

# Modelado y Simulación de Escenarios de Red Supervisado por Agentes Inteligentes para el Mejoramiento de la Calidad de Servicio

José Antonio Careaga Euán  
*Ingenieria en Telematica*  
*Universidad del Caribe*  
*korntony86@gmail.com*

Adonaí Castañeda Espinoza  
*Ingenieria en Telematica*  
*Universidad del Caribe*  
*adonahi16@hotmail.com*

Candelaria Sansores  
*Ingenieria en Telematica*  
*Universidad del Caribe*  
*csansores@ucaribe.edu.mx*

3 diciembre 2009

Palabras Clave: Tráfico de redes, Tráfico Auto-Similar, Parámetro de Hurst, QoS, Poisson, Gauss, Simuladores de Redes, Agentes Inteligentes

## Resumen

En este trabajo, se diseñó y construyó un escenario de simulación de redes datos con 50 nodos que transmiten tráfico con una distribución del tipo Pareto a través de un enlace cuello de botella, donde se comprobo que el tráfico generado tiene un comportamiento muy parecido al real, es decir, el tráfico generado tiene un comportamiento auto-similar. A este escenario se le agregaron agentes inteligentes, los cuales calculan el nivel de auto-similitud (parámetro de Hurst) en el tráfico por intervalos de tiempo, interpretan este parámetro, toman una decisión conjuntamente y proponen una opción de balanceo del tráfico en la red.

## 1. Introducción.

La primera red interconectada nace el 21 de noviembre de 1969, cuando se crea el primer enlace entre las universidades de UCLA y Stanford por medio de la línea telefónica conmutada. Esta red fue el producto inicial de ingenieros creativos y científicos informáticos quienes impulsaron la internet a terrenos inimaginables, logrando una evolución constante, interconectando millones (y muy pronto billones) de computadoras, proporcionando una comunicación global, almacenamiento e infraestructura computacional[5].

La tendencia actual de adoptar a la Internet como el medio universal de transporte para todo tipo de información, tanto para redes fijas como móviles, hace surgir la necesidad de proporcionar una Calidad de Servicio (QoS: Quality of Service) adecuada a los requerimientos de cada aplicación como voz, video, e-mail, etc., los cuales tienen muy variadas exigencias como retardo máximo, pérdida de paquetes, consumo de ancho de banda, etc. Con estas exigencias de calidad de servicio, surgió la necesidad de comprender el tráfico actual de las redes. Para esto, se han realizado estudios buscando un modelo matemático que se ajuste a los comportamientos observados en las redes reales. Uno de los modelos que se ha utilizado es la distribución de Poisson, pero en la actualidad este no han demostrado el suplir las expectativas pues las redes cada día, tienen comportamientos fractales. Es por esto que en este trabajo se ha manejado un modelo con procesos autosimilares, ya que este modelo matemático ha tenido una mayor precisión para el tráfico actual.

Una de las técnicas para realizar el estudio del tráfico en las redes de datos es la simulación de éstas, a través de escenarios artificiales. Sin embargo, modelar una red real no es sencillo, dado que se necesitan ciertas configuraciones en los escenarios de cada nodo. Por este motivo, se propone el uso de un escenario cuya su configuración y topología permitan que el tráfico generado sea lo más parecido posible al tráfico real. Por último, se emplean agentes inteligentes internamente en el escenario anteriormente mencionado, los cuales evalúan el tráfico generado y ofrecen conjuntamente una alternativa de balanceo del tráfico, para así mejorar la calidad de servicio de la red de datos.

## 2. Simulacion de escenarios de redes de datos.

La simulación de escenarios de redes tiene varios objetivos, entre ellos están la enseñanza, la obtención de conocimientos sobre protocolos, topologías, el uso de aplicaciones, etc. Uno de los objetivos más importantes es la investigación para el desarrollo de nuevos dispositivos, protocolos, estándares, etc. Es así, que en este trabajo se necesita de un software que pueda simular de manera estable y confiable escenarios de redes, que nos permitiera hacer un análisis del tráfico generado en estos escenarios y que fuera software libre.

OPNET	OMNET++	NS2
<b>Lenguaje</b>		
C/C++	C++	C++ y OTcl
<b>Plataforma</b>		
Linux	Cualquiera	Cualquiera
<b>Licencia</b>		
Venta y gratuita	Venta y gratuita	Gratuita
<b>Graficos</b>		
Si	Si	No

Cuadro 1: Comparacion de los mas importantes simuladores de escenarios de redes.

