



BENEMÉRITA
UNIVERSIDAD AUTÓNOMA DE PUEBLA
Facultad de ciencias de la computación
Teresa Martínez Flores



BENEMÉRITA
UNIVERSIDAD AUTÓNOMA DE PUEBLA
Facultad de ciencias de la computación

SIMULACIÓN ESTADÍSTICA DE EVENTOS
DISCRETOS EN TRÁFICO VEHICULAR

Tesis que presenta

Teresa Martínez Flores

Para obtener el grado de
Ingeniero en ciencias de la computación

Asesor Dr. Alejandro Rangel Huerta
Asesor C. Dr. Fernando Rojas Rodríguez



BENEMÉRITA
UNIVERSIDAD AUTÓNOMA DE PUEBLA
Facultad de ciencias de la computación
Teresa Martínez Flores

**Puebla, Puebla
Diciembre de 2008**

Resumen

La congestión vehicular es provocada por muchos factores entre ellos, el rápido aumento poblacional, lo cual generan una mayor demanda de transporte, la deficiencia en la construcción de infraestructura vial y la falta de educación vial en los conductores. Todos estos factores, causan que día a día exista una mayor congestión vehicular, demoras, accidentes y problemas ambientales.

El fenómeno de transporte vehicular es complejo, pero a través de modelos de simulación obtenemos una mejor comprensión del fenómeno, además nos ayuda a analizar el comportamiento del tráfico vehicular, para mejorar su administración y operación. Y así disminuir la congestión vehicular.

Este trabajo tiene como objetivo el desarrollo de una herramienta de simulación para describir algunas de las principales características del fenómeno de tráfico vehicular, tomando como base de datos de entrada, los provenientes del trabajo de campo, en el que se realiza en una serie de observaciones del flujo vehicular, de una calle principal del sistema vial urbano de Puebla. Dado que los modelos analíticos son muy complejos, realizaremos nuestra simulación basada en modelos de simulación estadística, orientados a procesos de eventos discretos. La interacción entre vehículos será modelada a través del modelo Car-Following. El ambiente de desarrollo utilizado es NetLogo.



Capítulo 1

INTRODUCCIÓN

1.1 Antecedentes

La mayoría de la actividad desarrollada en las áreas urbanas está asociada al desplazamiento de personas y mercancías entre diferentes zonas de las ciudades, haciendo uso diario de las infraestructuras disponibles. Un sistema de transporte eficiente es esencial para el desarrollo económico de la sociedad y la calidad de vida en las regiones urbanas y su entorno. El incremento en el parque vehicular de las ciudades, la mala planeación de las vías carreteras, nos ha llevado a enfrentarnos a un grave problema, la congestión vehicular, lo cual afecta los tiempos de traslado y distribución, provocando grandes pérdidas monetarias para el sector económico además del daño provocado al medio ambiente.

Esto ha llevado a la sociedad a la búsqueda de la optimización de las vialidades, para lograr que el tráfico vehicular fluya de una mejor manera, evitando la congestión vehicular y reduciendo los tiempos de traslado. [1]

El sistema de tráfico vehicular se describe como un sistema complejo difícil de modelar, debido a que interviene el factor humano, los conductores que responden a las condiciones del tráfico en torno a ellos, de diversas maneras ante la misma situación, un ejemplo claro en el tráfico vehicular, es la reacción que se tiene cuando el auto delantero frena, uno puede optar por rebasar o detenerse y esperar que avance. La mayoría de las veces los congestionamientos viales son provocados por los malos hábitos de manejo de los conductores, pocas veces por accidentes reparaciones de las vías carreteras mal funcionamiento del semáforo etc.

Los conductores también responden a la densidad de tráfico en torno a ellos, aceleran cuando hay luz verde y desaceleran (quizás involuntariamente) en el tráfico pesado o cuando la luz cambia a color ámbar. [2]

Existen diversos modelos de tráfico, para la implementación utilizaremos un modelo con enfoque microscópico como lo es Car-Following. Se realiza trabajo de campo para la recolección de muestras de flujo vehicular de entrada y salida en un tramo de carretera del Boulevard 5 de mayo, tiempos de semaforización. Con los datos obtenidos se crean tablas de distribución de frecuencias y se busca un modelo estadístico que se ajuste a esta distribución. El modelo resultante será utilizado para alimentar la entrada de flujo vehicular a la simulación.



1.2 Estado del arte modelación de tráfico vehicular

El estudio sobre el flujo vehicular se remota a 1930 con la aplicación de la probabilidad para la descripción del flujo vehicular, más tarde Greenshield (1935) inicia con sus estudios sobre flujo de tráfico modelando el comportamiento del flujo vehicular de una línea de tráfico, proponiendo una relación lineal entre velocidad y densidad.

El aumento en el número de vehículos hizo que en la década de los 50 imperara el interés por comprender el fenómeno de flujo vehicular, y resolver los problemas del tráfico, surgen modelos como Car-Following, teoría de ondas de tráfico y teoría de colas para la descripción del flujo vehicular. Se desarrollan trabajos muy importantes como los de Reuschel (1950) Wardrop (1952), Pipes (1953), Lighthill y Whitman (1955), Newell (1955), Richards (1956), Webster (1957), Eddie y Foote (1958). [3]

Reuschel y Pipes proponen un modelo de tráfico que describe el movimiento detallado entre los automóviles en un solo carril manteniendo lo más juntos posible, es un modelo de tráfico microscópico. Lighthill y Whitman proponen un modelo macroscópico, modelando el tráfico como un flujo continuo. Publican una de las primeras teorías de modelado macroscópico de flujo de tráfico en carreteras. Esta teoría estaba basada en la ecuación de continuidad y la relación fundamental entre el flujo y la densidad del tráfico. El tráfico vehicular cuando es visto desde una gran distancia, por ejemplo, desde un avión, el tránsito pesado aparece como el torrente de un fluido. Por lo tanto, el flujo de tránsito de autos se puede desarrollar en analogía con la teoría hidrodinámica de fluidos tratando al tránsito como un fluido uni-dimensional de izquierda a derecha. Richard propone un modelo continuo parecido, la diferencia es que Richard se enfocó en la derivación de las ondas de choque con respecto a la densidad.

Hasta los años 80 los investigadores habían prestado atención al estudio de los modelos de aceleración en régimen de seguimiento de vehículos, fue entonces cuando surge los modelos de aceleración libre, son empleados para aquellos casos donde el conductor no está influido por un vehículo precedente como por ejemplo cuando actúan en función de semáforos, obstáculos o bien se mueve con completa libertad.

En 1992 los científicos Kai Nagel y Michael Schreckenberg crea un modelo de flujo de tránsito vehicular con un autómata celulares, donde cada *célula* del autómata equivale ya sea a un vehículo en movimiento con cierta velocidad o a un espacio vacío de la avenida donde se encuentran los vehículos.[4]



Durante todo este tiempo se han formulado teorías y creado modelos que son utilizados como herramientas para el análisis del comportamiento del tráfico vehicular, para mejorar su administración y operación y el avance de la computación ha permitido estudiar el fenómeno de tráfico vehicular de una forma más detallada, permitiendo realizar simulaciones basadas en los modelos de tráfico, experimentar con ellas, aunque algunos modelos generan una gran carga computacional.

Algunos de estos modelos fueron desarrollados pensando en la realización de las simulaciones a través de computadoras. Hay varios proyectos que han implementado estos modelos uno de estos es el proyecto AISUM que es un software comercial para la simulación de tráfico. Este software utiliza como modelo para sus simulaciones al “modelo de Gipps de vehículo siguiente” (Peter Gipps) publicado en 1981 para realizar simulaciones digitales.

Hay varios proyectos de tipo independiente como el del Dr. Martín Treiber, del Instituto para el Transporte y Economía de la Universidad Técnica de Dresden, se ha estado ocupado desarrollando modelos matemáticos que describen el flujo del tráfico vehicular y, especialmente, las inestabilidades en este bajo algunas condiciones especiales. Dr. Treiber desarrolló un simulador que demuestra los modelos de una manera

El simulador le permite observar como ocurren embotellamientos sin causa aparente, el efecto de una incorporación de tráfico a una vía rápida, el bloqueo de un carril y más. Usted puede alterar los parámetros y observar el efecto del cambio.



Capítulo 2

MODELOS DE TRÁFICO VEHICULAR

Un modelo nos sirve para representar un fenómeno, en función de un propósito, existen diferentes modelos que describen el tráfico vehicular tomando algunas de sus características y nos dan aproximaciones matemáticas con cierto grado de apego a la realidad. Estos modelos son llevados a la simulación, elaborando un programa de cómputo que se comporte como el fenómeno físico, y así analizar y probar el modelo.

2.1 Clasificación de modelos de tráfico vehicular

Debido a la gran variedad de modelos de tráfico vehicular existentes es necesario clasificarlos, existen varios criterios por los cuales podemos de clasificar un modelo de tráfico vehicular.

Los modelos son clasificados según los siguientes criterios [5]:

1. Escala de las variables independientes (continuo discreto o semi-discreto)
2. Nivel de detalle(microscópico mesoscópico y macroscópico)
3. Representación de los procesos(determinístico, estocástico)
4. Funcionamiento(analítica o simulación)
5. Escala de aplicación

1.- De acuerdo al nivel de detalle con se describen las entidades en los modelos de tráfico vehicular pueden ser clasificados en:

- Modelo macroscópico:

Estudia el comportamiento de los vehículos a gran escala, estos tipos de modelos describen el tráfico como un flujo de automóviles, sin tomar en cuenta las partes que lo componen que en este caso serían los automóviles de forma individual, sólo es visto en conjunto, son modelos continuos que hace uso de ecuaciones diferenciales, se enfocan en captar las relaciones globales de flujo de tráfico

Los tres parámetros macroscópicos principales que describen el flujo vehicular son:

- Velocidad vehículos
- Tasa Flujo vehicular
- Densidad de tráfico



- Modelo microscópico:

Describe el comportamiento del flujo de tráfico vehicular con gran detalle a través de describir el comportamiento de conductores y los automóviles vistos como entidades discretas individuales y atómicas que interactúan unas con otras, así como sus interacciones a un nivel alto de detalle

Son modelos por lo general discretos entre estos modelos se incluyen: El modelos de carro siguiente (Car-Following) y modelos de autómatas celulares

- Modelo mesoscópicos

En los modelos mesoscópicos se estudian los autos por grupos, se clasifican de acuerdo a sus velocidades y se obtienen tipos de autos que se pueden estudiar de forma individual, las actividades e interacciones se describe a un nivel de detalle bajo

2.- Según al modo en que evolucionan sus variables en las escalas de tiempo pueden ser continuos o discretos.

- Continuo si las variables cambian de modo continuo en función del tiempo
- Discretos o de eventos discretos si las variables varían en un conjunto contable de instantes de tiempo.

3.-La forma de representación del proceso

- Modelos determinísticos la representación no contiene variables aleatorias y todas sus las variables se definen mediante relaciones exactas.
- Los modelos estocásticos contienen al menos una variable no determinista

4.-Por el área de aplicación

El modelo puede describir la dinámica de los vehículos en un sólo carril con un sentido de flujo, con intersecciones en una red de carretera etc.

5.-De acuerdo al uso

Si es usado para una simulación o como solución analítica de un grupo de ecuaciones



2.2 Modelos de enfoque macroscópico

Los modelos macroscópicos se derivan de la analogía entre el flujo vehicular y un flujo continuo (fluido o gases) considera el tráfico vehicular como un flujo de vehículos y desarrolla algoritmos que relacionan el flujo, con la densidad y la velocidad.

Flujo vehicular

La teoría de flujo del tránsito consiste en el desarrollo de las relaciones matemáticas entre los elementos primarios del flujo vehicular tales como flujo, densidad y velocidad.

Mediante el análisis de flujo vehicular se puede entender las características y el comportamiento del tránsito y se describe la forma como circulan los vehículos en cualquier sistema vial [6]. La teoría de flujo vehicular es aplicada en la simulación en la cual se utilizan complejos algoritmos matemáticos, para estudiar la interrelación que existe entre los elementos de un flujo vehicular y para estimar el efecto de los cambios en el flujo de tránsito.

Las tres características fundamentales son representadas por las variables principales velocidad, densidad y flujo, que representan el flujo vehicular. A continuación explicaremos éstas variables.

Flujo

Se define el flujo como el número de vehículos que pasa por un punto o sección transversal dada, de un carril o una calzada, durante un periodo determinado, se expresa como:

$$q = \frac{N}{T} \quad (2.1)$$

q = número de vehículos que pasan por unidad de tiempo (vehículos por periodo)

N = número total de vehículos que pasan (vehículos)

T = periodo determinado (unidad de tiempo)

A nosotros nos interesa calcular la tasa de flujo que es el número de vehículos que pasa durante un periodo inferior a una hora.

En este caso $T < 1$



Velocidad (V)

La velocidad del tráfico es la distancia recorrida por un vehículo durante una unidad de tiempo regularmente se expresa en kilómetros por hora (km/h)

La densidad o concentración (k)

Es el número de vehículos que ocupa una longitud específica, de una vialidad en un momento dado. La figura 2.1 muestra N número de autos en una distancia d.

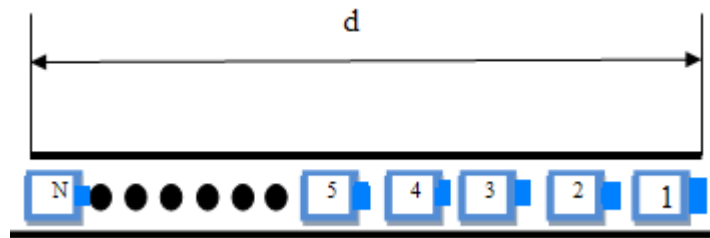


Figura 2.1 Densidad o concentración

Según la figura 2.1 la densidad se calcula como:

$$k = \frac{N}{d} \quad (2.2)$$

Estas tres variables están relacionadas mediante la ecuación fundamental del flujo vehicular, su fórmula general es:

$$q = v k \quad (2.3)$$

La mayoría de los modelos macroscópicos relacionan parejas de las tres variables para describir las características generales del tráfico vehicular.

