



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Posgrado en Ciencia e Ingeniería
de la Computación

ALGORITMOS LOCALES PARA COLOREAR ARISTAS
EN REDES INALÁMBRICAS

T E S I S

QUE PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS (COMPUTACIÓN)

P R E S E N T A :

CREVEL BAUTISTA SANTIAGO

Tutor de tesis:
DR. JORGE URRUTIA GALICIA

México, D.F. 2006



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos



A mi mamá, por el inagotable impulso que me ha proporcionado durante toda mi vida.

A mis hermanas Naye y Care, que han sabido confortarme en los momentos más difíciles.

A mi abuelita Eva, que con sus pláticas me hace sentir como un nieto muy amado y apreciado, más de lo que creo merecer.

A mi papá, cuya conciencia me da cada vez más una mayor satisfacción.

A mi tío Abel, mi tía Lupe, mis primos Leydi y Alberto que siguen siendo un soporte importante para mi y mi familia.

A mi abuelito Amancio, mi tíos Nacho y Rosario, mi tío Uri, mis primos Néstor y Evita por su solidez inquebrantable como núcleo familiar.

A mis super amigos Tita y Gil, que siempre me siguieron animando durante estos dos últimos dos años.

A Jorge Urrutia, mi tutor, que me ayudó enormemente a lograr este trabajo.

A Mónica y José, por su apoyo y resguardo; realmente me han servido sus comentarios.

A Elisa, quien me dió las bases de la línea que ahora sigo.

A mis amigos de la maestría, Wendy, Montse, Paty, Pollito, Dolores, Paola y Ana, con quienes enfrenté este reto académico.

A Lulú, Diana y Amalia, quienes me ayudaron a resolver algunos problemas con la mayor voluntad posible.

Índice general

Figuras	III
Introducción	1
1. Redes <i>ad hoc</i> móviles	5
1.1. Motivación	5
1.2. Antecedentes	5
1.3. Redes Celulares	6
1.4. Esquemas de comunicación entre nodos	8
1.5. Conceptualización de una red ad hoc móvil	8
1.6. Características	10
1.7. Funciones típicas en una red ad hoc	11
1.7.1. Establecimiento de rutas	11
1.7.2. Seguridad	12
1.8. Campos de aplicación	13
2. La técnica de localidad	15
2.1. Motivación	15
2.2. Alcances	15
2.2.1. Agrupamiento en una red ad hoc	16
2.3. El primer algoritmo local para establecer rutas	17
2.4. Garantía en la entrega de paquetes	21
2.5. Consideraciones en el proceso de establecimiento de rutas	23
3. Resultados Previos	27
3.1. Motivación	27
3.2. Teorema de Vizing	27
3.3. Árbol Generador de Peso Mínimo Local	31
3.3.1. El Algoritmo	33
3.3.2. Propiedades de la gráfica G_k^{\triangleleft}	34

4. Coloración de aristas localmente	39
4.1. Motivación	39
4.2. Un algoritmo local para la asignación de frecuencias	39
4.2.1. Mapeo del problema de asignación de frecuencias a la coloración de aristas	40
4.2.2. Coloración de las aristas de una gráfica de manera local	40
Comentarios Finales	47
Referencias	51

Figuras

1.1.	Red celular que hace uso de la reutilización de frecuencias.	7
1.2.	Posibles esquemas de transmisión de los nodos en una red ad hoc.	9
1.3.	Ejemplo de posibles nodos en una red ad hoc.	9
2.1.	Uso del ruteo por brújula para establecer una ruta de s a t	18
2.2.	Problema al aplicar el ruteo por brújula para alcanzar t desde s	19
2.3.	Triángulo que pertenece a la triangulación de Delaunay.	20
2.4.	Establecimiento de rutas en la triangulación de Delaunay.	20
2.5.	Uso del ruteo por caras en una gráfica geométrica.	23
2.6.	Selección de las aristas de Gabriel.	24
2.7.	Selección de las aristas para la RNG.	25
3.1.	Índice cromático de K_4 y C_5	28
3.2.	Trayectoria de u a v con aristas más pequeñas que e	34
3.3.	Suposición de que dos aristas se cruzan en G_k^{\triangleleft}	35
3.4.	División del círculo unitario con centro en u en regiones.	36
3.5.	Un vértice u en G_k^{\triangleleft} tiene grado menor a seis.	37
3.6.	El grado de un vértice u en G_k^{\triangleleft} puede ser cinco.	38
4.1.	La gráfica, \mathfrak{F} , que es el resultado de aplicar LocalMST ₂ sobre G	41
4.2.	División de la región en que se encuentra \mathfrak{F}	42
4.3.	Caso en el que la distancia entre dos vértices en \mathfrak{F}_{ij} es cuatro.	43
4.4.	Coloración de las aristas de las gráficas \mathfrak{F}_{ij} localmente.	44
4.5.	Coloración de una arista que cruzan las rectas $y = 2j + 2$ o $y = 2j$	45
4.6.	Coloración de una arista que cruzan las rectas $x = 2i$ o $x = 2i + 2$	45
4.7.	Coloración propia de las aristas de la gráfica \mathfrak{F} localmente.	46

Introducción



Durante los últimos años, el desarrollo de la tecnología ha permitido que más y más dispositivos electrónicos cuenten con la posibilidad de comunicarse con otros sin la necesidad de una conexión física, es decir, de forma inalámbrica. Esto trajo consigo el nacimiento de una nueva rama de la computación, el de las redes *ad hoc*¹ inalámbricas.

Más formalmente, una red ad hoc inalámbrica consiste de un conjunto de nodos inalámbricos los cuales, en un principio, cuentan con la capacidad de transmitir omnidireccionalmente de tal manera que una sólo transmisión de un nodo puede ser recibida por todos los nodos dentro de su vecindad. El movimiento de tales nodos, incluyendo su aparición o salida de la red, ocasiona que la topología de la red cambie constantemente, lo cual hace que el control de dicha topología y el ruteo eficaz no sea una tarea fácil.

Además, a diferencia de las redes inalámbricas tradicionales, este tipo de redes no cuenta con una infraestructura fija o un soporte centralizado para administrarla por lo que es también parte del problema lidiar con estas limitaciones.

Una red ad hoc inalámbrica puede ser vista como una gráfica en cierto momento de tal forma que su conjunto de vértices es aquel que compone el conjunto de nodos de la red y donde habrá una arista de un vértice a otro si es que este se encuentra dentro del radio de transmisión del primero. Si el radio de transmisión de los nodos es igual a 1, entonces la gráfica que representa a dicha red se conoce como una gráfica de disco unitario (*unit disk graph*, *UDG*) y son precisamente estas con las que trataremos en este trabajo.

En este nuevo campo se estudian aspectos similares a los de las redes tradicionales (las que cuentan con una conexión física), como lo es el desarrollo de nuevos algoritmos de ruteo para transmitir cierta información que necesite ser compartida, pero desde un enfoque distinto ya que aunque tales dispositivos se siguen viendo como nodos de una red, la forma de comunicación no puede ser la misma.

¹Este término ha sido utilizado para expresar la capacidad que deben tener este tipo de redes para amoldarse a los cambios que generen los nodos que las componen al moverse de manera dinámica.

Supongamos ahora que aunque los vecinos que se encuentran dentro del radio de transmisión de un cierto nodo comparten el mismo espectro de comunicación, el nodo en cuestión debe ocupar distintas frecuencias de dicho espacio de tal manera que no haya interferencia durante alguna de sus transmisiones, es decir, que la transmisión de un mensaje en determinada frecuencia llegue a un sólo destino.

Podemos modelar el problema anterior al nivel de la gráfica que represente a nuestra red de tal manera que asignemos a cada una de las aristas un color distinto (una frecuencia distinta), entonces nuestro problema se traduce en colorear las aristas de dicha gráfica para así obtener una coloración propia, es decir, que a dos aristas que sean adyacentes (que incidan en un mismo vértice) no les sea asignado el mismo color.

Así también, sería deseable que el número de colores a utilizar fuera el menor posible ya que no sería conveniente asignar más que las frecuencias necesarias sin desperdiciar otras que bien podrían ser utilizadas posteriormente.

Dentro de la rama de la teoría de gráficas, se cuenta con el concepto de *índice cromático*, el cual se define como el mínimo número de colores con el cual una gráfica puede ser coloreable por aristas (claro, respetando que la coloración sea propia). En 1964, Vizing demostró que el índice cromático de una gráfica no vacía G era el grado máximo de esta ($\Delta(G)$) ó $\Delta(G) + 1$.

No obstante no existe un algoritmo eficiente o fórmula que determine con exactitud tal número dada una gráfica y más aún, se ha demostrado que el problema de determinar si el índice cromático de una gráfica es igual a su grado máximo se encuentra dentro de la categoría de los problemas NP -completos; sin embargo si existe un algoritmo polinomial que determina una $\Delta + 1$ coloración.

Dado lo anterior, nuestro objetivo en este trabajo será tratar este problema de manera local con la intención de obtener una coloración propia por aristas de la gráfica que represente a nuestra red ad hoc inalámbrica y así poder dar una asignación de frecuencias adecuada. Además, deseamos utilizar pocos colores, es decir, que la solución obtenida sea c veces más grande que la solución óptima, donde c es una constante.

Para finalizar esta sección, considerando todos los argumentos de los párrafos anteriores, sólo describiremos los capítulos en los que se decidió dividir este trabajo escrito:

Capítulo 1. Dado que el objeto sobre el cual se plantea el problema es una red ad hoc, en este capítulo describiremos de manera más amplia este concepto, señalando algunas de sus características más importantes, así como algunas de sus aplicaciones y problemas que en ellas se suscitan.

Capítulo 2. Como ya lo mencionamos, nuestra idea es aplicar un paradigma local al algoritmo planteado para resolver el problema en cuestión, por lo que en esta sección

del trabajo detallaremos las ideas que hay detrás de este enfoque, así como de sus orígenes y problemas ya resueltos haciendo uso de él.

Capítulo 3. Aquí haremos un análisis de algunos resultados ya existentes que jugarán un papel importante en el planteamiento del resultado principal de este trabajo, entre ellos el teorema de Vizing. Nuestro objetivo al definir este capítulo es dar a entender el trasfondo de dichos resultados, de tal forma que su incorporación sea de manera sencilla.

Capítulo 4. En este último apartado, simplemente plantearemos a detalle el resultado logrado tomando en consideración las secciones anteriores.

1.1. Motivación

Hoy en día una gran cantidad de personas cuenta con uno o varios dispositivos portátiles como teléfonos celulares, asistentes personales digitales o *PDA's* (personal digital assistants) y laptops. Así mismo, gracias al desarrollo de la tecnología, esta clase de dispositivos pueden interactuar de tal manera que actividades como lo es el compartir documentos entre ellos y la lectura de correo electrónico, entre otras, sea ya una realidad.

El llevar a cabo dichas actividades se logra debido a que tales dispositivos, en cierto momento, se vuelven parte de una red *ad hoc* móvil o *MANET* (mobile ad hoc network), la cual permite establecer una comunicación entre los elementos incorporados en cualquier momento dentro de un área específica de manera inalámbrica.

En años recientes, la investigación hecha en el campo computacional de las *MANETs* ha tenido gran auge, así también, sus aplicaciones han gozado de un aumento de popularidad al hacer un mayor uso de esta tecnología en la vida cotidiana. Dado que es precisamente una red *ad hoc* el tipo de esquema del que partiremos para desarrollar este trabajo, en este capítulo estableceremos un panorama general que nos permita conocer algunos de sus antecedentes así como de algunas de características importantes acerca de este concepto.

1.2. Antecedentes

Aunque el acrónimo de *MANET* es relativamente nuevo, la idea detrás de él no lo es. Por ejemplo, al principio de los años 70, no mucho después del inicio del desarrollo de la tecnología que se convertiría en lo que hoy conocemos como Internet, el Departamento de Defensa de los Estados Unidos financió investigación concerniente al uso de dicha tecnología, en donde no se tuviera la restricción de una infraestructura fija o alámbrica. Por supuesto, una primera motivación para este desarrollo fue la necesidad militar para la supervivencia en el campo de batalla, ya que las condiciones que ahí se daban hacían deseable que los combatientes fueran capaces de moverse de una manera un poco más libre sin la necesidad de las restricciones impuestas por dispositivos de comunicación alámbricos.

Se puede decir que las redes ad hoc tienen su origen en el año de 1968 ([13]), durante el desarrollo del proyecto de la red *ALOHA*, cuyo objetivo era conectar instalaciones educativas en Hawaii. Aunque esta red hacía uso de estaciones fijas, el protocolo ALOHA proveía un acceso de canales distribuido y por tanto una base para el desarrollo de esquemas del mismo tipo que fueran apropiados para las redes ad hoc.

Con base en la red ALOHA, se desarrolló el proyecto DARPA, en 1973, el cual proveía mecanismos de administración de operación centralizados también sobre una base distribuida. Sin embargo, los desarrolladores se dieron cuenta de que algunas de las técnicas utilizadas incrementaban la capacidad de la red, ya que el espacio que esta abarcaba podía ser reutilizado.

Aunque muchos otros proyectos experimentales de redes fueron desarrollados posteriormente, estos sistemas inalámbricos no tuvieron realmente un gran impacto entre los consumidores. Fue hasta la puesta en marcha del proyecto IEEE 802.11, que es un estándar para redes de área local inalámbricas, que el Instituto de Ingeniería Eléctrica y Electrónica (IEEE) decidió adoptar el término de red ad hoc. Con esto, el IEEE esperaba dar la idea del despliegue de un escenario de desarrollo completamente nuevo.

1.3. Redes Celulares

Antes de comenzar a describir más a profundidad nuestro concepto de red ad hoc, es conveniente establecer algunas observaciones generales acerca de la tecnología de redes inalámbricas, en particular de las redes celulares, las cuales son sistemas que actualmente son muy usados y que preceden a los de las MANETs.

Estos sistemas se forman por un conjunto de celdas, cada una con su propio transmisor o estación base, las cuales son usadas para cubrir diferentes áreas y proveer una cobertura sobre una región más grande que la de una sola celda. También, estas redes hacen uso de tranceptores, los cuales son dispositivos que llevan a cabo funciones de transmisión y recepción, ya sea para ser utilizados por las estaciones o por los dispositivos que conforman la red (tranceptor móvil).

Una característica importante en este tipo de redes es el aumento en su capacidad dado que se puede reutilizar una misma frecuencia de transmisión en distintas celdas para transmisiones completamente distintas. Desafortunadamente, es inevitable un cierto nivel de interferencia en la señal producida por dos celdas que utilicen la misma frecuencia, por lo que el que dos celdas adyacentes utilicen la misma no es una configuración posible. La figura 1.1 muestra cómo una red cuyas estaciones base tienen un área de cobertura hexagonal, puede hacer uso de cuatro frecuencias para operar.

Los dispositivos o nodos que conforman este tipo de redes pueden ser móviles, lo cual significa que son capaces de poder cambiar la estación que los controla en la red en cierto

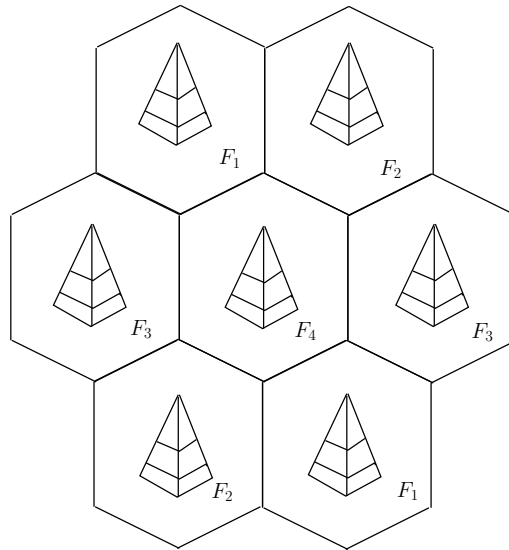


Figura 1.1: Red celular que hace uso de la reutilización de frecuencias.

momento. Así por ejemplo, en una red de teléfonos celulares, un nodo o teléfono que se encuentre dentro del área de control de una estación y que se quiera comunicar con algún otro teléfono, primeramente tiene que establecer una comunicación con su estación de control y posteriormente esta se encargará de establecer la conexión con el otro nodo, si es que este se encuentra dentro de su área de control, o con otra estación, si es que el dispositivo destino se encuentra en otra región.

