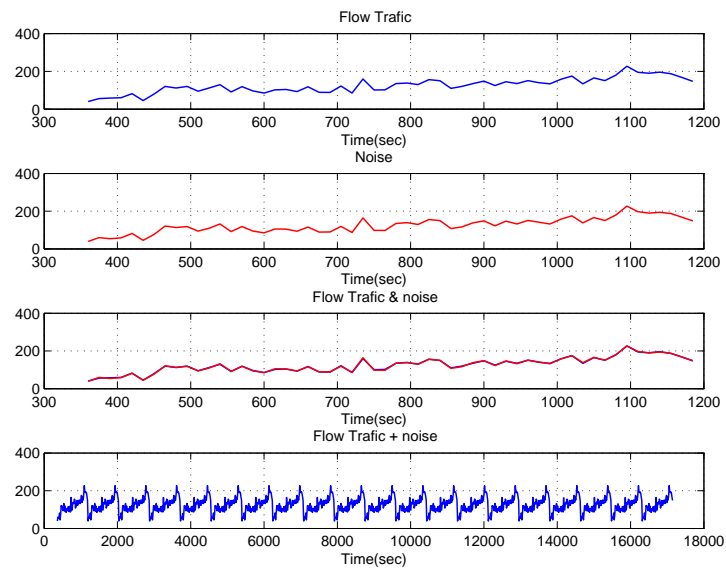


Figura 3.10: Datos de entrada procesados, intersección Luis Cordero

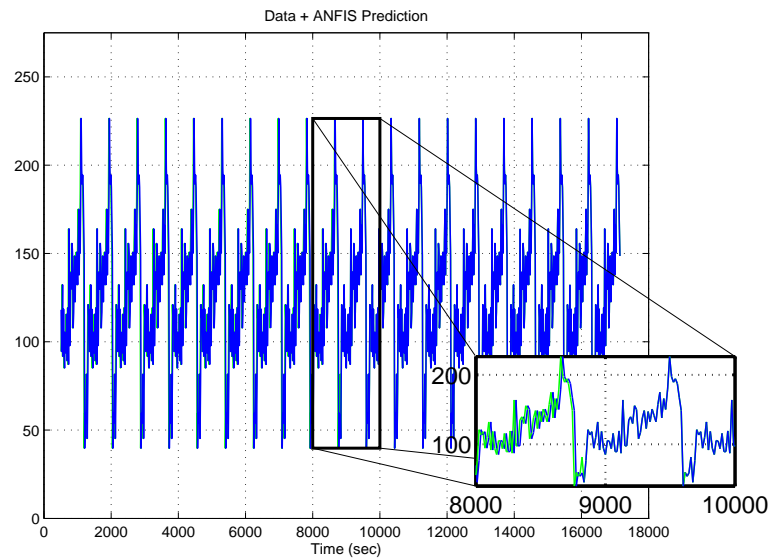


Figura 3.11: Datos de entrada y Modelo ANFIS vs. Tiempo

Una vez que los valores se encuentran procesados, se crean las matrices de *training* y *checking*

Como se explicó en el capítulo 2, se usará el 50 % de los datos para entrenamiento y el 50 % para comprobación. Para la generación del *anfis*, se crean las matrices de *training data* y *checking data*, con el mismo esquema de muestras pasadas $x(t-6)$, $x(t-4)$, $x(t-2)$, $x(t)$ para luego predecir $x(t+2)$, como se observa en el Anexo B.2

3.3.2. Construcción del modelo ANFIS

De la misma manera que en la intersección anterior, se genera el FIS usando las herramientas *genfis1* y *anfis* de MATLAB.

Es necesario mencionar que se parte de un FIS no adaptado y mediante la herramienta *anfis* se genera un FIS adaptado, luego se usa la herramienta *evalfis*, para obtener el cálculo de la inferencia difusa.

En la figura 3.11 se puede observar en color azul la gráfica de los datos de entrada vs. tiempo y en color verde los datos del modelo *anfis* vs tiempo, se puede notar la diferencia de los valores entre las dos funciones en la etapa de *training*, al inicio del intervalo de tiempo, pero luego en la etapa de *checking* la diferencia ya no es notoria, debido a que el modelo se ajusta de manera correcta.

Luego de obtener el *anfis*, MATLAB genera las variables *trn_error* y *chk_error*,

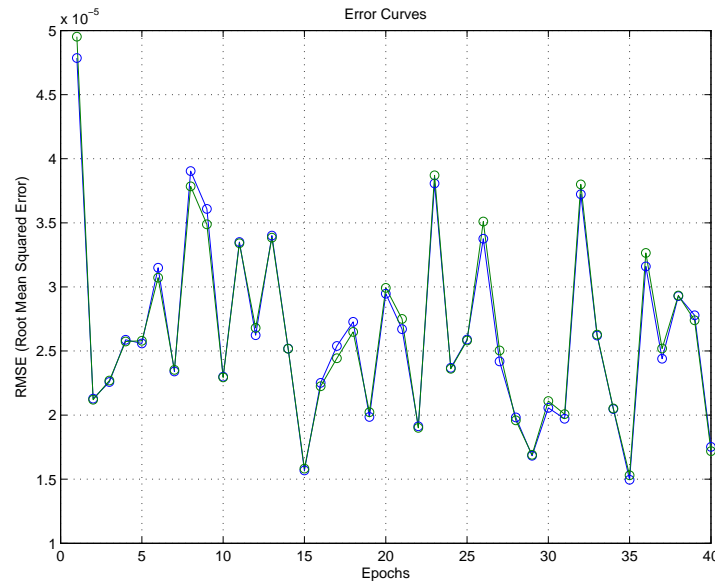


Figura 3.12: Errores de training y checking vs. Repeticiones de entrenamiento

las cuales son usadas para determinar el error de *training* y *checking* existente. Realizando un análisis de estos errores es posible determinar si el modelo obtenido es el adecuado. En la figura 3.12, se puede observar en color azul el error de *training* y en color verde el error de *checking*, se puede notar que los valores de los errores a lo largo del *ejeX* son menores a 5×10^{-5} , es decir aproximadamente 0, lo que indica que el modelo obtenido es apropiado

Por último, se encuentra la diferencia existente entre los valores de entrada y los datos de salida generados por el *anfis*, con el objetivo de conocer cual es la desviación existente entre ellos

En la figura 3.13, se puede observar que existe un error entre 70 y -110 en el instante de *training*, luego durante el *checking* el error es muy cercano a 0, lo que indica que el modelo *anfis* se generó correctamente.