Velocidad media

Que corresponde a la media aritmética de las velocidades observadas en algún punto designado de la carretera se expresa como:

$$\bar{u} = \frac{1}{n} \sum_{i=1}^n u_i \quad (2.4)$$



Donde:

u_i Es la velocidad del i ésimo vehículo.

n Número de vehículos que transitan por la carretera

Espaciamiento simple

Es la distancia entre dos vehículos consecutivos, expresado en metros

Espaciamiento promedio

El espaciamento promedio de todos los espaciamentos simples observados entre los vehículos que ocupan una vía, el espaciamento promedio esta dado por la ecuación (2.5)

$$S = \frac{\sum_1^N S_i}{N} = \frac{d}{N} = \frac{1}{k} \quad (2.5)$$

Donde:

S_i es el espaciamento simple entre automóviles

N es el número total de automóviles

Al simplificar la ecuación obtenemos que el espaciamento promedio es igual a 1 entre la densidad ($1/k$)

Existen varios modelos que tratan de modelar el flujo vehicular, estos modelos describen algunas características del flujo vehicular, los modelos macroscópicos hacen una descripción más general sobre el flujo vehicular viendo en tráfico vehicular como un fluido de partícula a través de un tubo que en este caso sería el carril en el que transitan los automóviles y cada partícula sería un auto. Algunos de los modelos macroscópicos más conocidos son los siguientes:

Modelo de Greenshields

Greenshields desarrolló uno de los primeros modelos que se conocen, en este se estudia la relación entre la velocidad y densidad, también entre la velocidad y el flujo. Postuló que existe una relación lineal decreciente entre la velocidad y la densidad.



La siguiente figura 2.2 nos muestra la curva generada de la velocidad en contra la densidad.

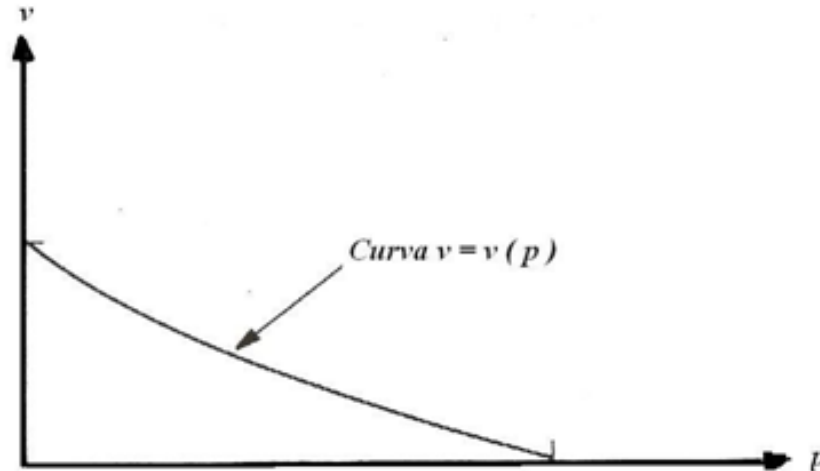


Figura 2.2 Grafica velocidad/densidad

En la figura podemos observar como el automóvil mantiene una velocidad máxima cuando la densidad es mínima, y mientras más aumenta la densidad se va disminuyendo la velocidad.

Hecho que puede ser confirmado intuitivamente, cuando pensamos que un grupo de vehículos que viaja a mayor velocidad, viaja más disperso, los vehículos conservan mayor distancia entre ellos y cuando la velocidad de estos vehículos disminuye, los vehículos se van juntando más, por que el avance es más lento. Por lo tanto a mayor velocidad menor densidad, y viceversa a menor velocidad mayor densidad.

La ecuación fundamental del modelo es:

$$V_e = V_l - \left(\frac{V_l}{k_c}\right) K \quad (2.6)$$

Donde

V_e : Es la velocidad media espacial (km/h)

k : Es la densidad (veh/km/carril)

V_l : Velocidad media espacial a flujo libre (km/h)

k_c : Densidad de congestionamiento según la capacidad de la vía (veh/km/carril).



Modelo logarítmico de H. Greenbergs

Varios modelos macroscópicos utilizan la analogía del flujo de fluidos para desarrollar relaciones macroscópicas, para el flujo de tránsito. Greenbergs desarrollo un modelo en el que se emplea esta analogía. De la combinación y discretización de las ecuaciones de movimiento y continuidad de fluidos, aplicadas al flujo vehicular, se obtienen las principales ecuaciones de su modelo.

La ecuación de este modelo esta dada como:

$$Ve = Vm \ln \frac{(Kc)}{K} \quad (2.7)$$

En la ecuación anterior se conserva la notación del modelo de Greenshields, además:

V_m : Es la velocidad del flujo máximo sobre la capacidad de la vía.

Modelo exponencial de R. T. Underwood

La formulación de este modelo buscaba representar más fielmente el comportamiento del flujo libre. Es más usado para modelar autopistas y avenidas ó carreteras urbanas, donde se presenta más éste fenómeno. Tiene mucha aceptación en condiciones de flujos no congestionados. Siguiendo con la notación de los modelos anteriores, sus ecuaciones Fundamentales son:

$$Ve = V_l e^{-\left(\frac{k}{k_m}\right)} \quad (2.8)$$

$$q = V_l k e^{-\left(\frac{k}{k_m}\right)} \quad (2.9)$$

Teoría cinética de gases

En la teoría cinética el tránsito se ve como un gas, donde cada partícula que lo compone interactúa una con otra y cada partícula representa un automóvil. Existen diferentes modelos como el de Prigogine y el de Paveri-Fontana para flujo de tránsito que han sido desarrollados tomando como base la teoría cinética de gases.



2.3 Modelos de enfoque microscópico

El enfoque microscópico considera los espaciamientos entre los vehículos individuales, así como sus velocidades, los modelos más conocidos con este enfoque son los siguientes:

Modelo de Nagel-Schreckenberger

Es un modelo basado en autómatas celulares (CA). Los modelos de autómatas celulares describen el sistema de tráfico como una malla de células de igual tamaño. Un CA-modelo describe de una manera discreta los movimientos de vehículos de célula a célula, un autómata celular origina el movimiento de los carros

El tamaño de las células es escogido de tal forma que un vehículo con velocidad igual a uno se puede mover a la próxima célula durante un paso de tiempo discreto. El vehículo puede asumir una velocidad que va de un rango de valores discretos de cero a velocidad máxima.

Existen procesos de:

- Aceleración
- Desaceleración
- proceso estocástico de desaceleración.

Modelo “Car-Following” (sigue al líder)

Fue desarrollado buscando estimar el comportamiento de dos vehículos en un flujo de tráfico. Es decir, estimar la respuesta del vehículo con respecto al comportamiento del vehículo que viaja adelante (vehículo predecesor).

Describen la aceleración de coches “seguidores” como funciones de la distancia entre el líder y el seguidor, la velocidad relativa de los dos coches, y el tiempo de la reacción del conductor del coche seguidor.

La figura 2.3 nos muestra un grupo de autos con distancias entre ellos $d_1, d_2, d_3, \dots, d_n$ su ubicación $x_1, x_2, x_3, \dots, x_n$ en el modelo de el carro siguiente son utilizadas estas mediciones además de la velocidad que lleva cada automóvil.

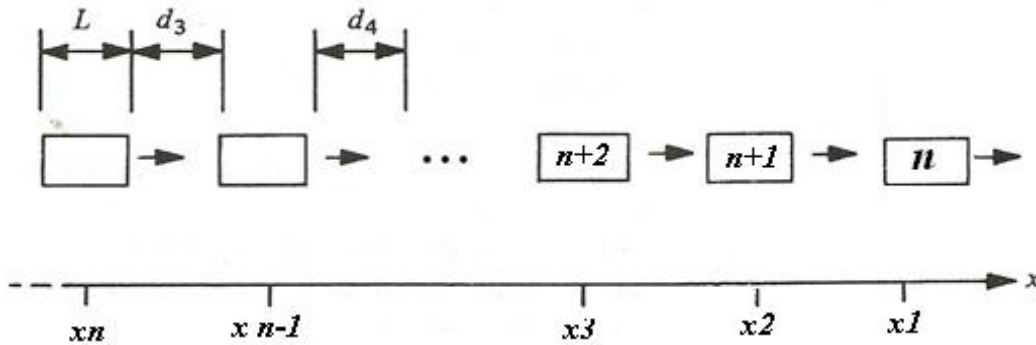


Figura 2.3 Componentes del modelo Car-Following

La ecuación básica de los modelos Car-Following para una línea de tráfico está dada en la ecuación (2.10)

$$\text{Respuesta} = \text{Sensibilidad} * \text{estimulo} \quad (2.10)$$

Esta ecuación constituye una ecuación fundamental para el diseño, y representa la ocurrencia de un hecho como estímulo que percibe un conductor y que lo anima a actuar, ya que dependiendo del grado de sensibilidad o advertencia del conductor en ese instante, se produce una reacción un instante después.

La respuesta está siempre representada por la aceleración o desaceleración del vehículo siguiente y el estímulo puede ser la distancia entre el vehículo siguiente y el vehículo de en frente, la velocidad relativa entre los dos carros o la aceleración relativa. La respuesta dependerá de la sensibilidad del conductor a un estímulo dado. La respuesta se tomará como la aceleración o desaceleración de los vehículos siguientes, denotado aquí por $d^2 x_f / dt^2$, mientras que el estímulo se tomará como la diferencia entre la velocidad del vehículo precedente dx_l / dt y la velocidad del vehículo que lo siguiente dx_f / dt

La diferencia entre los cinco modelos está en como se representa la sensibilidad

El Primer modelo

Es Desarrollado por Clander R.2, Herman E. y Montroll E. Utiliza un valor constante de sensibilidad definida como λ . Con $\lambda > 0$ si el coche siguiente va más rápido que el coche de en frente, el coche desacelera para evitar golpear el coche.



Cuando λ es una constante que podemos integrar fácilmente, obtenemos la ecuación lineal del Car-Following (2.11), un resultado que puede verificarse por diferenciación: [2]

$$\frac{dx_{n+1}(t+T)}{dt} = -\lambda[x_{n+1}(t) - x_n(t)] + D_F \quad (2.11)$$

T es el tiempo de respuesta del conductor ante la situación

D_F es una constante arbitraria de de integración con dimensión de velocidad

En este resultado se relaciona la velocidad del coche siguiente y el espacio entre los coches.

Segundo Modelo

Ahora la sensibilidad tiene uno de dos estados, de acuerdo a la distancia entre los vehículos. Para una corta distancia entre vehículos, λ_1 será usado como sensibilidad. Para una distancia larga entre vehículos, se utilizara una sensibilidad mayor λ_2 , con $\lambda_1 > \lambda_2$. Esto significa que los conductores son más sensibles a distancias cortas con respecto al vehículo siguiente que a mayores distancias.

Los vehículos avanzan más rápido cuando los automóviles están a una distancia amplia, y más lento cuando la distancia es más corta.

$$\frac{dx_{n+1}(t + T)}{dt} = \lambda_i[x_{n+1}(t) - x_n(t)] \quad i = 1 \text{ o } 2$$

Tercer Modelo

Toma en cuenta la distancia de separación entre los dos vehículos. Para la sensibilidad se tiene un valor constante α_0 y la distancia de separación entre los vehículos es incluida para reflejar cómo el valor de la sensibilidad se incrementa cuando los vehículos están cerca y cómo disminuye cuando la distancia de separación se empieza a hacer cada vez más grande.



La ecuación (2.12) nos muestra la ecuación de este modelo.

$$X_{n+1}(t + \Delta t) = \frac{\alpha_0}{x_n(t) - x_{n+1}(t)} [x_n(t) - x_{n+1}(t)] \quad (2.12)$$

Cuarto Modelo

Incorpora la velocidad del vehículo siguiente. Si la velocidad del tráfico aumenta, el conductor del vehículo siguiente va a estar más sensible a la velocidad relativa entre él y el vehículo delantero.

La ecuación de este modelo es la siguiente:

$$X_{n+1}(t + \Delta t) = \frac{\alpha x_{n+1}(t)}{x_n(t) - x_{n+1}(t)} [x_n(t) - x_{n+1}(t)] \quad (2.13)$$

Quinto Modelo

Este modelo puede representar el comportamiento del flujo del tráfico asumiendo que todos los conductores tienen el mismo comportamiento y que todos los vehículos tienen las mismas características, dado que estos parámetros no son considerados en el modelo.



Capítulo 3

Trabajo de campo

La presente tesis simula algunas características del tráfico vehicular, tomando como base de datos de entrada, los provenientes del trabajo de campo, en el que se realiza en una serie de observaciones de una calle principal del sistema vial urbano de Puebla, tal como el boulevard 5 de mayo, estudiaremos el tramo que va de la avenida Palafox y Mendoza a la 2 oriente en sentido sur-norte (Figura 3.1).



Figura 3.1 Boulevard 5 de Mayo de calle Palafox y Mendoza a 2 oriente

Se extraen muestra que representan el flujo vehicular, en condiciones operativas, en este tramo del boulevard, a diversas horas del día, en horas de tráfico normal y por separado en horas consideradas como horas pico.

Se tiene como objetivo del trabajo de campo, obtener algunas características básicas que influyen en el flujo del tráfico vehicular entre estas características se encuentran las siguientes:

- Tamaño de la carretera
- Número de autos que entran
- Número de autos que salen
- Tamaño promedio de la cola
- Espaciamiento promedio entre automóviles
- Tiempo de semáforo en verde y rojo
- Tiempo de reacción de los automóviles



3.1 Proceso de recolección de información

Empezaremos el trabajo de campo con la recolección de información la cual esta dividida en dos partes:

- En la primera parte extraemos los datos que se obtienen de manera directa y que no es posible obtenerlos del video, como lo son el largo de la carretera y la distancia entre automóviles.
- En la segunda parte se grabaran los videos.

