



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

POSGRADO EN CIENCIA E INGENIERÍA  
DE LA COMPUTACIÓN

“OBTENCIÓN LOCAL DE CONJUNTOS DOMINANTES  
EN REDES AD HOC MÓVILES”

T E S I S

QUE PARA OBTENER EL GRADO DE:

MAESTRA EN CIENCIAS  
(COMPUTACIÓN)

P R E S E N T A

DOLORES LARA CUEVAS

DIRECTOR DE TESIS: DR. JORGE URRUTIA GALICIA.

México, D.F.

Octubre de 2006.



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**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.

*“No hay más arbitrio, por lo visto,  
que seguir la lucha con lo que tenemos,  
con lo que podemos y hasta donde podamos.  
Éste es nuestro deber;  
el tiempo y la constancia nos ayudarán.  
Adelante y no hay que desmayar.”*  
Benito Juárez.

*A mi familia,  
con amor.*

# Agradecimientos.

Hace ya algún tiempo que supe lo que quería hacer en mi vida profesional, sin embargo hasta hace muy poco no tenía la menor idea de como hacerlo. Afortunadamente el Doctor Jorge Urrutia se ha tomado la molestia de enseñarme mucho sobre como hacer investigación. Por dedicarme tanto de su tiempo y de su esfuerzo, y por enseñarme mucho más de lo que he podido aprender, gracias.

Gracias a los investigadores Dr. Francisco Hernández, Dr. Javier Gómez, Dr. Julio Solano y Dr. Sergio Ranjsbaum por el tiempo dedicado a la revisión de este trabajo y por sus enriquecedoras aportaciones.

Agradezco al Consejo Nacional de Ciencia y Tecnología por brindarme una beca económica, apoyo sin el cual no hubiera nunca logrado terminar este trabajo.

Por aceptarme dentro del Posgrado en Ciencia e Ingeniería de la Computación, y por brindarme todas las facilidades, el conocimiento y la infraestructura necesarios para obtener la Maestría, agradezco al Instituto de Matemáticas Aplicadas y de Sistemas. Especialmente a Lulú, Diana y Amalia, por toda la ayuda que me han brindado.

Gracias a todos mis compañeros, alumnos también del Dr. Urrutia, por todas las valiosas e enriquecedoras discusiones y aportaciones que me hicieron, pero sobre todo a Crevel y a Oscar, porque de esas horas de discusión frente al pizarrón surgió gran parte de este trabajo.

A papá, mamá y a Rosi, porque han entendido y perdonado mi ausencia durante todo este tiempo, y me han regalado su valiosísimo apoyo y su gran amor, gracias infinitamente.

Miller, gracias por estar siempre dispuesto a escuchar mis cavilaciones, y por cavilar conmigo.

# Índice general

<b>Introducción</b>	<b>VII</b>
<b>1. Redes ad hoc móviles</b>	<b>1</b>
1.1. Introducción . . . . .	1
1.2. Redes ad hoc . . . . .	2
1.2.1. Panorama general . . . . .	2
1.2.2. Características . . . . .	4
1.2.3. Ejemplos de redes ad hoc . . . . .	6
1.2.4. Líneas de investigación . . . . .	7
<b>2. Antecedentes</b>	<b>15</b>
2.1. Gráficas y subgráficas . . . . .	15
2.2. Gráficas conexas y desconexas . . . . .	17
2.3. Tres tipos importantes de gráficas . . . . .	19
2.4. Conjunto independiente y conjunto dominante . . . . .	21
2.5. Otros conceptos . . . . .	23
<b>3. Algoritmos locales</b>	<b>27</b>
3.1. Búsqueda de rutas con brújula . . . . .	28
3.2. Búsqueda de rutas con brújula mejorada . . . . .	33
3.2.1. Extracción de una subgráfica plana conexa . . . . .	33
3.2.2. Búsqueda de rutas en gráficas aplanables . . . . .	35
<b>4. Conjuntos dominantes</b>	<b>38</b>
4.1. Trabajo relacionado . . . . .	38
4.1.1. Algoritmo glotón . . . . .	39
4.1.2. Un algoritmo no local . . . . .	39
4.2. El problema . . . . .	40
4.3. Subdivisión del plano. . . . .	41
4.3.1. Propiedades de la subdivisión hexagonal . . . . .	43
4.4. Conjuntos dominantes locales . . . . .	44
4.4.1. Conjuntos mínimos por regiones . . . . .	45
4.4.2. Tamaño del conjunto dominante . . . . .	48
4.5. Simulaciones . . . . .	52

---

4.5.1. Resultados . . . . .	54
<b>5. Conclusiones</b>	<b>63</b>
<b>A. Redes simuladas</b>	<b>65</b>
A.1. Redes de tamaño cien . . . . .	66
A.1.1. Grid de tamaño cinco . . . . .	66
A.1.2. Grid de tamaño seis . . . . .	77
A.1.3. Grid de tamaño siete . . . . .	88
A.2. Redes de tamaño doscientos . . . . .	99
A.2.1. Grid de tamaño cinco . . . . .	99
A.2.2. Grid de tamaño seis . . . . .	109
A.2.3. Grid de tamaño siete . . . . .	119
A.3. Redes de tamaño quinientos . . . . .	129
A.3.1. Grid de tamaño diez . . . . .	129
A.3.2. Grid de tamaño quince . . . . .	135
A.3.3. Grid de tamaño veinte . . . . .	141
A.4. Redes de tamaño ochocientos . . . . .	147
A.4.1. Grid de tamaño diez . . . . .	147
A.4.2. Grid de tamaño veinte . . . . .	152
A.4.3. Grid de tamaño treinta . . . . .	158
<b>Referencias</b>	<b>164</b>

# Índice de figuras

1.1. Red multihop vs. red single-hop. . . . .	3
1.2. Problemas de la terminal oculta y expuesta. . . . .	8
1.3. Distintas capas de las redes ad hoc. . . . .	12
2.1. Una gráfica. . . . .	16
2.2. Gráfica con pesos. . . . .	16
2.3. Gráfica inducida. . . . .	17
2.4. Un ciclo y un circuito en una gráfica. . . . .	18
2.5. Disco unitario centrado en un vértice. . . . .	20
2.6. Gráficas aplanable, plana y no aplanable. . . . .	20
2.7. Gráfica con tres regiones. . . . .	21
2.8. Gráfica con número de dominación igual a uno. . . . .	22
2.9. Polígonos simple y no simple. . . . .	24
2.10. Cierre convexo. . . . .	24
2.11. Subdivisión del plano en regiones cuadradas. . . . .	25
2.12. Mosaico. . . . .	26
3.1. Viajar de $s$ a $t$ usando compass routing. . . . .	29
3.2. Compass routing no siempre encuentra una ruta. . . . .	29
3.3. Compass routing en triangulaciones Delaunay. . . . .	31
3.4. Compass Routing II. . . . .	32
3.5. Ilustración de <code>compass routing II</code> en una gráfica no convexa. . . . .	33
3.6. Arista de Gabriel. . . . .	34
3.7. Se puede calcular localmente $GDU(P) \cap GG(P)$ . . . . .	35
3.8. Ilustra el comportamiento de <code>FACE-2</code> en una gráfica geométrica. . . . .	36
3.9. Caso en el que <code>FACE-2</code> requiere de $\Omega(n^2)$ pasos. . . . .	36
4.1. Algoritmo de Wang y Li. . . . .	40
4.2. Heurística para obtener conjuntos dominantes localmente. . . . .	41
4.3. Malla cuadrada. . . . .	42
4.4. Malla hexagonal. . . . .	43
4.5. Subdivisión de Panal. . . . .	43
4.6. Puntos a distancia mayor que uno en cuadrado. . . . .	46

4.7. No puede haber 6 círculos intersectando a $C$ sin que se inter- secten mutuamente. . . . .	50
4.8. Coloración de la subdivisión hexagonal. . . . .	51
4.9. Resultados simulación red con 100 nodos y grid 5. . . . .	54
4.10. Resultados simulación red con 100 nodos y grid 6. . . . .	55
4.11. Resultados simulación red con 100 nodos y grid 7. . . . .	55
4.12. Resultados simulación red con 200 nodos y grid 5. . . . .	56
4.13. Resultados simulación red con 200 nodos y grid 6. . . . .	56
4.14. Resultados simulación red con 200 nodos y grid 7. . . . .	57
4.15. Resultados simulación red con 500 nodos y grid 10. . . . .	58
4.16. Resultados simulación red con 500 nodos y grid 15. . . . .	59
4.17. Resultados simulación red con 500 nodos y grid 20. . . . .	59
4.18. Resultados simulación red con 800 nodos y grid 10. . . . .	60
4.19. Resultados simulación red con 800 nodos y grid 20. . . . .	60
4.20. Resultados simulación red con 800 nodos y grid 30. . . . .	61
A.1. Algoritmo glotón, $n=100$ , $grid=5$ . . . . .	66
A.2. Heurística subdivisión cuadrada, $n=100$ , $grid=5$ , $arista=1cm$	67
A.3. Heurística subdivisión cuadrada, $n=100$ , $grid=5$ , $arista=2cm$	68
A.4. Heurística subdivisión cuadrada, $n=100$ , $grid=5$ , $arista=3cm$	69
A.5. Heurística subdivisión hexagonal, $n=100$ , $grid=5$ , $arista=1cm$	70
A.6. Heurística subdivisión hexagonal, $n=100$ , $grid=5$ , $arista=2cm$	71
A.7. Heurística subdivisión triangular, $n=100$ , $grid=5$ , $arista=1cm$	72
A.8. Heurística subdivisión triangular, $n=100$ , $grid=5$ , $arista=2cm$	73
A.9. Heurística subdivisión triangular, $n=100$ , $grid=5$ , $arista=3cm$	74
A.10. Heurística subdivisión triangular, $n=100$ , $grid=5$ , $arista=4cm$	75
A.11. Heurística subdivisión triangular, $n=100$ , $grid=5$ , $arista=5cm$	76
A.12. Algoritmo glotón, $n=100$ , $grid=6$ . . . . .	77
A.13. Heurística subdivisión cuadrada, $n=100$ , $grid=6$ , $arista=1cm$	78
A.14. Heurística subdivisión cuadrada, $n=100$ , $grid=6$ , $arista=2cm$	79
A.15. Heurística subdivisión cuadrada, $n=100$ , $grid=6$ , $arista=3cm$	80
A.16. Heurística subdivisión hexagonal, $n=100$ , $grid=6$ , $arista=1cm$	81
A.17. Heurística subdivisión hexagonal, $n=100$ , $grid=6$ , $arista=2cm$	82
A.18. Heurística subdivisión triangular, $n=100$ , $grid=6$ , $arista=1cm$	83
A.19. Heurística subdivisión triangular, $n=100$ , $grid=6$ , $arista=2cm$	84
A.20. Heurística subdivisión triangular, $n=100$ , $grid=6$ , $arista=3cm$	85
A.21. Heurística subdivisión triangular, $n=100$ , $grid=6$ , $arista=4cm$	86
A.22. Heurística subdivisión triangular, $n=100$ , $grid=6$ , $arista=5cm$	87
A.23. Algoritmo glotón, $n=100$ , $grid=7$ . . . . .	88
A.24. Heurística subdivisión cuadrada, $n=100$ , $grid=7$ , $arista=1cm$	89
A.25. Heurística subdivisión cuadrada, $n=100$ , $grid=7$ , $arista=2cm$	90
A.26. Heurística subdivisión cuadrada, $n=100$ , $grid=7$ , $arista=3cm$	91
A.27. Heurística subdivisión hexagonal, $n=100$ , $grid=7$ , $arista=1cm$	92
A.28. Heurística subdivisión hexagonal, $n=100$ , $grid=7$ , $arista=2cm$	93

A.29.Heurística subdivisión triangular, n=100, grid=7, arista=1cm	94
A.30.Heurística subdivisión triangular, n=100, grid=7, arista=2cm	95
A.31.Heurística subdivisión triangular, n=100, grid=7, arista=3cm	96
A.32.Heurística subdivisión triangular, n=100, grid=7, arista=4cm	97
A.33.Heurística subdivisión triangular, n=100, grid=7, arista=5cm	98
A.34.Algoritmo glotón, n=200, grid=5. . . . .	99
A.35.Heurística subdivisión cuadrada, n=200, grid=5, arista=1cm	100
A.36.Heurística subdivisión cuadrada, n=200, grid=5, arista=2cm	101
A.37.Heurística subdivisión cuadrada, n=200, grid=5, arista=3cm	102
A.38.Heurística subdivisión hexagonal, n=200, grid=5, arista=1cm	103
A.39.Heurística subdivisión hexagonal, n=200, grid=5, arista=2cm	104
A.40.Heurística subdivisión triangular, n=200, grid=5, arista=1cm	105
A.41.Heurística subdivisión triangular, n=200, grid=5, arista=2cm	106
A.42.Heurística subdivisión triangular, n=200, grid=5, arista=3cm	107
A.43.Heurística subdivisión triangular, n=200, grid=5, arista=4cm	108
A.44.Algoritmo glotón, n=200, grid=6. . . . .	109
A.45.Heurística subdivisión cuadrada, n=200, grid=6, arista=1cm	110
A.46.Heurística subdivisión cuadrada, n=200, grid=6, arista=2cm	111
A.47.Heurística subdivisión cuadrada, n=200, grid=6, arista=3cm	112
A.48.Heurística subdivisión hexagonal, n=200, grid=6, arista=1cm	113
A.49.Heurística subdivisión hexagonal, n=200, grid=6, arista=2cm	114
A.50.Heurística subdivisión triangular, n=200, grid=6, arista=1cm	115
A.51.Heurística subdivisión triangular, n=200, grid=6, arista=2cm	116
A.52.Heurística subdivisión triangular, n=200, grid=6, arista=3cm	117
A.53.Heurística subdivisión triangular, n=200, grid=6, arista=4cm	118
A.54.Algoritmo glotón, n=200, grid=7. . . . .	119
A.55.Heurística subdivisión cuadrada, n=200, grid=7, arista=1cm	120
A.56.Heurística subdivisión cuadrada, n=200, grid=7, arista=2cm	121
A.57.Heurística subdivisión cuadrada, n=200, grid=7, arista=3cm	122
A.58.Heurística subdivisión hexagonal, n=200, grid=7, arista=1cm	123
A.59.Heurística subdivisión hexagonal, n=200, grid=7, arista=2cm	124
A.60.Heurística subdivisión triangular, n=200, grid=7, arista=1cm	125
A.61.Heurística subdivisión triangular, n=200, grid=7, arista=2cm	126
A.62.Heurística subdivisión triangular, n=200, grid=7, arista=3cm	127
A.63.Heurística subdivisión triangular, n=200, grid=7, arista=4cm	128
A.64.Algoritmo glotón, n=500, grid=10. . . . .	129
A.65.Heurística subdivisión cuadrada, n=500, grid=10, arista=1cm	130
A.66.Heurística subdivisión cuadrada, n=500, grid=10, arista=2cm	131
A.67.Heurística subdivisión cuadrada, n=500, grid=10, arista=3cm	132
A.68.Heurística subdivisión hexagonal, n=500, grid=10, arista=1cm	133
A.69.Heurística subdivisión hexagonal, n=500, grid=10, arista=2cm	134
A.70.Algoritmo glotón, n=500, grid=15. . . . .	135
A.71.Heurística subdivisión cuadrada, n=500, grid=15, arista=1cm	136
A.72.Heurística subdivisión cuadrada, n=500, grid=15, arista=2cm	137

A.73.Heurística subdivisión cuadrada, n=500, grid=15, arista=3cm 138  
 A.74.Heurística subdivisión hexagonal, n=500, grid=15, arista=1cm 139  
 A.75.Heurística subdivisión hexagonal, n=500, grid=15, arista=2cm 140  
 A.76.Algoritmo glotón, n=500, grid=20. . . . . 141  
 A.77.Heurística subdivisión cuadrada, n=500, grid=20, arista=1cm 142  
 A.78.Heurística subdivisión cuadrada, n=500, grid=20, arista=2cm 143  
 A.79.Heurística subdivisión cuadrada, n=500, grid=20, arista=3cm 144  
 A.80.Heurística subdivisión hexagonal, n=500, grid=20, arista=1cm 145  
 A.81.Heurística subdivisión hexagonal, n=500, grid=20, arista=2cm 146  
 A.82.Algoritmo glotón, n=800, grid=10. . . . . 147  
 A.83.Heurística subdivisión cuadrada, n=800, grid=10, arista=1cm 148  
 A.84.Heurística subdivisión cuadrada, n=500, grid=20, arista=2cm 149  
 A.85.Heurística subdivisión cuadrada, n=800, grid=10, arista=3cm 150  
 A.86.Heurística subdivisión hexagonal, n=800, grid=10, arista=1cm 151  
 A.87.Algoritmo glotón, n=800, grid=20. . . . . 152  
 A.88.Heurística subdivisión cuadrada, n=800, grid=20, arista=1cm 153  
 A.89.Heurística subdivisión cuadrada, n=500, grid=20, arista=2cm 154  
 A.90.Heurística subdivisión cuadrada, n=800, grid=20, arista=3cm 155  
 A.91.Heurística subdivisión hexagonal, n=800, grid=20, arista=1cm 156  
 A.92.Heurística subdivisión hexagonal, n=800, grid=20, arista=2cm 157  
 A.93.Algoritmo glotón, n=800, grid=30. . . . . 158  
 A.94.Heurística subdivisión cuadrada, n=800, grid=30, arista=1cm 159  
 A.95.Heurística subdivisión cuadrada, n=500, grid=30, arista=2cm 160  
 A.96.Heurística subdivisión cuadrada, n=800, grid=30, arista=3cm 161  
 A.97.Heurística subdivisión hexagonal, n=800, grid=30, arista=1cm 162  
 A.98.Heurística subdivisión hexagonal, n=800, grid=30, arista=2cm 163

# Introducción

Imagínese que un turista se encuentra en una ciudad que desconoce, y de la cual no posee ningún mapa. Supóngase además que dicho turista es un tanto cuanto desmemoriado, de tal forma que si ha pasado por algún punto de la ciudad más de una vez no necesariamente lo recordará. Al turista le interesa fotografiar todos los lugares turísticamente importantes de la ciudad sin tener que visitarlos todos. ¿Será posible que el turista cumpla con su cometido?

Este problema se puede modelar de la siguiente manera. Dibújese en el plano un punto por cada edificio, parque, mercado o cualquier punto importante de la ciudad, y una línea entre dos puntos si desde uno se puede fotografiar al otro. Llámese a los puntos vértices y a las líneas aristas. Entonces, el turista debería visitar un subconjunto particular de vértices de tal forma que desde ese subconjunto logre fotografías de todos los demás puntos de interés de la ciudad. Este dibujo formado por vértices y aristas se conoce como una gráfica geométrica, las cuales se definen formalmente en el capítulo 2 de esta tesis. El estudio de gráficas geométricas fue iniciado por S. Avital y H. Hanani [1], P. Erdős, Y. Kupitz [2] y M. Perles. Ellos se dieron cuenta que muchos problemas clásicos en teoría extremal de gráficas [3], tenían una analogía natural en gráficas geométricas. En [4] se puede encontrar un muy buen compendio de resultados y problemas abiertos en el área.

Agregar al problema las restricciones de que el turista es desmemoriado y de que no tiene ningún mapa de la ciudad no es una mera ocurrencia. La justificación para agregar estas condiciones resulta del hecho de que este problema tiene aplicaciones en un tipo especial de redes conocidas como redes ad hoc móviles, de las cuales se hablará en el capítulo 1. La topología de estas redes cambia continuamente, así que es prácticamente imposible tener conocimiento global de la misma en todo momento, de aquí la condición de que el turista no tenga un mapa. El darle memoria al turista implicaría que un algoritmo que corra sobre una de estas redes tendría que utilizar una enorme cantidad de memoria o dejar marcas en los vértices de la red, lo cual decrementaría en gran medida la eficacia del algoritmo. Así pues, algoritmos que cumplen con estas dos condiciones se conocen como algoritmos locales, de estos se habla con detalle en el capítulo 3.

En este trabajo se presenta un algoritmo que resuelve el problema descrito en el primer párrafo. Específicamente, se resuelve el problema de encontrar un subconjunto especial de vértices en una red ad hoc de manera local. La red ad hoc es modelada utilizando una gráfica geométrica particular, y después se encuentra en esta el subconjunto de vértices de interés, al cual se le conoce como conjunto dominante. Encontrar un conjunto dominante para una red ad hoc, es un problema de gran interés en la literatura, ya que de esto dependen muchos algoritmos para este tipo de redes. La trascendencia del problema se entenderá más fácilmente conforme se avance en la lectura de este trabajo. Encontrar un conjunto dominante de una red ad hoc de manera local es un problema abierto, en este trabajo se muestra una heurística local que lo encuentra.

El orden de los capítulos de esta tesis es el siguiente. En el capítulo 1 se presenta un panorama general sobre redes ad hoc, que va desde su definición y sus características, hasta las líneas de investigación que estas han generado. En el capítulo 2 se dan los conceptos preliminares necesarios para lograr la total comprensión de este trabajo. Después, en el capítulo 3 se discuten los dos artículos pioneros sobre algoritmos locales en redes ad hoc. Luego, en el capítulo 4 se describe formalmente el problema a resolver en esta tesis, así como la heurística a través de la cual el mismo fue resuelto y se presentan los resultados teóricos y de simulaciones. Finalmente se dan algunas conclusiones.

# Capítulo 1

## Redes ad hoc móviles

### 1.1. Introducción

Hoy en día, cualquier persona que desee comprar una computadora portátil deberá decidir si desea adquirirla con la función *bluetooth*. Lo mismo sucedería con muchos otros accesorios, como teléfonos móviles, organizadores PDA (*personal digital assistant*), reproductores de mp3, etcétera, ya que todos se encuentran equipados con esta tecnología. Pero, ¿qué es esta función y para que sirve? La palabra bluetooth proviene del nombre del rey Danés y Noruego Harald Blåtand (cuya traducción al inglés es Harald Bluetooth), quien unificó las tribus Noruegas, Suecas y Danesas. En este mismo sentido, la tecnología bluetooth pretende unificar la comunicación entre distintos accesorios.

Imagínese un escenario en el cual los participantes en una reunión puedan compartir documentos o presentaciones, gracias a la interacción directa de sus accesorios electrónicos, o bien en las cuales las tarjetas de presentación puedan automáticamente almacenarse en el registro de direcciones de una computadora portátil, así como en el directorio de un teléfono celular. O en el cual una persona reciba su correo electrónico en una computadora que está cargando en su portafolio, pero leer y responder a los correos desde su PDA. Esto es un ejemplo de comunicación espontánea entre dispositivos inalámbricos, y puede definir perfectamente un esquema conocido como establecimiento de una red ad hoc, el cual permite a los accesorios comunicarse en cualquier lugar y a cualquier hora sin la necesidad de alguna infraestructura. Actualmente, tecnologías como bluetooth hacen posible la materialización de este tipo de redes inalámbricas. En este capítulo se dan algunas bases sobre redes ad hoc.

## 1.2. Redes ad hoc

### 1.2.1. Panorama general

Favorecidas por los avances tecnológicos, muchos tipos de redes inalámbricas se hacen cada vez más populares en el mundo actual. Día con día los nuevos avances mejoran la calidad de la comunicación inalámbrica, se decrementa el costo y el tamaño de los componentes, y se incrementa la capacidad computacional de los mismos. Todos estos factores provocan un aumento cada vez mayor en la demanda del uso de redes inalámbricas, lo cual necesariamente obliga a la proliferación de las mismas. Actualmente, las redes inalámbricas se pueden clasificar en dos tipos: 1) Las redes con infraestructura, o redes celulares; y 2) las redes *ad hoc* móviles, o redes inalámbricas *multihop*.

Las redes celulares están compuestas por dispositivos móviles inalámbricos (*nodos*), los cuales se comunican con estaciones base, conectadas a una infraestructura fija. Si un nodo se mueve fuera del rango de alcance de una estación base, ocurre un cambio a una nueva estación, que esté dentro del rango de transmisión. Los datos se transmiten a través de la red alámbrica, y sólo el último salto es inalámbrico. Es decir, los datos son difundidos de una estación base a otra, hasta encontrar la estación que está dentro del alcance del nodo destino. Una vez ahí, los datos son transmitidos de esta última estación al nodo, es por esto que sólo el último salto es inalámbrico. Estas redes se conocen también como *single-hop*.

Las redes ad hoc móviles (MANETs, por sus siglas en inglés), están compuestas por dispositivos inalámbricos, que pueden estar colocados en aviones, barcos, autos, personas, etc. Estos dispositivos, se comunican entre sí en ausencia de una infraestructura fija, o de una administración central. Así que estas redes deben ser capaces de auto-configurarse y auto-organizarse. Dos nodos en una MANET se pueden comunicar si la distancia entre estos es menor que el mínimo de sus dos rangos de transmisión. Generalmente no es posible que todos los nodos de una MANET se puedan comunicar[1], es por esto que deben actuar como *ruteadores* y remitir los paquetes a nombre de otros nodos, para proveer comunicación en toda la red (esta característica se conoce como *multihop*). Debido a que los nodos de una MANET pueden moverse de manera aleatoria, los cambios en la topología de la red son frecuentes y no estructurados. En este ambiente dinámico, las funciones de la red deben correr de manera distribuida, ya que los nodos pueden aparecer o desaparecer de repente, y además estos no necesariamente conocen la topología de la red.

En la Figura 1.1, se puede apreciar la diferencia entre una red *single-hop* y una red *multihop*. Al nodo que desea enviar información se le llama origen, y al que debe recibirla se le llama destino. Estos están representados en la figura con colores rojo y azul, respectivamente. El rango de alcance de un

nodo es representado con un círculo punteado centrado en este. En la red single-hop, el destino está dentro del rango de transmisión del origen, así que el primero sólo debe transmitir la información al segundo. Por otro lado, en la red multihop, se necesitan nodos reeivo (mostrados en gris claro) pues en ocasiones el nodo destino no está dentro del círculo del origen. Cuando esto sucede el origen envía la información a un primer nodo reeivo, con el que se puede comunicar directamente, si este no tiene el destino dentro de su alcance lo reeiva a otro vértice, etcétera, hasta alcanzar el destino.

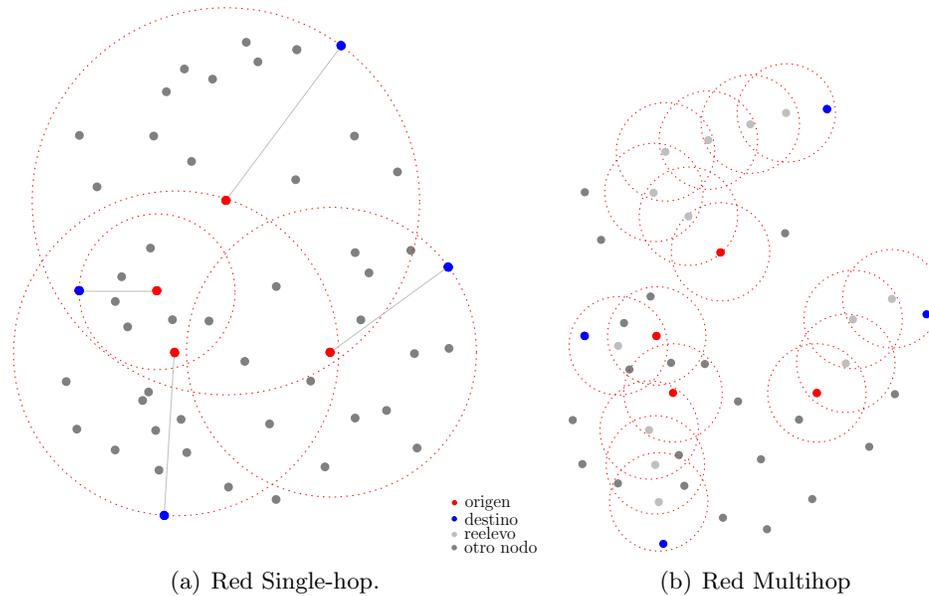


Figura 1.1: Comparación de una red multihop con una red single-hop. Ambas redes tienen una distribución idéntica de nodos. Figura tomada de [2].

Las MANETS tienen varias ventajas sobre las redes celulares. Algunas de estas son, la facilidad y rapidez con las que se pueden construir, su robustez y su flexibilidad e inherente soporte para la movilidad. En escenarios tales como campos de batalla o desastres (inundaciones, terremotos, incendios), las redes alámbricas no están disponibles, y la única opción para comunicarse y para tener acceso a la información, la brindan las redes ad hoc. Para una lista más detallada de sus aplicaciones véase [2] y las referencias ahí mencionadas.

La primer red ad hoc de la que se tiene conocimiento es la red ALOHA, que se empezó a construir en 1968 en Hawaii[3]. Después, en 1973, se desarrolló la red conocida como PRNet (Packet Radio Network), dentro del proyecto DARPA (Defense Advance Research Projects Agency)[4]. En esta red todos los componentes eran móviles, y aunque su movilidad estaba limitada, la red era ya multihop. En 1997, cuando se desarrolló el estándar IEEE 802.11 [5] para redes inalámbricas locales (WLAN), el Instituto de Ingeniería Eléctrica

y Electrónica (IEEE) reemplazó el término *packet-radio* por ad hoc [2].

La investigación en redes ad hoc empezó formalmente con la creación del grupo “*Mobile Ad hoc Networks*”<sup>1</sup>, dentro de la agrupación “*Internet Engineering Task Force*” (IETF) [6]. Este grupo se encuentra actualmente trabajando en protocolos de envío de paquetes apropiados para redes ad hoc.

### 1.2.2. Características

Las redes ad hoc presentan varias particularidades que las hacen muy distintas de las redes celulares. Estas diferencias se deben principalmente a su falta de infraestructura, la movilidad de sus nodos, y al hecho de que son inalámbricas. En 1992, el grupo “*Mobile Ad hoc Networks*”, listó en el RFC 2501[7] las características principales de las MANETs, las cuales se presentan a continuación.

- **Topología Dinámica.** La topología de la red puede cambiar rápida y aleatoriamente, debido principalmente a tres factores, la capacidad de movimiento de los nodos, su aparición o desaparición imprevista en la red, y al ajuste en sus rangos de transmisión. A pesar de esto, la conectividad de la red y todos los servicios que esta proporciona deben ser mantenidos, para que las aplicaciones y los servicios puedan operar de manera ininterrumpida.
- **Capacidad limitada de las ligas.** La capacidad de un liga<sup>2</sup> se mide de acuerdo al número de bits que esta transporta por segundo. Un sistema de comunicación inalámbrico ideal proveería alta capacidad de transmisión y confiabilidad, usando un mínimo consumo de energía y de ancho de banda<sup>3</sup>. Sin embargo, un sistema con tales características aún no existe.

Las MANETs utilizan ondas de radio para comunicarse. Los canales de radio transportan señales en el espectro electromagnético. Estas ondas son un medio de propagación muy atractivo, debido a sus características, como no requerir de la instalación de algún medio alámbrico para funcionar, poder penetrar paredes, proveer conectividad para usuarios móviles y potencialmente transportar señales por grandes distancias. Las características de un canal de radio dependen en gran medida del ambiente en el que se propagan las ondas, y a la distancia que una señal es transportada. Las condiciones del ambiente determinan varios

---

<sup>1</sup>En esta tesis, los nombres de los grupos, institutos, algoritmos, etc., que no tengan una traducción estándar al español, se mantendrán en inglés con el propósito de evitar las traducciones imprecisas.

<sup>2</sup>En este trabajo se utilizará *liga*, como la traducción de la palabra en inglés *link*

<sup>3</sup>El espectro de frecuencia de un liga es compartido entre las conexiones establecidas a través de esta. La liga dedica una frecuencia de banda a cada conexión. El ancho de esta se conoce como ancho de banda.

fenómenos de pérdida de la señal, como son, *shadow fading* (que es la atenuación de la señal debido a la presencia de obstáculos entre el transmisor y el receptor), *multipath fading* (que se presenta cuando el destino recibe varias “copias” de la señal original, debido a la reflexión de ésta al chocar con obstáculos), o interferencia (que es cuando la onda se superpone con alguna otra señal electromagnética).

Los canales de radio pueden ser terrestres o satelitales. Los canales terrestres pueden ser clasificados en dos tipos: aquellos que operan en redes locales (LAN, por sus siglas en inglés), que se expanden sólo algunos cientos de metros, y los que se utilizan para servicios de acceso móvil, que típicamente operan en una región metropolitana. Existe gran variedad de productos para redes locales, los cuales operan en rangos de transmisión que van desde uno hasta decenas de megabits por segundo. Los servicios para acceso móvil proveen canales que operan en rangos de decenas de kilobits por segundo. Estos rangos de comunicación, comparados con los de ligas alámbricas, son muy bajos, ya que medios como la fibra óptica pueden alcanzar rangos de transmisión de hasta centenas de gigabits por segundo.

En [8] se presenta una discusión clara y un resumen completo sobre la tecnología de canales inalámbricos.

- **Nodos con poca energía.** Generalmente los nodos de una MANET utilizan baterías como fuente de energía. Para estos nodos es de vital importancia que se diseñen sistemas cuya principal característica sea la optimización del consumo de la misma. A pesar de los esfuerzos para disminuir el uso de energía en los electrónicos que utilizan pilas [9], aún no se ha obtenido ningún logro significativo. Y además no se espera un aumento importante en la capacidad de las baterías en los próximos años [10].
- **Seguridad física limitada.** Para garantizar la seguridad en una MANET, se deben tomar en cuenta parámetros que a simple vista pueden parecer similares que aquellos para una red fija, pero en realidad difieren en gran medida, debido a que las redes ad hoc están más expuestas a ataques que las redes celulares o fijas. Para un análisis más detallado sobre la seguridad en redes ad hoc véase [11, 12].

En el diseño de protocolos para redes ad hoc se debe tomar en cuenta las particularidades descritas anteriormente, las cuales hacen que los mecanismos existentes para redes fijas no sean apropiados para MANETs. El trabajo presentado en esta tesis tiene que ver principalmente con las dos primeras características listadas.

### 1.2.3. Ejemplos de redes ad hoc

Como se ha mencionado antes la tecnología de las redes ad hoc tiene sus orígenes en la red conocida como Packet Radio Network, desde entonces, esta tecnología ha sufrido grandes cambios y día con día evoluciona, permitiendo la creación de nuevos modelos de redes. A continuación se mencionan algunos ejemplos de redes ad hoc comunes.