3.4. Modelado del flujo vehicular en la intersección Nicanor Merchán

La siguiente intersección semafórica que atraviesa a la avenida Héroes de Verdeloma es la calle Nicanor Merchán, se realizará una explicación breve, pues la obtención del modelo ya ha sido explicado en los dos casos anteriores.

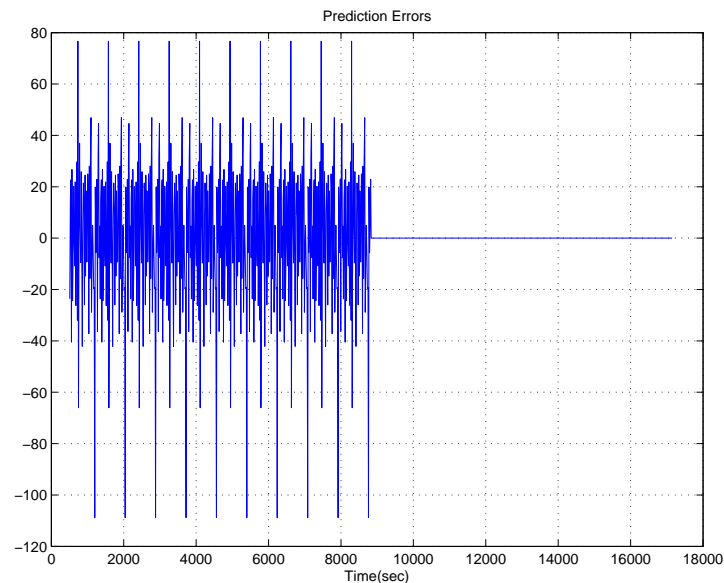


Figura 3.13: Errores de predicción

3.4.1. Procesamiento de datos

Los datos del aforo vehicular en esta intersección, se encuentran en el Anexo A. En la figura 3.14 se puede observar la gráfica de los datos de entrada vs tiempo.

Luego de conocer cuales son los datos de entrada, se realiza el remuestreo de los datos de ingreso, se sigue el mismo procedimiento que los casos anteriores, se añade un ruido aleatorio y se concatenan 20 veces el conjunto de datos, como se muestra en la figura 3.15

A continuación, se generan las matrices de *training* y *checking*, las cuales contienen los valores de entrada y salida del FIS. (véase Anexo B.2)

3.4.2. Construcción del modelo ANFIS

De la misma manera que las intersecciones anteriores, se parte de un FIS no adaptado y mediante la herramienta `anfis`, se obtiene un FIS adaptado, por lo que se usa las herramientas `genfis1` y `anfis` de MATLAB

Luego de obtener el `anfis`, se hace uso de la herramienta `evalfis` para realizar el cálculo de la inferencia difusa

En la figura 3.16 se puede observar el FIS generado, se muestra en color verde la gráfica de los datos de entrada, y en color azul la gráfica del modelo `anfis`, al igual que los