En su gran mayoría, las redes celulares utilizan el protocolo de internet móvil (*mobile IP*) para su operación. En este paradigma, la identidad de un usuario podría o no ser equivalente a su posición, dependiendo de si el usuario adopta un identificador de *posición dependiente* (temporal) o uno de *posición independiente* (permanente), respectivamente. Algunas veces, los usuarios con identificadores temporales son conocidos como “nómadas”, mientras que a los permanentes se les conoce como usuarios “móviles”. La diferencia entre estos términos es que los de tipo nómada podrían moverse, pero principalmente cumplen sus funciones dentro de la red en una posición fija; por otro lado los de tipo móvil deben trabajar sobre la marcha, cambiando su estación de control conforme sea necesario. En cualquiera de los casos, un soporte adicional podría ser necesario para encontrar la posición de un usuario dentro de la red, por lo que es indispensable hacer uso de técnicas de establecimiento de rutas teniendo como base las estaciones de control.

El punto a resaltar en este tipo de redes es que mientras los dispositivos de los usuarios podrían o no moverse, la infraestructura de la red permanece fija. Entonces, aunque los usuarios sean de tipo móvil, mucha de la tecnología que aporta dicha infraestructura puede ser utilizada para dar un soporte a dichos usuarios.

1.4. Esquemas de comunicación entre nodos

Al tratar con medios de transmisión inalámbricos no confiables se deben tener en cuenta consideraciones especiales en el sistema de comunicación entre los nodos de una red ad hoc, esto con el objetivo de asegurar una operación confiable y eficiente. Una manera de hacerlo es emplear un esquema de comunicación indirecta o multipunto (*multihop*), en la cual cada nodo es también un ruteador, lo cual facilita el reuso de recursos tanto en espacio como en tiempo, siempre y cuando los nodos que participen en la red estén razonablemente bien distribuidos en el área en cuestión ([13]). En contraste, las redes en las que sus nodos se comunican de manera directa (*singlehop*) principalmente comparten los recursos en el dominio temporal.

Básicamente, la forma de operar de estas dos técnicas es la siguiente:

Directa. En este caso, la red envía datos de manera directa desde el nodo origen al nodo destino, es decir, ambos nodos se encuentran mutuamente dentro de su radio de alcance de transmisión.

Indirecta. En este escenario, los paquetes de información son transmitidos del nodo origen al nodo destino a través de una ruta, la cual posiblemente considera algún otro nodo de la red. Principalmente esta técnica se utiliza cuando el nodo destino no está dentro del radio de transmisión del nodo origen.

Así por ejemplo, la figura 1.2 muestra ambos esquemas de transmisión considerando una misma distribución de los nodos, así como también los mismos nodos origen y destino.

Dada la descripción de ambos esquemas se puede decir que una razón evidente por la cual se usa el esquema multipunto es para obtener conectividad, ya que algunos nodos bien podrían estar fuera del rango de transmisión de otros y por tanto, un esquema como el directo no funcionaría.

1.5. Conceptualización de una red ad hoc móvil

Como ya lo mencionamos, el campo de las redes ad hoc móviles o MANETs ha recibido una gran atención durante los últimos años, esto debido principalmente al desarrollo de la tecnología y a las necesidades que no se habían podido satisfacer de una manera deseable, aunque claro, esto también ha traído consigo que los resultados adquiridos se apliquen a nuevos planteamientos.

Una red ad hoc móvil se define como una colección de plataformas móviles o nodos, donde cada uno de estos es libre de moverse arbitrariamente ([30]). Cada nodo consiste lógicamente de un ruteador que podría tener múltiples hosts y estos a su vez tener múltiples dispositivos de comunicación inalámbrica. Así también, en principio, cuenta con una antena omnidireccional que le permite comunicarse con otros nodos que estén

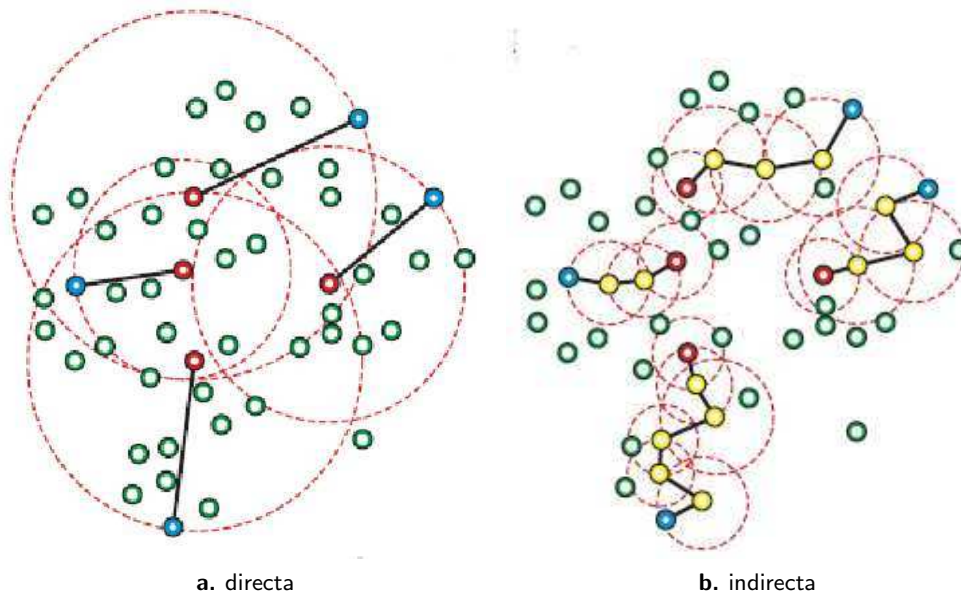


Figura 1.2: Posibles esquemas de transmisión de los nodos en una red ad hoc. Los nodos origen los identificamos de color rojo y los destino de color azul. Así mismo, en (a), los nodos de color amarillo representan aquellos nodos intermedios de una ruta entre un nodo rojo y uno azul.

dentro de su alcance de transmisión. Una representación de esto se muestra en la figura 1.3.

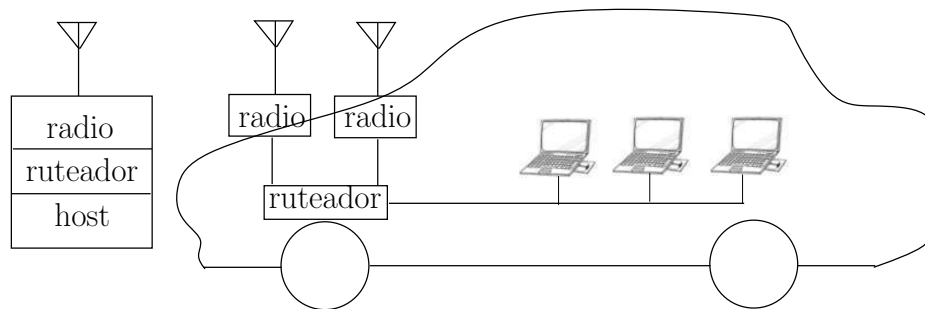


Figura 1.3: Ejemplo de posibles nodos en una red ad hoc.

El término MANET describe redes distribuidas, móviles, inalámbricas y con comunicación indirecta (multihop) que operan sin el beneficio de cualquier infraestructura existente, excepto por los nodos mismos. Una red ad hoc expande la visión que se tiene de Internet, en la cual nodos inalámbricos en los límites de la red están típicamente conectados y soportados de manera directa hacia la infraestructura fija.

Desde el punto de vista de la teoría de gráficas, una red ad hoc móvil es una gráfica

$G = (N, E(t))$, cuyo conjunto de vértices, N , está formado por cada uno de los nodos móviles de la red y donde habrá una arista, que pertenecerá a E , entre un par de nodos si es que estos pueden comunicarse de manera directa. El conjunto de aristas, $E(t)$, así definido, está dado en función del tiempo y cambia conforme los nodos cambian de posición al moverse dentro de la red.

Nota: De ahora en adelante ocuparemos los términos de red ad hoc móvil, red ad hoc o MANET indistintamente.

1.6. Características

En contraste con las redes alámbricas tradicionales, podría esperarse que una red ad hoc opere en un ambiente de red en el que algunos o todos los nodos sean móviles. Dentro de este ambiente dinámico, las funciones de red deben hacerse de una manera distribuida ya que los nodos podrían desaparecer o aparecer repentinamente en la red. Sin embargo, en general, los mismos requerimientos básicos para la conectividad y el control de tráfico que aplican para las redes tradicionales aplican también para las redes ad hoc.

A continuación estableceremos algunas características sobresalientes acerca de las redes ad hoc:

- *Operación distribuida.* Un nodo en una red ad hoc no puede contar con un soporte de seguridad o establecimiento de rutas como parte intrínseca de la red. En su lugar, estas funcionalidades deben diseñarse de tal manera que puedan llevarse a cabo de manera eficiente bajo condiciones distribuidas.
- *Topología dinámica.* En general, los nodos son libres de moverse arbitrariamente. La topología de la red, la cual típicamente es de tipo multipunto, podría cambiar rápida y aleatoriamente en tiempos impredecibles. No obstante, a pesar de esta situación, la conectividad en la red debería de mantenerse para permitir a las aplicaciones y a los servicios operar sin ningún problema. En particular, esto influenciará el diseño de los protocolos que permitan establecer rutas de comunicación.
- *Restricción de ancho de banda, capacidad variable, conexión asimétrica.* Las conexiones inalámbricas continuarán teniendo una capacidad significativamente menor que sus contrapartes alámbricas. Un efecto de esta moderación en las capacidades de conexión es que el aumento de demandas por aplicaciones podría exceder la capacidad de la red. Otra consecuencia es que las redes ad hoc tendrán que operar en ambientes heterogéneos con variaciones de retraso debido al ancho de banda disponible.
- *Operación con energía restringida.* Algunos o todos los nodos en una red ad hoc podrían depender de pilas para obtener la energía que les permita operar. Para tales nodos, la conservación de este recurso se vuelve un criterio de suma importancia al momento del diseño, ya que podría afectar, por ejemplo, el procesamiento en el

CPU, el uso y/o tamaño de la memoria y el procesamiento de señales. Las funciones referentes a la comunicación asumen directamente la responsabilidad de las aplicaciones y los servicios en el dispositivo. Entonces, los algoritmos y mecanismos que implementan funcionalidades de la red deben optimizarse tomando en cuenta un menor consumo de energía, esto con la finalidad de ahorrar capacidades para las aplicaciones mientras se provea un buen desempeño en la comunicación. Además de obtenerse una conectividad razonable en la red, la inclusión del paradigma de comunicación indirecta permite mejorar el desempeño desde el punto de vista del ahorro de energía. Hoy en día, sin embargo, esto sólo puede realizarse a un costo mayor en la complejidad del establecimiento de rutas.

- *Vulnerabilidades inalámbricas y seguridad física limitada.* Las redes inalámbricas móviles son generalmente más susceptibles a sufrir ataques de seguridad físicos y de información que las redes fijas y alámbricas. Una capa de técnicas de seguridad existentes puede aplicarse para reducir dichos ataques.
- *Cambio frecuente en la capacidad de la conexión.* Los efectos de altas tasas de error de bits podrían ser más profundos en una red ad hoc multipunto, ya que lo que afecta a una ruta multipunto es el número total de errores en la comunicación. Además, más de una ruta podría usar un punto de comunicación en particular, por lo que el no tenerlo disponible en cierto momento afectaría varias sesiones durante periodos de transmisión con altas tasas de error de bits. Esto trae consigo que las funciones de ruteo se vean afectadas, sin embargo, se pueden aplicar funciones eficientes que protejan la capa de comunicación para mejorar la calidad en la conexión ([13]).

1.7. Funciones típicas en una red ad hoc

Así como en las redes alámbricas, dentro del campo de las redes ad hoc existen funcionalidades que deben ser cubiertas. Dos de las más importantes a considerar son las que tienen que ver con el establecimiento de una ruta de comunicación entre dos nodos y la seguridad.

1.7.1. Establecimiento de rutas

Para las redes ad hoc móviles, el problema de establecer rutas para la transmisión de paquetes entre cualquier par de nodos llega a ser una tarea complicada debido a que tales nodos pueden moverse aleatoriamente dentro de la red. Una trayectoria que pudo considerarse como óptima en un cierto tiempo podría no servir poco tiempo después. Es más, las propiedades estocásticas de los canales de comunicación inalámbricos también forman parte de los puntos a considerar en la calidad de la trayectoria. Con este objetivo, una red puede hacer uso de protocolos que se encarguen de determinar una ruta de comunicación posible entre los nodos en cuestión.

Los protocolos que, dado un nodo en particular, mantienen una ruta hacia todos los posibles destinos en la red tienen la ventaja de que la comunicación con destinos arbitrarios

experimenta un ligero retraso inicial desde el punto de vista de la aplicación. Cuando una aplicación comienza, una ruta puede ser inmediatamente seleccionada de una tabla que contenga dichas trayectorias de comunicación. Tales protocolos se denominan *proactivos*, debido a que guardan información acerca de las rutas incluso antes de que alguna de ellas sea necesitada. De igual manera se les conoce como protocolos basados en tablas ([32]) ya que se puede pensar que las rutas están disponibles como parte de una tabla confiable.

Sin embargo, los protocolos proactivos sufren la desventaja de que adicionalmente se requiere un control de tráfico, esto con el objetivo de mantener actualizadas las entradas de la tabla que contiene las rutas, ya que la red puede cambiar su configuración debido a un evento imprevisto en alguno de sus nodos, en cuyo caso, las rutas que se encuentren en ese momento en la tabla podrían no ser reales. Si una ruta se rompe, es decir, si alguno de los nodos que se consideraba dentro de ella ya no está disponible, entonces esta debe ser reparada, lo cual se considera como un gasto de recursos que podría provocar escasez de ancho de banda y por tanto futuras congestiones en las comunicaciones de la red.

Como resultado de la situación anterior, protocolos *reactivos* han sido diseñados de tal forma que la información acerca de una ruta sea obtenida sólo cuando sea realmente necesaria. Los protocolos reactivos podrían usar menos ancho de banda para mantener las tablas en cada nodo, pero el tiempo de respuesta para muchas aplicaciones se incrementaría. La mayoría de las aplicaciones se verían afectadas por un retraso a la hora de comenzar, ya que será necesario obtener una ruta hacia algún destino antes de que la comunicación se inicie.

1.7.2. Seguridad

Obviamente, la seguridad es un punto importante a considerar dentro de las redes ad hoc, particularmente si se utilizan múltiples puntos para comunicar a un nodo con otro. ¿Cómo puede un usuario estar seguro de que nadie esté escuchando secretamente lo que transmite? ¿Es el usuario en el otro extremo del enlace realmente quien dice ser? Desde el punto de vista puramente criptográfico, los servicios ofrecidos en una red ad hoc no implican muchos problemas nuevos. Los requisitos para la autenticación, confidencialidad e integridad son los mismos que para muchas otras redes de comunicación públicas.

Sin embargo, en una red ad hoc, la confianza es un problema central. No podemos confiar en el medio, la única opción es usar criptografía, lo cual obliga a depender de las llaves criptográficas utilizadas. Entonces, el reto principal es crear relaciones basadas en confianza entre las llaves sin la ayuda de un tercer ente de certificación.

Dado que las redes ad hoc se crean espontáneamente entre entidades que se encuentran dentro de una misma área, no hay garantía de que cada nodo mantenga la llave pública de los demás nodos o que estos presenten certificados que sean validados por otro ente. Sin embargo, si permitimos que la confianza se delegue entre los nodos, entonces nodos que ya hayan establecido una relación de confianza pueden extender este privilegio a otros

miembros del grupo.

Ya que este trabajo no estará enfocado en este aspecto, sólo mencionaremos que en otros trabajos como en [13] se puede encontrar más información al respecto.

1.8. Campos de aplicación

Las redes ad hoc móviles han sido motivo de mucha investigación reciente y esfuerzos de desarrollo. Hasta ahora, este tipo de redes han sido principalmente consideradas para aplicaciones de tipo militar, donde una configuración de red descentralizada es una ventaja operativa o incluso una necesidad. Otro punto que motivó la utilización de esta tecnología en este campo es que la milicia no puede confiar en un acceso a una infraestructura de comunicación fija y precolocada en un campo de batalla. En algunas regiones, como desiertos o selvas, la existencia de una infraestructura de comunicación terrestre es nula. En otras regiones, el acceso puede no estar disponible debido a la destrucción o el daño de la infraestructura de comunicación local.

