

## Herramienta computacional para sugerir y validar políticas de luces de semáforos

Orlando Fonseca Guilarte<sup>1</sup>, Gabriel S. Pujol Fariña<sup>2</sup>, Sira M. Allende Alonso<sup>3</sup> y Gemayqzel Bouza Allende<sup>3</sup>

<sup>1</sup> Departamento de Matemática. Pontificia Universidade Católica do Rio de Janeiro  
ofonsek0702@mat.puc-rio.br  
<http://www.mat.puc-rio.br/index.htm>

<sup>2</sup> Departamento de Informática. Pontificia Universidade Católica do Rio de Janeiro  
gfarina@inf.puc-rio.br  
<http://www.inf.puc-rio.br/>

<sup>3</sup> Departamento de Matemática Aplicada. Facultad de Matemática y Computación.  
Universidad de La Habana  
{sira, gema}@matcom.uh.cu  
<http://www.uh.cu/>

**Resumen.** En una intersección urbana controlada por semáforos es importante determinar cuándo cambiar las luces de forma tal que se minimicen los valores de los indicadores relativos a la congestión del tráfico. Dado un cruce de calles arbitrario del que se conoce el orden de las luces de los semáforos y la tasa media de arribos y de salidas de los autos, en este trabajo se presenta una herramienta para sugerir y evaluar políticas de cambio de tiempos de luces. La herramienta estudia estadísticamente el flujo de tráfico y evalúa la política de luces usando distintas medidas de congestión. Para proponer una estrategia resuelve, utilizando algoritmos metaheurísticos de tipo Recocido Simulado, el problema de optimización, mono-objetivo o multi-objetivo resultante de minimizar una o varias medidas de congestión. Mediante una simulación se evalúa la solución propuesta.

### 1 Introducción

Con el aumento del tamaño de la población en las ciudades, el creciente avance de la economía y el mejoramiento del nivel de vida, el número de vehículos que circula por las calles es significativamente elevado. La población tiende a desplazarse a las ciudades y esto conlleva a que muchas de las urbes del mundo, como Beijing, Nueva Delhi y Sao Paulo, muestren altos indicadores de flujo de tráfico, y por ende, se le preste especial atención a este problema, ver [16]. La circulación de vehículos en las vías causa daños al medio ambiente, a la economía y la salud humana. Inicialmente se intentó afrontar esta situación construyendo nuevas vías y estimulando el uso del transporte público. Sin embargo, estas acciones no fueron suficientes. Quedaban muchos puntos de conflicto, en particular en los lugares donde confluyen varias calles y el flujo de tráfico está regulado por semáforos: las intersecciones semaforizadas.

Los semáforos se utilizan para controlar la circulación de los vehículos y los peatones. Para ello muestran indicadores de luces de color rojo, amarillo y verde, que se repite cíclicamente. De acuerdo a [9], con los semáforos se busca un equilibrio entre la movilidad, accesibilidad y la seguridad en las vías. Como se reporta en [11], es importante contar con una buena política de control de semáforos en las intersecciones con el fin de resolver los problemas de congestión de tráfico.

Debido a la incertidumbre inherente al comportamiento del tráfico, probar en la práctica varias políticas de cambios de luces y comparar cómo se comporta el tránsito con ellas es costoso. Para ello, es necesario tener una herramienta que sugiera y valide políticas para minimizar la congestión vehicular en las intersecciones. Esta herramienta debe permitir que la comparación de estrategias sea de forma rápida y económica.

Sea una intersección en la que confluyen un determinado número de calles, en cada una de las cuales hay semáforos cuyas combinaciones de luces definen, en cada momento, los movimientos permitidos. El objetivo de este trabajo es presentar una propuesta de solución al Problema de Regulación y Control Óptimo de Semáforos (PRCOS), o sea, determinar la duración adecuada de cada una de estas combinaciones para minimizar la congestión del tráfico.

El Problema de Regulación y Control Óptimo de Semáforos ha sido modelado matemáticamente de distintas formas y para su solución se han implementado algoritmos de programación en enteros mixta y varios tipos de métodos metaheurísticos. A continuación se presentan algunos ejemplos: En [7] se modela el problema de control usando teoría de juegos y se propone un procedimiento para la búsqueda de un equilibrio de Nash. Por otra parte [4], [5] consideran el problema de determinar secuencias de tiempo de cambio de luces que minimice una medida de congestión y lo describe como un modelo de optimización con restricciones de complementariedad lineal extendida. Este problema también se aborda en [1], donde para su solución se propone un algoritmo inspirado en Recocido Simulado. Existen también modelos computacionales que describen cómo determinar políticas de luces óptimas. En [8], [12] se estudia el problema usando técnicas de redes neuronales como son las redes de Petri y reglas difusas en una Red Neuronal. El flujo en una intersección se modela en [13], [14] con un autómata celular y se minimiza a partir de un Algoritmo Genético. Con la teoría difusa de control, [6] provee un sistema de control distribuido, el cual contiene algunos controladores difusos y un controlador de casos especiales. En este trabajo se optimiza la extensión de la luz verde utilizando la metaheurística Recocido Simulado. Como se reporta en [10], la planificación óptima para una intersección puede utilizar cámaras y procesadores de imágenes instalados en el cruce y se aplican reglas difusas para obtener dicha planificación.

En la literatura consultada, en el campo de la programación matemática, solo se consideran las tasas de arribo y salida constantes en todo el estudio. Queda abierta la pregunta de cómo incorporar la aleatoriedad, por lo que resulta interesante la creación de un simulador de tráfico para este fin en el problema de minimizar la congestión. El simulador puede no solo apoyar la evaluación de un criterio, si no también validar la solución hallada comparándola con otras. Para esto será necesario disponer de una herramienta que describa el tráfico de una manera rápida, con un mínimo de recursos a diferencia de las existentes en la literatura, para los cuales cámaras y sensores son imprescindibles.

