

# Estrategias de ruteo alternativas para redes móviles

Berón, Mario Marcelo

Hernández Peñalver, Gregorio

Gagliardi, Edilma Olinda

Departamento de Informática

Departamento de Matemática Aplicada

Facultad de Ciencias Físico, Matemáticas y Naturales

Facultad de Informática

Universidad Nacional de San Luis – Argentina

Universidad Politécnica de Madrid – España

e-mail: {mberon,oli}@unsl.edu.ar

e-mail: gregorio@fi.upm.es

Fax: 54-2652-430224

Fax: 34-91-3367426

## Resumen

El ruteo de paquetes en una red de computadoras móviles presenta un desafío de investigación. Esto se debe a que este tipo de redes carece de una infraestructura física debido a la movilidad constante de las máquinas que la componen. Esta característica inhibe el uso de los algoritmos de ruteo tradicionales y motivan el desarrollo de nuevas estrategias que permitan el intercambio eficiente de paquetes.

Los avances tecnológicos han posibilitado el desarrollo de métodos de ruteo teóricos que se basan en la utilización de información geográfica para encontrar el camino entre un nodo origen y otro destino. Algunos de estos algoritmos pertenecen al campo de la Geometría Computacional y son de mucha importancia porque utilizan escasos recursos y proponen una solución elegante al problema del envío y recepción de paquetes en una red móvil.

En este artículo, presentamos los resultados obtenidos a través de la evaluación de estrategias de ruteo aplicadas sobre clases de grafos que modelan adecuadamente las conexiones en una red móvil.

Palabras claves: Ruteo, grafos, redes móviles, criterios de evaluación.

## 1. INTRODUCCIÓN

Una de las características principales de las redes móviles es el intercambio de información que se lleva a cabo entre las máquinas que las componen. La información que se desea transmitir está dividida en mensajes y estos a su vez en paquetes. Los paquetes son los que se transmiten y permiten la recuperación de la información por medio de un proceso de ensamble. Para enviar un paquete desde una máquina origen a otra destino se necesita de un *Algoritmo de Ruteo*. El algoritmo de ruteo es un software que se encarga de encontrar el camino existente entre un par de máquinas pertenecientes a una red. Si bien actualmente existen una gran variedad de algoritmos su diseño e implementación sigue siendo un tema de investigación. Esto se debe a que los avances tecnológicos han inhibido el uso de los algoritmos de ruteo que proporcionaban una solución adecuada y eficiente al problema del envío de paquetes. Esta inhibición se produce porque se desconoce las topologías de algunas redes, como por ejemplo las móviles. El algoritmo de ruteo entonces debe enviar el paquete en un ambiente que en el mejor de los casos será parcialmente conocido. Como podemos observar, la construcción de algoritmos de ruteo es una tarea compleja que fuerza al programador a compenetrarse con el funcionamiento de la red y a diseñar estrategias sencillas, robustas, equitativas y óptimas con el fin de aumentar el desempeño de la misma.

En este artículo describimos las nuevas tendencias de algoritmos de ruteo [BBC+01][DSW01][CDK+04] cuya finalidad es enviar paquetes en ambientes desconocidos o parcialmente conocidos con una mínima utilización de recursos. Además, mostramos los resultados obtenidos al evaluar el comportamiento de estas estrategias de ruteo cuando son aplicadas sobre clases de grafos que modelan adecuadamente a las redes móviles [PGB03] [GTB02]. Dichos resultados fueron conseguidos utilizando un Evaluador de Algoritmos de Ruteo de nuestra creación.

Los detalles de esta herramienta pueden ser vistos en [BPG05]. En las secciones siguientes las máquinas serán referenciadas como nodos y las conexiones entre ellas como arcos.

El artículo está organizado como sigue. En primer lugar describimos las estrategias de ruteo. Luego presentamos las distintas clases de grafos sobre las cuales se ejecutaron las estrategias de ruteo. Como paso siguiente definimos los criterios de evaluación y exponemos los resultados. Finalmente presentamos las conclusiones.

## 2. LOS ALGORITMOS DE RUTEO

En esta sección describimos las estrategias de ruteo. En términos generales podemos decir que esta clases de algoritmos de ruteo tienen las siguientes propiedades: i) *información local*: cada nodo sólo conoce la posición de sus vecinos; ii) *memoria limitada*: sólo se recuerda un número constante de nodos ya visitados, así como las coordenadas del origen y el destino; iii) *ecológico*: no se permite dejar marcas en los nodos visitados; iv) *decisiones locales*: la elección del camino a seguir en cada nodo se basa exclusivamente en la información local almacenada en el nodo y en la información que lleva el objeto que recorre el camino (de tamaño constante). Además es importante mencionar que las estrategias de ruteo presentadas en este artículo funcionan cuando el grafo sobre el cual se aplican es geométrico y plano.

### 2.1 RUTEO VORAZ

El ruteo voraz (greedy routing) funciona como sigue. Suponga que el nodo  $o$  desea enviar un paquete al nodo  $d$ . El nodo  $o$  sólo conoce la las coordenadas de sus vecinos y del destino. El ruteo voraz envía el paquete al vecino de  $o$  que minimiza la distancia euclidiana con el nodo  $d$ . Cuando más de un nodo se encuentran a la misma distancia hacia el nodo destino, el ruteo voraz selecciona uno de ellos al azar. La figura 1 muestra como actúa la función de selección del próximo nodo sobre una porción de un grafo geométrico plano.