En el sector comercial, el equipamiento para cómputo móvil e inalámbrico no ha estado disponible a un precio atractivo para grandes mercados. Sin embargo, conforme se incrementa la capacidad de computadoras móviles, también crece la necesidad de una red sin límites. Las redes ad hoc comerciales podrían ser usadas en situaciones donde no se cuente con una infraestructura disponible, tal como en operaciones de rescate en áreas remotas o cuando una cobertura local debe utilizarse eficiente y rápidamente en algún sitio de construcción remoto. Así también, podría utilizarse como acceso público inalámbrico en áreas urbanas.

A nivel local, esta tecnología puede ser usada para unir computadoras portátiles y/o PDA's de tal manera que se pueda difundir y compartir información en una conferencia. Estas redes también podrían ser propias para aplicaciones en redes caseras, donde los dispositivos puedan comunicarse directamente para intercambiar información, como audio, video o actualizaciones de configuración. Tal vez las aplicaciones más difíciles de lograr en este contexto son redes de tipo autónomo, donde los nodos que las conformen sean robots que lleven a cabo tareas domésticas o actividades de vigilancia.

Algunas personas han incluso propuesto el utilizar redes ad hoc para el monitoreo ambiental, de tal manera que las redes puedan ser usadas para la predicción de la contaminación del agua o para proveer con una buena anticipación una alerta de maremoto.

Redes ad hoc móviles de corto alcance pueden simplificar la intercomunicación entre varios dispositivos móviles (como teléfonos celulares y PDA's) para formar una red de área personal (personal area network, PNA) y por tanto eliminar la tediosa necesidad de cables. Esto podría también extender la movilidad proveída por la red fija, es decir, el protocolo de Internet móvil, a nodos que se encuentran más allá del dominio de la red ad

hoc. La tecnología *Bluetooth* es tal vez la más prominente en este ámbito.

Con esto damos por terminada la presentación de uno de los conceptos más importantes que se ocuparán a lo largo de este trabajo. Ya en los siguientes capítulos, se especificará su utilización junto con otros conceptos.

La técnica de localidad

2

2.1. Motivación

Como ya lo mencionamos en el capítulo anterior, uno de los principales problemas en las redes ad hoc tiene su origen en la característica de que la comunicación directa de cada uno de los nodos que la conforman se restringe a un subconjunto de ellos, conocidos como sus vecinos. Por otro lado, a pesar de que la comunicación de cada nodo se restringe a un *esquema local*, la totalidad del sistema debe hallar algún tipo de solución global que le permita seguir proveyendo los servicios que esta ofrece.

Sin embargo, el lograr un *objetivo global* basándose en información de *tipo local* es algo desafiante. Ninguno de los nodos de una red ad hoc es capaz de mantener información global acerca de la red. En lugar de esto, cada uno de los nodos debe llevar a cabo una tarea específica haciendo uso solamente de información local y cuyos resultados impacten de manera global en la red.

Dada esta problemática, en este capítulo haremos algunas observaciones acerca de la utilización de esta técnica, que por la forma en la que operan las redes ad hoc, la hacen una línea de investigación importante.

2.2. Alcances

Algunas de las cosas que uno tiene en mente a la hora de trabajar en esta área es qué tipo de tareas globales pueden llevarse a cabo por los nodos de la red de manera que estos basen sus decisiones en información local, o cuánta información local se requiere para dar una solución global óptima.

A los algoritmos que hacen uso de esta técnica de localidad se les conoce como *algoritmos locales*, en los cuales, al aplicarlos en una red, cada nodo conoce solamente a sus vecinos, o en general a los vecinos a distancia a lo más k , con k una constante.

Claramente existen algunos problemas que no pueden solucionarse de manera local, como el que dada una red, se calcule un árbol generador¹ de esta. En particular para dicho problema, inclusive no es posible averiguar localmente si un conjunto de aristas dado forman un árbol generador de la gráfica que represente a dicha red, ya que para esto cada nodo necesitaría hacer uso de información global de la red para evitar la formación de algún ciclo, lo cual no es posible dentro del esquema de localidad.

No obstante, para algunas tareas como el colorear una gráfica con un número dado de colores c , es al menos posible el comprobar la viabilidad de una asignación dada con tan sólo observar a los nodos vecinos. Entonces, ¿basta el conocer información local para asignar colores a los nodos de tal manera de que obtengamos una c -coloración propia (que dos nodos adyacentes no tengan el mismo color)? Investigación concerniente a esta pregunta ya ha sido desarrollada, [22] es un buen ejemplo de esto. Para darnos una idea de la utilización de esta técnica podemos tomar como ejemplo el problema descrito en la siguiente subsección.

2.2.1. Agrupamiento en una red ad hoc

Como ya se estableció en el capítulo 1, cuando se envía un mensaje en una red ad hoc de un nodo a otro, los nodos intermedios deben servir como ruteadores y debido a la escasez de energía y a la movilidad de los nodos en estos sistemas, la aplicación de los protocolos tradicionales (de redes alámbricas) como el mantener actualizadas adecuadamente tablas de ruteo se convierte en un problema, ya que en cualquier momento, por ejemplo, uno o más nodos podrían dejar la red sin previo aviso y las rutas que los consideraban para el paso de mensajes ya no servirían. De hecho, aunque se ha desarrollado un gran número de sugerencias al respecto ([6, 33]), el encontrar algoritmos eficientes para dar solución a este proceso aún sigue siendo uno de los principales problemas dentro del área.

El agrupamiento de los nodos ([3]) es una manera efectiva de ahorrar energía ([18]) y en general de mejorar el desempeño de los algoritmos que permiten establecer rutas entre nodos de la red, ya que el determinar tales rutas se hace considerando los distintos grupos, cada uno de estos consistiendo de una cabeza y de un número de nodos vecinos. Por supuesto, lo que deseamos es minimizar el número de grupos bajo la condición de que cada nodo de la red pertenezca a uno de ellos, es decir, lo que queremos es encontrar el mínimo número posible de cabezas de grupo de tal forma que cada nodo que no sea cabeza tenga a uno que si lo sea dentro de su vecindad. Formalmente, a este problema se le conoce como el problema del *conjunto dominante mínimo* ([1, 2]).

Para establecer una ruta entre dos nodos, usualmente las cabezas de grupo se conectan entre sí haciendo uso de otros nodos que sirvan como puentes. Alternativamente, existe una variante a este problema, conocido como el problema del *conjunto dominante conexo*

¹En el sentido de la teoría de gráficas se desea que la gráfica obtenida tenga el mismo conjunto de vértices de la gráfica original.

([2, 8, 23]), en donde las cabezas de grupo pueden seleccionarse de tal manera que la gráfica inducida por ellos sea conexa.

2.3. El primer algoritmo local para establecer rutas

Como ya lo hemos mencionado, uno de los principales retos en el diseño de las redes ad hoc tiene que ver con el desarrollo de protocolos dinámicos que puedan encontrar eficientemente rutas entre cualquier par de nodos de la red. En la última década, se han propuesto algunos protocolos específicos para este tipo de redes que llevan a cabo dicha tarea, tales como [7, 20, 34]; sin embargo, no hacen uso de la localidad descrita.

En [21], Kranakis, Singh y Urrutia propusieron el primer algoritmo que permitía establecer una ruta entre un nodo origen s y un nodo destino t teniendo como base una gráfica con ciertas características y haciendo uso de la técnica de localidad. El objetivo de dicho trabajo era precisamente el de desarrollar un algoritmo que pudiera aplicarse a las redes de comunicación existentes y cuya única restricción es que las gráficas que las representen sean planas.

Además, establecieron formalmente los puntos que necesitan ser tomados en cuenta para determinar que un algoritmo sea considerado como uno de tipo local. Dichas condiciones son las siguientes:

1. En cada punto en el tiempo, un agente que haga uso del algoritmo, conoce las coordenadas del nodo origen s , así como también las del nodo destino t . Además, este dispone de una cantidad finita de almacenamiento en la cual se puede mantener un número constante de identificadores de vértices de la red. Observemos que esta condición implica que en ningún momento se tiene conocimiento de la topología de la red en su totalidad.
2. Al llegar el agente a un vértice v , habiendo comenzado en s , se puede hacer uso de la información local almacenada en él en lo que concierne a sus vecinos, así como también de las aristas que los conectan con v . Usando esta información mas la almacenada en su memoria local, el agente puede seleccionar una arista incidente a v para atravesarla y continuar su camino hacia t , al menos de que v ya sea t , en cuyo caso nos detenemos.
3. No se permite que el agente cambie la información local almacenada en v . Observemos que en particular, una vez que el agente salga de un nodo, si posteriormente regresa a él, no se puede determinar que ya se ha visitado, al menos que su identificador se encuentre en la memoria local.

Cabe mencionar que la tercera condición sobre un algoritmo que se considere local se debe principalmente al hecho de que no se quiere dejar marcas sobre los vértices que

se vayan visitando mientras se trata de llegar al vértice destino, ya que de ser así, se podría tener un problema. Por ejemplo, si suponemos que tenemos un servidor conectado a Internet, quisiéramos poder evitar el mantener el rastro de los mensajes que pasen por ahí, ya que el hacerlo podría requerir fácilmente de una gran cantidad de memoria si es que nuestro servidor es muy solicitado como puente en la comunicación de mensajes.

El algoritmo desarrollado se conoce como *Ruteo por Brújula (Compass Routing)* y su comportamiento puede ser descrito de la siguiente manera:

Supongamos que queremos enviar un mensaje de un nodo inicial s a un nodo final t y recordemos que la única información con la que se cuenta en cualquier punto en el tiempo son las coordenadas del nodo destino, la posición actual y las direcciones de las aristas incidentes al vértice en el que estamos parados. De manera recursiva se escoge y se recorre una arista $e = (u, v)$ de la gráfica geométrica incidente a la posición actual, u , de tal manera que la abertura formada entre e y la recta que une a u con t sea la más pequeña, es decir, se selecciona una arista tal que el ángulo $\angle vut$ sea el menor de entre todos los vecinos de u en la topología.

Así, para la figura 2.1, la ruta que habría que recorrer para llegar de s a t al aplicar nuestro algoritmo es s, d, e, g, t .

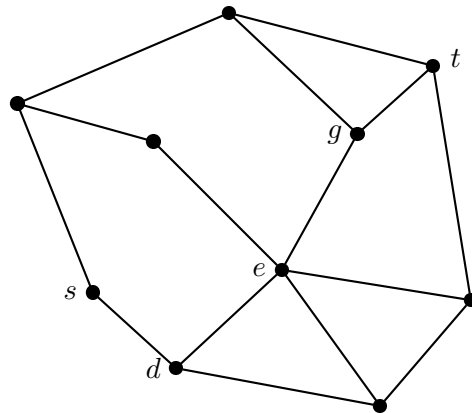


Figura 2.1: Uso del ruteo por brújula para establecer una ruta de s a t .

No obstante, el aplicar el algoritmo de ruteo por brújula en una gráfica geométrica no nos garantiza que siempre se pueda establecer una ruta desde un punto inicial a cualquier otro vértice en la gráfica. Para demostrarlo, tomemos como ejemplo la gráfica de la figura 2.2, en la cual al aplicar el algoritmo para ir de s a t caemos en un ciclo que contiene al conjunto de vértices $\{u_0, v_i\}$ con $i = 0, \dots, 3$.

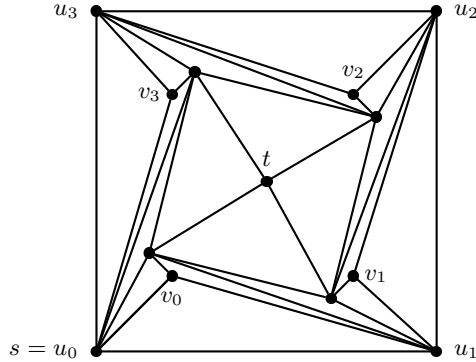


Figura 2.2: Problema al aplicar el ruteo por brújula para alcanzar t desde s

Si para cualquier par de vértices s y t de una gráfica geométrica G , el algoritmo establece una ruta cuyo vértice inicial es s y termina en t , entonces decimos que G soporta el ruteo por brújula. Dado lo anterior, la pregunta que se plantea es en qué tipo de redes geométricas es posible utilizar este algoritmo de manera que garantizemos que el envío de un mensaje llegue a su destino.

Kranakis, Singh y Urrutia también demostraron en [21] que si el conjunto de puntos de la gráfica geométrica junto con sus aristas forman una triangulación de Delaunay, entonces la gráfica soporta el ruteo por brújula. Antes de demostrar este hecho, establezcamos formalmente el concepto de una triangulación de este tipo.

Definición 1. La *triangulación de Delaunay* de un conjunto de puntos P_n en el plano, denominada $\mathcal{D}(P_n)$, es la partición de su cierre convexo en un conjunto de triángulos con interiores ajenos de tal manera que se cumplan las siguientes restricciones:

- los vértices de los triángulos son puntos de P_n y
- para cada triángulo en la triangulación, el círculo que pasa por sus vértices no contenga ningún punto de P_n en su interior (ver figura 2.3).

Se sabe que si los elementos de P_n están en *posición circular general*, es decir, que no existen cuatro puntos por los que pase un círculo, entonces $\mathcal{D}(P_n)$ está bien definida.

Dado lo anterior, tenemos el siguiente resultado:

Teorema 1. Sea P_n un conjunto de puntos en el plano, entonces $\mathcal{D}(P_n)$ soporta el ruteo por brújula.

Demostración. La idea es mostrar que si el algoritmo de ruteo por brújula selecciona una arista (u, z) para recorrerla, entonces la distancia de z a v es estrictamente menor que la distancia de u a v , y dado que $\mathcal{D}(P_n)$ tiene un número finito de vértices, el

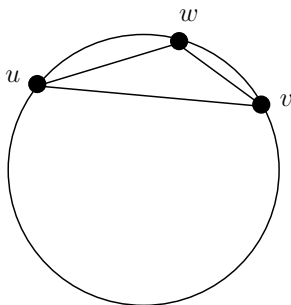


Figura 2.3: Triángulo que pertenece a la triangulación de Delaunay.

mostrar que se da esta condición es suficiente para probar que eventualmente se llegará a v .

Dada $\mathcal{D}(P_n)$, supongamos que queremos ir de un vértice u a otro vértice v . Sea \overline{uv} el segmento de recta que une a u y v y demos por hecho que este intersecta al triángulo $\triangle uxy$. Sea C el círculo que pasa por los vértices de dicho triángulo y c su centro. También, definamos como u' al punto espejo de u en el círculo con respecto a la línea que pasa por c y t . Todo esto lo podemos apreciar en la figura 2.4.

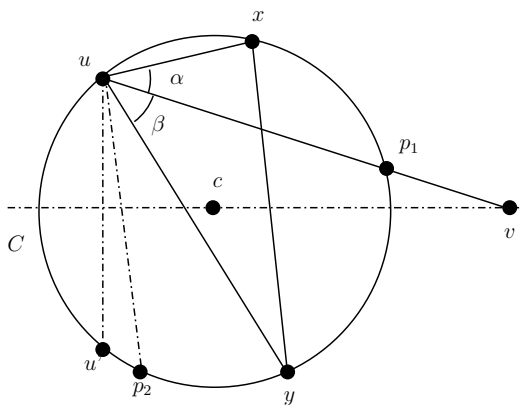


Figura 2.4: Establecimiento de rutas en la triangulación de Delaunay.

Sean α y β los ángulos $\angle xuv$ y $\angle vuy$ respectivamente. Dado esto, tenemos dos casos posibles:

$\alpha < \beta$. En este caso, el algoritmo de ruteo por brújula escogería la arista (u, x) y es claro que la distancia entre x y v es menor que la que hay entre u y v .

$\beta \leq \alpha$. Sea p_1 el punto de intersección del segmento de recta \overline{uv} con el círculo C y sea p_2 el punto sobre C tal que su distancia hacia p_1 sea igual a la distancia de u a p_1 . Con esto, podemos asegurar que p_2 se encuentra en el arco abierto C' , que va de u' a u

en el sentido contrario a las manecillas del reloj. Además, dado que $\beta \leq \alpha$, entonces y debe estar en C' , por lo que su distancia a v es menor que la distancia de u a v .