Este artículo está estructurado en ocho secciones. En la próxima sección se modela el problema y se generaliza el modelo presentado por [4] al caso de una intersección de  $n$  calles con  $k$  combinaciones de luces en los semáforos. El diseño del procedimiento que se propone para representar la dinámica y estocasticidad del problema y algunas alternativas de solución se describen en las secciones 3 y 4 respectivamente. La implementación computacional es el contenido de la quinta sección. La verificación experimental de los algoritmos y el análisis de los resultados se incluyen en la sección 6. Las conclusiones y las referencias bibliográficas completan el trabajo.

## 2 Formulación del PRCOS como un modelo de optimización

En esta sección se describe la topografía de la intersección y se precisan las principales definiciones de tráfico, así como el modelo matemático del problema.

Con respecto a la forma en la que funciona el semáforo se tienen los siguientes conceptos:

Definición 1: Una fase es una combinación de luces para la cual una serie de semáforos de la intersección permanecen con las luces verde y amarilla, y el resto con la luz roja.

Definición 2: Una estructura de fase es la descripción de los movimientos permitidos a cada carril de la intersección en cada una de las fases.

Definición 3: Se dice duración de una fase al tiempo en que los semáforos de una intersección permanecen con la luz correspondiente a la misma (roja o verde-amarilla).

Definición 4: Un ciclo es la combinación de diferentes fases de los semáforos. La duración del ciclo puede ser constante o variable.

Para evaluar una estrategia de tiempo de luces es necesario definir una medida de congestión del cruce. Para este objetivo se tiene la siguiente definición:

Definición 5: Se dice tasa media de llegada del carril  $j$  a la cantidad esperada de autos que llegan al carril  $j$  en cada segundo. Se denota por  $\lambda_j$ . La tasa media de salida del carril  $j$  es la cantidad esperada de autos que salen del carril  $j$  en cada segundo. Se denota por  $\mu_j$  ( $\kappa_j$ ) si la luz verde (amarilla) está encendida para ese carril.

Los valores de las tasas  $\lambda_j$ ,  $\mu_j$  y  $\kappa_j$  pueden variar por intervalos. Se tomarán, como medida de congestión, indicadores definidos en términos de las longitudes de las colas de cada carril, y estos dependen de la duración de las fases y de  $\lambda_j$ ,  $\mu_j$  y  $\kappa_j$ .

El modelo propuesto generaliza el dado en [4] al considerar cualquier número de carriles, fases y ciclos. Por tanto se tienen los siguientes parámetros:

- $N$ : cantidad de ciclos.
- $I$ : cantidad de fases.
- $\delta_{ama}$ : duración de la luz amarilla.
- $\delta_{min.verde,i}$ ,  $\delta_{max.verde,i}$  : cotas mínima y máxima de duración de la luz en la fase  $i$ :  $i = 1, \dots, I$ .
- $J$ : cantidad de carriles.
- $\lambda_j$ ,  $\mu_j$ ,  $\kappa_j$

A continuación se definen las variables del sistema. Como variable de control se tiene:

- $\delta_{ki}$  : tiempo de luz para el ciclo k y la fase i

Y las variables de estado son:

- $X_{kji}$ : cantidad de autos que quedaron en el ciclo k del carril j al final de la fase i.

Para medir las entradas y las salidas del sistema se definen los siguientes elementos:

$$\alpha_{ij} = \begin{cases} 1, & \text{si en el carril j se permite el paso en la fase i} \\ 0, & \text{en otro caso} \end{cases}$$

$\delta_{ki} * \lambda_j$  : cantidad de autos que llegan durante el ciclo k del carril j en la fase i.

$\alpha_{ij}[(\delta_{ki} - \delta_{ama}) * \mu_j + \delta_{ama} * \kappa_j]$  : cantidad de autos que salen durante el ciclo k del carril j en la fase i.

Con estos datos se obtiene la restricción:

$$X_{kji} = \max(X_{kji-1} + \delta_{ki} * \lambda_j - \alpha_{ij}[(\delta_{ki} - \delta_{ama}) * \mu_j + \delta_{ama} * \kappa_j], 0)$$

que describe el número de autos al final de la fase.

Sumando y restando  $\delta_{ama} \lambda_j$  se obtiene:

$$X_{kji} = \max(\max(X_{kji-1} + (\delta_{ki} - \delta_{ama}) * (\lambda_j - \alpha_{ij} * \mu_j), 0) + \delta_{ama}(\lambda_j - \alpha_{ij} * \kappa_j), 0)$$

Si se definen los siguientes parámetros:

$b_{ij}^1 = (\lambda_j - \alpha_{ij} * \mu_j)$  : diferencia entre los autos que llegan y sale en un carril por segundo en la fase.

$b_{ij}^2 = \alpha_{ij} * \delta_{ama} (\mu_j - \kappa_j)$  : diferencia entre la cantidad de autos que cruzarían la intersección con la luz verde y la cantidad de autos que lo harían con la luz amarilla, cuando el semáforo tiene esta luz.

Sumando y restando  $\delta_{ama} * b_{ij}^1 + b_{ij}^2$ , se obtiene:

$$X_{kji} = \max(\max(X_{kji-1} + \delta_{ki} * b_{ij}^1 + b_{ij}^2, \delta_{ama} * b_{ij}^1 + b_{ij}^2) + \delta_{ama} (\lambda_j - \alpha_{ij} * \kappa_j) - \delta_{ama} * b_{ij}^1 - b_{ij}^2, 0)$$

Sustituyendo los valores de las constantes  $b_{ij}^1, b_{ij}^2$  se tiene que:

$$\delta_{ama} (\lambda_j - \alpha_{ij} * \kappa_j) - \delta_{ama} * b_{ij}^1 - b_{ij}^2 = 0$$

Luego, sustituyendo:

$$X_{kji} = \max(\max(X_{kji-1} + \delta_{ki} b_{ij}^1 + b_{ij}^2, \delta_{ama} (\lambda_j - \alpha_{ij} * \kappa_j)), 0) \quad (1)$$

Finalmente sea:

$b_{ij}^3 = \max(\delta_{ama} * (\lambda_j - \alpha_{ij} * \kappa_j), 0)$  : parámetro que determina la cantidad de autos que quedaron en la intersección al final de la luz amarilla.