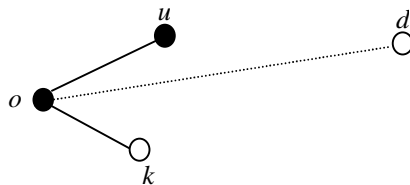


Figura 1: Selección del próximo nodo con el ruteo voraz

El ruteo voraz presenta una serie de inconvenientes. En primer lugar este algoritmo necesita que el grafo sobre el cual se aplica sea plano, es decir, los arcos no se cortan. Esta situación nunca o casi nunca aparece en las comunicaciones móviles reales.

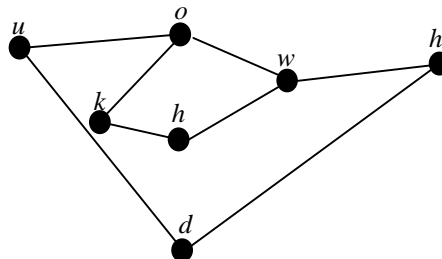


Figura 2: El ruteo voraz fracasa en entregar un paquete desde el nodo  $o$  al nodo  $d$

Por otra parte, el ruteo voraz presenta el problema de no encontrar el camino para todo grafo geométrico plano. Observe la figura 2 y asuma que el nodo  $o$  desea enviar un paquete a  $d$ . El nodo  $o$  calcula sus vecinos y analiza cual de ellos está más próximo al destino  $d$ . El resultado de esta operación es el nodo  $k$ . El nodo  $k$  realiza el mismo procedimiento y envía el paquete al nodo  $h$ . Este nodo calcula sus vecinos y selecciona como próximo nodo al nodo  $k$ ; retornando el paquete a su emisor. En este caso, el ruteo voraz finaliza porque el nodo  $h$  es el nodo más próximo al destino.

## 2.2 RUTEO POR BRÚJULA

El ruteo por brújula (compass routing) [KSU99] se comporta como describimos a continuación. Suponga que el nodo  $o$  desea enviar un paquete al nodo  $d$ . El nodo  $o$  sólo conoce las coordenadas de sus vecinos y del destino. El ruteo por brújula envía el paquete al nodo  $w$  vecino de  $o$  que minimiza el ángulo  $\angle wod$ . Cuando exista más de un nodo que posean igualdad de ángulos uno de ellos se selecciona al azar. La figura 3 muestra como el ruteo por brújula selecciona el próximo nodo sobre una porción de un grafo geométrico plano.

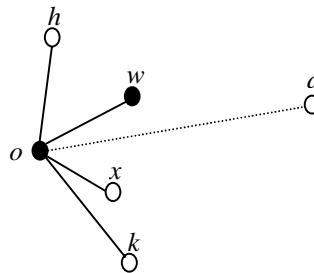


Figura 3: Selección del próximo nodo con el ruteo por brújula

El ruteo por brújula presenta los mismos inconvenientes que el ruteo voraz. Necesita de un grafo geométrico plano y no siempre encuentra el camino entre un nodo origen y otro destino. La figura 4 presenta una porción de un grafo geométrico plano (triangulación) que hace que el ruteo por brújula no encuentre el destino del paquete.

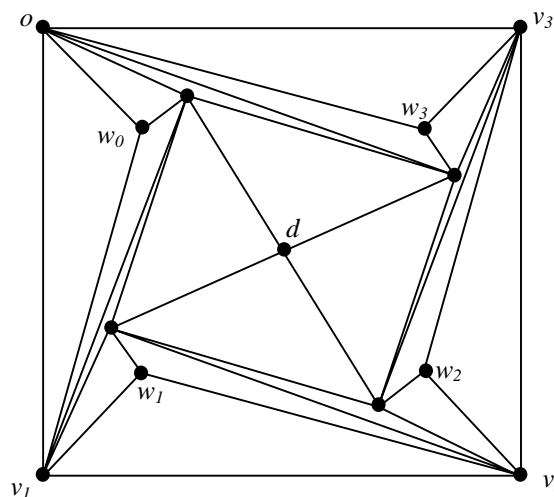


Figura 4: Triangulación que vence al ruteo por brújula

Asuma que el nodo origen es  $o$  y el destino  $d$ . La secuencia de nodos que conforman el camino encontrado por esta estrategia de ruteo es  $o, w_0, v_1, w_1, v_2, w_2, v_3, w_3, o, w_0, v_1, \dots$ . Por lo tanto el paquete queda atrapado en el ciclo  $o, w_0, v_1, w_1, v_2, w_2, v_3, w_3, o$  y nunca llegará a destino.

