

Consideraciones metodológicas para la investigación con simuladores de red

Methodological considerations for research with network simulators

Franco A. Trinidad, Carlos A. Talay
al.franco@pm.me, carlostalay@yahoo.com.ar
Rectorado - Universidad Nacional de la Patagonia Austral
Av. Lisandro de la Torre 860 - Río Gallegos - Santa Cruz - Argentina

Recibido: 15/02/2019. Aceptado: 07/10/2019

RESUMEN

La simulación permite recrear escenarios para su estudio, mediante programas de computadora, sin necesidad de implementarlos en el mundo real. Esto reduce el costo de experimentación e incrementa el nivel de control que se tiene sobre un modelo de sistema. En el contexto de la investigación en redes de computadoras, los simuladores resultan muy útiles en el análisis de protocolos y componentes de red.

Uno de los problemas más estudiados es el análisis de rendimiento del protocolo TCP. En sus orígenes, éste sólo contemplaba pérdidas de paquetes por problemas de congestión, típicos de redes cableadas predominantes en la época. La aparición de redes inalámbricas puso de relieve falencias de TCP, al no poder responder a la multiplicidad de razones por las que éstas pierden paquetes: salidas del rango de cobertura, movilidad, interferencia de señales, etc.

Hay muchos autores que abordan las ventajas y desventajas del modelado de sistemas con enfoque de simulación, estándares para seleccionar herramientas más adecuadas, la estructura de cursos de instrucción y capacitación, y aplicaciones en ámbitos particulares. También existe una literatura amplia sobre el análisis de protocolos haciendo uso de simuladores de red, en particular NS2. Este trabajo busca integrar conocimientos adquiridos del área de redes de computadora y telecomunicaciones, con el manejo de herramientas de simulación y la experiencia en el estudio de rendimiento del protocolo TCP, para explicar algunas consideraciones metodológicas que resultan útiles en dicho emprendimiento.

Palabras clave: TCP; Simulación; Rendimiento.

ABSTRACT

Simulation allows one to recreate scenarios for studying, through the use of computer programs, without needing to implement them in the real world. This reduces the cost of experimentation and increases the level of control one has over the system model. In the context of computer networking research, simulators are very useful for the analysis of network protocols and components.

Analysis of TCP performance is one of the most studied problems. The original implementation of TCP used to contemplate packet loss due to network congestion only, typical of wired connections. The emergence of wireless networks called the attention to some faults in TCP, mainly its inability to distinguish between the many sources of packet loss in this kind of connections: out-of-range devices, mobility, signal interference, etc.



There are many authors that talk about the advantages and disadvantages of system modeling with a simulation approach, the structure for courses, and its application in certain domains. There's also a vast literature on the topic of protocol analysis using network simulators, NS2 in particular. This paper attempts to integrate knowledge of computer networking and telecommunications, with simulation tools capabilities and expertise in TCP performance evaluation research, in an attempt to explain some useful methodological considerations in such enterprise.

Key words: TCP; Simulation; Performance.

1. INTRODUCCIÓN

Debido al rol que ocupan los sistemas de telecomunicación en la era de la información, las redes de computadora constituyen un objeto de estudio particularmente importante. Un problema que aparece es que el diseño de experimentos para su investigación puede resultar muy costoso: se debe construir la infraestructura de una red, configurar los dispositivos acorde a las especificaciones deseadas, monitorear su funcionamiento, controlar fuentes de incertidumbre, incorporar mecanismos para recolectar información, etc. La comprensión de sistemas complejos resultaría prácticamente imposible sin la ayuda de herramientas que permitan estudiarlos sin necesidad de desplegarlos.

Los simuladores de red permiten analizar distintos escenarios y aislar problemas para un estudio más eficiente. Son herramientas que facilitan el control y monitoreo, reducen costos de implementación y disponen de mecanismos de visualización y recolección de datos estadísticos para su análisis. Esto hace que sean ideales para el ámbito de investigación.

Algunos autores han desarrollado estándares para la elección de una buena herramienta de simulación, han discutido la historia, ventajas y desventajas de su uso en investigación, enseñanza, demostración de protocolos o componentes y resolución de problemas en el mundo real. Otros se han enfocado en los problemas a resolver en el área de redes y comunicaciones, denotando la elección de un simulador en particular pero sin explicar la metodología de trabajo con el mismo¹. Tomándolos como punto de partida, este trabajo integra conocimientos sobre redes, simulaciones y metodología de investigación con el objetivo de mostrar qué consideraciones deben hacerse a la hora de utilizar simuladores de red en investigación, particularmente, en el análisis de rendimiento del protocolo TCP.

La publicación está estructurada de la siguiente forma: la sección 2 explica el origen de la simulación, y los conceptos básicos de redes, el protocolo TCP y los simuladores de red; la sección 3 describe la metodología de trabajo típica en simulación, comentando cómo varía para adaptarse al estudio de redes; la sección 4 muestra y discute algunos casos en los que se definieron los datos a analizar en base a limitaciones y características propias de herramientas de simulación; la sección 5 presenta algunas conclusiones; finalmente, se realizan algunas recomendaciones para trabajos futuros y se agradece a las entidades que hicieron posible este trabajo.

¹ Este tipo de trabajo suele ser útil ya que muestra el modelo utilizado y las condiciones iniciales del mismo. Sin embargo, no suelen indicar la metodología asociada al uso de simuladores, porque no es su objeto de estudio.

2. MARCO DE REFERENCIA

Marco histórico

Los métodos más directos para estudiar la naturaleza de un fenómeno son la observación y la experimentación, que requieren la interacción con un sistema real o la reproducción de una versión simplificada del mismo bajo situaciones controladas, respectivamente. Pero no siempre pueden utilizarse: dependiendo del objeto de estudio, éstos pueden resultar muy costosos, la observación puede ser imposible con la tecnología disponible, el diseño de un experimento puede ser poco práctico, etc. El modelado de sistemas surge como alternativa para representar un sistema sin tener que implementarlo.

Durante mucho tiempo, la construcción de modelos para estudiar el comportamiento de sistemas dependía exclusivamente del enfoque analítico. Esto es, se desarrollaban descripciones matemáticas de un sistema y se aplicaban métodos numéricos para obtener una percepción más precisa y extensa sobre su funcionamiento. Sin embargo, hay situaciones en las que el uso de un modelo matemático no resulta conveniente, práctico o siquiera factible. La complejidad de algunos sistemas no da lugar a las simplificaciones y suposiciones necesarias para desarrollar una descripción matemática. En tales casos es preferible recrear el sistema mediante programas de computadora, es decir, utilizar un enfoque de simulación.

El origen de las simulaciones se remonta al año 1929, con el primer simulador de vuelos llamado “Blue Box” o “Link Trainer”, desarrollado por el ingeniero Edwin Albert Link para uso del ejército estadounidense. Sin embargo, el concepto de simulación no se consolidó hasta la segunda guerra mundial, con el trabajo de Stanislaw Ulam y John von Neuman en el proyecto Monte Carlo. La simulación les permitió resolver problemas de reacciones nucleares ante los altos costos de una solución experimental y la complejidad del análisis matemático.

Hoy en día, la simulación se ha convertido en una técnica muy útil para experimentar y desarrollar aplicaciones en distintos ámbitos: su difusión ha incrementado junto a la capacidad de procesamiento y el acceso a las computadoras. En particular, puede destacarse la simulación de redes de computadoras debido a la amplia disponibilidad de herramientas y documentación, y el hecho de que se usa no sólo en investigación sino también en capacitación y para resolución de problemas reales.

Uno de los temas más estudiados es el análisis de rendimiento del protocolo TCP en redes inalámbricas. Dado que el foco de este texto es la metodología de trabajo con simuladores de redes en investigación, sólo se mencionarán detalles en cuanto sean necesarios para entender decisiones de diseño. Debido a su accesibilidad y flexibilidad, los simuladores de software libre Network Simulator 2 (NS2) y Network Simulator 3 (NS3) han facilitado el desarrollo de una gran cantidad de publicaciones sobre análisis de protocolos. Las alternativas de software privativo no se abordan en este trabajo.

Marco conceptual

Redes

Una **red** es un conjunto de nodos conectados entre sí mediante enlaces. Para que la comunicación entre ellos sea posible, los dispositivos deben saber qué datos mandar, en qué formato, en qué momento, como responder a ciertos mensajes, etc. El conjunto de reglas que indica cómo debe llevarse a cabo la comunicación es lo que se conoce como un **protocolo**.

El protocolo más común en Internet es **Transmission Control Protocol (TCP)**. TCP opera en la capa de transporte del modelo TCP/IP, por lo que se encarga de llevar datos de un extremo de la red a otro. Su popularidad se debe en gran parte a las siguientes características:

- Es orientado a la conexión, por lo que antes de transmitir datos, se asegura de haberse conectado con el otro extremo.
- Es robusto y fiable, ya que no asume que los protocolos de las capas subyacentes sean confiables y asume la responsabilidad de asegurar una transmisión exitosa.
- Implementa mecanismos de control para garantizar que la información llegue de forma completa, ordenada y correcta.
- Utiliza el estado de la red para modificar la tasa de transmisión de paquetes con el objetivo de utilizar el uso de recursos.

Simulación

En términos sencillos, **simular** es “representar algo, fingiendo o imitando lo que no es”. Algunos expertos como Thomas H. Naylor, Giuliano Gnugnoli y Herbert Maisel, referentes al hablar de simulación, la definen como “una técnica numérica para conducir experimentos en una computadora digital”. Aunque varían en cuanto a los aspectos del mundo que se pueden abarcar, hablan del uso de modelos matemáticos y lógicos para describir el comportamiento y estructura de sistemas a lo largo del tiempo.

R. Shannon, otro estudioso del tema, prefiere hablar de la simulación como el proceso de diseñar y desarrollar un modelo computarizado de un sistema, y conducir experimentos para entender su comportamiento o evaluar estrategias con las cuales se puede operar.

La definición que resulta más adecuada en base a las actividades realizadas en la simulación de redes, es: el proceso de diseñar y desarrollar un modelo computarizado para recrear un aspecto del mundo real, con el objetivo de entender su funcionamiento, estudiar un problema particular o tomar decisiones estratégicas.