Como  $b_{ij}^3 \geq 0$ , si la fase i del ciclo k tiene una duración  $\delta_{ki}$ , la ecuación (1) se puede escribir de forma equivalente como la siguiente restricción:

$$X_{kji} = \max(X_{kji-1} + \delta_{ki} * b_{ij}^1 + b_{ij}^2, b_{ij}^3) \quad (2)$$

La igualdad anterior puede escribirse en la forma:

$$\begin{aligned} X_{kji} - X_{kji-1} - \delta_{ki} * b_{ij}^1 - b_{ij}^2 &\geq 0 \\ X_{kji} - b_{ij}^3 &\geq 0 \\ (X_{kji} - X_{kji-1} - \delta_{ki} * b_{ij}^1 - b_{ij}^2) * (X_{kji} - b_{ij}^3) &= 0 \end{aligned} \quad (3)$$

Se trata de un problema de complementariedad lineal extendido.

Si bien las tasas de arribo se han asumido constantes, podría considerarse su variación por intervalos y así expresarse como una función que depende del tiempo, constante a trozos.

En el período se describen los siguientes indicadores para cada carril:

Longitud media de cola al final de las fases:

$$\bar{C}_j = \frac{\sum_{k=1}^N \sum_{i=1}^I X_{kij}}{T}$$

donde T es la longitud del intervalo de tiempo considerado para el estudio.

Longitud máxima de cola:

$$C_j^{\max} = \max_{k,i} (X_{kij})$$

Si los carriles difieren en importancia, estos criterios se pueden ponderar usando valores  $w_j$ ,  $j = 1, \dots, J$ ,  $w_j > 0$ . Los indicadores antes descritos definen las siguientes medidas de congestión de la intersección:

Suma ponderada de las longitudes medias o máximas de las colas:

$$\bar{F}_1 = \sum_{j=1}^J w_j * C_j, \text{ donde } C_j = \bar{C}_j \text{ o } C_j^{\max}.$$

Longitud ponderada de la cola más larga:

$$\bar{F}_2 = \max_j (w_j * C_j^{\max})$$

La selección de los tiempos de fases que minimice la medida de congestión fijada es el objetivo del trabajo. Por tanto, el PRCOS puede representarse como:

$$\text{minimizar } \bar{F}, \quad \text{donde } \bar{F} = \bar{F}_1, \text{ o } \bar{F}_2 \quad (4)$$

Sujeto a:

$$\begin{aligned} X_{kji} &= \max(X_{kji-1} + \delta_{ki} * b_{ij}^1 + b_{ij}^2, b_{ij}^3) \\ \delta_{ki} - \delta_{ama} &\in [\delta_{\min.\text{verde},i}, \delta_{\max.\text{verde},i}] \end{aligned}$$

Podría ser de interés considerar dos o más medidas de congestión del cruce. Una opción es tomar las funciones  $\bar{F}_1$  y  $\bar{F}_2$ . Otra, es el análisis diferenciado de la congestión en cada carril usando para ello los indicadores antes definidos. Ambos son problemas multi-objetivos y aunque el segundo modelo es más complejo pues tiene más medidas, será el que se considerará pues permite fijar estrategias viales diferenciadas. El modelo resultante es:

$$\begin{aligned} \min C_1 \\ \min C_2 \\ \vdots \\ \min C_j \end{aligned}$$

Sujeto a:

$$X_{kji} = \max(X_{kji-1} + \delta_{ki} * b_{ij}^1 + b_{ij}^2, b_{ij}^3)$$

$$\delta_{ki} - \delta_{ama} \in [\delta_{\min.verde,i}, \delta_{\max.verde,i}]$$

donde  $C_j = \bar{C}_j$  o  $C_j^{\max}$ .

Este problema se puede resolver mediante el método de ponderaciones. Por ejemplo, para  $C_j = \bar{C}_j$ , es equivalente a (4) con  $\bar{F} = \bar{F}_1$  y  $w_j > 0$ . Otro enfoque sería generar el frente de Pareto. Se detalla a continuación el manejo de la aleatoriedad del sistema.

### 3 Procedimientos para propuesta de una estrategia de luces bajo incertidumbre

De acuerdo al carácter aleatorio de los arribos y salidas de los autos en una intersección semaforizada, es necesario considerar los distintos valores posibles. Una suposición común es describir las entradas y salidas como un proceso Poisson de nacimientos y muertes. Bajo este supuesto, la cantidad de arribos (salidas en verde y amarilla) en el carril  $j$  en un intervalo de longitud  $t$ , tiene una distribución Poisson de parámetro  $\lambda_j * t$  ( $\mu_j * t$  y  $\kappa_j * t$ ). Los tiempos entre arribos consecutivos y salidas consecutivas de la intersección, se generan por variables aleatorias con distribución exponencial de parámetro  $\lambda_j$ ,  $\mu_j$  y  $\kappa_j$  respectivamente.

Por tanto, se obtienen una cantidad infinita de escenarios y es preciso generar un subconjunto adecuado de los mismos para estimar la medida de congestión.