## 2.3 RUTEO VORAZ-BRÚJULA

El ruteo voraz-brújula (greedy-compass routing) combina las estrategias para la selección del próximo nodo utilizadas por el ruteo voraz y por el ruteo por brújula. Suponga que el nodo  $o$  desea enviar un paquete al nodo  $d$ . El nodo  $o$  sólo conoce las coordenadas de sus vecinos y del destino. Sean  $w$  y  $k$  dos vecinos del nodo  $o$  talque el nodo  $w$  minimiza el ángulo  $\angle wod$  en sentido contrario a las agujas del reloj y  $k$  el nodo que minimiza el ángulo  $\angle kod$  en el sentido de las agujas del reloj. Los nodos  $k$  y  $w$  se denominan los vecinos por brújula de  $o$ . Sea  $VPB$  el conjunto formado por los vecinos por brújula del nodo  $o$ . El ruteo voraz-brújula envía el paquete al nodo de  $VPB$ , que minimiza la distancia con el nodo destino  $d$ . En caso de igualdad se selecciona un miembro de  $VPB$  en forma aleatoria.

La figura 5 muestra como el ruteo voraz-brújula selecciona el próximo nodo sobre una porción de un grafo geométrico plano. El nodo  $w$  minimiza el ángulo  $\angle wod$  en sentido contrario a las agujas del reloj. El nodo  $k$  minimiza el ángulo  $\angle kod$  en sentido de las agujas del reloj. Por lo tanto  $VPB=\{w,k\}$ . El próximo nodo en el camino es  $w$  puesto que se encuentra más cerca del destino  $d$ .

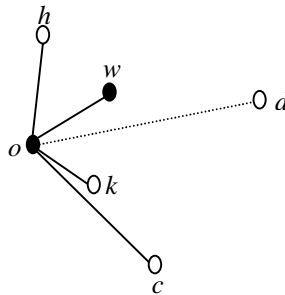


Figura 5: Selección del próximo nodo con el ruteo voraz-brújula

## 2.4 RUTEO POR BRÚJULA ALEATORIZADO

El ruteo por brújula aleatorizado (random-compass routing) [BM99][BM01] es otra variante del ruteo por brújula. Suponga que el nodo  $o$  desea enviar un paquete al nodo  $d$ . El nodo  $o$  sólo conoce las coordenadas de sus vecinos y del destino. Sea  $w$  el nodo que minimiza el ángulo  $\angle wod$  en sentido contrario a las agujas del reloj y  $k$  el nodo que minimiza el ángulo  $\angle kod$  en el sentido de las agujas del reloj. Los nodos  $k$  y  $w$  se denominan los vecinos por brújula de  $o$ . Sea  $VPB$  el conjunto formado por los vecinos por brújula de  $o$ . El ruteo por brújula aleatorizado selecciona como próximo nodo en el camino a un miembro de  $VPB$  con igual probabilidad. La figura 6 muestra una porción de un grafo geométrico y los nodos que el ruteo por brújula aleatorizado puede seleccionar.

Para finalizar esta sección es importante resaltar que el ruteo por brújula aleatorizado tiene los mismos inconvenientes que el algoritmo presentado en la sección anterior. Bose y Morin [BM01] demostraron que el ruteo por brújula aleatorizado encuentra siempre el camino entre un nodo origen y otro destino cuando el grafo subyacente es una subdivisión convexa. Dicha característica lo hace más robusto que las estrategias de ruteo al vuelo presentadas en las secciones anteriores.

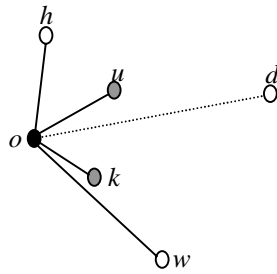


Figura 6: Selección del próximo nodo con el ruteo por brújula aleatorizado

### 3. CLASES DE GRAFOS

Las redes móviles pueden ser modeladas por un grafo  $G=(N,R)$  que cumple con las siguientes condiciones:

- cada nodo  $x \in N$  tiene como atributos un radio de transmisión  $Radio(x)$  y sus coordenadas en el plano;
- $R$  se construye de la siguiente manera: dos nodos  $x,y \in N$  son vecinos en la red si y solo si la distancia euclidiana entre ellos es menor que el mínimo de sus radios de transmisión.

Si todos los radios de transmisión son iguales esta clase de grafo se denomina *Grafo Unidad* y se designa por  $U(N)$ .

Formalmente:

$$N = \{ x / x \text{ es un nodo de la red } \}$$

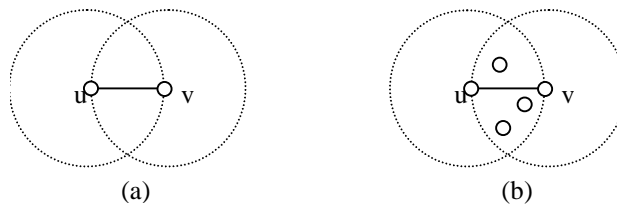
$$R = \{ (x,y) / \exists r \in \mathbf{R} \bullet r \text{ es un radio de transmisión } \wedge dist^1(x,y) \leq r \}$$

Si bien el modelo presentado por los grafos Unidad representan adecuadamente a las redes móviles, no es posible su utilización cuando el algoritmo de ruteo de paquetes usado supone que el grafo subyacente es plano. Sin embargo, se pueden extraer subgrafos planos del grafo Unidad realizando la intersección del mismo con otras clases de grafos como por ejemplo los grafos de Gabriel y los de Vecindad Relativa.