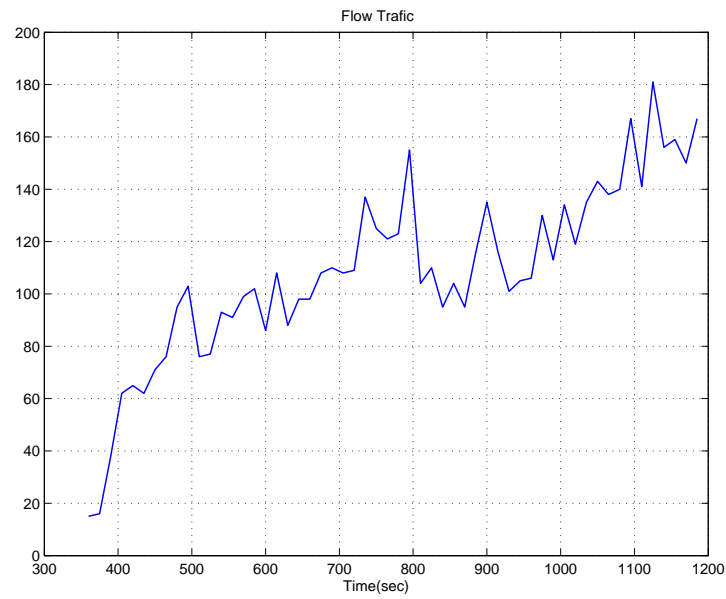


Figura 3.14: Datos vs Tiempo, intersección Nicanor Merchán

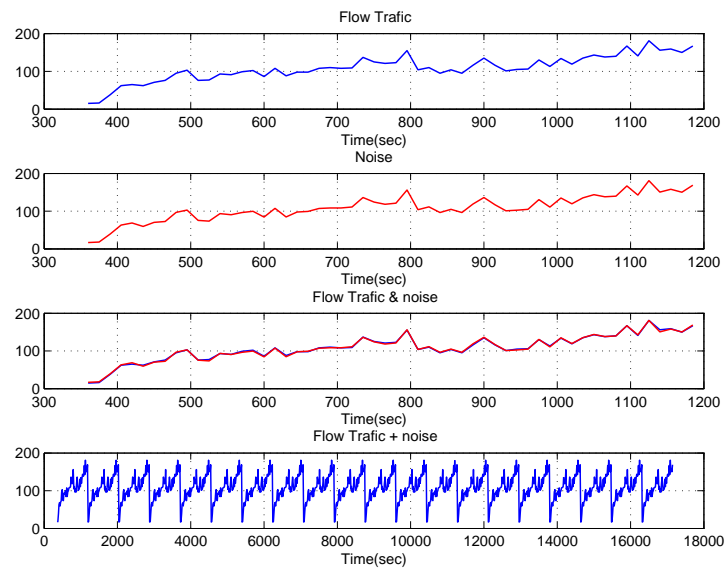


Figura 3.15: Datos de entrada procesados, intersección Nicanor Merchán

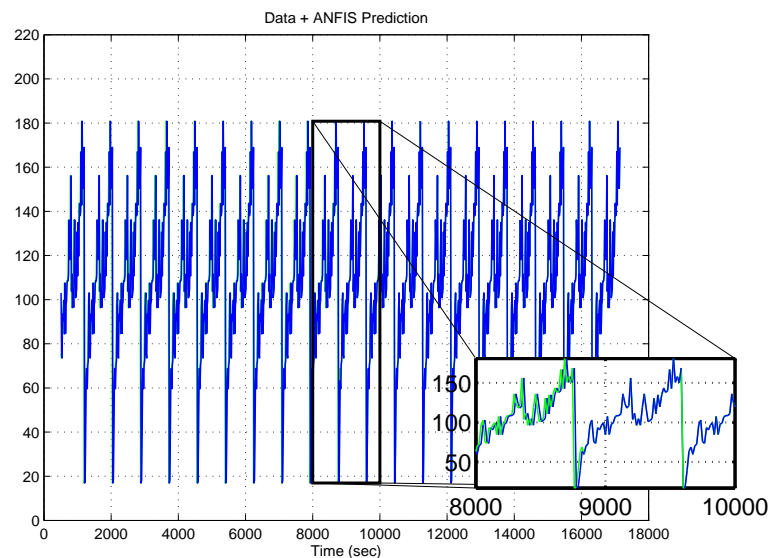


Figura 3.16: Datos de entrada y Modelo ANFIS vs. Tiempo

casos anteriores el modelo se ajusta de manera correcta, se puede observar en el instante de *training* una variación de valores entre las dos funciones, pero en el intervalo de *checking* el modelo se ajusta apropiadamente.

Para la validación del modelo, la función `anfis` devuelve las variables `trn_error` y `chk_error`, con ellas es posible conocer cual es el error de *training* y *checking* existente al generar el FIS. En la figura 3.17, se puede observar que el error de *training* se encuentra graficado en color azul, mientras que el error de *checking* se encuentra en color verde, también se puede notar que los valores de los errores son menores a 3×10^{-5} , lo que indica que el resultado del `anfis` es suficientemente general, pues es aproximadamente 0.

Para finalizar, se demuestra la desviación existente entre los valores de los datos de entrada y los valores de salida generados por el `anfis`, realizando una resta entre ellos.

En la figura 3.18, se puede observar que existe un error entre 30 y -150 en el instante de *training*, luego, en el intervalo de *checking* el error es muy cercano a 0, lo que indica que el modelo `anfis` obtenido es apropiado.

Hasta el momento se ha realizado la obtención de un modelo del sistema, que permite predecir el flujo vehicular en una intersección semafórica, a partir de los datos pasados de flujo vehicular en esa misma intersección, se ha aplicado este método a lo largo de la Avenida Héroes de Verdeloma en 3 intersecciones semafóricas consecutivas y se ha podido comprobar que el método funciona para los 3 casos establecidos, prediciendo el

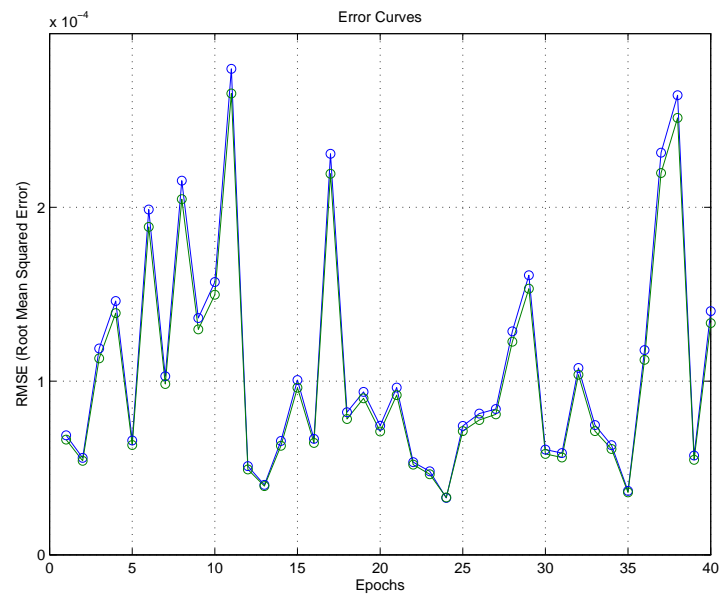


Figura 3.17: Errores de training y checking vs. Repeticiones de entrenamiento

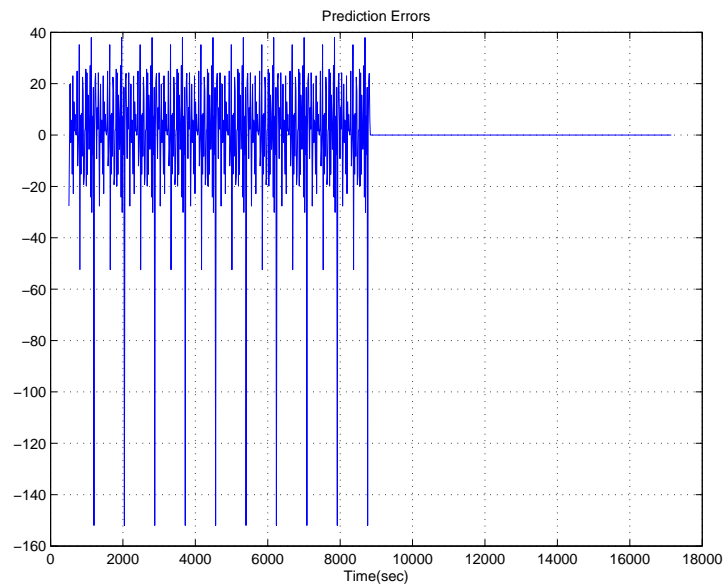


Figura 3.18: Errores de predicción

flujo para un instante $x(t + 2)$. En el próximo capítulo se aplicará el mismo método para la obtención de otro modelo, donde esta vez serán interpretados como salida, los datos de flujo vehicular correspondientes a la siguiente intersección semafórica.