Los datos arrojados en la primera parte de la recolección de datos son los siguientes:

El tramo de carretera mide 68 metros.

El semáforo en verde dura 56 segundos y en rojo 60 segundos.

El espacio promedio entre autos varía de 1.5 a 3.5 metros.

El tiempo de reacción promedio de los automóviles es de 2 segundos

Para poder realizar la segunda parte del proceso de recolección este estudio, se busca un lugar estratégico que permita observar el tráfico desde lo alto y tener una buena perspectiva que ayude a realizar las mediciones correspondientes.

Se eligió el puente peatonal que conecta el centro de convenciones con el barrio del artista, se realizan grabaciones desde diferentes ángulos, con el fin de lograr un enfoque que facilite el conteo de los automóviles.

Se observa que la calle que tiene un mejor punto de observación y permite contar la densidad numérica por unidad de tiempo es la calle que va de avenida Palafox y Mendoza a 2 oriente.

Se grabó con una videocámara Sony con Zoom de largo alcance, diferentes días de la semana, cabe mencionar que no es factible recolectar datos continuos todo el día, debido al costo involucrado que se tendría.

Así que se realizaron grabaciones periódicas durante 10 minutos, los periodos estuvieron localizados entre las 9:00 a.m. y las 5:00 p.m. durante este lapso de tiempo encontramos muestras de bajo, medio y alto flujo vehicular. Las grabaciones se suspendieron el fin de semana debido a las grandes variaciones en el tráfico.



2.1 Análisis de la información, tablas de frecuencia e histogramas

Flujo de entrada

Se analiza la información para conocer los volúmenes de tránsito de entrada al tramo de carretera (Figura 3.2), los resultados se usarán para alimentar nuestro sistema. Para obtener el flujo de volumen de tránsito, primero se define un periodo de tiempo de conteo, tomando como intervalo un minuto, se cuenta manualmente el número de autos que entran, se almacenan los datos obtenidos en una tabla de Excel se genera una gráfica del flujo de entrada de automóviles, para poder realizar un análisis del flujo más claramente.



Figura 3.2 Flujo de entrada

Al observar la gráfica se nota que el tiempo definido para realizar el conteo no era el apropiado, la gráfica que nos mostraba, no ayudaba a una clara interpretación del flujo de entrada al mostrar grandes variaciones en los datos.

Debido a este hecho se eligió tomar como tiempo de conteo el tiempo que tarda el semáforo, esto es el tiempo que tarda en cambiar un semáforo de verde a rojo y de rojo a verde. Se incluye el tiempo de semáforo amarillo en los tiempos de verde debido a que los conductores toman el tiempo del amarillo como verde.

Durante el día hay diversos cambios en los tiempos del semáforo, por lo que se tomó un promedio de estos tiempos, como el tiempo que tarda en cambiar el semáforo.



La gráfica generada por los datos de entrada de automóviles es la que muestra la figura 3.3

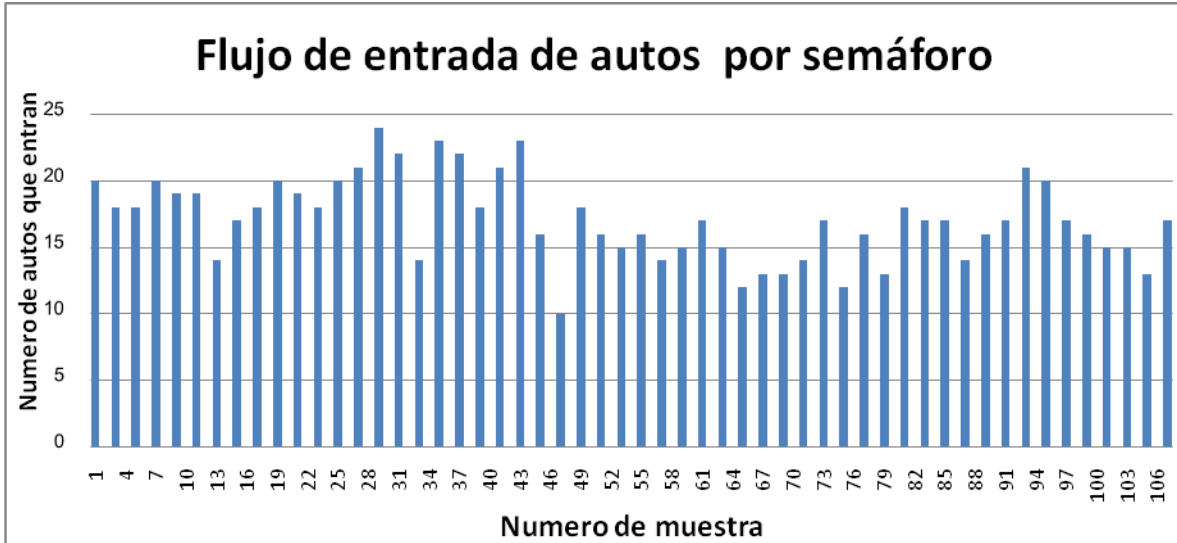


Figura 3.3 Tasa de flujo de entrada autos por semáforo

Se nota claramente que la afluencia de automóviles va de mayor a menor esto es debido a que cuando el semáforo anterior a esta calle esta en verde entran automóviles y cuando el semáforo esta en rojo no. La afluencia máxima de automóviles la encontramos entre las muestras 28 a 42 que representan el horario de 1:30 a 2:00 de la tarde.

El siguiente paso es encontrar la distribución de probabilidad que mejor represente los datos de entrada, para esto construiremos una tabla de frecuencias basado en los datos recolectados y se graficarán.

Los parámetros necesarios para determinar la tabla de frecuencia son los siguientes:

- El tamaño de la muestra
- El valor mínimo
- Valor máximo
- La frecuencia absoluta en cada intervalo, contando cuantos datos están en este intervalo

Se calcula la distribución de frecuencia utilizando una hoja de cálculo de Excel para la realización de las operaciones.



La Tabla 3.1 muestra las frecuencias absolutas observadas y las frecuencias teóricas de arribo de automóviles por semáforo.

Tabla de frecuencia de entrada de automóviles		
Número de Automóviles	Frecuencia Observada	Frecuencia Teórica
10	1	0.53
11	0	1.03
12	2	1.80
13	4	2.84
14	5	4.07
15	5	5.29
16	6	6.23
17	7	6.64
18	7	6.41
19	3	5.61
20	5	4.45
21	3	3.19
22	2	2.08
23	2	1.23
24	1	0.65

Tabla 3.1

Tabla de frecuencia de entrada

Determinados estos parámetros se traza el histograma de frecuencias de entrada de automóviles a la calle, el histograma surge de graficar la columna 1, como eje de las abscisas representando el número de automóviles observados que entran, y el eje de las ordenadas esta representado por la frecuencia observada para cada número de automóviles.

Los parámetros obtenidos de la tabla de frecuencias de entrada son los siguientes:

- El Tamaño de la muestra es de 54 datos.
- El valor mínimo del rango es 10.
- El valor máximo del rango es 24.
- La frecuencia máxima observada esta en 17 y 18 automóviles que entran.
- La frecuencia mínima observada esta en 11 automóviles que entran.

El histograma de distribución de frecuencias observadas y teóricas de entrada de automóviles es mostrado en la figura 3.4

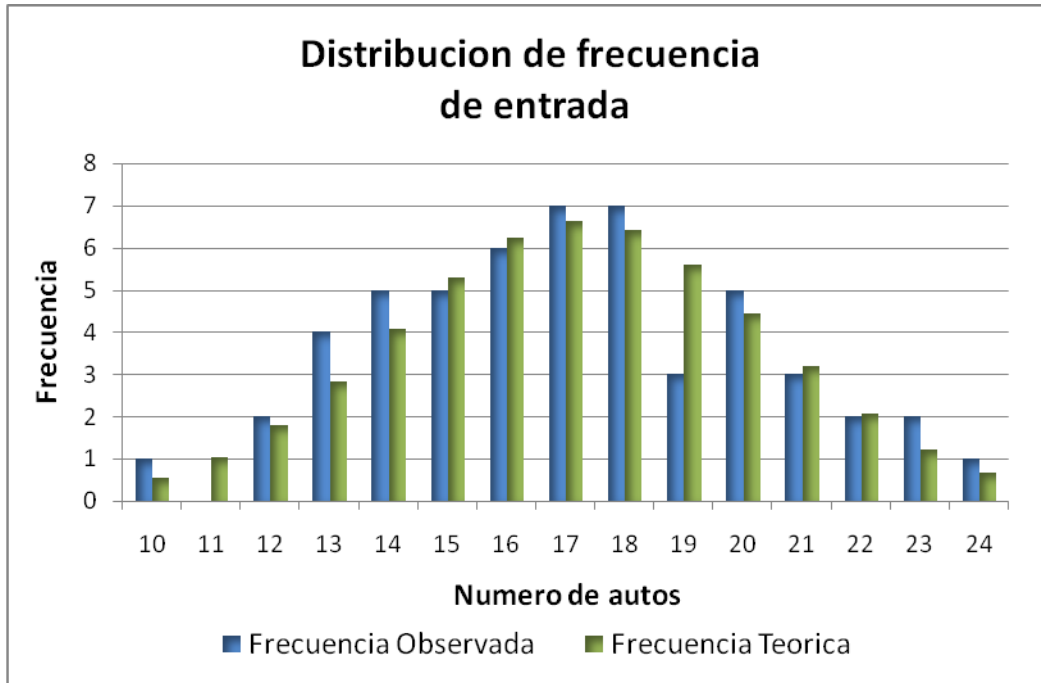


Figura 3.4 Grafica de frecuencia de entrada

Flujo de salida

Una vez concluido el análisis de flujo automóviles que entran se empieza con el análisis de flujo de salida. Se sigue el mismo procedimiento que con el flujo de entrada, se analiza los videos para realizar el conteo de automóviles que salen en cada semáforo en verde, se almacenan los datos obtenidos en una tabla de Excel y se grafican. La figura 3.5 nos muestra la gráfica del flujo de salida de automóviles.

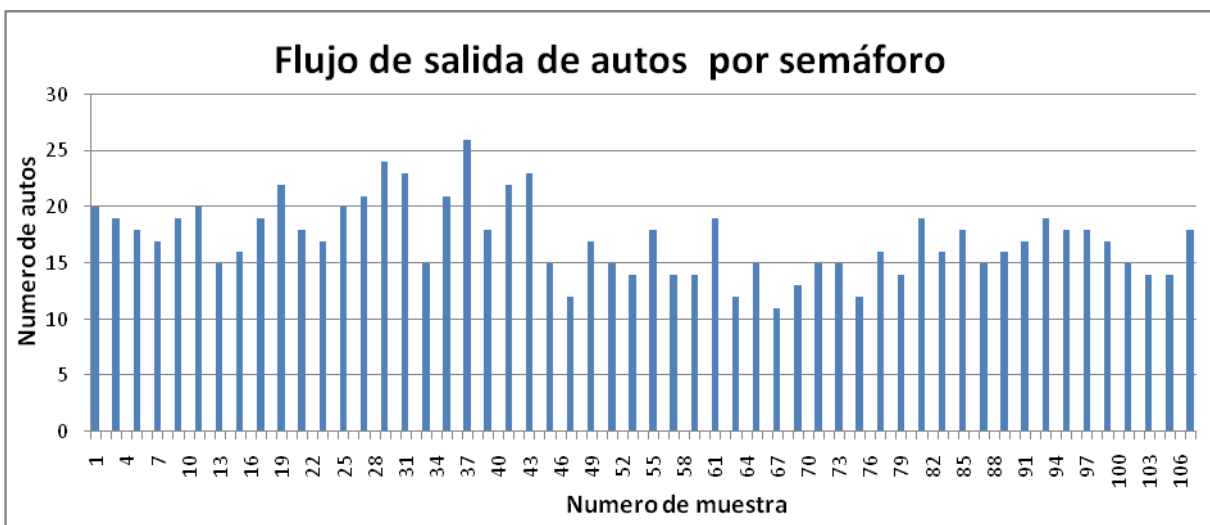


Figura 3.5 Flujo de salida



En la figura 3.5 se observa que cuando el semáforo esta en verde el número de salidas tiene una grafica uniforme, esto debido a la poca variación que existe entre el número de vehículos que salen cada vez que el semáforo esta en verde.

Cuando el semáforo esta en rojo nos arroja números en 0, debido a que ningún vehículo debe pasar cuando el semáforo esta en rojo.

La Tabla 3.2 muestra las frecuencias observadas y las frecuencias teóricas de salida de automóviles por semáforo.

Tabla de frecuencias de flujo de salida de automóviles		
Número de Automóviles	Frecuencia observada	Frecuencia Teórica
11	1	0.96
12	3	1.70
13	1	2.75
14	5	4.00
15	7	5.28
16	6	6.30
17	6	6.80
18	8	6.64
19	6	5.86
20	3	4.68
21	2	3.38
22	2	2.21
23	2	1.31
24	1	0.70
25	1	0.34

Tabla 3.2

Tabla de frecuencia de salida

Los parámetros obtenidos de la tabla de frecuencias de salida son los siguientes:

- El Tamaño de la muestra es de 54 datos.
- Número de autos que salen mínimo son 11.
- Número de autos que salen como máximo es de 25.
- La frecuencia máxima observada esta en 18 automóviles que salen.
- La frecuencia mínima observada esta en 11, 13, 24, 25 automóviles que salen.