La cantidad de salidas que ocurren durante una fase, depende de la duración de la misma. De ahí que generar la cantidad de arribos y salidas que ocurren en un intervalo implica conocer la duración de la fase a priori. Es por esto que solo se usará esta política de generación para validar una política de luces. Este procedimiento se nombrará *procedimiento arribo-salida*, y su pseudocódigo se muestra a continuación:

1. Inicio:  
Sea  $\delta^0$  la política de luces (puede ser generada aleatoriamente en el intervalo prefijado o ser sugerida por el experto) y sea  $E$  el número de escenarios.  
Calcular el número  $K$  de ciclos que le corresponde.
2. Para cada escenario  $e$ ,  $e = 1, \dots, E$  y cada  $\delta$  :  
Generar arribos  $A_{kij}$  y salidas  $D_{kij}$ ,  $k = 1, \dots, K$ ,  $i = 1, \dots, I$ ,  $j = 1, \dots, J$ .  
 $A_{kij} \sim \text{Poisson}(\lambda_j * \delta_{ki})$  y  $D_{kij} \sim \text{Poisson}(\mu_j * (\delta_{ki} - \delta_{ama})) + \text{Poisson}(\kappa_j * \delta_{ama})$   
Calcular las longitudes de cola  $X_{kij}^e(\delta)$ ,  $k = 1, \dots, K$ ,  $i = 1, \dots, I$ ,  $j = 1, \dots, J$  y el vector de medidas de congestión correspondiente al escenario  $MC^e(\delta)$ .
3. Calcular la media estimada del vector de medida de congestión:  $\frac{\sum_{e=1}^E MC^e(\delta)}{E}$

En aras de una mayor eficiencia al evaluar la función objetivo durante un algoritmo solución, sería conveniente generar escenarios válidos para cualquier tiempo de fase. Esto es posible si se determinan los momentos de arribos, los cuales serían indepen-

dientes de  $\delta$ , no así las salidas. Los tiempos entre arribos consecutivos se generan por variables aleatorias con distribución exponencial de parámetro  $\lambda_j$ . Se generan aleatoriamente los arribos y se asume de forma determinista que la cantidad de salidas es  $\mu_j(\delta_{ki} - \delta_{ama}) + k * \delta_{ama}$ . Este procedimiento se nombrará *procedimiento-arribo* (pasos 2-4), y se incorpora al algoritmo solución de la siguiente forma:

1. Inicio: Sea  $\delta = \delta^o$  la política inicial de luces y  $E$  el número de escenarios.
2. Generar  $E$  escenarios de arribos, determinando el momento en que llega el próximo auto a la intersección.
3. Para el vector  $\delta$  fijado y cada escenario  $e$ ,  $e = 1, \dots, E$ , determinar:
  - a) el número  $K$  de ciclos que le corresponde.
  - b) los arribos del escenario generado que ocurren en cada fase de cada ciclo.
  - c) las salidas, según las tasas de salida  $\mu$  y  $k$ .
4. Determinar las longitudes de cola  $X_{kij}^e(\delta)$ ,  $k = 1, \dots, K$ ,  $i = 1, \dots, I$ ,  $j = 1, \dots, J$  y el vector de medidas de congestión correspondiente  $MC^e(\delta)$ .
5. Calcular la media estimada del vector de medida de congestión:  $\frac{\sum_{e=1}^E MC^e(\delta)}{E}$
6. Explorar el conjunto de posibles vectores  $\delta$  mediante un procedimiento heurístico:
 

Si se cumple el criterio de parada de la heurística: Fin. El vector  $\delta^* = \text{argmin } MC(\delta)$  es el vector de tiempo de luces propuesto. En otro caso, sea  $\delta^{new}$  nueva solución generada por la heurística: Repetir (ir al Paso 3).

A continuación se expone cómo hacer la búsqueda de los mejores valores de  $\delta_{ki}$ .

#### 4 Alternativas de solución y su validación

En esta sección se muestran varios enfoques de solución y cómo se evalúa la calidad de una solución. Para la generación de los escenarios y el cálculo de la congestión en los procedimientos descritos anteriormente, se concibe el diseño e implementación de sendos modelos de simulación. Para la generación de las políticas de luces se diseña un modelo de simulación basado en el *procedimiento-arribo*. En la exploración de los posibles vectores de tiempo de luces se emplearán métodos metaheurísticos. Otro modelo de simulación basado en el *procedimiento arribo-salida*, se utiliza para validar las políticas de luces propuestas. Los resultados de los simuladores y de las metaheurísticas son evaluados con un análisis estadístico de los resultados.

Se considera como criterio de congestión los definidos por las funciones  $\bar{F}_1$  y  $C_j$ ,  $j = 1 \dots J$ . De esta forma se representa el PRCOS como el problema (4). Como el tráfico no se mantiene igual en todo el día, se considera una partición en sub-intervalos en los cuales los tiempos de llegada y salida de los autos son similares. De ahí que los parámetros  $\lambda_j, \mu_j, \kappa_j, j = 1 \dots J$ , se consideran funciones constantes a trozos y por tanto, es conveniente analizar el sistema en los intervalos en que dichos parámetros son constantes. Se asume la variabilidad de los ciclos de los semáforos en

dependencia de los instantes en que haya variabilidad del parámetro  $\lambda_j$ . El tiempo total de simulación queda particionado de acuerdo a esos instantes. En cada sub-intervalo de tiempo y con los parámetros que le caracteriza, se calcula el tiempo de cada fase para este mediante la aplicación de procedimientos heurísticos. El modelo resulta de ciclo constante en cada sub-intervalo y de ciclo variable en el tiempo total de simulación. Por tanto, la ejecución del *procedimiento-arribo* puede realizarse considerando el tiempo total de simulación o sobre cada sub-intervalo de la partición. Es necesario destacar que para el primer caso la cantidad de variables a considerar aumenta notablemente. En cualquiera de las dos opciones, las heurísticas empleadas en el *procedimiento-arribo* están inspiradas en la metaheurística de Recocido Simulado. Para generar puntos eficientes del problema multi-objetivo, se proponen dos enfoques: un algoritmo que resuelve el problema ponderado y la metaheurística AMOSA, ver [2].