Capítulo 4

MODELADO PREDICTIVO DE UNA INTERSECCIÓN SEMAFÓRICA Y SUS RESULTADOS

Introducción

En esta última parte del proyecto de investigación, se obtiene un modelo `anfis` de la Avenida Héroes de Verdeloma, en el cual, se interpretan como datos de entrada los obtenidos del aforo vehicular, en la intersección Mariano Cueva y como datos de salida los valores de las dos intersecciones siguientes: Luis Cordero y Nicanor Merchán, generalizando lo antes mencionado, se genera un modelo de una avenida cuyos datos de entrada pertenecen a una intersección A y los datos de salida a una intersección B, luego se analizan los resultados para validar el modelo obtenido

4.1. Antecedentes

En el capítulo 3, se obtuvo el modelo del flujo vehicular prediciendo 2 valores de muestras de datos futuros, en un instante de tiempo $x(t + 2)$, los resultados obtenidos permitieron observar un buen ajuste entre los datos experimentales y los predichos por el modelo. Para demostrar que el modelo del sistema obtenido es válido para rangos más grandes de muestras futuras, se procede a realizar el modelado en dos casos:

- Estableciendo como entrada los datos del aforo vehicular en la intersección semafórica Mariano Cueva y como salida los datos de la siguiente intersección semafórica, en este caso la calle Luis Cordero
- Estableciendo como entrada los datos del aforo vehicular en la intersección semafórica Luis Cordero y como salida los datos de la siguiente intersección semafórica, en este caso la calle Nicanor Merchán



Figura 4.1: Avenida Héroes de Verdeloma

En la figura 4.1 se puede observar la Avenida Héroes de Verdeloma, en la cual se muestran las 3 intersecciones semafóricas donde fue realizado el aforo vehicular. Se pueden observar unas flechas en color verde, que indican el sentido de circulación de los vehículos de los cuales se busca obtener el modelo, también se encuentran unas flechas en color rojo, las cuales representan las perturbaciones del sistema, es decir, al analizar el flujo vehicular desde la intersección “1” hasta la “2”, los vehículos pueden desviarse o integrarse por las vías que intersectan a la Avenida Héroes de Verdeloma, lo mismo sucede en la intersección “3”. Es necesario mencionar que en este proyecto de investigación, para obtener el modelo del sistema, no se consideran las perturbaciones existentes, por lo que se espera obtener un error en el ajuste, sin embargo, no significa que la obtención del modelo no sea apropiada.

A continuación se obtiene el modelo para cada caso.

4.2. Modelado predictivo del flujo vehicular en la intersección Luis Cordero

El método aplicativo para la obtención del modelo, es similar al que se explicó en el capítulo 3, pero esta vez, los datos de entrada pertenecen a una intersección y los de salida a otra.

4.2.1. Procesamiento de datos

En primer lugar se adecuan los datos que se van a utilizar, en este caso los adquiridos del aforo vehicular en la intersección Héroes de Verdeloma - Mariano Cueva y Héroes de Verdeloma - Luis Cordero. Los datos del aforo vehicular se encuentran en el Anexo A.1 y Anexo A.2 respectivamente. Para este modelado se crean los vectores a y b donde se almacenan los datos de cada intersección, como se muestra en el Anexo B.4

A continuación, se realiza el remuestreo de los datos, con el objetivo de tener una mayor cantidad de muestras para el entrenamiento del modelo. Para lo cual se procede a añadir un ruido aleatorio a los valores del aforo vehicular y luego se concatenan 20 veces cada conjunto de 56 muestras, obteniendo 1120 muestras para analizar.

En el Anexo B.4, se puede observar la creación de dos vectores: *noise1* y *noise2*, conteniendo cada uno de ellos 56 muestras a las que se añade un ruido aleatorio con un rango de ± 2 mediante la función `random` de MATLAB. Estos vectores contienen los valores de flujo vehicular en las intersecciones Mariano Cueva y Luis Cordero respectivamente. Luego, mediante la herramienta `vertcat`, se crean los vectores *Signal* y *Signal2*, que contienen el conjunto de muestras de cada intersección, pero esta vez, concatenadas 20 veces como se puede observar en la figura 3.4. Esta técnica de adecuación de datos es, hasta aquí, idéntica a la descrita en el capítulo 3.

Una vez que los datos han sido adecuados para ser utilizados, se procede a generar las matrices de *training* y *checking*, que son necesarias para generar el FIS.

De igual manera se inicializan en cero todos los valores de las matrices *trn_data* y *chk_data*, cuyo tamaño es de 555 filas y 5 columnas.

La elección de datos para crear las matrices es la misma que en el capítulo anterior, es decir, las cuatro primeras columnas contienen respectivamente las muestras $x(t-6)$ $x(t-4)$, $x(t-2)$, $x(t)$, esta vez, en lugar de llenar la quinta columna con las muestras en un instante de tiempo $x(t+2)$, se incluyen los datos del aforo vehicular de la siguiente intersección semafórica (Calle Luis Cordero), en ese instante de tiempo. Se puede observar que las cuatro columnas (entradas) se forman con los datos del vector *Signal*, al que se le asignó anteriormente los datos de la intersección Mariano Cueva, pero la quinta columna (salida) es creada con los datos del vector *Signal2*, al que se le asignó los datos de la intersección Luis Cordero. (véase Anexo B.4)

4.2.2. Construcción del modelo ANFIS

Luego de realizar el procesamiento de datos se obtiene el modelo del FIS, de la misma manera como se explicó en el capítulo 3, para lo cual se usan las herramientas `genfis1` y `anfis` de MATLAB.

En la figura 4.2 se puede observar la gráfica de la función de los datos de entrada, en color azul y los datos del modelo `anfis` en color verde, se puede apreciar que existe un contraste entre ellas durante todo el intervalo de tiempo, sin embargo, a la mitad, cuando inicia el intervalo de *checking*, la diferencia disminuye. Como se explicó en la sección 4.1 existen perturbaciones en el sistema debido a las calles existentes entre las intersecciones, por lo que existe un error al ajustar el modelo, mas adelante se ilustrará el error antes mencionado.