- **Redes inalámbricas personales (WPAN).** Una WPAN es una red formada por dispositivos electrónicos tales como, computadoras portátiles, teléfonos móviles, organizadores PDA, reproductores de mp3/DVD, etc. El alcance de una WPAN es típicamente de algunos metros, y es utilizada para la comunicación entre los dispositivos o para conectarse a Internet. La tecnología comúnmente utilizada en las WPAN es bluetooth [13].
- **Redes de sensores.** Los avances en la tecnología de microelectrónicos han hecho posible la creación de sensores multifuncionales, pequeños y capaces de comunicarse entre si en distancias cortas, que además son de bajo costo y necesitan poca energía para trabajar. Esto ha originado la creación de redes de sensores, las cuales se componen de un gran número de nodos, que son densamente desplegados sobre algún área de interés. Pueden utilizarse para encontrar objetos, conocer los campos de batalla o de desastres, medir datos fisiológicos de una persona remota, la concentración de elementos contaminantes en el ambiente, etc. Esencialmente una red de este tipo es utilizada para obtener información sobre el medio en el que ha sido desplegada. Los nodos deben recopilar los datos, para enviarlos después a un nodo responsable de unirlos, el cual a su vez los envía a un centro de monitoreo. Los sensores son capaces de procesar parcialmente los datos obtenidos antes de enviarlos.

Mucho más puede decirse de estas redes, pero hacerlo desviaría la atención del enfoque de este trabajo. Para un buen resumen sobre redes de sensores véase [14].

- **Redes espontáneas.** Una red espontánea puede definirse como la integración de servicios y dispositivos en una red, con la finalidad de disponer instantáneamente de servicios, sin la necesidad de intervención humana[15]. Una forma más intuitiva de describirla es la siguiente. Una red espontánea es una red ad hoc que es creada cuando un grupo de personas se reúnen y utilizan dispositivos inalámbricos para alguna actividad computacional conjunta. Los usuarios no necesariamente identifican a todos los participantes, ni tampoco deben configurar sus computadoras previamente para poder establecer la red [16].

- **Redes ad hoc vehiculares.** Como su nombre lo indica, una red ad hoc vehicular es simplemente una red ad hoc en donde los nodos móviles son vehículos. Existe gran diversidad de aplicaciones para estas redes, señalamientos de emergencia[17], distribución del tráfico, información sobre las condiciones de los caminos, comunicación entre vehículos[18], etc. Una red de este tipo tiene algunas ventajas sobre otras redes ad hoc. Los vehículos pueden fácilmente proveer la energía y el espacio necesario para los nodos; y muchos automóviles cuentan ya con sistema GPS (*Global Positioning System*) que les permite conocer su posición geográfica exacta[19]. Sin embargo, la movilidad de los nodos puede ser muy alta, aunque restringida solamente a los caminos.

Hay gran interés en la industria de financiar investigación en esta línea [20].

Existen muchos otros tipos de redes, que en ocasiones se clasifican como redes ad hoc, pero en la mayoría de los casos son redes híbridas que combinan características de redes ad hoc con características de redes fijas o celulares.

#### 1.2.4. Líneas de investigación

La implementación exitosa de las redes ad hoc móviles impone nuevos retos en la investigación que difieren de los ya existentes para las redes fijas o celulares. A continuación se describen brevemente algunas de las diferentes líneas de investigación en redes ad hoc, generadas por dichos retos.

##### Control de acceso al medio (*MAC*)

De la misma manera que en las redes alámbricas se debe controlar el acceso a los canales de comunicación, las redes ad hoc deben tener un mecanismo que controle el acceso al uso de la frecuencia de radio que se utiliza para las comunicaciones, dicho mecanismo se conoce como mecanismo de control de acceso al medio (*MAC*, por sus siglas en inglés). Como en una red ad hoc el medio es un recurso compartido y escaso, controlar eficientemente el acceso al mismo es una tarea complicada[21], que no puede ser resuelta con los protocolos *MAC* utilizados para redes fijas. Recuérdese que los principales protocolos *MAC* para redes fijas están basados en mecanismos que sensan el canal de comunicación antes de utilizarlo [22, 23, 24, 25]. Sin embargo, esto no es suficiente para redes ad hoc.

Los protocolos *MAC* para redes ad hoc deben lidiar principalmente con dos fenómenos que afectan su capacidad: el problema de la terminal oculta, y el de la terminal expuesta. Supóngase que un nodo *A* está transmitiendo a un nodo *B*, y que un nodo *C* está transmitiendo al nodo *B*, tal como se muestra en la Figura 1.2. El problema de la terminal oculta surge cuando ambos

nodos  $A$  y  $C$  están fuera de sus alcances de transmisión, y esto impide que detecten sus señales. Si ambos nodos se comunican con en destino  $B$  al mismo tiempo, estas señales interfieren en  $B$ , destruyendo los datos originales. El problema de la terminal expuesta puede explicarse de la siguiente forma. Las terminales expuestas (por ejemplo el nodo  $D$  en la Figura 1.2), son nodos dentro del rango del transmisor  $A$ , pero no del receptor  $B$ . Si se utilizara un mecanismo clásico de censo del canal,  $D$  detectaría que este está ocupado y se abstendría de comunicarse con  $A$ , cuando en realidad dicha comunicación no generaría ningún problema.

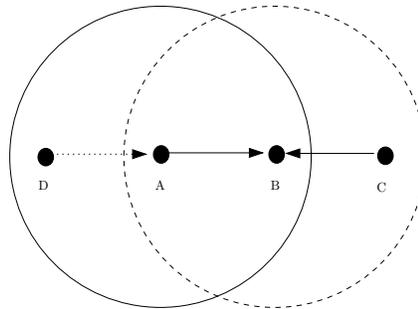


Figura 1.2: Ejemplificación de los problemas de la terminal oculta y expuesta.

Existen básicamente dos categorías de protocolos MAC: 1) Protocolos de Acceso Aleatorio, y 2) Protocolos de Acceso Controlado. En el primero, los nodos deben competir entre ellos para ganar el acceso al medio y poder transmitir datos. Mientras que en el segundo, alguna autoridad decide qué nodo tiene el acceso al medio.

Debido a la naturaleza de las redes ad hoc, los protocolos de acceso aleatorio son comúnmente utilizados. Entre estos se pueden mencionar, MACA (*Multiple Access Collision Avoidance*) [26], MACAW (*MACA with Acknowledgment*) [27], MACA-BI (*MACA by Invitation*) [28], DBTMA (*Dual Busy Tone Multiple Access*) [29], entre otros.

El protocolo, CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) [30], se ha elegido para estar en el estándar IEEE 802.11. Este protocolo está basado en CSMA (*Carrier Sense Multiple Access*) [22], un protocolo MAC para redes LAN. CSMA/CA resuelve los problemas de la terminal oculta y expuesta, mencionados anteriormente. Sin embargo, aún quedan algunos otros problemas por resolver [29].

Existen gran variedad de protocolos de acceso controlado, los cuales se pueden clasificar de acuerdo al método que utilizan para dividir los recursos. Tres métodos aceptados son FDMA (*Frequency Division Multiple Access*), TDMA (*Time Division Multiple Access*), y CDMA (*Code Division Multiple Access*) [31]. Este tipo de métodos se utilizan principalmente en aplicaciones bluetooth y algunos tipos de redes ad hoc, donde el acceso al recurso

compartido es controlado por nodos maestros [32].

En [14] se puede encontrar una descripción más detallada de algunos protocolos MAC para redes ad hoc.

### Búsqueda de rutas (*routing*)

La búsqueda de rutas (camino) en una red (*routing* en inglés), es el problema de enviar un paquete de un nodo origen a un nodo destino. Una ruta entre dos nodos consiste de nodos que funcionan como reelevos en el paso del mensaje a través de la red, tal y como se mencionó en la sección 1.2.1 al describir las redes multihop. La tarea de encontrar y mantener rutas en las MANETS no es trivial, debido a los cambios frecuentes en la topología de las mismas. Existen en la literatura una gran variedad de protocolos para la búsqueda de rutas, de acuerdo con Mauve, Widmer y Hartenstein[33], se pueden distinguir entre dos enfoques diferentes: los basados en la topología de la red, y los basados en la posición de los nodos.

Los algoritmos de búsqueda de ruta basados en la topología, utilizan la información acerca de las ligas que existen en la red para hacer el envío de paquetes. Estos se pueden clasificar en tres tipos: pro-activos, reactivos e híbridos.

La característica que distingue a los algoritmos pro-activos es que mantienen información sobre todos los caminos disponibles en la red, aún si estos no están siendo utilizados. Este tipo de algoritmos han sido derivados de los algoritmos clásicos de búsqueda de rutas, como son *Distance-Vector Routing* o *Link-State Routing* (para una descripción básica de estos algoritmos véase [34]). La información sobre los caminos disponibles es mantenida en tablas. Cada nodo almacena una o mas tablas con esta información, y cuando un cambio en la topología sucede esta información es propagada a través de toda la red, de tal forma que las tablas se mantengan consistentes. Este tipo de algoritmos presentan grandes desventajas cuando son utilizados en redes ad hoc. Su principal desventaja es tener grandes dificultades para mantener las tablas con información sobre rutas válidas, debido a la topología dinámica de la red, teniendo como consecuencia la pérdida de mensajes [35] y además, mantener todas las tablas causa un gran aumento en el tráfico de la red.

La búsqueda de soluciones para estos problemas dio origen a los algoritmos reactivos (por ejemplo DSR [36], TORA[37], AODV[38]). En contraste con los algoritmos pro-activos, los reactivos buscan rutas sólo cuando estas se necesitan. Una vez que una ruta es encontrada, la información sobre esta es mantenida hasta que la ruta deja de estar en uso o se rompe. Esto reduce la carga de la red cuando hay un número pequeño de rutas siendo utilizadas. Sin embargo, este tipo de algoritmos presentan también algunas limitaciones. Primero, como las rutas se mantienen sólo mientras están siendo utilizadas, generalmente es necesario hacer una búsqueda de ruta antes de poder enviar algún paquete, ocasionando un retraso en el envío del primer

paquete transmitido entre un par de nodos. Segundo, si los cambios en la topología de la red son frecuentes, lo cual generalmente sucede en MANETS, el mantener las rutas, aunque sean solamente las utilizadas, puede causar aún un significativo aumento en el tráfico de la red. Finalmente, si la ruta hacia el destino cambia los paquetes que estén en camino hacia este pueden perderse.

Los algoritmos de búsqueda de ruta basados en la posición de los nodos, eliminan algunas de las limitaciones de los basados en la topología utilizando información extra. Esta información es la posición física de los nodos de la red. Encontrar dicha posición no es mayor problema, gracias a los sistemas de posicionamiento como el GPS [39, 40] o algún otro [41, 42]. Un resumen sobre estos sistemas de posicionamiento es [43].

En estos algoritmos, el nodo origen incluye en el paquete a enviar la posición del nodo destino, y cada nodo reelevo toma la decisión de hacia donde enviar el paquete basándose principalmente en dicha posición. Así que no se requiere del almacenamiento ni mantenimiento de tablas. Existen en la literatura gran variedad de algoritmos de este tipo, pues debido a que presentan grandes ventajas sobre los algoritmos basados en la topología, recientemente hay un gran interés en desarrollar algoritmos como estos. En [33] se puede encontrar un buen resumen de algoritmos basados en posición para búsqueda de rutas. En el capítulo 3 de este trabajo se presentan dos algoritmos para búsqueda de rutas basados en posición: *compass routing* y *face routing*.

Cabe mencionar que aquí se ha definido la búsqueda de rutas como el problema de enviar un paquete de un sólo origen a un sólo destino. Sin embargo, existe un problema extremadamente relacionado con este, conocido como difusión múltiple (*multicasting*): el envío de paquetes de un sólo origen a múltiples destinos. El criterio con el cual se seleccionan los destinos sirve para subdividir el problema en categorías, dos de estas son:

Difusión Geográfica (*Geocasting*)—Cuando se desea enviar el paquete a todos los nodos en cierta región geográfica.

Difusión Masiva (*Broadcasting*)—Cuando se desea enviar el paquete a todos los nodos de la red.

Si el lector se encuentra interesado en este tema de búsqueda de rutas se sugiere consultar [44, 45, 46, 47] y las referencias ahí listadas, además de las otras citas mencionadas en esta sección.

Si se tuviera que clasificar el trabajo presentado en esta tesis, dentro de alguna de estas líneas de investigación, la mejor opción sería colocarlo dentro de búsqueda de rutas, puesto que otra técnica para esto es la de buscar rutas en un conjunto particular de nodos, que tiene la característica de que a través del mismo se puede llegar a cualquier nodo de la red, véase [48, 49, 50] para darse una amplia idea al respecto.

### Ahorro de energía

Como se mencionó antes, al describir las características de una red ad hoc, los componentes móviles dependen de baterías para el suministro de energía. Es por ello que en este tipo de redes es de vital importancia que los algoritmos utilicen la menor cantidad de energía posible, para alargar así el tiempo de vida.

La importancia de este tema ha producido gran cantidad de investigación. Se han propuesto algoritmos y protocolos *conscientes de la conservación de la energía* de los nodos (*power-aware*). La mayoría de estos algoritmos tratan de conservar la energía en una cierta capa de la red, las distintas capas se presentan en la Figura 1.3. Las técnicas incluyen[51]:

- Capa Física:
  - Uso de antenas direccionales [52, 53].
  - Control de poder de transmisión [54, 55].
- Capa de Liga:
  - Evitar retransmisiones innecesarias y colisiones en el acceso al medio [56, 57].
  - Los nodos pueden entrar en estado de alerta (*standby*) o de transmisión según convenga [58, 57].
- Capa de Red:
  - Considerar la carga en el link debido a las retransmisiones [59].
  - Considerar la vida de la batería al elegir una ruta [60, 61, 62].
- Capa de Transporte:
  - Apagar el canal de comunicación durante periodos de tiempo [63].

De entre las técnicas utilizadas, las más comunes y eficientes son “apagar los nodos” [64, 65, 66], “búsqueda de rutas conscientes de la energía” (*power-aware routing*) [61, 67] y “fijar horarios” (*scheduling*) [68].

### Calidad de servicio (*QoS*)

El RFC 2386 [70] define la Calidad de Servicio (*Quality of Service (QoS)* en inglés) como: “un conjunto de requisitos de servicio que la red debe cumplir al transportar un paquete de un nodo origen a uno destino”. Intrínseca a esta noción hay una garantía que proporciona la red de satisfacer al usuario, es decir de proveer un servicio con ciertas garantías preespecificadas. Así que suficientes recursos deben estar disponibles para satisfacer dicha garantía.

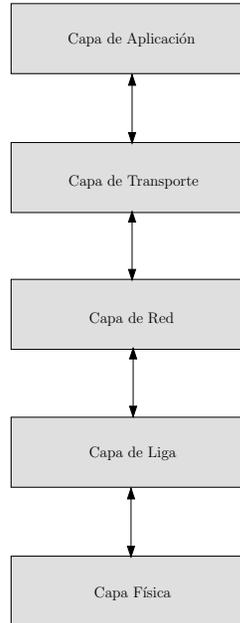


Figura 1.3: En la figura se presenta una pila de las diferentes capas de una red ad hoc. A diferencia de las redes tradicionales, estas capas responden a variaciones locales e información de otras capas, lo que se conoce como el diseño de *cruce de capas* (*cross-layer*) [69].

Gran cantidad de trabajo se ha hecho para tratar de garantizar QoS en la Internet y en otras redes (véase por ejemplo [71, 72, 73]). Sin embargo, ninguno de estos trabajos puede aplicarse en MANETs. Para garantizar QoS información como retrasos, ancho de banda, radio de errores, radio de pérdida de paquetes, etcétera, debe estar disponible. Sin embargo, poseer y manejar esta información en redes ad hoc es muy difícil.

El proveer calidad de servicio en MANETs no es tarea exclusiva de una sola capa, sino por el contrario, requiere de los esfuerzos coordinados de todas estas. En la literatura la investigación sobre QoS incluye: Modelos de QoS, QoS reservación de servicios, QoS en la búsqueda de rutas y QoS en MAC.

Un modelo de calidad de servicio especifica una arquitectura en la cual algunos servicios pueden ser facilitados. Es decir, un modelo QoS es un mecanismo para alcanzar la calidad de servicio. Garantizar QoS en Internet ha sido propuesto usando dos modelos diferentes: El modelo *Integrated Services (IntServ)* [74] y el modelo *Differentiated Services (DiffServ)* [75]. A pesar de que estos modelos han sido diseñados para redes alámbricas, se han hecho algunos intentos para tratar de adaptarlos a redes inalámbricas [76] y en particular a redes ad hoc [77].

QoS reservación de servicios es el proceso de establecer una conexión del

nodo origen al nodo destino, que implica la reservación de recursos en los nodos intermedios del camino. QoS reservación de servicios actúa como el centro de control en el soporte de calidad de servicio [78]. Este mecanismo reserva y libera recursos, establece, termina y renegocia los flujos en la red.

La tarea de la calidad de servicio en la búsqueda de rutas, es encontrar un camino entre un nodo origen y uno destino, que tenga los recursos necesarios para garantizar las exigencias del servicio deseado. Algunos autores separan QoS en la búsqueda de rutas, y QoS reservación de servicios como dos tareas distintas [78], mientras que otros las definen como la misma tarea [79]. Es decir, señalan como parte de la tarea de QoS en la búsqueda de rutas encontrar un camino con suficientes recursos, y hacer la reservación de los mismos.

*Sequential Assigment Routing* (SAR) propuesto en [80], es uno de los primeros protocolos para búsqueda de rutas en redes ah hoc que introducen la noción de QoS en la decisión de rutas. Decidir una ruta en SAR se basa en tres factores: los recursos de energía, QoS en cada camino, y el nivel de prioridad de cada paquete. Existen muchos otros artículos en la literatura que proponen algoritmos de búsqueda de rutas que soportan calidad de servicio ([81, 82, 83, 84], y muchas otras).

Finalmente, QoS en el control de acceso al medio, es el componente que coordina y del que depende el funcionamiento de QoS en la búsqueda de rutas y QoS reservación de servicios.

Aquí se han discutido sólo de forma muy breve algunas líneas de investigación sobre redes ad hoc móviles, sin embargo, mucho más queda por decir sobre el tema. Se puede consultar [51], y las referencias ahí listadas, para tener una idea un poco mas amplia de estas y otras líneas de investigación en MANETS.

# Capítulo 2

## Antecedentes

En este capítulo se introducirán algunos conceptos que son básicos para una mejor comprensión del resto de la tesis. En las primeras cuatro secciones se presentan nociones sobre teoría de gráficas, y en la quinta y última algunas ideas básicas sobre polígonos.

### 2.1. Gráficas y subgráficas

La mayoría de los conceptos que se presentarán en este capítulo son de teoría de gráficas, por tanto es indispensable comenzar definiendo lo que es una gráfica.

**Definición 1.** *Una gráfica  $G$  está formada por dos conjuntos: un conjunto finito no vacío  $V$ , y un conjunto  $E$  (no necesariamente vacío y disjunto de  $V$ ). A su vez  $E$  está formado por subconjuntos de dos elementos distintos de  $V$ . Cada elemento de  $V$  se conoce como vértice y  $V$  como el conjunto de vértices de  $G$ . Los elementos de  $E$  se conocen como aristas y  $E$  como el conjunto de aristas.*

En general, el conjunto de vértices y el conjunto de aristas de una gráfica  $G$  se representan por  $V(G)$  y  $E(G)$ , respectivamente.

Se dice que una arista  $e = \{u, v\}$  une los vértices  $u$  y  $v$ . Además se dice que  $u$  y  $v$  son *vértices adyacentes*, mientras que  $u$  y  $e$  son *incidentes*, así como  $v$  y  $e$ . Además si  $e_1$  y  $e_2$  son aristas distintas de  $G$  que inciden en un vértice en común, entonces  $e_1$  y  $e_2$  son *aristas adyacentes*. Muchas veces una arista  $e = \{u, v\}$  es denotada también como  $e = uv$ . Una gráfica  $G$  con  $p$  vértices en la que cada par de sus vértices son adyacentes se conoce como *gráfica completa*, y generalmente se denota como  $K_p$ .

Dada una gráfica  $G$  es conveniente expresarla usando un diagrama. Un *dibujo* de  $G$  es una representación de  $G$  en el plano, de tal forma que cada vértice está representado por un punto distinto (que generalmente se representa por un círculo pequeño) y cada arista por un arco simple (que

no se auto-intersecta) y continuo que conecta los dos puntos correspondientes a los vértices de una arista. En la Figura 2.1 se muestra una gráfica con conjunto de vértices  $V = \{v_1, v_2, v_3, v_4, v_5\}$  y conjunto de aristas  $E = \{v_1v_4, v_1v_5, v_2v_3, v_2v_4, v_2v_5, v_3v_4, v_3v_5\}$ .

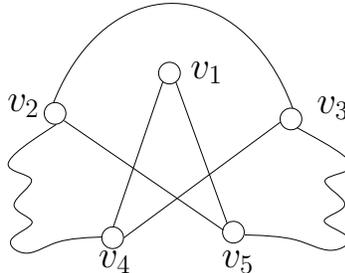


Figura 2.1: Una gráfica.

Para una gráfica  $G$ ,  $p = |V(G)|$  se conoce como el *orden* de  $G$  y  $q = |E(G)|$  como el *tamaño* de  $G$ , es decir una gráfica  $G = (p, q)$  es una gráfica de orden  $p$  y tamaño  $q$ . El *grado de un vértice*  $v$  en  $G$  es el número de aristas de  $G$  incidentes en  $v$ . El grado de un vértice  $v$  en  $G$  será denotado como  $deg_G(v)$ , si se entiende sobre que gráfica se está hablando entonces se omitirá el subíndice  $G$ . Un vértice con grado 0 se conoce como *vértice aislado*.

Si a cada arista  $e$  de una gráfica se le asocia una etiqueta numérica  $w(e)$ , la gráfica es llamada *gráfica con pesos*, y la etiqueta  $w(e)$  se le llama el *peso* de  $e$ . Los pesos de las aristas pueden ser números enteros o racionales, los cuales pueden representar conceptos como distancias, costo de conexión, etc. Un ejemplo de una gráfica con pesos se da en la Figura 2.2.

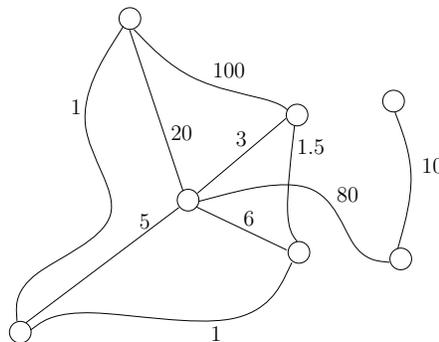


Figura 2.2: Gráfica con pesos. Nótese que los pesos en las aristas podrían representar cualquier cosa, distancias entre ciudades, tiempo de recorrido, etc.

Muchas veces se presenta el caso en el que una gráfica de interés esta

contenida en alguna otra más grande. Esto se define formalmente a continuación.

**Definición 2.** Una gráfica  $H$  es una subgráfica de una gráfica  $G$  si  $V(H) \subseteq V(G)$  y  $E(H) \subseteq E(G)$ . Si  $H$  es una subgráfica de  $G$  escribimos  $H \subset G$ .

Si una subgráfica  $H$  de una gráfica  $G$  tiene el mismo orden que  $G$ , entonces  $H$  es llamada *subgráfica generadora* de  $G$ .

Si  $U$  es un subconjunto no vacío de  $V(G)$ , entonces la subgráfica  $\langle U \rangle$  de  $G$  *inducida* por  $U$  es la gráfica que tiene a  $U$  como conjunto de vértices y cuyo conjunto de aristas consiste de aquellas aristas de  $G$  incidentes con dos elementos de  $U$ . La Figura 2.3 muestra la gráfica inducida  $\langle U \rangle$  de  $G$ , en donde  $U = \{v_1, v_4, v_5\}$ .

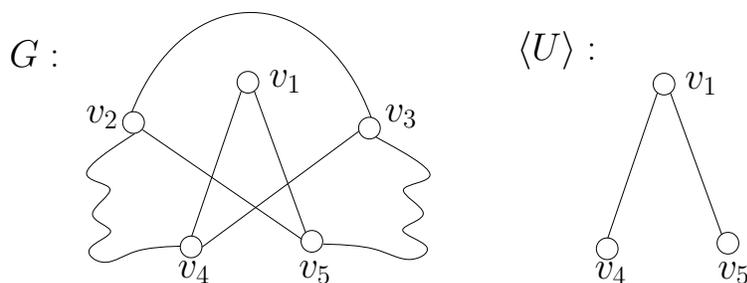


Figura 2.3: Gráfica inducida.

## 2.2. Gráficas conexas y desconexas

Una característica muy importante de algunas gráficas es la de ser conexas, es decir la de no poseer vértices ni “componentes” aislados. En esta sección se define formalmente la noción de conexidad en gráficas, para hacerlo es necesario presentar antes algunos conceptos.

Sean  $u$  y  $v$  dos vértices de una gráfica  $G$ . Un  $u - v$  recorrido de  $G$  es una secuencia alternante y finita de vértices y aristas de  $G$ , la cual comienza en  $u$  y termina en  $v$ , de tal forma que cada arista es inmediatamente precedida y sucedida por los vértices con los que es incidente. Un *recorrido trivial* es aquel que no contiene aristas. Nótese que en un recorrido las aristas y los vértices se pueden repetir. A menudo solamente los vértices de un recorrido son listados, ya que es obvio cuales son las aristas correspondientes.

Un  $u - v$  recorrido es *cerrado* si  $u = v$  y es *abierto* de otro modo. A un  $u - v$  recorrido en el cual todas las aristas son distintas se le llama  $u - v$  *trayectoria*. Un  $u - v$  *camino* es un  $u - v$  recorrido en el cual no se repite ningún vértice; un vértice  $u$  forma un  $u - u$  *camino trivial*. Por lo tanto, cada camino es una trayectoria.

Una trayectoria no trivial cerrada de  $G$  es llamada *circuito* de  $G$ . En una gráfica  $G$  un circuito  $v_1, v_2, v_3, \dots, v_n, v_1$  ( $n \geq 3$ ) en el cual todos sus  $n$  vértices  $v_i$  son distintos es llamado *ciclo* de  $G$ . En la Figura 2.4  $C1 : v_1, v_2, v_3, v_4, v_1$  es un ciclo, mientras que  $C2 : v_1, v_2, v_5, v_3, v_4, v_5, v_1$  es un circuito que no es un ciclo. A  $C1$  se le llama 4-ciclo.

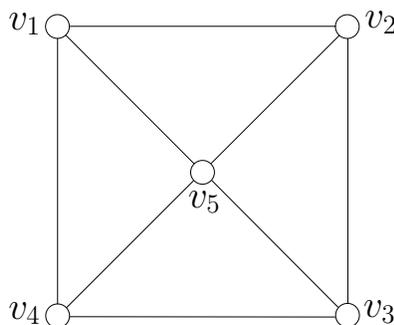


Figura 2.4: Un ciclo y un circuito en una gráfica.

Una vez que se han presentado estos conceptos se puede definir la conectividad.

**Definición 3.** *Un vértice  $u$  está conectado con un vértice  $v$  en una gráfica  $G$  si existe un  $u-v$  camino en  $G$ . Una gráfica  $G$  es conexa si para cualquier par de vértices de  $G$  existe un camino que los une. Una gráfica es desconexa si no es conexa.*

Un *componente* de una gráfica  $G$  es una subgráfica conexa que no está propiamente contenida en ninguna otra subgráfica de  $G$ . Así que una gráfica es conexa si está formada por un solo componente.

Para una gráfica conexa  $G$ , se define la *distancia*  $dist(u, v)$  entre dos vértices  $u$  y  $v$  como la longitud mínima de los  $u-v$  caminos de  $G$ .

Un tipo especial de gráficas conexas son las gráficas conocidas como árboles. Un *árbol* es una gráfica acíclica y conexa. Un teorema muy conocido sobre árboles, que en muchas ocasiones se toma como una definición, es el siguiente.

**Teorema 1.** *Un gráfica  $G = (p, q)$  es un árbol si y sólo si es acíclica y  $p = q + 1$ .*

Para una demostración básica véase [85].

Al igual que para una gráfica, se dice que un árbol  $T$  es un árbol generador de una gráfica  $G$ , si  $T \subset G$  y  $T$  tiene todos los vértices de  $G$ . Un árbol puede también tener pesos en las aristas.

**Definición 4.** *Un árbol generador de peso mínimo (AGM) de una gráfica  $G$  con pesos en las aristas  $w(e)$ , es un árbol generador que minimiza la suma  $\sum_{e \in E(G)} w(e)$ .*

### 2.3. Tres tipos importantes de gráficas

En esta sección se definirán tres tipos de gráficas que serán de gran importancia para el entendimiento de esta tesis: las gráficas geométricas, las gráficas de disco unitario y las gráficas aplanables.

Antes de presentar una definición formal de las gráficas geométricas es necesario dar algunos conceptos. Dos aristas se intersectan si tienen un punto interior en común, este punto es llamado *cruce*. Un cruce  $p$  es *propio* si en una vecindad pequeña de  $p$  una arista pasa de un lado al otro de la arista que cruza. Entonces en un dibujo:

1. Ninguna arista pasa por un vértice que no sea uno de sus extremos.
2. Si dos aristas se cruzan, lo hacen propiamente.
3. Tres aristas nunca se cruzan en el mismo punto.

Un dibujo en el cual cada arista es representada por un segmento de línea recta es llamado *dibujo de línea recta* o bien *gráfica geométrica*. Siempre se supone que los vértices de una gráfica geométrica están en posición general, es decir que no hay tres vértices colineales.

Un tipo de gráficas geométricas son las llamadas *gráficas de disco unitario*.

**Definición 5.** Sea  $P_n$  un conjunto de puntos en posición general en el plano. La gráfica de disco unitario  $GDU(P_n)$ , asociada a  $P_n$  es una gráfica geométrica cuyo conjunto de vértices son los elementos de  $P_n$ , dos de los cuales son conectados por una arista si la distancia Euclídeana entre ellos es menor o igual a uno.

Se puede pensar en una gráfica de disco unitario de la siguiente forma. Sea  $C_v$  el círculo de radio 1 centrado en  $v \in V(G)$ , para una gráfica  $G$ . Supóngase que  $v$  es un dispositivo capaz de comunicarse con los vértices de  $G$  que se encuentren dentro de  $C_v$ . Entonces, si  $v$  puede comunicarse con algún vértice  $u$ , se indica en el dibujo agregando una arista entre ellos. La Figura 2.5 muestra esta idea.

Un vértice  $u$  está en la  $k$ -vecindad de  $v$  si  $dist(u, v) \leq k$ . La  $1$ -vecindad o vecindad de  $v$  es el conjunto de vértices que son vecinos de este. Denotaremos la  $k$ -vecindad de un vértice por  $\mathcal{N}_k(v)$ .

Por último es conveniente hablar de las *gráficas aplanables*.

**Definición 6.** Una gráfica es aplanable si puede ser dibujada en el plano de tal forma que las aristas de  $G$  se crucen solamente en sus vértices.

Dicho de otra manera, una gráfica  $G$  de orden  $p$  y tamaño  $q$  es aplanable si puede ser empotrada en el plano, es decir si es posible distinguir una

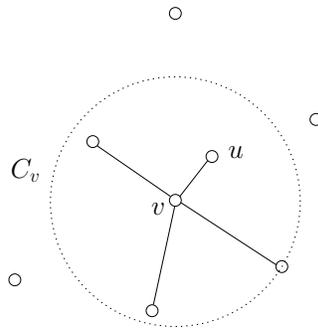


Figura 2.5: Disco unitario centrado en un vértice.

colección de  $p$  puntos distintos en el plano que correspondan a los vértices de  $G$  y una colección de  $q$  curvas en el plano, tal que el único punto en común posible entre cualquier par de estas es un punto final, que correspondan a las aristas de  $G$ , de tal forma que si una curva  $A$  corresponde a la arista  $e = uv$ , entonces sólo los puntos finales de  $A$  corresponden a vértices de  $G$ , es decir a  $u$  y  $v$ .

Si una gráfica aplanable es empotrada en el plano se dice que la gráfica es *plana*. En la Figura 2.6 la gráfica  $G_1$  es aplanable, aunque su dibujo no es plano. La gráfica  $G_2$  es aplanable y plana. Y la gráfica  $G_3$  no es aplanable.

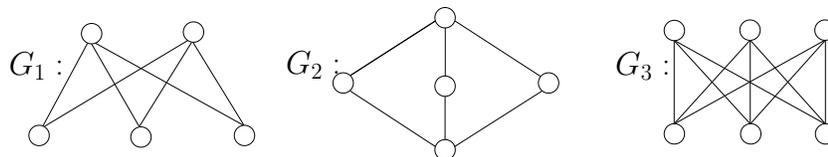


Figura 2.6: Gráfica aplanable, plana y no aplanable.

Para una gráfica plana  $G$ , una *cara* o *región* de  $G$  es una porción maximal del plano en la cual cualquier par de puntos pueden unirse por una curva  $A$ , de tal forma que cada punto de  $A$  no corresponde a un vértice de  $G$ , ni pertenece a una curva que sea una arista de  $G$ .

Intuitivamente, las caras de  $G$  son las porciones conectadas de plano que permanecen después de eliminar todas las curvas y los puntos que correspondan a las aristas y a los vértices de  $G$ . La *frontera* de una cara  $f$  consiste de todos aquellos puntos  $x$  que corresponden a los vértices y aristas de  $G$  que tienen la propiedad de que  $x$  puede unirse con un punto de  $f$  por una curva, en la cual todos sus puntos diferentes de  $x$  corresponden a  $f$ . Toda gráfica plana tiene una región no acotada llamada la *cara exterior*. Se dice que un vértice  $v$  es incidente con una cara  $f$ , si  $v$  pertenece a la frontera de  $f$ .

La gráfica de la Figura 2.7 tiene tres regiones, la frontera de cada una de estas es un 4-ciclo. La frontera de una de esas regiones son los vértices  $\{v_1, v_5, v_3, v_4\}$  y las aristas  $\{v_1v_5, v_5v_3, v_3v_4, v_4v_1\}$ . La frontera de la cara exterior la constituyen los vértices  $\{v_1, v_2, v_3, v_4\}$  y las aristas  $\{v_1v_2, v_2v_3, v_3v_4, v_4v_1\}$ .

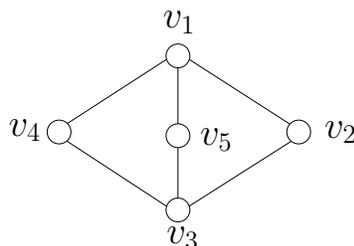


Figura 2.7: Gráfica con tres regiones.

El orden, el tamaño y el número de regiones de una gráfica plana conexa están relacionados por la fórmula de Euler, la cual lleva ese nombre en honor a su descubridor Leonhard Euler [86]. Esta fórmula se presenta en el siguiente teorema, una demostración puede consultarse en [85].

**Teorema 2.** *Si  $G$  es una gráfica plana conexa con  $p$  vértices,  $q$  aristas y  $r$  regiones, entonces*

$$p - q + r = 2 \quad (2.1)$$

Del teorema anterior se sigue que cualquiera dos empotramientos de una gráfica conexa aplanable, resultan en gráficas planas con el mismo número de regiones.