Simulación de redes

La **telemática** resulta de la convergencia entre los sistemas informáticos y de telecomunicaciones. Una de las aplicaciones más destacables de esta disciplina son los simuladores de redes. Los **simuladores de redes** son herramientas que permiten recrear sistemas y escenarios de telemática antes de que sean desarrollados, permitiendo su estudio sin necesidad de implementar la infraestructura subyacente ni la configuración de los dispositivos acorde a las especificaciones. Esto hace que sean ideales para el ámbito de investigación, en donde los recursos deben invertirse cuidadosamente y los tiempos son cortos.

Componentes en una simulación de red

Llevar a cabo una simulación en una computadora puede pensarse como hacer interactuar distintos componentes de red. Éstos interactúan entre sí, detonan eventos, cambian el estado del sistema, pelean por recursos, etc. Según Ingalls, los componentes esenciales de una simulación de eventos discretos² son:

- Entradas: acciones del entorno sobre el sistema, que modifican su estado.
- Salidas: Cantidades medidas que se derivan del estado del sistema, necesarias para analizar los resultados de la simulación. Por ejemplo: throughput, sequence number, end-to-end delay, jitter, PLR, PDR, RTT, RTO, etc³.
- Estado: Condición interna del sistema.
- Variables globales: Variables que llevan registro del estado del sistema. Almacenan información concerniente a múltiples entidades, como número de nodos, disposición del espacio de simulación, referencias a eventos, etc.
- Entidades: Elementos estructurales que efectúan cambios en las variables de estado del sistema. Son objetos que interactúan entre sí durante la simulación. Estas pueden ser: nodos, paquetes, flujo de paquetes, el reloj de simulación (entidad no física), etc.
- Atributos: Características únicas de cada entidad.
- Actividades: hace referencia a los procesos y la lógica de la simulación. Hay tres grandes grupos: delay, cuando se suspende el flujo de una entidad por determinado período; queue, cuando se suspende el flujo por un período de tiempo no especificado; y lógica, que permiten efectuar cambios en el sistema mediante la manipulación de variables de estado y decisiones lógicas.
- Eventos: Son condiciones que ocurren en determinado momento y causan cambios en el estado de la simulación. Las entidades interactúan con las actividades para crear eventos.
- Recursos: Representa todo aquello que se tiene una capacidad limitada y es compartido por las entidades. Por ejemplo: ancho de banda, nodos en una comunicación, intersecciones de tráfico, etc.
- Programador de eventos: Mantiene la lista de eventos a ejecutar. Durante la simulación se mueve acorde al reloj de simulación y ejecuta los eventos de forma cronológica
- RNG (*Random Number Generator*): genera números para introducir aleatoriedad en el modelo de simulación.
- Recolector de estadísticas: utilizan variables para recolectar datos relevantes como tiempo de llegada, tiempo de salida, etc.

² Hay dos tipos de simuladores: por tiempo y por eventos discretos. NS2 es del segundo tipo. Se dejan las distinciones a autores como Issariyakul y Hossain.

³ Estos son algunos de los parámetros de rendimiento que permiten estudiar el comportamiento de TCP en distintos escenarios. Según lo que se estudie se hará énfasis en alguno, pero es raro que se necesiten todos.

Marco teórico

TCP en redes inalámbricas

TCP fue desarrollado a finales de los '70 y adoptado en la década del '80, cuando la mayoría de las redes utilizaban conexiones cableadas. Un problema típico de ellas era la pérdida de paquetes debida a la congestión: la cantidad de datos transmitidos superaba la capacidad del canal o la velocidad de transmisión superaba la de procesamiento, y por ende se debían descartar para evitar el colapso de la red. En base a este contexto, TCP incluyó en su diseño algoritmos de control de congestión para regular la tasa de envío y detectar pérdidas por congestión.

A finales de los 90', con la primera versión del estándar IEEE 802.11, se pusieron de manifiesto las falencias de TCP en el medio inalámbrico. A diferencia de las redes cableadas, éstas otras sufrían pérdidas por múltiples razones: movilidad, salidas del rango de cobertura, fenómenos de hand-off (cuando un nodo móvil pasa de una celda a otra, conectándose a una nueva estación base), interferencia de señales, errores en tránsito, etc. TCP no fue diseñado para diferenciar entre todas esas fuentes de error, y trataba a toda pérdida de paquetes como congestión. En consecuencia, se deterioraba el rendimiento de la red porque se intentaba aplicar una solución incorrecta.

Las pérdidas de paquetes debidas a las razones mencionadas generan patrones de comportamiento distintivos en la red, por lo que pueden estudiarse para plantear potenciales soluciones. Lo que siguió al descubrimiento de la insuficiencia de TCP, fue una serie de propuestas con el objetivo de mejorar su rendimiento en redes inalámbricas, según el tipo de aplicación y el contexto en el que funcionan.

Trabajo con simuladores de red

Entre las técnicas más utilizadas para el análisis y evaluación de rendimiento de TCP, se encuentra la simulación de redes. Existen simuladores como NS2 y NS3, ricos en funcionalidades y documentación, que gracias a su licencia libre, son ampliamente utilizados en el ámbito de investigación. Son herramientas tan útiles como complejas, por lo que requieren un conocimiento relativamente avanzado sobre su arquitectura, características y en particular, alcances y limitaciones. Estos últimos determinan la validez de los datos y, en definitiva, la credibilidad del estudio.

Los investigadores deben comprender las suposiciones y generalizaciones que se hacen al desarrollar las herramientas de simulación, así como las limitaciones que plantean las computadoras, el software y el error humano. Los resultados deben entenderse en el contexto de la simulación y no ser tomados como reproducciones perfectas del fenómeno estudiado. Aunque lo ideal sería que las simulaciones sean tan cercanas a la realidad como sea posible, la meta es poder trabajar con un modelo del mundo para entender su comportamiento, por lo que los niveles de precisión necesaria pueden variar acorde a ello.

Hay ciertas consideraciones a tener en cuenta a la hora de analizar los resultados de una simulación⁴:

⁴ Aunque el equipo de investigación está en el proceso de aprender NS3, tiene más experiencia de trabajo con NS2. Por eso, la mayoría de los comentarios y resultados que se presentan en el trabajo hacen referencia a éste último. Para más información sobre herramientas de simulación, ver: M. Köksal. A Survey of Network Simulators Supporting Wireless Networks. Graduate School of Natural and Applied Sciences. Middle East Technical University. Turquía: 2008.

- Las capa física y de control de acceso al medio juegan un rol muy importante en el comportamiento de una red. La forma en que éstas estén implementadas en el simulador afectarán crucialmente los resultados. En NS2, por ejemplo, la abstracción llega a simular cada paquete que se envía, pero no cada byte.
- La resolución del reloj determinará qué tan precisos son los datos obtenidos, pero también condicionará el tiempo de ejecución. Aunque NS2 tiene una capacidad de resolución de picosegundos, la opción predeterminada es nanosegundos ya que la mayoría de los escenarios no requieren ni se benefician de un nivel de detalle mayor.
- No es posible lograr una implementación idéntica de un algoritmo en distintos simuladores. Dependiendo de la arquitectura, la forma en que se integran nuevos protocolos y se crean escenarios es muy diferente.
- Como todo producto de software, los simuladores tienen bugs que pueden causar fallas en los resultados. Esto resalta la importancia de estar familiarizado con los conceptos del dominio en el que se desarrollan las simulaciones, para poder detectar inconsistencias en los resultados. Es común que, al diseñar un ensayo, se encuentren fallas durante la validación del modelo al notar resultados teóricamente inconsistentes. En la mayoría de los casos es un problema de la simulación y no de la teoría.

Errores que se cometen al trabajar con simuladores

Algunos autores mencionan errores comunes que cometen los investigadores en la simulación. En conjunto con la experiencia del equipo de investigación, se puede mencionar:

- No se define un objetivo alcanzable. Toda el proceso de simulación se estructura en torno a la información que se desea obtener para el análisis. En la evaluación de rendimiento de TCP, ésto suele hacer referencia a parámetros como throughput instantáneo y promedio, delay, número de secuencia, etc. Si no se tiene en claro qué información permite comprender el comportamiento del protocolo en determinadas circunstancias, puede que los resultados no sean útiles.
- No se dispone de los conocimientos esenciales o se falla en coordinarlos adecuadamente. Los investigadores pueden no estar familiarizados con todas las opciones para implementar un determinado escenario por lo que se adapta el objeto de estudio de forma inadecuada. Por ejemplo: las implementaciones de variantes de TCP en NS2 suelen tener variables específicas según el algoritmo que utilicen. Si no se conocen, pueden no aprovecharse el rango de posibilidades para diseñar las pruebas.
- Inadecuado nivel de participación. En un grupo de trabajo, es útil que haya roles diferenciados para lograr una división de tareas adecuadas: una persona especializada en conocer la arquitectura del simulador; otra que entienda protocolos de comunicación; alguien familiarizado con la literatura relevante para seleccionar escenarios y problemas para estudiar; alguien que sepa implementar los modelos en el simulador; etc. Todos deberían entender lo que pasa, pero la especialización dada por los roles facilita la coordinación de conocimientos, y deja en claro el nivel de participación adecuado durante la simulación.
- Nivel de detalle inapropiado en la elaboración del sistema a simular. Si se invierte demasiado tiempo en recrear hasta el más mínimo detalle del escenario, incrementa el costo de la simulación y no es evidente que la diferencia en los resultados sea notable después de cierto punto. Por otro lado, si el nivel de detalle es insuficiente, puede que los datos obtenidos sean incorrectos o insuficientes para analizar lo que pasa.