#### 3.1 GRAFO DE VECINDAD RELATIVA

El grafo de Vecindad Relativa de un conjunto de nodos  $N$ , designado por  $GVR(N)$ , es un grafo cuyos nodos son los nodos de  $N$  y tal que, para cada par de nodos  $x,y \in N$ , existe el arco  $(x,y)$  si ningún otro nodo de  $N$  se encuentra en la zona de intersección de los círculos  $C$  y  $C'$ , de radio  $dist(x,y)$ , con centro en los puntos  $x$  e  $y$  respectivamente.

La figura 7-a muestra un arco que cumple con las restricciones impuestas por  $GVR(N)$  mientras que la figura 7-b presenta el caso contrario.



<sup>1</sup>  $dist(x,y)$  hace referencia a la Distancia Euclidiana entre los nodos  $x$  e  $y$ .

Figura 7: Arcos válidos e inválidos en un Grafo de Vecindad Relativa

Karp y Kung en [KK00] demuestran que el grafo obtenido a través de la operación  $GVR(N)$  es muy útil para extraer un subgrafo conexo de  $U(N)$  realizando la operación  $GVR(N) \cap U(N)$ . Es destacable mencionar que el grafo resultante puede contener pocos arcos debido a las restricciones impuestas por la definición de  $GVR(N)$ . La figura 8 muestra un ejemplo de un grafo de Vecindad Relativa.

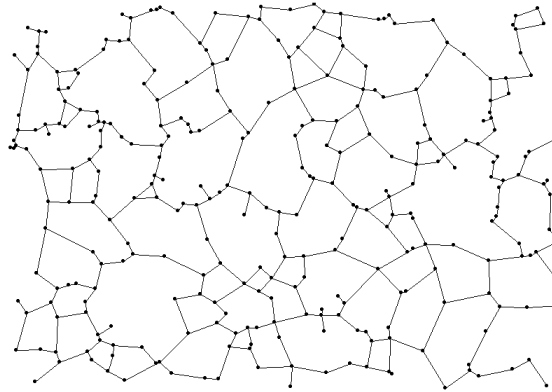


Figura 8: Ejemplo de grafo de Vecindad Relativa

### 3.2 GRAFO DE GABRIEL

El grafo de Gabriel de un conjunto de nodos  $N$ , designado por  $GG(N)$ , es un grafo cuyos nodos son los nodos de  $N$  y tal que, para cada par de nodos  $x, y \in N$ , existe el arco  $(x, y)$  si el círculo de diámetro  $d(x, y)$  no contiene más nodos de  $N$ . En la figura 9-a se muestra un arco que cumple con las restricciones impuestas por  $GG(N)$  y en la figura 9-b se presenta el caso contrario.



Figura 9: Arcos válidos e inválidos en un Grafo de Gabriel

El grafo de Gabriel es útil para extraer un subgrafo plano del grafo Unidad. Urrutia, J. y su grupo de investigación demostraron que el grafo obtenido realizando la operación  $GG(N) \cap U(N)$  es conexo y plano. Es importante notar que esta clase de grafos tiene más arcos que la descrita en la sección anterior porque  $GG(N)$  impone menos restricciones que  $GVR(N)$ . No obstante, es posible que el grafo resultante posea pocos arcos. La figura 10 muestra un ejemplo de un grafo de Gabriel.

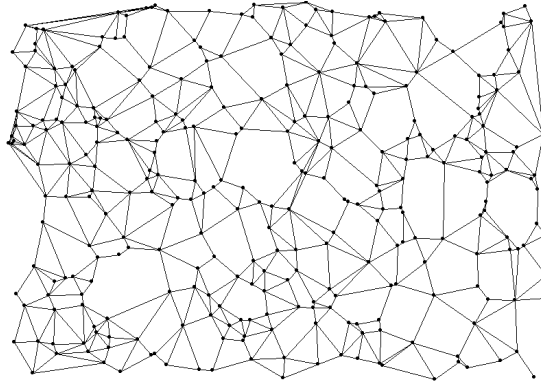


Figura 10: Ejemplo de grafo de Gabriel

#### 4. CRITERIOS DE EVALUACIÓN

Los criterios de evaluación permiten analizar el comportamiento de los algoritmos de ruteo. En lo siguiente se asume que  $CG$  es una clase de grafo que se corresponde con alguna de las clases de grafos presentadas;  $G=(N,R)$  es una instancia de  $CG$  y  $A$  es un algoritmo de ruteo. La función  $CAMINOS(A,G)$  referencia a todos los pares de nodos  $(o,d)$  en  $G$  para los cuales el algoritmo  $A$  envía con éxito un paquete desde  $o$  a  $d$ . La función  $NV(A,G)$  representa el número de veces que el algoritmo  $A$  fue usado para enviar un paquete entre un par de nodos de  $G$ . La función  $NV(G,v,A)$  indica el número de veces que el algoritmo  $A$  encuentra un camino entre un par de nodos de  $G$  que pasa por el nodo  $v$ . Las funciones  $CMC(G,u,v)$  y  $CMCE(G,u,v)$  referencian la longitud del camino más corto entre el nodo  $u$  y  $v$  sobre el grafo  $G$ ; el primero con respecto a la distancia euclidiana (longitud euclídea del camino) y el segundo teniendo en cuenta la distancia de enlace (número de enlaces del camino). Finalmente con  $C(A,G,u,v)$  y  $CE(A,G,u,v)$  denotamos la longitud de un camino entre los nodos  $u$  y  $v$  que el algoritmo  $A$  encuentra sobre el grafo  $G$  en cuanto a distancia euclidiana y distancia de enlace respectivamente.