Una gráfica plana  $G = (p, q)$  es llamada *aplanable maximal* si para cualquier par de vértices no adyacentes  $u$  y  $v$  de  $G$ , la gráfica  $G + uv$  no es aplanable. Entonces en un empotramiento de una gráfica aplanable maximal con  $p \geq 3$ , la frontera de cada región de  $G$  es un triángulo. Por esta razón, las gráficas aplanables maximales se conocen también como *gráficas aplanables trianguladas*, a las gráficas planas maximales se les llama *triangulaciones*.

## 2.4. Conjunto independiente y conjunto dominante

Para finalizar de presentar los antecedentes de teoría de gráficas que son importantes para este trabajo, es necesario mencionar los conceptos de conjunto dominante y de conjunto independiente. Es importante enfatizar que estos dos conceptos, al igual que los anteriores, deben quedarle muy

claros al lector antes de seguir adelante, ya que su importancia es vital para el entendimiento del material.

Los juegos de tablero proveen ilustraciones sencillas sobre dominación. Por ejemplo, un tablero de ajedrez de  $8 \times 8$  puede ser representado por una gráfica con 64 vértices. Una arista  $uv$  implica que piezas similares colocadas en los cuadrados  $u$  y  $v$ , podrían retarse una a la otra. Se dice que cualquier vértice adyacente a  $v$  es *dominado* por este, mientras que cualquier otro vértice es *independiente* de  $v$ .

**Definición 7.** Sea  $G$  una gráfica, y sea  $D \subseteq V(G)$ .  $D$  es un conjunto dominante de  $G$  si  $\forall v \in V(G) \exists (v, u) \in E(G)$ , para algún  $u \in D$ .

Es decir que  $D$  es un conjunto dominante de  $G$  si para cualquier vértice  $v$  de  $G$ ,  $v$  está en  $D$  o  $v$  es vecino de algún vértice en  $D$ .

Un conjunto dominante  $D$  es *minimal* si este no contiene ningún subconjunto propio que sea también un conjunto dominante. La cardinalidad del conjunto dominante minimal de menor tamaño se conoce como el *número de dominación* de  $G$ , al conjunto dominante con esta cardinalidad se le llama *conjunto dominante mínimo*. Se dice que los vértices del conjunto dominante *cubren* al resto de estos. Un conjunto dominante  $D$  de  $G$  es *conexo* si la subgráfica inducida  $\langle D \rangle$  es conexa. Para la gráfica  $K_5$  en la Figura 2.8, el número de dominación es 1.

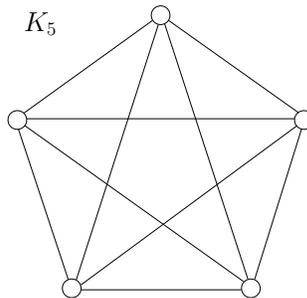


Figura 2.8: Gráfica en la que un vértice es suficiente para cubrir a todos los demás.

Una vez definido lo que es un conjunto dominante es más fácil entender el concepto de independencia.

**Definición 8.** Sea  $G$  una gráfica, y sea  $I \subseteq V(G)$ .  $I$  es un conjunto independiente de  $G$  si no hay ningún par de vértices en  $I$  que sean adyacentes.

Un conjunto independiente  $I$  es *maximal* si cualquier vértice que no este en  $I$  es dominado por al menos un vértice en  $I$ . El *número de independencia* de  $G$ , es la cardinalidad de un conjunto independiente maximal de  $G$ . En la Figura 2.8, el número de independencia de  $K_5 = 1$ .

## 2.5. Otros conceptos

Una vez introducidos los conceptos sobre teoría de gráficas, se deben mencionar algunas nociones básicas sobre polígonos para concluir con los antecedentes necesarios.

**Definición 9.** *Un polígono es la región de un plano acotada por una colección finita de segmentos de línea que forman una curva cerrada simple.*

Otra definición más simple de polígono es la que sigue. Sean  $v_1, v_2, v_3, \dots, v_n$ ,  $n$  puntos en el plano. Aquí y en el resto del documento la aritmética en los índices será módulo  $n$ , implicando un orden cíclico de los puntos, es decir que  $v_1$  sigue a  $v_n$ . Sean  $e_1 = v_1v_2, e_2 = v_2v_3, \dots, e_i = v_iv_{i+1}, \dots, e_n = v_nv_1$ ,  $n$  segmentos de recta conectando a los puntos. Estos segmentos acotan un polígono si y sólo si:

1. La intersección de cada par de segmentos adyacentes, en el orden cíclico, es el único punto en común de estos:  $e_i \cap e_{i+1} = v_{i+1}$ , para toda  $i = 1, 2, 3, \dots, n$ .
2. Segmentos no adyacentes no se intersectan:  $e_i \cap e_j = \emptyset$ , para toda  $j \neq i + 1$ .

La razón por la que estos segmentos definen una curva es porque sus finales están conectados; la razón por la que la curva es cerrada es porque forman un ciclo; la razón por la que la curva cerrada es simple es porque segmentos no adyacentes no se intersectan. A menudo, objetos que cumplen sólo con la condición (1) son llamados polígonos, mientras que objetos que cumplen con ambas condiciones son llamados *polígonos simples*, véase la Figura 2.9. Durante este trabajo se llamará polígono a aquel objeto que cumpla con ambas condiciones.

Al igual que en una gráfica, los puntos  $v_i$  se conocen como vértices, y los segmentos  $e_i$  como aristas. Nótese que un polígono con  $n$  vértices tiene  $n$  aristas.

Los polígonos pueden dividirse en dos grupos: polígonos convexos y polígonos con un vértice concavo. Un vértice es llamado *concavo* si su ángulo interno es estrictamente mayor que  $\pi$ ; de otro modo el vértice es llamado *convexo*. Un *polígono es convexo* si todos sus vértices lo son, la Figura 2.9(a) muestra un polígono no convexo. Existen muchas otras definiciones de convexidad, una definición primaria es la siguiente.

**Definición 10.** *Un conjunto de puntos  $S$  es convexo si  $x \in S$  y  $y \in S$  implica que el segmento de curva cerrada  $xy \subseteq S$ .*

Nótese que esta definición no especifica ninguna dimensión en particular para los puntos, ni tampoco si  $S$  es conexo, acotado o no, ni tampoco si es abierto o cerrado. Esta definición es útil para presentar la de *cierre convexo*.

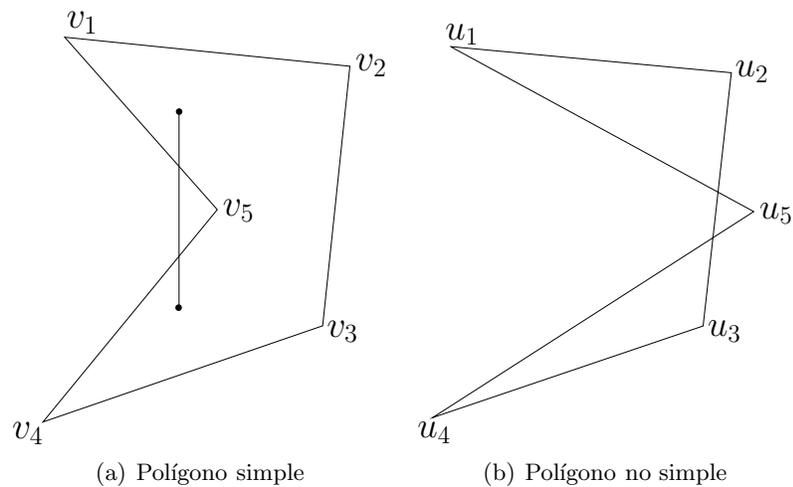


Figura 2.9: Dos polígonos con seis vértices.

**Definición 11.** *El cierre convexo de un conjunto de puntos  $S$  en el plano, es el polígono mas pequeño  $P$  que encierra  $S$ . Mas pequeño en el sentido de que no existe ningún otro polígono  $P'$  tal que  $P \supset P' \supseteq S$ .*

La Figura 2.10 muestra el cierre convexo de un conjunto de puntos.

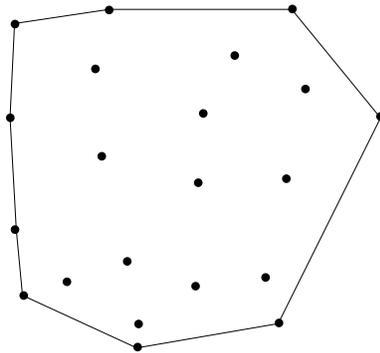


Figura 2.10: Cierre convexo de un conjunto de puntos en el plano.

Otro tipo de polígonos son los llamados *polígonos regulares* los cuales son aquellos que tienen todos sus lados y sus ángulos iguales. Ejemplos de polígonos regulares son, triángulos equiláteros, cuadrados, pentágonos regulares, hexágonos regulares, etc. Claramente existe un polígono regular con  $n$  lados para cada entero  $n \geq 3$ .

Usando solamente polígonos se puede *recubrir* o *subdividir* un plano.

**Definición 12.** *Una subdivisión del plano es una partición de este en regiones, de modo que la unión de estas es el plano, y su intersección es el*

*conjunto de aristas que forman la frontera de las regiones.*

Esta definición implica que dos regiones no pueden superponerse ni dejar huecos sin recubrir. La Figura 2.11 muestra una subdivisión del plano en regiones cuadradas.

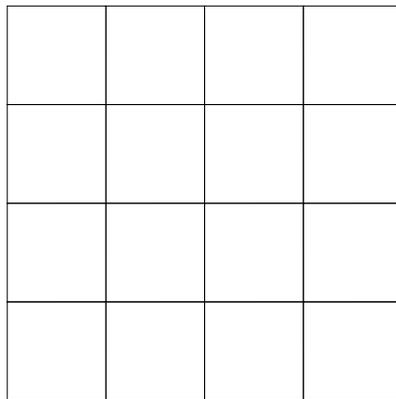


Figura 2.11: Subdivisión del plano en regiones cuadradas.

Una subdivisión del plano en la que todas las piezas tienen la misma forma y estas son polígonos regulares, se conoce como *subdivisión regular*. Existen sólo tres polígonos regulares con los que se puede recubrir el plano: los triángulos equiláteros, los cuadrados y los hexágonos regulares. Nótese que en una subdivisión del plano las regiones no necesariamente deben tener la misma forma, véase la Figura 2.12. A las subdivisiones del plano se les conoce también como *mosaicos*.

Con esto se concluyen los antecedentes y se espera que el trabajo sea autocontenido, en el sentido de que estos conceptos sean suficientes para entender todas y cada una de las ideas presentadas en este documento.

La organización de los capítulos consecuentes es como sigue. En el capítulo 3 se describen dos importantes artículos sobre algoritmos locales en redes ad hoc. La importancia de estos artículos radica en el hecho de que en estos se introduce por primera vez el concepto de localidad. En el capítulo 4 se presenta el problema a resolver, los resultados y conclusiones. Al final se puede encontrar una conclusión general del trabajo y posibles mejoras al mismo.

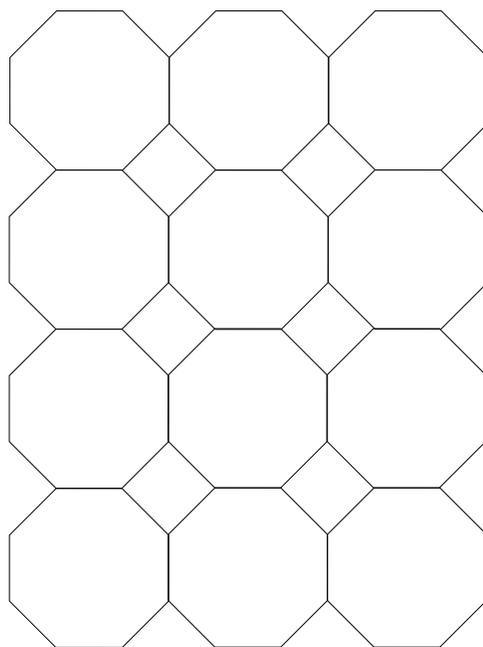


Figura 2.12: Mosaico formado con dos polígonos distintos.

## Capítulo 3

# Algoritmos locales

Como se ha mencionado antes, el objetivo de esta tesis es desarrollar un algoritmo *local* para encontrar un conjunto dominante en una red ad hoc, de tal forma que el tamaño de dicho conjunto sea solamente un número constante de veces el número de dominación de la gráfica asociada con la red. Sin embargo, hasta ahora no se ha definido formalmente qué es un algoritmo local. En este capítulo se introduce la noción de localidad en algoritmos, y se presenta un resumen sobre los dos primeros artículos que introdujeron la noción de localidad en algoritmos para gráficas geométricas.

En 1999, Kranakis, Singh y Urrutia en [87] introducen el concepto de algoritmo local para búsqueda de rutas en redes geométricas. El objetivo del algoritmo introducido es el de encontrar una ruta que lleve de un vértice origen  $s$  hasta un vértice destino  $t$ , en una red geométrica. Empezando en  $s$ , el algoritmo “se mueve” de un vértice a otro hasta encontrar el destino.

Según Kranakis et. al., un algoritmo con estas características es local si satisface lo siguiente:

1. En cada momento en el tiempo se conocen sólo las coordenadas del punto origen y las del punto destino. Se tiene disponible una cantidad finita de memoria en donde se puede almacenar un número constante de identificadores de los vértices de la red. Nótese que estas dos condiciones implican que en ningún momento se tiene conocimiento de la topología de la red entera.
2. Cuando se llega a un vértice  $v$ , se puede utilizar información local guardada en éste, es decir la información sobre sus vecinos y sobre las aristas incidentes a  $v$ .
3. No está permitido cambiar la información local almacenada en  $v$ . Nótese que en particular una vez que se ha dejado un vértice, si se regresa a él no será posible identificar que ya se ha visitado con anterioridad, al menos que su identificador siga en la memoria, pero recuérdese que sólo se puede almacenar un número constante de identificadores.

La última condición implica que no se dejan marcas en los vértices de la red que han sido visitados. Los autores del artículo refieren que la motivación para esta condición surge naturalmente en el envío de información entre distintos vértices de una red, ya que dejar marcas cargaría a los vértices de la red con una cantidad enorme de información innecesaria.

Así pues, se entiende por un *algoritmo local* en una gráfica geométrica aquel que cuenta solo con memoria finita, no deja marcas en la red y utiliza solamente la información local de un vértice para resolver un problema global de la gráfica. Por información local de un vértice se entiende información sobre su identidad y sobre la identidad de sus vecinos. El identificador más comúnmente utilizado para un vértice es un vector que contiene sus coordenadas geográficas, las cuales pueden obtenerse usando tecnología como GPS, véase la sección 1.2.4. Esta definición de localidad puede extenderse así: se dice que un algoritmo es *k-local* si utiliza la información de los vértices en la *k*-vecindad de alguno.

Es conveniente destacar la importancia de los algoritmos locales para redes ad hoc. Como se mencionó en el capítulo 1.2.4, los cambios en la topología de las MANETs son frecuentes y no estructurados, y los nodos no necesariamente tienen conocimiento de la misma. Debido a esto, un algoritmo de tipo local posee grandes ventajas en comparación con un algoritmo no local, ya que el primero no necesita conocer la topología de la red entera, y los cambios que sucedan en la misma le afectarán a un nodo solo si pasan en su vecindad.

En las próximas secciones se describen algoritmos locales que resuelven distintos problemas en gráficas geométricas.

### 3.1. Búsqueda de rutas con brújula

Como se mencionó antes, Kranakis et. al. fueron los primeros en introducir el concepto de localidad en algoritmos para gráficas geométricas, esto lo hicieron en el artículo titulado “*Compass Routing on Geometric Networks*” [87]. En ese artículo presentan un algoritmo local para resolver el siguiente problema. Supóngase que se desea viajar en una gráfica geométrica de un vértice origen  $s$  hacia un vértice destino  $t$ , y que en cualquier momento del viaje la única información disponible son las coordenadas geográficas del destino y de la posición actual, además de la dirección de las aristas incidentes con el vértice en el que se está actualmente. Bajo estas circunstancias, ¿se puede garantizar que siempre se llega de  $s$  a  $t$ ?

El problema se resuelve de la siguiente forma. Sea  $v$  el vértice actual, al principio  $v = s$ . Elegir y recorrer la arista  $vu$  de la gráfica, tal que  $vu$  tiene la pendiente más cercana a la del segmento de línea que une a  $v$  con  $t$ . Los empates se resuelve aleatoriamente. Este procedimiento se conoce como “*compass routing*” (búsqueda de rutas con brújula). En la Figura 3.1 se

presenta un ejemplo del funcionamiento del algoritmo.

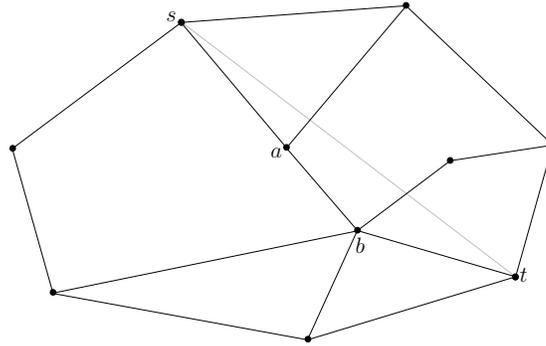


Figura 3.1: Si se desea viajar de  $s$  a  $t$  usando compass routing se procede como sigue. Estando en  $s$  se elige la arista  $sa$  puesto que su pendiente es la más cercana a la de la línea gris, se viaja a través de  $sa$  hasta llegar a  $a$ , donde se elige con el mismo criterio la arista  $ab$ , y al llegar a  $b$  finalmente se elige la arista  $bt$ .

Compass routing no garantiza siempre la llegada desde cualquier vértice  $s$  hacia cualquier vértice  $t$  de una gráfica geométrica. La Figura 3.2 muestra un caso en el que no se puede llegar de  $s$  a  $t$ , ya que al usar el algoritmo se empieza a viajar una y otra vez en el ciclo con vértices  $\{v_0, u_0, v_1, u_1, \dots, u_3, v_0\}$ .

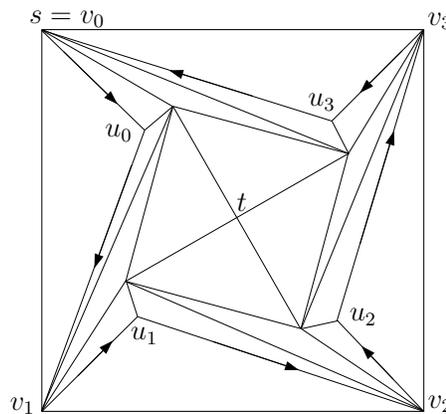


Figura 3.2: Usando compass routing no se llegará a  $t$  desde ningún  $s = v_i$ , para  $i = 0, \dots, 3$ .

Se dice que una gráfica geométrica *soporta* compass routing si para cada par de sus vértices  $s$  y  $t$ , compass routing siempre produce un  $s-t$  recorrido.

La *triangulación Delaunay*  $\mathfrak{D}(P_n)$  de un conjunto de  $n$  puntos  $P_n$  en el plano, es la partición del cierre convexo de  $P_n$  en un conjunto de triángulos con interiores disjuntos de tal forma que:

- Los vértices de estos triángulos son puntos en  $P_n$ .

- Para cada triángulo, el círculo que pasa a través de sus vértices no contiene ningún otro punto de  $P_n$  en su interior.

Es bien sabido que cuando los elementos de  $P_n$  están en *posición circular general*, es decir cuando no hay cuatro puntos en  $P_n$  que pasen por el mismo círculo,  $\mathfrak{D}(P_n)$  está bien definida. En adelante se asume que  $P_n$  está en posición circular general.

El siguiente teorema, tomado de [87] dice que compass routing garantiza que siempre se puede llegar de un vértice  $s$  a un vértice  $t$  de una gráfica, si esta es una triangulación Delaunay.

**Teorema 3.** *Sea  $P_n$  un conjunto de  $n$  puntos en el plano, entonces  $\mathfrak{D}(P_n)$  soporta compass routing.*

**Demostración.** Supóngase que se desea ir del vértice  $s$  al vértice  $t$  en  $\mathfrak{D}(P_n)$ . Se mostrará que si compass routing elige la arista  $sv$  de  $\mathfrak{D}(P_n)$ , entonces  $d(v, t) < d(s, t)$ , donde  $d(v, t)$  es la distancia Euclideana entre  $v$  y  $t$ . Como  $\mathfrak{D}(P_n)$  tiene un número finito de vértices probar esto es suficiente para demostrar que eventualmente se llegará a  $t$ . Sea  $\overline{st}$  el segmento de línea que une a  $s$  con  $t$ , y supóngase que interseca el triángulo  $\triangle(s, x, y)$ . Sea  $C$  el círculo que pasa por  $\{s, x, y\}$ . Sea  $c$  el centro de  $C$ , y sea  $s'$  el reflejo de  $s$  con respecto a la línea que une a  $c$  con  $t$ , véase la Figura 3.3. Sean  $\alpha$  y  $\beta$  los ángulos formados entre  $\overline{xs}$  y  $\overline{st}$ , y entre  $\overline{ts}$  y  $\overline{sy}$ , respectivamente. Existen dos casos:

1.  $\alpha < \beta$ . En este caso, compass routing escogerá la arista  $sx$ . Sea  $C'$  el arco abierto que une a  $s$  con  $s'$  en la dirección de las manecillas del reloj. Es fácil darse cuenta que para cualquier punto  $q$  que pertenezca al arco  $C'$ ,  $d(q, t) < d(s, t)$ ; por lo tanto  $d(x, t) < d(s, t)$ .
2.  $\beta \leq \alpha$ . En este caso, compass routing escogerá la arista  $sy$ . Sea  $p_1$  el punto de intersección entre  $\overline{st}$  y  $C$ , y sea  $p_2$  el punto en  $C$  de tal forma que  $d(p_1, p_2) = d(s, p_1)$ . Es fácil ver que  $p_2$  cae en  $C'$ . Además,  $y$  debe caer en  $C'$  puesto que  $\beta \leq \alpha$ , entonces  $d(y, t) < d(s, t)$ .

■

Basándose en lo anterior, se puede obtener un algoritmo local de búsqueda de rutas que garantice que para cualquier par de vértices  $s, t$  de una gráfica geométrica, siempre se puede encontrar un  $s - t$  recorrido. Se presentará un algoritmo que funciona para el caso en el que la gráfica es un empotramiento convexo en el plano, es decir que todas las caras de la gráfica son polígonos convexos, excepto por la cara exterior, la cual se supone es el complemento de un polígono convexo. Este algoritmo se puede extender de tal forma que funcione con gráficas geométricas arbitrarias, dicha modificación es presentada mas adelante.

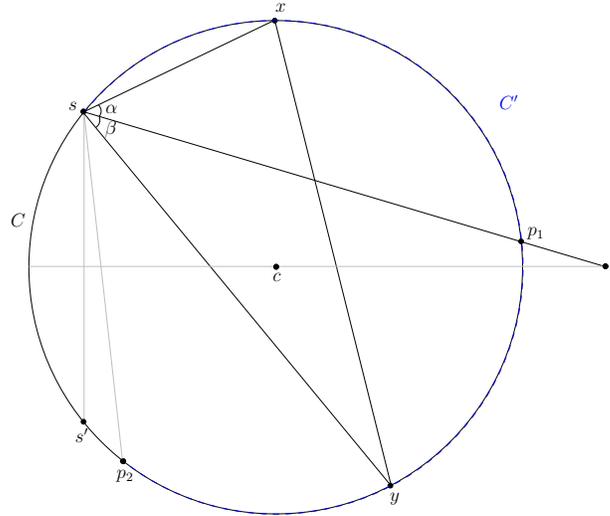


Figura 3.3: En una triangulación Delaunay, compass routing siempre elegirá la arista que minimice la distancia hacia  $t$ .

Antes de presentar el algoritmo es importante notar que una gráfica  $G$  empotrada en el plano hace una subdivisión de este en regiones llamadas caras. Si el empotramiento se hace de manera convexa, cualquier arista de  $G$  pertenece siempre a sólo dos caras de  $G$ .

#### Compass Routing II.

1. Empezando en  $v = s$  determinar cual es la cara  $f = f_0$  incidente con  $s$  que es intersectada por el segmento de recta  $\overline{st}$ . Elejir cualquiera de las dos aristas de  $f_0$  incidentes con  $v$ , y recorrer las aristas de  $f_0$  hasta encontrar una segunda arista, digamos  $uw$  en la frontera de  $f_0$  intersectada por  $\overline{st}$ .
2. En este momento actualizar  $f$  a la segunda cara de la gráfica geométrica que contiene a  $uw$  en su frontera. Recorrer las aristas de la nueva cara  $f$  hasta encontrar una segunda arista  $xy$  de  $f$ , que sea intersectada por  $\overline{st}$ . Actualizar  $f$  nuevamente como antes. Iterar este paso hasta llegar a  $t$ .

Este algoritmo garantiza encontrar un  $s - t$  recorrido en la gráfica por la siguiente razón. Sean  $f_0, f_1, \dots, f_r$  las caras intersectadas por  $\overline{st}$ . Obsérvese que inicialmente  $f = f_0$ , y que cada vez que  $f$  es actualizada cambia su valor de  $f_i$  a  $f_{i+1}$ , así que eventualmente se llegará a  $f_k$ , la cara que contiene a  $t$ , y al recorrer su frontera se alcanzará  $t$ .

La Figura 3.4 provee una ilustración de compass routing II.

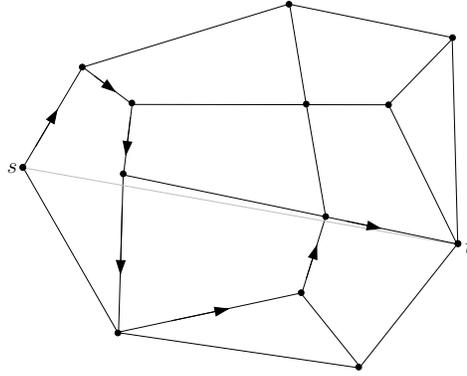


Figura 3.4: Llegar de un vértice origen  $s$  a un vértice destino  $t$  en una gráfica de caras convexas utilizando el algoritmo `compass routing II`.

Si se hace una pequeña modificación al algoritmo `compass routing II`, se puede obtener un algoritmo que garantice el encuentro de un  $s-t$  recorrido entre cualquier par de vértices  $s, t$  de una gráfica plana arbitraria  $G$ .

Dada una cara  $f$  de una gráfica  $G$ , la frontera de esta puede recorrerse en el orden inverso a las manecillas del reloj usando la *regla de la mano derecha* [88]. Esta regla puede entenderse de manera sencilla como sigue. Estando en un cuarto, si se mantiene la mano derecha pegada a la pared y se camina hacia adelante, pueden recorrerse todas las paredes del cuarto y regresar al punto de partida. Para una gráfica plana arbitraria  $G$ , las caras no necesariamente son convexas. La frontera  $\mathcal{B}_i$  de una cara  $f_i$  es una poligonal cerrada en la cual se admite que alguna arista de  $G$  aparezca dos veces. Por ejemplo, en la gráfica de la Figura 3.5 la frontera de la cara exterior es  $\{v_1, v_2, v_3, v_4, v_6, v_7, v_8, v_7, v_9, v_{10}, v_{11}, v_6, v_4, v_5, v_1\}$ . Nótese que la arista  $v_4v_6$  se lista dos veces, una de  $v_4$  a  $v_6$  y la otra de  $v_6$  a  $v_4$ .

Supóngase que se desea encontrar un  $s-t$  recorrido en una gráfica plana conexa  $G$ . Al igual que antes, trazar el segmento de recta  $\overline{st}$  que une a  $s$  con  $t$ . Empezando en  $s$  determinar cual es la cara  $f = f_0$  que es intersectada por el segmento de recta  $\overline{st}$ . Recorrer, utilizando la regla de la mano derecha, la poligonal que acota la frontera  $\mathcal{B}_0$  de  $f_0$  hasta regresar a  $s$ . Durante el recorrido cada vez que se encuentre una arista  $uv$  que sea intersectada por  $\overline{st}$  en el punto  $p_i$  se mide la distancia entre  $s$  y  $p_i$ , al encontrar una nueva arista intersectada por  $\overline{st}$  en  $p_{i+1}$ , si  $d(s, p_{i+1}) \geq d(s, p_i)$ , entonces recordar el punto  $p_{i+1}$ , de otra forma recordar  $p_i$ . Entonces, al regresar a  $s$  se recordará solamente  $p_0$ , el punto de intersección de  $\overline{st}$  con  $\mathcal{B}_0$  que maximiza la distancia a  $s$ . Una vez hecho esto, viajar por  $\mathcal{B}_0$  hasta encontrar  $p_0$ , una vez ahí actualizar  $f$  a la otra cara que contiene a  $p_0$  en su frontera, y repetir el proceso. Es fácil darse cuenta que en algún momento se llegará a  $t$ . Nótese que cada arista  $e$  de  $G$  es recorrida a lo más 4 veces, dos veces por cada cara

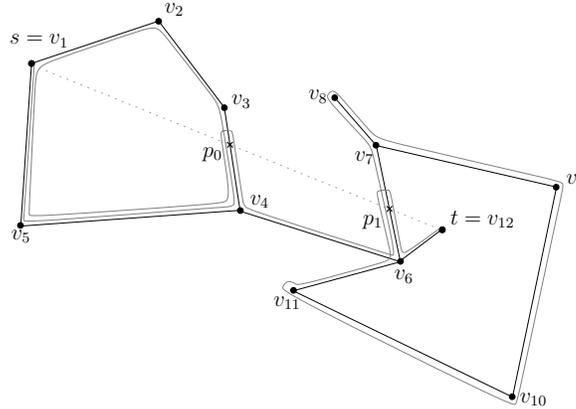


Figura 3.5: Ilustración de compass routing II en una gráfica no convexa.

a la que pertenece.

La Figura 3.5 ilustra la manera de proceder del algoritmo compass routing II en una gráfica plana conexa.

## 3.2. Búsqueda de rutas con brújula mejorada

En el artículo “*Routing with Guaranteed Delivery in Ad Hoc Wireless Networks*” [1], Bose *at. al.* presentan un algoritmo local para extraer una subgráfica plana conexa de una gráfica de disco unitario conexa. Además, presentan un algoritmo local para búsqueda de rutas que garantiza el encuentro de una ruta entre cualquier par de vértices en una MANET. Debido a que este artículo ha tenido un gran impacto en la literatura y por lo interesante de las aportaciones del mismo, se presentará aquí de manera breve.

### 3.2.1. Extracción de una subgráfica plana conexa

En esta sección se presenta un algoritmo, tomado de [1], para extraer una subgráfica plana conexa de una gráfica de disco unitario conexa. Sea  $P$  un conjunto de puntos en el plano y sea  $GDU(P)$  su gráfica de disco unitario. La única información que el algoritmo necesita en cada vértice, es la posición de cada uno de los vecinos de este en  $GDU(P)$ . El algoritmo calcula la intersección de  $GDU(P)$  con una gráfica aplanable bien conocida, la *gráfica de Gabriel*.

Una arista  $uv$  de una gráfica de un conjunto de puntos  $P$ , es una *arista de Gabriel* si el disco abierto con diámetro  $uv$  no contiene ningún punto de  $P$ . La gráfica de Gabriel  $GG(P)$  es la gráfica geométrica formada por todas las aristas de Gabriel de  $P$ . La Figura 3.6 ilustra lo que es una arista de Gabriel.

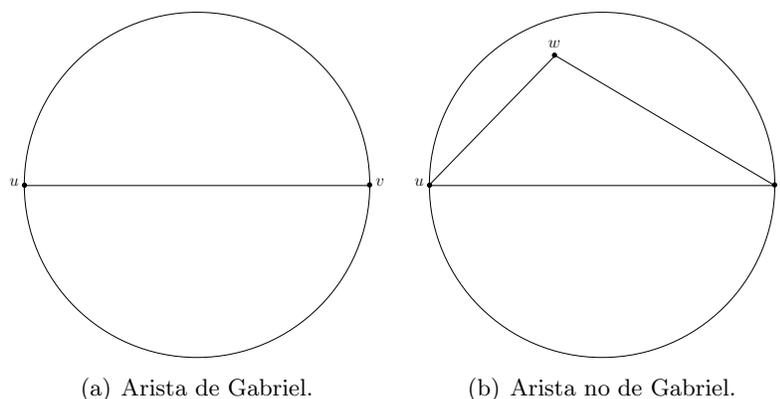


Figura 3.6:

El siguiente lema, tomado de [1], muestra que la gráfica de Gabriel es útil para extraer una subgráfica conexa. Se sugiere al lector que antes de leer la demostración del lema recuerde la definición de árbol generador de peso mínimo que se dió en la sección 2.2.

**Lema 1.** *Si  $GDU(P)$  es conexa, entonces  $GG(P) \cap GDU(P)$  es conexa.*

**Demostración.** Sea  $AGM(P)$  el árbol generador de peso mínimo de la gráfica completa con conjunto de vértices  $P$  y en la cual cada arista  $uv$  tiene el peso de la distancia Euclideana entre  $u$  y  $v$ . Es bien sabido que  $AGM(P)$  es una subgráfica de  $GG(P)$ , y por lo tanto  $GG(P)$  es conexa [89]. Entonces, para mostrar la afirmación del lema basta con probar que  $AGM(P) \subseteq GDU(P)$  si  $GDU(P)$  es conexa. Por contradicción, supóngase que  $AGM(P)$  contiene una arista  $uv$  tal que su longitud es mayor que 1. Al eliminar esta arista de  $AGM(P)$  se producirá una gráfica con dos componentes conexas,  $C_u(P)$  y  $C_v(P)$ . Como  $GDU(P)$  es conexa, entonces tiene una arista  $wx$  de longitud no mayor que uno y tal que  $w \in C_v(P)$  y  $x \in C_u(P)$ . Al reemplazar  $uv$  por  $wx$  en  $AGM(P)$  se obtiene una gráfica conexa en  $P$  con menor peso que  $AGM(P)$ , lo cual es una contradicción. ■

Gracias al lema anterior se puede extraer una subgráfica conexa de una gráfica conexa de disco unitario, sin embargo interesa que la extracción pueda hacerse de manera local. Esto se garantiza con el siguiente lema, cuya demostración fue tomada de [1].