- Falta de comunicación en los miembros. Relacionado a la división de trabajos mencionada, las tareas deben coordinarse adecuadamente. Si no hay una comunicación fluida, puede ser que se confundan los roles (que haya más de una persona realizando una tarea) o se ignoren los objetivos (que no haya ninguna persona realizando alguna tarea).
- Uso inadecuado del lenguaje de simulación para el proceso de simulación. Además de entender cómo está estructurado el simulador y las funciones de las que dispone, debe estudiarse el lenguaje en el que se implementan las simulaciones, de lo contrario no se cuenta con todas las herramientas para desarrollar las pruebas. En el caso de NS2, el lenguaje es Otcl/Tcl. También resulta útil entender C/C++ para consultar el código fuente en caso de que la documentación sea escasa.
- Documentación obsoleta o inexistente. Cuando se dejó de lado el desarrollo de NS2 en el 2010, para enfocarse en NS3, este problema comenzó a crecer. NS2 tiene más de dos décadas de desarrollo, y por ende, muchos ejemplos y modelos de prueba. Sin embargo, con la llegada de NS3, se dejó de incorporar nuevos protocolos y componentes de red. La documentación quedó estancada y dejó de completarse. Hoy en día, hay muchos documentos y tutoriales que ya no sirven, así como funcionalidades que requieren estudiar el código fuente para ser utilizadas. El investigador debe conocer la historia y el estado de soporte actual del simulador y un función antes de intentar implementar un escenario.
- Utilización de modelos no verificables. Como se mencionó anteriormente, el software tiene bugs y los simuladores no representan la realidad de forma exacta. No es obvio que la implementación de cualquier algoritmo vaya a funcionar, por lo que debe validarse antes de aceptar los resultados que se puedan obtener.
- Mal uso de herramientas y técnicas en la dirección del desarrollo completo. En el caso de NS2, hay muchos ejemplos online. Se pueden cometer muchos errores: copiar y pegar código sin entender lo que dice, utilizar scripts sin validar los métodos y las fórmulas que usan, aceptar explicaciones de foros que no estén en concordancia con la documentación por no conocerla, etc.

Desventajas de la simulación

- La simulación de un sistema requiere equipo computacional costoso⁵.
- La técnica de simulación implica la participación de recursos humanos especializados. El costo de formación del investigador puede ser costoso, principalmente en términos de tiempo. Los simuladores son herramientas complejas, y no son fáciles de aprender. Adicionalmente, se debe tener conocimiento teórico del dominio en que se trabaja.
- El desarrollo y validación de un modelo suele requerir bastante tiempo. Aunque ciertos aspectos pueden reutilizarse, en investigación se busca innovar, por lo que la introducción de nuevos elementos es algo constante.
- Debido a las consideraciones mencionadas anteriormente, es posible que uno sea escéptico de los resultados obtenidos y prefiera acudir la experimentación directa.

Ventajas de la simulación

⁵ Es posible trabajar con equipos más baratos, pero el tiempo de procesamiento de pruebas incrementa significativamente. Por ejemplo, resulta muy útil tener un GPU dedicado.

Aunque ya se mencionó brevemente, vale la pena explicar algunas de las ventajas⁶ que justifican el uso de la simulación a pesar de los riesgos que conlleva:

- La técnica de simulación puede ser un instrumento pedagógico⁷ para reforzar las habilidades prácticas de los estudiantes e investigadores en formación, y para capacitación de personal en una organización a fines.
- Un estudio de simulación permite estudiar los efectos que traen consigo cambios internos y externos a un sistema.
- La observación detallada del sistema simulado conduce a un mejor entendimiento y por consiguiente, sugiere estrategias para mejorar la operación y eficiencia del mismo.
- La simulación ayuda a entender mejor la operación de sistemas complejos, a detectar los factores más importantes que interactúan con él, así como sus interrelaciones.
- La técnica de simulación sirve para experimentar con situaciones nuevas, de las que se tiene poca o nada de información. Así se pueden anticipar resultados no previstos.
- Se pueden anticipar problemas que pueden surgir en la ejecución de un sistema al introducir elementos nuevos.
- Permite recrear y repetir, en condiciones prácticamente idénticas, procesos difíciles de hallar en el mundo real.
- Proporciona un marco para analizar modelos en una amplia variedad de aplicaciones para experimentar y tomar decisiones. Se puede trabajar en diversas áreas.
- Los modelos ofrecen un mayor nivel de control sobre lo que ocurre, ya que se pueden manipular acorde a lo que interese estudiar.
- Elimina riesgos relacionados con la falla de hardware, costos de preparación del banco de pruebas, accidentes durante la experimentación, etc.
- Permite una retroalimentación inmediata. Se pueden modificar algunos parámetros y obtener nuevos resultados inmediatamente sin tener que rediseñar toda la infraestructura del experimento.

3. METODOLOGÍA

Materiales y métodos

El trabajo con NS2 y NS3 requiere conocimientos acerca de varios lenguajes de programación⁸:

- AWK (filtrado y procesamiento de datos)
- Perl (ídem a AWK)
- Otcl/Tcl (scripts de simulación en NS2)

⁶ Las primeras ventajas son variaciones de las dadas por el economista Thomas H. Naylor.

⁷ Vale notar que los simuladores facilitan la comprensión de distintos fenómenos pero no reemplazan la interacción con el sistema real. Esto debe quedar claro en cualquier curso de herramientas de simulación.

⁸ El equipo de investigación ha trabajado con NS2 durante más de 3 años, por lo que se utilizará como estándar para todas las referencias al trabajo con simuladores.

- Python (scripts de simulación en NS3)
- C++ (lenguaje en el que se encuentran implementados los simuladores; scripts de simulación en NS3)
- Bash (automatización de ejecución de pruebas, procesamiento de datos, y generación de gráficos y tablas de datos estructuradas)

El análisis de rendimiento de TCP implica el dominio de conceptos básicos de redes de computadoras, telecomunicaciones, capas del modelo TCP/IP y los protocolos que la componen, parámetros de rendimiento, tipos de dispositivos y conexiones, y fenómenos comúnmente asociados que puedan afectar su funcionamiento.

La simulación es una herramienta que resulta muy útil en diversos ámbitos: economía, biología, comunicaciones, hardware, software, automatización, etc. Debido a ello, se han escrito muchas guías sobre cómo llevar a cabo un estudio de simulación. Sin embargo, la mayoría de los autores concuerda, en mayor o menor medida, con los siguientes pasos:

- **Definición del sistema:** Antes de crear un escenario, es necesario entender adecuadamente el sistema a simular: su interacción con otros sistemas, restricciones, variables relevantes e interrelaciones. A partir de ello, se puede determinar un objetivo y los resultados esperados.
Esto incluye la comprensión del fenómeno o problema a estudiar. El estudio de rendimiento de TCP implica, por un lado, conocimientos básicos de redes (nodos, tipos de conexiones), protocolos de comunicación (modelo OSI, TCP/IP, interacción entre las capas, etc) y parámetros de rendimiento (throughput, delay, número de secuencia, RTT, RTO, PLR, PDR, jitter, etc). Por otro lado, es necesario entender cómo interactúan los componentes de un sistema de comunicación para saber qué datos facilitarían la comprensión del fenómeno estudiado. Finalmente dominarse la herramienta a utilizar para entender qué puede recrearse o no, y cuales son las posibilidades de manipulación.
- **Formulación del modelo:** Definir y construir el modelo con el que se obtendrán los resultados deseados, consiste en especificar las variables y sus relaciones lógicas.
- **Colección de datos:** A partir de documentación, publicaciones y opiniones expertas, se definen los datos que el modelo necesita para obtener ciertos resultados. Por ejemplo, puede utilizarse un ancho de banda de 10Mbps en gran parte de la red, con un cuello de botella de 2Mbps para incrementar las probabilidades de que la congestión ocurra en dicho tramo.
- **Implementación del modelo en la computadora:** En NS2, esto implica⁹: un script de simulación en Otcl/Tcl; scripts para procesamiento de datos en perl, awk y python; scripts para automatización del proceso en bash; y scripts de procesamiento gráfico para gnuplot.
- **Validación:** Validar un modelo implementado es fundamental para asegurar que se van a obtener los datos deseados y que no se han cometido errores en el proceso, ya sea en el uso del simulador o en el ingreso de datos pertinentes. Algunas formas en las que esto se lleva a cabo son:

⁹ Se describe la implementación utilizada por el grupo de trabajo.

- Opinión experta que permite detectar inconsistencias teóricas. Por ejemplo, definiendo una capacidad de canal de 2Mbps, se pone un tope al valor de throughput que se puede obtener.
- Datos históricos que predicen la exactitud de los resultados
- El ingreso de datos que produzcan fallas en el sistema para asegurar que responde adecuadamente ante cantidades inadecuadas.

En términos prácticos, esto puede lograrse: haciendo corridas de prueba y comprobando que los valores de magnitud en los gráficos y las tablas obtenidas sean adecuados; visualizando la ejecución del escenario en herramientas como Network Animator; revisando los archivos de traza para verificar que el tráfico se de entre los nodos correctos.

- **Experimentación:** Luego de validar el modelo, se procede a ejecutar las pruebas y generar los datos. Dependiendo del escenario y el equipo disponible, este proceso puede tardar desde algunos minutos hasta días. De ser necesario, se pueden procesar los datos para obtener nueva información o presentarla de otra forma. Por ejemplo, puede ocurrir que resulte más conveniente presentar los resultados en una tabla que en un gráfico, pero que la idea no haya surgido hasta después de haber ejecutado las pruebas.
- **Interpretación:** En esta etapa se interpretan los datos generados y se toman decisiones. Es posible que los resultados no den información interesante acerca del comportamiento de un sistema y terminen descartándose. También es posible que se observen fenómenos inesperados que ayuden a explicar por qué funciona de cierta forma bajo esas condiciones. Por ejemplo, el ajuste de parámetros alfa y beta de la variante TCP Vegas ayuda a minimizar el tiempo de recuperación ante errores en ráfaga. Esto se descubrió al notar que cuando se incrementaba la cantidad de paquetes perdidos en ráfaga, el tiempo de recuperación incrementaba exponencialmente en TCP Vegas pero no en otras variantes.
En base a los datos obtenidos, suelen detectarse insuficiencias en el modelo de simulación, que se tienen en cuenta para futuros experimentos. Con cada prueba se perfecciona el proceso de simulación.
- **Documentación:** Finalmente se documentan los resultados obtenidos. En el ámbito de investigación, esto hace referencia a una publicación técnica, que incluye el problema abordado, el escenario simulado, los datos obtenidos y la interpretación de los mismos. Adicionalmente es útil dejar asentado los aspectos más importantes de las pruebas realizadas para poder utilizarlas como referencia en futuros experimentos.