El histograma de distribución de frecuencias observadas y teóricas de salida de automóviles es mostrado en la figura 3.6

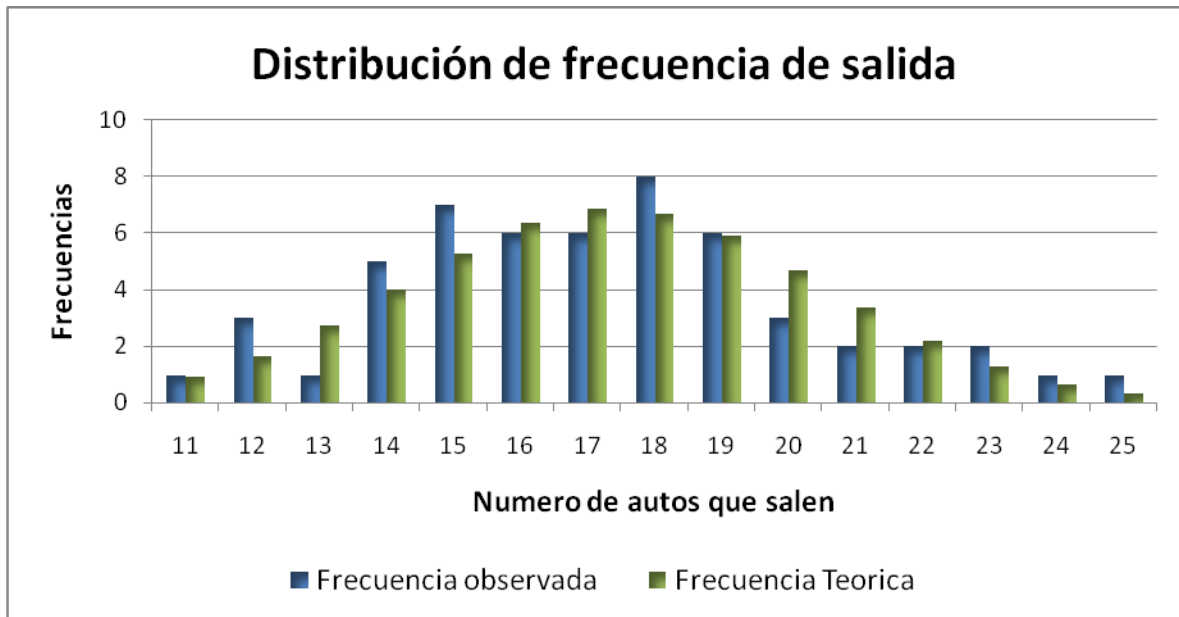


Figura 3.6 Histograma de frecuencias de salida de automóviles

Tamaño de la cola cuando el semáforo esta en rojo

Se analizan los videos cada vez que el semáforo se pone en rojo se cuenta cuantos automóviles quedan en la calle a espera de que el semáforo cambie a verde.

Se capturan los datos en una tabla, se obtienen los parámetros necesarios para la construcción de la tabla de frecuencias, los parámetros obtenidos son los siguientes:

- El Tamaño de la muestra es de 54 datos.
- Número de autos mínimo en la cola de espera es de 0.
- Número de autos máximo en la cola de espera es de 7.
- La frecuencia máxima observada de autos en la cola de espera es de 3 autos.
- La frecuencia mínima observada de autos en la cola de espera es de 6 y 7 autos.

La tabla 3.3 muestra las frecuencias observadas y las frecuencias teóricas de automóviles en la cola de espera cuando el semáforo está en rojo.



Tabla de frecuencias de automóviles en la cola de espera		
Número de automóviles	Frecuencia observada	Frecuencia Teórica
0	4	0.93
1	4	4.26
2	12	11.04
3	11	16.16
4	11	13.35
5	8	6.23
6	2	1.64
7	2	0.24

Tabla 3.3

Tabla de frecuencias de automóviles en la cola de espera.

La figura 3.7 muestra el histograma de frecuencias observadas y las frecuencias teóricas de automóviles, en la cola de espera, en la calle cuando el semáforo está en rojo.

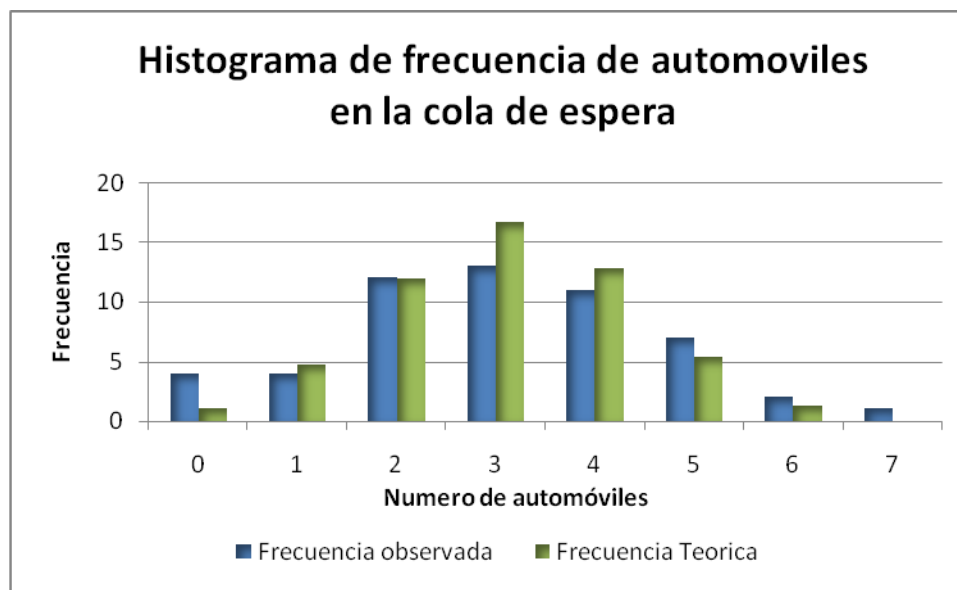


Figura 3.7

Histograma de frecuencias de automóviles en la cola de espera

Las frecuencias observadas son menores que las teóricas en el área central de la curva que forma al graficar la frecuencias teóricas.



3.3 Modelos estadísticos

Modelo estadístico de la entrada de automóviles

Basándonos en la tabla 3.1 de frecuencia de entrada de automóviles a la calle y su histograma determinamos que la distribución de probabilidad que más se ajusta a nuestros datos es la Distribución normal o Gaussiana.

La distribución normal, también llamada distribución de Gauss o distribución Gaussiana, es la distribución de probabilidad que con más frecuencia aparece en estadística y teoría de probabilidades. Esto se debe a dos razones fundamentalmente:

- Su función de densidad es simétrica y con forma de campana, lo que favorece su aplicación como modelo a gran número de variables estadísticas.
- Es, además, límite de otras distribuciones y aparece relacionada con multitud de resultados ligados a la teoría de las probabilidades gracias a sus propiedades matemáticas. [7]

La forma de la función es la llamada *campana de Gauss*, la figura 3.8 nos muestra la forma de la campana de una distribución Gaussiana.

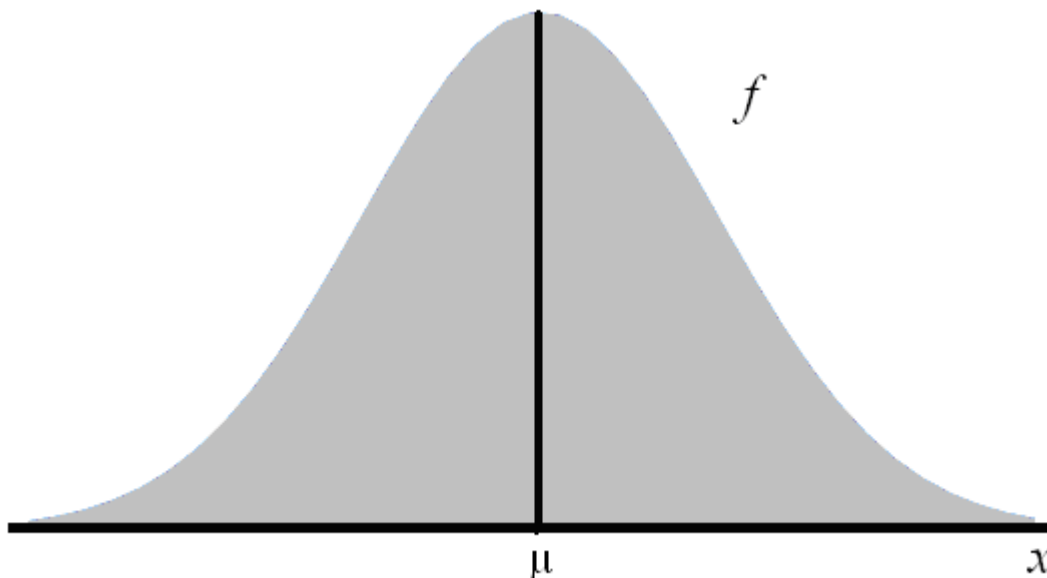


Figura 3.8 Distribución Gaussiana



La función de densidad está dada por la siguiente función:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.1)$$

Donde μ es la media y σ es la desviación estándar

Propiedades de la distribución normal:

La distribución normal posee ciertas propiedades importantes que conviene destacar:

- Tiene una única moda, que coincide con su media y su mediana.
- La curva normal es asintótica al eje de abscisas
- Es simétrica con respecto a su media μ . Según esto, para este tipo de variables existe una probabilidad de un 50% de observar un dato mayor que la media, y un 50% de observar un dato menor.
- La distancia entre la línea trazada en la media y el punto de inflexión de la curva es igual a una desviación típica. Cuanto mayor sea σ , más aplanada será la curva de la densidad.

La forma de la campana de Gauss depende de los parámetros μ y σ . La media indica la posición de la campana, de modo que para diferentes valores de μ la gráfica es desplazada a lo largo del eje horizontal. Por otra parte, la desviación estándar determina el grado de apuntamiento de la curva. Cuanto mayor sea el valor de σ , más se dispersarán los datos en torno a la media y la curva será más plana. Un valor pequeño de este parámetro indica, por tanto, una gran probabilidad de obtener datos cercanos al valor medio de la distribución.

Estimación de parámetros

Después de haber seleccionado la distribución estimaremos los parámetros correspondientes a esta distribución. Para calcular una distribución Normal necesitamos la media μ y la desviación estándar σ , así que calcularemos primero estos parámetros.



Para calcular la media aritmética del histograma y encontrar la tendencia central más utilizada la fórmula es la siguiente:

$$\mu = \frac{\sum(X_i)}{N} \quad (3.2)$$

X_i = Número de autos que entran por semáforo

N = Número total de muestras

La varianza nos indica el parámetro de dispersión de la tasa de entrada de los automóviles

$$\sigma^2 = \frac{\sum(X_i - \mu)^2}{(N-1)} \quad (3.3)$$

Para calcular nuestro modelo estadístico que alimentara la simulación es necesario calcular la desviación estándar. La varianza se calcula con la siguiente fórmula:

$$\sigma = \sqrt{(\sum(X_i - \mu)^2) / ((N - 1))} \quad (3.4)$$

Utilizamos la ecuación (3.2) para calcular la media del histograma de distribución de frecuencia de los automóviles que entran esta dada por:

$$\mu = \frac{926}{54}$$

Dando como resultado que la media es igual a :

$$\mu = 17.1481481$$

Utilizamos la ecuación 3.3 para el cálculo de la varianza nos da como resultado que la varianza es igual a:

$$\sigma^2 = 10.1285814$$

Por lo tanto la desviación estándar es igual a

$$\sigma = \sqrt{10.1285814}$$

$$\sigma = 3.18254323$$



La frecuencia máxima observada esta dada en 15 y 18 automóviles.

La media es igual a: 17.1481481

La desviación estándar es: 3.18254323

De acuerdo al modelo podemos calcular el número de entradas de los automóviles sustituyendo los valores de la media, la desviación estándar en la ecuación (3.1) de la distribución Normal.

$$F(x) = \frac{1}{3.1825\sqrt{2(3.141516)}} e^{-\frac{(x-17.1481481)^2}{2(3.1825)^2}}$$

Esta función será utilizada en nuestra simulación para obtener el flujo de entrada de los automóviles. Se genera un random Gaussiano con los parámetros obtenidos de la media y la desviación estándar, para que nos de el valor de flujo de entrada por semáforo en la simulación.



Capítulo 4

Ambiente de desarrollo de la simulación NetLogo

NetLogo es un lenguaje de modelación programable, que permite la simulación de fenómenos que se dan en la naturaleza, las sociedades o aéreas de las ciencias exactas, de una forma muy simple, fue creado por Uri Wilensky in 1999. Esta es una de las principales razones por las cuales fue elegido este ambiente de desarrollo, la facilidad que ofrece para programar de una forma fácil un modelo complejo como es el tráfico vehicular.

NetLogo permite crear modelos simples o complejos que tienen cientos de individuos (personas, animales, partículas, bacterias, nodos de un grafo etc.) llamados agentes que operan independientemente, interactúan entre si y con su ambiente además de evolucionar con el tiempo. Cada automóvil será un individuo o agente, con características diferentes, interactuando entre si en un medio ambiente que es la carretera.

Los sistemas y modelos basados en agentes tienen sus raíces en los intentos de modelar el comportamiento social de los seres humanos y de modelar la capacidad de las personas para tomar decisiones ante la información que reciben del mundo exterior. Entre las áreas de las ciencias que han aprovechado los modelos y diseños orientados a agentes están las ciencias sociales, para estudiar los comportamientos complejos entre las relaciones de los seres humanos; las ciencias médicas para simular las propagaciones de las enfermedades; las ciencias económico administrativas para encontrar clientes en la WEB, y muchos otros campos.

Un agente es visto como una pieza de software con métodos propios que tiene las siguientes características [8]:

- Es identificable
- Es ubicable
- Tiene un objetivo que cumplir
- Es autónomo
- Es flexible

Cada automóvil situado en la carretera será un agente que cumple con estas características.



Un agente debe ser claramente identificable, teniendo al menos un identificador que lo distinga, dotado de reglas o características que gobiernan su comportamiento y su capacidad de tomar decisiones. NetLogo proporciona una variable llamada *Who* para cada agente, en este caso cada automóvil tendrá una variable *who* que nos da el número de identificación del automóvil.

En cualquier momento el programador o el usuario dueño del agente debe tener control sobre el agente, de modo que debe ser posible obtener sus coordenadas de ubicación. NetLogo nos permite en todo momento obtener la posición de nuestro automóvil, la posición esta dada por la variable *xcor* y *ycor*

Los agentes tienen que cumplir cierta tarea. La tarea de nuestros autos es la de transitar por la carretera basándose en el modelo *Car-Following* para interactuar con el automóvil de enfrente.