**Lema 2.** *Sean  $u$  y  $v$  puntos en  $GDU(P)$  tales que la arista  $uv \in DGU(P)$  y  $uv \notin GG(P)$  y sea  $w$  un punto contenido en el disco con diámetro  $uv$ . Entonces  $uw \in GDU(P)$  y  $vw \in GDU(P)$ .*

**Demostración.** Sea  $m$  el punto medio de  $uv$ , véase Figura 3.7. Como  $u, v \in GDU(P)$  entonces  $d(u, v) \leq 1$ , por lo que  $d(u, m) \leq 1/2$ ,  $d(v, m) \leq 1/2$  y

$d(w, m) \leq 1/2$ . Por la desigualdad del triángulo  $d(u, w) \leq 1$ ,  $d(v, w) \leq 1$ , por lo tanto  $uw$  y  $vw$  están en  $DGU(P)$ . ■

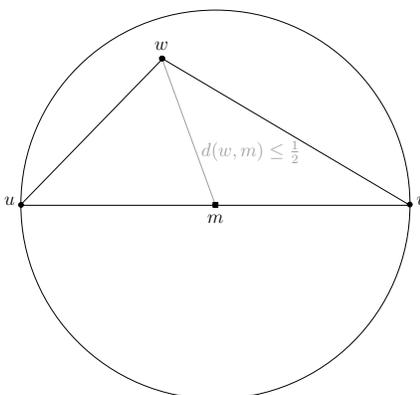


Figura 3.7: Si la arista  $uv$  de la  $GDU(P)$  no es de Gabriel, entonces  $uw$  y  $vw$  están en  $DGU(P)$ .

Entonces, se puede calcular localmente  $GDU(P) \cap GG(P)$ , ya que al llegar a un vértice  $v$  el algoritmo puede eliminar las aristas que no están en  $GDU(P) \cap GG(P)$  simplemente eliminando las aristas que no están en  $GG(\mathcal{N}_1(v) \cup \{v\})$ , recuérdese que  $\mathcal{N}_1(v)$  es la 1-vecindad de  $v$ .

Con todo lo anterior se obtiene un algoritmo distribuido, que funciona de la siguiente manera. El algoritmo es ejecutado en cada vértice  $v$  de  $GDU(P)$ . Para cada vértice  $u \in \mathcal{N}_1(v)$  checar si el disco con diámetro  $uv$  está vacío de puntos de  $\mathcal{N}_1(v)$ , es decir si  $disco(u, v) \cap \mathcal{N}_1(v) \setminus uv \neq \emptyset$ , entonces eliminar la arista  $uv$ . El Lema 1 garantiza que si dicho algoritmo es ejecutado en cada vértice de la gráfica, la gráfica resultante es conexa. Además, como  $GG(P)$  [89] es aplanable, la gráfica resultante también es aplanable. Nótese que el algoritmo descrito requiere tiempo  $O(deg(v)^2)$ , ya que para cada  $u \in \mathcal{N}_1(v)$  se debe verificar si  $disco(u, v)$  está vacío, y esto último requiere tiempo  $O(deg(v))$ . En [1] se argumenta que el tiempo puede reducirse a  $O(deg(v) \log deg(v))$ , utilizando las técnicas adecuadas.

### 3.2.2. Búsqueda de rutas en gráficas aplanables

En [1] se presentan técnicas para búsqueda de rutas, difusión masiva (broadcasting) y difusión geográfica (geocasting) en gráficas aplanables conexas. En esta sección se hace un resumen de las técnicas para búsqueda de rutas expuestas en dicho artículo.

El algoritmo para búsqueda de rutas **Compass Routing II** presentado en la sección 3.1 es retomado en el artículo [1] con el nombre **FACE-1**. El siguiente teorema, tomado de [87], resume el comportamiento del algoritmo.

**Teorema 4.** *El algoritmo **FACE-1** alcanza  $t$  después de máximo  $4|E|$  pasos, donde  $|E|$  es el número de aristas de  $G$ .*

Recuérdese que cada arista  $e$  de  $G$  en el algoritmo **FACE-1**, es recorrida a lo más 4 veces. Esto se debe a que el algoritmo primero debe recorrer toda la frontera  $\mathcal{B}_f$  de la cara actual  $f$  y después recorrer nuevamente algunas de sus aristas hasta encontrar  $p_0$ , el punto de intersección de  $\mathcal{B}_f$  con el segmento de línea  $\overline{st}$  que maximiza la distancia a  $s$ . Entonces, la cota  $4|E|$  del teorema anterior, puede reducirse a  $3|E|$  si al recorrer la frontera de la cara para encontrar  $p_0$  nuevamente, se hace por el camino mas corto de las dos posibilidades, y ya no siguiendo la regla de la mano derecha. Sin embargo, según simulaciones presentadas en [1], en la práctica el siguiente método, conocido como **FACE-2**, funciona mejor. Sea  $f$  la cara de  $G$  con  $p = s$  en su frontera  $\mathcal{B}_f$ , intersectada por el segmento de recta  $\overline{pt}$ . Recorrer  $\mathcal{B}_f$  hasta encontrar la arista  $uv$  que intersecta  $\overline{st}$  en algún punto  $p' \neq p$ . Repetir el proceso con  $p = p'$ , hasta que  $p = t$ . La Figura 3.8 ilustra el comportamiento del algoritmo descrito. Claramente al igual que **FACE-1**, este algoritmo termina en un número finito de pasos. Sin embargo, existen casos patológicos en los que la complejidad del algoritmo puede llegar a ser  $\Omega(n^2)$ , donde  $n$  es el número de vértices de  $G$ . Para un ejemplo de uno de estos casos véase la Figura 3.9.

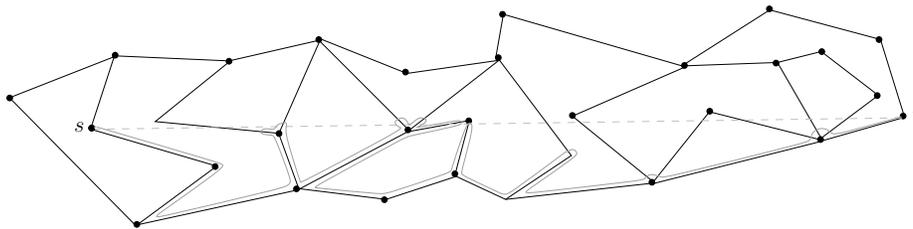


Figura 3.8: Ilustra el comportamiento de **FACE-2** en una gráfica geométrica.

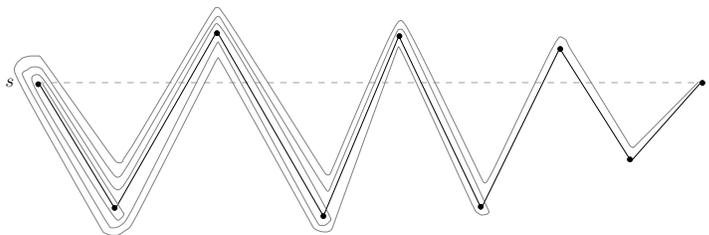


Figura 3.9: Caso en el que **FACE-2** requiere de  $\Omega(n^2)$  pasos.

Entonces se puede concluir lo siguiente.

**Teorema 5.** *El algoritmo FACE-2 alcanza  $t$  en un número finito de pasos.*

En este capítulo se presentó un resumen de los dos primeros artículos que introdujeron la noción de localidad en algoritmos en redes ad hoc, esto con el fin de ilustrar lo que en esta tesis se entiende por un algoritmo local en este tipo de redes.

## Capítulo 4

# Conjuntos dominantes

El problema del conjunto dominante mínimo se define de la siguiente manera. Dada una gráfica  $G = (V, E)$  encontrar un conjunto dominante minimal de  $G$ , cuya cardinalidad sea el número de dominación de  $G$ . Recuérdese que un conjunto dominante  $D$  de  $G$  es un subconjunto  $D \subseteq V$  tal que para cualquier vértice  $v$  de  $G$ ,  $v$  está en  $D$  o es vecino de algún vértice en  $D$ . Un conjunto dominante minimal es aquel que no contiene ningún subconjunto propio que sea también un conjunto dominante. Y el número de dominación de  $G$  es la cardinalidad del conjunto dominante minimal de menor tamaño. No existen algoritmos eficientes para resolver el problema del conjunto dominante mínimo, de hecho para cualquier entero positivo  $k$  la pregunta ¿contiene  $G$  un conjunto dominante de tamaño menor que  $k$ ? es  $NP$ -completa [90]. Para demostrar que el problema del conjunto dominante mínimo es  $NP$ -completo basta con mostrar que se puede reducir al problema de la cubierta por vértices, todo esto puede encontrarse en [91].

Debido a que el problema del conjunto dominante mínimo es  $NP$ -completo, a lo más que se puede aspirar es a encontrar algoritmos que obtengan conjuntos dominantes con tamaño un número constante de veces mayor que el mínimo. En la siguiente sección se mencionan algunos algoritmos existentes en la literatura que aproximan una solución para el problema del conjunto dominante mínimo.

### 4.1. Trabajo relacionado

El problema del conjunto dominante mínimo puede tratarse en una gráfica general [92, 93], o en algún tipo de gráfica en particular. Los algoritmos que son de interés para este trabajo son aquellos que aproximan el conjunto dominante en MANETS. Sin embargo, antes de hablar sobre esto es importante discutir de manera breve un algoritmo que aproxima el conjunto dominante para cualquier gráfica, conocido como algoritmo glotón, el cual se presenta en la siguiente sección. Después, en la sección 4.1.2 se discute sobre

un artículo en donde se presenta un algoritmo para encontrar un conjunto dominante, al cual sus autores presentan como local sin serlo.

Como se mencionó en la sección 1.2.4, existen en la literatura gran cantidad de algoritmos de búsqueda de rutas en redes ad hoc, que utilizan un conjunto dominante conexo como base del algoritmo. Se comenta esto nuevamente para destacar que las aplicaciones del problema del conjunto dominante mínimo en redes ad hoc son importantes.

#### 4.1.1. Algoritmo glotón

Uno de los algoritmos mayormente conocidos para aproximar una solución al problema del conjunto dominante mínimo es el *algoritmo glotón* (o *greedy* en inglés), este algoritmo está basado en el algoritmo muy bien conocido de Chvátal para el problema de la cubierta por vértices [94]. Para encontrar un conjunto dominante  $D$  de una gráfica  $G$ , el algoritmo glotón comienza con un conjunto vacío de vértices  $D$ , y agrega vértices al mismo mientras este no sea un conjunto dominante de  $G$ . El criterio con el cual el algoritmo elige un vértice es de acuerdo con su grado, es decir, que siempre se agrega a  $D$  el vértice de  $G - D$  con mayor grado. El siguiente teorema es muy bien conocido, demostraciones del mismo pueden encontrarse en [94, 95], entre otros.

**Teorema 6.** *El algoritmo glotón para el conjunto dominante mínimo, encuentra un conjunto dominante  $D$  el cual es mayor que un conjunto dominante mínimo por un factor de a lo más  $H(\Delta + 1) < \ln(\Delta + 1) + 1$ , donde  $H(q)$  denota al  $q$ -ésimo número armónico.*

Es decir, que el factor de aproximación del algoritmo glotón es de  $\ln(\Delta)$ . Se ha demostrado que no existe algoritmo polinomial con mejor radio de aproximación, a menos que la clase de problemas  $NP$  sea igual a la clase de problemas  $P$  [96].

#### 4.1.2. Un algoritmo no local

Existen muchos algoritmos en la literatura que aproximan el conjunto dominante mínimo en una gráfica geométrica [49, 97, 98, 99, 100, 101, 102], pero ninguno de estos es local.

En [102] Wang y Li presentan un algoritmo para construir un conjunto dominante conexo y afirman que es local. El conjunto dominante es encontrado de la siguiente forma. Cada nodo cuenta con una etiqueta que lo identifica. Un nodo  $v$  se declara como dominador si posee el identificador más pequeño de entre su 1-vecindad, y envía un mensaje a sus vecinos avisándoles que lo ha hecho. Si un nodo  $u$  recibe un mensaje de  $v$  diciendo que es dominador, entonces  $u$  se declara como dominado y envía el mensaje a su 1-vecindad. Este proceso aparentemente es local, pero considérese la situación

mostrada en la Figura 4.1. Al inicio, el vértice 1 se declarará dominador, y enviará un mensaje a 2 de que es así. Hasta este momento, ni 3, ni 4, ni 5, etcétera, pueden decidir si pertenecen o no al conjunto dominante, 3 tiene que esperar hasta recibir un mensaje de 2 de que este no es dominador, para a su vez declararse dominador y enviar el mensaje a 2 y 4. A su vez, 4 tiene que esperar el mensaje de que 3 se ha declarado dominador, para decidir que él no pertenece al conjunto dominante. Mientras que 5 debe seguir esperando recibir algún mensaje para saber que decidir, y no es sino hasta que 4 le envía un mensaje de que se ha declarado dominador que el primero puede decidir que no pertenece al conjunto. Es evidente entonces que esta situación se propaga a lo largo de toda la red, y que los nodos no pueden tomar una decisión solamente con la información que le dan sus vecinos, sino que, por el contrario, deben esperar a que la información se propague a través de toda la red. Por esto, el algoritmo de Wang y Li no es local, contrario a lo que ellos afirman.



Figura 4.1: El algoritmo de Wang y Li no es local.

## 4.2. El problema

El problema resuelto en esta tesis se puede describir de la siguiente manera. Dada una red ad hoc  $\mathcal{N}$ , representada por su gráfica geométrica unitaria  $GDU(\mathcal{N})$ , se desea encontrar localmente un conjunto dominante de  $GDU(\mathcal{N})$ , de tal forma que el tamaño del mismo sea sólo un número constante de veces mayor que el conjunto dominante mínimo de  $GDU(\mathcal{N})$ . A continuación se describe brevemente la forma en que se procedió para resolver el problema.

La Figura 4.2 ilustra el procedimiento. Primero, dada  $\mathcal{N}$  obtener  $GDU(\mathcal{N})$ , recuérdese que  $GDU(\mathcal{N})$  es una gráfica geométrica, y por lo tanto sus vértices son puntos en  $\mathbb{R}^2$ . Segundo, subdividir el plano en regiones  $R_i$  de cierto tamaño, asignar a un vértice  $v \in V(\mathcal{N})$  con coordenadas  $(x, y)$ , una región  $R_i$  si el punto  $(x, y)$  está dentro de la región  $R_i$ . Tercero, sea  $\mathcal{N}_i$  la subgráfica inducida por los vértices de  $\mathcal{N}$  asignados a  $R_i$ , y las aristas de estos en  $GDU(\mathcal{N})$ . Calcular el conjunto dominante mínimo  $D_i$  de  $\mathcal{N}_i$ . Finalmente, si  $\mathcal{Z}$  es la subdivisión del plano, el conjunto dominante de  $GDU(\mathcal{N})$  es

$$\bigcup_{\forall \mathcal{N}_i \in \mathcal{Z}} D_i.$$

En las siguientes secciones se detallan cada uno de los pasos de la heurísti-

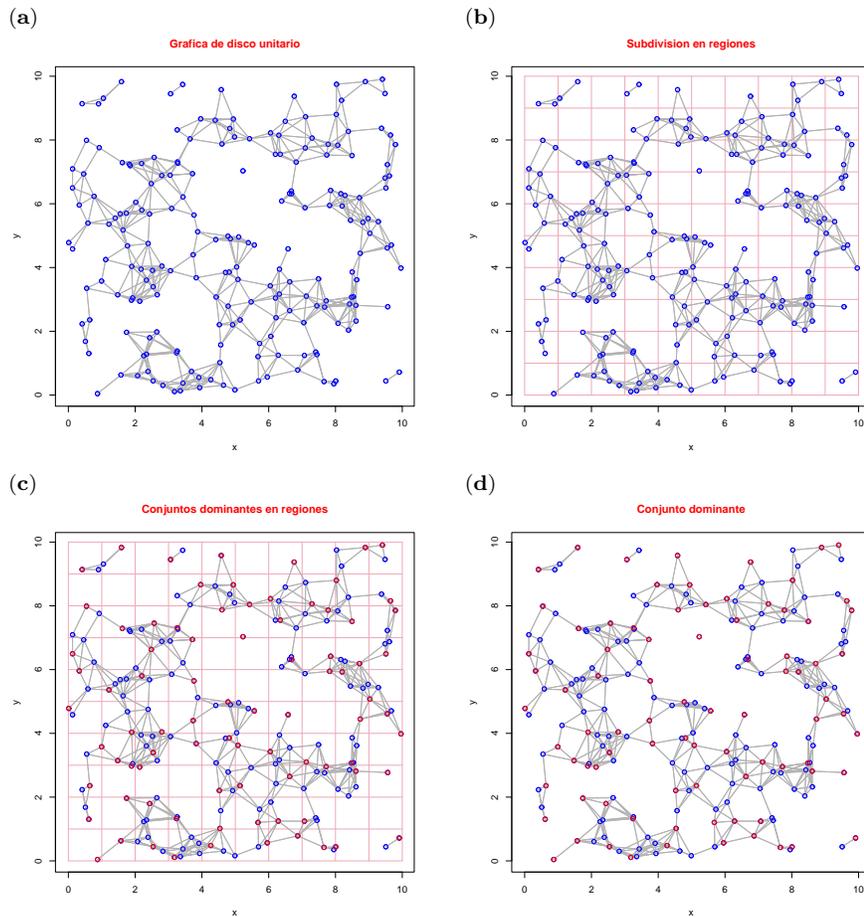


Figura 4.2: La figura ilustra la heurística propuesta. En (a) se muestra la gráfica de disco unitario del conjunto de puntos. En (b) se aprecia la subdivisión del plano en regiones cuadradas de tamaño  $1 \times 1$ . Una vez que se obtiene la subdivisión se encuentra el conjunto dominante mínimo en cada una de las regiones, los vértices dominantes de cada celda se presentan en rojo en (c). Finalmente, en (d) se muestra el conjunto dominante de  $GDU(\mathcal{N})$  obtenido con la heurística.

ca, se describen formalmente y se presentan los resultados.

### 4.3. Subdivisión del plano.

Chaves *et. al.* en [103] subdividen el plano en regiones para resolver un problema de manera local, y después combinan dichas soluciones para obtener una solución global. En particular ellos resuelven el problema de  $t$ -colorear propiamente y de manera local los vértices de una gráfica plana, es decir, asignar colores  $\{1, 2, \dots, t\}$  a los vértices de una gráfica de tal manera que vértices adyacentes reciban colores distintos. En el presente trabajo se

usa la misma técnica para encontrar conjuntos dominantes en redes ad hoc.

Lo primero que se hizo fue subdividir el plano en regiones cuadradas de  $\sigma \times \sigma$ . Las regiones se construyen fácilmente de la siguiente manera. Los vértices de un cuadrado  $R_{i,j}$  de tamaño  $\sigma \times \sigma$  son los puntos

$$\left\{ (\sigma i, \sigma j), (\sigma(i+1), \sigma j), (\sigma i, \sigma(j+1)), (\sigma(i+1), \sigma(j+1)) \right\}. \quad (4.1)$$

Esto se muestra en la Figura 4.3.

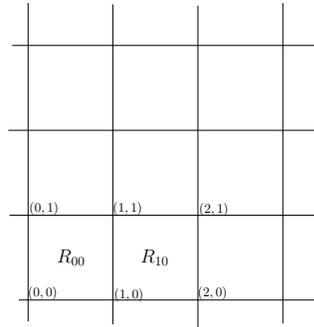


Figura 4.3: En la figura se muestra la relación entre los índices de la región  $R_{i,j}$  y sus vértices.

Un nodo  $v \in V(\mathcal{N})$  con coordenadas  $(x, y)$ , pertenece a  $R_{i,j}$  si

$$\begin{aligned} \sigma i &\leq x < \sigma(i+1) \\ \sigma j &\leq y < \sigma(j+1). \end{aligned}$$

De la misma manera se puede dividir el plano en otras regiones, por ejemplo hexágonos regulares. Los vértices de un hexágono regular  $R_{i,j}$  con aristas de tamaño  $\sigma$  son:

$$R_{i,j} = \begin{cases} \begin{aligned} u_1 &= \left(\frac{3i}{2}\sigma, 2jc\sigma\right), & u_2 &= \left(\frac{5i}{2}\sigma, 2jc\sigma\right), \\ u_3 &= \left(\frac{3i+3}{2}\sigma, (2j+1)c\sigma\right), & u_4 &= \left(\frac{5i}{2}\sigma, (2j+2)c\sigma\right), \\ u_5 &= \left(\frac{3i}{2}\sigma, (2j+2)c\sigma\right), & u_6 &= \left(\frac{(3i-1)\sigma}{2}, (2j+1)c\sigma\right) \end{aligned} & \text{si } i \text{ es par} \\ \begin{aligned} u_1 &= \left(\frac{3i}{2}\sigma, (2j-1)c\sigma\right), & u_2 &= \left(\frac{3i+2}{2}\sigma, (2j-1)c\sigma\right), \\ u_3 &= \left(\frac{3i+3}{2}\sigma, 2jc\sigma\right) & u_4 &= \left(\frac{3i+2}{2}\sigma, (2j+1)c\sigma\right), \\ u_5 &= \left(\frac{3i}{2}\sigma, (2j+1)c\sigma\right), & u_6 &= \left(\frac{3i+3}{2}\sigma - 2, 2jc\sigma\right) \end{aligned} & \text{si } i \text{ es impar} \end{cases}$$

Donde  $c = \sqrt{\frac{3}{4}}$ . Esto se ilustra en la Figura 4.4

Así que cualquier nodo que conozca sólo sus coordenadas y el tamaño de la celda, puede decidir localmente y en tiempo constante a que región pertenece.

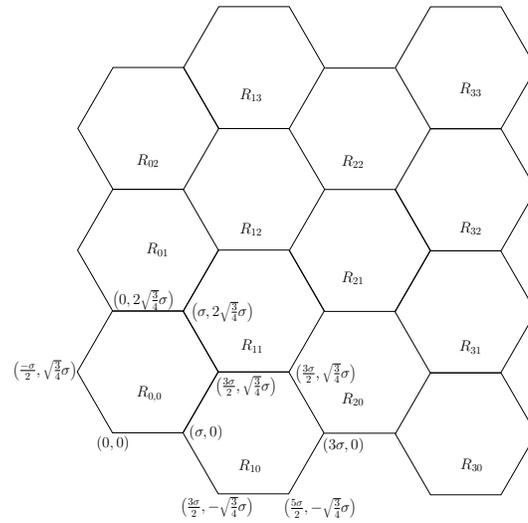


Figura 4.4: En la figura se muestra el orden de los índices que corresponden a las celdas  $R_{ij}$ . Se muestran también los vértices correspondientes a las regiones  $R_{00}$  y  $R_{10}$ .

### 4.3.1. Propiedades de la subdivisión hexagonal

Kershner en su artículo [104], demostró que el mínimo número de círculos necesarios para cubrir el plano se obtiene al hacer que los círculos inscriban las regiones de una subdivisión del plano en hexágonos regulares, conocida como “*panal*”. Dicha subdivisión se muestra en la Figura 4.5.

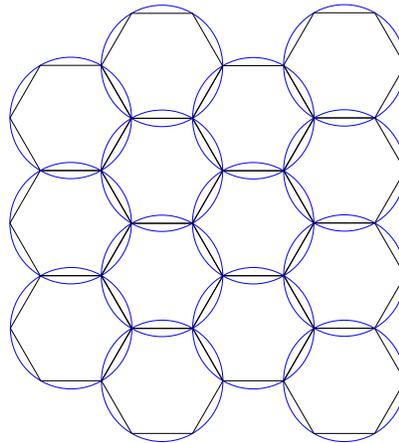


Figura 4.5: La configuración que garantiza el menor número de círculos necesarios para cubrir el plano.

Específicamente el teorema demostrado es el siguiente.

**Teorema 7.** *Sea  $M$  un conjunto acotado de puntos en el plano, y sea  $N(\epsilon)$  el mínimo número de círculos de radio  $\epsilon$  que pueden cubrir  $M$ . Entonces*

$$\lim_{\epsilon \rightarrow 0} \pi \epsilon^2 N(\epsilon) = \frac{2\pi}{3\sqrt{3}} mM, \quad (4.2)$$

donde  $mM$  es la medida de  $M$ .

Como el lado izquierdo de 4.2 es el área total de los círculos que cubren a  $M$ , entonces de acuerdo con Kershner, puede pensarse en la constante  $\frac{2\pi}{3\sqrt{3}} = 1,209\dots$  como una medida de la porción del inevitable traslape de los círculos.

En el caso especial en el que  $M$  es un rectángulo, dicho resultado puede deducirse usando cotas superiores e inferiores convenientes para  $N(\epsilon)$ . Véase la demostración en [104].

Nótese que el problema del conjunto dominante mínimo en una gráfica geométrica se reduce a cubrir un conjunto de puntos en el plano con círculos de radio uno, puesto que como se mencionó anteriormente, estos círculos representan el alcance de los nodos. Así que el Teorema 7 podría sugerir una cota inferior para el problema del conjunto dominante mínimo en una gráfica de disco unitario. Sin embargo esto no es así, puesto que el centro de los círculos en el problema que deseamos resolver deben ser puntos del conjunto, mientras que en el Teorema 7, el centro de los círculos no necesariamente son puntos del conjunto. Todo esto es mencionado porque al encontrarse esta propiedad de la subdivisión hexagonal, quien escribe pensó que podría tomar provecho de la misma para resolver el problema que se desea. Sin embargo no se logró nada en esta línea, ya que finalmente se encontró que los problemas son completamente distintos. A pesar de esto este resultado sirvió de inspiración para utilizar la subdivisión hexagonal en la heurística y con esto se logró obtener algunos resultados, los cuales se presentan en la sección 4.4.2.

## 4.4. Conjuntos dominantes locales

Como se mencionó anteriormente, una vez que el plano ha sido subdividido en regiones, la heurística procede a calcular el conjunto dominante mínimo dentro de cada región. Una vez que dichos conjuntos mínimos han sido encontrados, basta con unirlos para formar el conjunto dominante de la gráfica. En la sección 4.4.1 se presenta una discusión sobre por qué es válido suponer que el conjunto dominante mínimo de cada región puede ser encontrado. Después en la sección 4.4.2 se presenta el teorema principal de este trabajo, el cual da una cota para el tamaño del conjunto dominante obtenido por la heurística.

### 4.4.1. Conjuntos mínimos por regiones

La heurística encuentra el conjunto dominante mínimo en cada región como parte del proceso de encontrar el conjunto dominante de la gráfica. Es importante justificar por que se puede suponer que este conjunto puede calcularse en un tiempo polinomial respecto al número de vértices de la gráfica. Antes de presentar esta justificación es necesario dar algunos antecedentes.

¿Cuál es el tamaño máximo que puede alcanzar un conjunto dominante mínimo en una gráfica geométrica de disco unitario? Para el caso en el que la gráfica está acotada por un polígono, esta pregunta puede responderse si se replantea de la siguiente forma: ¿Cuál es el máximo número de puntos con distancia mutua mayor que uno que pueden colocarse dentro de un polígono? Es fácil darse cuenta que las dos preguntas son equivalentes, ya que en el peor de los casos la configuración de los puntos de la gráfica puede ser tal que se alcance el máximo de puntos a distancia mutua mayor que uno, y por lo tanto el tamaño del conjunto dominante mínimo alcance su máximo.

En 1961 Oler [105] demostró un teorema que implica el siguiente colorario.

**Colorario 1.** *Si  $X$  es un subconjunto convexo compacto en el espacio Euclideo, entonces  $N_{p_{dist \leq 1}}$ , el número de puntos con distancia mutua al menos 1, es a lo más*

$$\frac{2}{\sqrt{3}}A(X) + \frac{1}{2}P(X) + 1, \quad (4.3)$$

donde  $A(X)$  es el área y  $P(X)$  el perímetro del subconjunto convexo.

Véase [106] para una demostración básica del colorario.

Nótese que si  $A(X)$  o  $P(X)$  crecen mucho, entonces también lo hace el número de posibles puntos a distancia mutua al menos uno en el conjunto. Sin embargo, en el caso de subdivisiones regulares del plano<sup>1</sup>, que es nuestro caso particular, tanto el área como el perímetro están acotados (por el área y el perímetro de un círculo de tamaño adecuado).

Si  $X$  es un cuadrado de  $\sigma \times \sigma$  entonces  $A(X) = \sigma^2$  y  $P(X) = 4\sigma$  entonces, por el colorario 1, se tiene que el máximo número de puntos con distancia mutua al menos 1 en un cuadrado es

$$N_{p_{dist \leq 1}} \leq \frac{2}{\sqrt{3}}\sigma^2 + 2\sigma. \quad (4.4)$$

Si  $\sigma = 1$ , entonces  $N_{p_{dist \leq 1}} \leq 4,1547$ . Si  $\sigma = 2$ , entonces  $N_{p_{dist \leq 1}} \leq 9,6188$ . Si  $\sigma = 3$ , entonces  $N_{p_{dist \leq 1}} \leq 17,3923$ , etc.

Por otro lado, si  $X$  es un hexágono regular con aristas de tamaño  $\sigma$ , entonces

$$N_{p_{dist \leq 1}} \leq 3\sigma^2 + 3\sigma + 1. \quad (4.5)$$

---

<sup>1</sup>Este concepto se mencionó en la sección 2.5.

Es decir, si  $\sigma = 1$ , entonces  $N_{p_{dist \leq 1}} \leq 7$ . Si  $\sigma = 2$ , entonces  $N_{p_{dist \leq 1}} \leq 19$ . Si  $\sigma = 3$ , entonces  $N_{p_{dist \leq 1}} \leq 37$ , etc.

Claramente  $N_{p_{dist > 1}}$ , el máximo número de puntos con distancia mutua **mayor que uno**, es menor que  $N_{p_{dist \leq 1}}$ . Es decir que

$$N_{p_{dist > 1}} < N_{p_{dist \leq 1}} \leq \frac{2}{\sqrt{3}}A(X) + \frac{1}{2}P(X) + 1. \quad (4.6)$$

El siguiente teorema da cotas superiores e inferiores para  $N_{p_{dist > 1}}$ , cuando  $X$  es un cuadrado.

**Teorema 8.** *Sea  $X$  un cuadrado de  $\sigma \times \sigma$ .  $N_{p_{dist > 1}}$ , el número de puntos a distancia mutua mayor que uno, que pueden ser colocados dentro de  $X$  es*

$$(\sigma + 1)^2 - 1 \leq N_{p_{dist > 1}} < \frac{2}{\sqrt{3}}\sigma^2 + 2\sigma.$$

**Demostración.** La cota superior se obtiene del Colorario 1. La cota inferior se obtiene de la siguiente manera. Supóngase que el cuadrado está subdividido en cuadrados de  $1 \times 1$ . Colóquese el primer punto tan cerca como sea posible de una de las esquinas de  $X$ , sea dicho punto  $p_1$ , después trácese un rayo con ángulo  $\alpha$ , como se muestra en la figura 4.6, hasta que este intersekte a una arista de los cuadrados de  $1 \times 1$  en un punto  $p_2$ , y de tal forma que  $\alpha$  sea el mínimo ángulo necesario tal que la distancia entre  $p_1$  y  $p_2$  sea apenas mayor a uno. Repítase el proceso usando ahora  $p_2$ , etcétera, hasta que no sea posible colocar mas puntos. Haciendo esto se lograrán colocar  $(\sigma + 1)^2 - 1$  puntos. Es claro que todos los puntos colocados de esta manera estarán a distancia mutua mayor que uno. ■

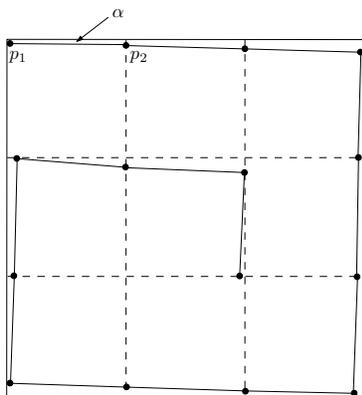


Figura 4.6: Proceso para colocar el máximo número de puntos a distancia mutua mayor que uno en un cuadrado.

Por lo tanto, si  $A(X)$  y  $P(X)$  permanecen constantes, el tamaño máximo que puede alcanzar el conjunto dominante mínimo  $OPT$  de una gráfica

geométrica  $\mathcal{N}_i$  de la región  $R_i$  de una gráfica  $G$ , es constante comparado con el número de vértices de  $G$ , es decir que  $|OPT|$  permanece sin cambios aún cuando el tamaño de  $G$  aumente.

En lo sucesivo nos referiremos a  $N_{p_{dist>1}}$  como  $N_p$ .

Ahora bien, el conjunto dominante mínimo de cada región puede ser calculado usando “fuerza bruta” de la siguiente manera. Para la gráfica  $\mathcal{N}_i$  con  $p$  vértices, encontrar todos los posibles subconjuntos de vértices de tamaño uno, y verificar para cada uno si forma un conjunto dominante. De no ser así, encontrar todos los posibles subconjuntos de tamaño dos y verificar si alguno forma un conjunto dominante, etc. Detener el proceso cuando se encuentre el primer conjunto dominante, que claramente será el de menor tamaño. En el peor de los casos se deberán calcular todos los posibles subconjuntos de tamaño menor o igual que  $N_p$ . Es decir que para la gráfica  $\mathcal{N}_i$  el algoritmo calcularía

$$\binom{p}{1} + \binom{p}{2} + \binom{p}{3} + \dots + \binom{p}{N_p} = \sum_{j=1}^{N_p} \binom{p}{j} \quad (4.7)$$

subconjuntos de vértices. Se sabe que no existe forma cerrada para la suma parcial 4.7 [107]; sin embargo, siempre se puede escribir lo siguiente

$$\begin{aligned} \sum_{j=1}^{N_p} \binom{p}{j} &= p + \frac{p(p-1)}{2!} \\ &+ \frac{p(p-1)(p-2)}{3!} \\ &+ \dots \\ &+ \frac{p(p-1)(p-2)\dots(p-(N_p-1))}{N_p!}. \\ &= O(p^{N_p}). \end{aligned} \quad (4.8)$$

Obsérvese que  $N_p$  es una constante, es decir que sin importar que  $p$  crezca  $N_p$  permanece sin cambios siempre que ni el área ni el perímetro aumenten. Por lo tanto, 4.8 es polinomial, es decir, que para la gráfica  $\mathcal{N}_i$  en cierta región de la subdivisión, su conjunto dominante mínimo se puede calcular en tiempo polinomial.

Supóngase que la subdivisión de la gráfica  $G$  con  $m$  vértices, tiene  $k$  celdas no vacías, y que el máximo número de puntos en cada celda es  $p$ , por lo tanto se deberán calcular máximo

$$k \sum_{j=1}^{N_p} \binom{p}{j} = O(kp^{N_p}) \quad (4.9)$$

subconjuntos de vértices para encontrar el conjunto dominante mínimo de todas las subgráficas de  $G$  en la subdivisión. Nótese que nuevamente 4.9 es polinomial. Si se deseara reescribir dicho polinomio en términos de  $m$ , sólomente tendría que tomarse en cuenta que  $p \ll m$ , por lo que

$$kp^{N_p} < cm^{N_p} \quad (4.10)$$

para alguna constante  $c > 0$ .