En la tabla 2 se pueden observar los tres simuladores que fueron evaluados. Primero se encuentra el OPNET, este software está hecho en c++, tiene la capacidad de poder agregarle tu propio código, funciona para cualquier plataforma. OPNET cuenta con una versión gratuita que no contiene todas las características de la versión completa. Funciona con una interfaz grafica muy amigable que consta de una barra de herramienta que contiene objetos (routers, switches, nodos, etc) y solo es cuestión de arrastrarlos al area de trabajo para crear la topologia que se piensa estudiar y configurar los objetos agregados de acuerdo a las necesidades del usuario..

El siguiente simulador se llama OMNET++, este software como el anterior cuenta con una versión comercial y una versión libre. Es un simulador de eventos discretos que no está orientado específicamente a las redes, sin embargo, es muy utilizado para estudiar el comportamiento de redes wire-less. Este software es compatible tanto con sistemas Linux como con sistemas

Windows. Cuenta con una interfaz grafica la cual facilita el diseño de simulaciones asi como el análisis de los resultados.

Por último, tenemos al Network Simulator 2 o mejor conocido como NS2. Este simulador funciona en cualquier plataforma ya sea Linux, Windows o Mac OSx. Utiliza 2 lenguajes de programación, uno para la funcionalidad del sistema (c++) y el otro para el diseño de simulaciones (OTcl/Tcl). Permite la modificación o agregación de código del sistema para la implementacion de nuevas funcionalidades. La licencia de este simulador es gratuita. Este software no cuenta con interfaz grafica, pero trabaja en conjunto con otro software llamado NAM que se encarga de traducir el modelo de la simulación a un entorno visual más amigable.

Para la elección del software que usamos en este trabajo, se consult

## **2.1. Escenario de simulacion de redes reales**

Con la ayuda de esta herramienta, evaluamos un modelo propuesto en [6] donde muestran un escenario de simulación de redes que genera tráfico con un comportamiento autosimilar o parecido al real, ya que simular el comportamiento del tráfico real no es nada sencillo, de hecho es un poco complejo. Ademas, este artículo es muy referenciado por la comunidad cientifica (mas de 250 referencias) por lo que podemos decir que el escenario propuesto en este trabajo ha dado muy buenos resultados en la simulacion de tráfico real.

En el artículo, mencionan el escenario de simulacion con las siguientes configuraciones:

- Topologia con una configuración de cuello de botella (Figura 1), formada por dos gateways, con una latencia de 15ms y un ancho de banda que va desde 1.5Mbps hasta 155.52Mbps (Segun el estandar SONET).
- 2 servidores transimitiendo tráfico del tipo Pareto a 32 nodos.
- Los demas enlaces tienen un ancho de banda de 10Mbps y una latencia de 15ms.

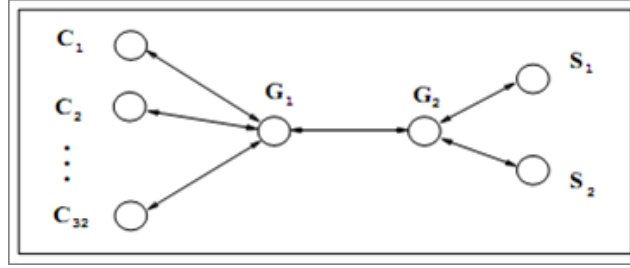


Figura 1: Topología cuello de botella.

## 2.2. Tráfico auto-similar o real.

Para que el tráfico generado en nuestras simulaciones sea considerado como tráfico real, debe tener un comportamiento auto-similar[7] [6]. Esta propiedad proviene de la geometría fractal propuesta por B. Mandelbrot, la cual dice que una figura básica es muy similar a alguna de sus partes en diferentes escalas de espacio o tiempo.

Un ejemplo de modelos auto-similares en la dimensión espacial, es el modelado de aviones y aeroplanos pequeños para la experimentación de túneles de viento. El comportamiento de los modelos pequeños es similar al de un aeroplano real, pero es mucho más fácil y barato para fines de experimentación. En procesos aleatorios de tráfico de datos, el fenómeno de la auto-similitud se encuentra en la dimensión del tiempo.