Entonces, de los casos anteriores, podemos concluir que nuestro algoritmo siempre establece una ruta entre cualesquiera dos vértices de $\mathcal{D}(P_n)$, es decir, $\mathcal{D}(P_n)$ soporta el algoritmo de ruteo por brújula. ■

2.4. Garantía en la entrega de paquetes

Antes de continuar, establezcamos algunos puntos básicos acerca de las redes ad hoc que se tomarán como base para los resultados que comentaremos a continuación.

Podemos suponer que todos los nodos inalámbricos que conforman la red cuentan con identificadores únicos y que en cualquier momento un nodo puede saber su posición, ya sea a través de un sistema de posicionamiento global (GPS) o algún otro medio similar. Hoy en día podemos hacer esta suposición gracias a que tales sistemas se han mejorado en precisión y también a que su tamaño y costo cada vez son menores, por lo que el tener acceso a ellos ya no es problema.

También, daremos por hecho que todos los nodos tiene el mismo alcance de transmisión, el cual tomaremos como nuestra unidad. Por otro lado, con una simple comunicación, cada nodo puede recolectar la información acerca de la posición de cada uno de los nodos dentro de su radio de transmisión.

Dado lo anterior, el conjunto de nodos de una red definen lo que se conoce como una *gráfica de disco unitario* o *UDG*, en la cual, habrá una arista entre un par de nodos u y v si y sólo si la distancia euclidiana entre ellos (normalizada) es menor o igual que la unidad, es decir, $\|uv\| \leq 1$.

Retomando el problema de establecer una ruta entre dos vértices de una gráfica, Kranakis *et al.* [21] propusieron otro algoritmo local denominado *Ruteo por Caras (Face Routing)* que puede aplicarse en gráficas planas y conexas.

Una gráfica plana y conexa G divide el plano en caras que están acotadas por poligonales definidas por las aristas de G . Dado un vértice en una cara F , su frontera puede recorrerse en sentido contrario a las manecillas del reloj (salvo por la cara exterior, la cual se recorrerá en el mismo sentido de las manecillas) usando la *regla de la mano derecha*. Teniendo esto como base, el algoritmo propuesto debe tener disponible en cualquier momento la siguiente información:

- Las posición del nodo origen, s , y la del nodo destino, t .
- La posición de los últimos dos vértices que se visitaron.

- Un punto p , que al comienzo, tiene el valor de s .
- Una distancia d , que inicialmente toma el valor de la distancia que hay entre s y t .

Sea $\mathcal{L} = \overline{pv}$ el segmento de recta que une a p y v (recordemos que al principio, $p = s$). Notemos que dicho segmento siempre estará disponible, dado que el algoritmo siempre recordará las coordenadas de p y v . Con dicha información, el algoritmo de ruteo por caras puede describirse de la siguiente manera:

Primeramente, el algoritmo detecta la cara, \mathcal{F} , incidente a p e intersectada por \mathcal{L} . El motivo de esto es que podamos recorrer \mathcal{F} hasta regresar al punto inicial del recorrido, considerando que si el vértice destino forma parte de \mathcal{F} , entonces el algoritmo se detendrá en el instante en que se llegue a él. Como lo mencionamos anteriormente, \mathcal{F} está definida por las aristas de la gráfica G , por lo que el recorrer la cara significa recorrer las aristas que la definen. Al recorrer una de las aristas e , el algoritmo puede caer en alguno de los siguientes casos:

- Que la arista e interseque a \mathcal{L} en un punto p' , en cuyo caso se calcula la distancia del punto de intersección al vértice t , llamémosla d' . Si $d' < d$, entonces d y p tomarán nuevos valores, el de d' y p' respectivamente.
- Que e no interseque a \mathcal{L} , por lo que simplemente continuamos recorriendo la cara.

Si t no es parte de la cara \mathcal{F} , el algoritmo regresará al inicio del recorrido teniendo un nuevo valor para d y p . En este punto, el algoritmo recorrerá la cara nuevamente hasta llegar a la posición p en una de las aristas, digamos a . Cabe mencionar que cualquier arista que pertenezca a la gráfica G , pertenece a la frontera de dos caras distintas, que para el caso de a son \mathcal{F} y \mathcal{F}' . A partir de esto, el algoritmo hará un cambio de cara en p , de \mathcal{F} a \mathcal{F}' , iniciando ahí de nueva cuenta el proceso de recorrido ya descrito, pero ahora en la cara \mathcal{F}' .

La idea del algoritmo de ruteo por caras es ir recorriendo las caras de la gráfica que representa a nuestra red ad hoc de tal manera que aseguremos que cada vez que crucemos hacia una cara nueva, realmente nos estemos acercando hacia el nodo destino. Lo anterior es posible gracias a que se tiene disponible la información de las posiciones tanto del nodo origen como la del nodo destino. Así también, es claro ver que si la gráfica G es conexa, el paquete que se envíe desde s siempre llegará a t . Como ejemplo consideremos la figura 2.5, la cual muestra la aplicación de este algoritmo en una gráfica geométrica.

Dado lo anterior, se puede plantear el siguiente teorema:

Teorema 2. *Existe un algoritmo de establecimiento de rutas en gráficas geométricas que hace uso de información local, el cual garantiza el alcance del nodo destino. Mas aún, tal algoritmo llega a dicho vértice en a lo más $4||E||$ pasos, donde $||E||$ es el número de aristas en la gráfica.*

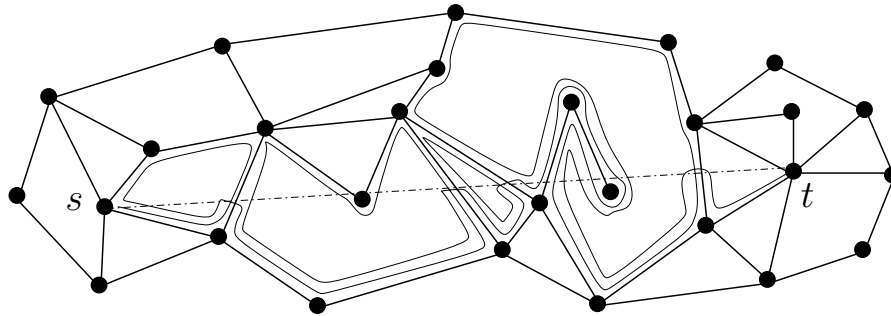


Figura 2.5: Uso del ruteo por caras en una gráfica geométrica.

La demostración a este hecho, así como la de otros resultados relacionados se pueden consultar en [6, 21].

2.5. Consideraciones en el proceso de establecimiento de rutas

Como lo hemos mencionado hasta ahora, el problema de establecer una ruta entre cualquier par de nodos que conformen una red ad hoc ha sido una gran fuente de investigación y es por esto que varios algoritmos que hacen uso de información local han sido desarrollados.

Ya describimos el uso del algoritmo de ruteo por brújula propuesto por Kranakis *et al.* en [21]. Por otro lado, Lin *et al.* ([29]) y Bose *et al.* ([6]) propusieron algoritmos “glotones” que también dan solución a este problema, en los cuales un nodo u envía el paquete que se desea transmitir a su vecino v si este se encuentra en una topología cercana al vértice destino, basándose en la gráfica de Gabriel (la cual describiremos un poco más adelante). Así también, Bose *et al.* ([5]) propusieron un método que hace uso de información local para establecer una ruta entre un par de vértices en la triangulación de Delaunay de una red de manera que dicha ruta tenga un costo constante con respecto a la distancia entre el vértice origen y el destino.

Dentro de cada una de las soluciones mencionadas, se han tomado como base ciertas características de la gráfica sobre la cual se aplican los algoritmos, como por ejemplo, que sea plana o que sea una triangulación de Delaunay. Sin embargo, al tener la gráfica que representa a una red ad hoc, esta no forzosamente tiene que cumplir con dichas características, por lo que se toma un subconjunto de sus aristas de tal manera que la gráfica generadora inducida por tal subconjunto se acople a la solución.

Un claro ejemplo de lo anterior lo tenemos con el algoritmo de ruteo por caras, en el cual, tomabamos como un hecho que teníamos una gráfica plana. Entonces, si existe al menos un cruce entre un par de las aristas de la gráfica que representa nuestra red ad

hoc, el algoritmo podría no servir.

Una buena solución al problema anterior es obtener lo que se conoce como la *subgráfica de Gabriel* ([14]), ya que esta tiene la propiedad de ser plana, además de que si la gráfica original es conexa, la subgráfica obtenida también lo es. Decimos que una arista que une a dos vértices de nuestra gráfica G es de Gabriel si la circunferencia cuyo diámetro es el segmento de recta que une a tales vértices no contiene algún otro vértice de G . Entonces, la *subgráfica de Gabriel* de la gráfica G se define como la subgráfica que contiene a todas las aristas de Gabriel. La figura 2.6 muestra este concepto, en donde las aristas (q, r) y (r, s) se conservan, mientras que la arista (p, r) se descarta como arista de Gabriel.

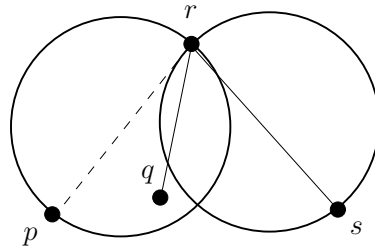


Figura 2.6: Selección de las aristas de Gabriel.

Podemos mencionar también que la obtención de la gráfica de Gabriel fue la primera solución que se planteó para obtener una subgráfica plana de una gráfica de disco unitario; sin embargo, hoy en día dentro de la literatura podemos encontrar algunas otras opciones, las cuales se conocen como *Gráficas de cercanía* ([36, 27]) y muchas de las cuales se pueden construir de manera local para lograr obtener una subgráfica plana de la gráfica que representa a una red ad hoc.

Una de las gráficas de cercanía más usadas es la *gráfica de vecinos relativos* o *RNG* (acrónimo de *Relative Neighbourhood Graph*), la cual consiste de todas las aristas (u, v) tales que $\|uv\| \leq 1$ (ya que estamos hablando de una UDG) y que no existe vértice w en $V(G)$ tal que $\|uw\| < \|uv\|$ y $\|wv\| < \|uv\|$.

Tomemos como ejemplo la figura 2.7, en la cual la arista (u, v) formará parte de la RNG, ya que la intersección de las circunferencias con centros en u y v con radio $\|uv\|$ no contiene a ningún otro vértice de la UDG, que es precisamente la definición que acabamos de dar.

Al igual que en la gráfica de Gabriel, la gráfica de vecinos relativos tiene la propiedad de ser plana, además de que si la gráfica original era conexa, la RNG lo es también. Por otra parte, es fácil ver que cualquier arista de la RNG de una gráfica G es también una arista de Gabriel, mas no al revés. Para ejemplificar más claramente estas dos gráficas, tomemos la figura 2.8, donde (a) representa a una gráfica de disco unitario y (b) y (c)

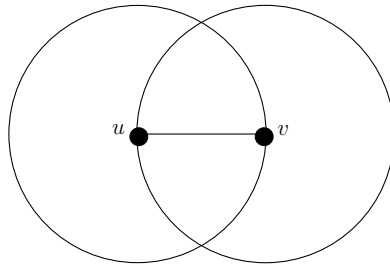


Figura 2.7: Selección de las aristas para la RNG.

representan su gráfica de Gabriel y su gráfica de vecinos relativos respectivamente. En particular, la arista e_1 pertenece a ambas gráficas, pero la arista de Gabriel e_2 no es una arista de la RNG.

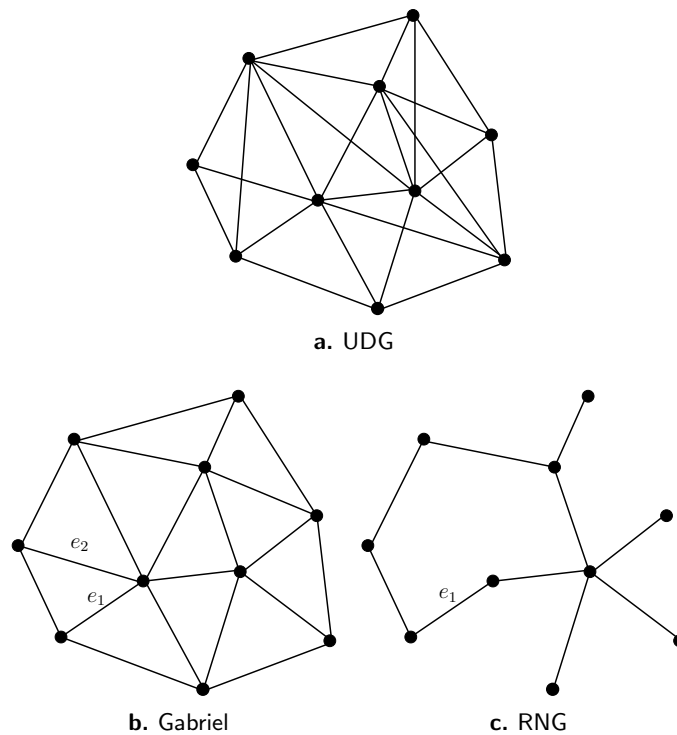


Figura 2.8

Otra de las gráficas de aproximación es la gráfica de Delaunay, cuyas aristas son parte de una triangulación de Delaunay entre los nodos que la componen. La principal desventaja de esta gráfica, de acuerdo a nuestros propósitos, es que no se puede construir de manera local; sin embargo, tiene propiedades que la hacen una gráfica interesante para la cuestión del establecimiento de una ruta entre un par de vértices

de una UDG. No obstante a este obstáculo, existe una familia de gráficas que obtienen una *triangulación de Delaunay local* ([27, 1]) y que heredan muchas de las propiedades de la gráfica de Delaunay, siendo lo más importante que pueden calcularse de manera local.

Hablando un poco más en general con respecto al problema original del establecimiento de rutas localmente, aún no se ha podido desarrollar un algoritmo que actúe sobre cualquier tipo de gráfica, por lo que en los últimos años las distintas investigaciones al respecto se han visto forzadas a encaminarse a obtener primeramente una subgráfica de la gráfica que representa a la red y ya sobre esta basar el desarrollo de un nuevo algoritmo. Para dicha subgráfica, se ha establecido un conjunto de características que sería deseable obtener, características que impactan directamente en qué tan óptimo será el algoritmo que permita llevar a cabo la tarea de establecer una ruta. Entonces, dada una UDG, G , las propiedades que deseamos tenga la subgráfica G' son las siguientes:

- G' sea plana.
- Si G es conexa, entonces que G' sea conexa.
- G' sea una *gráfica generadora*² de G , es decir, que además de tener ambas el mismo conjunto de vértices, la longitud o costo (algunas veces dado por la distancia euclidiana) de la trayectoria más corta en G' sea a lo más t veces el costo o la longitud de la trayectoria más corta en G . Si esto pasa, llamamos a G' una *gráfica t -generadora* de G y a t el *factor de estiramiento* (*stretch factor*).
- El *grado máximo de la gráfica*, $\Delta(G)$, esté acotado, es decir, $\Delta(G) \leq c$, con c una constante.

Para finalizar este capítulo sólo mencionaremos que a partir de la aparición de los algoritmos de ruteo por brújula y ruteo por caras, han surgido algunos otros que se basan en ideas completamente distintas y otros que los toman como base ([24, 27]), pero la línea de investigación continúa siendo la misma: llevar a cabo las operaciones o pasos necesarios de manera local.

²En el área de Teoría de Gráficas, el concepto de una gráfica generadora se maneja de manera distinta. Se pide solamente que los conjuntos de vértices de las gráficas G y G' respectivamente sean los mismos.

3.1. Motivación

En los dos capítulos anteriores, desarrollamos los conceptos generales sobre los que se basa este trabajo: el primero fue la descripción de una red ad hoc, que es el objeto sobre el cual trabajaremos, y el segundo fue la manera en la que trataremos tal objeto, que es aplicando la técnica de localidad en el o los algoritmos que nos permitan obtener una solución a un problema dado.

En este capítulo revisaremos dos resultados que jugarán un papel importante en el desarrollo del resultado principal para este trabajo. El primero tiene que ver con la coloración de las aristas de una gráfica y el segundo con la obtención de manera local de una subgráfica de la gráfica que representa a la red ad hoc, con base en la idea del árbol generador de peso mínimo .