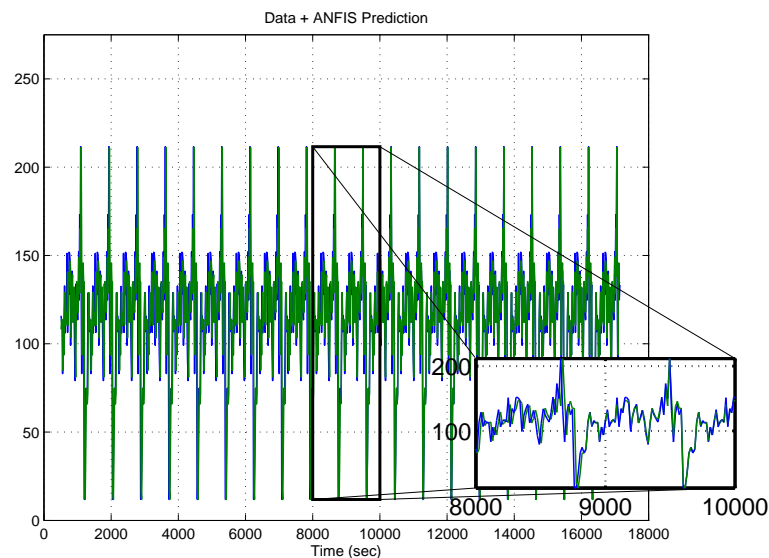


Figura 4.2: Datos de entrada y Modelo ANFIS vs. Tiempo

Luego se obtienen las variables *trn_error* y *chk_error*, para validar el modelo obtenido. En la figura 4.3, se puede observar en color azul el error de *training* y en color verde el error de *checking*. Puede apreciarse que el error de *training* está por debajo del valor 11 y el error de *checking* por debajo del valor 6, se puede notar que las gráficas no están distantes entre sí y sus valores aún siguen cercanos a 0. Considerando las perturbaciones existentes en el transcurso de la vía entre las dos intersecciones, es posible indicar que el modelo obtenido es apropiado.

Para finalizar, se encuentra la desviación existente entre los valores de los datos de entrada procesados y los generados por el modelo *anfis* como salida. En la figura 4.4, puede notar que durante el instante de *training* existe un margen de error entre 75 y -100, luego disminuye entre -15 y 15 aproximadamente, debido a las perturbaciones, lo que indica que el modelo *anfis* se generó correctamente y se obtuvieron resultados suficientemente generales.

4.3. Modelado predictivo del flujo vehicular en la intersección Nicanor Merchán

Para validar el modelo en esta intersección, se procede de manera semejante que el caso anterior, pero esta vez los datos de entrada pertenecen a la Intersección Avenida Héroes de Verdeloma - Luis Cordero, mientras que los de salida corresponden a la intersección

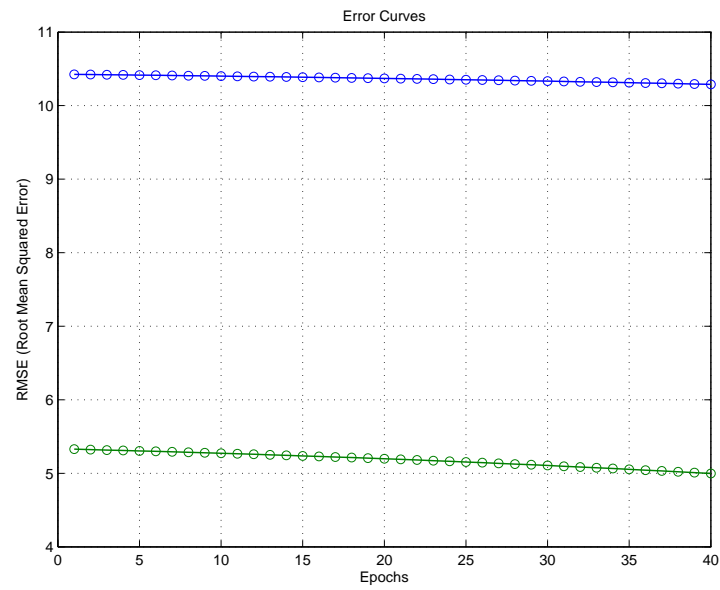


Figura 4.3: Errores de training y checking vs. Repeticiones de entrenamiento

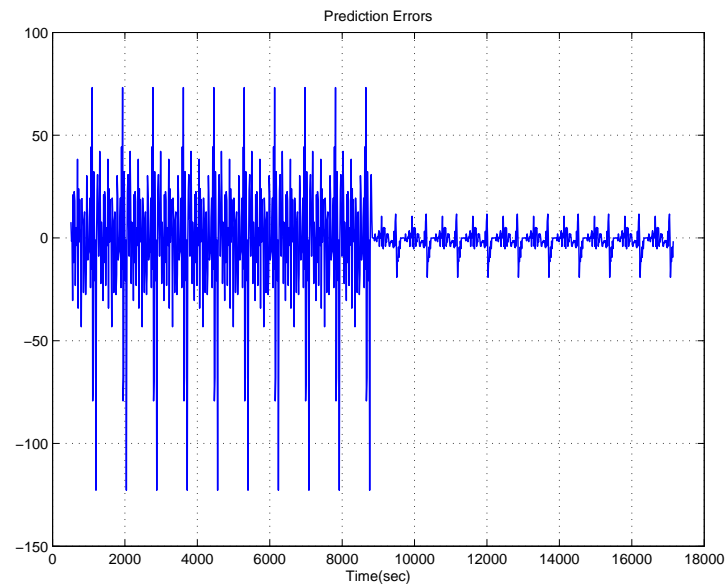


Figura 4.4: Errores de predicción

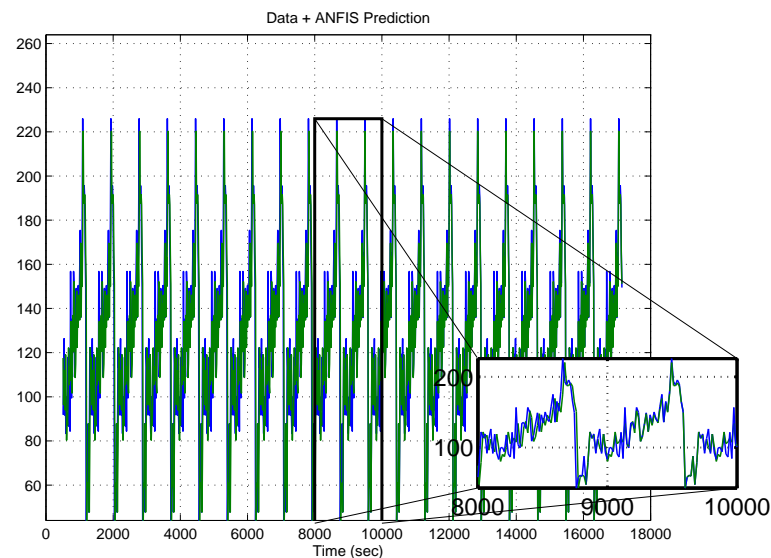


Figura 4.5: Datos de entrada y Modelo ANFIS vs. Tiempo

Avenida Héroes de Verdeloma - Nicanor Merchán, (véase Anexo A.3)