Por lo tanto, aún al algoritmo menos sofisticado le tomaría tiempo polinomial encontrar el conjunto dominante mínimo de todas las gráficas  $\mathcal{N}_i$  de la subdivisión de  $G$ .

Es importante destacar que entre mayor sea  $N_p$ , es decir, entre más grandes se tomen las regiones de la subdivisión, mayor será el tiempo que le tomará al algoritmo calcular los conjuntos dominantes mínimos de cada celda, puesto que mayor será el grado del polinomio mostrado anteriormente. Pero, por otro lado, entre más grandes sean las regiones, menor será la diferencia de tamaños entre el conjunto dominante calculado por la heurística y el conjunto dominante mínimo de  $G$ . Es pues necesario encontrar un balance entre ambas cosas, es decir, un tamaño de celda que sea relativamente rápido de procesar, pero que a la vez genere una buena aproximación del conjunto dominante mínimo. Esto se discute un poco más en la sección 4.5.

#### 4.4.2. Tamaño del conjunto dominante

Como se ha mencionado, el objetivo del presente trabajo es encontrar una heurística que resuelva el problema de encontrar localmente un conjunto dominante para una gráfica de disco unitario, de tal forma que el conjunto obtenido sea solamente un número constante de veces mayor que el conjunto dominante mínimo de la misma gráfica. En la presente sección se demuestra lo siguiente:

- i) Que la heurística propuesta calcula un conjunto dominante.
- ii) Que el conjunto dominante calculado es a lo más 15 veces más grande que el conjunto dominante óptimo.

Para demostrar dichos puntos es necesario utilizar los siguientes lemas, los cuales son ya muy conocidos. Antes de leerlos se sugiere al lector recordar las definiciones de conjunto dominante Def.7 y conjunto independiente Def.8, presentadas en la sección 2.4.

**Lema 3.** *Sea  $G$  una gráfica,  $D$  un conjunto dominante mínimo de  $G$ , e  $I$  un conjunto independiente maximal de  $G$ . Entonces  $|D| \leq |I|$ .*

**Demostración.** Se sigue directamente de las definiciones. Cualquier vértice que no este en un conjunto independiente maximal es adyacente a al menos

un vértice en el conjunto, y por tanto es dominado por este, entonces un conjunto independiente maximal es también un conjunto dominante. Por otro lado, un conjunto independiente que es también un conjunto dominante tiene que ser maximal, pues cualquier vértice que no este en el conjunto es dominado por al menos un vértice en el conjunto. Por lo tanto, un conjunto independiente es también un conjunto dominante si y sólo si el primero es maximal. Entonces,  $|D| \leq |I|$ . ■

El siguiente lema, cuya demostración fue tomada de [108], es una propiedad de los empaquetamientos de discos unitarios en el plano, la cual será utilizada para demostrar el lema 5.

**Lema 4.** *Sea  $C$  un círculo de radio  $r$  y sea  $S$  un conjunto de círculos de radio  $r$ , de tal forma que cada círculo en  $S$  intersecta a  $C$  y que no hay dos círculos en  $S$  que se intersecten mutuamente. Entonces  $|S| \leq 5$ .*

**Demostración.** Supóngase que  $|S| \geq 6$ . Sean  $s_i$ , para  $1 \leq i \leq 6$ , los centros de cualquiera seis círculos en  $S$ . Sea  $c$  el centro de  $C$ . Denótese el rayo  $\overrightarrow{cs_i}$  por  $r_i$  ( $1 \leq i \leq 6$ ), véase la Figura 4.7. Como hay seis rayos emanando de  $c$ , debe haber al menos un par de rayos  $r_j$  y  $r_k$  tales que el ángulo entre ellos sea a lo mas de  $60^\circ$ . Puede verificarse que la distancia entre  $s_j$  y  $s_k$  es a lo mas  $2r$ , lo cual implica que los círculos centrados en  $s_j$  y  $s_k$  se intersectan, contradiciendo la hipótesis. Por lo tanto,  $|S| \leq 5$ . ■

Dado el lema anterior, es fácil demostrar lo siguiente.

**Lema 5.** *Sea  $G$  una gráfica de disco unitario. Sea  $D$  un conjunto dominante mínimo de  $G$ , y sea  $S$  cualquier conjunto independiente de  $G$ . Entonces,  $|S| \leq 5|D|$ .*

**Demostración.** Sea  $D$  un conjunto dominante de  $G$ . Para cada vértice  $v \in D$ , considérese el círculo de radio uno centrado en  $v$ . Entonces, cada vértice de  $G$  debe caer en al menos uno de estos círculos centrado en algún elemento de  $D$ . Por el Lema 4.4.2, un conjunto independiente  $S$  puede tener máximo cinco elementos en cada uno de sus círculos. Por lo tanto,  $|S| \leq 5|D|$ . ■

Como ya se ha descrito anteriormente, la heurística procede de la siguiente forma para encontrar el conjunto dominante. Primero, subdividir el plano en regiones poligonales regulares. Sea  $\mathcal{Z}$  dicha partición. Para cada región  $R_i \in \mathcal{Z}$ , sea  $\mathcal{N}_i$  la subgráfica inducida por los nodos de  $\mathcal{N}$  en  $R_i$ . Sea  $D_i$  el conjunto dominante mínimo de  $\mathcal{N}_i$ . Como se ha discutido anteriormente, tanto el proceso de decidir a que región pertenece un vértice, como el de calcular el conjunto dominante mínimo de una región, son factibles y pueden hacerse de manera local. Finalmente, el conjunto dominante  $D$  de  $GDU(\mathcal{N})$  es  $\bigcup_{\mathcal{N}_i \in \mathcal{Z}} D_i$ . Nótese que si  $I_i$  es el conjunto independiente maximal de  $\mathcal{N}_i$ , entonces por el Lema 3,  $|D_i| \leq |I_i|$ .

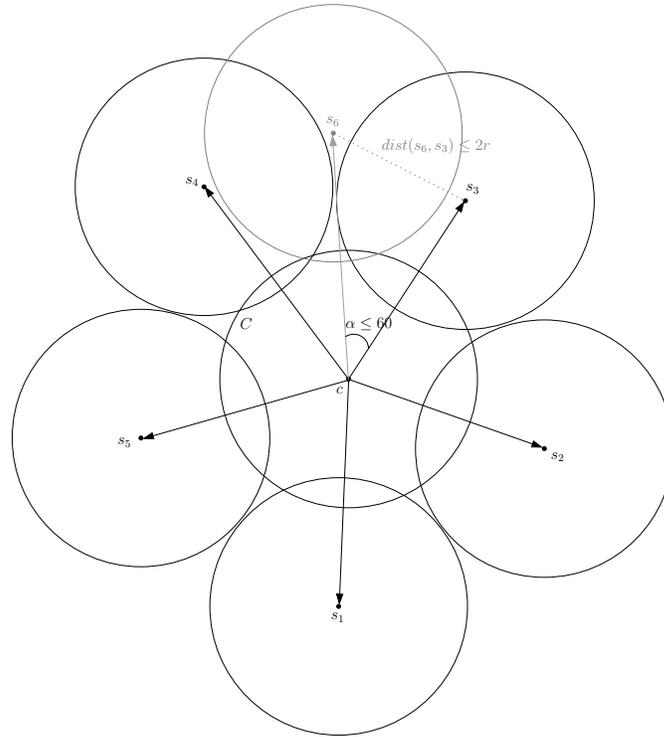


Figura 4.7: No puede haber 6 círculos intersectando a  $C$  sin que se intersecten mutuamente.

Entonces, es necesario demostrar que efectivamente el conjunto  $D$  calculado por la heurística es un conjunto dominante de  $GDU(\mathcal{N})$ . Y además, es necesario dar una cota para su tamaño. Todo esto se presenta en el lema y teorema siguientes.

**Lema 6.** *El conjunto  $D$  calculado por la heurística es un conjunto dominante de  $GDU(\mathcal{N})$ .*

**Demostración.** Supóngase por contradicción que  $D$  no es un conjunto dominante de  $GDU(\mathcal{N})$ , entonces existe  $v \in V(\mathcal{N})$ , de tal forma que  $v$  no está dominado. Sea  $R_i$  la región a la que pertenece  $v$  según la heurística, y sea  $D_i$  el conjunto dominante de dicha región. Como  $v$  no está dominado, entonces  $D_i$  no es un conjunto dominante de  $R_i$ , pero esta es una contradicción porque la heurística supone que  $D_i$  es el conjunto dominante mínimo de  $R_i$ . Por lo tanto  $v$  está dominado, y  $D$  es un conjunto dominante de  $GDU(\mathcal{N})$ . ■

El siguiente teorema da una cota en el tamaño de  $D$ .

**Teorema 9.** Sea  $GDU(\mathcal{N})$  una red inalámbrica unitaria. Sea  $OPT$  el conjunto dominante mínimo de  $\mathcal{N}$ . Se puede calcular con un algoritmo local un conjunto dominante  $D$  de  $GDU(\mathcal{N})$ , de tal forma que  $|D| \leq 15|OPT|$ .

**Demostración.** Utilícese la heurística descrita anteriormente para encontrar  $D = \bigcup_{N_i \in \mathcal{Z}} D_i$ . Sea  $\mathcal{Z}$  una partición del plano en hexágonos regulares, de tal forma que el centro de uno de estos hexágonos es el origen. Obsérvese que  $\mathcal{Z}$  es 3-coloreable, es decir que se puede colorear con tres colores de tal forma que hexágonos vecinos reciban colores distintos, véase la Figura 4.8. Obsérvese además, que para cada color, la unión de todos los vértices en hexágonos con el mismo color, forma un conjunto independiente en  $GDU(\mathcal{N})$ , entonces por el Lema 5, el tamaño de este conjunto es a lo más cinco veces mayor que el del conjunto dominante mínimo de  $GDU(\mathcal{N})$ . Entonces, la unión de los conjuntos dominantes mínimos de todos los hexágonos, tiene cardinalidad a lo más 15 veces mayor que la del conjunto dominante mínimo de  $GDU(\mathcal{N})$ . ■

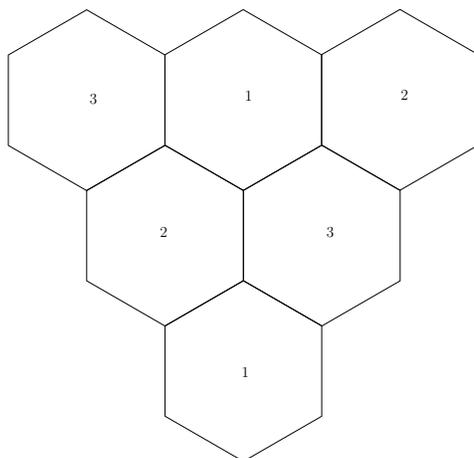


Figura 4.8: La subdivisión del plano en hexágonos regulares puede ser tres coloreada.

Resultados similares se pueden conseguir para el caso en el que la subdivisión se haga con otros polígonos regulares, como cuadrados o triángulos. Si la subdivisión se hace con cuadrados, entonces como esta es 4-coloreable, entonces  $|D| \leq 20|OPT|$ . Lo mismo pasa si se subdivide el plano en triángulos, pues la subdivisión puede colorearse también con cuatro colores.

Es importante mencionar que aunque el tamaño de las aristas de la subdivisión no afecta el resultado presentado, la idea original del trabajo era la siguiente. Intuitivamente si la subdivisión tiene regiones “pequeñas” comparadas con el tamaño de la región que encierra a la gráfica completa, el conjunto dominante encontrado por la heurística no resulta tan buena

aproximación como cuando las regiones con “grandes”. Es decir, que en el límite cuando la región en la subdivisión tiende a la región que contiene toda la gráfica, el tamaño del conjunto dominante encontrado por la heurística tiende al tamaño del conjunto dominante mínimo. A pesar de los esfuerzos para tratar de encontrar un resultado que usara este hecho, no se pudo lograr nada. Sin embargo, se realizaron experimentos tomando en cuenta esto, y como se esperaba conforme el tamaño de las regiones aumenta, se logra una mejor aproximación del conjunto dominante mínimo.

Cabe mencionar que en todos los experimentos realizados, el tamaño del conjunto dominante calculado por la heurística es siempre menor o igual que dos veces el tamaño del conjunto dominante calculado por el algoritmo glotón de Chvátal. Los detalles de estos experimentos se presentan en la siguiente sección.

## 4.5. Simulaciones

En esta sección se presentaran los resultados obtenidos al experimentar la heurística en redes generadas por una computadora.

Para simular una red, utilizando el programa  $\mathcal{R}$  [109], se generó de manera aleatoria un conjunto de puntos en  $\mathbb{R}^2$ , de tal forma que la probabilidad siguiera una distribución uniforme. Después, se calcularon las distancias Euclidianas entre pares de puntos y se agregaron aristas entre aquellos puntos con distancia mutua menor que uno. Esto para generar la gráfica de disco unitario. Se procedió entonces a seguir la heurística, es decir, a dividir el plano en regiones y a calcular el conjunto dominante mínimo de cada región. Esto último se hizo de la siguiente forma. Sea  $\mathcal{N}_i$  la gráfica inducida dentro de la región  $R_i$  con  $p$  puntos, el programa toma todos los posibles subconjuntos de vértices de  $\mathcal{N}_i$  de tamaño  $k$ , para  $k = 1, 2, 3, \dots, p$ . Para cada uno de estos subconjuntos verifica si este forma un conjunto dominante para  $\mathcal{N}_i$ . El programa se detiene cuando encuentra el primer conjunto dominante, que por tanto será el más pequeño.

La tabla 4.1 resume los experimentos que se corrieron utilizando en la heurística tres tipos de subdivisiones distintas: triangulares, cuadradas y hexagonales. Además, se corrieron experimentos para redes de tamaños  $n = 500$  y  $n = 800$ , utilizando solamente subdivisiones cuadradas y hexagonales. Esto se resume en la tabla 4.2. En cada uno de los casos se aplicó el algoritmo glotón de Chvátal, descrito anteriormente, para encontrar el conjunto dominante de la red y tener un punto de comparación.

Los límites de  $x$  y  $y$  se conocen también como *grid*. Es decir, que cuando los límites de  $x$  son 0 y 5 se dice que el grid es 5.

Obsérvese que aún cuando las aristas para los tres tipos de regiones utilizadas tengan el mismo tamaño, sus áreas son distintas. Entonces se tiene que ser cuidadoso en esto, puesto que una región con menor área muy

n	límites de x y y
100	0 y 5
	0 y 6
	0 y 7
200	0 y 5
	0 y 6
	0 y 7

Tabla 4.1: Experimentos para subdivisiones triangulares, cuadradas y hexagonales.

n	límites de x y y
500	0 y 10
	0 y 15
	0 y 20
800	0 y 10
	0 y 20
	0 y 30

Tabla 4.2: Experimentos para subdivisiones cuadradas y hexagonales.

probablemente contendrá una red con menor número de vértices que una con mayor área. Es por esto que debemos comparar el comportamiento de las subdivisiones en regiones con áreas similares. La tabla 4.3 muestra el tamaño de las áreas de las tres formas distintas de regiones cuando se varía el tamaño de las aristas.

Tamaño de la arista	Área triángulo	Área cuadrado	Área hexágono
1	0.433	1	2.5980
2	1.7320	4	10.3923
3	3.8971	9	
4	6.9282		
5	10.8253		

Tabla 4.3: Áreas de las Regiones

Por lo tanto, se puede decir que las regiones con aristas de tamaño 5 para triángulos, 3 para cuadrados y 2 para hexágonos, tienen áreas aproximadamente iguales. Aunque es claro que una mejor equivalencia se encontraría si se usaran aristas con longitudes reales, como se verá en los resultados el hecho de usar sólo longitudes enteras no afecta en gran medida el desempeño de la heurística con las distintas regiones.

### 4.5.1. Resultados

Las gráficas de barras 4.9 – 4.14, que se presentan a continuación, resumen los distintos tamaños de conjuntos dominantes encontrados por la heurística, utilizando cada una de las tres posibles subdivisiones. El punto de comparación que se tomó es el algoritmo glotón de Chvátal. Se sugiere al lector analizar las gráficas antes de leer la discusión de los resultados. Las figuras de todas las redes utilizadas en las simulaciones, y de cada uno de los resultados de los experimentos, se encuentran en el Apéndice A.

Es importante mencionar que aunque anteriormente se dijo que era conveniente comparar los resultados de las distintas subdivisiones tomando en cuenta el área de las mismas, y que el punto de comparación sería de longitudes de  $2cm$  para hexágonos,  $3cm$  para cuadrados y  $5cm$  para triángulos, en ocasiones bastó con utilizar aristas de  $4cm$  para triángulos pues con esto la heurística ya encontraba un conjunto dominante de menor tamaño que el algoritmo glotón; cuando este sea el caso, se omite en la gráfica la barra correspondiente a aristas de  $5cm$  para triángulos.

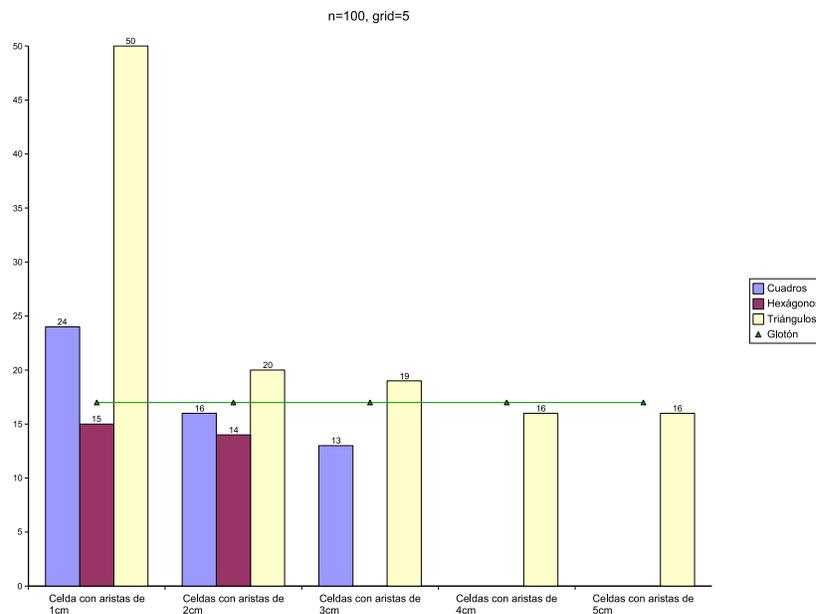


Figura 4.9: La gráfica presenta los resultados de la simulación realizada para una red con 100 nodos y usando un grid de 5.

Para las redes de tamaños  $n = 100$  y  $n = 200$  se puede observar que siempre la heurística, sin importar cual de los tres tipos de subdivisión fue utilizada, encuentra un conjunto dominante de menor tamaño que el encontrado por el algoritmo glotón. Nótese que en cuatro de los seis casos, al usar la subdivisión de cuadros, se encuentra un conjunto dominante de menor ta-

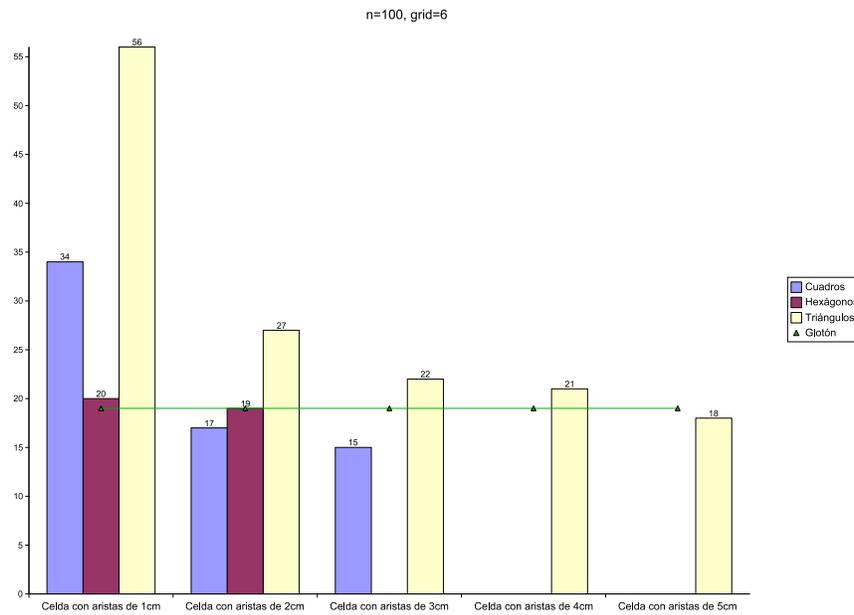


Figura 4.10: Simulación de la heurística en una red con 100 nodos y un grid de 6.

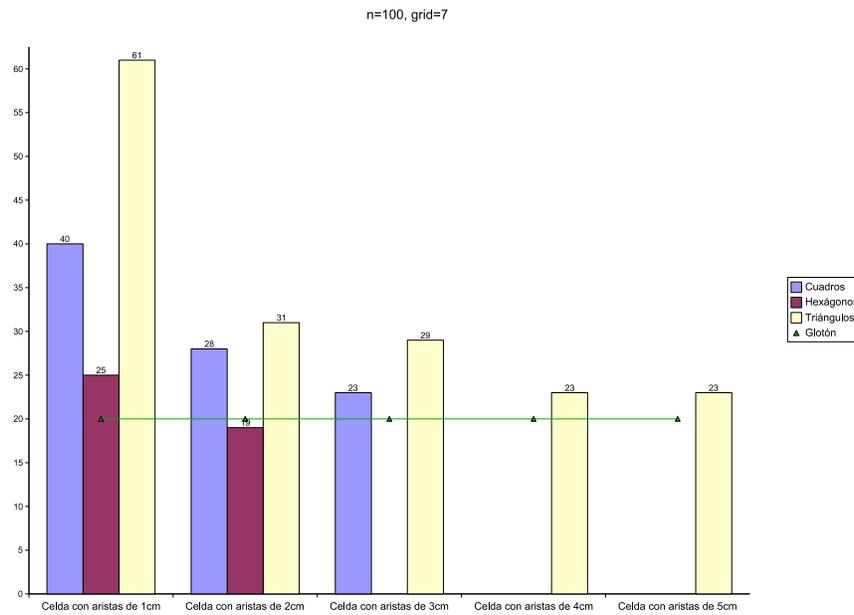


Figura 4.11: Resultados de la simulación de la heurística para una distribución de puntos con grid 7 y 100 elementos.

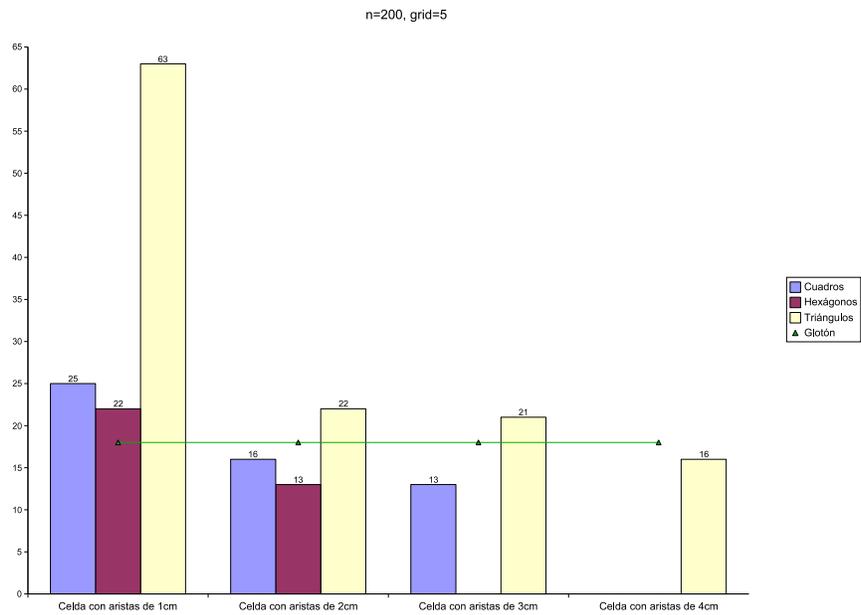


Figura 4.12: El tamaño de la red es de 200 puntos y la grid es de 5.

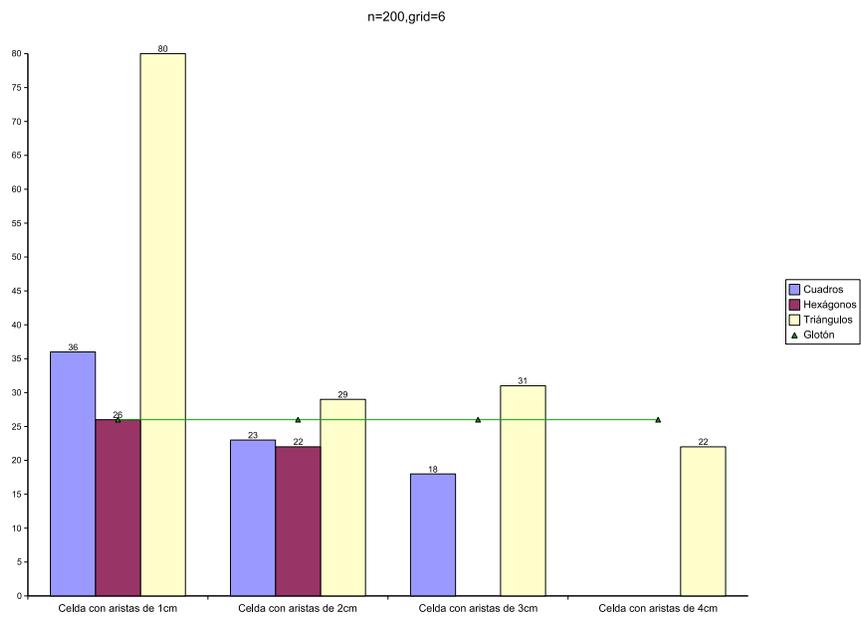


Figura 4.13: Simulación de la heurística en una red con 200 nodos y un grid de 6 para la distribución de los puntos.

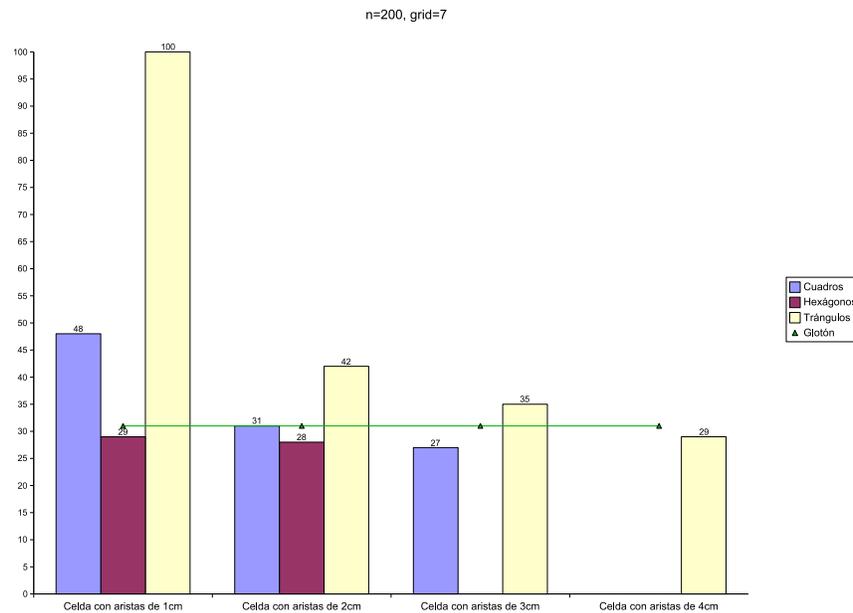


Figura 4.14: Resultados para una red de 200 puntos con grid 7.

maño que usando cualquiera de las otras dos subdivisiones. En solo un caso ( $n = 100$ ,  $\text{grid} = 7$ ) la subdivisión hexagonal presentó un mejor desempeño, y no solamente obtiene el conjunto dominante de menor tamaño, sino que es con la única que se vence al algoritmo glotón. En ningún caso la heurística usando la subdivisión triangular logra vencer a las otras.

Nótese como conforme el tamaño de la celda aumenta, el tamaño del conjunto dominante encontrado por la heurística disminuye. Sin embargo, el tiempo que tardaba el programa en dar el resultado se incrementaba conforme lo hacía el tamaño de la región. Para darse una idea de como cambia el tiempo de corrida de la heurística conforme cambia el tamaño de la región, en la Tabla 4.4 se muestran algunos valores para  $p$ , recuérdese que  $p$  es el máximo número de puntos dentro de una región, y que el tiempo de corrida tiene cota superior de  $kp^{N_p}$ . El caso mostrado en la tabla se tomó de los resultados de las simulaciones cuando  $n = 100$ , y las regiones son cuadradas. Para el valor de  $k$  se contaron también las celdas vacías, y  $N_p$  es menor o igual el valor mostrado en la tabla, según la teoría analizada anteriormente.

En la teoría la subdivisión hexagonal garantiza un conjunto dominante de menor tamaño que la subdivisión cuadrada (sección 4.4.2), sin embargo contrario a lo esperado, fue utilizando esta última que en la mayoría de los casos se encontró el conjunto dominante de menor tamaño. Con el fin de observar el comportamiento de la heurística con redes de mayor ta-

Grid	Tamaño de la arista	$N_p$	k	p
5	1	4	25	10
	2	9	9	24
	3	17	4	40
6	1	4	36	8
	2	9	9	18
	3	17	4	29
7	1	4	49	8
	2	9	16	14
	3	17	9	27

Tabla 4.4: La tabla muestra algunos valores para  $N_p, k, p$ , en el caso en que  $n = 100$  y la subdivisión es cuadrada. Nótese como  $p$  aumenta conforme lo hace el tamaño de la arista.

maño, usando solo subdivisiones cuadradas y hexagonales, se realizaron los experimentos listados en la tabla 4.2, ya que se deseaba observar un mejor comportamiento de la heurística con la subdivisión hexagonal que con la cuadrada. Las gráficas de barras 4.15 – 4.20 que se presentan a continuación resumen los resultados. Nuevamente se sugiere al lector observarlas detalladamente antes de proseguir con la lectura.

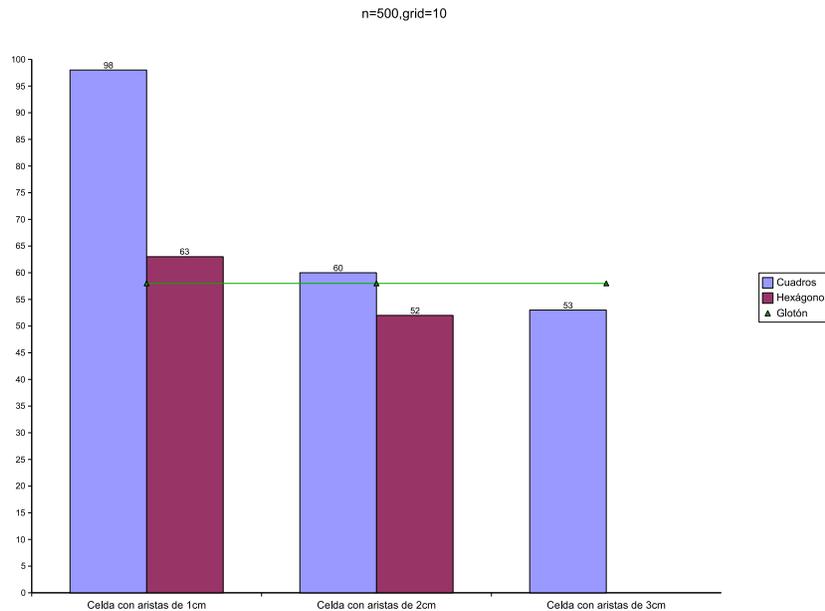


Figura 4.15: Resultados de la simulación realizada con una red con 500 puntos y grid de 10.

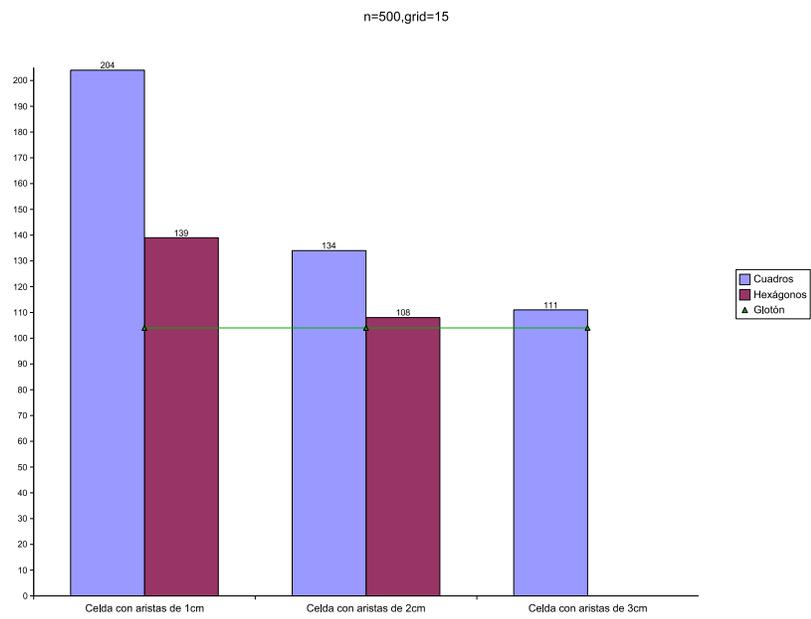


Figura 4.16: Resultados para una red de 500 puntos con grid 15.

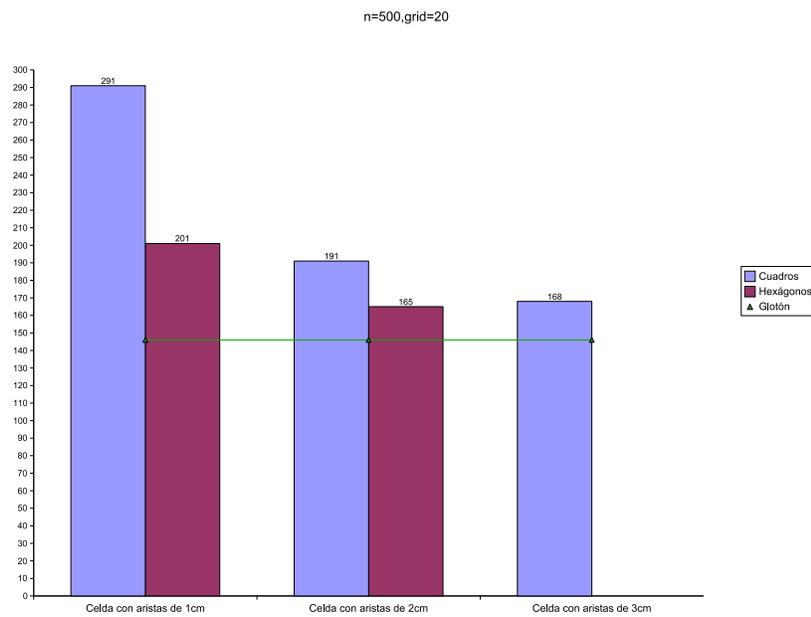


Figura 4.17: Muestra lo obtenido al simular la heurística en una red de 500 puntos y un grid de 20.