Dado un vector  $\delta^* = (\delta_1^*, \dots, \delta_{I*S}^*)$ , se considera la vecindad  $V(\delta^*)$  definida de la siguiente forma:

$$V(\delta^*) = \{(\delta_1, \dots, \delta_{I*S}) / \exists k \in \{1, \dots, I * S\}: \delta_i = \delta_i^*, i \neq k, \delta_k \neq \delta_k^*, \delta_k \in [\delta_{min}, \delta_{max}]\}$$

Para cada punto, la vecindad en la cual se encuentran las posibles perturbaciones está dada por las soluciones factibles que pueden ser obtenidos al cambiar el valor de un elemento del vector. La posición  $k$  es generada aleatoriamente con igual probabilidad para todas las componentes, y el valor  $\delta_k$  se halla usando la distribución uniforme en el intervalo de factibilidad.

Los parámetros de temperatura inicial  $T_{max}$ , temperatura final  $T_{min}$ , esquema de enfriamiento  $\alpha$  y cantidad de repeticiones para una temperatura fija  $L$  han sido fijados con iguales valores para ambos algoritmos:  $T_{max} = 100$ ,  $T_{min} = 1$ ,  $\alpha = 0.9$ ,  $L = 10$ . Para la generación del frente de Pareto, el algoritmo basado en Recocido Simulado considera diferentes valores de los coeficientes de ponderación.

El resultado final de AMOSA es el conjunto Archivo en el que se almacena el frente de Pareto generado. Se asume que Archivo tiene como máximo  $SL = 20$  elementos. De superar este valor, se realiza el proceso de agrupación para obtener y guardar en el Archivo los objetos representativos de los  $HL = 10$  grupos.

Con ambos algoritmos se genera el frente de Pareto del problema con función objetivo  $(C_1, \dots, C_j)$  y se sugiere como estrategia de luces, el vector del frente de Pareto que tenga menor norma  $L_\infty$  (norma del supremo). Al considerar esta norma se busca minimizar la cola del carril más cogestionado del cruce.

El número de escenarios se ajusta dinámicamente estableciendo un compromiso entre la calidad de la estimación y el tiempo de cálculo del sistema computacional.

Para los modelos de simulación y las heurísticas es necesario realizar estimaciones de los valores de las variables y es importante evaluar su calidad. Este análisis incluye los cálculos de la media, la mediana, la moda, la varianza y el intervalo de confianza del estimador. De tal análisis puede concluirse la conveniencia de aumentar el número de corridas de la heurística y el número de escenarios del simulador.

El intervalo de confianza para la media estará dado por:



$$\bar{X} \pm \left[ t_{(\alpha/2, n-1)} \sqrt{(S^2/n)} \right]$$

Donde  $n$  es la muestra de corridas,  $t$  es el percentil de orden  $\alpha/2$  de la  $t$  de Student con  $n - 1$  grados de libertad y  $S^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1}$  el estimador muestral.

Este intervalo puede aplicarse en la determinación de las corridas que deben hacerse de la simulación para lograr un intervalo de confianza de longitud  $e$ . Si  $e$  es mayor que  $2 \left[ t_{(\alpha/2, n-1)} \sqrt{(S^2/n)} \right]$ , entonces son necesarias  $n - n^*$  corridas más, donde

$$n^* = \left( (t_{(\alpha/2, n-1)} * S) / e \right)^2$$

En otro caso, si  $e$  es menor que la longitud que se tenía, entonces es necesario realizar  $n^* - n$  corridas más.

Para validar las soluciones obtenidas se selecciona el *procedimiento arribo-salida*. Con los cálculos de la congestión, se realiza un análisis estadístico que permite validar la propuesta de tiempo de luces considerando todos los carriles del cruce para cada período. Si al validar la estrategia propuesta los resultados no fueran satisfactorios, estos serán tomados en cuenta para construir nuevos escenarios en los que aplicar las heurísticas generadoras de las estrategias de luces.

## 5 Herramienta computacional

La herramienta fue construida utilizando técnicas de *soft computing* e ingeniería de software, tales como heurísticas de búsqueda, modelos de simulación y orientación a objetos. Se desarrolló sobre la plataforma .NET 4.0 y el lenguaje de programación C#, utilizando la biblioteca gráfica ZedGraph, ver [3]. Sus módulos son:

**Portada:** interfaz principal con el usuario. Recibe la información de entrada del sistema. Este módulo es el encargado de procesar la información sobre características de la intersección (cantidad de carriles, estado inicial), del semáforo (cantidad de fases, estructura de fase), las llegadas y salidas de los vehículos a la intersección (tasas de arribo y salida) y el intervalo de tiempo de la simulación.

**Simulador:** maneja los eventos discretos actualizando el reloj interno y la generación de los valores de las variables aleatorias. Proporciona como salida una estructura de datos que contiene todos los valores de las variables de estado del sistema a lo largo de la simulación, destacándose las longitudes de cola al final de cada fase para cada carril. Entre sus implementaciones cuenta con las variantes Ligerero y Fuerte, que implementan el *procedimiento-arribo* y *arribo-salida* respectivamente.

**Evaluador:** ofrece funciones para el cálculo de estadígrafos descriptivos como la media, la mediana, la moda y el máximo de longitudes de cola por carril. Además incluye el cálculo de la varianza y el intervalo de confianza para los estimadores. En particular para determinar la medida de congestión asociada a una estrategia de luces se calcula la media de la medida de congestión en los  $E$  escenarios. Para evaluar la

calidad de las soluciones que ofrecen los algoritmos heurísticos programados, se requiere también de los cálculos estadísticos que aquí se implementan.

**Validador:** permite al usuario hacer validaciones experimentales de políticas de luces propuestas, las cuales pueden ser resultado de la heurística de búsqueda o la proposición de un decisor o experto. Para ello se genera un número de corridas del simulador fuerte (Módulo Simulador) y procesa estadísticamente los resultados utilizando para ello el módulo Evaluador.