El conocer la tasa de éxito de estas clases de algoritmos de ruteo es de suma importancia porque permite analizar la efectividad de la estrategia. Por esta razón utilizamos para evaluar los algoritmos de ruteo presentados en esta tesis la *Tasa de Éxito*, denotada por  $TE$ , cuya definición es la siguiente:

$$TE(A,g) = CAMINOS(A,g) / NV(A,g)$$

La eficiencia de un algoritmo de ruteo puede ser medida por la *Dilatación Euclidiana Promedio*, denotada por  $DEP$ , y la *Dilatación de Enlace Promedio*, referenciada como  $DEnP$ .

La definición  $DEP$  es la siguiente:

$$DEP = \frac{1}{CAMINOS(A,g)} \sum_{(o,d) \in CAMINOS(A,g)} \frac{CE(A,g,o,d)}{CMCE(A,g,o,d)}$$

y la definición de  $DEnP$  se presenta a continuación:

$$DEnP = \frac{1}{CAMINOS(A,g)} \sum_{(o,d) \in CAMINOS(A,g)} \frac{C(A,g,o,d)}{CMC(A,g,o,d)}$$

Estas cantidades son de importancia porque nos permitirán determinar la calidad de los caminos encontrados por las estrategias de ruteo.

## 5. RESULTADOS OBTENIDOS PARA GRAFOS DE VECINDAD RELATIVA

Los resultados experimentales realizados con la herramienta descrita en [BPG05], sobre grafos de hasta 200 nodos revelan que los algoritmos de ruteo que presentan una mayor tasa de éxito sobre los grafos de Vecindad Relativa son: ruteo voraz y ruteo voraz-brújula, (ver fig. 11). Mientras que para el caso del ruteo por brújula la tasa de éxito es muy próxima, pero inferior, a los algoritmos de ruteo voraz y voraz-brújula. El peor caso lo presenta el ruteo por brújula aleatorizado con tasas que oscilan entre el 11% y el 1%.

El comportamiento similar entre los algoritmos de ruteo voraz, voraz-brújula se debe, intuitivamente, a dos factores principales: i) ambos algoritmos poseen estrategias similares de selección del próximo nodo en el camino; ii) la estrategia de planarización elimina muchos arcos que pueden marcar las diferencias entre ambos algoritmos.

La leve disminución en la tasa de éxito del ruteo por brújula se puede deber a que, al intentar disminuir la distancia euclidiana [BM01], recorre más enlaces que el ruteo voraz y el voraz-brújula. Por consiguiente, el ruteo por brújula, tiene más chances de que el paquete no encuentre el destino. Esta característica puede estar potenciada por la gran disminución de arcos provocada por la estrategia de planarización.

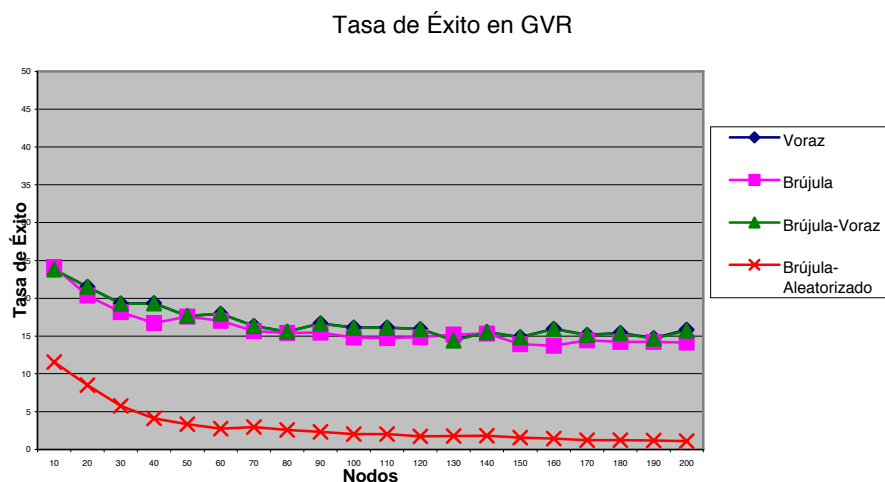


Figura 11: Tasa de éxito de los algoritmos de ruteo al vuelo aplicados sobre grafos de Vecindad Relativa

El pobre desempeño del ruteo por brújula aleatorizado se debe a la forma que tiene el algoritmo de seleccionar el próximo nodo en el camino. Es muy probable que el paquete retorne al nodo que lo emitió porque la estrategia de planarización del grafo Unidad elimina muchos arcos. Por consiguiente uno de los nodos, el que se encuentra en el sentido de las agujas del reloj o su contrario, será el que envió el mensaje.

Para finalizar con el análisis destacamos que no se puede establecer un ranking de algoritmos de ruteo al vuelo. Sólo afirmamos que el ruteo voraz y voraz-brújula son los algoritmos que alcanzan el destino mayor cantidad de veces. Por otra parte, el ruteo por brújula aleatorizado presenta el peor desempeño.

