

Evaluación de Algoritmos de Ruteo en Redes de Computadoras

Edilma Olinda Gagliardi⁽¹⁾

Mario Marcelo Berón⁽²⁾

⁽¹⁾LIDIC

⁽²⁾Ingeniería del Software Métodos y Herramientas
Facultad de Ciencias Físico Matemáticas y Naturales
Universidad Nacional de San Luis – Argentina
{oli, mberon}@unsl.edu.ar

Gregorio Hernández Peñalver

Departamento de Matemática Aplicada

Facultad de Informática

Universidad Politécnica de Madrid – España

gregorio@fi.upm.es

Resumen

En este trabajo, se presenta una línea de investigación que se encarga del análisis de algoritmos de ruteo de paquetes en una red de computadoras. Nuestra herramienta trabaja en la capa de aplicación dentro de un protocolo de redes, obteniendo una colección de métricas que determinan el mejor ruteo de paquetes en cada momento. Presentamos los aspectos más relevantes del diseño de la herramienta, sus alcances y futuras etapas en este proyecto, como así también enlaces de interés.

Palabras Claves: Ruteo con brújula. Redes sin cables. Sistemas Operativos.

Introducción

La geometría computacional [2,5,14,15] se encarga de resolver problemas geométricos de modo constructivo. Se interesa por demostrar la existencia de la solución a un problema y por encontrar los algoritmos y estructuras [3,4,6,13] de datos eficientes, medidos por su complejidad espacial y temporal. En ocasiones, la Geometría brinda soluciones más eficientes en problemas que no parecen ser geométricos, tal es el caso del *ruteo de paquetes* en una red de computadoras. Si bien este es un tema que ha sido estudiado debido al gran uso de las redes de computadoras, en la actualidad se investiga la forma de reducir los costos en cuanto ha espacio y tiempo de los algoritmos de enrutamiento. Es aquí donde entra en juego la aplicación de la Geometría Computacional y la teoría de grafos, ya que permite modelar fielmente a una red de computadoras.

Se han definido un conjunto de algoritmos que proponen una nueva forma de enviar un paquete desde una estación de trabajo S_1 a otra S_2 [7,8,11,12]. Estos algoritmos usan información geográfica como, por ejemplo, latitud y longitud, para detectar dónde se encuentra la estación de trabajo destino. Teniendo en cuenta que en la actualidad se dispone de un dispositivo que permite recuperar de manera instantánea esta información (*Global Position Systems - GPS*), estos algoritmos adquieren importancia debido a que pueden representar una mejor alternativa a los ya existentes.

El algoritmo de *Ruteo por Brújula* [10] adquiere importancia ya que propone una forma de enviar un paquete que intenta minimizar el uso de almacenamiento. En términos matemáticos podemos modelizar este tipo de problemas como grafos geométricos planos. Este algoritmo trabaja de la siguiente forma: desde un vértice inicial s se desea viajar hasta un vértice destino d de un grafo geométrico. La información disponible en cada nodo es local, sólo se conoce las coordenadas del punto y las de sus vecinos inmediatos en el grafo. Cuando se alcanza un nodo v , continúa por la arista incidente a v cuya pendiente es la más próxima a la pendiente del segmento que conecta a v con el nodo destino d .

Si bien el Ruteo por Brújula, al igual que otros algoritmos tales como el *Ruteo por Brújula Aleatorizado*, *Ruteo Ávido*, entre otros, brindan una forma diferente de concebir el envío de paquetes, es importante contar con una herramienta que posibilite su evaluación y comparación con los algoritmos de enrutamiento existentes.

En este sentido, a partir de las nociones de ruteo, hemos concebido un *Simulador*. Esta herramienta trabaja en la capa de aplicación dentro de un protocolo de redes, obteniendo una colección de métricas que determinan el mejor ruteo de paquetes en cada momento. Nuestro trabajo consiste en el desarrollo de esta herramienta, para la evaluación de algoritmos de ruteo de paquetes en redes, con el fin de analizar las bondades y dificultades de estos nuevos algoritmos.