Un agente no necesita de otros agentes para poder vivir, a menos que así sea programado, en general, un agente es autónomo y auto-dirigido; puede funcionar de manera independiente en su ambiente y en su trato con otros agentes, siempre que compartan los mismos protocolos de comunicación. Nuestros automóviles serán autónomos.

Los agentes son flexibles pueden tener reglas para modificar sus reglas de comportamiento

En la búsqueda del conocimiento y entendimiento de nuestro entorno se ha buscado la manera de crear modelos que representen los fenómenos que observamos, jugar con las variables que interactúan en él, analizar los resultados que se obtienen, predecir comportamientos etc. NetLogo es una herramienta que permite simular los modelos y experimentar con las simulaciones, comprobar hipótesis sobre los sistemas rápidamente.

4.1 Características de NetLogo

NetLogo es una herramienta de programación muy novedosa en la simulación del tráfico vehicular explotaremos la mayoría de estas características.

- NetLogo se basa en el paradigma de modelización por agentes, cada automóvil será un agente con sus respectivas variables



- Es multiplataforma ya que funciona en MacOS, Windows, Linux esto permite que la simulación pueda ser corrida en la mayoría de las computadoras siempre que tenga instalado NetLogo.
- Es software gratuito, esto permite que cualquier persona lo descargue de internet y pueda ejecutar la simulación sin problemas.
- Posee una interfaz gráfica intuitiva y fácil, que permitirá que los usuarios corran la simulación de una manera muy fácil sin necesidad de tener amplio conocimiento de la herramienta.
- Nuestros modelos de simulación de tráfico vehicular pueden ser guardados como applets y subidos en páginas Web para su publicación
- Constructor de la interfaz: botones, sliders, switches, choosers, monitores, texto boxes, notes, output área. En este caso para la creación de la interfaz de la simulación utilizaremos botones para manipular la simulación, sliders para introducirle datos a el modelo y monitores para mostrar los resultados que va arrojando nuestro modelo conforme avanza el tiempo. en la creación de la interfaz de la simulación.
- Tiene un deslizador de velocidad le permite cambiar la velocidad a la que se ejecutamos nuestro modelo.
- Poder y flexibilidad del sistema plotting que permitirá ir graficar los datos que representan la evolución del modelo de tráfico vehicular.
- Consta de un área información en la que se describirá brevemente el modelo.
- Monitores de agentes para inspeccionar y controlar los agentes.

4.2 Interfaz

Al abrir NetLogo nos muestra una pantalla (Figura 4.1) que esta dividida en el menú de opciones, y el área de la interfaz que tiene un área en blanco y otra negro. La región en blanco es donde se colocaremos los objetos que sirven para controlar la simulación, para evaluar resultados y para obtener información de lo que sucede en la simulación.



La parte en color negro es la ventana de gráficos se utiliza principalmente en la versión 2D aquí correrá la simulación del tráfico vehicular.

El área de la interfaz tiene tres etiquetas Interface, Information y procedure, sus funciones son las siguientes:

Interface:

Es la parte con la que va interactuar el usuario, de nuestro sistema la cual consta de instrumentos como botones, con los cuales se puede inspeccionar o alterar el modelo, además, slider, monitores y plotting que permite ver el estado de evolución de la simulación. La figura 4.1 nos muestra los elementos que requiere la interfaz de la simulación, más adelante serán explicados cada uno de los objetos de los que consta la interfaz de la simulación del tráfico vehicular.

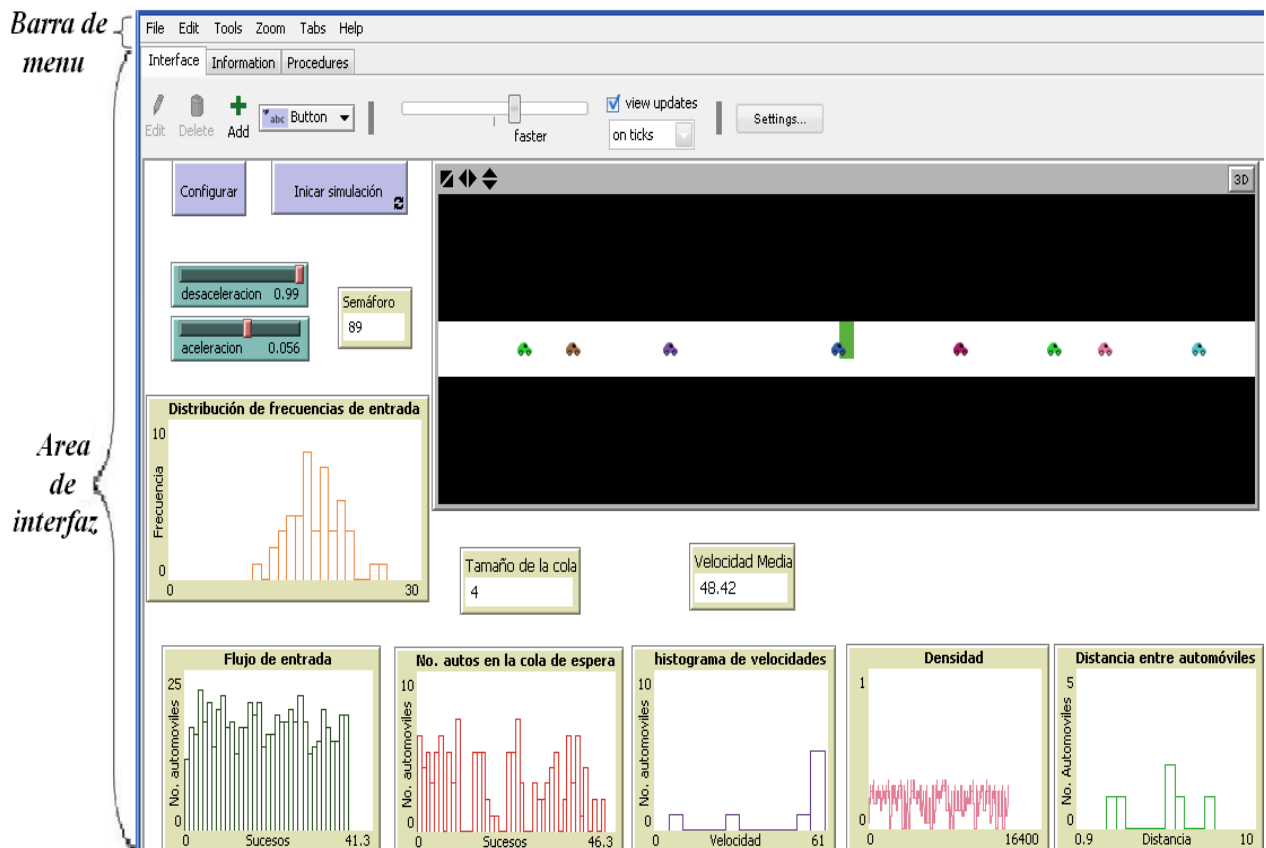


Figura 4.1 Ventana principal de NetLogo



Information:

El área de información se proporciona una introducción al modelo y una explicación de cómo usarlo.

Procedures:

Es la parte en donde esta colocado el código del modelo de simulación de tráfico vehicular.

El mundo de NetLogo se compone de agentes existen tiene tres tipos de agentes

Turtles: Son agentes visibles en la pantalla de simulación, que se pueden mover en el mundo (El mundo es 2D y está dividido en una malla de patches) según los requerimientos del usuario, las tortugas tienen propiedades y reaccionan ante la información que reciben del mundo exterior, de acuerdo a lo programado por el observador. Programaremos cada turtle o agente para que actúe como un automóvil que interactúa con otros automóviles en una carretera.

Patches: La pantalla en la que corre la simulación de los modelos está dividida en celdas llamadas Patches, cada Patche es un agente, que no se mueven, tienen menos atributos que las tortugas, pero tienen propiedades que dan el estado del medio-ambiente que pueden ser aprovechadas por las tortugas. La carretera se creará en los patches cada patche es un tramo de la carretera.

Además de estos agentes existe un observe, el observador es el usuario final, el que puede crear y modificar los atributos y métodos de los agentes; solamente existe un observador.

Elementos de la interfaz de la simulación

La interfaz está compuesta por varios componentes gráficos como son botones, sliders, monitores, área de salidas y plots a continuación se definen los elementos utilizados.

Botones

Cuando pulsa un botón la simulación responde con una acción, un botón puede definirse para ser ejecutado una sola vez o ejecutarse indefinidamente.

La interfaz de la simulación tendrá dos botones el botón *Configurar* e *Iniciar simulación* mostrados en Figura 4.2

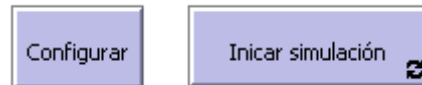


Figura 4.2 Botones de la interfaz

El botón *Configurar* esta definido para ejecutarse una sola vez, este botón realiza la inicialización de variables necesarias en la simulación y pinta la carretera y el semáforo en la pantalla. El botón *Iniciar simulación* ejecuta las acciones definidas para este indefinidamente, dejara de ejecutarse cuando vuelva a ser presionado, las dos flechitas en la parte inferior derecha del botón, indican que este se ejecutara indefinidamente. Iniciar simulación llama a los procedimientos necesarios para que inicie la simulación y los automóviles entren a la calle e interactúen moviéndose de izquierda a derecha, utilizando como base para esta interacción el modelo Car-Following.

Slider

Son variables globales, que van cambiando su valor de acuerdo al valor que el usuario le asigne, todos los agentes pueden utilizar estas variables. En la simulación se utilizaran dos slider mostrados en la figura 4.3 una llamado *desaceleración*, que tendrá como fin que el usuario pueda modificar el valor de la desaceleración de los automóviles para modificar su valor se tiene que arrastrar la barra hacia la izquierda para disminuir y a la derecha para aumentar su valor. El otro slider es llamado *aceleración* con este se define el valor de la aceleración de los automóviles.

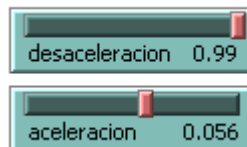


Figura 4.3 Slider de la interfaz

Monitor

Muestra información del modelo y despliega el valor de cualquier expresión. La expresión podría ser una variable, una expresión compleja, o una llamada para un informe. La interfaz de la simulación cuenta con tres monitores Figura 4.4 el monitor llamado *semáforo* muestra el número de semáforo en el que va recorriendo la simulación, el monitor *Tamaño de la cola* nos informa del tamaño de cola que esta formado en la carretera antes del semáforo, el monitor *Velocidad media* calcula y nos informa de la velocidad media que llevan los automóviles.

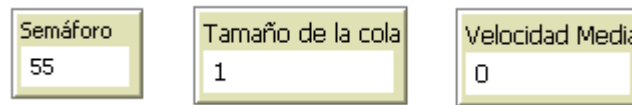


Figura 4.4 Monitores utilizados en la simulación.

Plot

Son gráficos que nos muestran lo que ocurre con el modelo a través del tiempo. La interfaz de la simulación cuenta con seis plots, el primero de ellos es mostrado en la figura 4.5 a) Este plot nos genera una grafica de la distribución de frecuencia de entrada de automóviles a la carretera definimos valores de 0 a 30 para el eje de las abscisas y de 0 a 10 para las ordenadas, pero NetLogo permite que el plot se vaya ajustando de acuerdo a las necesidades de la simulación, si se rebasara el número de frecuencias de entrada Netlogo agrandaría el rango de graficado.

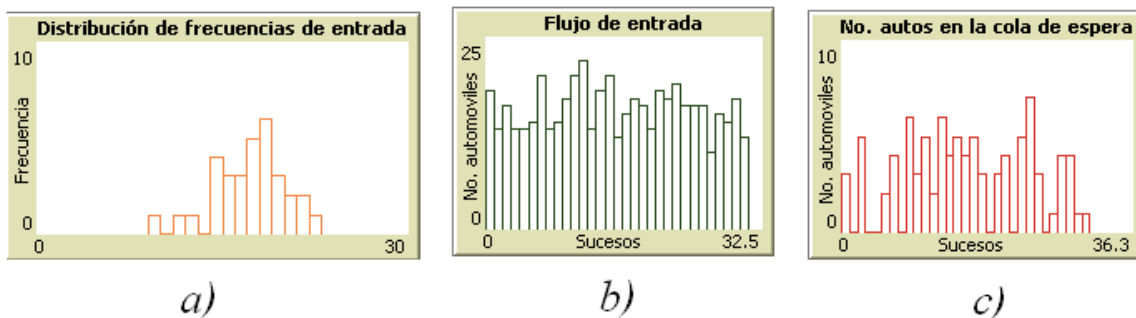


Figura 4.5 Plots de a) Distribución de frecuencias de entrada
b) Flujo de entrada y c) No. De autos en la cola de espera

La figura 4.5 b) gráfica el flujo de entrada a la calle por semáforo, el rango de las ordenadas es de 0 a 25 este rango no cambia debido a que la función de densidad utilizada para la simulación tiene como flujo de entrada máximo 25 automóviles por semáforo, el eje de las abscisas, si varia, se va agrandando de acuerdo al número de entradas. La figura 4.5 c) gráfica el tamaño de la cola antes el semáforo.