Aunque el orden puede variar según el escenario, es una buena guía para llevar a cabo un estudio de simulación. En trabajos anteriores se ha indicado una metodología similar.

Para este trabajo se recolectaron algunos ejemplos de pruebas realizadas para ilustrar las consideraciones que se tuvieron en cuenta a la hora de diseñar el modelo e interpretar los resultados.

4. RESULTADOS Y DISCUSIÓN

En esta sección se presentan algunos de los escenarios adaptados acorde a las limitaciones de la herramienta. Seguida la presentación y discusión de cada uno se hará una breve explicación sobre los pasos involucrados en el estudio de la simulación. Se optó por éste orden para que el

énfasis se mantenga en los resultados obtenidos. Vale aclarar que los pasos no son normativos, sino que se adecuan al estudio, a las herramientas y a la experiencia de los investigadores.

Resultados y discusión

Datos de simulación estabilizada:

A continuación se presentan algunos ejemplos de la competencia por recursos entre dos flujos de paquetes distintos, correspondientes a nodos que utilizan distintas variantes de TCP. Algunos fueron configurados para comenzar al mismo tiempo, y otros con una diferencia de tiempo considerable.

Al utilizar modelos de simulación estocástica, generalmente se busca estudiar un sistema en estado estable. Sin embargo, por cuestiones técnicas suelen existir estados transitorios en las etapas iniciales. Debido a la naturaleza secuencial del procesamiento de eventos en los simuladores de red, al implementar un escenario con dos o más flujos de paquetes, éstos comenzarán a destiempo aunque se indique el mismo momento de inicio. En otras palabras, debido a que los eventos nos se ejecutan de forma simultánea, siempre habrá un flujo que empiece primero.

Para lidiar con este problema se utilizaron dos enfoques: el primero consiste en utilizar el mismo tiempo de inicio en la implementación del modelo, y el segundo incluye una diferencia de 5 segundos entre cada comienzo. En ambos casos se extiende el tiempo de transmisión lo suficiente como para que el tráfico se estabilice, y se analiza el período de tiempo intermedio para atenuar la influencia de qué protocolo comienza primero.

- *Escenario 1:*

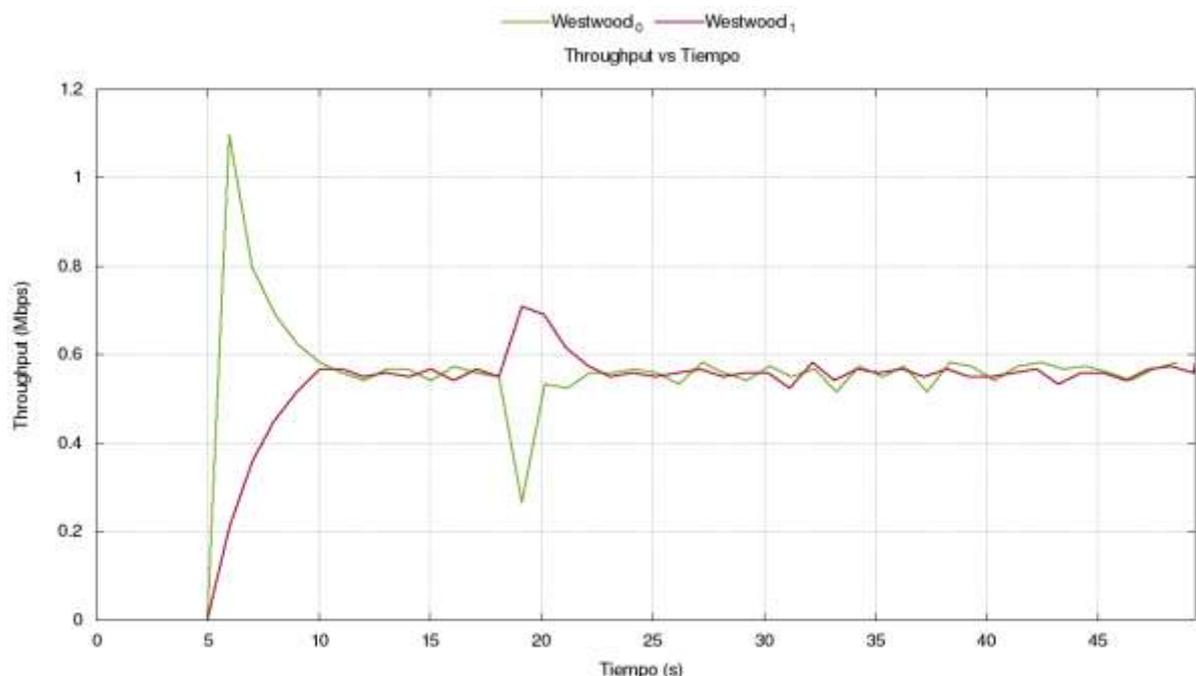


Figura 1. Throughput vs Tiempo. Dos flujos de TCP Westwood en competencia. Ambos comienzan al mismo tiempo y transmiten 3000 paquetes. Cerca de los 18 segundos se introduce un error en ráfaga de 10 segundos al segundo tráfico (Westwood₁).

La figura 1 ejemplifica el primer caso. Aunque ambos flujos fueron programados para empezar a los 5 segundos, se hace evidente que Westwood₀ comienza primero, por lo que ocupa la totalidad del canal durante un instante hasta que Westwood₁ incrementa su uso de recursos. A partir de los 11 segundos aproximadamente se estabiliza la competencia entre tráficos.

En este caso se busca estudiar el efecto de la ráfaga de errores en uno de los tráficos. Para evitar que la disparidad inicial influya en los resultados, la estrategia utilizada es comenzar el análisis a partir del momento en que se estabiliza la simulación.

- Escenario 2:

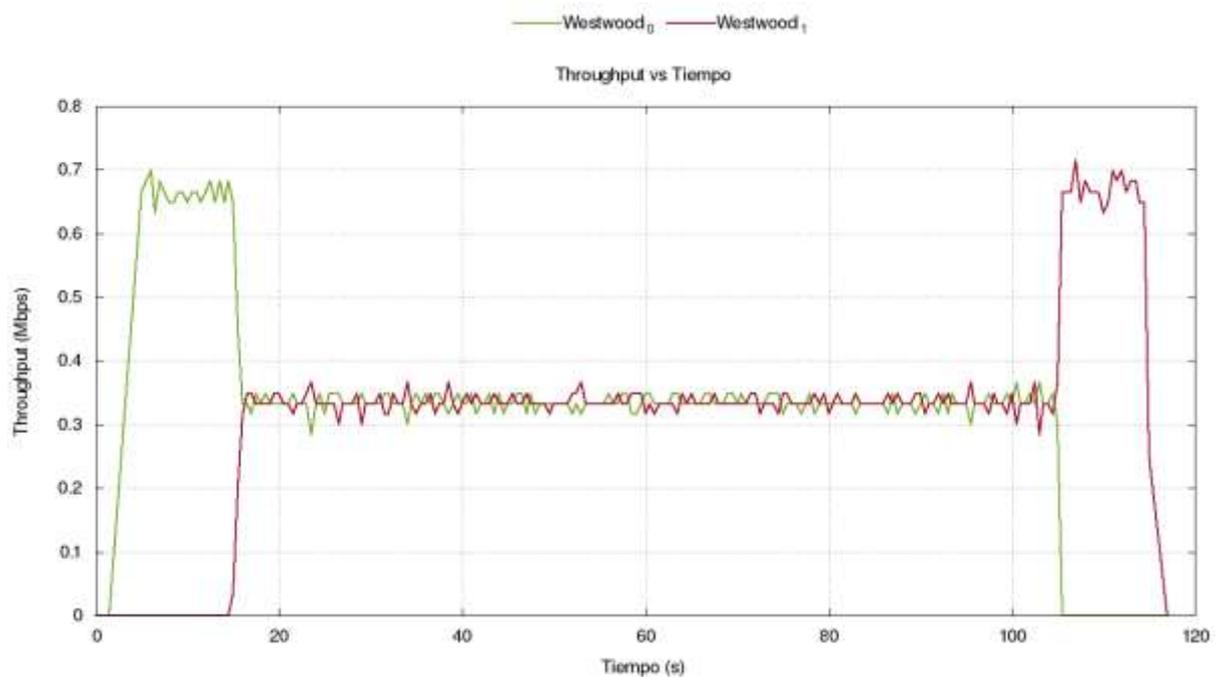


Figura 2. Throughput vs Tiempo. Dos flujos de TCP Westwood en competencia. Westwood₀ comienza a los 5 segundos, y Westwood₁ comienza a los 10. Ambos transmiten durante 100 segundos. No se introducen errores.

En la figura 2 se puede observar un escenario similar en cuanto muestra el throughput de una competencia de dos flujos de TCP Westwood. En este caso, se transmite por tiempo y no por cantidad de paquetes, sin errores y el segundo tráfico comienza 5 segundos después del primero. Al igual que en el escenario 1, se ve que no tarda mucho en estabilizarse la simulación y la competencia por recursos es pareja.

Se utiliza un período más extenso para poder minimizar el efecto de la disparidad inicial y para poder tomar un recorte de tiempos intermedios con una cantidad suficiente de datos. Ambos escenarios pertenecen a pruebas diferentes. Fueron seleccionados para ilustrar la estrategia de recorte para trabajar con datos del sistema en estado estable. Se utilizó la misma variante de TCP para probar que la razón de uso de recursos se mantiene sin importar la diferencia de tiempo, y el sistema, llega al mismo estado de equilibrio.