Un parámetro que nos ayuda a conocer el grado de auto-similitud en el tráfico de redes de datos por escalas de tiempo, es el llamado parámetro de Hurst. El parámetro de Hurst puede ser estimado de varias formas, en el dominio del tiempo mediante un análisis R/S, mediante el análisis de la varianza, mediante el análisis del periodograma en el dominio de la frecuencia, etc.

## 3. Agentes inteligentes.

Un agente inteligente, según Michel Wooldridge [1], es un sistema (hardware o software) situado en un determinado entorno, capaz de actuar de forma autónoma y razonada en dicho entorno para llevar acabo unos objetivos predeterminados. Las propiedades de un agente inteligente son las siguientes:

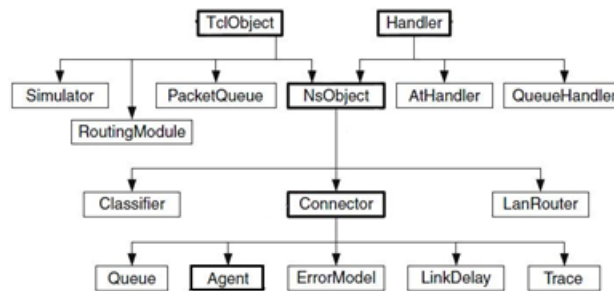


Figura 2: Jerarquía de clases.

- **Autonomía:** Actúan por cuenta propia a nombre del usuario.
- **Inteligencia:** Capaz de aprender en base a la experiencia.
- **Reactividad:** Es reactivo, ya que actúa en función de los sucesos producidos en su entorno.
- **Proactividad:** Es proactivo, ya que toma la decisión de actuar antes que se produzcan los sucesos.
- **Sociabilidad:** Capaz de comunicarse con el usuario, con el sistema y con los demás agentes.
- **Cooperacion:** Se ayuda de otros agentes inteligentes para la realización de tareas de mucha complejidad.
- **Movilidad:** Es capaz de moverse de un sistema a otro para acceder a recursos remotos o para unirse con otros agentes.

### 3.1. Integración de agentes inteligentes a NS2

El tener el código fuente y el poder compilar el simulador fue una característica importante para la selección de NS-2, ya que con estas propiedades, se tiene la oportunidad de agregar un agente inteligente a cada uno de los nodos emisores y receptores, de esta manera poder realizar el estudio del tráfico de la red. En la figura 2 se muestra de que clases hereda el agente inteligente.

El agente es capaz de situarse dentro de un nodo ya sea un host o un enrutador y realizar el cálculo del parámetro Hurst. Después de obtenerlo, el agente les envía esta información a otros agentes adyacentes, esto para realizar la evaluación de estos parámetros y concluir con una propuesta de distribución de tráfico.

## **4. Resultados.**

### **4.1. Evaluación y mejoramiento del escenario**

Como ya se había mencionado anteriormente, nos basamos en un escenario de simulación de redes para la generación de tráfico con un comportamiento auto-similar propuesto en el artículo [6]. En la figura 3, podemos observar los resultados proporcionados por el artículo, mientras que en la figura 4, se muestran los resultados obtenidos mediante la replica de este modelo.

Nuestros resultados son bastante aproximados a los resultados mostrados en el artículo, por lo que podemos decir que usando este escenario se puede generar tráfico con un comportamiento auto-similar, es decir, el tráfico generado es muy parecido al comportamiento del tráfico real. Asimismo, se realizaron más pruebas incrementando el número de nodos emisores de tráfico tipo Pareto de 32 a 50, 70 y 100. Con los resultados de estas simulaciones observamos que el comportamiento del tráfico con 50 nodos emisores (Figura 5) se acerca más a la realidad que el escenario propuesto anteriormente con 32 nodos. También se pudo observar, que si la simulación contiene más de 50 nodos transmitiendo tráfico del tipo Pareto (Figura 6), los niveles de auto-similitud permanecen similares a los obtenidos en la simulación de 50 nodos emisores. En las figuras siguientes se muestran los resultados obtenidos del parámetro de Hurst con diferentes estimadores (R/S, Periodograma, Varianza en el tiempo, Wavelets, Whittle).