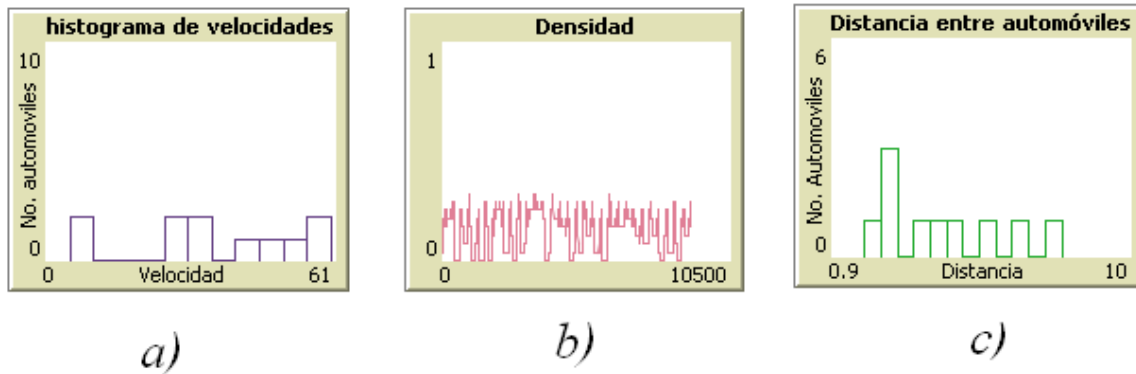


Figura 4.6 a) Histograma de velocidades b) Gráfica de densidad
c) Distancia entre automóviles

El plot de la figura 4.6 a) grafica las velocidades que llevan los automóviles, el plot de la figura 4.6 b) grafica la densidad de automóviles en la carretera y el plot de la figura 4.6 c) grafica la distancia entre los automóviles entre un auto y el de en frente.

Slider de Velocidad

Este Slider de velocidad (Figura 4.7) permite controlar la velocidad de un modelo. Cuando el slider es movido a la derecha la velocidad con la que corre la simulación aumenta y cuando se mueve a la izquierda disminuye la velocidad de corrida.



Figura 4.7 Slider de velocidad

Así que el usuario podrá hacer que el modelo de tráfico vehicular corra más lento o más rápido con solo mover el slider.

Settings

Al darle clic a este botón nos muestra una ventana en la que podemos ajustar la vista del área donde corre la simulación, el tamaño de la pantalla, del patch etc.



La figura 4.8 nos muestra la ventana de configuración de la pantalla de simulación.

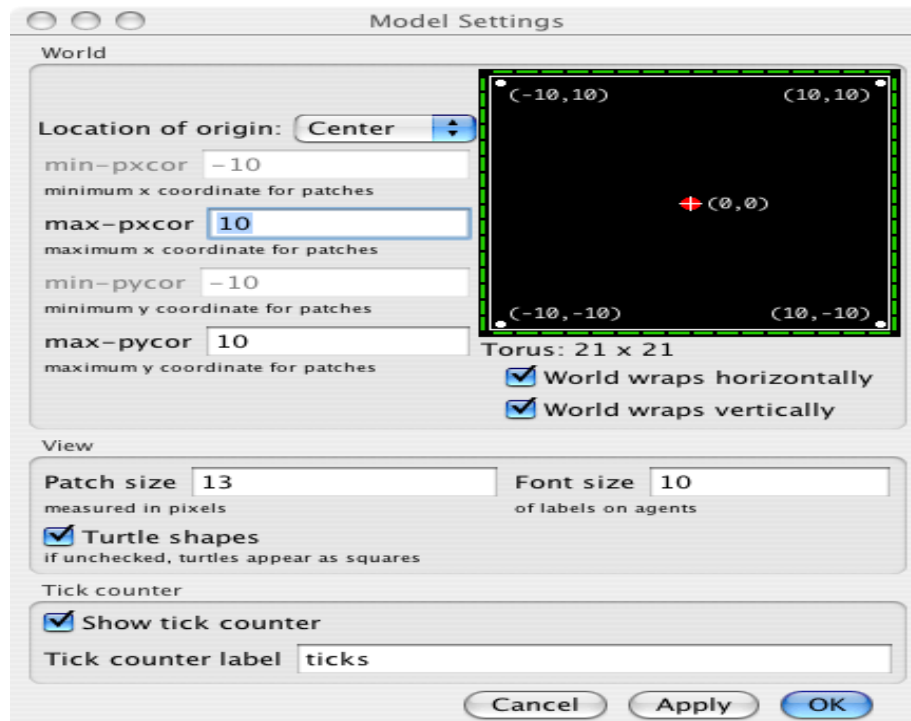


Figura 4.8 Ventana Model Settings

También nos permite configurar a los agentes para que no puedan pasar de la orilla de la pantalla, con las opciones *world wraps horizontal* y *vertical*, si no se seleccionan estas opciones el agente podrá caminar hasta el final de la pantalla y regresar al inicio nuevamente.

4.3 Programación del sistema de simulación

Variables

NetLogo tienen variables Globales y locales, los agentes tienen algunas variables definidas por default en NetLogo y además se pueden definir nuevas variables para los agentes, en NetLogo no es necesario definir el tipo de una variable, cada variable almacena el valor que se le asigne sin necesidad de decirle de que tipo es.



Cada agente tiene sus propias variables ya definidas veremos la más importantes de cada agente.

Variables de Turtles

Existen variables definidas por default por NetLogo para cada automóvil como lo son:

- `xcor`, `ycor`: Nos indican las coordenadas x , y de donde está situado el auto en la pantalla.
- `Shape`: Variable que almacena la forma que mostrará el agente en la pantalla, en el caso de nuestra simulación mostrará la figura de un auto y la variable almacenará el nombre de la imagen que es "car"
- `Who`: Almacena el identificador del automóvil, el cual es un número.
- `Heading`: Almacena la orientación de las tortugas, es una medida en grados a partir del eje y y hacia donde se dirige la tortuga.

Variables de Patches

Al igual que con las turtles los patches también tienen variables definidas por default por NetLogo a continuación las más importantes:

- `pxcor`, `pycor`: Nos indican las coordenadas x , y de la celda o Patche en la pantalla.
- `Pcolor`: Variable que almacena el color del patche modificando el color del patche pintamos la carreta el semáforo etc.

El usuario puede definir sus propias variables para las tortugas y patches, cada tortuga tendrá una variable con este nombre y se le puede asignar valores diferentes.

Variables globales

Una variable global que será la misma para todos los agentes, y podrá ser utilizada en todo el programa. Las variables globales definidas en la simulación del tráfico vehicular son:

*globals [num_semaforo contador luz número-autos temp autos_calle
densidad contador_distribuciones velocidad-media tiempo-semaforo]*



num_semáforo almacena el contador del número de cambios de semáforo que van ocurriendo.

densidad almacena la densidad de tráfico en la carretera.

Número de autos almacena el valor del flujo de entrada de automóviles.

Autos_calle almacena el número de autos que están antes del semáforo.

Velocidad-media almacena los cálculos de la velocidad media de los automóviles.

Variable local

Las variables que define el usuario para cada gente son definidas con la palabra reservada `turtles-own` y entre corchetes el nombre de las variables.

```
turtles-own [ velocidad velocidad-max velocidad-min distancia]
```

Para cada automóvil serán creadas cuatro variables aparte de las que ya están definidas por default en Netlogo las variables son las siguientes:

Velocidad almacena la velocidad que lleva el automóvil en cada momento.

Velocidad-max almacena la velocidad máxima a la que puede ir el automóvil.

Velocidad-min almacena la velocidad mínima del automóvil.

Distancia almacena la distancia entre este automóvil y el de en frente.

Para modificar el valor de una variable, se debe hacer uso del comando `set nombre-variable valor`.

```
set num_semáforo 1
```

Una variable local es definida y usada solo en el contexto de un procedimiento o parte de un procedimiento.

Si queremos acceder al valor de una variable global sólo se llama con su nombre, con las variables locales es lo mismo la única diferencia es que una variable local solo puede ser llamada dentro del procedimiento o reporte, y la variable global es utilizada en todo el programa.

En la siguiente instrucción llamamos una variable global y la envía a graficar su contenido:

```
plot int número-autos
```



Las variables asociadas a agentes debe realizarse desde dentro de un entorno que especifique a qué agente nos referimos, y en ese caso, el nombre de la variable devolverá el valor de la variable asociada a dicho agente.

Utilización de variables de un agente *ask turtle a[set distancia distance turtle b]*

a es una variable que tiene almacenado el número de identificador del automóvil del cual estamos utilizando su variable.

Procedimientos y reportes

En net logo hay dos formas de formas de comunicarse con un agente y es a través de los procedimientos y los reportes. Un procedimiento dice a los agentes lo que deben de hacer y los reportes calculan un valor y devuelve un resultado. En NetLogo existen comandos y reportes ya definidos a estos se les llama primitivos, y el usuario puede definir también sus propios procedimientos y reportes.

Un procedimiento debe tener la siguiente sintaxis:

```
to inicializar_autos  
  set heading 90  
  set velocidad 0.1 + random-float .9  
  set velocidad-max 1  
  set velocidad-min 0  
end
```

to nombre del procedimiento, después el grupo de instrucciones y cerrar con end.

Para llamar un procedimiento o reporte se usa directamente su nombre junto con los valores asociados a los parámetros de entrada.

En la simulación utilizaremos varios procedimientos a continuación la forma de llamar un procedimiento de un agente *turtle* que en este caso serán los automóviles es la siguiente:

```
ask turtles[inicializar_autos]
```

Para llamar procedimientos definidos por Netlogo para un patch es necesario definir sobre que patch se va a realizar el procedimiento, así que se definen las coordenadas y después se escriben los procedimientos que se calcularan sobre este patch.

```
ask patch min-pxcor 0 [ ifelse any? turtles in-radius 2 [set temp 0] [set temp 1] ]
```



Agentsets

Un Agentsets puede contener un conjunto de agentes un conjunto de Patches pero nunca ambos. Esto nos permite aplicar comandos selectivamente a un grupo de agentes que tengan cierta característica una de las formas utilizadas para seleccionar un grupo de agentes es con *with*.

En la siguiente instrucción se seleccionan el grupo de agentes que están situados antes del semáforo y se cuentan.

```
set autos_calle count turtles with [xcor <= 0]
```

Y luego se aplican algunos comandos sobre este grupo de agentes con la siguiente sintaxis

```
ask conjunto_de_agentes [condición] [comandos]
```

En NetLogo se diseña por separado la Interfaz de la simulación y la programación del código, lo ideal es primero crear la interfaz para que cuando se vaya programando no marque errores al ser por que falta algún objeto al cual se esta haciendo referencia.



Capítulo 5 Simulación

Simulación es el proceso de diseñar un modelo de un sistema real y llevar a cabo experiencias con él, con la finalidad de aprender el comportamiento del sistema o de evaluar diversas estrategias para el funcionamiento del sistema [Shannon, 1988]. La simulación nos permite observar el comportamiento de un fenómeno o sistema en minutos, segundos el comportamiento que duraría horas, días o incluso meses

Una de las mayores virtudes de los modelos de simulación es lograr entregar información detallada frente a los posibles cambios del sistema permitiendo anticipar soluciones correctas a los problemas reales [9].

5.1 Sistemas de Eventos Discretos

Los modelos de eventos discretos son modelos dinámicos, estocásticos y discretos en los que las variables de estado cambian de valor en instantes de tiempo dados. Los elementos fundamentales en una simulación por medio de eventos discretos son las variables y los eventos

El evento es la representación de un cambio instantáneo en alguna parte de un sistema, el mismo que puede caracterizarse por un valor y un instante en el que ocurre. El valor puede ser un número, un vector, una palabra o en general un elemento cualquiera de un conjunto determinado. En la simulación del tráfico vehicular los eventos más importantes para realizar la simulación es el estado del semáforo si el semáforo esta en verde pueden entrar y salir a automóviles pero si el semáforo cambia a rojo dejan de salir y entrar automóviles.

En resumen, un cambio en el estado de un sistema se denomina evento.

Atributos de eventos discretos

Los atributos de eventos discretos que caracterizan a una red urbana, pueden ser divididos en cuatro clases:

- Cambios de fases del semáforo, esta clase de eventos afecta la dinámica de tiempo continuo, ya que el cambio de cada fase, habilita o deshabilita los flujos de entrada



(Acceden al recurso). Los cambios de estado están determinados por los semáforos, que tienen su propia dinámica de evento discreto.

En las intersecciones señalizadas, en lo que concierne a la dinámica del semáforo, cada fase es caracterizada por los flujos f que están habilitados dentro de dicha fase. En consecuencia solamente los intervalos de verde y ámbar se toman en cuenta, puesto que todos los flujos de entrada que no encuentran un recurso disponible (luz verde o ámbar), están frente a un rojo, lo cual da lugar a la formación de una cola q y los tiempos de espera.

- Los eventos de requerimiento de control, cuando un flujo de entrada cambia, cambia la dinámica de la cola y el plan óptimo de semáforos deberá ser re calculado, utilizando un controlador prioritario.
- Eventos especiales, caracterizados por situaciones extraordinarias como accidentes, paro de vehículos, en los que se produce un bloqueo o cambio en los flujos.
- Eventos de tipo general como son el destino, el número y tipo de vías en una carretera, el tipo de usuario.

En la presente simulación sólo se tomará en cuenta los eventos cambio de fase del semáforo, la luz roja y la luz verde se tomarán como un mismo evento, ya que para la mayoría de los conductores la luz ámbar es como si fuera verde.

5.2 Descripción del fenómeno

El fenómeno a simular es el tráfico vehicular de un carril del boulevard 5 de mayo de la ciudad de Puebla, entre las calles Palafox y Mendoza y 2 oriente en sentido norte-sur el modelo a utilizarse es el modelo microscópico Car-following.

Las variables necesarias para el modelo Car-Following son velocidad de cada automóvil, distancia entre automóviles, aceleración y desaceleración de los automóviles

La velocidad de los automóviles estará representada a través de una variable aleatoria denominada velocidad-automóvil. La distancia entre automóviles deberá ser calculada en la simulación.

En el trabajo de campo determinamos otros aspectos de la simulación, como lo es el arribo de automóviles a la calle, la cual va a estar dada por una distribución Normal o Gaussiana.