- Escenario 3:

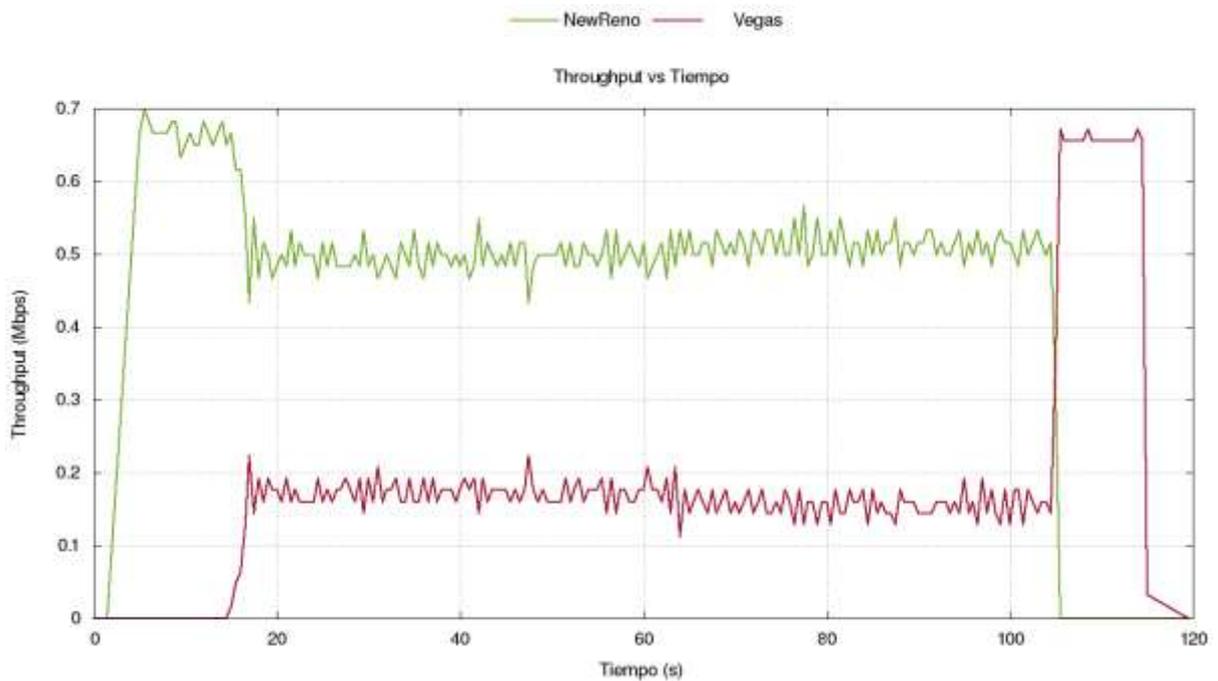


Figura 3. Throughput vs Tiempo. TCP New Reno vs TCP Vegas en competencia. New Reno comienza a los 5 segundos, y Vegas comienza a los 10. Ambos transmiten durante 100 segundos. No se introducen errores.

Al hacer competir diferentes variantes de TCP pueden verse resultados más interesantes. Por ejemplo, la figura 3 muestra la competencia entre TCP New Reno, que debido a su carácter reactivo suele ocupar todo el ancho de banda posible; y TCP Vegas, que utiliza un enfoque más conservador por lo que sólo puede ocupar toda la capacidad de canal cuando la otra variante deja de transmitir.

- Escenario 4:

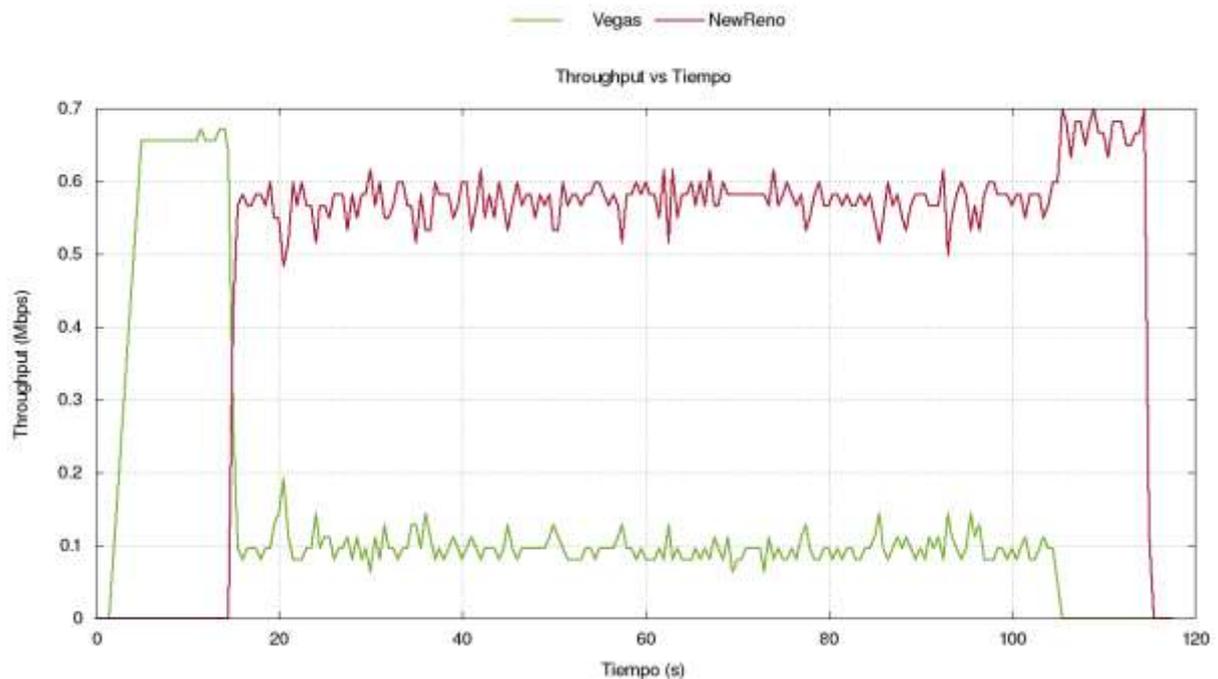


Figura 4. Throughput vs Tiempo. TCP Vegas vs TCP New reno en competencia. Vegas comienza a los 5 segundos, y New Reno comienza a los 10. Ambos transmiten durante 100 segundos. No se introducen errores.

La figura 4 muestra un escenario similar al de la figura 3, con el orden de inicio alterado. Se puede observar que aunque TCP Vegas comienza primero, al estabilizarse se llega a la misma situación: la proporción de uso de ancho de banda se inclina a favor del protocolo más agresivo. Sin embargo, si el protocolo más agresivo comienza después, ocupa un mayor ancho de banda, evidenciado por una diferencia de 0,1 Mbps entre el throughput de la figura 2 y 3.

Interrupción de tráfico vs movimiento: entradas y salidas del rango de cobertura

A continuación se muestra como se ve afectada la transmisión de paquetes por distintos tipos de errores.

El problema de las interrupciones de conexión en redes inalámbricas debidas a entradas y salidas del rango de cobertura es multivariado. Debido a esto, incrementa la complejidad de su estudio, en particular cuando se desea evaluar el rendimiento de un protocolo como TCP: por un lado, debe hacerse un análisis para determinar la influencia del movimiento (distancia variable entre los nodos, velocidad no constante, etc.) en el tiempo de transmisión y en consecuencia, en el comportamiento del protocolo en función de su algoritmo de control de congestión; por otro, debe diferenciarse entre las pérdidas de paquetes por corte de conexión, interferencia y por agotamiento del tiempo de espera. Adicionalmente hay consideraciones sobre la intensidad de la señal, el tipo de antena utilizada, los protocolos que complementan a TCP en otras capas y los algoritmos que utilizan, los dispositivos, etc. Es decir, hay muchas variables que influyen en la respuesta del protocolo, por lo que se dificulta aislar la fuente del problema para plantear alguna solución.

La comprensión de un determinado fenómeno es una de las metas al simular un escenario. Para ello se busca simplificar el escenario tanto como sea posible para poder

estudiar una variable a la vez. Como se mencionó, el problema se debe, al menos en parte, a la movilidad y a la interrupción del tráfico.

En los ejemplos mostrados se busca enfatizar los tipos de errores que pueden aparecer: errores estadísticos (que pueden representar pérdidas por interferencia, por congestión, etc.), errores en ráfaga (adecuados para estudiar salidas del rango de cobertura lo suficientemente extensas como para causar pérdidas continuas, pero no para desconectar el dispositivo), y errores de corte de línea (en las que el dispositivo debe restablecer la conexión una vez que pasa un determinado tiempo). Vale notar que son aproximaciones basadas en el criterio de los investigadores, por lo que el conocimiento que pueda derivarse sobre el comportamiento del protocolo en estos escenarios debe ir acompañado de una aclaración.

- *Escenario 5:*

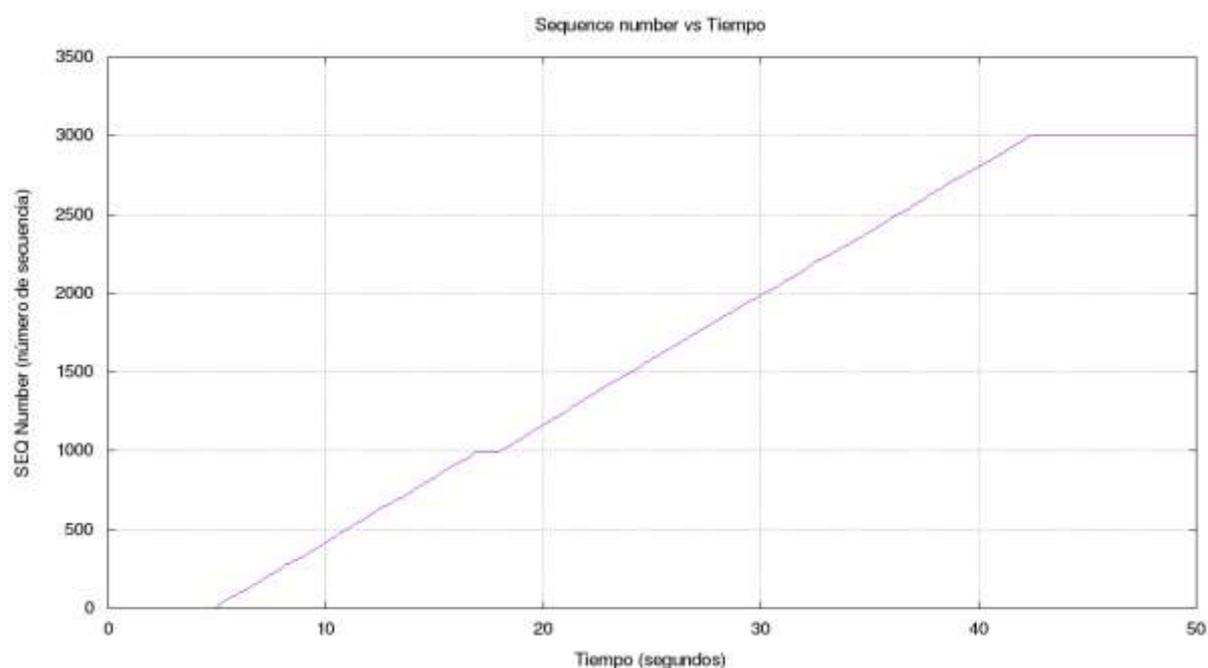


Figura 5. Número de secuencia. TCP Cubic con error de corte de línea durante 0,8 segundos.