**Explorador:** contiene la implementación de los algoritmos heurísticos propuestos para la generación de luces, ya sea considerando el intervalo completo de tiempo o la partición de este en sub-intervalos. Para cada solución generada construye la solución vecina y determina la aceptación o no de la misma. La evaluación de cada candidata en la solución construida es dada por el módulo Evaluador. Este módulo realiza la exploración de soluciones de forma que se minimicen los índices de congestión especificados. En particular para Recocido Simulado permite fijar los coeficientes de ponderación y de ese modo generar elementos del frente de Pareto.

**Graficador:** ilustra mediante gráficos 2-D el conjunto de valores producidos para cada carril por el estadígrafo seleccionado en el Evaluador, mostrándose los datos en cada fase, ciclo y hora. Esto permite identificar la cantidad de ciclos problemáticos y el intervalo de tiempo en el que estos tienen lugar.

**Experimentador:** ofrece la posibilidad de obtener tantas soluciones al problema se desee según las configuraciones de los módulos principales y realizar cualquier número de validaciones a estas soluciones. Se diseñó especialmente para ver el comportamiento de las metaheurísticas.

## 6 Resultados experimentales

En la figura siguiente se ilustra el comportamiento de un cruce de cuatro calles que tiene la siguiente estructura de fase: en la primera fase, los semáforos  $T_1$  y  $T_2$  de las vías  $L_1$  y  $L_2$  respectivamente, tienen luz verde para permitir el movimiento hacia adelante y girar a su derecha. Los semáforos  $T_1$  y  $T_3$  de las vías  $L_1$  y  $L_3$  permanecen con luz roja. En la segunda fase, solo el semáforo  $T_3$  de la vía  $L_3$  tienen luz verde para permitir el movimiento hacia adelante, girar a su izquierda y girar a su derecha. En la tercera fase, solo están permitidos los movimientos anteriores en la vía  $L_4$ .

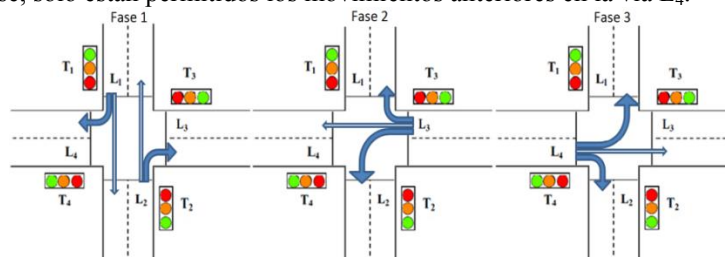


Fig. 1. Estructura de fases de un cruce de 4 calles con 3 fases.

Para el estudio se considerará el período de un día, es decir, 24 horas. Las tasas de llegada se agruparon en períodos de 3 horas, que son en donde el tráfico del cruce se

comporta de manera similar. Las tasas de salida se escogen constantes durante todo el día. Es preciso destacar que para realizar un estudio más profundo estas tasas variarán de acuerdo a la reacción de los conductores en la salida cuando hay un cambio de luz o cuando la misma está finalizando su tiempo. Los valores de los parámetros para los 4 carriles se muestran en las siguientes tablas:

**Tabla 1.** Tasas de llegada en intervalos de tres horas por carril.

	00h-03h	03h-06h	06h-09h	09h-12h	12h-15h	15h-18h	18h-21h	21h-24h
L 1	0.01388	0.05555	0.00462	0.00231	0.00231	0.00231	0.00462	0.01388
L 2	0.00231	0.00231	0.00231	0.00231	0.00231	0.00925	0.05555	0.00462
L 3	0.00231	0.00231	0.00925	0.00231	0.00462	0.02777	0.22222	0.02777
L 4	0.00231	0.00231	0.22222	0.02777	0.00231	0.00925	0.00925	0.02777

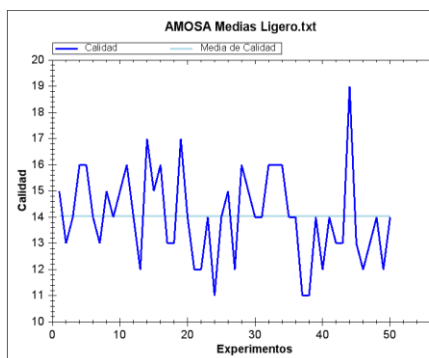
**Tabla 2.** Tasas de salida en verde.

	00h-24h
L 1	0.60000
L 2	0.50000
L 3	0.70000
L 4	0.70000

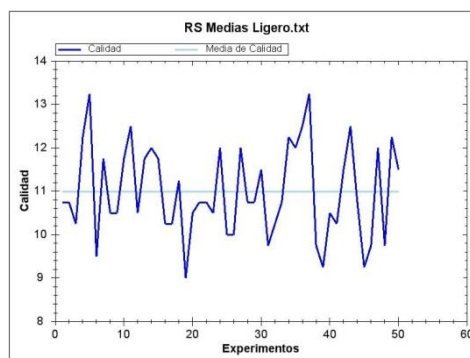
**Tabla 3.** Tasas de salida en amarillo.

	00h-24h
L 1	0.35000
L 2	0.50000
L 3	0.70000
L 4	0.70000

En los gráficos se ilustra con color azul oscuro, la calidad o *fitness* de la política de luces resultante de la aplicación de las heurísticas. Con color azul claro se muestra la media de estos valores, correspondientes a 50 corridas de la heurística especificada. El número E de escenarios para el Simulador Ligero es 30, y los mismos escenarios son utilizados para cada corrida.