Concepción del Simulador

La estructura del simulador a alto nivel de abstracción se muestra en la figura 1:

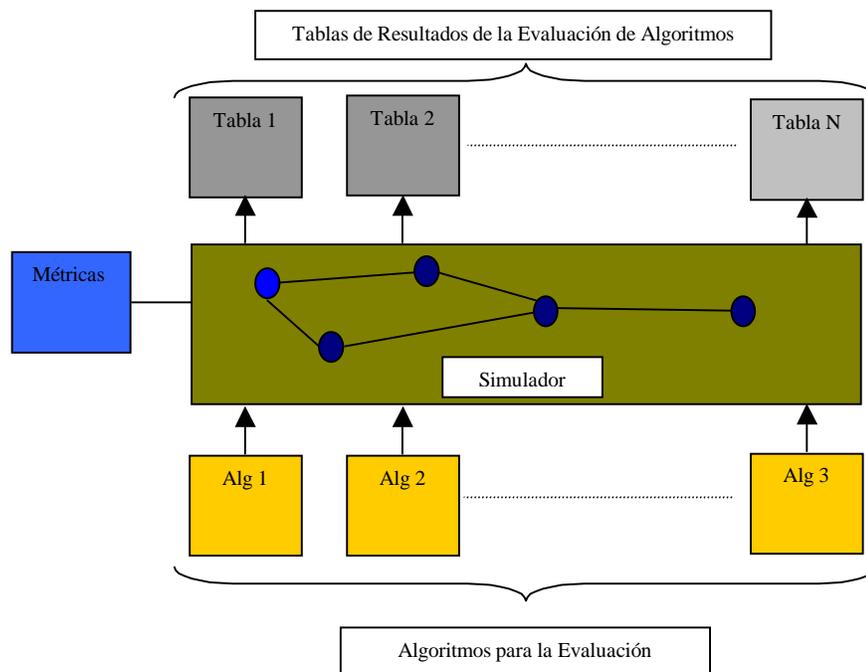


Figura 1: Abstracción del Simulador

La principal componente de la herramienta es el módulo *Simulador*, el cual define red de computadoras virtual como un grafo dirigido $G=(P,E)$, donde P es el conjunto de computadoras que conforman la red y E es la relación de comunicación entre ellas. Esta componente se encarga de simular el envío de mensajes en la red virtual existente.

Este simulador tiene dos posibilidades para llevar a cabo el estudio del comportamiento de los algoritmos a evaluar: una interfaz gráfica y un modo automático de evaluación.

La interfaz gráfica se encarga de visualizar la red de computadoras y de mostrar los nodos visitados por el algoritmo de enrutamiento que se está evaluando. Esta interfaz da la posibilidad de ejecutar el algoritmo en su totalidad o paso a paso; éste último modo de operación facilita el estudio del comportamiento de algoritmos particulares y la observación de los valores de las métricas que están siendo evaluadas.

El modo automático de evaluación está pensado para ejecutar cada uno de los algoritmos disponibles en el simulador para un gran número de casos de prueba (*Algoritmos para la evaluación*); de esta forma, se recolectan los valores que toman las métricas que se desean evaluar. Como resultado de este proceso el simulador emite un conjunto de relaciones (*Tablas de Resultados de la Evaluación de Algoritmos*), donde se pueden observar los resultados de las métricas.

Se desea evaluar, en principio, las métricas: *Sobrecarga de nodos*, *distancia de enlace* y *Sobrecarga del centro de la red*.

Los algoritmos implementados para esta tarea fueron los siguientes: *Ruteo por Brújula*, *Búsqueda de pasos (en Profundidad y por Niveles)*, *Dijkstra* y *Centro del grafo*[1], con el objetivo de poder medir la distancia de enlaces de los algoritmos a ser evaluados y detectar si los algoritmos de enrutamiento sobrecargan el centro del grafo.