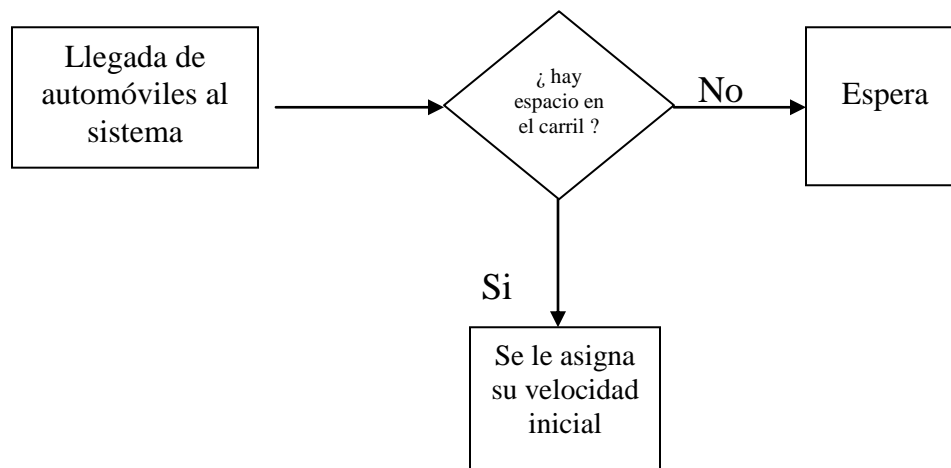
La simulación representa un tramo de carretera con semáforo, por esta razón será necesario determinar el valor del tiempo que tarda en cambiar el semáforo y el tamaño de la calle estos datos ya fueron obtenidos en el trabajo de campo el valor de los tiempos es de tipo determinante. Así también en el trabajo de campo obtuvimos la distribución de entrada y de salida de automóviles

Nuestro sistema es dinámico evoluciona a lo largo del tiempo, estocástico por que la llegada de los automóviles esta dada por una variable aleatoria, y discreto por que va a cambiar de acuerdo a la fase del semáforo.

5.3 Formulación del modelo

Llegada de automóviles a la calle

El siguiente diagrama nos modela que es lo que sucede cuando un automóvil llega a la calle.



Cuando llega un automóvil a la calle primero verifica si hay lugar en la calle para que el pueda ingresar si hay espacio entra y se le asigna una velocidad aleatoriamente si no hay espacio espera.



Aceleración y desaceleración

El movimiento de los automóviles está basado en la ecuación básica del modelo Car-Following [ecuación 2.10]

Una vez que entra el automóvil a la carril sus acciones van a estar determinadas de acuerdo al estímulo que reciba, el estímulo estará dado por la distancia y la velocidad del auto de en frente. El conductor trata de mantener una distancia prudente del vehículo de en frente para evitar colisiones pero a la vez tratara de estar lo más cerca del vehículo siguiente.

Se definirá una distancia segura mínima entre automóviles, si la distancia entre los automóviles es mayor a la mínima el automóvil acelera, si la distancia es menor o igual desacelera.

Salida de automóviles

La salida de la calle depende totalmente del estado de fase del semáforo cuando llega un automóvil al semáforo verifica el estado del semáforo, si esta en verde sigue adelante, si no, se detiene y espera a que el semáforo cambie a verde. Estas acciones son calculadas por el siguiente código. La primera línea pregunta si ya llego al semáforo y de que color es, si es rojo desacelera y su velocidad se vuelve 0 de la línea 3 a 6 es el caso en el que el semáforo esta en verde si esta en verde realiza los cálculos para avanzar.

1. *ask turtles [ifelse pcolor = red [set velocidad 0]*
2. *[*
3. *ifelse any? turtles-at 2 0*
4. *[set velocidad ([velocidad] of one-of turtles-at 2 0) desacelerar-auto]*
5. *[acelerar-auto]*
6. *fd velocidad]*

5.4 Implementación del modelo en la computadora

En el capítulo anterior se dio una breve introducción al ambiente de desarrollo de esta simulación, NetLogo da muchas facilidades de modelar sistemas complejos de una forma rápida y sencilla sin necesidad de tantas líneas de código, y reduce el tiempo invertido en la programación.



Código de la simulación

A continuación se dará una breve explicación de cómo se implementó el modelo en Netlogo. Lo primero que se tiene que hacer es definir los valores iniciales de las variables y dibujar la carretera y el semáforo, el código que se muestra a continuación se encarga de ello.

Este código es llamado por el botón de Inicializar mostrado en la figura 4.2 del capítulo anterior.

```
to inicializar_variables
```

```
  set num_semaforo 1
```

```
  set luz 1
```

```
  set contador_distribuciones [ ]
```

```
  ask patch 0 0 [set pcolor green ]
```

```
end
```

```
to dibujar-carretera
```

```
  if ( pycor < 2 ) and ( pycor > -2 ) [ set pcolor white ]
```

```
end
```

El procedimiento dibujar carretera pinta los Patches que sus coordenadas y estén entre 2 y -2 de color blanco

Una vez inicializadas las variables, dibujada la carretera y el semáforo podemos empezar a correr nuestra simulación.

El código siguiente es ejecutado con el botón Inicializar simulación (figura 4.2) y llama a las funciones que se encargan de hacer las tareas principales de la simulación, como lo son los cambios de la luz del semáforo, el calcular velocidades si acelera o desacelera un automóvil, la distancia entre los automóviles, genera la entrada de automóviles a la calle y manda a graficar las variables y también las graficas generadas por el sistema.



to go

cambiar-semaforo

calcular-velocidad

if (count turtles > 2)

[ask turtles [if (xcor > min [xcor] of turtles + 1) [calcular-distancia]]]

ask patch min-pxcor 0 [ifelse any? turtles in-radius 2

[set temp 0]

[set temp 1]

]

crear-autos

ask turtles[desaparecer]

tick

plot_densidad

histograma-velocidades

histograma-distancia

end

A continuación vamos a analizar cada uno de los procedimientos de código

Procedimiento distribuir

Genera el flujo de automóviles, utilizando la función *random-normal* para calcular un valor de acuerdo a la distribución Normal tiene como parámetros la media y la desviación estándar de la distribución, *lpul* nos sirve para ir almacenando en una lista todos los valores del flujo de entrada y así poder graficar después un histograma de frecuencia con estos valores

to distribuir

set número-autos int (random-normal 17.1481481 3.18254323)

set contador_distribuciones lput número-autos contador_distribuciones

set contador número-autos

end



Procedimiento inicializar-autos

Cada vez que se crea un automóvil se inicializa con su velocidad inicial que esta dada de forma aleatoria, con heading se le indica hacia donde dirigirse, la velocidad máxima y mínima que es la misma para todos los automóviles.

```
to inicializar_autos
  set heading 90
  set velocidad 0.1 + random-float .9
  set velocidad-max 1
  set velocidad-min 0
end
```

Procedimiento desaparecer

Elimina los automóviles una vez que llegan al final de la calle si no se eliminan los agentes se amontonan todos hasta el final de la calle, en Netlogo es necesario siempre eliminar los agentes que son creados, cuando ya no se necesitan.

```
to desaparecer
  if xcor > max-pxcor - 1 [ die ]
end
```

Procedimiento cambiar-semáforo

Revisa si el tiempo es igual al tiempo que dura el semáforo si ya llego al tiempo que es, incrementa el número del semáforo que nos dice cuantos cambios de semáforo han ocurrido en la simulación, si el número de semáforo es non indica que el semáforo esta en verde por lo tanto ahora deberá cambiar a rojo.

Si el número de semáforo es par indica que el semáforo está en rojo y ahora se deberá cambiar a verde, cuando ocurre este cambio llama a la función distribuir que ya fue explicado anteriormente.

Dentro de este procedimiento se ejecuta el comando *count* que nos cuanta cuantos automóviles hay sobre el carril, y manda a graficar el número de autos.

Por último *reset-ticks* regresa el tiempo a ceros para que vuelva a iniciar la cuenta.



to cambiar-semaforo

if ticks = tiempo-semaforo

[set num_semaforo num_semaforo + 1

ifelse ((num_semaforo mod 2) != 0)

[ask patch 0 0 [set pcolor green] ask patch 0 1 [set pcolor green] set luz 1]

[ask patch 0 0 [set pcolor red] ask patch 0 1 [set pcolor red]set luz 0

distribuir

set autos_calle count turtles with [xcor <= 0]

autos-grafica

autos-semaforo-grafica]

reset-ticks

]

End

Procedimiento calcular velocidad

En este procedimiento lo primero que se checa es si el automóvil a llegado al semáforo, y si esta en verde o en rojo, si esta en rojo el automóvil se detiene su velocidad se vuelve 0.

ifelse any? turtles-at 2 0 revisa si existe un automóvil que este a menos de 1.5 metros, si existe algún automóvil a esa distancia el automóvil manda a llamar a la función desacelera-auto, si no, llama el procedimiento acelerar-auto.

to calcular-velocidad

ask turtles [ifelse pcolor = red [set velocidad 0]

[

ifelse any? turtles-at 2 0

[set velocidad ([velocidad] of one-of turtles-at 2 0) desacelerar-auto]

[acelerar-auto]

fd velocidad]

]

end



Procedimiento desacelerar-auto

Lo que hace este procedimiento es que a la velocidad que tenía el automóvil se le resta la desaceleración, si al desacelerar la velocidad quedara menor a la velocidad mínima se le pondra como velocidad, la *velocidad minima*

```
to desacelerar-auto
  set velocidad velocidad - desaceleracion
  if velocidad < velocidad-min [ set velocidad velocidad-min ]
end
```

Procedimiento acelerar-auto

Lo que hace este procedimiento suma a la velocidad que tenia el automóvil, la aceleración, si al acelerar la velocidad es mayor que la velocidad máxima se le asigna como velocidad la *velocidad-max*

```
to acelerar-auto
  set velocidad velocidad + aceleracion
  if velocidad > velocidad-max [ set velocidad velocidad-max ]
end
```

Procedimiento Calcular-distancia

Cada automóvil tiene un número que lo identifica este número esta almacenado en la variable *who* de acuerdo a esto si queremos calcular la distancia entre dos automóviles necesitamos su identificador *who* y el del siguiente que seria *who + 1*
Para cada automóvil calcularemos la distancia entre el y el vehículo que lo sigue la función *distance* calcula la distancia entre dos agentes Turtles que en este caso son los automóviles.

```
to calcular-distancia
  let a who
  let b a + 1
  ask turtle a[set distancia distance turtle b]
  set distancia (distancia - .5)
end
```



Procedimiento crear- autos

En el procedimiento go tenemos este pedazo de código.

```
ask patch min-pxcor 0 [ ifelse any? turtles in-radius 2
                        [set temp 0]
                        [set temp 1]
                      ]
crear-autos
```

En el cual se verifica antes de que entre un nuevo automóvil al carril, si hay espacio si hay espacio le asigna a la variable temp 1 y si no hay espacio le asigna un 0, La variable temp será utilizada dentro del código crear- autos le indica al programa si hay espacio para el nuevo auto, una vez asignado el valor a la variable manda a llamar al procedimiento *crear-autos*.

Procedimiento crear-autos

Verifica que el semáforo este en verde (luz=1) que haya espacio en la carretera (temp=1) y que no se rebasé el número flujo máximo calculado por la distribución Normal. Si todas estas condiciones se cumplen crea un automóvil al inicio de la calle [*create-turtles 1 [setxy min-pxcor 0]*] y manda llamar el procedimiento inicializar automóvil para que le de la velocidad de inicial y su posición velocidad máxima y mínima

```
to crear-autos
  if (luz = 1) and (contador > 1) and (temp = 1)
    [create-turtles 1 [setxy min-pxcor 0]
     ask turtles[inicializar_autos]
     set contador contador - 1
    ]
End
```



Graficas

Procedimiento autos-semáforo-grafica

Manda a llamar al elemento de la interfaz llamado "*flujo de entrada*" y envía a que pinte el número de autos que entraron mientras el semáforo estaba en verde. También manda a llamar el Plot llamado "*histograma de distribución de entrada*" y envía a graficar el Distribución de frecuencias de entrada.

```
to autos-semáforo-grafica  
  set-current-plot "flujo de entrada"  
  plot int número-autos
```

```
set-current-plot " Distribución de frecuencias de entrada"  
  histogram contador_distribuciones  
end
```

Procedimiento autos-grafica

Cada vez que el semáforo se pone en rojo grafica el número de automóviles que se quedan esperando en la cola, envía el contenido de *autos_calle* al plot llamado "*No. autos en la cola de espera*"

```
to autos-grafica  
  set-current-plot "No. autos en la cola de espera"  
  plot autos_calle  
end
```

Procedimiento histograma-distancia

Grafica en el plot llamado "*Distancia entre automóviles*" un histograma de la distancia que hay entre cada automóvil y el siguiente.



```
to histograma-distancia
  set-current-plot "Distancia entre automóviles"
  histogram [distancia] of turtles
end
```

Procedimiento *histograma- velocidades*

Grafica en el plot llamado "*Histograma de velocidades*" un histograma de las velocidades de todos los automóviles que se encuentran en la calle.