La figura 5 muestra un error de corte de línea, en la que se interrumpe todo tipo de tráfico durante un determinado período de tiempo. Esto resulta en una línea recta en el gráfico de número de secuencia cerca de los 18 segundos.

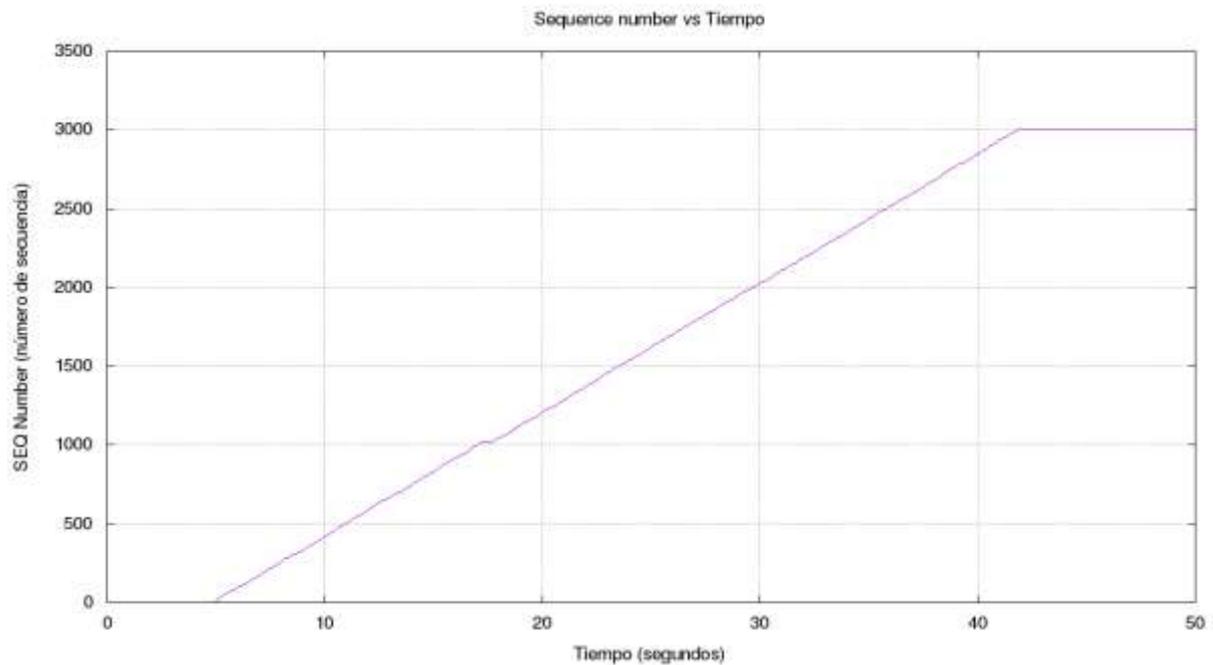
- *Escenario 6:*

Figura 6. Número de secuencia. TCP Cubic con error en ráfaga de 15 paquetes.

La figura 6 muestra un error en ráfaga, en donde se pierden 15 paquetes seguidos, pero el resto del tráfico no se ve afectado. El efecto en la curva es similar al de la figura 5, aunque tarda más en quedar llana, y lo hace durante menos tiempo.

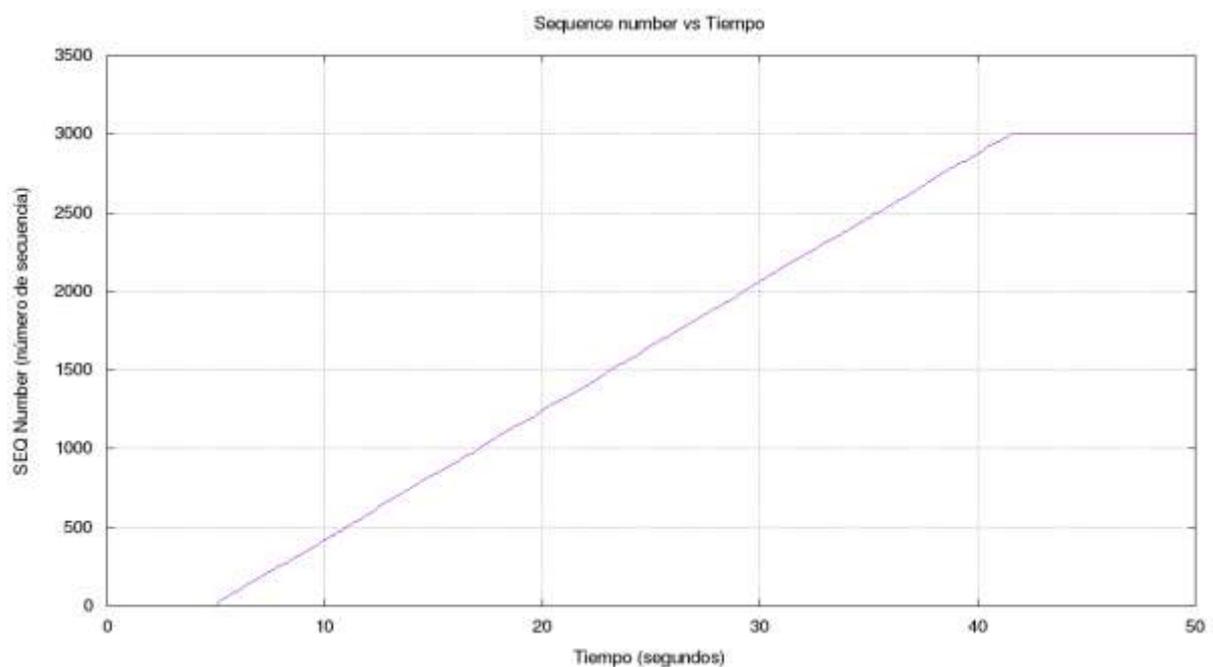
- *Escenario 7:*

Figura 7. Número de secuencia. TCP Cubic con error estadístico de 0,1%

La figura 5 muestra un error estadístico, en el que se pierde uno de cada 1000 paquetes. Este efecto es apenas perceptible en secciones escalonadas de la curva.

Para poder utilizar este tipo de errores para representar un determinado fenómeno, debe entenderse cuál es el efecto que éste tendrá sobre el tráfico y cómo reaccionarán los protocolos en uso.

Pasos del estudio de simulación

Existen muchas formas de evaluar el rendimiento de TCP, pero todas ellas dependen de una metodología adecuada en la simulación. Para evaluar determinados aspectos de un problema, se debe diseñar un modelo que reproduzca adecuada o aproximadamente el fenómeno de interés; caso contrario, los métodos de evaluación no servirán para detectar fallas o aspectos a mejorar.

Datos de simulación estabilizada:

- Definición del sistema:

En la actualidad, la mayoría de las redes de computadoras conectan múltiples dispositivos, por lo que se debe lidiar con el problema de recursos compartidos. Esto implica la necesidad de comprender el comportamiento de los protocolos involucrados en la transmisión de datos, ante escenarios de competencia por recursos. En el modelo TCP/IP, las capas interactúan entre sí a través de interfaces, por lo que pueden realizarse cambios en cada una, sin afectar al resto. Gracias a esta independencia se puede estudiar el rendimiento de un protocolo perteneciente a la capa de transporte sin mucho énfasis en el resto de los niveles.

Al estudiar la competencia por recursos, enfocada en el rendimiento de TCP, puede resultar útil entender el efecto que tienen los estados de transición: qué pasa cuando un tráfico empieza antes que otro; qué pasa cuando varios tráficos comienzan en simultáneo; qué ocurre cuando se enfrentan distintas variantes de TCP; qué consecuencias tiene el enfrentamiento entre variantes que utilizan estrategias de control de congestión agresiva y variantes que optan por enfoques más conservadores; cuanto tiempo tarda en estabilizarse un sistema luego de un estado de transición; etc.

Se busca representar un sistema que permita estudiar la competencia por recursos en una red. Se deben contemplar las etapas de transición; la comparación debe surgir de un estado estable. Debido al enfoque del proyecto, interesa que la red estudiada incluya enlaces inalámbricos.

- Formulación del modelo:

En base a experiencias anteriores, se optó por un modelo clásico para la comparación de tráficos, conocido como "modelo X" (figura 8). Éste consiste en 5 nodos, uno de los cuales (2) funciona de intermediario para la comunicación entre el resto. Los nodos 0 y 1 están se comunican con el 2 a través de conexiones cableadas; los nodos 3 y 4 tienen conexiones inalámbricas. Los tráficos en competencia se establecen: del nodo 0 al 3 y del nodo 1 al 4. Los tráficos pueden corresponder a una misma variante de TCP, o a diferentes. El tiempo de inicio puede ser el mismo, o no.