3.2. Teorema de Vizing

Antes de establecer el resultado principal de esta sección, que es precisamente el *Teorema de Vizing*, comenzaremos por establecer algunas definiciones que nos ayudarán a su comprensión.

Definición 2. Una *coloración (propia) por aristas* de una gráfica G , es una asignación de colores a sus aristas de tal forma que a aristas adyacentes les correspondan colores distintos. Una coloración por aristas de G parte al conjunto de aristas, $E(G)$, en conjuntos independientes (que no existe una arista entre cualquier par de sus elementos) denominados *clases cromáticas de aristas*, cada una de las cuales consiste de todas las aristas a las que se les asigna un color en específico.

Definición 3. Dada una coloración por aristas de una gráfica G en la que se utilizaron k colores, decimos que G posee una *k -coloración*. Así mismo, podemos decir que G es *k -coloreable*.

Definición 4. El mínimo número k , $k \geq 0$, para el cual una gráfica G es k -coloreable se denomina como su *número cromático por aristas* o simplemente *índice cromático* y se denota por $\chi_1(G)$.

Definición 5. Una gráfica G es r -regular o *regular de grado r* si cada vértice de G tiene grado r .

Definición 6. Un *factor* de una gráfica G es una *subgráfica generadora*¹ de G , es decir, una subgráfica F , cuyo conjunto de vértices es igual al conjunto de vértices de G , esto es, $V(G) = V(F)$. Un factor r -regular de una gráfica G es un r -factor de G .

Definición 7. Supongamos que G_1, G_2, \dots, G_n son subgráficas generadoras de G , las cuales cumplen que para cualesquiera dos de ellas, la intersección de su conjunto de aristas es vacía (son ajenas dos a dos por aristas). Así también, demos por hecho que $\bigcup_{i=1}^n E(G_i) = E(G)$. En esta caso decimos que G es *factorizable* en factores G_1, G_2, \dots, G_n . Análogamente, si existe una factorización de G en r -factores, entonces decimos que G es *r -factorizable*.

Es importante hacer notar que en una coloración de las aristas de una gráfica G , las aristas adyacentes deben tener colores distintos, por lo que se tiene que el índice cromático de la gráfica es al menos tan grande como el grado máximo de sus vértices, es decir, $\chi_1 \geq \Delta(G)$.

Como ejemplo, tomemos las gráficas K_4 y C_5 de la figura 3.1, donde $\Delta(K_4) = 3$ y $\Delta(C_5) = 2$. De esto podemos concluir que $\chi_1(C_5) \geq 2$; sin embargo, $\chi_1(C_5) \neq 2$ ya que si tratáramos de colorear las aristas de C_5 con sólo dos colores, digamos 1 y 2, tendríamos que alternar estos en sus aristas, obteniendo que dos de ellas, adyacentes, tuviesen el mismo color, lo cual no se permite. Por otro lado sí existe una 3-coloración de C_5 (b), por lo que $\chi_1(C_5) = 3$. Así también, se puede hacer un análisis similar para la gráfica K_4 (a).

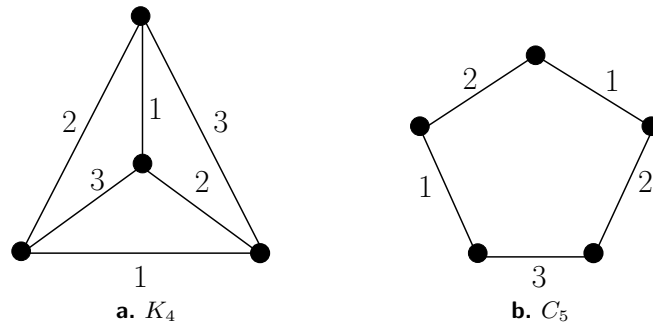


Figura 3.1: Índice cromático de K_4 y C_5 .

¹En el sentido de teoría de gráficas

Otro hecho que podemos hacer notar es que si G es una gráfica r -regular, para algún entero $r \geq 1$, entonces $\chi_1(G) \geq r$. Si $\chi_1(G) = r$, entonces cada una de las r clases cromáticas debe inducir un 1-factor de G , por lo que G debe ser 1-factorizable. Recíprocamente, si G es 1-factorizable, entonces $\chi_1(G) = r$. Por lo tanto, $\chi_1(G) = r$ si y sólo si G es 1-factorizable. Esta observación es útil en la demostración del siguiente resultado, la cual puede consultarse en [10].

Teorema 3. *Si G es una gráfica no vacía bipartita, entonces $\chi_1(G) = \Delta(G)$.*

Ya mencionamos que $\chi_1(G) \geq \Delta(G)$ para cada gráfica no vacía G , además de que esta desigualdad debe ser estricta si G no es bipartita. No obstante, Vizing demostró que si $\chi_1(G) \neq \Delta(G)$, entonces $\chi_1(G) = 1 + \Delta(G)$. Esta afirmación es uno de los resultados más importantes en lo que a coloración de aristas se refiere.

A continuación, presentamos la demostración del resultado propuesto por Vizing ([10]), ya que esencialmente proporciona un algoritmo de tiempo polinomial para obtener una $(\Delta + 1)$ -coloración de una gráfica no vacía G .

Teorema 4 (Teorema de Vizing). *Para cada gráfica no vacía G ,*

$$\chi_1(G) \leq \Delta(G) + 1$$

Demostración. Por contradicción, demos por hecho que el teorema es falso. Entonces, existen gráficas H para las cuales $\chi_1(H) \geq 2 + \Delta(H)$. De entre todas estas gráficas H , tomemos una gráfica G de tamaño² mínimo y denotemos por Δ a $\Delta(G)$. Por consecuencia, no existe una $(\Delta + 1)$ -coloración de G . Por otro lado, si $e = (u, v)$ es una arista de G , entonces existe una $(\Delta(G - e) + 1)$ -coloración de $G - e$.

Ya que $\Delta(G - e) \leq \Delta(G)$, la gráfica $G - e$ es $\Delta + 1$ -coloreable. Dado esto, consideremos una $(\Delta + 1)$ -coloración de $G - e$. Entonces, cada arista de G , excepto e , tiene asignado uno de los $\Delta + 1$ colores de tal manera que aristas adyacentes tengan colores diferentes.

Para cada arista $e' = (u, v')$ de G , que es incidente a u (incluyendo a e), definamos el *color dual* de e' como cualquiera de los $\Delta + 1$ colores que no sean usados para colorear las aristas incidentes a v' . Dado que ningún vértice de G tiene grado mayor que Δ , entonces siempre hay al menos un color disponible para ocuparlo como el color dual. Así también, notemos que es posible que aristas distintas tengan el mismo color dual.

Sea c_1 el color dual para $e_0 = e$ (lo cual implica que c_1 no es el color de ninguna de las aristas de G que sean adyacentes a v). Ahora bien, debe haber una arista e_1 incidente a u a la que se le haya asignado el color c_1 , ya que de no ser así, e podría colorearse de color

²El tamaño de una gráfica es el número de vértices en ella.

c_1 y esto produciría una $(\Delta + 1)$ -coloración de G , lo cual no es posible (debido a nuestra hipótesis). Sea c_2 el color dual de e_1 . Si hay una arista incidente a u a la que se le haya asignado el color c_2 , llamémosla e_2 y a su color dual c_3 . De esta manera, podemos construir una secuencia $S = e_0, e_1, \dots, e_k$ ($k \geq 1$) que contenga un número máximo de aristas distintas. La última de las aristas de tal secuencia, e_k , tiene asignado color c_k y color dual c_{k+1} .

Si ninguna de las aristas de G incidentes a u tiene asignado el color c_{k+1} , entonces a cada una de las aristas de S podría asignársele su color dual para producir una $(\Delta + 1)$ -coloración de G . Ya que dicha coloración es imposible, existe una arista e_{k+1} de G incidente a u que tiene asignado el color c_{k+1} y dado que S contenía el máximo número de aristas distintas, $e_{k+1} = e_i$ para alguna i ($1 \leq i \leq k$), o, de manera equivalente, $c_{k+1} = c_i$ para alguna i ($1 \leq i < k$).

Dado lo anterior, podemos hacer algunas observaciones con respecto a la coloración de $G - e$. La arista e no puede tener asignado ninguno de los $\Delta + 1$ colores sin que esto produzca que dos aristas adyacentes tengan el mismo color. Por lo tanto, para cada color c de entre los $\Delta + 1$ colores, hay una arista de G adyacente a e que tiene color c . Entonces, debe haber colores asignados a las aristas incidentes a v y que no están asignados a ninguna arista incidente a u ; sea b uno de estos colores. Ahora, sea $e_i = (u, v_i)$ ($0 \leq i \leq k$), con $v_0 = v$. El color b debe asignarse a alguna arista incidente a v_i para cada i , $1 \leq i \leq k$. Supongamos que existe un vértice v_m , $1 \leq m \leq k$, tal que ninguna arista adyacente a él se colorea con el color b . Podríamos entonces cambiar el color de e_m por b y colorear e_i , $0 \leq i < m$ con su color dual para obtener una $(\Delta + 1)$ -coloración de G , por lo que esta suposición tampoco es posible.

Ahora, supongamos que $c_{k+1} = c_{t+1}$, con $0 \leq t < k - 1$; entonces las aristas e_k y e_t tienen el mismo color dual. Definimos dos trayectorias, P y Q . P es de longitud máxima y tiene a v_k como uno de sus extremos, además de que sus aristas están coloreadas alternadamente por b y c_{k+1} . Por otro lado, Q es una trayectoria de longitud máxima con vértice inicial v_t y cuyas aristas están coloreadas alternadamente por los colores b y $c_{t+1} = c_{k+1}$. Supongamos que P y Q tienen respectivamente a w y w' como su otro vértice extremo. Esto nos da paso a 4 casos, dependiendo de como sean w y w' :

Caso 1. Supongamos que $w = v_m$ para algún m , $0 \leq m \leq k - 1$. En este caso, la primera y última arista de P deben tener color b . Así también, ninguna arista adyacente a v_m tiene color c_{k+1} . Observemos que a menos que $v_m = v_t$, el vértice v_t no está en P . Intercambiamos los colores b y c_{k+1} de las aristas de P . Una vez hecho esto, no hay arista incidente a v_m que tenga color b ; es más, el color dual de cada arista e_i ($i < m$) sigue siendo el mismo. Si $m = 0$, asignemos a e el color b . Si $m > 0$, entonces, como describimos anteriormente, podríamos cambiar el color de e_m a b y colorear cada arista e_i ($0 \leq i < m$) con su color dual. La coloración resultante mostraría que G es $(\Delta + 1)$ -coloreable, lo cual, de nueva cuenta, es imposible.

Caso 2. Supongamos que $w' = v_m$ para alguna m , $0 \leq m \leq k$ y $m \neq t$. En este caso la

primera y la última arista de Q tienen color b y ninguna arista incidente a v_m tiene color c_{k+1} . Es más, Q no contiene a v_k al menos que $v_m = v_k$. Intercambiamos los colores b y c_{k+1} de las aristas de Q . Si $m < t$, entonces hacemos lo mismo que en el Caso 1. Si $m > t = 0$, entonces cambiamos el color de la arista e por b . Si $m > t > 0$, entonces cambiamos el color de e_t por b y coloreamos cada e_i , $0 \leq i < t$, con su color dual. Este proceso implica que G puede ser $(\Delta + 1)$ -coloreable, lo cual no puede ser posible.

Caso 3. Supongamos ahora que $w \neq v_m$ ($0 \leq m \leq k - 1$), pero también que $w \neq u$; o que $w' \neq v_m$ para cualquier $m \neq t$ y $w' \neq u$. Tomemos el caso para w , ya que la conclusión es la misma para w' . Si intercambiamos los colores b y c_{k+1} de las aristas de P , entonces el color b no se le asigna a ninguna de las aristas incidentes a v_k y por lo tanto el color dual de cada una de las aristas e_i ($0 \leq i < k$) permanece intacto. No obstante, ya vimos que este caso nos lleva a una contradicción.

Caso 4. El caso que nos falta considerar es que $w = u$ y $w' = u$. Ya que u no es incidente a ninguna arista que tenga color b , entonces la primera arista de P y Q se colorea con este color, mientras que su última arista toma el color c_{k+1} . Si P y Q no tienen ninguna arista en común, entonces u es incidente a dos aristas distintas que tienen el color c_{k+1} , lo cual no puede pasar. Por lo tanto, las trayectorias P y Q deben tener una arista en común, lo cual significa que existe un vértice incidente a tres aristas que pertenecen a P o a Q . Necesariamente, al menos dos de estas aristas tienen color ya sea b o c_{k+1} , lo cual, nuevamente, es imposible.

Dado que hemos considerado todos los casos posibles, esto quiere decir que la suposición que hicimos al principio no es verdadera. Por lo tanto, tenemos que $\chi_1(G) \leq \Delta(G) + 1$. ■

Para terminar esta sección, un punto importante que debemos mencionar es que no importando que la demostración del Teorema de Vizing induce un algoritmo eficiente para producir una $(\Delta + 1)$ -coloración de las aristas de cada gráfica no vacía G , no existe un algoritmo que se ejecute en tiempo polinomial para decidir si $\chi_1(G)$ es $\Delta(G)$ o $\Delta(G) + 1$. De hecho, se sabe que este problema de decisión entra en la categoría de los problemas *NP-completos*³, ya que en principio, el algoritmo que lo determina, toma tiempo exponencial.

3.3. Árbol Generador de Peso Mínimo Local

En el capítulo anterior mencionamos que en los últimos años, el problema de encontrar una gráfica generadora⁴ de manera eficiente para su aplicación en ambientes que cambian dinámicamente, como lo es el de las redes ad hoc, ha sido fuente de mucha investigación. También mencionamos que dentro de este desarrollo se ha puesto particular interés

³La definición de esta clase de problemas puede consultarse en [12, 16].

⁴No en el sentido de la teoría de gráficas

en ciertos parámetros como lo son el peso de la gráfica obtenida con relación al árbol generador de peso mínimo (*minimum spanning tree*, *MST*) de la gráfica original, su grado máximo, su diámetro, etcétera, pero sobre todo en aplicar un paradigma de comunicación que permita operar a los elementos (los nodos de la red) de dicho ambiente tomando solamente en consideración la información de otros elementos que se encuentren dentro de una vecindad geográfica cercana, es decir, conseguir tales características haciendo uso de la técnica de localidad.

En particular en el ambiente de las redes inalámbricas, existen dos puntos importantes. El primero tiene que ver con llevar a cabo tareas de comunicación, como el establecimiento de rutas, de manera local y eficiente. Esto puede ser resuelto fácilmente si la gráfica que representa a nuestra red es plana, al hacer uso, por ejemplo, del algoritmo de establecimiento de rutas con base a caras (*face routing*), como lo vimos en el capítulo 2. Por otro lado, el segundo punto se refiere a la construcción de una gráfica generadora plana que sea lo más simple posible y ya sobre esta trabajar algún otro problema. De hecho, ya mencionamos este aspecto también en el capítulo 2 cuando hicimos referencia a la gráfica de Gabriel.

Además de contar con la posibilidad de obtener una subgráfica plana al determinar las aristas de Gabriel, existen otros algoritmos ([26, 27, 28]) para construir de manera distribuida y local una subgráfica que tenga grado acotado y un factor de estiramiento (*stretch factor*) constante para gráficas unitarias, que como lo mencionamos desde el capítulo 1, son con las que trabajaremos. No obstante, su construcción es relativamente complicada, el grado de un vértice es hasta cierto punto alto (mayor que veinticinco) y su peso o costo puede ser mucho más grande que el del MST.

Dado lo anterior, Chávez *et. al* ([11]) propusieron un algoritmo para construir una subgráfica generadora dada una gráfica geométrica de disco unitario con irregularidad r . Con esto nos referimos a que el rango de transmisión de cada uno de los nodos de la red es la región contenida dentro del círculo unitario alrededor de un nodo y que contiene a todos los puntos a distancia a lo más r ($0 < r \leq 1$). Describiremos este resultado en las siguientes secciones tomando $r = 1$, lo cual significa que supondremos que los nodos que compongan a nuestra red ad hoc tendrán el mismo rango de transmisión.