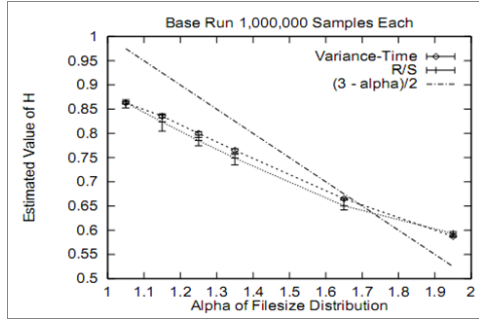


Figura 3: Simulación Original.

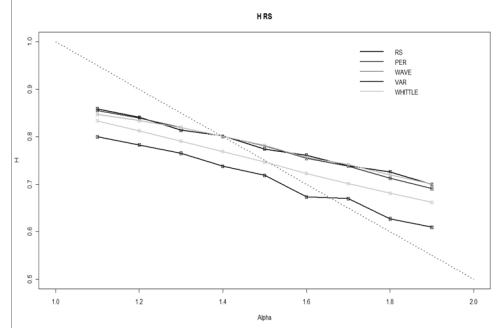


Figura 4: Replica de la simulación.

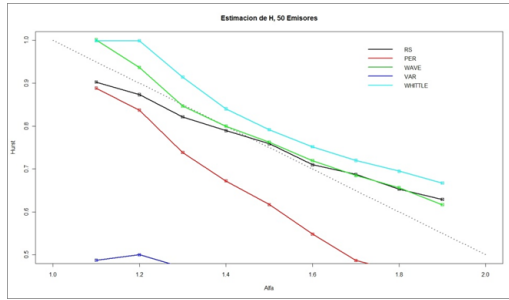


Figura 5: Simulación mejorada con 50 nodos.

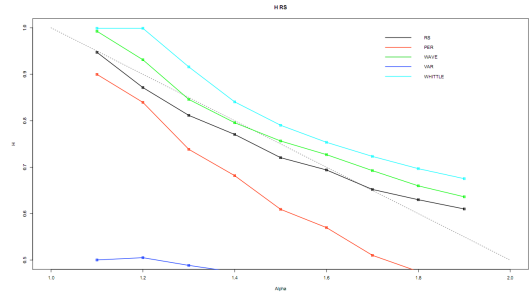


Figura 6: Simulación 70 nodos.

## 5. Implementacion de agentes inteligentes

Para la creación del agente inteligente se desarrollo una clase llamada **agentLASSC** que hereda de la clase conector; esta clase nos permite tener una conexión con los nodos del simulador. Después se estableció la comunicación de los agentes, mediante el envío de paquetes, para esto fue necesario adicionarle al simulador un nuevo paquete IP llamado como el agente LASSC. Con estas características, se realizaron estudios de cuál sería la topología más apta para la implementación de los agentes, una propuesta, es que no es necesario que los agentes estén instalados en cada uno de los nodos, si no que estén en lugares estratégicos en la red. En la Figura 7 mostramos una



topología ideal, donde los agentes deben de estar conectados por los menos a dos nodos. De esta manera, no se tiene tanto trafico por el envío de paquetes de los agentes, y no se disminuye la eficiencia en la toma de decisiones en la distribución del tráfico.

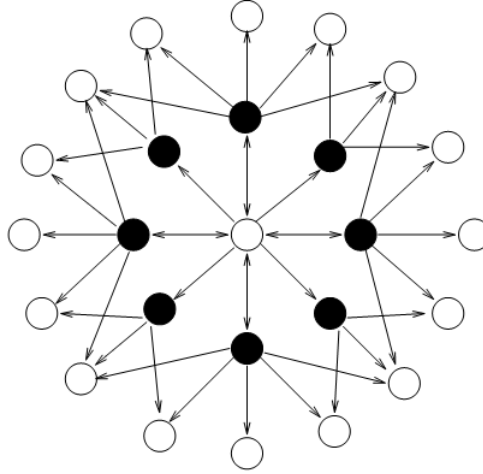


Figura 7: Topología ideal para el estudio de los agentes inteligentes en escenarios de simulacion de redes.