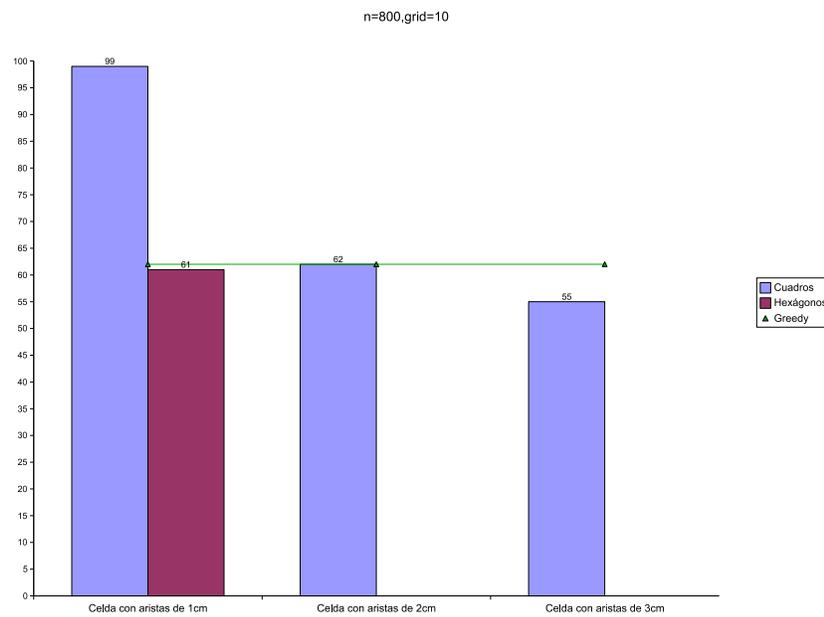


Figura 4.18: La gráfica muestra los resultados de la simulación hecha en una red de 800 puntos con grid 10.

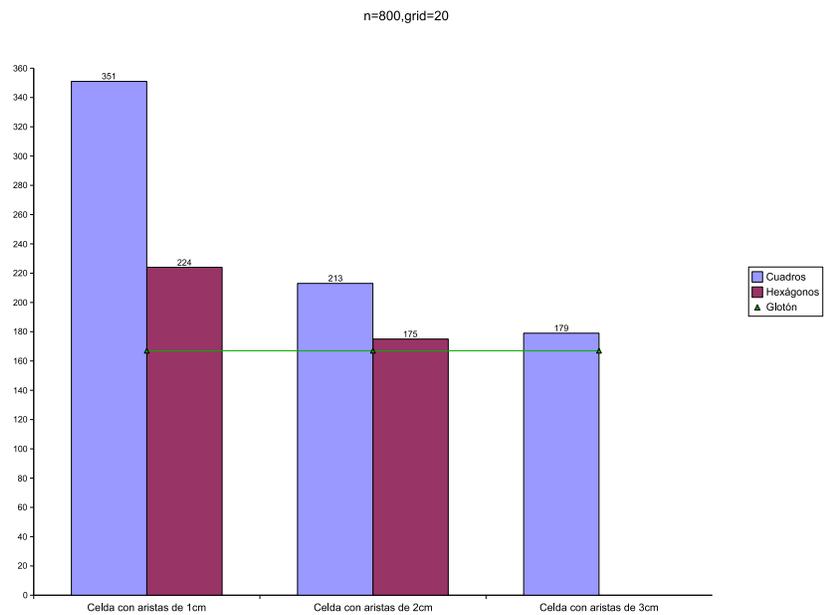


Figura 4.19: Resultados para una red de 800 puntos con grid 20.

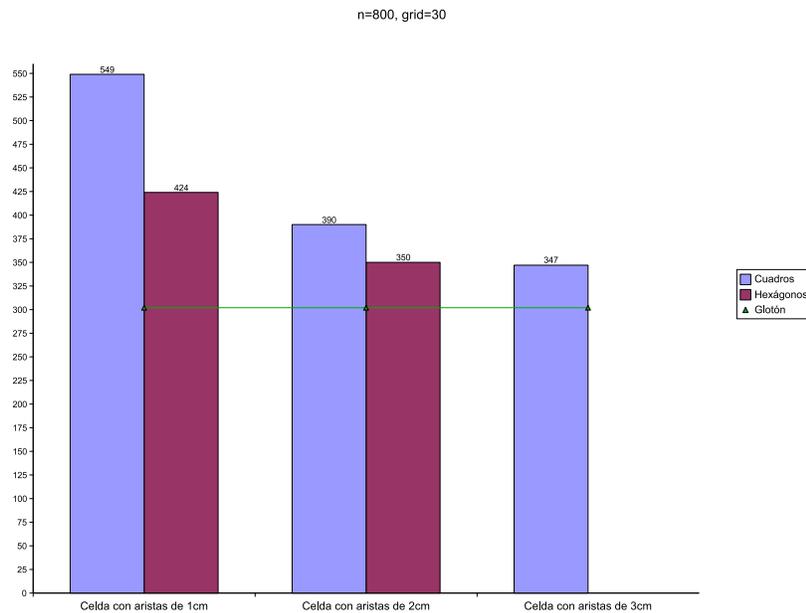


Figura 4.20: El tamaño de la red es de 800 puntos con grid 30.

Para estas simulaciones se observa que en cuatro de los seis casos la heurística con la subdivisión hexagonal encuentra un conjunto dominante de menor tamaño que usando la subdivisión cuadrada. Sin embargo en cuatro de los seis casos, con ninguna de las dos subdivisiones la heurística logra encontrar un conjunto dominante de menor tamaño que el calculado por el algoritmo glotón de Chvátal. A pesar de esto, obsérvese que en ningún caso el conjunto encontrado por la heurística es del doble de tamaño que el conjunto encontrado por el algoritmo glotón, es decir que en todos y cada uno de los casos se cumple lo siguiente. Si  $D_{local}$  es el conjunto dominante encontrado por la heurística y  $D_{gloton}$  es el conjunto dominante encontrado por el algoritmo glotón, entonces para las simulaciones presentadas se cumple:

$$|D_{local}| \leq |D_{gloton}| + c, \quad (4.11)$$

con  $c = 45$ . La diferencia mayor entre  $D_{local}$  y  $D_{gloton}$  es cuando  $D_{local} = 347$  y  $D_{gloton} = 302$ , que es el caso en el que  $n = 800$  y  $grid = 30$ , se observa esta red está bastante desconectada (véase la Figura A.93) y esto demerita el desempeño de la heurística. Lo mismo pasa en los otros casos en los que no se puede superar al algoritmo glotón, pues entre más dispersos estén los vértices más desconectada estará la red, y esto provocará la elección de mayor cantidad de nodos para los conjuntos dominantes en las celdas, lo que

no sucede con el algoritmo glotón, puesto que en el algoritmo no se separan los componentes, lo que si pasa en la heurística.

De todo lo anterior se puede concluir lo siguiente, en la teoría se garantiza que el tamaño del conjunto dominante encontrado por la heurística es a lo más 15 veces el tamaño del conjunto dominante mínimo de una gráfica, las simulaciones indican que el conjunto encontrado por la heurística se encuentra dentro del mismo orden de aproximación que el encontrado por el algoritmo glotón, es decir que las simulaciones sugieren que el factor de aproximación del algoritmo glotón de Chvátal es constante en gráficas de disco unitario, ¿será esto verdad? Responder a esta pregunta sería muy interesante.

## Capítulo 5

# Conclusiones

Las redes ad hoc móviles están compuestas por dispositivos inalámbricos que se comunican entre sí en ausencia de una infraestructura fija. Dos nodos en una red ad hoc se pueden comunicar si la distancia entre estos es menor que el mínimo de sus rangos de transmisión. Por esto, este tipo de redes deben ser capaces de auto-configurarse y auto-organizarse. Además, debido a que los nodos en una red ad hoc pueden moverse de manera aleatoria, los cambios en la topología son frecuentes y no estructurados. Debido a estas características de las redes ad hoc, es de suma importancia que los algoritmos diseñados para las mismas sean de tipo local. Un algoritmo local no necesita conocer toda la topología de la red, y le basta con información local para poder tomar una decisión, que generalmente afecta a la  $k$ -vecindad de un nodo.

El problema del conjunto dominante mínimo es  $NP$ -completo, y el factor de aproximación mas pequeño que puede alcanzar un algoritmo es de  $\ln(\Delta)$ , para una gráfica geométrica. Muchos algoritmos existen en la literatura que obtienen conjuntos dominantes en redes ad hoc, puesto que encontrar este conjunto se ha convertido en la base para muchos algoritmos de búsqueda de rutas. La mayoría de estos algoritmos alcanzan el factor de aproximación óptimo, sin embargo, hasta la fecha no se ha presentado un algoritmo que obtenga un conjunto dominante de manera local. En este trabajo se presentó una heurística que calcula un conjunto dominante en un red ad hoc, de manera local. La heurística alcanza el siguiente factor de aproximación

$$|D| \leq 15|OPT|, \quad (*)$$

donde  $D$  es el conjunto dominante calculado por la heurística y  $OPT$  es el conjunto dominante minimal de  $GDU(\mathcal{N})$ , de una red ad hoc  $\mathcal{N}$ .

Nótese que (\*) alcanza el factor de aproximación óptimo cuando

$$\begin{aligned}\ln(\Delta) &= 15 \\ \Delta &= e^{15}\end{aligned}$$

y esto sucede para gráficas con grado muy alto. Sin embargo, cabe destacar que esta cota encontrada no es una cota justa, y lo único que se puede afirmar, por el Teorema 9, es que el factor de aproximación de la heurística es constante, y que dicha constante es menor o igual que 15. Sería sumamente interesante encontrar una cota justa para el factor de aproximación de la heurística local presentada en este trabajo.

En las simulaciones se comparó el tamaño del conjunto dominante encontrado por la heurística, con el encontrado por el algoritmo glotón de Chvátal. Para los casos en que la red es de 500 puntos con grid 15 y 20, y de tamaño 800 con grid 20 y 30, la heurística no superó al algoritmo glotón. Sin embargo, la heurística encuentra un conjunto dominante de menor tamaño en todos los demás casos. Sin embargo, siempre se puede escribir que:

$$|D_{local}| \leq |D_{gloton}| + c.$$

Es decir, las simulaciones muestran que ambos algoritmos se encuentran dentro del mismo orden de aproximación. Por lo tanto, las simulaciones sugieren que el factor de aproximación del algoritmo glotón es constante para gráficas de disco unitario. Un problema abierto es demostrar que esto es así o probar lo contrario.

En las simulaciones la heurística, al usar la subdivisión cuadrada, encontró el conjunto dominante de menor tamaño en la mayoría de los casos, para redes de hasta 200 nodos. Por otro lado, para redes de 500 y 800 nodos, la subdivisión hexagonal funcionó mejor.

Es importante mencionar que el conjunto dominante calculado por la heurística no es conexo. Sin embargo, la conexidad es una característica importante para que los conjuntos dominantes puedan utilizarse en algoritmos de búsqueda de rutas. Sería interesante buscar una manera de extender esta heurística para que calcule un conjunto dominante conexo. Una posible manera de hacer esto es calculando primero el conjunto dominante como hasta ahora, y después intentar conectarlo, puesto que para hacer esto sería necesario tomar en cuenta solamente las regiones vecinas, que sin importar el tipo de subdivisión que se use, es un número constante.

## Apéndice A

# Redes simuladas

En este apéndice se han incluido las figuras de todas las redes generadas para las simulaciones, usando las diferentes subdivisiones y tamaños de aristas descritos en la sección 4.5. En cada una de las figuras se muestran en rojo los vértices que pertenecen al conjunto dominante y en azul los que no pertenecen. En los pies de figura se indica el tamaño de la red, de grid, el tipo de subdivisión y el tamaño de las aristas.

## A.1. Redes de tamaño cien

### A.1.1. Grid de tamaño cinco

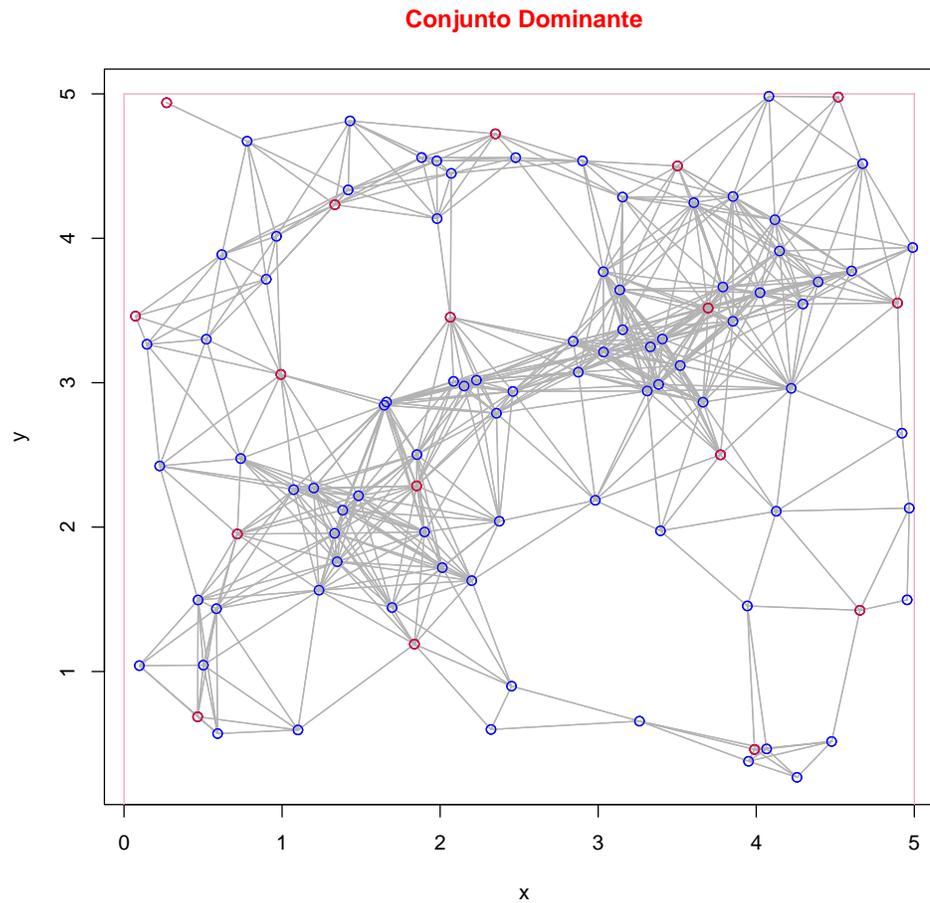


Figura A.1: Resultado del algoritmo glotón para una red de tamaño 100 y grid 5.

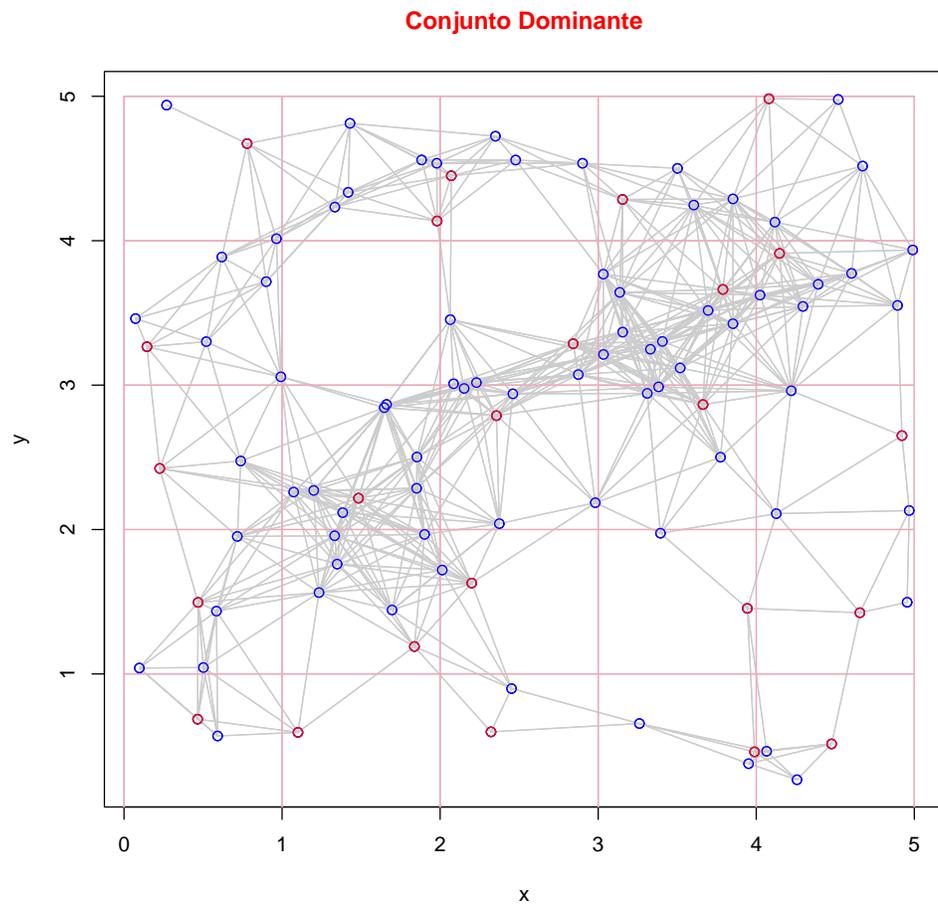


Figura A.2: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión cuadrada y aristas de 1cm.

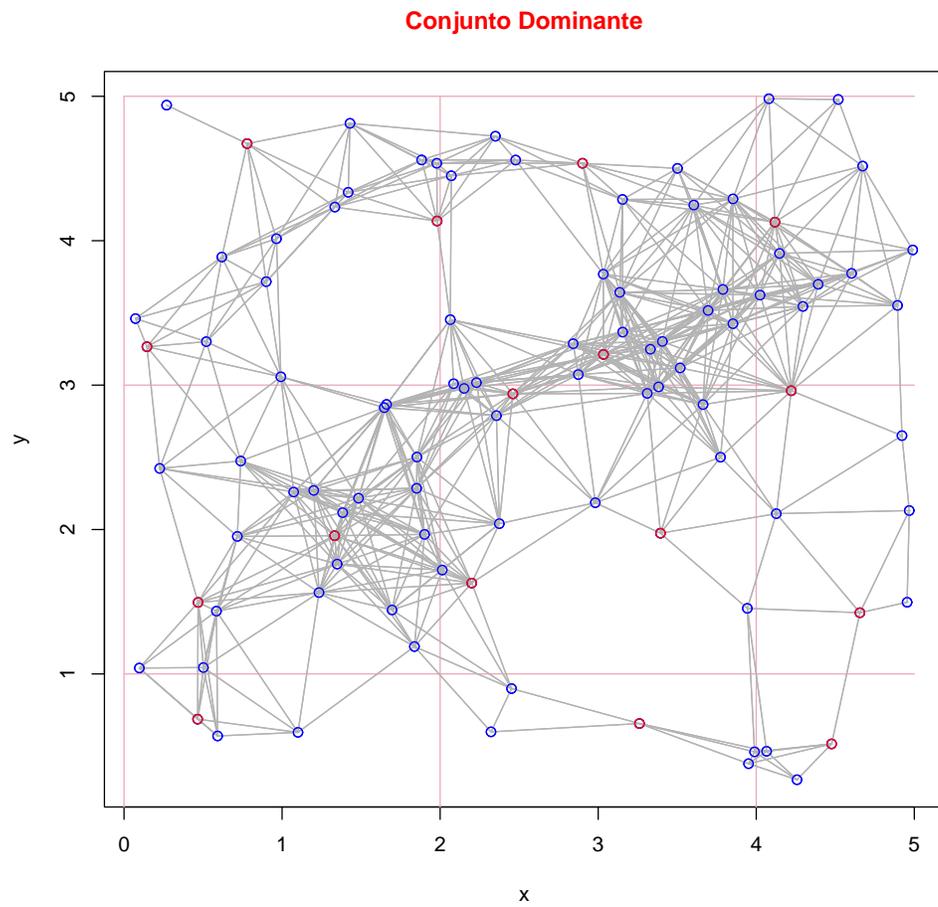


Figura A.3: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión cuadrada y aristas de 2cm.

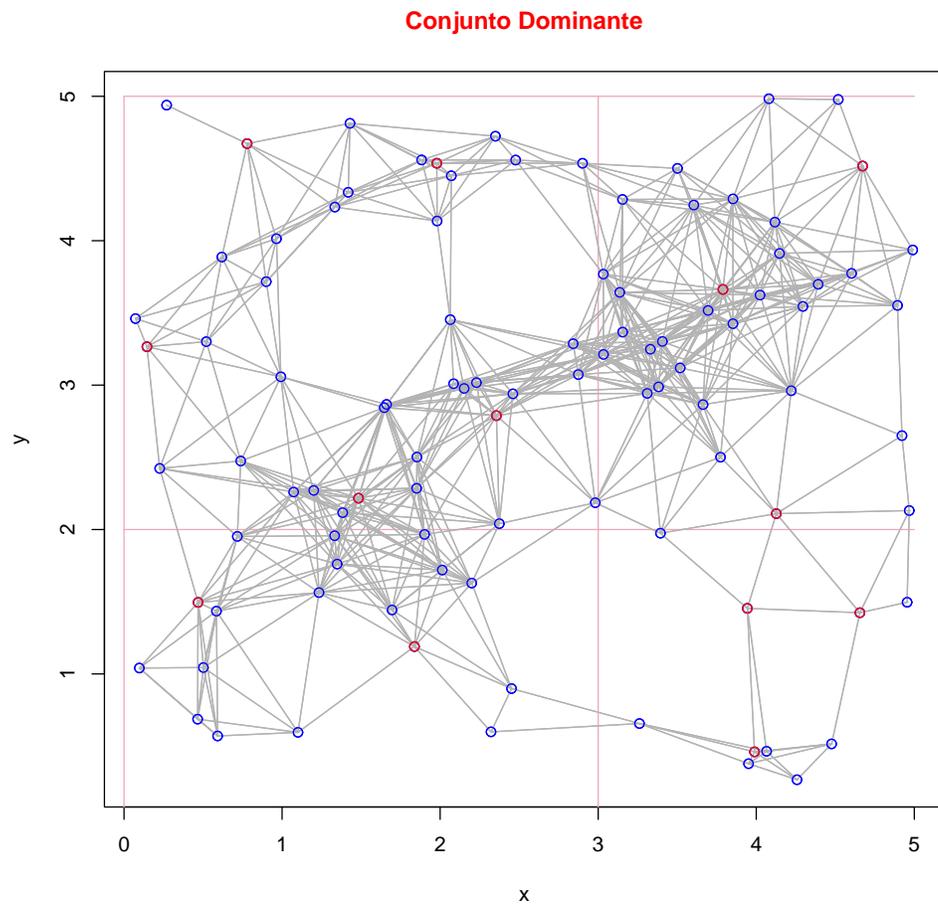


Figura A.4: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión cuadrada y aristas de 3cm.

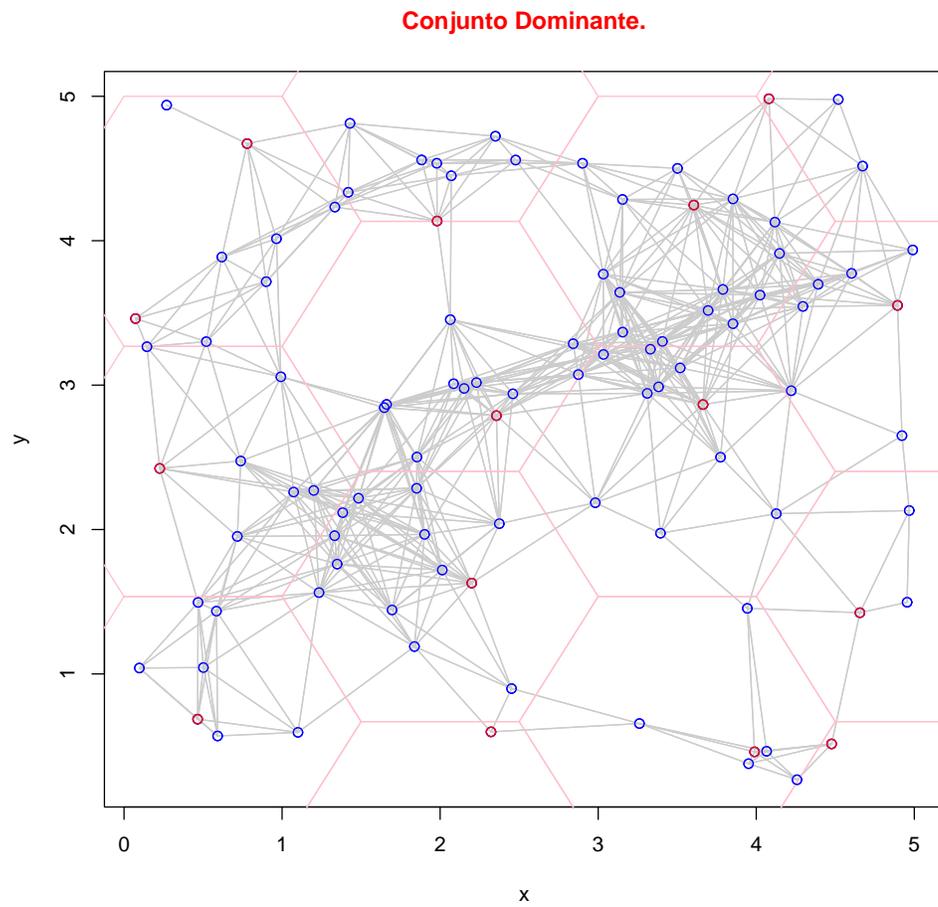


Figura A.5: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión hexagonal y aristas de  $1cm$ .

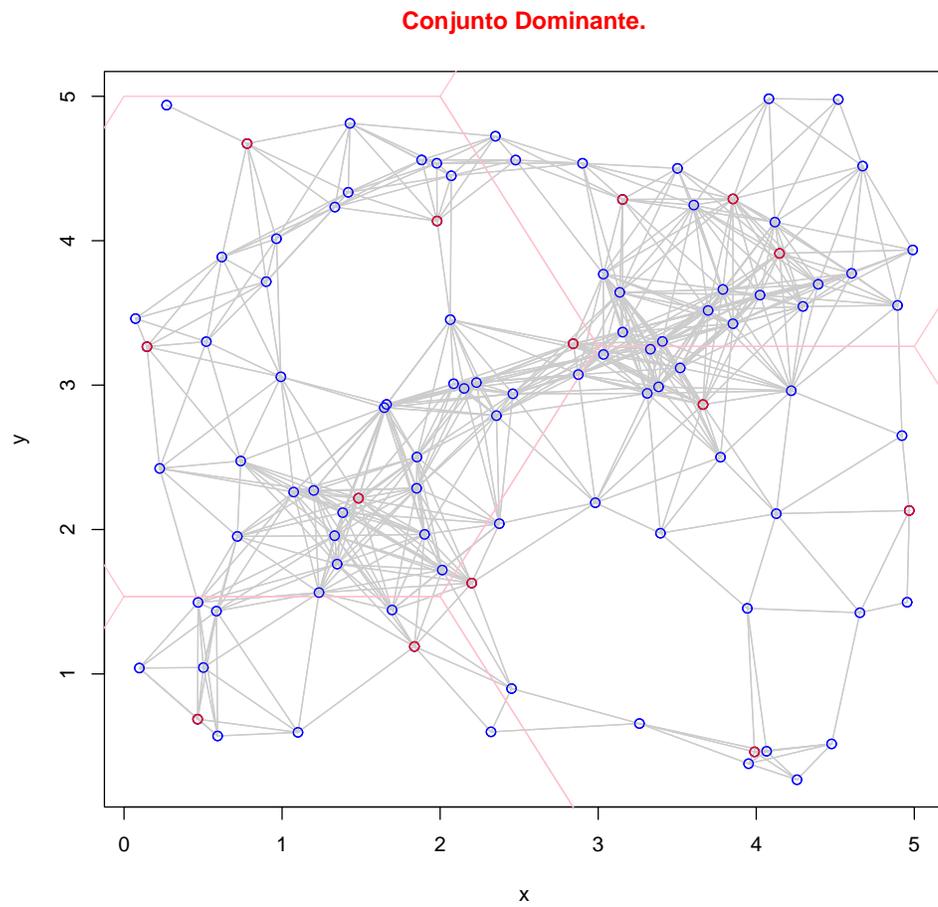


Figura A.6: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión hexagonal y aristas de 2cm.

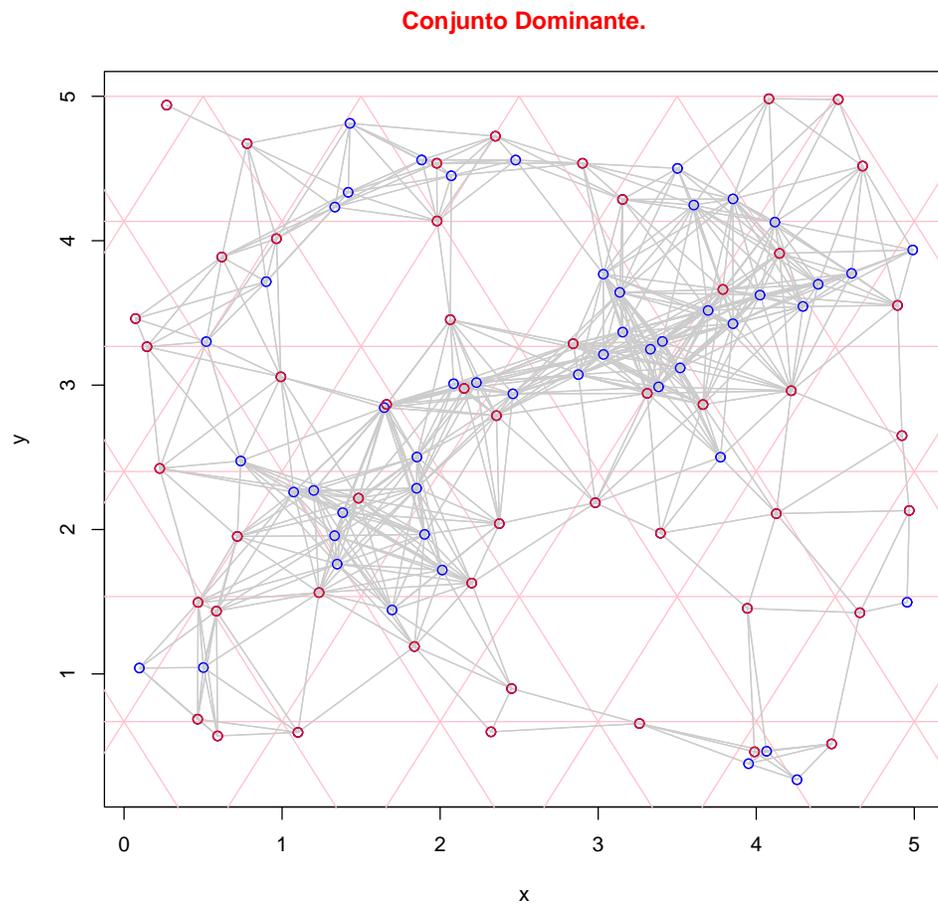


Figura A.7: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión triangular y aristas de 1cm.

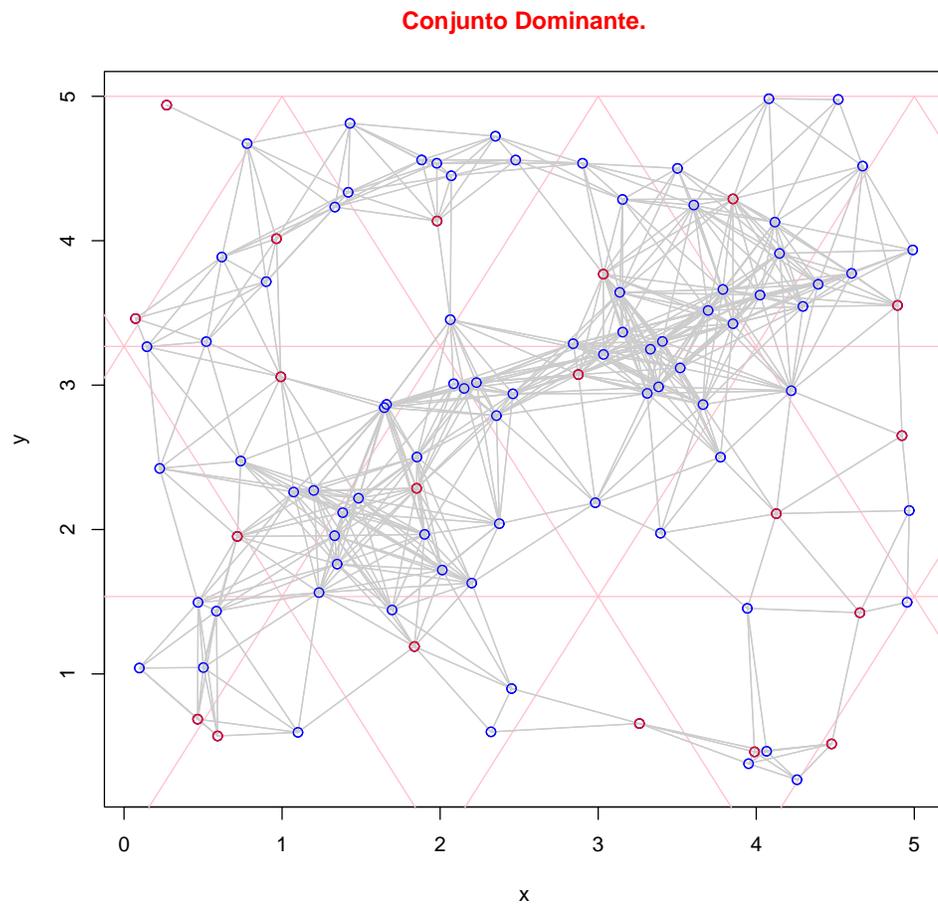


Figura A.8: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión triangular y aristas de 2cm.