4.3.1. Procesamiento de datos

Se crean las matrices de *training* y *checking*, tomando en cuenta que las 4 primeras columnas representan los valores de entrada de la intersección Héroes de Verdeloma y Luis Cordero, mientras que la salida es generada por los valores de la siguiente Intersección semafórica Héroes de Verdeloma y Nicanor Merchán.

Luego se realiza el remuestreo de los datos, tal y como se explicó en el caso y capítulo anterior, añadiendo un ruido aleatorio con un rango de ± 2 y concatenando las muestras 20 veces. (véase Anexo B.4)

4.3.2. Construcción del modelo ANFIS

A continuación se procede con el mismo método, es decir, se parte de un FIS no adaptado y mediante las herramientas de MATLAB `genfis1` y `afis`, se obtiene un FIS adaptado.

En la figura 4.5 se puede observar el FIS obtenido. La gráfica en color azul muestra los datos de entrada y en color verde los datos del modelo del `anfis`, al igual que el caso anterior existe un error debido a las perturbaciones, esta vez, entre las intersecciones

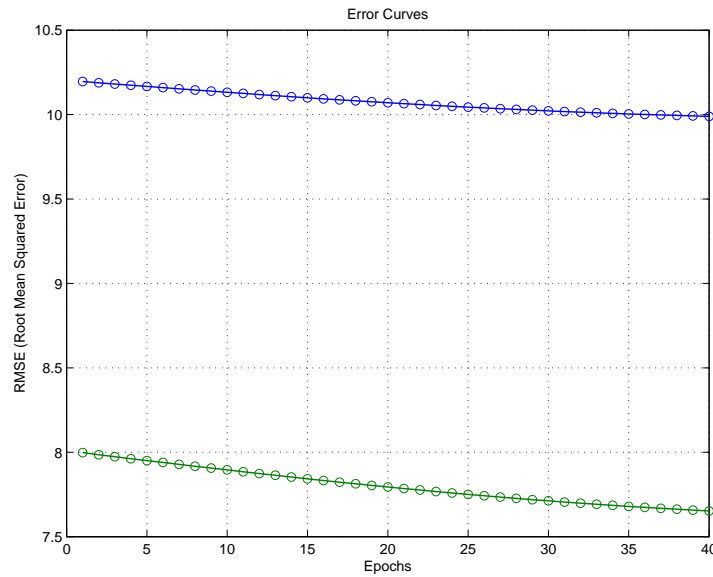


Figura 4.6: Errores de training y checking vs. Repeticiones de entrenamiento

“2” y “3” (véase sección 4.1). Se puede observar que a partir de la mitad, al iniciar el intervalo de *checking*, el error disminuye y el modelo se ajusta de mejor manera, pero aún se conserva la presencia del error, el cual se ilustrará posteriormente.

En el capítulo 3 las gráficas llegaban a ajustarse acercándose al valor 0, debido a que no se debían considerar las perturbaciones del sistema, pues se analizaron muestras de una misma intersección.

A continuación se obtienen los errores de *training* y *checking* para analizar la validación del modelo. En la figura 4.6, se puede observar en color azul el error de *training* cuyo valor está por debajo de 10.5, y de color verde el error de *checking* cuyo valor está por debajo de 8, esto se produce debido a las perturbaciones existentes, es decir, las calles que se encuentran entre las dos intersecciones, por donde pueden vincularse o desviarse los vehículos que circulan por la Avenida Héroes de Verdeloma, sin embargo los valores siguen siendo cercanos a 0.

Al final se encuentran las desviaciones existentes entre los datos de entrada y los datos de salida generados por el *anfis*

En la figura 4.7, se puede observar que existe un error entre 50 y -110 en el instante de *training*, luego el error disminuye entre -20 y 20 aproximadamente, debido a las perturbaciones, lo que indica que el modelo *anfis* obtenido es apropiado.

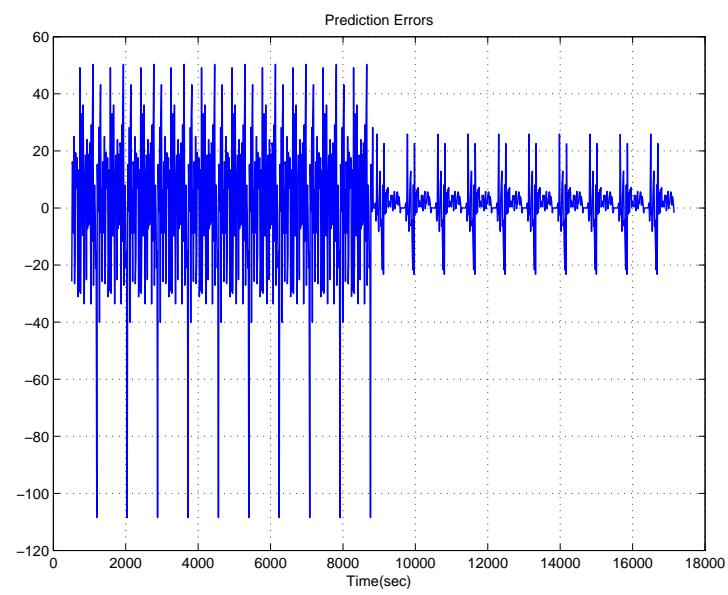


Figura 4.7: Errores de predicción

Conclusiones

Recorriendo la Avenida Héroes de Verdeloma de la ciudad de Cuenca y verificando su alto flujo vehicular, surgió la idea de realizar un proyecto de investigación que pueda modelar el comportamiento del mismo. Contando con el apoyo de funcionarios del Municipio de la ciudad, se buscaron datos estadísticos existentes de ciertas intersecciones. Teniendo en cuenta que en la Avenida Héroes de Verdeloma, existen intersecciones semafóricas consecutivas, se inició con el desarrollo de este proyecto. Investigando la disponibilidad de aforos vehiculares realizados sobre esta avenida, se pudo obtener los datos de 3 intersecciones semafóricas en las calles Mariano Cueva, Luis Cordero y Nicanor Merchán. Luego de tener las herramientas listas para el proyecto, se analizaron los modelos matemáticos existentes, y se pudo comprobar que para buscar el modelo de un sistema de tránsito vehicular es necesario determinar cuál es el enfoque y cuáles son los parámetros que intervienen en el mismo, como son: flujo, densidad y velocidad. Una vez analizados los conceptos y parámetros básicos, se investigó sobre una rama de la inteligencia artificial, que se aplica para el modelado y control de sistema de flujo vehicular, que es la lógica difusa.