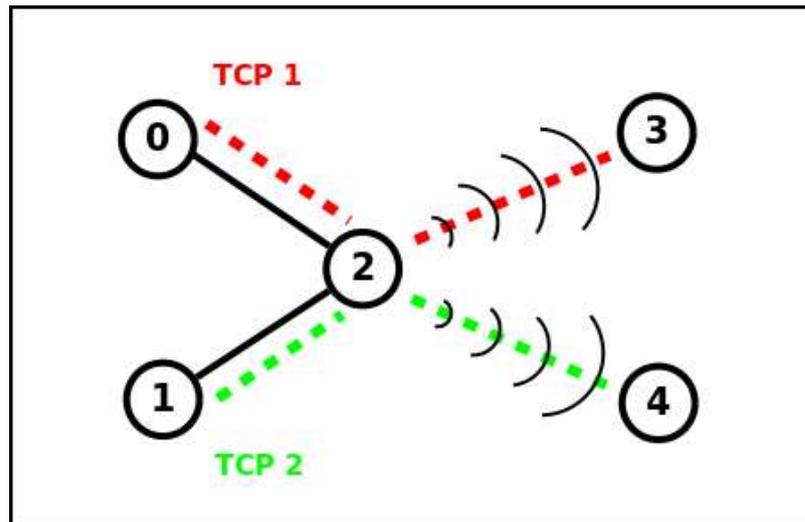


Figura 8. Modelo X elegido para comparar tráficos en competencia por recursos.

- Colección de datos:

Se definió un ancho de banda de 1Mbps (valor predeterminado para conexiones inalámbricas en NS2) como cuello de botella, y una transmisión de 3000 paquetes generados por una aplicación de tipo FTP. La variable de interés es el tiempo; se definen dos casos: ambos tráficos comienzan a los 5 segundos; un tráfico comienza a los 5 segundos, y el otro comienza a los 10. Se busca medir el tiempo que el sistema tarda en llegar a una transmisión estable, y una vez fuera de la etapa de transición, obtener parámetros de rendimiento de TCP relativos entre las variantes comparadas.

- Implementación del modelo en la computadora:

La simulación propiamente dicha se implementa en OTcl. Los parámetros de rendimiento se obtienen a través de scripts en awk. Los gráficos se generan por lote con gnuplot. Todo el proceso se automatiza con scripts en bash. Las pruebas consisten en todo el proceso de simular, procesar y mostrar los resultados. Una descripción completa de la implementación utilizada escapa a los intereses de este trabajo, pero los detalles pueden consultarse en *Automatización del proceso de simulación para el análisis de rendimiento del protocolo TCP en redes inalámbricas* (Trinidad y Talay, 2018).

- Validación:

Durante la implementación del modelo se prueba cada funcionalidad y se verifica que los resultados obtenidos son coherentes con las restricciones teóricas. Por ejemplo, habiendo definido un cuello de botella de 1Mbps, la suma del throughput instantáneo no debe superar 1Mbps en ningún momento. Los gráficos facilitan estas comprobaciones.

- Experimentación:

La ejecución de estas pruebas tardó aproximadamente 2 horas. El lote de pruebas fue significativamente mayor a los casos expuestos. Se compararon entre sí las variantes Vegas, Reno, Cubic, Bic, Westwood, Highspeed, Veno, New Reno, HTCP, Sack, Fack e Hybla, entre

otras. Como resultado se obtuvo una serie de gráficos (entre los que se incluyen los presentados), archivos de traza (registrando el comportamiento de la red en cada simulación).

- Interpretación:

La interpretación, en este caso, fue presentada en resultados y discusión. Para una explicación más detallada y completa, puede consultarse el artículo publicado.

- Documentación:

La documentación consistió en registrar los resultados obtenidos para futuras referencias y en su publicación.

Interrupción de tráfico vs movimiento:

- Definición del sistema:

La transmisión de datos a través de redes de computadoras puede sufrir pérdidas. Esto puede deberse a la competencia por recursos que genera congestión en la red, a problemas en el dispositivo o en el canal, a interferencias externas, salidas del rango de cobertura, fenómenos propios del medio de transmisión, etc. El efecto de cada situación se traduce en distintos tipos de errores. En redes inalámbricas en particular, deben tenerse en cuenta las consecuencias del movimiento de los nodos, principalmente: efectos de hand-off (cuando el dispositivo se conecta a un nuevo punto de acceso inalámbrico más cercano); entradas y salidas del rango de cobertura (al moverse cerca del límite); y variación de la latencia debido a que la distancia al punto de acceso no es fija.

Cuando las salidas del rango de cobertura son lo suficientemente extensas como para producir errores, pero no tanto como para desconectar el dispositivo, se producen pérdidas de paquetes continuas antes de reanudar la transmisión. Esto puede reproducirse mediante la introducción de errores en ráfaga, es decir, la pérdida continua de una cantidad de paquetes en determinado momento. Para reducir el número de variables que pueden influir en el comportamiento de los protocolos, puede eliminarse el movimiento.

Se busca representar un sistema que permita estudiar cómo las pérdidas continuas de paquetes afectan el comportamiento de distintas variantes del protocolo TCP.

- Formulación del modelo:

El modelo elegido en este caso es más sencillo, ya que interesa estudiar el comportamiento de una variable a la vez. Hay tres nodos: Al igual que en el anterior, se utilizan conexiones cableadas e inalámbricas, del nodo 0 al 1 y del nodo 1 al 2, respectivamente.

Este mismo modelo se utilizó anteriormente para estudiar el efecto de errores periódicos, errores estadísticos, errores de corte de conexión y errores en ráfaga. Al elegirlo para el abordar éste fenómeno particular, ya se contaba con experiencia de su uso en casos similares.

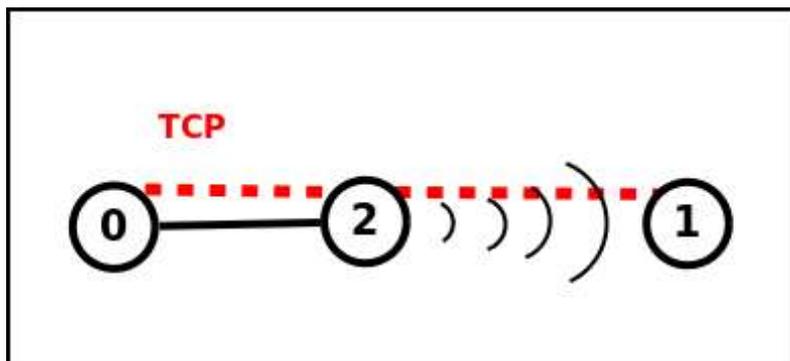


Figura 9. Modelo de tres nodos, elegido para estudiar comportamiento ante errores en ráfaga.

- Colección de datos:

Se determina un cuello de botella de 2Mbps y una transmisión de 3000 paquetes generados por una aplicación de tipo FTP. Interesa saber el tiempo que tarda cada variable en volver a un estado de transmisión estable luego de introducir errores en ráfaga de distinta duración. El número de secuencia resulta particularmente útil para observar dicho comportamiento; al introducir los errores en ráfaga una vez que el número de secuencia llega a 1000, se puede ver fácilmente cuanto tiempo se mantiene constante antes de volver a incrementar.

- Implementación del modelo en la computadora y validación:

Este paso se llevó a cabo de forma similar al caso anterior.

- Experimentación:

La ejecución de estas pruebas tardó aproximadamente 8 horas. Se realizaron pruebas con las variantes Vegas, Reno, Cubic, Bic, Westwood, Highspeed, Veno, New Reno, HTCP, Sack, Fack e Hybla, entre otras. Se hicieron pruebas sin errores como punto de referencia, y con ráfagas de errores de 5, 10, 15, 20 y 25 paquetes.

Como resultado se obtuvo una serie de gráficos (entre los que se incluyen los presentados), archivos de traza (registrando el comportamiento de la red en cada simulación).

- Interpretación:

Parte de la interpretación puede verse en la sección de resultados y discusión. La explicación completa puede encontrarse en el artículo publicado.

- Documentación:

La documentación consistió en registrar los resultados obtenidos para futuras referencias y en su publicación.

5. CONCLUSIONES

Las redes de computadora son inherentemente complejas, por lo que los modelos matemáticos que pueden utilizarse en su estudio resultan útiles únicamente al abordar aspectos específicos. Al diseñar o investigar una red deben tenerse en cuenta los dispositivos que la conforman, el tipo de tecnología que los conecta, los protocolos de comunicación implementados y los algoritmos que utilizan, la clase de aplicaciones empleadas, los potenciales usuarios, el entorno en el que se va a implantar, etc. Hay una cuasi infinidad de sutilezas que definen el carácter de una red, y que sólo pueden dejarse de lado a riesgo de simplificar demasiado el sistema y obtener información errónea del mismo.

Las simulaciones ofrecen información acerca del comportamiento de un sistema bajo determinadas condiciones, la cual alimenta la toma de decisiones estratégica, el conocimiento acerca de un determinado fenómeno, la evaluación de un protocolo, etc.

Es difícil determinar qué herramienta de simulación es mejor. En la mayoría de los casos dependerá de lo que se quiera simular. Sin embargo, el dominio avanzado de un simulador puede resultar más útil que el conocimiento básico de varios, siempre y cuando éste ofrezca las funcionalidades requeridas.

La metodología de trabajo con simuladores puede variar entre distintas áreas, pero hay algunos elementos en común: se debe entender el problema y el sistema; se debe formular y validar un modelo, asegurando que las entradas sean correctas; se deben ejecutar las pruebas y generar los datos; luego se interpretan y documentan los resultados. Este orden garantiza que los resultados sirvan para comprender el problema adecuado y no se generen datos innecesarios.

Durante el proceso de simulación se pone a prueba el dominio del área, particularmente durante el diseño del modelo, su validación y la interpretación de los datos. No basta con saber usar la herramienta de simulación. Si el investigador no entiende conceptos básicos, no podrá contribuir en el desarrollo del modelo a implementar. Si no está familiarizado con la arquitectura del simulador, no podrá reconocer los alcances y limitaciones del mismo.