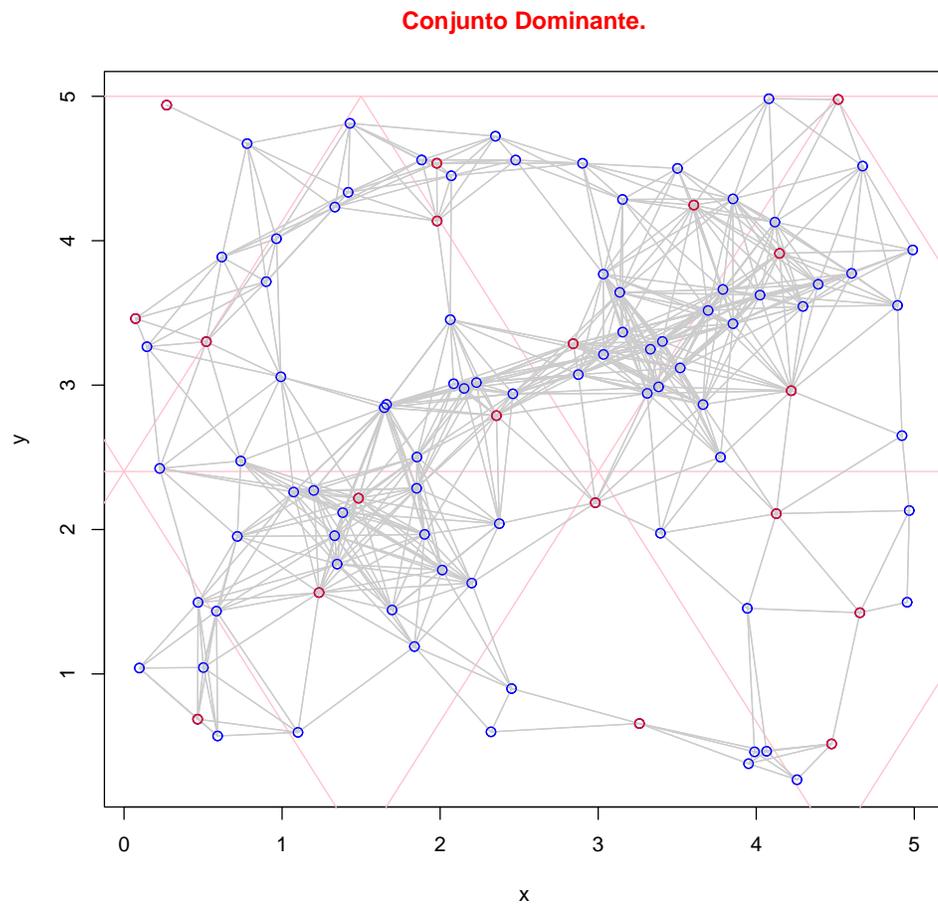


Figura A.9: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión triangular y aristas de 3cm.

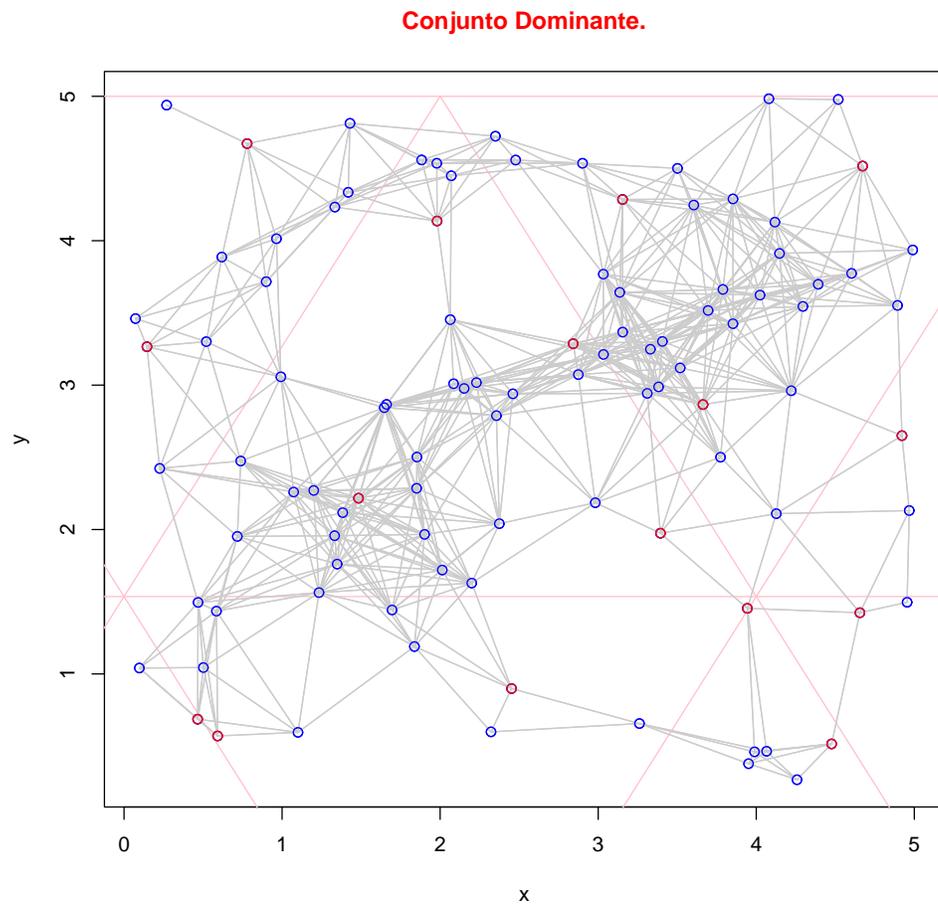


Figura A.10: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión triangular y aristas de  $4cm$ .

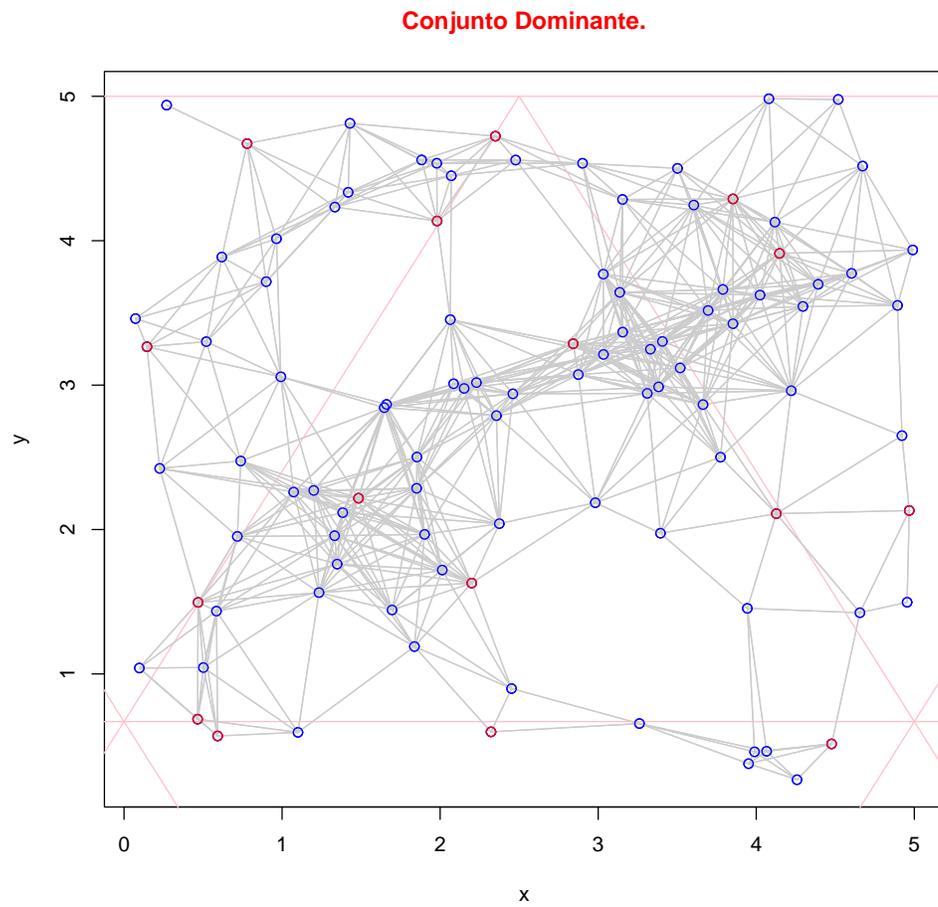


Figura A.11: Resultado de la heurística para una red de tamaño 100 y grid 5, con subdivisión triangular y aristas de  $5cm$ .

## A.1.2. Grid de tamaño seis

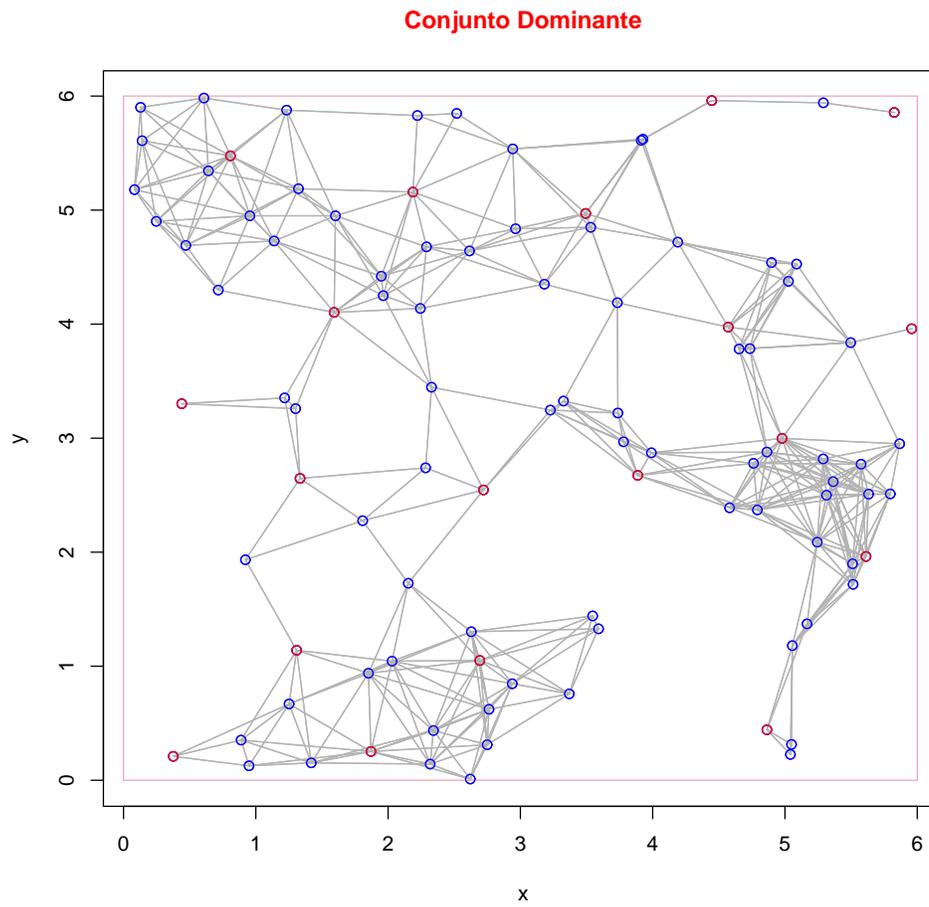


Figura A.12: Resultado del algoritmo glotón para una red de tamaño 100 y grid 6.

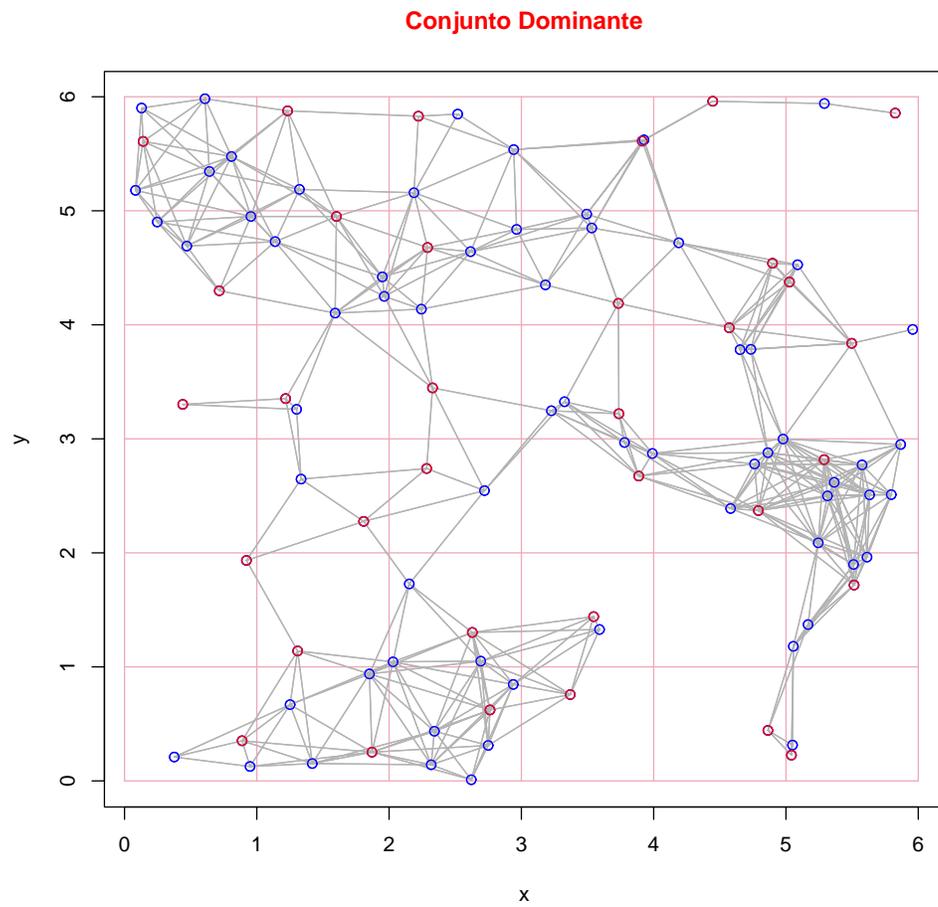


Figura A.13: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión cuadrada y aristas de  $1cm$ .

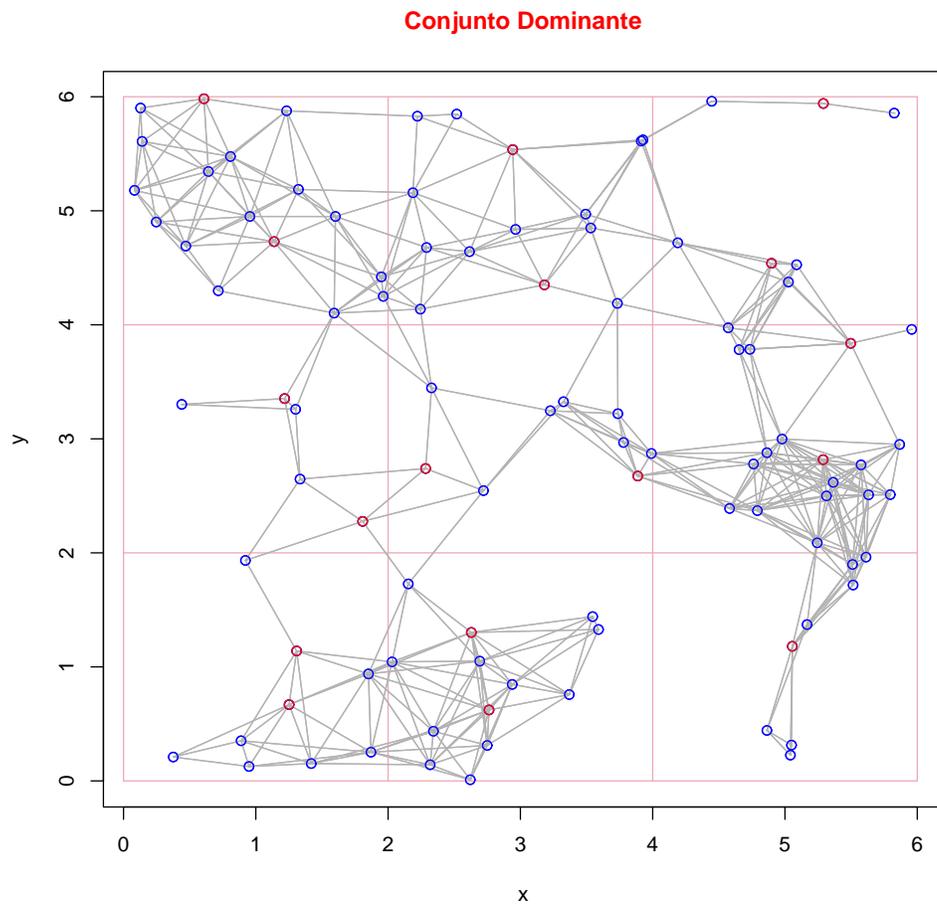


Figura A.14: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión cuadrada y aristas de 2cm.

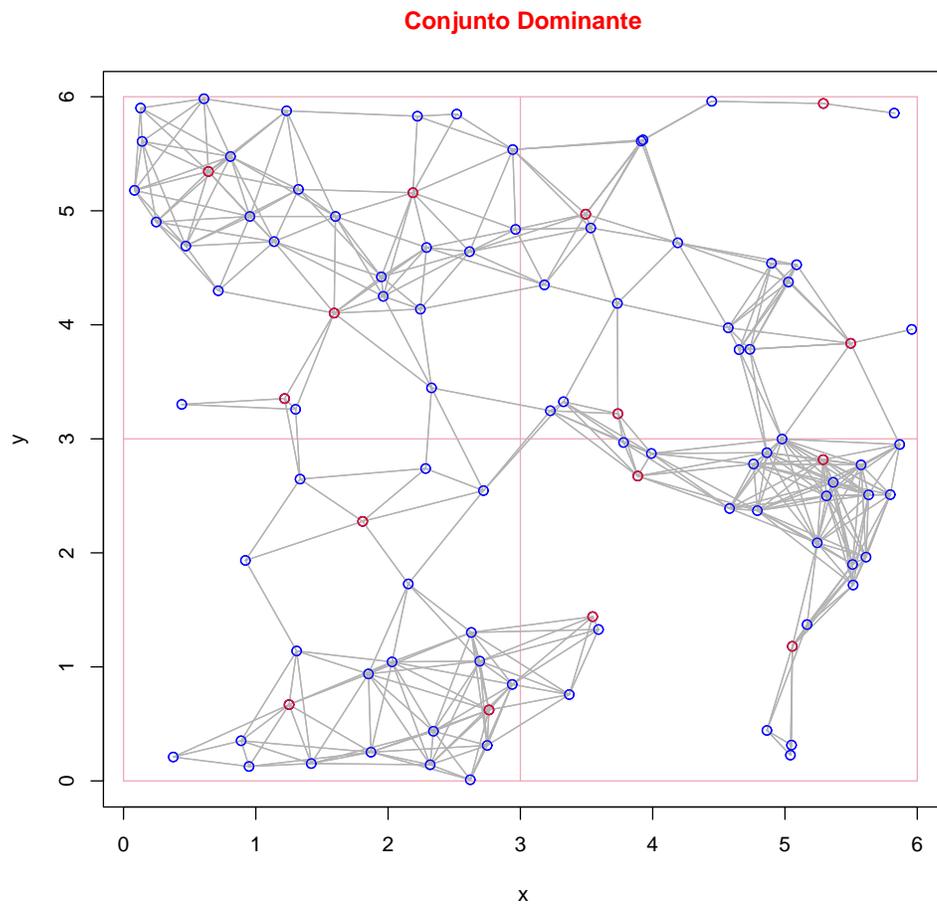


Figura A.15: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión cuadrada y aristas de  $3cm$ .

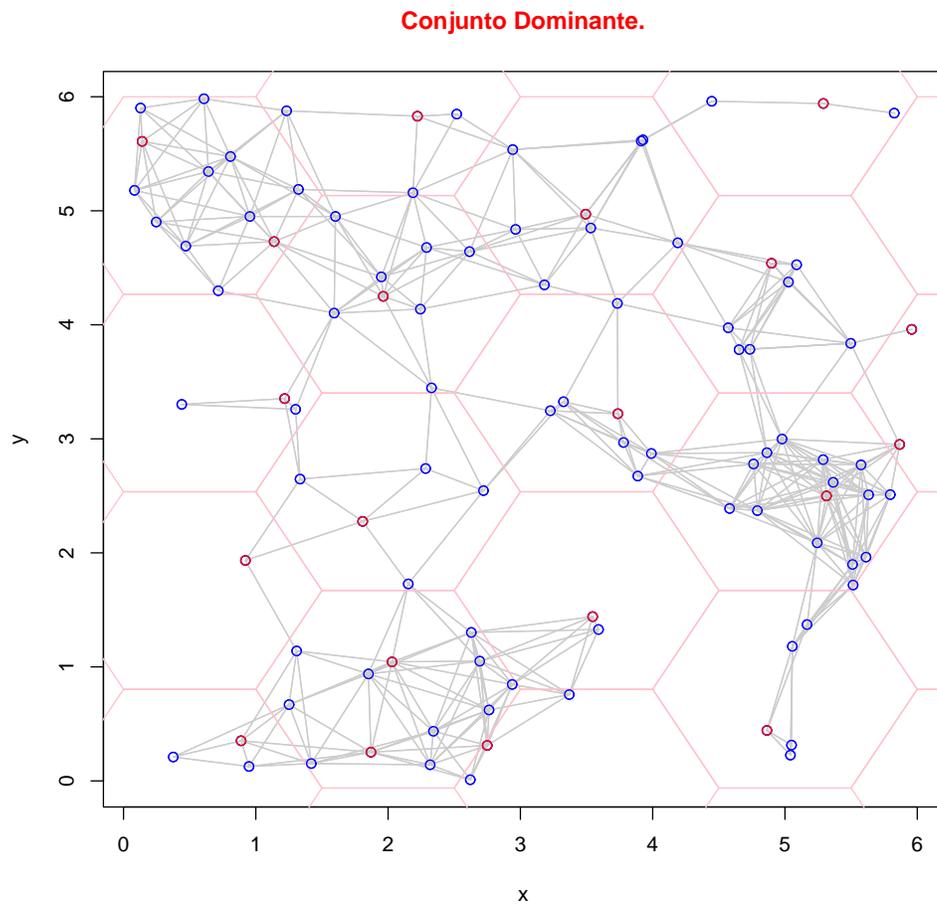


Figura A.16: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión hexagonal y aristas de 1cm.

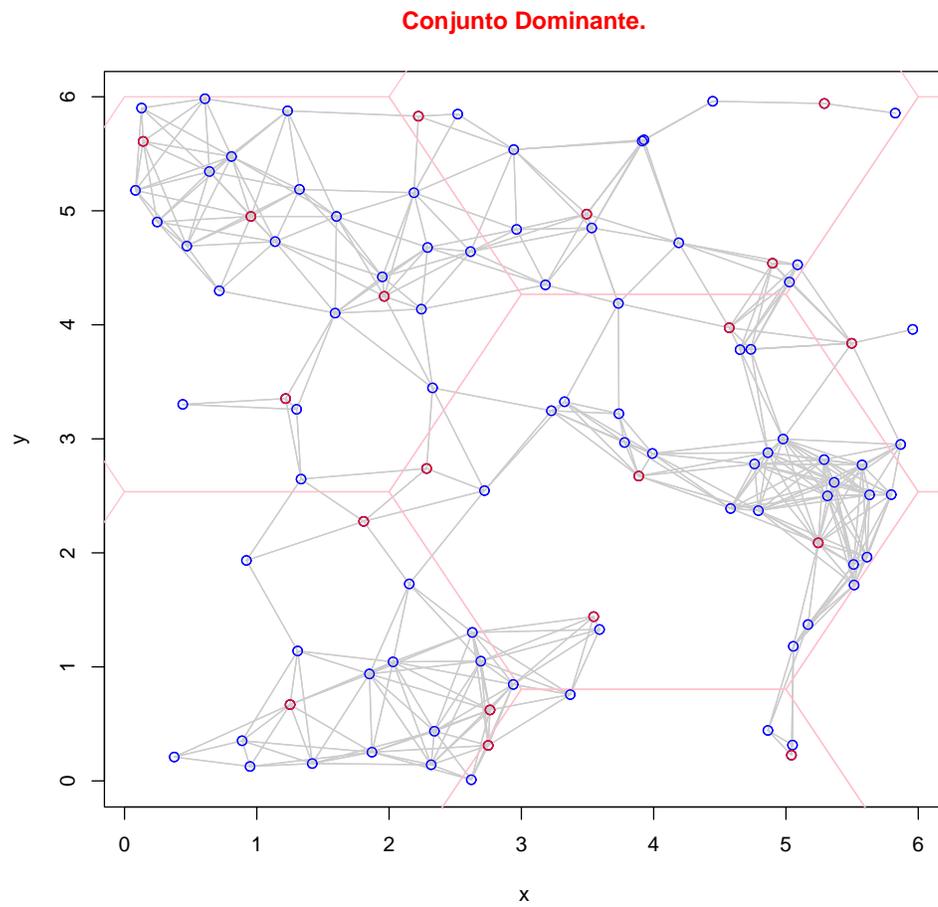


Figura A.17: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión hexagonal y aristas de 2cm.

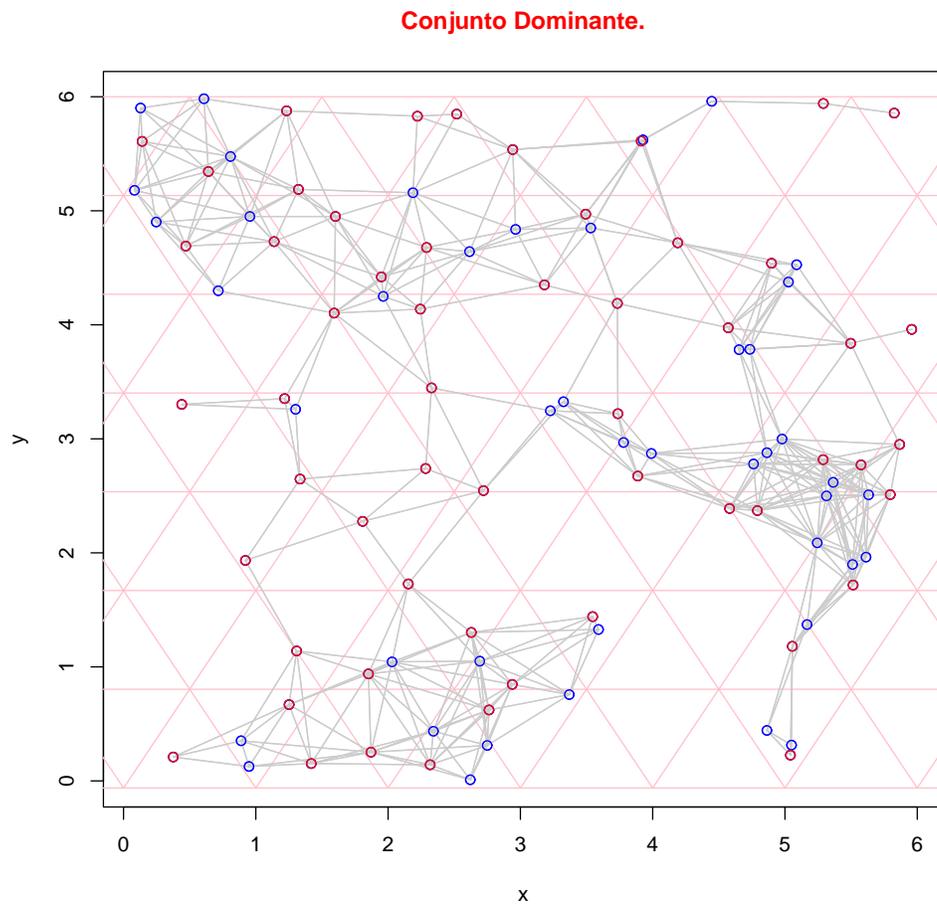


Figura A.18: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión triangular y aristas de  $1cm$ .

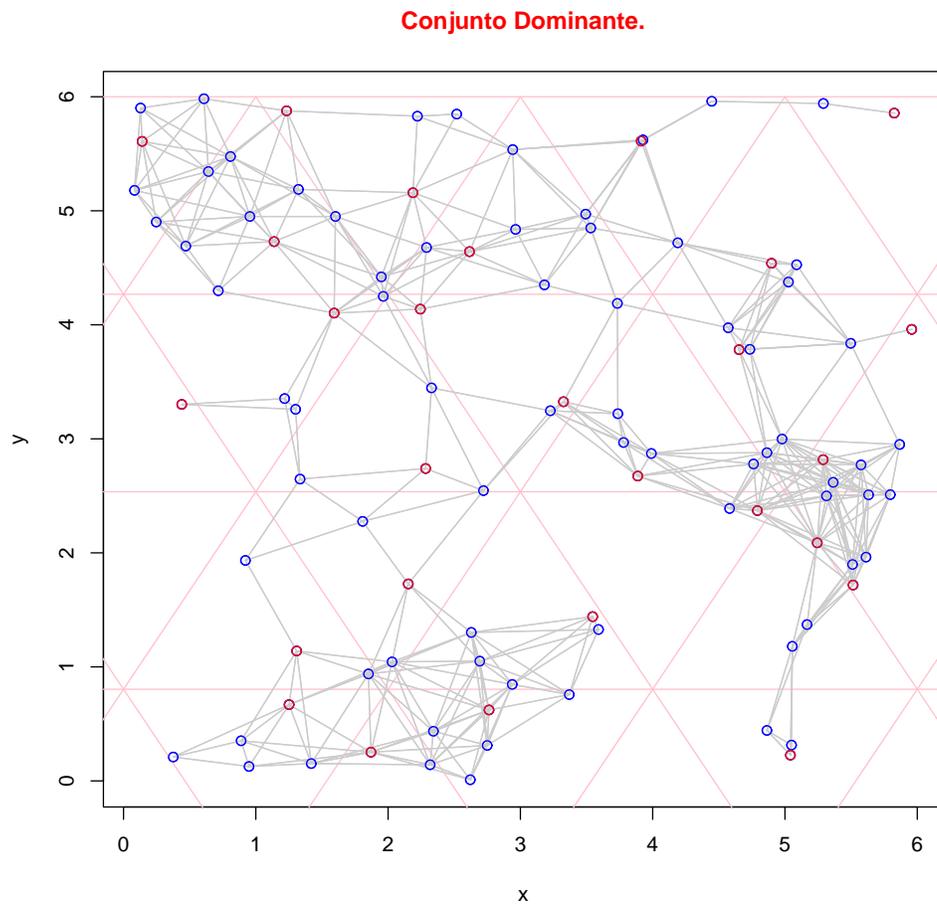


Figura A.19: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión triangular y aristas de 2cm.

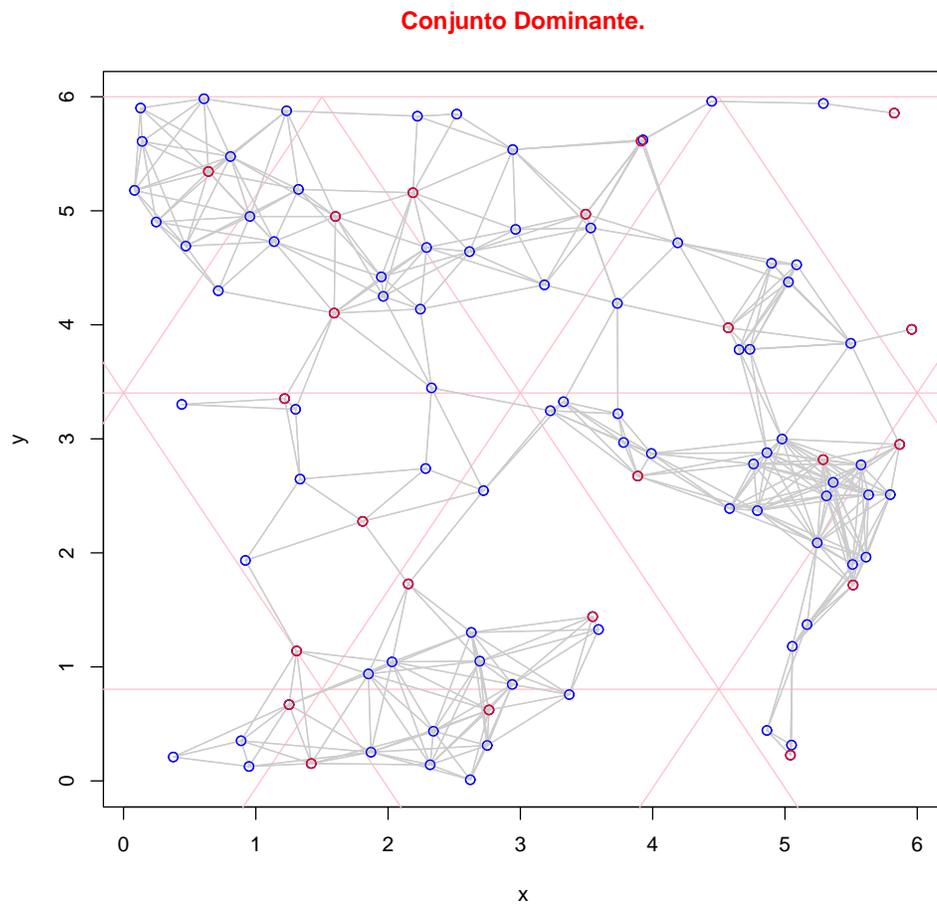


Figura A.20: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión triangular y aristas de  $3cm$ .

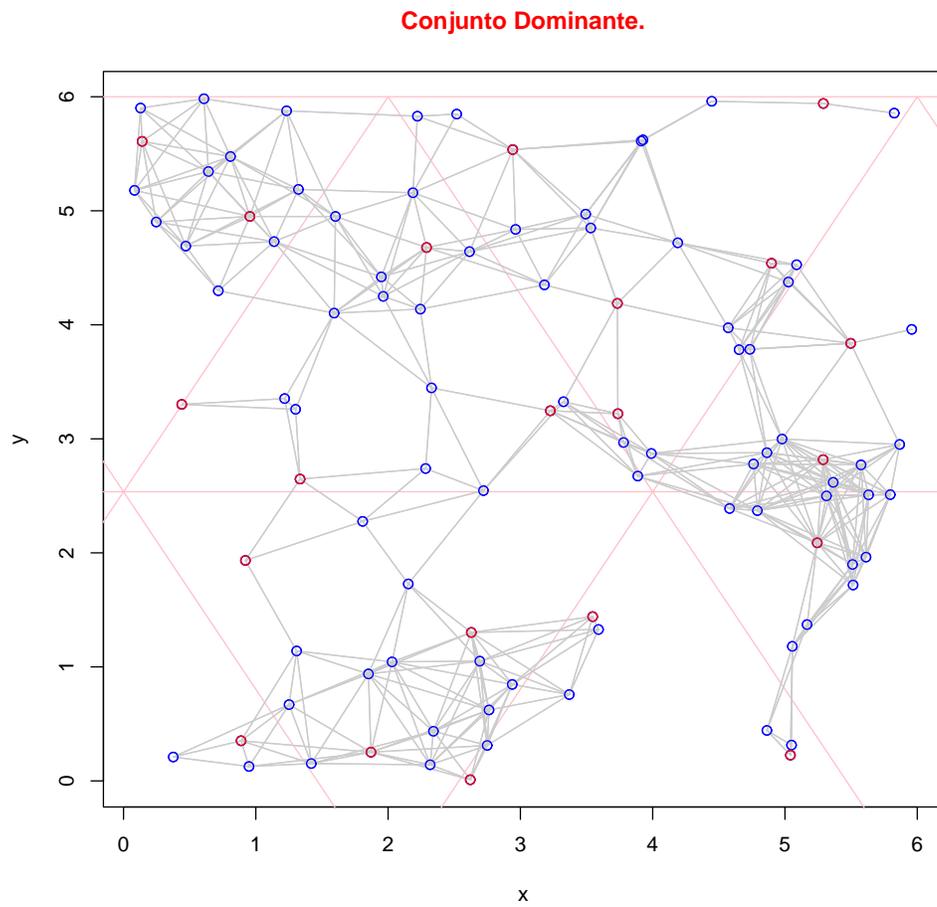


Figura A.21: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión triangular y aristas de 4cm.