En la figura 12 mostramos los resultados de la dilatación de enlace promedio obtenida por los algoritmos de ruteo al vuelo aplicados sobre los grafos de Vecindad Relativa. Esta evaluación se llevó a cabo comparando el camino más corto con el camino encontrado por los algoritmos de ruteo al vuelo.

Los resultados indican que el ruteo por brújula aleatorizado presenta el peor desempeño. Dicho algoritmo encuentra caminos cuya longitudes exceden al camino más corto en valores que oscilan entre 25,9 % y 27,33%. El mecanismo de selección del próximo nodo del ruteo por brújula



aleatorizado es similar al del ruteo por brújula. La diferencia entre ambos radica en que el primero incorpora una componente aleatoria cuando selecciona el próximo nodo en el camino.

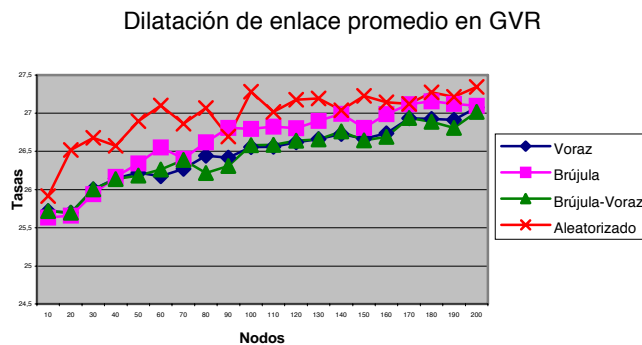


Figura 12: Dilatación de enlace promedio de los algoritmos de ruteo al vuelo aplicados sobre los grafos de Vecindad Relativa

La peculiaridad mencionada afecta la calidad del camino encontrado porque ambas estrategias intentan minimizar la distancia euclidiana [BM01] y no la distancia de enlace. Por consiguiente, es aparente que ambos algoritmos presenten los peores desempeños. Por otra parte, la componente aleatoria del ruteo por brújula aleatorizado disminuye la calidad del camino encontrado.

El ruteo voraz intenta minimizar el número de enlaces que debe recorrer el paquete para llegar al destino [BM01] y por consiguiente obtiene caminos de mejor calidad.

El ruteo voraz-brújula, al utilizar una estrategia parecida al ruteo voraz, también encuentra caminos que son más próximos al camino más corto. Sin embargo, es llamativo que su comportamiento sea, en la mayoría de las veces, mejor que el presentado por el ruteo voraz. Esta observación surge a partir del conocimiento de que dicha estrategia incorpora técnicas definidas para el ruteo por brújula que, como mencionamos previamente, minimizan la distancia euclidiana.

Para finalizar es importante señalar que a medida que el número de nodos crece las estrategias de ruteo al vuelo tienden a encontrar caminos más largos.

En la figura 13 mostramos los resultados de la evaluación de la dilatación euclidiana promedio. Dichos valores se obtuvieron a través de la aplicación de los algoritmos de ruteo al vuelo sobre los grafos de Vecindad Relativa.

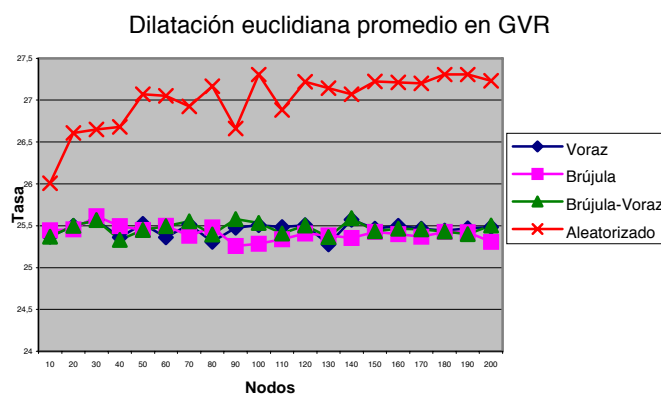


Figura 13: Dilatación euclidiana promedio de los algoritmos de ruteo al vuelo aplicados en los grafos de Vecindad Relativa

Los resultados muestran una semejanza en el comportamiento de los algoritmos ruteo voraz, ruteo por brújula y ruteo voraz-brújula. Aparentemente, a medida que la cantidad de nodos del

grafo, sobre el cual los algoritmos de ruteo se aplican, crece el ruteo por brújula obtiene caminos que están más próximos al camino más corto.

## 6. RESULTADOS OBTENIDOS PARA GRAFOS DE GABRIEL

En la figura 14 presentamos los resultados obtenidos con respecto a la tasa de éxito de los algoritmos de ruteo al vuelo aplicados sobre los grafos de Gabriel.

Como observamos la tasa de éxito es superior a la presentada por los grafos de Vecindad Relativa. La razón de este comportamiento se debe a que la estrategia de planarización del grafo Unidad admite una mayor cantidad de arcos. Por otra parte, la figura muestra que el desempeño del ruteo voraz, ruteo por brújula y el ruteo voraz-brújula es similar. Es notable que el comportamiento del ruteo por brújula se aproxime y en algunos casos supere al ruteo voraz y voraz-brújula. Esto se debe a que, si bien el ruteo por brújula tiene más chances de extraviar el paquete por que recorre más arcos, el incremento de arcos disminuye la probabilidad de fracaso del algoritmo.