6. RECOMENDACIONES

Un tema de interés para futuros trabajos sería una comparación de metodologías utilizadas al trabajar con diferentes simuladores. Se especula que la variación es mínima, pero la disponibilidad de documentación puede hacer que se alteren las etapas.

Otra cuestión a investigar es cómo el uso de la simulación como instrumento pedagógico afecta la formación de estudiantes en las universidades.

7. AGRADECIMIENTOS

Se agradece el soporte de la Universidad Nacional de la Patagonia Austral, la supervisión y colaboración del Ing. Diego Rodriguez Herlein, la Lic. Claudia Gonzalez, y María Luz Almada.

6. REFERENCIAS

- AA.VV. DARPA Internet Program, (1981). *RFC 793: Transmission Control Protocol*. Formation Sciences Institute, University of Southern California. Recuperado el 18 de mayo de 2017, de <https://tools.ietf.org/html/rfc793>
- AA.VV. INTERNET ENGINEERING TASK FORCE. (2015). *RFC 7414: A roadmap for Transmission Control Protocol Specification Documents*. Recuperado el 18 de mayo de 2017, de <https://tools.ietf.org/html/rfc7414>
- AA.VV. Real Academia Española. *Diccionario RAE*. En línea: <https://dle.rae.es/?id=Xw4s6f6>. Recuperado el 22 de enero de 2019..
- AA.VV. PROYECTO NS-3. (2016). *ns-3 tutorial 3-29*. En línea: <https://www.nsnam.org/ns-3-27/documentation/>. Recuperado el 22 de enero de 2019.
- ALCARAZ ESPÍN, J., EGEA LÓPEZ, E., GARCÍA HARO, J. *Curso de Simulación de Redes de Comunicaciones: Planeamiento, objetivo y metodología*. Departamento de tecnologías de la Información y Comunicaciones. <https://doi.org/10.15425/redecom.14.2015.07>
- ALLMAN, M., FALK, A. (1999). *On the effective evaluation of TCP*. ACM Computer Communication Review. Recuperado el 26 de marzo de 2017, de <https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/200000059208.pdf>
- CARAVAJAL, L. *Observación y experimentación científicas*. Recuperado el 20 de enero de 2019, de <https://www.lizardo-carvajal.com/observacion-y-experimentacion-cientificas/>.
- CEVALLOS VILLACRÉS, Y. E., LARA, L. R. (2010) *Incidencia del uso del simulador de redes de computadoras Packet Tracer de CISCO Systems en el desarrollo de competencias procedimentales*. Tesis de maestría en tecnología de la información y multimedia educativa. Facultad de Ciencias Humanas y de la educación. Universidad Técnica de ámbito. Ecuador. <https://doi.org/10.1344/ridas2019.7.7>
- COSS BU, R. (2005). *Simulación. Un enfoque práctico. 2da edición*. Editorial Limusa S.A. México: 2005. ISBN: 9681815068.
- ELAARAG, H. (2002). *Improving TCP Performance over Mobile Networks*. Revista ACM Computing Surveys, Vol. 34, No. 3, September 2002, pp. 357–374. <https://doi.org/10.1145/568522.568524>
- FORUZAN B. A., FEGAN, S. (2007). *Transmisión de datos y redes de computadoras (4ta. Edición)*. New York: McGraw-Hill. ISBN 978-0-07-296775-3
- GARCÍA DÁVALOS, A., ESCOBAR PAZ, L. M., NAVARRO CADAVID, A., Y VÁSQUEZ MEJÍA, A. *Método de evaluación y selección de herramientas de simulación de redes*. Artículo corto. Revista Sistemas y Telemática. Vol. 9, No. 16, 55-71. Universidad Icesi. Colombia: 2011. <https://doi.org/10.18046/syt.v9i16.1029>
- GASS, S. I., ASSAD, A. D. (2005) *Modern World: Tales from the Time Line – The definition of OR and the Origins of the Monte Carlo Simulation*. Revista Interfaces, Vol. 35, No. 5, Septiembre-Octubre 2005, pp. 429-435. <https://doi.org/10.1287/inte.1050.0160>
- HERNÁNDEZ GALVÁN, M. I., GARCÍA REYES, N. Y PÉREZ, I. (2007). *Simulación de un laboratorio de redes de área local*. Tesis de fin de carrera, Licenciatura en sistemas computacionales. Instituto de Ciencias básicas e Ingeniería (ICBI), Universidad Autónoma del Estado de Hidalgo. México. <https://doi.org/10.15415/jnp.2016.41006>
- HERRERA, O. J., BECERRA, L. A. (2014). *Diseño general de las etapas de simulación de procesos con énfasis en el análisis de entrada*. XII Latin American and Caribbean Conference for Engineering and Technology (LACCEI '2014): Ecuador.
- INGALLS, R. (2008). *Introduction to simulation*. Proceedings of 2008 Winter of Simulation Conference. IEEE 2008.

- ISSARIYAKUL, T., HOSSAIN, E. (2012). *Introduction to Network Simulartor NS2 (2nd edition)*. Springer: New York. 978-1-4614-1405-6
- LORENTE, O., DE CERIO, D. (2005). *Implementación de un modelo de canal inalámbrico para redes 802.11 bajo el simulador ns-2*. Escuela Politécnica Superior de Catalunya.
- MINAKOV, I., PASSERONE, R., RIZZARDI, A., SICARI, S. (2016). *A Comparative Study of Recent Wireless Sensor Network Simulators*. ACM Transactions on Sensor Networks, Vol. 12, No. 3, Article 20. <https://doi.org/10.1145/2903144>
- PANDEY, S., TYAGI, V. (2013). *Performance Analysis of Wired and Wireless Networks using NS2 Simulator*. International Journal of Computer Applications. Volume72–No.21, June 2013.
- PENTIKOUSIS, K. (2001). *TCP in wired-cum-wireless environments*. State University of New York. Publicado en IEEE Communications Surveys & Tutorials. Recuperado el 26 de marzo de 2017, de https://www.researchgate.net/publication/2356074_TCP_in_Wired-cum-Wireless_Environments. <https://doi.org/10.1109/comst.2000.5340805>
- RODRÍGUEZ HERLEIN, D. M. TALAY C. A., GONZÁLEZ C. N., TRINIDAD F. A., MARRONE, L. (2018). *Performance of TCP Vegas According to Alfa and Beta Parameters in Hybrid Scenarios with Bursts Errors*. Presentado en International Conference on Information Systems and Computer Science (INCISCOS), Quito (Ecuador) año 2017. Publicado en IEEE Xplore. Artículo: <https://ieeexplore.ieee.org/document/8328110> <https://doi.org/10.1109/inciscos.2017.16>
- RODRÍGUEZ HERLEIN, D. R., TALAY, C. A., GONZÁLEZ, C. N., TRINIDAD, F. A. (2018). *Contienda entre las variantes del protocolo TCP Vegas y Reno por los recursos de la red en un modelo híbrido simple*. Presentado en el XX Workshop de Investigadores en Ciencias de la Computación(WICC). Corrientes, Argentina. Artículo: <http://hdl.handle.net/10915/67209> <https://doi.org/10.4206/rev.austral.cienc.soc.2019.n36-08>
- RODRÍGUEZ HERLEIN, D. R., TALAY, C. A., GONZÁLEZ, C. N., TRINIDAD, F. A.. (2017). *Consideraciones sobre el comportamiento del protocolo TCP en sus variantes Vegas, Reno, Cubic y Westwood ante errores en ráfaga en una topología híbrida*. Presentado en el XXIII Congreso Argentino de Ciencias de la Computación (CACIC). La Plata, Argentina. Artículo: <http://hdl.handle.net/10915/63846> <https://doi.org/10.4206/rev.austral.cienc.soc.2019.n36-08>
- SCHILLING, B. (2005). *Qualitative Comparison of Network Simulation Tools*. Technical report, Institute of Parallel and Distributed Systems (IPVS), University of Stuttgar. Recuperado el 22 de enero de 2019, de <https://pdfs.semanticscholar.org/f577/8c8218964b222ab048bd8d2e7ea145cbe52a.pdf>.
- STALLINGS W. (2002). *Comunicaciones y Redes de Computadoras (6ta edición)*. Prentice-Hall.
- TALAY C. A, TRINIDAD F. A., RODRÍGUEZ HERLEIN, D. R., ALMADA M.L, GONZÁLEZ, C. N, MARRONE, L. A. (2018). *Analysis of the performance of TCP Vegas and its relationship with alpha and beta parameters in a wireless links network and burst errors*. Presentado en Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI), Buenos Aires (Argentina). Publicado en IEEE Xplore. Artículo: <https://ieeexplore.ieee.org/document/8584350> <https://doi.org/10.1109/cacidi.2018.8584350>
- TERRAZAS PASTOR, R. (2010). *Aplicación de la simulación a un sistema de colas de canal simple*. Revista Perspectivas, num. 26. pp. 99-112. Universidad Católica Boliviana San Pablo. Bolivia.

- TRINIDAD F. A, GONZALEZ, C. N, TALAY, C. A, RODRÍGUEZ HERLEIN, D. R. (2018). *Automatización del proceso de simulación para el análisis de rendimiento del protocolo TCP en redes inalámbricas*. Publicado en el volumen 10 de la revista de Informes Científico Técnicos de la Universidad Nacional de la Patagonia Austral (Código ICT-UNPA-186-2018). <https://doi.org/10.22305/ict-unpa.v10i3.284>
- TSAOUSSIDIS, V., MATTA, I., (2001). *Open Issues on TCP for Mobile Computing*. Boston University and Northeastern University. Recuperado el 26 de marzo de 2017, de <http://www.cs.bu.edu/techreports/pdf/2001-013-open-issues-tcp-wireless.pdf> <https://doi.org/10.1002/wcm.30>
- XYLOMENOS G., MÄHÖNEN P., SAARANEN, M., (2001). *TCP performance issues over Wireless link*. Recuperado el 8 de junio de 2017, de <https://cseweb.ucsd.edu/groups/csl/pubs/journal/commag01.pdf>. S/R. <https://doi.org/10.1109/35.917504>