El algoritmo propuesto obtiene una gráfica generadora conexa de disco unitario de manera local, en el sentido de que cada nodo usa la información disponible en su vecindad, la cual, dada una k , se define como el conjunto de nodos que se pueden alcanzar por medio de trayectorias de longitud a lo más k . La gráfica resultante, además, posee cotas considerables en relación a las características que mencionamos anteriormente y que son deseables en una gráfica como esta, las cuales veremos un poco más adelante.

Una vez que hemos mencionado que el algoritmo es local, podemos dar la siguiente definición, la cual introduce un poco de la notación que se utilizará.

Definición 8. Dada una gráfica G y un vértice v de G , denotamos como $N_k(v)$ a la vecindad cerrada de distancia a lo mas k de v , esto es, $N_k(v)$ contiene a todos los vértices alcanzables desde v por una trayectoria de longitud a lo mas k . Por supuesto, v también está incluido en este conjunto. Así también, denotaremos a la gráfica inducida por $N_k(v)$ como G_v^k .

3.3.1. El Algoritmo

Antes de continuar, recordemos que una gráfica G es geométrica si se puede incrustar en el plano Euclidiano y sus aristas son segmentos de línea recta entre los nodos. Por otra parte, el algoritmo que mostraremos lleva a cabo una selección de las aristas con base en un orden lineal de estas, por lo que primeramente definiremos dicho orden.

Definición 9. Dada una gráfica geométrica G , asignamos a cada una de sus aristas, (u, v) , una quintupla: $(|uv|, x_1, y_1, x_2, y_2)$, donde x_1, y_1 y x_2, y_2 son las coordenadas de los vértices de la arista en cuestión y $x_1 > x_2$ ó $x_1 = x_2$ y $y_1 > y_2$. Notemos que esta tupla es única para cada una de las aristas de G . Entonces, el orden lineal \triangleleft para las aristas de G se define con base al orden lexicográfico de las tuplas.

Observemos que la definición anterior implica que si no se puede determinar si una arista es más grande que otra debido a que tienen la misma longitud (distancia euclidiana), entonces el siguiente discriminante es considerar las coordenadas de sus vértices. Otro punto importante es que a pesar de que una gráfica G puede tener varios árboles generadores de peso mínimo, obtenidos por un algoritmo como lo es el algoritmo de Kruskal ([12]), la utilización de este orden para la selección de las aristas tiene como consecuencia que G tenga un único MST, llamémoslo T^{\triangleleft} .

Una vez establecidos los conceptos anteriores, podemos dar la descripción del algoritmo, denominado $LocalMST_k$ para $k \geq 2$, en el listado 3.1, el cual se ejecuta en cada uno de los nodos v de la gráfica geométrica G para obtener una gráfica que denotaremos como G_k^{\triangleleft} .

Listado 3.1: Algoritmo $LocalMST_k$.

1	Determinar $N_k(v)$.
2	
3	Construir en G_v^k el MST, $T_k(v)$.
4	
5	Transmitir a $N_1(v)$ las aristas que esten en $G_v^k \cap T_k(v)$.
6	
7	Determinar G_k^{\triangleleft} : una arista (u, v) pertenece a $G_k^{\triangleleft} \Leftrightarrow$ esta se retiene tanto en u como en v .

Como podemos observar en el listado 3.1, el algoritmo se ejecuta de manera distribuida además de que el parámetro k determina la localidad del algoritmo.

3.3.2. Propiedades de la gráfica G_k^\triangleleft

En esta sección, mostraremos algunos resultados que nos permitirán establecer las propiedades con las que cuenta la gráfica G_k^\triangleleft , y que se convertirán en puntos importantes para la presentación del resultado principal de este trabajo en el capítulo 4.

Proposición 1. $T^\triangleleft \subseteq G_k^\triangleleft$.

Demostración. Supongamos que existe una arista $e = (u, v)$ que está en T^\triangleleft y que no está en G_k^\triangleleft . Si esto pasa, entonces e debió de haber sido rechazada ya sea en $T_k(u)$ o $T_k(v)$, supongamos que fue en $T_k(u)$.

Por otro lado, dado que e pertenece a T^\triangleleft , entonces la única trayectoria en él que une a u y v es precisamente e . Ahora bien, ya que la arista fue rechazada en $T_k(u)$, existe una trayectoria en este último, denotémosla como P , que une a u y v y que además hace uso solamente de aristas más pequeñas (dado el orden lineal) que e .

Tomemos una arista $a = (w, w')$ en P tal que no esté en T^\triangleleft . Sabemos que en T^\triangleleft existe una única trayectoria, Q , uniendo a w y w' y que debe usar aristas menores que a . Dado que esto es cierto para cualquier arista en P que no esté en T^\triangleleft , entonces podemos construir una trayectoria en T^\triangleleft uniendo a u con v haciendo uso solamente de aristas más pequeñas que e (como podemos ver en la figura 3.2), lo cual es una contradicción ya que habíamos dado por hecho que e estaba en T^\triangleleft .

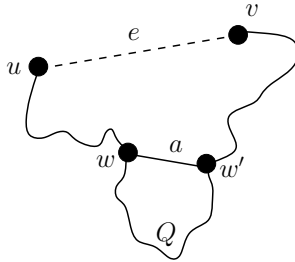


Figura 3.2: Trayectoria de u a v con aristas más pequeñas que e .

Por lo tanto, e no pudo haber sido rechazada ni en $T_k(u)$ ni en $T_k(v)$. Entonces, $T^\triangleleft \subseteq G_k^\triangleleft$. ■

Chávez *et al.* establecieron en [11] el siguiente teorema que lista todas las propiedades que posee la gráfica G_k^\triangleleft , la cual es el resultado de la ejecución del algoritmo

en el listado 3.1 sobre una gráfica geométrica G con irregularidad r y $k \geq 2$. De tales propiedades, sólo demostraremos las tres primeras, en particular considerando $r = 1$, ya que son las que nos van a interesar más adelante, por lo que los detalles restantes acerca de esta demostración se pueden consultar en el trabajo ya mencionado.

Teorema 5. *Si G es una gráfica conexa y geométrica con irregularidad r y $k \geq 2$, entonces la gráfica G_k^\triangleleft tiene las siguientes propiedades:*

- a) G_k^\triangleleft es conexa.
- b) Si la distancia entre cualquier par de nodos de la red es al menos $\sqrt{1 - r^2}$, entonces la gráfica G_k^\triangleleft es plana.
- c) $\Delta(G_k^\triangleleft) \leq \begin{cases} 5 & r = 1 \\ 3 + \frac{6}{\pi r} + \frac{r+1}{r^2} & 0 < r < 1 \end{cases}$
- d) Si G_k^\triangleleft es plana y $kr > 1$, entonces $\text{costo}(G_k^\triangleleft) \leq \frac{kr+1}{kr-1} \cdot \text{costo}(T^\triangleleft)$.

Demostración.

- a) Que G_k^\triangleleft sea conexa es consecuencia directa de la proposición 1.
- b) Supongamos que G_k^\triangleleft no es plana, por lo que existen al menos dos aristas en ella, digamos $e = (u, v)$ y $e' = (w, x)$, que se cruzan. Arbitrariamente podemos pensar en que el ángulo $\angle u w v$ es el más grande en el cuadrilátero inducido por los vértices de e y e' , $u w v x$ (ver figura 3.3), además de que es claro que este es mayor o igual que $\pi/2$.

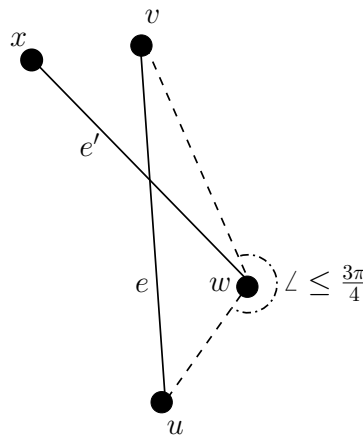


Figura 3.3: Suposición de que dos aristas se cruzan en G_k^\triangleleft .

Ya que $\|uv\| \leq 1$, entonces por la desigualdad del triángulo tenemos que $\|uw\|^2 + \|wv\|^2 \leq 1$. Esto implica que $\|uw\|^2 \leq 1 - \|wv\|^2 \leq r^2$, ya que por

hipótesis teníamos que $\|wv\| \geq \sqrt{1-r^2}$, por lo que entonces la arista (u, w) debe pertenecer a las aristas de G_k^{\triangleleft} . Análogamente, tenemos que (w, v) también pertenece a las aristas de G_k^{\triangleleft} .

Por otro lado, supongamos que u construye $T_k(u)$ por medio del algoritmo de Kruskal. Si la arista (u, w) no fuera retenida en $T_k(u)$, entonces eso significaría que en el árbol existe una trayectoria uniendo a u y w y que hace uso solamente de aristas más pequeñas. De la misma manera, esto es cierto para la arista (w, v) . Entonces, al momento de considerar a la arista e para su inclusión en $T_k(u)$, ya existe una trayectoria que conecta a u y v , lo que ocasionaría que e fuera rechazada en $T_k(u)$ para no formar un ciclo y por tanto no pertenecería a G_k^{\triangleleft} , lo cual contradice el hecho de que e pertenezca a G_k^{\triangleleft} .

Por lo tanto, no existe ningún cruce en G_k^{\triangleleft} , es decir, la gráfica que resulta de aplicar el algoritmo LocalMST $_k$ es plana.

- c) Para demostrar que G_k^{\triangleleft} tiene grado acotado, empezemos por tomar un vértice u de la gráfica G_k^{\triangleleft} . Ahora, dividamos el círculo unitario con centro en u en seis regiones iguales, es decir, cada una de ellas de arco igual a $\pi/3$.

Dado que G es finita, podemos suponer que los límites que definen estas regiones no pasan a través de ningún vecino de u , es decir, que ninguna de las aristas que tienen a u como uno de sus extremos recae sobre tales límites. Entonces, para cualquier par de vecinos de u , digamos v y w , dentro de un mismo sector, el ángulo $\angle wuv$ es menor que $\pi/3$. Además, tenemos que $\|vw\| < \max\{\|uw\|, \|uv\|\}$. Lo anterior lo podemos ver más claramente en la figura 3.4.

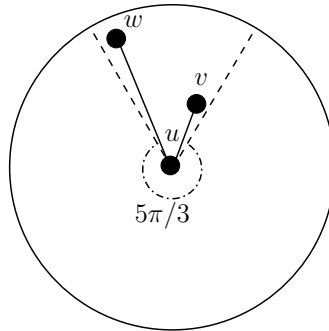


Figura 3.4: División del círculo unitario con centro en u en regiones.

Ahora bien, si $\|vw\| \leq r$, entonces ya sea (u, v) ó (u, w) habría sido remplazada en G_k^{\triangleleft} por (v, w) , por lo que podemos concluir que $\|vw\| > r$. Si suponemos que $r = 1$, entonces sucedería que u podría tener a lo mas un vecino por cada uno de los

sectores que definimos, es decir, u sólo tendría a lo mas seis nodos como vecinos en G_k^\triangleleft .

Supongamos entonces que u tiene exactamente seis vecinos. Tomando en cuenta los puntos anteriores, esto sólo pasaría si tales vecinos se encontraran en las intersecciones de los lados (de longitud uno) de un hexágono perfecto y u estuviera en el centro de dicho polígono. Sin embargo, si se diera este caso, sólo dos de las aristas incidentes en u se retendrían, ya que las otras 4 serían eliminadas, en particular, al construir $T_k(u)$, con lo que queda demostrado que el grado máximo de un vértice de G_k^\triangleleft es a lo mas cinco.

Lo anterior lo podemos ver en la figura 3.5, en donde en la subfigura 3.5a están numeradas las aristas de menor a mayor de acuerdo al orden lineal que se definió y en la subfigura 3.5b se muestra el árbol $T_k(u)$.

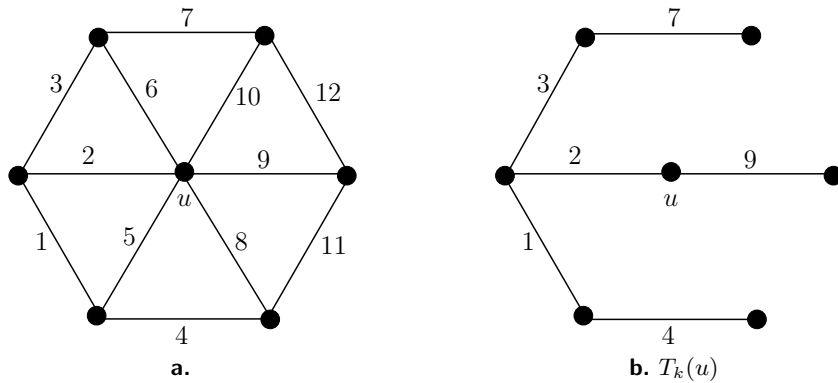


Figura 3.5: Un vértice u en G_k^\triangleleft tiene grado menor a seis.

Sólo falta comentar que en realidad, la cota de que el grado máximo de un vértice u sea cinco es la cota más pequeña que podemos obtener, es decir, la cota es justa. Para mostrarlo, consideremos la configuración de los vértices de la figura 3.6, en donde los vecinos de u se encuentran sobre la circunferencia de radio uno con él como su centro, además de que estos están igualmente distribuidos, es decir, forman un pentágono regular. En este caso, todas las aristas necesitarán ser retenidas para garantizar lo conexidad.

■

Es importante recalcar que el orden lineal antes definido para ordenar las aristas de una gráfica geométrica juega un papel importante para lograr este resultado, ya que de no haberlo hecho, el algoritmo bien podría producir una gráfica disconexa. En particular, esto lo podemos observar en la figura 3.5a, donde el orden impuesto nos permitió discernir sin

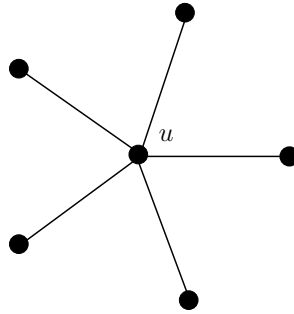


Figura 3.6: El grado de un vértice u en G_k^{\triangleleft} puede ser cinco.

problema qué arista era más grande que otra de las que formaban parte de dos triángulos equiláteros adyacentes, eliminando toda posibilidad de que cualquiera de las aristas que forman $T_k(u)$ no perteneciera a alguno de los otros árboles $T_k(v)$, con v cualquier vecino de u .

Con esto damos por terminado este capítulo, mencionando solamente que en el siguiente se utilizarán los resultados que aquí se describieron para plantear un nuevo algoritmo de tipo local.

Coloración de aristas localmente

4

4.1. Motivación

Debido a que el rápido crecimiento del área de las redes inalámbricas ha traído consigo una gran variedad de servicios móviles, se puede decir que es prácticamente imposible que en un futuro cercano tales servicios no se vean involucrados en nuestra vida cotidiana. Sin embargo, algo que también es cierto es que estos servicios hacen uso de un espectro de comunicación para llevar a cabo sus objetivos y dado que este es finito, se requiere de mecanismos eficientes que permitan hacer una asignación de los canales o frecuencias disponibles de manera que se puedan llevar a cabo las comunicaciones entre los nodos que forman parte de la red, pero que a su vez también permita hacer un reuso de los recursos.

A la problemática de encontrar un mecanismo como el que se describió anteriormente se le conoce como el *problema de asignación de frecuencias* y es precisamente para este que se planteó una solución (un algoritmo de tipo local) y que representa el resultado principal de nuestro trabajo.

Una vez que hemos establecido a lo largo de los tres capítulos anteriores los conceptos que servirán de base para describir nuestro resultado, podemos pasar entonces a plantear de manera más amplia el problema que dicho resultado trata, así como de la solución que se propone.

4.2. Un algoritmo local para la asignación de frecuencias

Recordemos que dada una red ad hoc, queremos encontrar una asignación de frecuencias de manera que evitemos problemas como el de la interferencia pero que además, tratemos de reusar los recursos lo más posible. Sabemos también que podemos representar dicha red por medio de una gráfica de disco unitario y es precisamente esta con la que trabajaremos para dar solución a nuestro problema.

4.2.1. Mapeo del problema de asignación de frecuencias a la coloración de aristas

Dados dos nodos, digamos u y v , de la gráfica G que representa a nuestra red, lo que queremos es asignar una frecuencia al canal que utilizan u y v para comunicarse. Este canal, en la gráfica, juega el papel de una de sus aristas y entonces requerimos que aristas adyacentes tengan asignadas diferentes frecuencias.