### 5.1. Cálculo de parámetro de Hurst

El parámetro de Hurst es una parte esencial para la distribución del tráfico, este cálculo lo realizan los agentes inteligentes, los cuales manejan una lista doblemente ligada para realizar el muestreo del tráfico; el muestreo se realiza mediante el conteo de paquetes en un intervalo de tiempo (intervalos de 0.005 ms). Para que el agente pueda realizar el cálculo de Hurst de manera optima, se requieren más de 1000 muestras. Después de calculado el parámetro de Hurst, el agente envía esta información a los agentes vecinos, esto para que los agentes propongan una distribución de tráfico.

### 5.2. Balanceo de tráfico

En este trabajo es el de tener este parámetro de Hurst como una métrica de enrutamiento, esto se puede implementar de varias formas, una de ellas es

modificar directamente la tabla de enrutamiento de los nodos o utilizar un protocolo ya establecido como Vector de Distancia o el Estado del Enlace, otra propuesta es el realizar el balanceo del tráfico asignándole un cierto porcentaje de tráfico a cada nodo. La opción que se tomo en este trabajo fue el de proporcionarle un peso a los enlaces, esto de acuerdo al cálculo del parámetro de Hurts. Al analizar la relación que existe entre el tráfico de las redes y el parámetro de Hurts, podemos concluir en 3 etapas: estabilidad, prevención y corrección. Llamamos estabilidad al estado donde la red está trabajando de una manera eficiente, teniendo una calidad de servicio satisfactoria. Este estado lo encontramos en el intervalo de 0 a 0.6 del parámetro de Hurts. Para este punto el costo que se le dio al enlace fue de 1, indicando que al enlace se le puede aumentar el tráfico. En la parte de prevención, es una etapa que en cierta manera la capacidad del enlace se encuentra entre el 80desbordamiento de la cola, los valores del parámetro de Hurst se encuentran entre los 0.61 a 0.85. El costo del enlace para esta etapa fue de 2. La Tercera parte se encuentra entre los parámetros de 0.86 a 1, en donde el enlace esta a su máxima capacidad, tendiendo a la perdida de paquetes, para esto se le puso un costo de 4. Los costos de los enlaces permiten al protocolo de Vector de Distancia actualizar la tabla de enrutamiento y de esta manera redireccionar el tráfico.

## **6. Conclusiones.**

Podemos concluir, que con este trabajo se pudo diseñar un escenario de simulacion de redes, que genera tráfico con un comportamiento auto-similar más, es decir, mas apegado a la realidad. Igualmente, a este escenario se le integraron agentes inteligentes, los cuales son colocados en nodos específicos donde realizan el análisis del tráfico que atraviesa ese nodo, calcularán el parámetro de Hurst en tiempo real, y por último, proponen una opción de balanceo de tráfico (tomando en cuenta el parametro de hurst) para mejorar el desempeño de ese escenario de red.

## **7. Trabajo futuro.**

Como trabajo futuro, se planea buscar nuevas opciones de balanceo de tráfico; una en la que se esta pensando es, en no solo enviar el tráfico de un enlace a otro menos congestionado,sino enviar porcentajes de tráfico a

diferentes enlaces dependiendo del nivel de saturacion de estos.  
Otro punto muy importante, es sobre la calidad de servicio. Es por esto que se necesita realizar un estudio del impacto que tiene el tráfico con un comportamiento auto similar a los parámetros de calidad de servicio.

## Referencias

- [1] Autor: Wooldridge, Michael J. *An introduction to multiagent systems*. Wiley, 2002.
- [2] Autor: Black, Uyless D, *Tecnologías emergentes para redes de computadoras*. Prentice Hall, 1999.
- [3] Autor: Oleg i. Sheluhin, Sergey M. Smolskiy, Andrey V. Osin, *Self-similar processes in telecommunications*. Wiley, 2007.
- [4] Autor: David Nicolás Ruiz, *Redes y Servicio de Comunicaciones*. Mexico, 2001.
- [5] Autor: Diego R. López, *Internet la red con mayúsculas: una introducción a la estructura y servicios de la red global*. Sevilla, 1997.
- [6] Autor: Kihong Park, Gitae Kim y Mark Crovella, *On the Effect of Traffic Self-similarity on Network Performance*.
- [7] Autor: Will E., Murad S., Walter W. and Daniel V. *On the self-similar nature of ethernet traffic*.
- [8] The Network Simulator: <http://www.isi.edu/nsnam/ns/>