La Estructura Interna del Simulador

En la figura 2 se presenta un esquema del simulador.

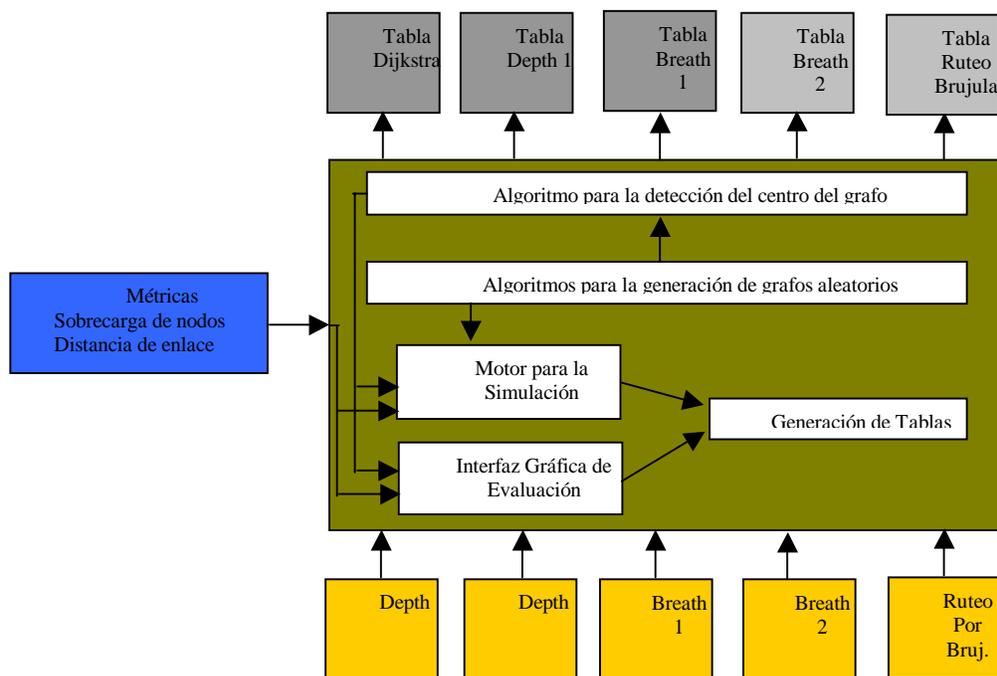


Figura 2: Visión Interna del Simulador

La interfaz gráfica y el modo automático de evaluación fueron descritos en la sección previa, pero éste último merece un explicación en mayor detalle.

Para realizar la evaluación de manera automática se necesita una estrategia de generación de grafos aleatorios, algoritmos para la detección de los nodos centrales del grafo y un motor para la simulación.

En cuanto a la primera, el simulador provee dos estrategias, las cuales tienen en común la generación de los nodos del grafo (se utiliza el generador de números aleatorios del lenguaje de

programación para obtener las coordenadas x e y de cada uno de los nodos de la red de computadoras), y difieren en la construcción de la relación que vincula los nodos.

El primer método consiste en asignar a cada arco del grafo una probabilidad y luego establecer un rango de probabilidades en forma aleatoria. Los arcos que forman parte de la relación son aquellos que se encuentran en el rango de probabilidad establecido.

El segundo método consiste en generar una distancia n , así los arcos que forma parte de la relación son aquellos que unen nodos cuya relación de distancia es menor o igual que esa distancia n .

Con el fin de medir la sobrecarga de los nodos en la red, se implementaron el algoritmo de búsqueda de pasos propuesto por *Floyd* y los procedimientos que toman como entrada este resultado y realizan el cálculo de los nodos centrales de un grafo. Estos algoritmos se requieren para validar la hipótesis de que los algoritmos de pasos incorporados en el simulador sobrecargan el centro del grafo.