Inmediatamente, podemos pensar en que la asignación de frecuencias distintas para aristas que inciden en un mismo vértice se puede ver como la obtención de una coloración propia de las aristas de la gráfica que representa a la red ad hoc, es decir, ahora nuestro problema se convierte en un problema de coloración de aristas en el área de la teoría de gráficas de manera que obtengamos el menor índice cromático posible.

Como ya lo mencionamos en el capítulo 3, el índice cromático de una gráfica es igual a $\Delta(G)$ ó $\Delta(G) + 1$, donde $\Delta(G)$ representa el grado máximo de los vértices de la gráfica G . No obstante, también establecimos que el problema de decidir cuál de estos valores era el mínimo para una gráfica está catalogado como un problema *NP-completo*, lo cual quiere decir que no existe un algoritmo que determine la respuesta y cuyo tiempo sea polinomial.

4.2.2. Coloración de las aristas de una gráfica de manera local

Teniendo en cuenta el mapeo descrito en la subsección anterior, nos encontramos con el primer problema: tener la posibilidad de dar una coloración de las aristas de la gráfica G de manera que tratemos de optimizar el número de colores, es decir, tratar de ocupar la menor cantidad de colores en comparación con la cota impuesta por el teorema de Vizing.

Claramente, la primera opción es ocupar el algoritmo implícito en la demostración del teorema de Vizing, mostrada en el capítulo 3, el cual, como también lo mencionamos, genera una $(\Delta(G) + 1)$ -coloración, con lo que obtendríamos una excelente aproximación, si no es que la solución óptima. No obstante, la aplicación de este algoritmo asume el conocer completamente la topología de la gráfica sobre la que se actúa por lo que esto la descarta como una solución apropiada dado el contexto en el que estamos trabajando, el de las redes ad hoc, ya que para obtener esta información se necesitaría que la red se mantuviera estática por un tiempo considerable, lo cual obviamente no es posible, para que los datos recopilados se consideraran confiables, además de utilizar muchos de los recursos con los que cuenta cada uno de los nodos de una red de este tipo.

Dado lo anterior, nuestra propuesta para dar solución al problema de coloración consiste de un algoritmo de tipo local, el cual se basa en dos de los resultados vistos previamente: el teorema de Vizing y el uso de LocalMST_k .

El resultado principal de este trabajo se enuncia en el siguiente teorema, el cual quedará demostrado una vez que describamos cada uno de los pasos de los que consta nuestro algoritmo local:

Teorema 6. *Sea U una red de disco unitario. Se puede hacer uso de un algoritmo local para obtener una subgráfica plana de la gráfica G que representa a U , denominémosla \mathfrak{F} , de tal manera que esta tenga grado máximo cinco; además, si G es conexa, entonces \mathfrak{F} también lo será. De igual manera, podemos colorear las aristas de \mathfrak{F} con doce colores por medio de un algoritmo local de tal forma que para cada v de U , este haga uso solamente de $N_4(v)$.*

Dada la gráfica G que representa a una red de disco unitario U , obtenemos en primer lugar la subgráfica generada al aplicar el algoritmo LocalMST_2 , llamémosla \mathfrak{F} , con lo cual, en principio, garantizamos que no existen cruces entre sus aristas, es decir, \mathfrak{F} es plana, además de que también es conexa. Para ejemplificar, supongamos que la gráfica \mathfrak{F} es la que se encuentra en la figura 4.1.

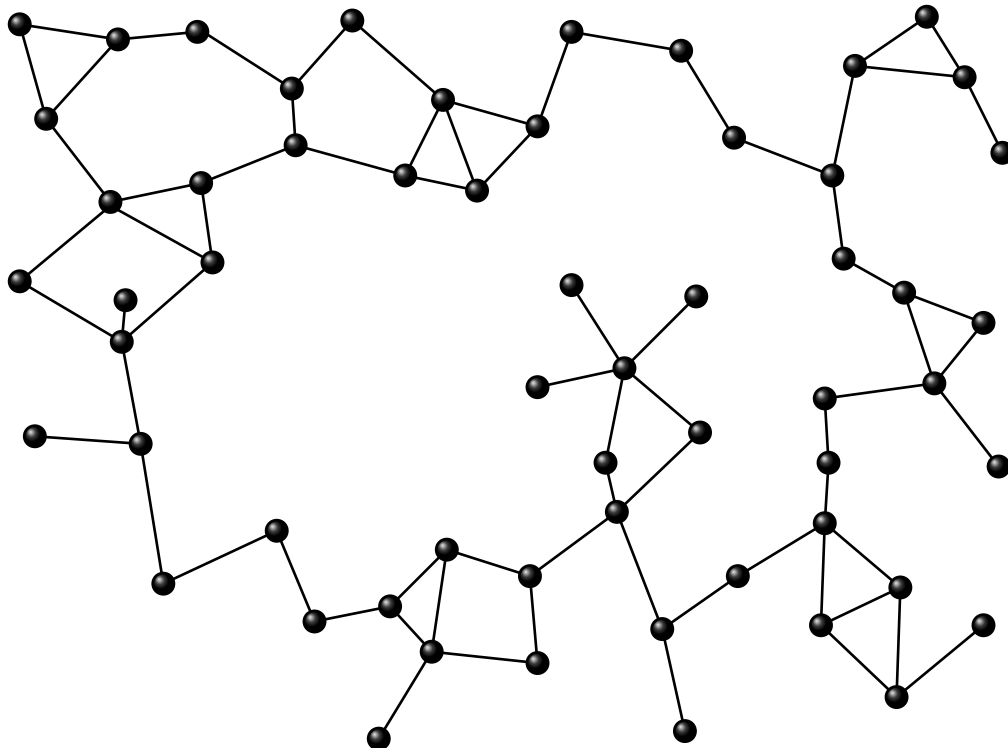


Figura 4.1: La gráfica, \mathfrak{F} , que es el resultado de aplicar LocalMST_2 sobre G .

Recordemos que el objetivo de colorear las aristas de la gráfica que representa a una red ad hoc es que cualesquiera dos vértices de ella puedan comunicarse haciendo uso

del menor número de frecuencias posibles. Entonces, al utilizar alguno de los algoritmos descritos en el capítulo 2 sobre la gráfica \mathfrak{F} , podemos establecer una ruta entre cualquier par de nodos, por lo que si coloreamos las aristas de \mathfrak{F} estaríamos resolviendo el problema.

Mencionamos en el capítulo 3 que una de las propiedades que adquiere la gráfica \mathfrak{F} por como se construyó es que el grado máximo de sus vértices es a lo más cinco, así, por el teorema de Vizing, sabemos que sus aristas pueden ser coloreadas propiamente con a lo mas seis colores.

Para ocupar el resultado anterior en la gráfica \mathfrak{F} haremos lo siguiente: podemos pensar en que \mathfrak{F} está incrustada en el plano y dividimos este en regiones cuadrangulares de tamaño 2×2 , es decir, podemos ver al plano como un conjunto de cuadrados C_{ij} tal que las posiciones de cada uno de sus vértices se encuentran en los puntos $(2i, 2j)$, $(2(i+1), 2j)$, $(2i, 2(j+1))$ y $(2(i+1), 2(j+1))$ respectivamente. Así también, cada nodo u , cuya posición es (x, y) en U , tiene asociado un C_{ij} si es que $2i \leq x < 2(i+1)$ y $2j \leq y < 2(j+1)$. Lo anterior lo podemos ver en el figura 4.2.

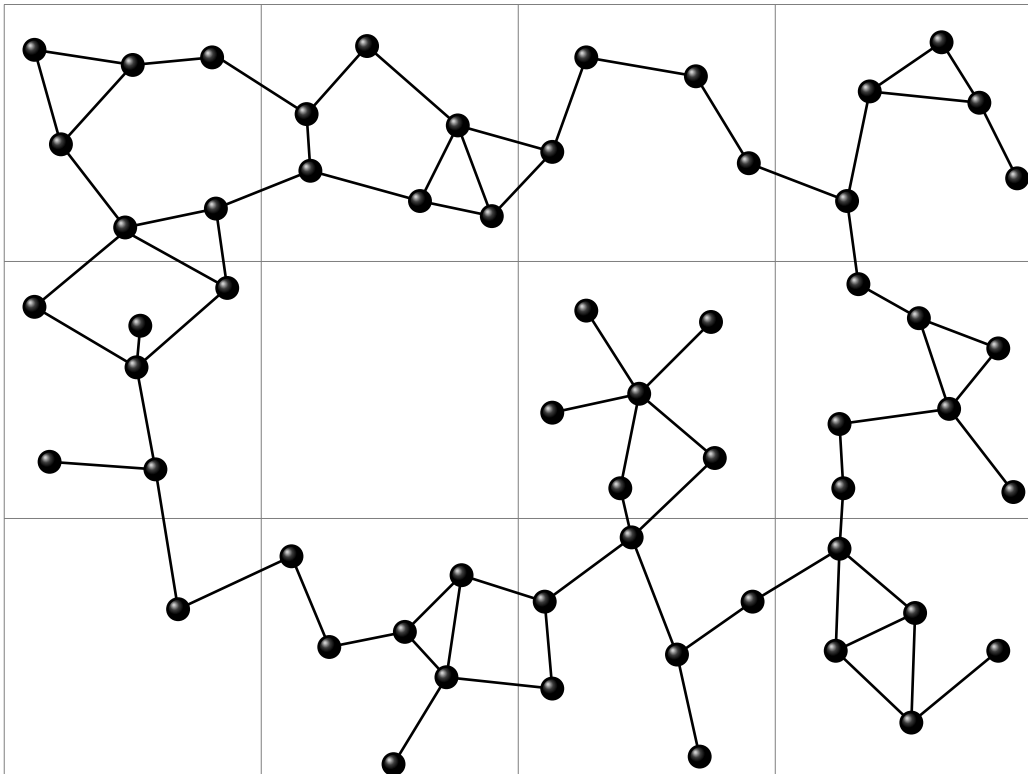


Figura 4.2: División de la región en que se encuentra \mathfrak{F} .

Dada la división anterior, podemos definir la gráfica \mathfrak{F}_{ij} como la subgráfica inducida por los vértices de G que se encuentran dentro del cuadrado C_{ij} . Hay que hacer notar

que dicha gráfica no necesariamente es conexa, ya que al aislarla del resto de \mathfrak{F} pudimos haber eliminado la única o todas las trayectoria entre un par de vértices que pertenezcan al mismo C_{ij} . Así también, hay que remarcar que dados dos vértices cualesquiera de una misma componente conexa del cuadrado en cuestión, estos se encuentran a distancia a lo más cuatro (ver figura 4.3). Lo anterior nos permite asegurar que para que un vértice cualquiera de \mathfrak{F} , digamos u , determine la componente conexa a la que pertenece en \mathfrak{F}_{ij} es necesario que este obtenga las coordenadas de los vértices en $N_4(u)$, claro, eliminando aquellos que no estén en \mathfrak{F}_{ij} .

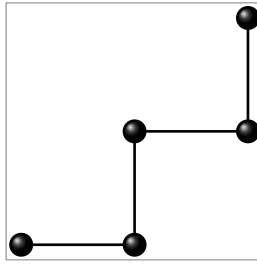


Figura 4.3: Caso en el que la distancia entre dos vértices en \mathfrak{F}_{ij} es cuatro.

Entonces, en cada C_{ij} , podemos aplicar localmente el resultado que nos proporciona el teorema de Vizing, por lo que todas las gráficas \mathfrak{F}_{ij} se colorearían propiamente con los colores del conjunto $\{1, \dots, 6\}$. Retomando la gráfica de la figura 4.2, la coloración resultante al aplicar este paso de nuestro algoritmo se puede observar en la figura 4.4.

Una vez que hemos coloreado todas las aristas de las gráficas \mathfrak{F}_{ij} , sólo nos falta colorear las aristas que cruzan los cuadrados C_{ij} . Para establecer la manera en que lo haremos, ocuparemos el siguiente resultado:

Proposición 2. *Sea u un vértice de \mathfrak{F} que pertenece a un cuadrado C_{ij} . Entonces u es adyacente a lo más a tres vértices que se encuentran por encima de la recta horizontal $y = 2j + 2$*

Demostración. La demostración a este lema se sigue inmediatamente del hecho de que la gráfica \mathfrak{F} es el resultado de aplicar el algoritmo LMST_2 en la gráfica G , la cual, como vimos en el capítulo 3, tiene la propiedad de que el ángulo formado entre cualesquiera dos de sus aristas es mayor que $\pi/3$, por lo que solamente puede haber tres vértices por encima de la recta horizontal en cuestión. ■

Ahora bien, tomemos un vértice u en alguno de los C_{ij} que sea adyacente a otro vértice w que se encuentre en otro C_{kl} cuya base sea la recta $\mathcal{L} \equiv y = 2j + 2$. Con base en la proposición anterior, colorearemos la arista $e = (u, w)$ de la siguiente manera:

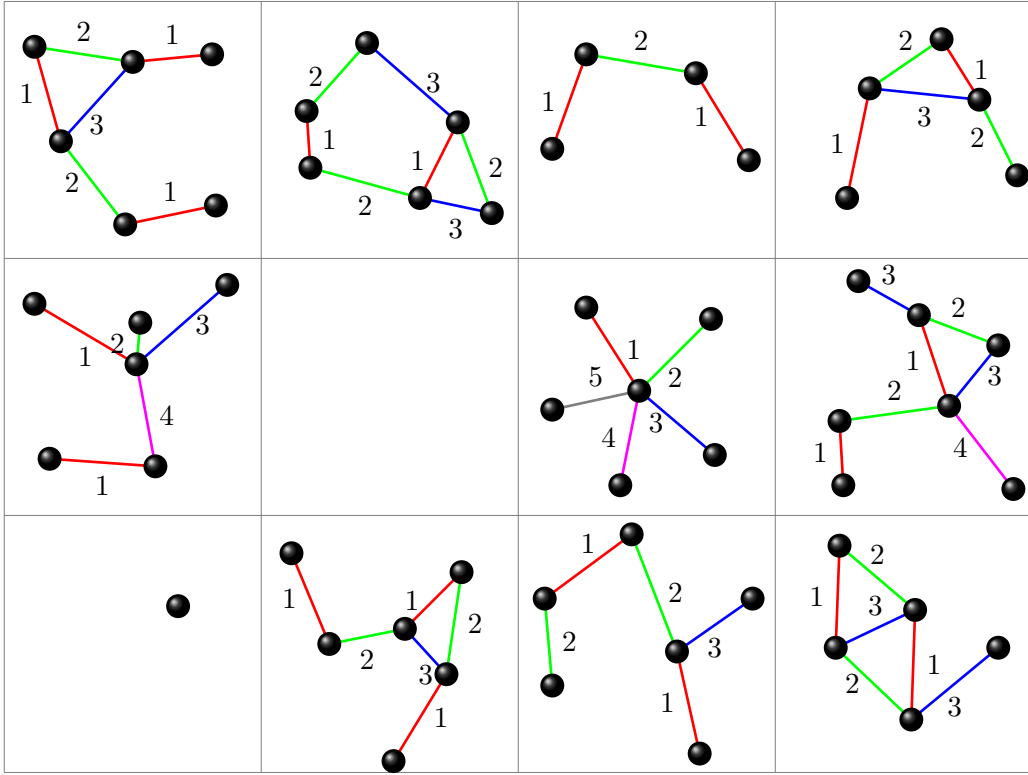


Figura 4.4: Coloración de las aristas de las gráficas \mathfrak{F}_{ij} localmente.

- Si el ángulo formado entre la arista e y la recta \mathcal{L} , en sentido contrario a las manecillas del reloj, es igual o menor a $\pi/3$, asignamos a e el color 7.
- Si el ángulo entre e y \mathcal{L} es mayor a $\pi/3$ pero menor igual a $2\pi/3$, coloreamos la arista con el color 8.
- En caso contrario, es decir, que la arista forme un ángulo \hat{a} con respecto a la recta \mathcal{L} tal que $2\pi/3 < \hat{a} \leq \pi$, la arista tomará el color 9.

La forma anterior de asignar los colores $\{7, 8, 9\}$ puede verse en la figura 4.5. Además, de manera análoga, podemos hacer una asignación a las aristas que crucen la recta horizontal $y = 2j$ haciendo uso de los mismos colores.