En la figura 14 también se puede observar que el ruteo por brújula aleatorizado presenta el peor desempeño. Sin embargo, su comportamiento es superior al presentado cuando se lo aplica a los grafos de Vecindad Relativa.

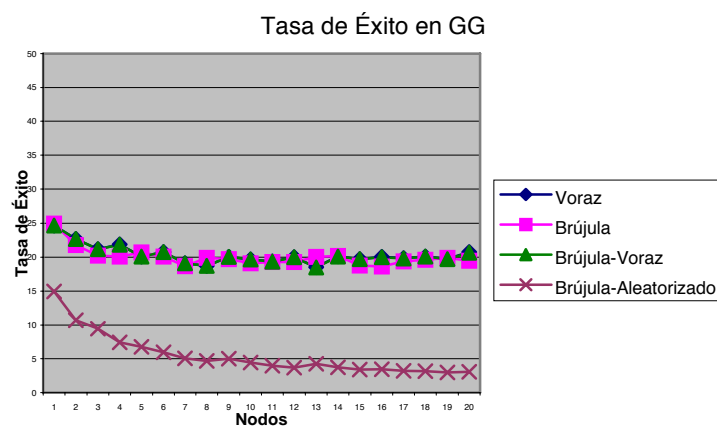


Figura 14: Tasa de éxito de los algoritmos de ruteo al vuelo aplicados a los grafos de Gabriel

En la figura 15 mostramos los resultados de la dilatación de enlace promedio obtenida a través de la aplicación de los algoritmos de ruteo al vuelo sobre los grafos de Gabriel.

Como observamos el ruteo voraz y voraz-brújula obtienen los mejores comportamientos. En apariencia la mayor cantidad de arcos, admitidas por la estrategia de planarización del grafo Unidad, permite que los algoritmos de ruteo encuentren caminos que se aproximan más al camino más corto.

Estos resultados eran esperables ya que el ruteo voraz y voraz-brújula tienden a disminuir la cantidad de enlaces recorridos por el paquete. Otra característica importante es el crecimiento en los valores obtenidos en los experimentos. A medida que la cantidad de nodos de los grafos de estudio crece las estrategias de ruteo encuentran caminos más largos.

A diferencia de los resultados obtenidos para los grafos de Vecindad Relativa, éstos permiten el establecimiento de un ranking de algoritmos; el cual presentamos a continuación: i) ruteo voraz-brújula; ii) ruteo voraz; iii) ruteo por brújula; iv) Ruteo por brújula aleatorizado.

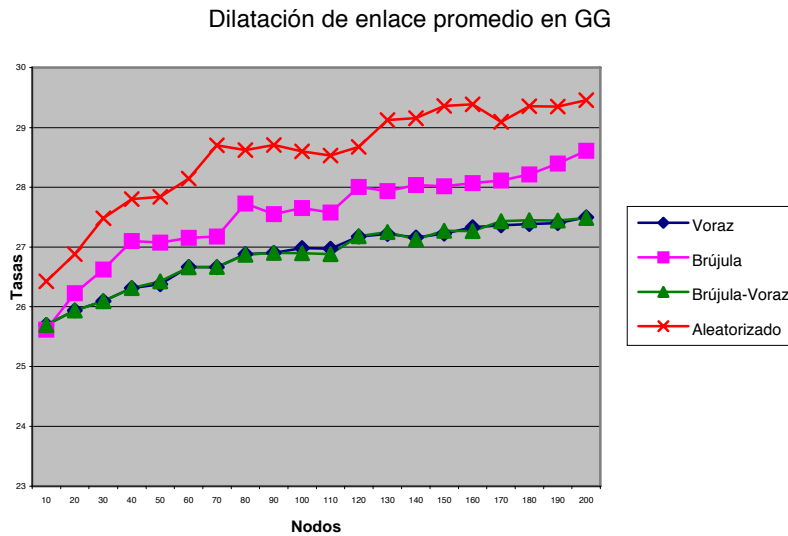


Figura 15: Dilatación de enlace promedio de los algoritmos de ruteo al vuelo aplicados sobre los grafos de Gabriel

En la figura 16 presentamos los resultados de la dilatación euclidiana promedio obtenidos al aplicar los algoritmos de ruteo al vuelo sobre los grafos de Gabriel.

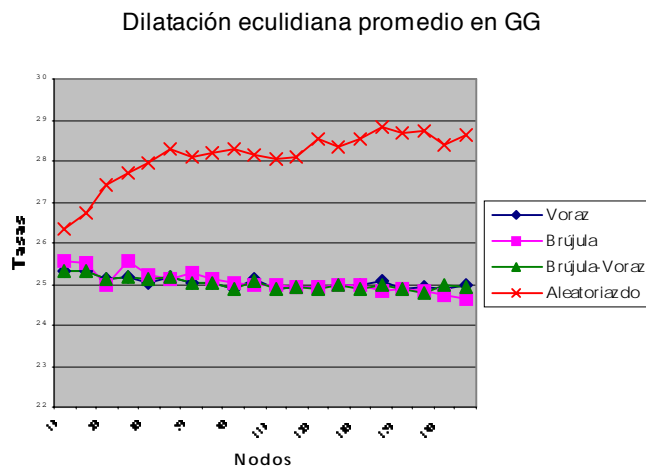


Figura 16: Dilatación euclidiana promedio de los algoritmos de ruteo al vuelo aplicados sobre los grafos de Gabriel

Como podemos observar los resultados presentan características similares a los obtenidos para el caso de los grafos de Vecindad Relativa. Los resultados muestran que las tasas obtenidas tienden a disminuir a medida que la cantidad de nodos de los grafos de estudio aumenta. Como podemos observar, a excepción del ruteo por brújula aleatorizado, las estrategias de ruteo presentan un comportamiento similar.