**Fig. 2.** AMOSA con análisis de medias.



**Fig. 3.** Recocido Simulado con análisis de medias.

Del presente experimento se concluye que para un mismo ejemplo, para iguales parámetros y escenarios en distintas corridas, se obtienen valores de calidad similares, ya que sus valores de media son cercanos y la varianza es pequeña para cada caso. La aplicación del método de Recocido Simulado fue el que produjo los mejores valores

de media y varianza, y con el que fue posible encontrar el mejor valor óptimo. La siguiente tabla muestra los valores que avalan lo antes dicho:

AMOSA Medias Ligerito.txt		RS Medias Ligerito.txt	
Media	14,04	Media	10,995
Varianza	2,937	Varianza	1,124
Valor Óptimo	11	Valor Óptimo	9

Fig. 4. Resultados con AMOSA y Recocido Simulado.

Si se mantienen los mismos parámetros para la ejecución de las heurísticas y el número E de escenarios, pero siendo estos diferentes para cada corrida, se obtienen los gráficos que se presentan a continuación:

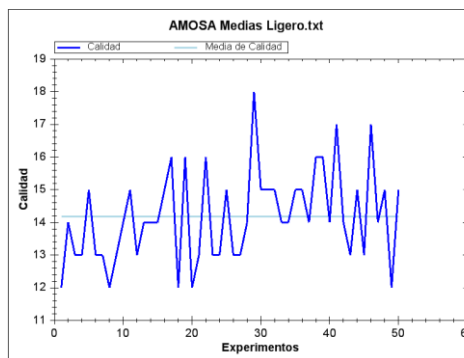


Fig. 5. AMOSA con análisis de medias.

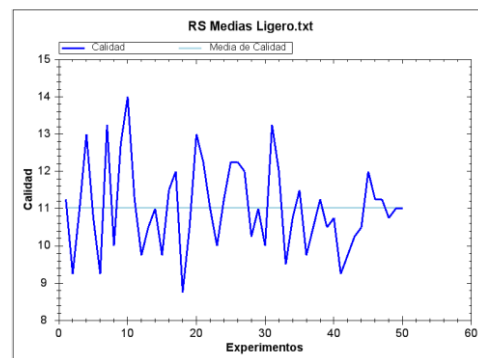


Fig. 6. Recocido Simulado con análisis de medias.

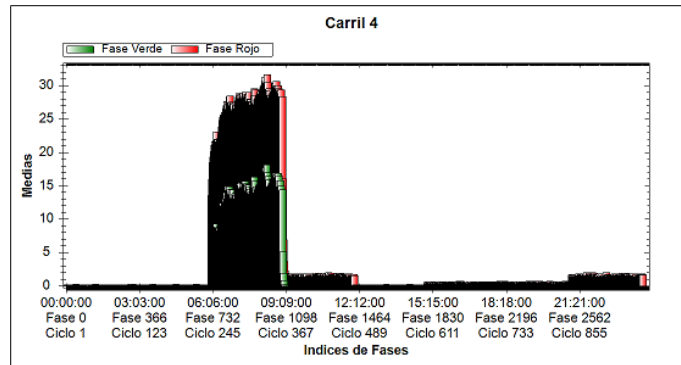
En este experimento se obtienen también valores de calidad similares. La Figura 7 muestra estos valores en cada caso. Por tanto, la aleatoriedad en el fenómeno de cambiar la realidad no afecta el funcionamiento de las heurísticas en la recomendación de políticas de luces. Recocido Simulado fue el que produjo los mejores valores de media y varianza, y con el que fue posible encontrar el mejor valor óptimo.

AMOSA Medias Ligerito.txt		RS Medias Ligerito.txt	
Media	14,18	Media	11,03
Varianza	1,987	Varianza	1,389
Valor Óptimo	12	Valor Óptimo	8,75

Fig. 7. Resultados con AMOSA y Recocido Simulado.

En el proceso de validación de una política de luces propuesta por el usuario, dada por mantener los tiempos de fases constantes e iguales a 30 en todo el intervalo de tiempo de simulación, se realizan 30 corridas del Simulador Fuerte. A continuación, los gráficos de los valores de la longitud media de cola para el carril 4 en cada intervalo de tiempo. Para todos los gráficos, la longitud de la cola durante cada fase es mostrada por la altura de las barras, y el color de la barra indica el tipo de fase ocurrida. Se

puede apreciar que la congestión a lo largo del día es baja, con excepción de la ocurrida entre las 6h y las 9h.



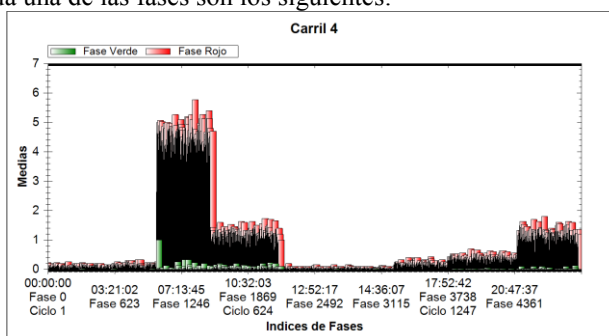
**Fig. 8.** Longitud media de cola del Carril 4.

En el proceso de sugerir y validar una política de luces, se realizan 30 corridas del Simulador Fuerte con la estrategia dada como solución óptima por el Simulador Ligero empleando Recocido Simulado. Se aplicó la modalidad de considerar todo el intervalo de simulación. Los resultados se muestran en la Tabla 4.