Algo que hay que aclarar acerca de esta asignación de colores es que dos aristas que son adyacentes y que cruzan alguna de las rectas horizontales no pueden colorearse del mismo color, por la misma razón de que entre ellas existe un ángulo mayor a los $\pi/3$.

Así también, si ahora consideramos las aristas que crucen alguna de las rectas verticales $x = 2i$ o $x = 2i + 2$, podemos obtener una coloración para ellas utilizando otros tres colores: $\{10, 11, 12\}$, obteniendo una configuración como la que se muestra en

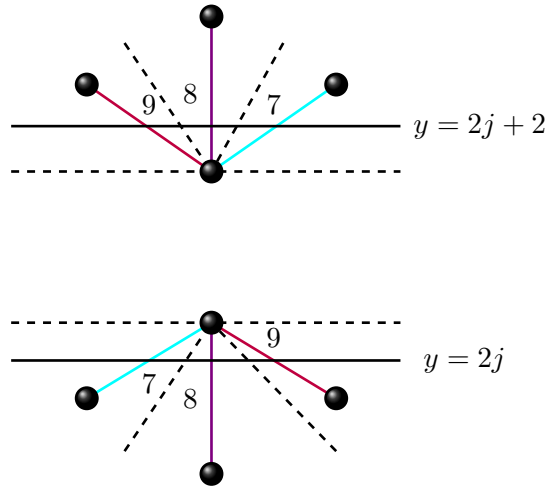


Figura 4.5: Coloración de una arista que cruzan las rectas $y = 2j + 2$ o $y = 2j$.

la figura 4.6.

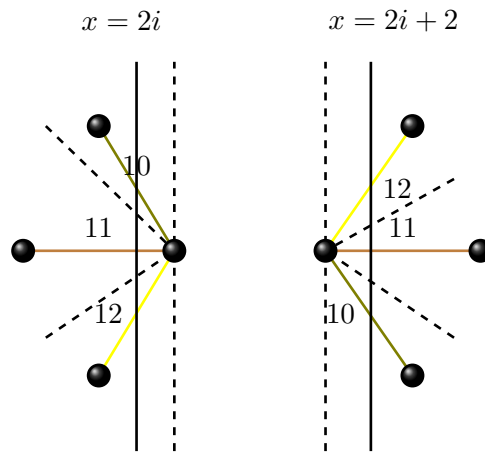


Figura 4.6: Coloración de una arista que cruzan las rectas $x = 2i$ o $x = 2i + 2$.

Con lo anterior, garantizamos que todas las aristas de la gráfica \mathfrak{F} quedan propiamente coloreadas, con un total de 12 colores, lo cual demuestra el resultado planteado por el teorema 6.

Aplicando este último paso del algoritmo local para colorear las aristas que faltan en la figura 4.4, podemos obtener una coloración propia de las aristas de \mathfrak{F} como se muestra en la figura 4.7.

Con esto damos por terminada la descripción de la solución dada al problema principal

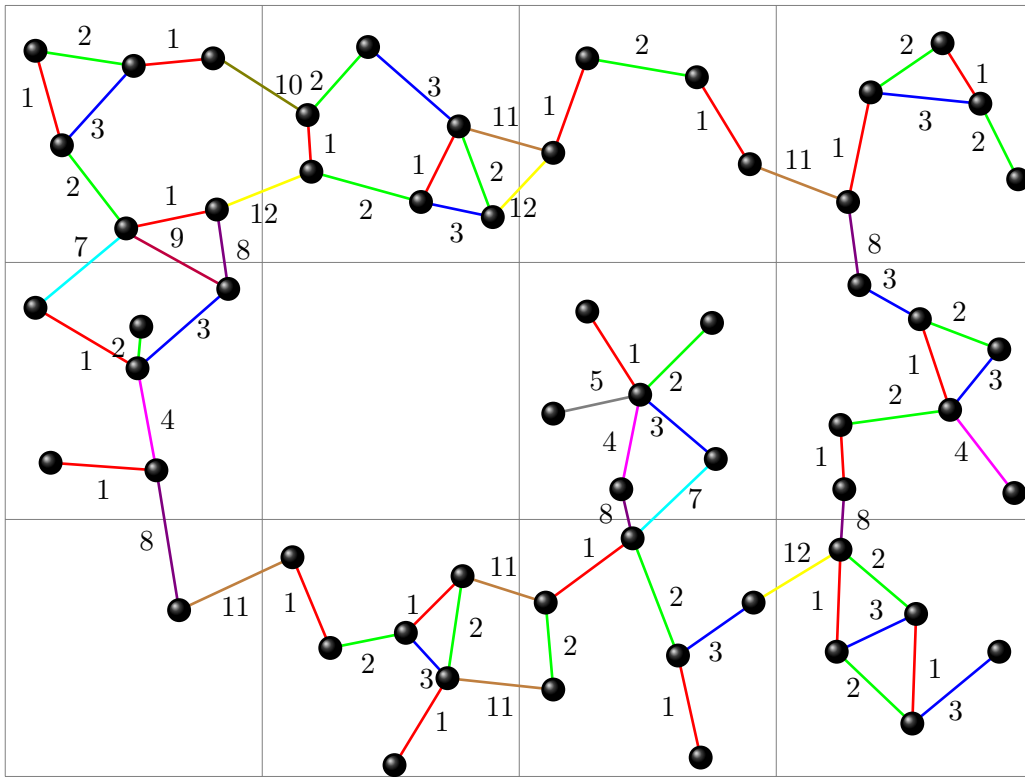


Figura 4.7: Coloración propia de las aristas de la gráfica \mathfrak{F} localmente.

del que consta este trabajo, con la cual se pretendía asignar el menor número de frecuencias posibles, utilizando la técnica de localidad, para poder llevar a cabo la transmisión de paquetes entre los nodos de una red ad hoc sin que esto ocasione algún problema.

Comentarios Finales



A lo largo de la descripción de este trabajo se planteo como objetivo principal el desarrollar un nuevo algoritmo que permitiera dar solución al problema de la asignación de frecuencias, el cual se presenta comúnmente en el ambiente que se impone al hacer uso de una tecnología de comunicación inalámbrica, especialmente al tener una red ad hoc.

La importancia de obtener dicho resultado se da principalmente por las características que una red de comunicación móvil posee, en particular por la forma en que se comportan los nodos que la componen: de una manera dinámica. La posibilidad de que los nodos se muevan dentro de una cierta área, además de poder incorporarse y retirarse como elementos de la red, obligan a que la investigación encaminada en esta área considere distintos paradigmas a los que se hace uso tradicionalmente en las redes alámbricas existentes.

Por otro lado, el obtener un resultado como el que se especificó en el capítulo 4, haciendo uso de otros descritos en los capítulos anteriores a este, resuelve de buena manera el problema en cuestión considerando la metodología utilizada: *la técnica de localidad*. Así también, este trabajo continúa o aporta nuevos elementos de investigación en el área de las redes ad hoc móviles, los cuales pueden ser resumidos básicamente por los siguientes puntos:

- La investigación lograda al resolver problemas, distintos al de este trabajo, utilizando dicha metodología, ha impulsado que esta sea tomada en cuenta para tratar de obtener resultados similares e inclusive mejores a los ya obtenidos, tanto en aquéllos que se basan en esta técnica como con los que lo hacen de manera distribuida, por ejemplo.
- No existen resultados previos que traten o resuelvan el problema de la asignación de frecuencias y cuya forma de proceder sea el aplicar la técnica de localidad.

Hablando un poco acerca de los elementos con los que se trabajó, se puede apreciar que en general, los resultados se encuentran dentro del área de la computación teórica; sin embargo, para lograrlos, se incorporaron argumentos de otras ramas matemáticas como la

Geometría y la Teoría de las Gráficas, lo cual demuestra nuevamente que las Ciencias de la Computación tienen fuertes lazos con otras ciencias.

Trabajo a futuro

El resultado obtenido en este trabajo provee un algoritmo para que, con doce frecuencias, una red ad hoc móvil pueda llevar a cabo la función de comunicación entre los nodos que la componen sin mayor problema. Así también, hay que resaltar que esta meta se logró con base a la obtención de una subgráfica con ciertas características, las cuales permitían incorporar intuitiva y fácilmente otros resultados ya conocidos en la teoría de gráficas (el teorema de Vizing).

No obstante, un buen planteamiento del trabajo a futuro que pudiera darse con respecto al problema de la asignación de frecuencias de manera local podría encaminarse tomando en cuenta alguno de los siguientes aspectos:

- Construir una gráfica generadora diferente a la que obtiene el algoritmo $LMST_k$.
- Considerar una malla distinta a la que se considera en este trabajo (cuadrada).
- Cambiar la manera en que aplicamos la idea de localidad para resolver el problema.

Con respecto al primer punto, una posibilidad que podría explorarse es desarrollar un nuevo algoritmo que permita obtener una subgráfica generadora de la gráfica que represente a la red y que tenga características similares a las que posee la que se obtiene al aplicar el algoritmo $LMST_k$, poniendo énfasis particularmente en el grado máximo de esta nueva gráfica. Si se lograra un grado máximo menor a cinco, esto impactaría directamente en la disminución de al menos un color, por ejemplo.

Ahora bien, el cambio de malla también podría dar un giro distinto a la manera en la que se lleva a cabo el análisis para la asignación de los colores de las aristas que cruzan distintas celdas. Sin embargo, el hacer este cambio no garantiza que se haga uso de una cantidad menor de colores a los que se utilizan con respecto a la malla cuadrangular, por lo que también sería necesario analizar si el hacer esta modificación permite una optimización.

La otra posibilidad sería cambiar el enfoque con que se aplica la técnica de localidad, es decir, tal vez no seguir el camino en el que primero encontramos una subgráfica de la gráfica original y posteriormente coloreábamos localmente con el algoritmo que nos proporciona Vizing. En lugar de todo esto, podríamos proceder de una manera totalmente distinta al considerar algún otro resultado, nuevo o ya existente, adaptándolo, de ser necesario, a nuestra idea de localidad con el objetivo de lograr acercarnos lo más posible a la cota que Vizing demostró con respecto a que una gráfica tiene índice cromático a lo

más $\Delta + 1$.

Para terminar, sólo mencionaremos que cualquiera de estas posibilidades sería una aportación interesante e importante al área de los algoritmos locales, ya que como lo mencionamos anteriormente, no existe resultado previo que considere el resolver este problema basándose en esta metodología. Además, esto también permite destacar la relevancia que la investigación en este campo de la computación teórica ha alcanzado como resultado de la evolución de la tecnología.

Referencias



- [1] K. Alzoubi, X.-Y. Li, Y. Wang, P.-J. Wan, and O. Frieder. Geometric spanners for wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(5), May 2003.
- [2] K. M. Alzoubi, P.-J. Wan, and O. Frieder. Message-optimal connected dominating sets in mobile ad hoc networks. In *MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 157–164, New York, NY, USA, 2002. ACM Press.
- [3] S. Basagni. Distributed clustering for ad hoc networks. In *Fourth International Symposium on Parallel Architectures, Algorithms and Networks*, pages 310–315, June 1999.
- [4] J. Bondy and U. Murty. *Graph Theory with Applications*.
- [5] P. Bose and P. Morin. Online routing in triangulations. In *ISAAC '99: Proceedings of the 10th International Symposium on Algorithms and Computation*, pages 113–122, London, UK, 1999. Springer-Verlag.
- [6] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7(6):609–616, 2001.
- [7] J. Broch, D. Johnson, and D. Maltz. The dynamic source routing protocol for mobile ad hoc networks. *Internet Draft*, 1998.
- [8] M. Cardei, X. Cheng, and D. Du. Connected domination in multihop ad hoc wireless networks. In *Proceedings of the 6th Int. Conference on Computer Science and Informatics*, 2002.
- [9] S. Chakrabarti and A. Mishra. Qos issues in ad hoc wireless networks. *IEEE Communications Magazine*, February 2001.
- [10] G. Chartrand and O. R. Oellermann. *Applied and Algorithmic Graph Theory*. McGraw-Hill, 1993.

-
- [11] E. Chávez, S. Dobrev, E. Kranakis, J. Opatrny, L. Stacho, and J. Urrutia. Local construction of planar spanners in unit disk graphs with irregular transmission ranges. *Lectures Notes in Computer Science*, pages 286–297, 2006.
- [12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. McGraw-Hill, 2nd edition, 2001.
- [13] M. Frodigh, P. Johansson, and P. Larsson. Wireless ad hoc networking - the art of networking without a network. *Ericsson Review*, (8):248–263, 2000.
- [14] K. Gabriel and R. Sokal. A new statistical approach to geographic variation analysis. *Systemic Zoology*, 18:259–278, 1972.
- [15] S. Gandham, M. Dawande, and R. Prakash. Link scheduling in sensor networks: Distributed edge coloring revisited. In *INFOCOM 2005. 24Th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2492–2501, March 2005.
- [16] M. Garey and D. Johnson. *Computers and Intractability*. W.H. Freeman, 1979.
- [17] D. A. Grable and A. Panconesi. Nearly optimal distributed edge colouring in $o(\log \log n)$ rounds. In *SODA '97: Proceedings of the eighth annual ACM-SIAM symposium on Discrete algorithms*, pages 278–285, Philadelphia, PA, USA, 1997. Society for Industrial and Applied Mathematics.
- [18] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, volume 2, pages 3005–3014, January 2000.
- [19] I. Hoyer. The np-completeness of edge-colouring. *SIAM J. COMPUT*, 10(4):718–720, November 1981.
- [20] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.
- [21] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Proc. 11 th Canadian Conference on Computational Geometry*, pages 51–54, Vancouver, August 1999.
- [22] F. Kuhn. *The Price of Locality: Exploring the Complexity of Distributed Coordination Primitives*. PhD thesis, Swiss Federal Institute of Technology Zurich, 2005.
- [23] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Initializing newly deployed ad hoc and sensor networks. In *Proceedings of the 10th Annual Int Conference on Mobile Computing and Networking*, pages 260–274, 2004.

- [24] F. Kuhn, R. Wattenhofer, and A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing. In *DIALM '02: Proceedings of the 6th international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 24–33, New York, NY, USA, 2002. ACM Press.
- [25] X. Li, G. Calinescu, P. Wan, and Y. Wang. Localized delaunay triangulation with application in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(10):1035–1047, October 2003.
- [26] X. Li and Y. Wang. Efficient construction of low weight bounded degree planar spanner. In *COCOON: Annual International Conference on Computing and Combinatorics*, 2003.
- [27] X.-Y. Li, G. Calinescu, and P.-J. Wan. Distributed construction of a planar spanner and routing for ad hoc wireless networks. In *IEEE INFOCOM 2002*, volume 3, pages 1268–1277, 2002.
- [28] X.-Y. Li and Y. Wang. Localized construction of bounded degree and planar spanner for wireless ad hoc networks. In *DIALM-POMC '03: Proceedings of the 2003 joint workshop on foundations of mobile computing*, pages 59–68, New York, NY, USA, 2003. ACM Press.
- [29] X. Lin and I. Stojmenovic. Gps based distributed routing algorithms for wireless networks, 2000.
- [30] J. Macker and M. Corson. Mobile ad hoc networking and the ietf. *ACM Mobile Computing and Communication Review*, 2(1):9–14, January 1998.
- [31] C. E. Perkins. *Ad Hoc Networking*. Addison-Wesley Professional, first edition, 2000.
- [32] E. Royer and C. Toh. A review of current routing protocols for ad hoc mobile networks. *IEEE Personal Communications*, 6(2):46–55, April 1999.
- [33] E. Royer and C.-K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 6(2):46–55, April 1999.
- [34] S. Murthy and J. García-Luna-Aceves. An efficient routing protocol for wireless networks. *ACM Mobile Networks and Applications Journal, Special issue on Routing in Mobile Communication Networks*, 1(2):183–197, June 1996.
- [35] I. Stojmenovic. Position-based routing in ad hoc networks. *IEEE Communications Magazine*, 40(7):128–134, July 2002.
- [36] G. Toussaint. The relative neighbourhood graph of a finite planar set, 1980.
- [37] J. Urrutia. Local solutions for global problems. 2006. To appear.
- [38] J. Wun and I. Stojmenovic. Ad hoc networks. Technical report, IEEE, February 2004.