El comportamiento del motor de la simulación consiste en la generación aleatoria los nodos origen y destino, para la transmisión del mensaje; además, realiza la recolección de datos para luego enviar esta información hacia el módulo que se encarga de generar las tablas que son las salidas del simulador.

Conclusiones y Visión de Futuro

Si bien el diseño e implementación de las rutinas del simulador no presentan un alto grado de complejidad, es importante señalar que éste es un prototipo, ya que se pretende que el simulador pueda evaluar una mayor cantidad de métricas y además compruebe el comportamiento de algoritmos de ruteo que se utilizan en la redes móviles.

Por otra parte, en esta primera versión del prototipo, el mensaje que se envía es de una unidad, pero cabe destacar que en la realidad los mensajes están formados por un conjunto de paquetes. Este hecho dificulta aún más el diseño e implementación del motor de la simulación ya que se debe pensar si los segmentos de paquetes que conforman un mensaje llegan por un mismo camino o por otros diferentes a la máquina destino y si, además, se contempla la posibilidad de que los mensajes puedan alterarse por errores producidos durante su viaje desde el nodo origen al nodo destino. Esto representa un gran desafío para este grupo de investigación y abre un abanico de posibilidades para la realización de trabajos futuros.

De las conclusiones y las futuras etapas del proyecto, se definen diferentes líneas de trabajo interdisciplinario. Una línea dentro de la Geometría Computacional y Teoría de grafos con el objetivo de la definición de modelos analíticos, estudio de técnicas algorítmicas y estructuras de datos adecuadas. Una línea en el desarrollo de software para la implementación de los modelos analíticos de la línea anterior con el objetivo de probar empíricamente las hipótesis planteadas. Una línea dentro de la Ingeniería de Software con el fin de aplicar Ingeniería Inversa e Ingeniería de Software para realizar el desarrollo de los módulos necesarios dentro de marcos formales en Ciencias de La Computación. Y por último una línea de aplicación de teoría de Sistemas Operativos y Redes.

Bibliografía

1. Aho, A.V.; Hopcroft, J. E.; Ullman, J. *The desing and analysis of computer algorithms*, Addison-Wesley Series in Computer Science and Information Processing, 1974.

2. Berg, Kreveld, Overmars, Schwarzkopf. *Computational Geometry: algorithms and applications*, Springer Verlag, BH 1997
3. Boissonnat, J.D.; Yvinec, M. *Algorithmic Geometry*, Cambridge University Press, 1998.
4. Gonnnet G.;Baeza Yates, R. *Handbook of Algorithms and Data Structure*, Addison – Wesley, New York, 1991.
5. Goodman, J.; O'Rourke, J. *Handbook of Discrete and Computational Geometry*. CRC Press 1997.
6. Günther, O. *Efficient Structures For Geometric Data Management*. Springer Verlag 1988.
7. I. Heywood, S. Corneluis, S. Carver, *Geographical Information Systems*, Addison-Wesley Longman, New York, 1998.
8. J. Serra , *Image Analysis and Mathematical Morphology*, Academic Press, New York, 1982.
9. J. Serra, *Image Analysis and Mathematical Morphology, Vol II: Theoretical Advances*. Academic Press, New York, 1988.
10. Kranakis E., Singh H., J. Urrutia. *Compass rotuing on Geometric network*,2000.
11. Kurt Melhorn *Multidimensional Searching and Computational Geometry* Springer Verlag 1984.
12. Latombe J.C., *Robot Motion Planning*, Kluwer Academic Publisher, Boston, MA, 1991.
13. Pach, J. *New trends in Discrete and Gomputational Geometry*. Springer Verlag NY 1993.
14. Preparata, F.; Shamos,M. *Computational Geometry: an introduction*, /Springer Verlag, NY 1985
15. Sack, J.R.; Urrutia, J.. *Handbook of Computational Geometry*. Elsevier Science B.V. 2000.