**Tabla 4.** Política de luces.

Política de luces	Duración - fase 1	Duración - fase 2	Duración - fase 3
00:00h-03:00h	37	10	10
03:00h-06:00h	49	10	10
06:00h-09:00h	10	10	44
09:00h-12:00h	11	30	10
12:00h-15:00h	10	10	10
15:00h-18:00h	10	10	45
18:00h-21:00h	10	30	10
21:00h-24:00h	32	10	14

Los gráficos de los valores de la longitud media de cola y varianza para el carril 4 en cada una de las fases son los siguientes:



**Fig. 9.** Longitud media de cola del carril 4.

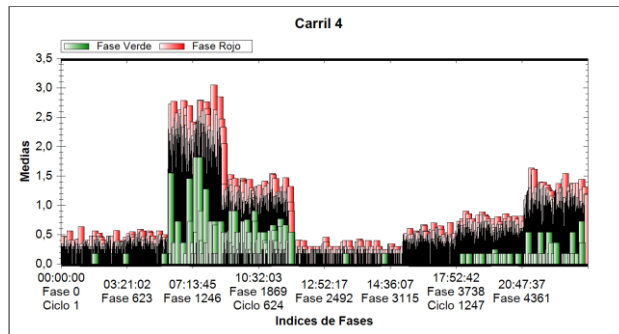


Fig. 10. Varianza de la longitud media de cola del Carril 4.

Con esta política de luces recomendada por la herramienta, se observa que en el carril 4, que se tenía aproximadamente 30 como longitud media de cola, disminuyó este valor hasta aproximadamente 5 en el horario de 6 a 9. Este mismo análisis puede realizarse en los demás carriles del cruce, los cuales también redujeron sus colas: el 1 y el 2, disminuyeron su longitud media de cola de 5 vehículos (con la política de luces del usuario) a 1,5 y 4 en el horario de 3 a 6 y de 18 a 21 respectivamente. El carril 3 que tenía aproximadamente 30 como longitud media de cola, disminuyó este valor hasta 5 en el horario de 18 a 21.

## 7 Conclusiones

Con esta herramienta se puede observar las consecuencias de la variación de los parámetros tales como las tasas de llegada y salida y la duración de las fases, permitiendo analizar la fluidez del tráfico y comprobar la eficacia de las nuevas políticas de control de una forma barata y rápida, siendo el manejo de la herramienta muy fácil para el usuario, en donde se le da la opción de validar una política de luces cualquiera o simplemente aceptar una recomendación que el mismo sistema ofrece mediante el uso de metaheurísticas.

El algoritmo inspirado en Recocido Simulado mostró mejores resultados que AMOSA en la recomendación de políticas de luces. Es necesario tener en cuenta que cuando se comparan estos resultados con situaciones de la vida real, los errores de medición pueden aparecer, pero con datos más precisos, las mejoras son posibles.

Como líneas futuras se recomienda recopilar y experimentar con datos reales de flujo en una intersección dada, considerar otros procesos de arribo y variar las tasas de salida para representar mejor en el modelo la dinámica real del tráfico. Se sugiere además complementar los resultados del trabajo con el uso de técnicas de minería de datos para reconocer los períodos críticos y evaluar estrategias para la incorporación al estudio de intersecciones consecutivas pertenecientes a una red donde se tenga en cuenta la sincronización entre ellas (ola verde).

## Referencias

1. Allende, S., Lema, C. S., Pedreira, L. P., Bouza, G.: Un método híbrido para resolver el problema de control óptimo de semáforos. (2009) 10(1):285–298
2. Bandyopadhyay, S.: A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA. IEEE transactions on evolutionary computation, Vol. 12, No. 3. (2008) 269-283
3. Champion, J.: A flexible charting library for .NET. [en línea]. Code project, for those who code, 2007. [Consultado el 19 de marzo de 2014]. Disponible en <http://www.codeproject.com/Articles/5431/A-flexible-charting-library-for-NET>
4. De Schutter, B., De Moor, B.: Optimal traffic light control for a single intersection. European Journal of Control. ( 1998) 4(3):260–273
5. De Schutter, B.: Optimizing acyclic traffic signal switching sequences through an extended linear complementarity problem formulation. European Journal of Operational Research. (2002) 139(2):400–415
6. Haibo, M., Jianning, Y., Linzhong, L.: Traffic signals control of urban traffic intersections group based on fuzzy control. In 7th International Conference on Fuzzy Systems and Knowledge Discovery. (2010) 763–767
7. Huijberts H.J.C.: Routing control of a motorway network: A summary. Proceedings of the 3rd European Control Conference. (1995) 781-784
8. Li, L., Tang, N., Mu, X., Shi, F.: Implementation of traffic lights control based on Petri nets. In The 7th IEEE International Conference on Intelligent Transportations Systems. (2004) 3:1749–1752
9. Lucena, C J.: Modernización de la semaforización en Colombia, un reto en los planes de movilidad. Revista Digital Apuntes de Investigación. (2014).
10. Motameni, H., Barzegar, B., Salmanpour, S.: Modeling of fuzzy control mechanism of Intersection by using Colored Petri Net in the VANET in order to reduce urban traffic congestion. In Technical Journal of Engineering and Applied Sciences. (2013)
11. Nagatani, T.: Dispersion and scaling of fluctuating vehicles through a sequence of traffic lights. (2006) 361(2):619–629
12. Royani, T., Haddadnia, J., Alipoor, M.: Control of Traffic Light in Isolated Intersections Using Fuzzy Neural Network and Genetic Algorithm. In International Journal of Computer and Electrical Engineering. Vol. 5, No. 1. (2013)
13. Sánchez, J., Galán, M., Rubio, E.: Genetic algorithms and cellular automata: A new architecture for traffic light cycles optimization. In Proceedings of the Congress on Evolutionary Computation 2004 (2004) 2:1668–1674
14. Sánchez, J., Galán, M., Rubio, E.: Applying a traffic lights evolutionary optimization technique to a real case: Las Ramblas. IEEE Transactions on Evolutionary Computation. (2008) 12(1):25–40
15. Troschuetz, S.: NET random number generators and distributions. [en línea]. 2007. [Consultado el 5 de enero de 2014]. Disponible en <http://www.codeproject.com/Articles/15102/NET-random-number-generators-and-distributions>
16. Wang, X.: Simulation System for Optimizing Urban Traffic Network Based on Multi-scale Fusion. International Journal of Smart Home. (2014) 227-236