## 7. CONCLUSIONES

En primer lugar debemos destacar la imposibilidad de aplicación de las estrategias de ruteo al vuelo en redes móviles reales. Esto se debe a que las tasas de éxito obtenidas son bajas para las dos clases de grafos estudiadas. Sin embargo, estimamos que su combinación con estrategias de ruteo al vuelo con entrega garantizada son una alternativa importante para sustituir a la inundación

(estrategia que actualmente se usa para transmitir paquetes en redes móviles). La razón de esta suposición se debe a que dichos algoritmos disminuirían la cantidad de paquetes dispersos en la red, y por lo tanto se evitaría el congestionamiento de la misma.

Los resultados de la dilatación euclidiana promedio muestran que los algoritmos de ruteo estudiados tienen un comportamiento similar. No obstante, los mejores resultados son obtenidos cuando estos algoritmos se aplican a los grafos de Vecindad Relativa.

En cuanto a la dilatación de enlace los resultados son muy irregulares y no se puede establecer claramente que algoritmo de ruteo tiene el mejor desempeño. Los resultados indican que el ruteo por brújula obtiene valores más bajos a medida que la cantidad de nodos en el grafo, sobre el cual se aplica, crece.

De los criterios usados para el estudio de los algoritmos de ruteo creemos que, en principio, la tasa de éxito es el más relevante. Esto se debe a que, si bien es deseable que los algoritmos recorran menos enlaces y distancias, se necesita una buena efectividad en la transmisión de paquetes. Una vez logrado este objetivo los estudios deberían centrarse en disminuir los valores de la distancia de enlace y distancia euclidiana.

Con respecto a las estrategias de ruteo podemos decir que la simplicidad del ruteo voraz es muy atractiva porque implica menos costo de implementación en los dispositivos utilizados en redes reales.

Para finalizar destacamos que en [BPG05] se presentan más estudios relacionados con esta temática. Dichas investigaciones versan sobre el comportamiento de los algoritmos de ruteo descritos en este artículo aplicados sobre los grafos de Morelia. Lamentablemente, la extensión de esta investigación imposibilitó su incorporación.

## 8. REFERENCIAS

- [BBC+1] Bose, P.; Brodnik, A.; Carlsson, S.; Demaine, E.; Fleischer, R.; López-Ortiz, A.; Morin, P.; Munro, J. *Online Routing in Convex Subdivision*. Natural Science and Engineering Research Council of Canada. 2001.
- [BGH03] Berón, M.; Gagliardi, O.; Hernández, G. *Evaluación de Algoritmos de Ruteo en Redes de Computadoras*. Congreso Argentino de Ciencias de la Computación. 2003
- [BM01] Morin, P. *Online Routing in Geometric Graph*. Tesis Doctoral. Canada. 2001.
- [BM99] Bose, P.; Morin, P. *Online Routing in Triangulations*. In Proceedings of the Tenth International Symposium on Algorithms and Computation (ISAAC'99), volume 1741 of Springer LNCS, pages 113-122, 1999.
- [BPG05] Berón, M.; Peñalver, G.; Gagliardi, O. *Un Evaluador de Algoritmos de Ruteo*. Tesis de maestría. Universidad Nacional de San Luis. Argentina. 2005.
- [CDK+04] Chavez, E.; Dobrev, S.; Kranakis, E.; Opatrny, J.; Stacho, L.; Urrutia, J. *Route Discovery Width constant Memory in Oriented Planar Geometric Networks*. 2004.
- [DSW01] Datta, S.; Stojmenovic, I.; Wu, J. *Internal node and shortcut base routing with guaranteed delivery in wireless networks*. In Proc. IEEE Int. Conf. on Distributed Computing and Systems Workshops; Cluster Computing, to appear, pages 461-466, 2001. USA. 2001.
- [GTB02] Gagliardi, O.; Taranilla, M.; Beron, M. *La Geometría Computacional a nuestro alrededor*. III Workshop de Informática y Ciencias de la Computación. Argentina. 2002.
- [KK00] Karp, B.; Kung, T. *GPSR: Greedy Perimeter Stateless Routing for Wireless Networks*. Proc of 6<sup>th</sup> Annual ACM/IEEE International Conference on Mobile Computing and Networking. 2000.
- [KSU99] Kranakis E., Singh H., J. Urrutia. *Compass routing on Geometric network*. Proc. 11<sup>th</sup> Canadian Conference on Computational Geometry, Vancouver. 1999.
- [PGB03] Peñalver, G. Gagliardi, O.; Beron, M. *Evaluación de Algoritmos de Ruteo en Redes de Computadoras*. IV Workshop de Informática y Ciencias de la Computación. Argentina. 2003.