Partiendo de los principios de la lógica difusa, se planteó buscar un modelo, que sea capaz de predecir dos muestras de datos en el futuro $x(t+2)$, para esto fue necesario procesar los datos conocidos adecuándolos para su uso. Mediante herramientas de MATLAB fue posible encontrar el modelo del sistema.

Luego se obtuvo el modelo predictivo de una intersección semafórica. Los datos obtenidos fueron: gráficas del modelo del sistema, errores de entrenamiento, errores de comprobación, y desviaciones existentes entre datos de entrada y datos del modelo obtenido.

Los resultados obtenidos fueron apropiados, se obtuvo un modelo que se ajustó de manera correcta a los datos obtenidos por el aforo vehicular. Este proyecto de investigación puede ser utilizado para obtener el comportamiento del flujo vehicular en avenidas conociendo los primeros indicios de datos estadísticos, la obtención del modelo del sistema es el paso previo al control, el modelo obtenido podrá ser usado en el futuro para predecir el tráfico vehicular existente sobre una vía y de esta manera poder desarrollar el control de diferentes dispositivos como por ejemplo: Semáforos inteligentes, Radares, etc.

El método aplicado, es independiente de la forma en la cual se obtiene la información del flujo vehicular. Para el caso presentado en este proyecto de investigación, los datos conocidos fueron obtenidos mediante observación directa, pero podría obtenerse a partir de sensores instalados a los lados de la vía, o mediante reconocimiento de patrones, en las imágenes captadas por las cámaras actualmente instaladas en las intersecciones. Es necesario también mencionar que este método podría implementarse en microcontroladores presentes en tarjetas de control de los semáforos, debido a que desde el punto de vista matemático consume pocos recursos.

Anexos

Anexo A

Aforo Vehicular - Av. Héroes de Verdeloma

A.1. Intersección Av. Héroes de Verdeloma - Mariano Cueva

HORA (SEC)	NRO. DE VEHÍCULOS
360	13
375	12
390	37
405	66
420	76
435	68
450	73
465	116
480	132
495	108
510	114
525	113
540	111
555	82
570	108
585	95
600	116
615	133
630	104
645	114
660	117
675	115
690	150
705	103

720	136
735	126
750	152
765	151
780	147
795	99
810	125
825	136
840	142
855	113
870	120
885	128
900	119
915	91
930	79
945	107
960	135
975	124
990	120
1005	113
1020	130
1035	153
1050	123
1065	171
1080	148
1095	212
1110	134
1125	84
1140	124
1155	139
1170	123
1185	133

A.2. Intersección Av. Héroes de Verdeloma - Luis Cordero

HORA (SEC)	NRO. DE VEHÍCULOS
360	40
375	56
390	58

405	60
420	82
435	45
450	80
465	120
480	112
495	120
510	95
525	112
540	129
555	91
570	119
585	96
600	85
615	102
630	104
645	93
660	118
675	89
690	88
705	122
720	85
735	159
750	101
765	102
780	135
795	138
810	130
825	156
840	150
855	110
870	120
885	135
900	148
915	125
930	145
945	135
960	151
975	140

990	134
1005	158
1020	175
1035	134
1050	165
1065	151
1080	180
1095	227
1110	195
1125	190
1140	196
1155	187
1170	168
1185	147

A.3. Intersección Av.Héroes de Verdeloma - Nicanor Merchán

HORA (SEC)	NRO. DE VEHÍCULOS
360	15
375	16
390	38
405	62
420	65
435	62
450	71
465	76
480	95
495	103
510	76
525	77
540	93
555	91
570	99
585	102
600	86
615	108
630	88
645	98
660	98

675	108
690	110
705	108
720	109
735	137
750	125
765	121
780	123
795	155
810	104
825	110
840	95
855	104
870	95
885	116
900	135
915	116
930	101
945	105
960	106
975	130
990	113
1005	134
1020	119
1035	135
1050	143
1065	138
1080	140
1095	167
1110	141
1125	181
1140	156
1155	159
1170	150
1185	167

Anexo B

Código de MATLAB para la Obtención del Modelo

B.1. Procesamiento de Datos

```
load datosMC.dat;  
a = datosMC;  
load timeMC.dat;  
time = timeMC;  
  
time2 = a(:,1);  
x_t = a(:,2);  
n = 20;  
j=1;  
  
for i=1:56  
noise(i,j) = random('Normal',a(i,2),2,1,1);  
end  
  
Signal=[];  
for i=1:n  
Signal = vertcat(noise,Signal);  
end
```

B.2. Remuestreo de Datos

```
trn_data = zeros(555,5);  
chk_data = zeros(555,5);  
  
% los datos seran usados desde
```

```
x(t-6) x(t-4), x(t-2),x(t);x(t+2)
```

```
% training data
```

```
trn_data(:, 1) = Signal(2:556);
trn_data(:, 2) = Signal(4:558);
trn_data(:, 3) = Signal(6:560);
trn_data(:, 4) = Signal(8:562);
trn_data(:, 5) = Signal(10:564);
```

```
% checking data
```

```
chk_data(:, 1) = Signal(558:1112);
chk_data(:, 2) = Signal(560:1114);
chk_data(:, 3) = Signal(562:1116);
chk_data(:, 4) = Signal(564:1118);
chk_data(:, 5) = Signal(566:1120);
```