La velocidad máxima fue definida con el número 1 por lo tanto si mandamos a graficar directamente la velocidad de los automóviles directamente, ningún automóvil tendrá una velocidad mayor que , así que para tener la velocidad en términos de 1 a 60 kilómetros por hora es necesario que la velocidad de los automóviles sea multiplicada por 60.

```
to histograma-velocidades
  set-current-plot "Histograma de velocidades"
  histogram [velocidad * 60] of turtles

end
```

Procedimiento *plot_densidad*

Grafica en el plot llamado "*densidad*" la densidad de automóviles que hay en ese momento en la calle.

```
to plot_densidad
  set densidad (count turtles with [xcor <= 1] / 27)
  set-current-plot "Densidad"
  set-current-plot-pen "densidad"
  plot densidad
end
```



La figura 5.1 muestra la pantalla principal de nuestra simulación

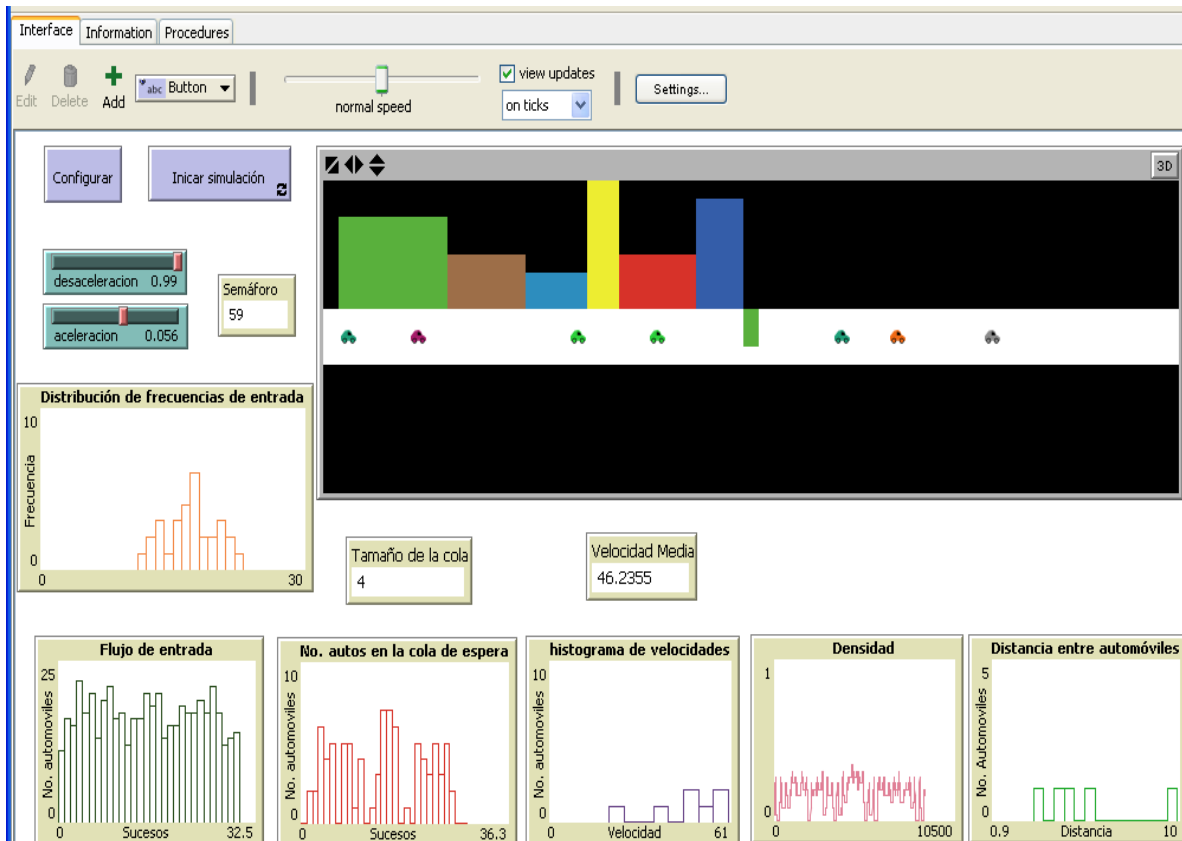


Figura 5.1 ventana principal de la simulación

Algunos elementos de la interfaz se programan directamente sobre la interfaz del programa

Botón iniciar simulación

El botón Iniciar simulación figura 4.2 es programado para que llame al procedimiento llamado *go* y lo repite hasta que se vuelva a dar clic sobre el botón

Monitor semáforo figura 4.4 Nos muestra el número de cambio de semáforo que va hasta el momento en la simulación. Se programa para que reporte la variable *num_semaforo* figura 5.2

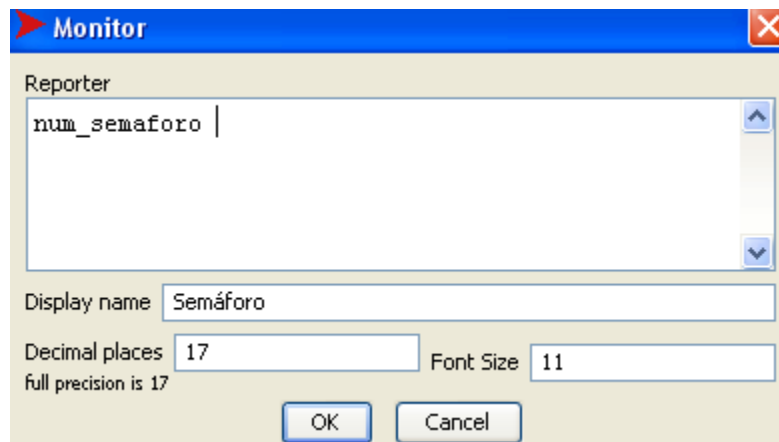


Figura 5.2 Programación del monitor Semáforo

Monitor tamaño de la cola

Este monitor cuenta cuantos automóviles están esperando antes del semáforo la siguiente instrucción lleva a cabo el conteo *count turtles with [xcor <= 1]*

Monitor velocidad media

Este monitor calcula la velocidad media de todos los automóviles en la simulación las instrucciones que realiza para este cálculo son las siguientes:

$$(sum [velocidad] of turtles) / (count turtles)$$

5.5 Validación

Una vez terminada la programación se prosigue con la validación en esta etapa se comprueba que nuestra simulación cumpla con el modelo establecido, comparamos los resultados obtenidos en la simulación con los resultados esperados.

Esto se lleva a cabo comparando los resultados del modelo en la simulación con: mediciones realizadas en el sistema real.

Lo primero es verificar que el histograma de distribuciones de entrada cumpla con la distribución Normal que formaban los que obtuvimos en el trabajo de campo.



La figura 5.3 nos muestra el histograma de distribución de frecuencia de entradas generado en la simulación del sistema



Figura 5.3 Histograma de frecuencias

La grafica de la figura 5.4 nos muestra el histograma de distribución de frecuencias de entrada generado por los datos recolectados en el trabajo de campo

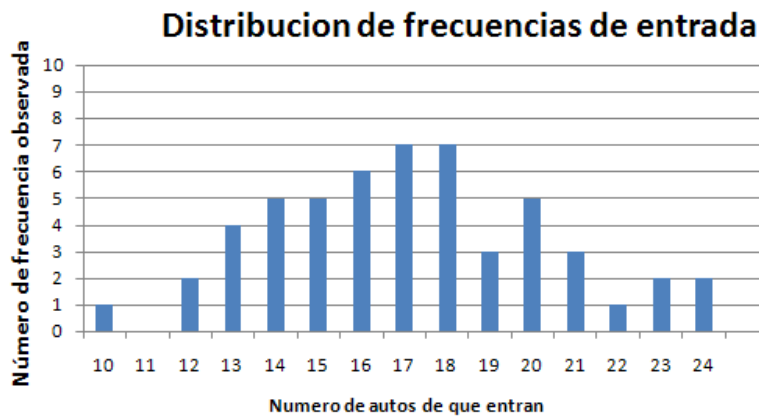


Figura 5.4 histograma de frecuencias trabajo de campo

Podemos observar claramente que la simulación cumple con los requisitos de entrada de automóviles.

El siguiente paso es verificar que el primer carro en entrar sea el primero en salir esto lo podemos verificar parando la simulación y revisando las coordenadas de los automóviles al igual que su número de identificación que nos indica cual se creo primero. Para hacer esto se tiene que dar clic derecho sobre el auto y nos muestra un menú seleccionar inspeccionar turtle como lo muestra la figura 5.5.

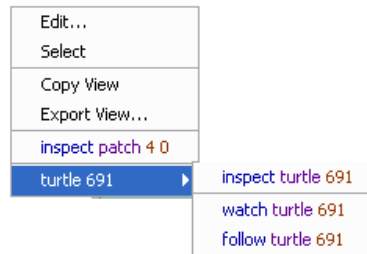


Figura 5.5 Menú desplegable Inspeccionar turtle

Cuando inspeccionamos un automóvil nos muestra todas sus características como podemos observar en la figura 5.6.

Attribute	turtle 693	turtle 689	turtle 688
who	693	689	688
color	55	135	5
heading	90	90	90
xcor	-11.10207239824945	12.512775962775745	17.864647970503047
ycor	0	0	0
shape	"car"	"car"	"car"
label	==	==	==
label-color	9.9	9.9	9.9
breed	turtles	turtles	turtles
hidden?	false	false	false
size	1	1	1
pen-size	1	1	1
pen-mode	"up"	"up"	"up"
velocidad	0.950506133243708	0.355502109131382	0.7744673312581214
velocidad-max	1	1	1
velocidad-min	0	0	0
distancia	4.284534522156635	4.772964651516104	4.851872007727302

Figura 5.6 Características de los automóviles

Por deducción nos podemos dar cuenta que al calcular la distancia entre delantero y el siguiente si en algún momento cambiaran de lugar la distancia sería negativa.

El modelo de car-following funciona correctamente, lo podemos observar con la siguiente secuencia de imágenes de la figura 5.7. Observen el automóvil azul que tiene detrás de él un automóvil morado.

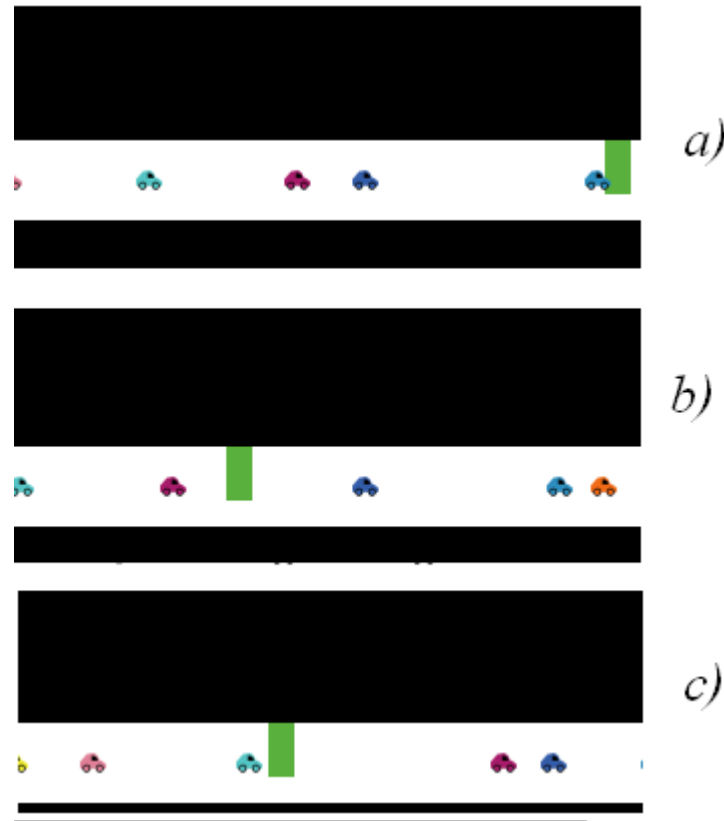


Figura 5.7 secuencia de imágenes interacción entre automóviles.

El auto azul que se encuentra en frenten reduce su velocidad, y el auto morado que lo sigue tiene que hacer lo mismo Figura 5.7 a), en la figura 5.7 b) el auto azul acelera, cuando el automóvil azul acelera el auto morado después de un tiempo de reacción también acelera Figura 5.7 c).



Conclusión

Existen varios modelos de simulación de tráfico vehicular hay algunos muy complejos que nos llevan a soluciones analíticas complicadas y otros muy simples y eficientes como lo es el Modelo Car-Following que describe el complejo comportamiento del fenómeno de tráfico vehicular a partir del comportamiento básico de sus componentes, definiendo este complejo fenómeno desde la interacción entre dos automóviles.

Se crea una interfaz grafica para visualizar diferentes aspectos del tráfico vehicular con semáforo, la herramienta utilizada fue Netlogo. Se observan los automóviles moviéndose, de sur a norte acelerando y desacelerando según el estímulo que reciban del vehiculo de enfrente.

Cuando un automóvil desacelera los que lo siguen tienen que desacelerar también formando atascos o congestionamientos momentáneos que son liberados cuando el vehiculo líder vuelve a acelerar.

Al ejecutar la simulación se observan claramente lo que ocurre con el fenómeno de tráfico vehicular, cuando se varían las diversas características del fenómeno, por ejemplo si aumentamos la velocidad de aceleración se observa que es difícil que los automóviles puedan frenar y evitar las colisiones, si el flujo de entrada aumenta la densidad de tráfico se vuelve más densa. Si la velocidad de desaceleración es muy baja también pueden ocurrir coaliciones. De forma inversa si disminuimos la aceleración el tráfico se vuelve muy lento la densidad aumenta y empieza a aparecer un mayor congestionamiento.

Esta es una de las principales finalidades de la simulación el poder observar jugar con los factores que intervienen en la simulación y entender cómo evoluciona el fenómeno.

Como trabajo a futuro se podría diseñar una simulación con dos o más carriles y con varias intersecciones.



Bibliografía

- [1] Traffic and Highway Engineering , Garber, Nicholas J. Boston: PWS publishing company; International Thomson, c1997
- [2] Principles of mathematical modeling C.L. Dym E.S Ivey academic Press 1980
- [3] an analytical shock-fitting approach to the solution of the Lighthill-Whitham-Richards traffic flow model, CHEN WENQIN, Department of Civil Engineering, The University of Hong Kong
- [4] es.wikipedia.org/wiki/Modelo_Nagel-Schreckenberg
- [5] State-of-the-art of Vehicular Traffic Flow Modeling, Serge P. Hoogendoorn and Piet H.L. Bov, Special Issue on Road Traffic Modeling and Control of the Journal of Systems and Control Engineering.
- [6] Rafael Cal y Mayor, James Cárdenas Grisales. Ingeniería de tránsito fundamentos y aplicaciones 7ª edición editorial Alfa omega
- [7] Enrique Eduardo tarifa, Teoría de modelos y simulación
- [8] Dr. Abdiel Emilio Cáceres González Sistemas complejos basados en multi-agentes Universidad Juárez Autónoma de Tabasco
- [9] Rodrigo González Rivera, “Calibración del Micro simulador AIMSUN para flujo Ininterrumpido en la ciudad de Concepción”, octubre del 2005.