B.3. Construcción del Modelo Anfis

```
% Build anfis model
```

```
epoch_n = 40;           % Repectaciones de entrenamiento
dispOpt = NaN;          % Opciones de Visualizacion
numMFs = 3;             % Numero de funciones de membresia
inmfType = 'trimf';     % Tipo de funciones de membresia
outmfType = 'linear';   % Tipo de salida
fismat = genfis1(trn_data,numMFs,inmfType,outmfType);
[trn_fismat, trn_error, stepsize, chkFis, chk_error] =
anfis(trn_data, fismat, epoch_n, dispOpt, chk_data, 1);
```

B.4. Procesamiento y Remuestreo de Datos

```
load datosMC.dat;
load datosLC.dat;
a = datosMC;
b = datosLC;
load timeMC.dat;
time = timeMC;
time2 = a(:,1);
```

```
x_t = a(:,2);
x_t2 = b(:,2);
```



```

n = 20; % numero de veces que se anade ruido
j=1;
for i=1:56
noise1(i,j) = random('Normal',a(i,2),2);
noise2(i,j) = random('Normal',b(i,2),2);
end

Signal=[];
Signal2=[];
for i=1:n
Signal = vertcat(noise1,Signal);
Signal2 = vertcat(noise2,Signal);
end

trn_data = zeros(555,5);
chk_data = zeros(555,5);

```

```

% los datos seran usados desde
x(t-6) x(t-4), x(t-2),x(t);x(t+2)
% training data
trn_data(:, 1) = Signal(2:556);
trn_data(:, 2) = Signal(4:558);
trn_data(:, 3) = Signal(6:560);
trn_data(:, 4) = Signal(8:562);
trn_data(:, 5) = Signal2(10:564);

% checking data
chk_data(:, 1) = Signal(558:1112);
chk_data(:, 2) = Signal(560:1114);
chk_data(:, 3) = Signal(562:1116);
chk_data(:, 4) = Signal(564:1118);
chk_data(:, 5) = Signal2(566:1120);

```

B.5. Cálculo del FIS

```

anfis_output = evalfis([trn_data(:,1:4);chk_data
(:,1:4)],chkFis);
index = 11:1120;
subplot(334), plot(time(index),Signal(index),'g');
hold on

```

```
grid on  
xlabel('Time (sec)');  
title('Data');
```

Bibliografía

- [1] CAL Y MAYOR R., RAFAEL *Ingeniería de Tránsito, Fundamentos y Aplicaciones*, Séptima Edición, Editorial Alfaomega, México, 1994.
- [2] NIETO CASTILLO LIZBETH, ROQUE MORALES JORGE, SÁNCHEZ PACHECO HUMBERTO y VALDÉS HERNÁNDEZ GUSTAVO DANIEL, *Sistema Móvil de Información Vial – SMIV*, México D.F., México, 2012.
- [3] J. DELGADO, P. SAAVEDRA y R. M. VELASCO, *Modelación de problemas de flujo vehicular*, Iztapalapa, México, 2011.
- [4] RUI NEVES-SILVA y MARIA MARQUES, *Traffic flow-density models based on systems theory*, New York City, USA, 2007.
- [5] HUGO BENAICH, *Exploring traffic and congestion Policies*, Cambridge, 2015.
- [6] NATHAN G., *Traffic Flow Theory*, University of Massachusetts-Lowell, 1975
- [7] GREENSHIELDS B.D., *A study of highway capacity*, *Proceedings Highway Research Record*, Washington Volume 14, 1935.
- [8] LIDTHILL M. J., WITHHAM G. B., *A theory of traffic flow on long crowded roads*, Manchester, 1954
- [9] TOM V. MATHEW, K. V. KRISHNA RAO, *Introduction to Transportation Engineering*, NEPTel, 2007
- [10] JANG SUN MIZUTANI *Neuro-Fuzzy and Soft Computing*, Prentice Hall, pp 335 - 368, ISBN 0-13-261066-3, 1997
- [11] MANDAMI E.H., ASSILIAN S. *An experiment in linguistic synthesis with a fuzzy logic controller*, *International Journal of Man-Machine Studies*, Vol. 7, No. 1, pp. 1-13, 1975
- [12] SUGENO M. *Industrial applications of fuzzy control*, Elsevier Science Pub, Co., 1985
- [13] BR. CARLOS JOSÉ D. PÉREZ R., *Desarrollo de modelos de tráfico vehicular en Galatea*, Mérida, Venezuela, 2010.
- [14] JAIRO ESPINOZA., JOOS VANDEWALLE, VINCENT WERTZ *Fuzzy Logic, Identification and Predictive Control*, Springer, New York, 2004

UNIVERSIDAD POLITÉCNICA SALESIANA
UNIDAD DE POSGRADOS

**MAESTRÍA EN CONTROL Y AUTOMATIZACIÓN
INDUSTRIALES**

Autor:
Ing. Ismael Campoverde Borja

Dirigido por:
Dr. Julio César Viola

**MODELADO DE TRÁFICO VEHICULAR
EN AVENIDAS
A PARTIR DE DATOS ESTADÍSTICOS**

Actualmente la cantidad de vehículos existentes en las zonas urbanas de las ciudades ha incrementado, lo que produce largas filas de vehículos en las vías, que son altamente dependientes de la sincronización de encendido-apagado de los semáforos existentes en las intersecciones. En la zona céntrica de la ciudad de Cuenca, se encuentra implementado un Sistema de Control de Tráfico Urbano Adaptativo (SCATS), mientras que en las partes exteriores los semáforos se encuentran funcionando con tiempos fijos previamente establecidos.

En la Avenida Héroes de Verdeloma los semáforos se encuentran funcionando bajo el método de tiempos fijos, los cuales son establecidos en base a la toma de datos estadísticos. Para determinar los tiempos fijos en las intersecciones o implementar algún sistema inteligente de semáforos, es necesario modelar el tráfico existente, para luego realizar cualquier tipo de control. El modelo del sistema no existe actualmente y su desarrollo es propuesto como tema central de este trabajo de investigación.

En la Ciudad de Cuenca existen varias avenidas importantes que carecen del control de flujo vehicular, por esta razón, se ha visto necesario hallar el modelado del tráfico existente en una avenida de alta circulación de vehículos, para luego usar el mismo principio en diferentes zonas de la ciudad.