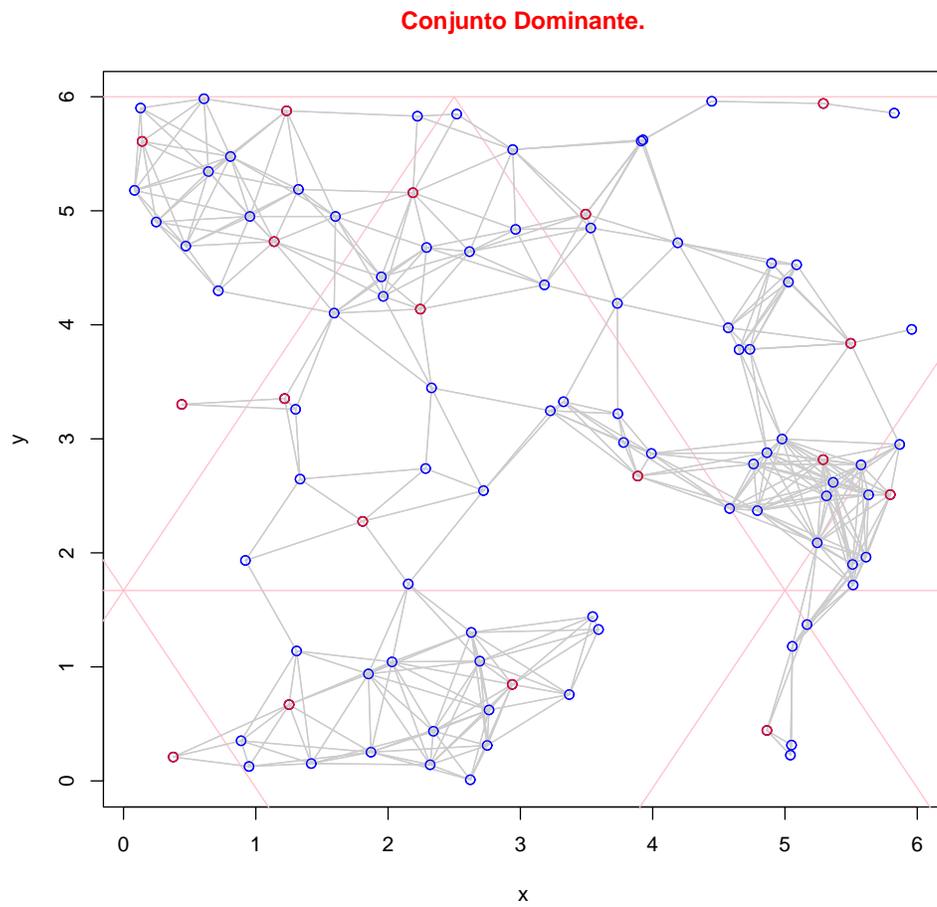


Figura A.22: Resultado de la heurística para una red de tamaño 100 y grid 6, con subdivisión triangular y aristas de 5cm.

## A.1.3. Grid de tamaño siete

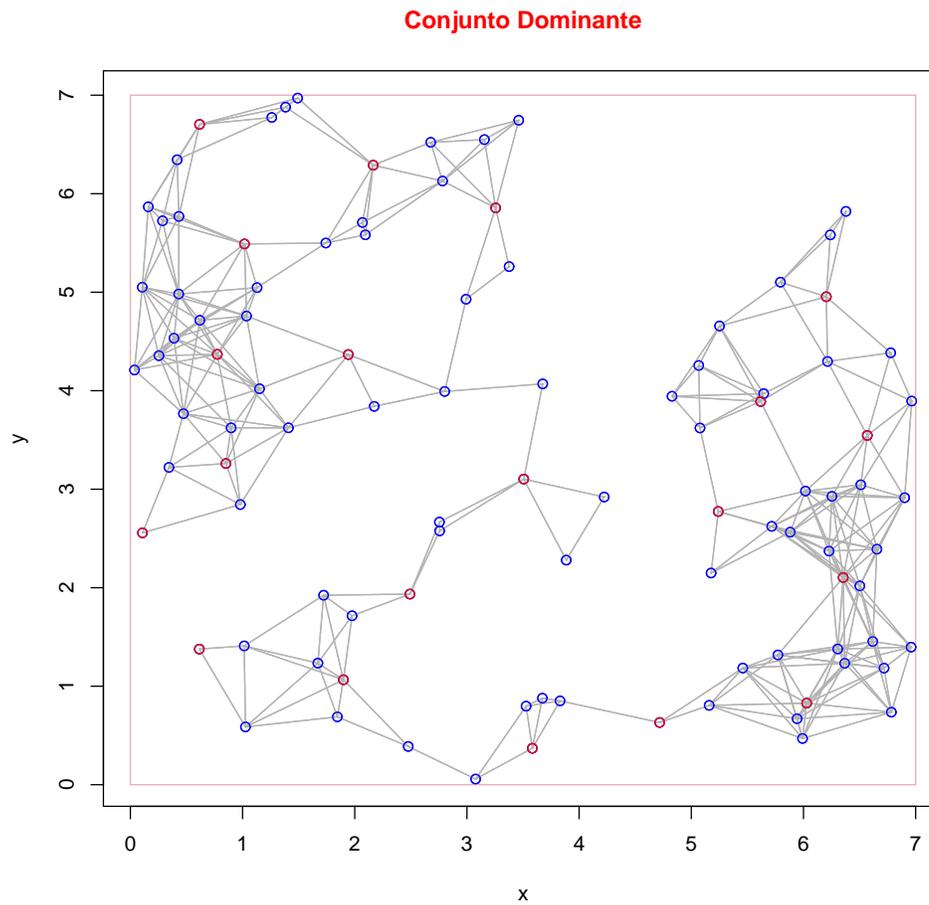


Figura A.23: Resultado del algoritmo glotón para una red de tamaño 100 y grid 7.

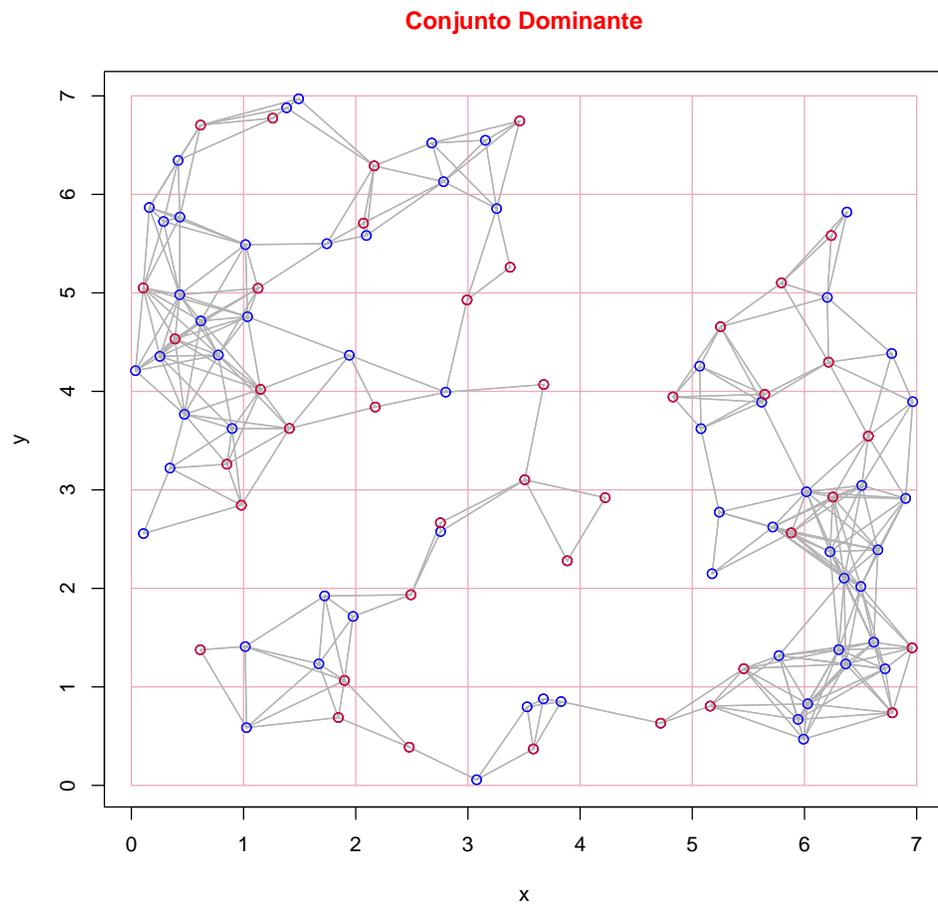


Figura A.24: Resultado del algoritmo glotón para una red de tamaño 100 y grid 7, con subdivisión cuadrada y aristas de 1cm.

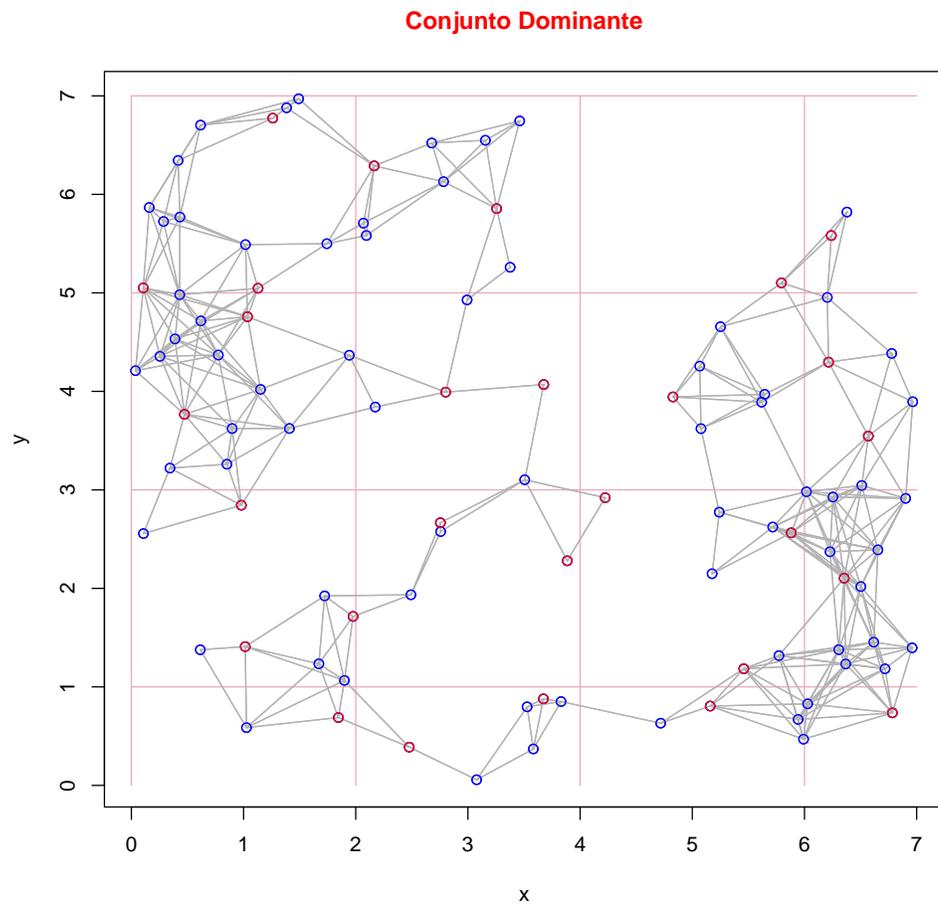


Figura A.25: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión cuadrada y aristas de 2cm.

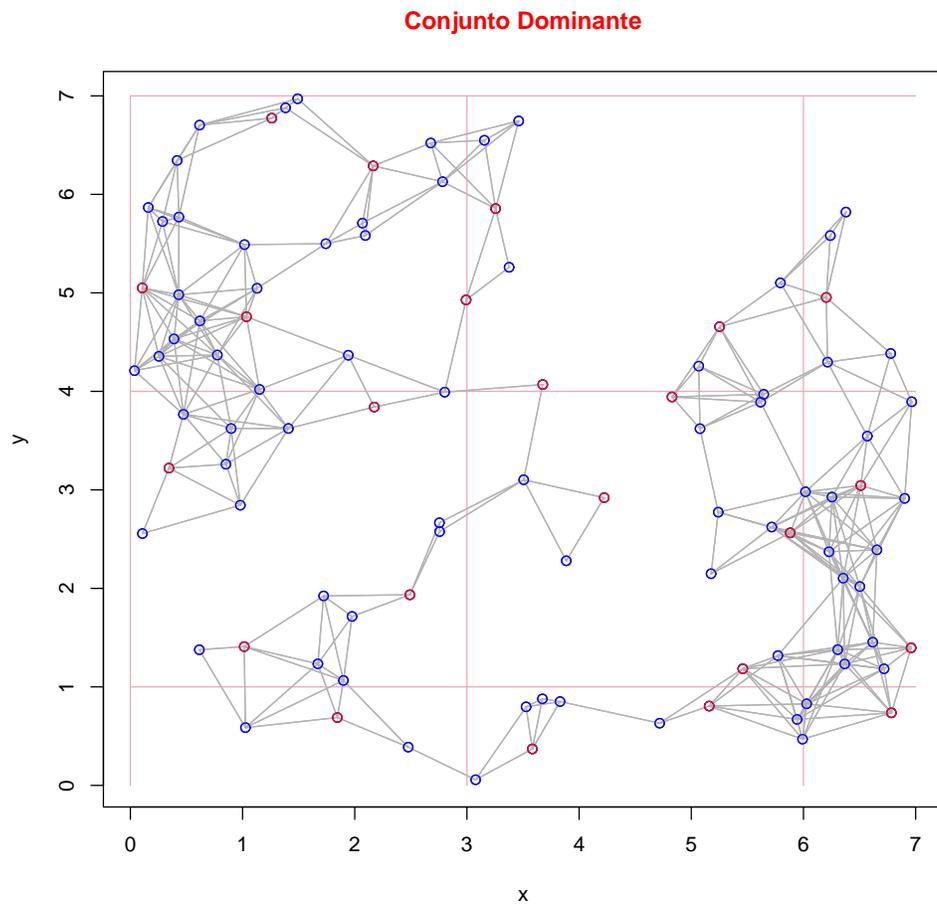


Figura A.26: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión cuadrada y aristas de 3cm.

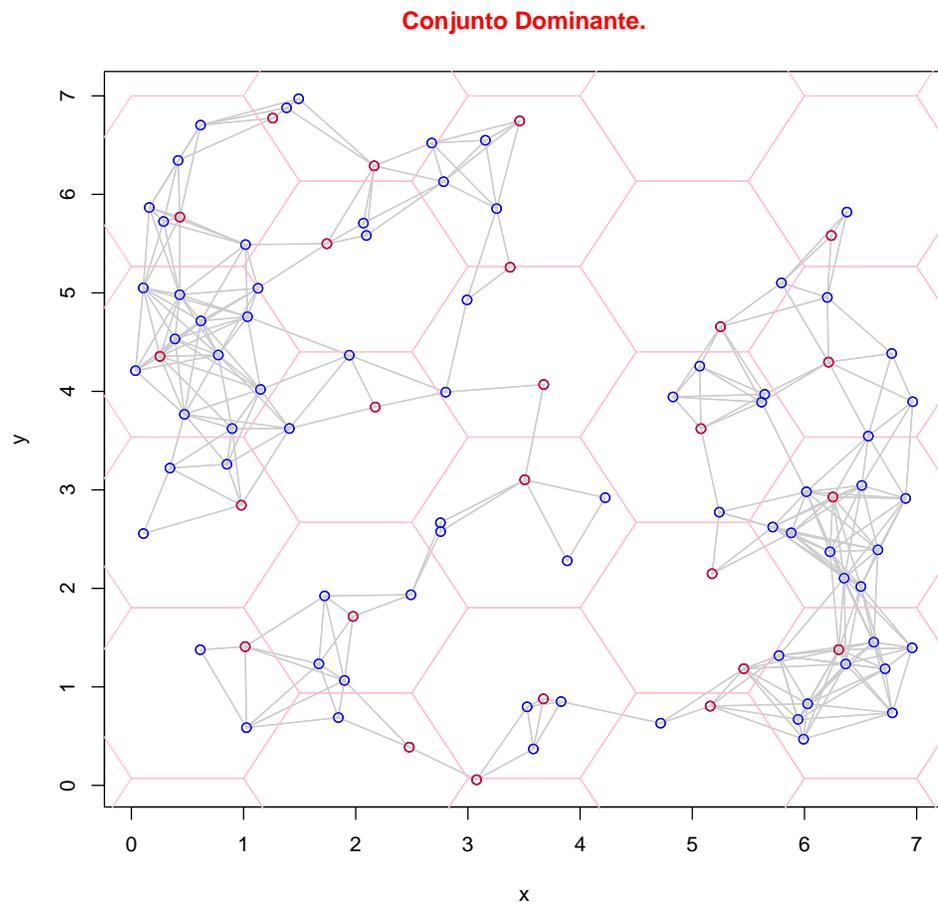


Figura A.27: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión hexagonal y aristas de  $1\text{cm}$ .

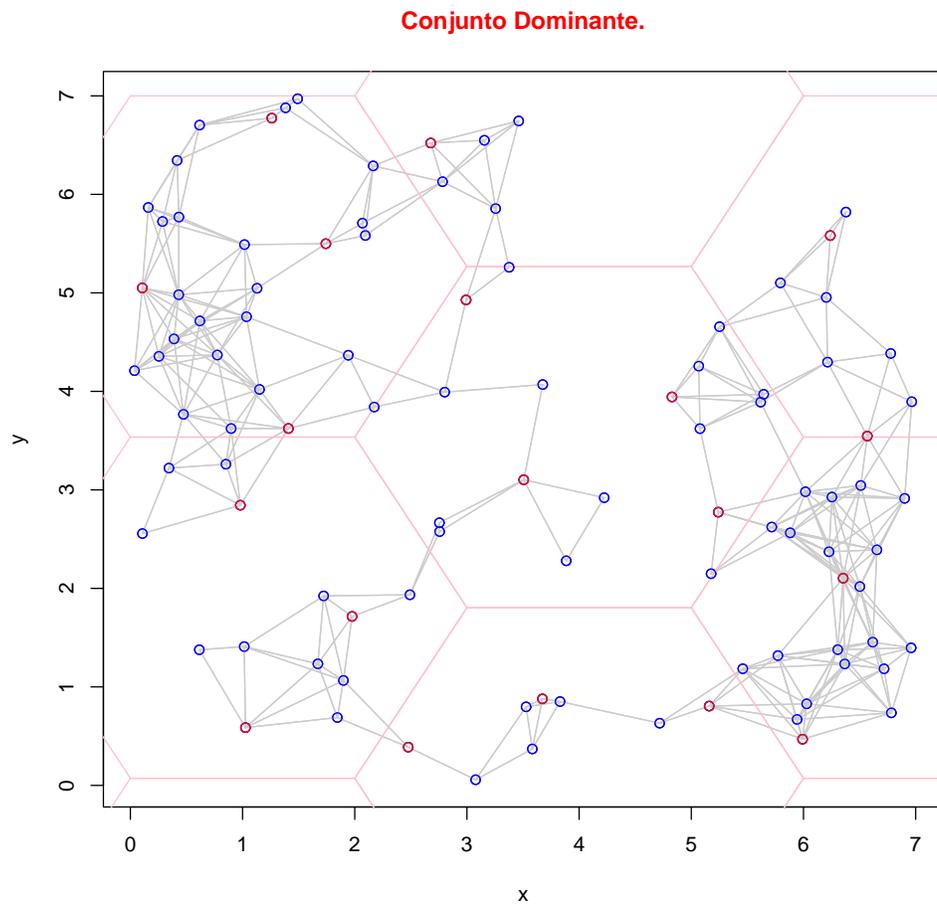


Figura A.28: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión hexagonal y aristas de  $2\text{cm}$ .

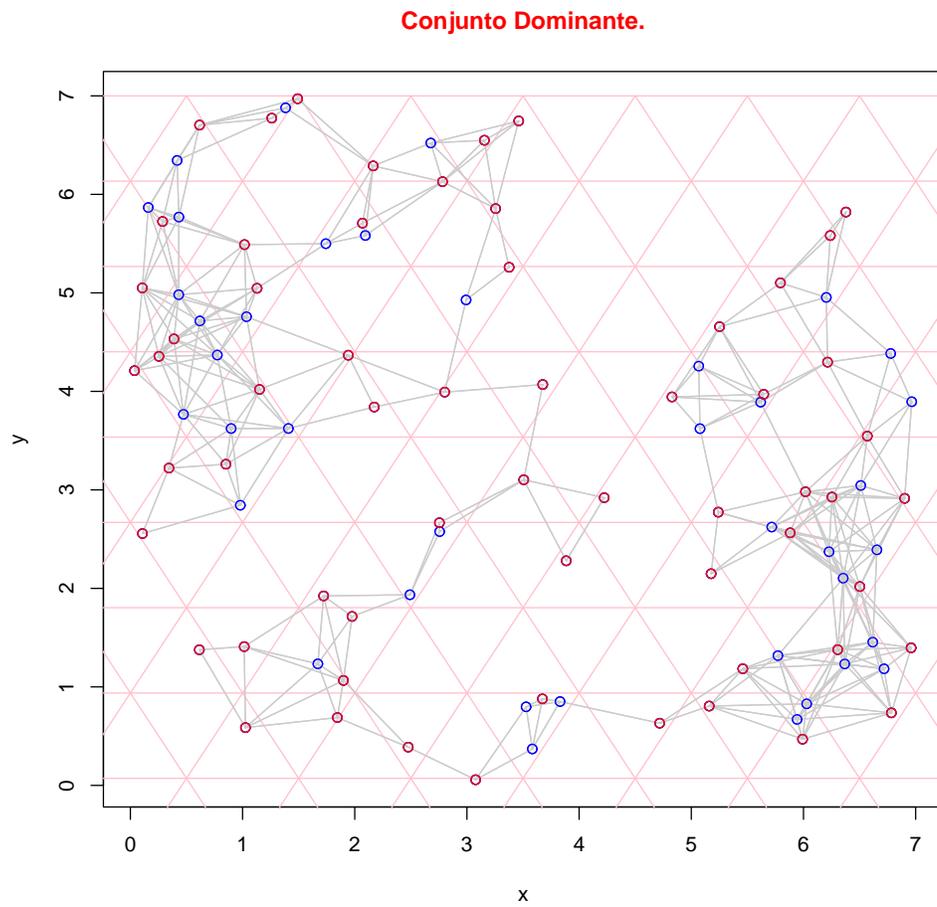


Figura A.29: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión triangular y aristas de  $1cm$ .

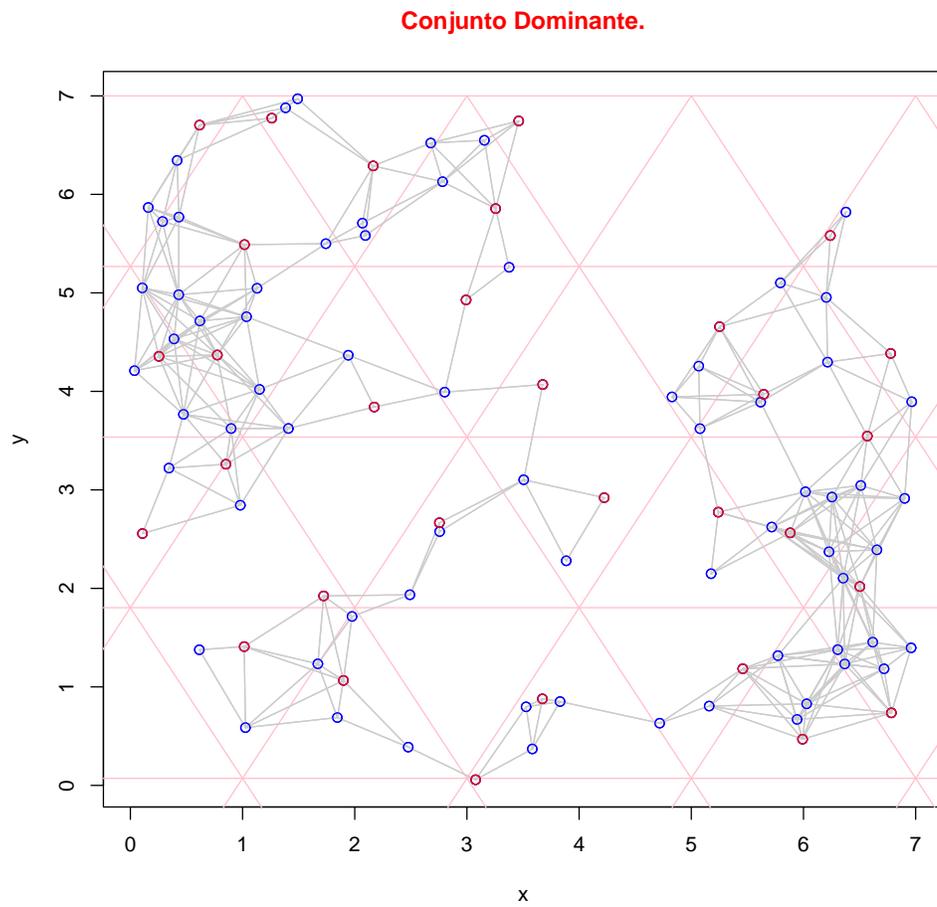


Figura A.30: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión triangular y aristas de  $2cm$ .

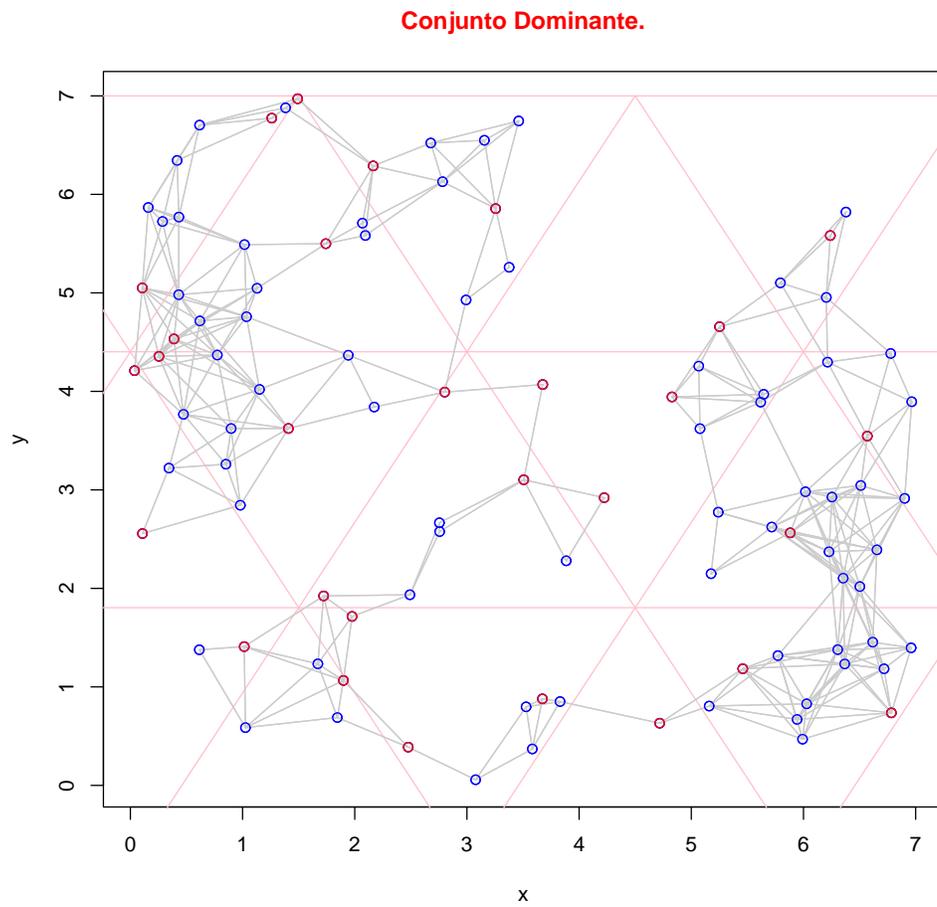


Figura A.31: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión triangular y aristas de 3cm.

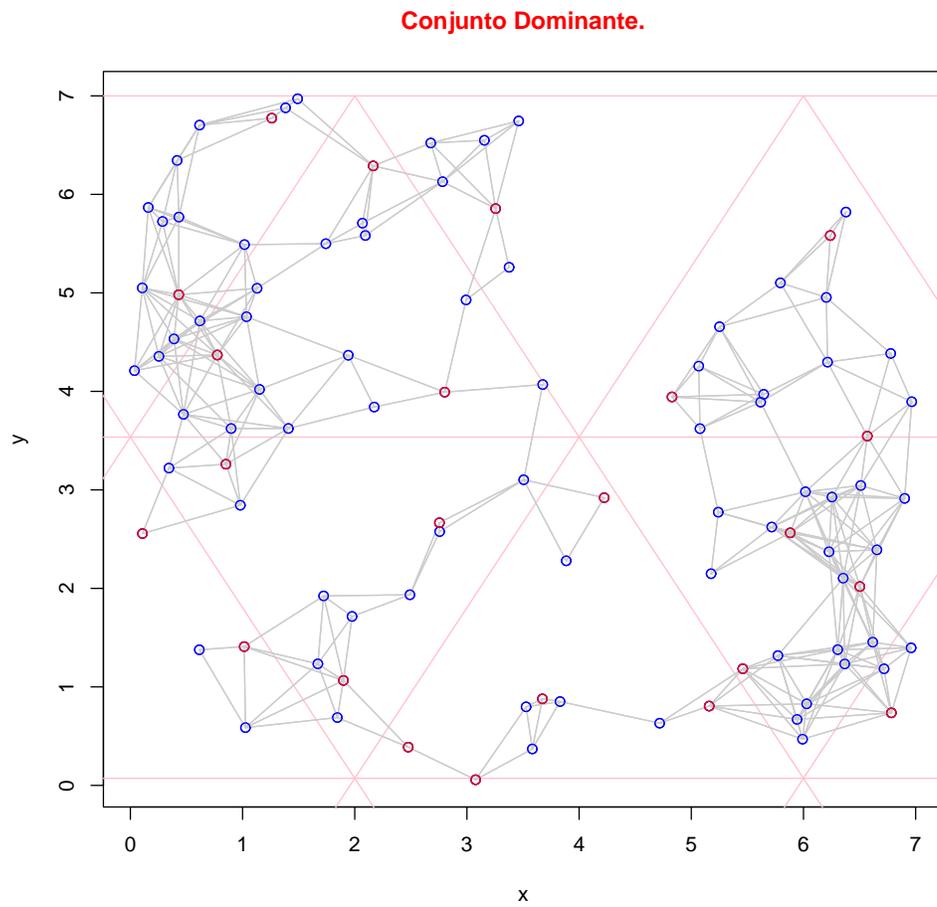


Figura A.32: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión triangular y aristas de 4cm.

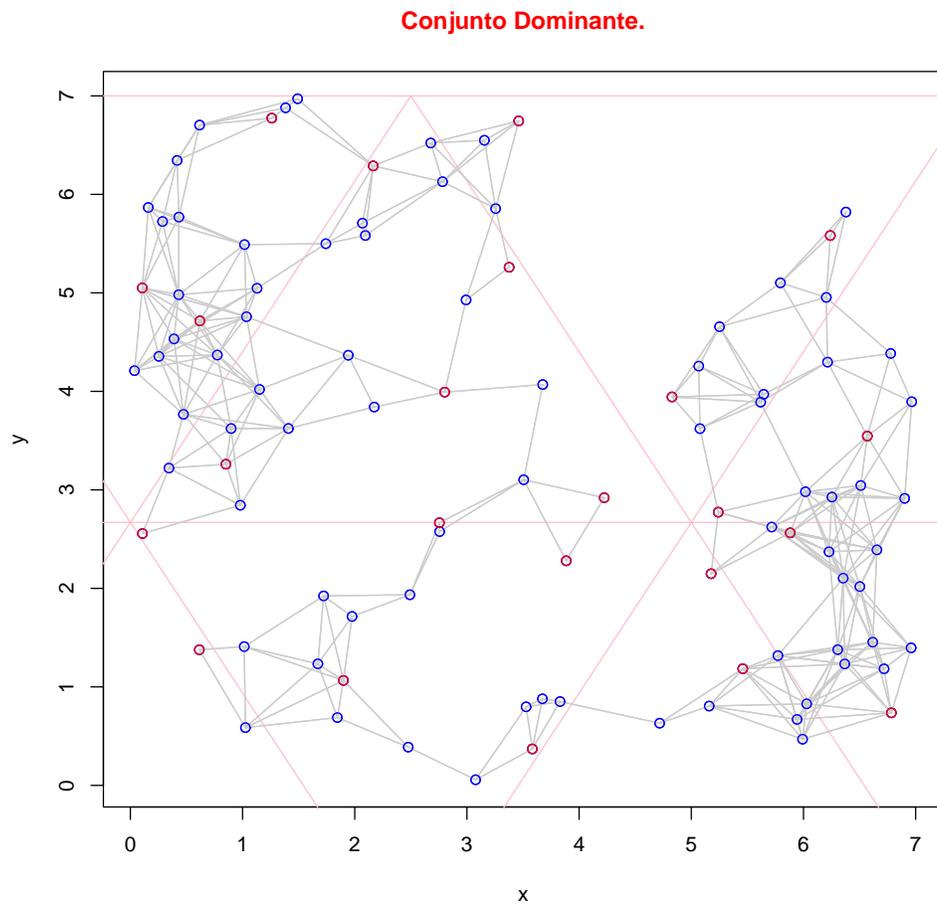


Figura A.33: Resultado de la heurística para una red de tamaño 100 y grid 7, con subdivisión triangular y aristas de 5cm.

## A.2. Redes de tamaño doscientos

### A.2.1. Grid de tamaño cinco

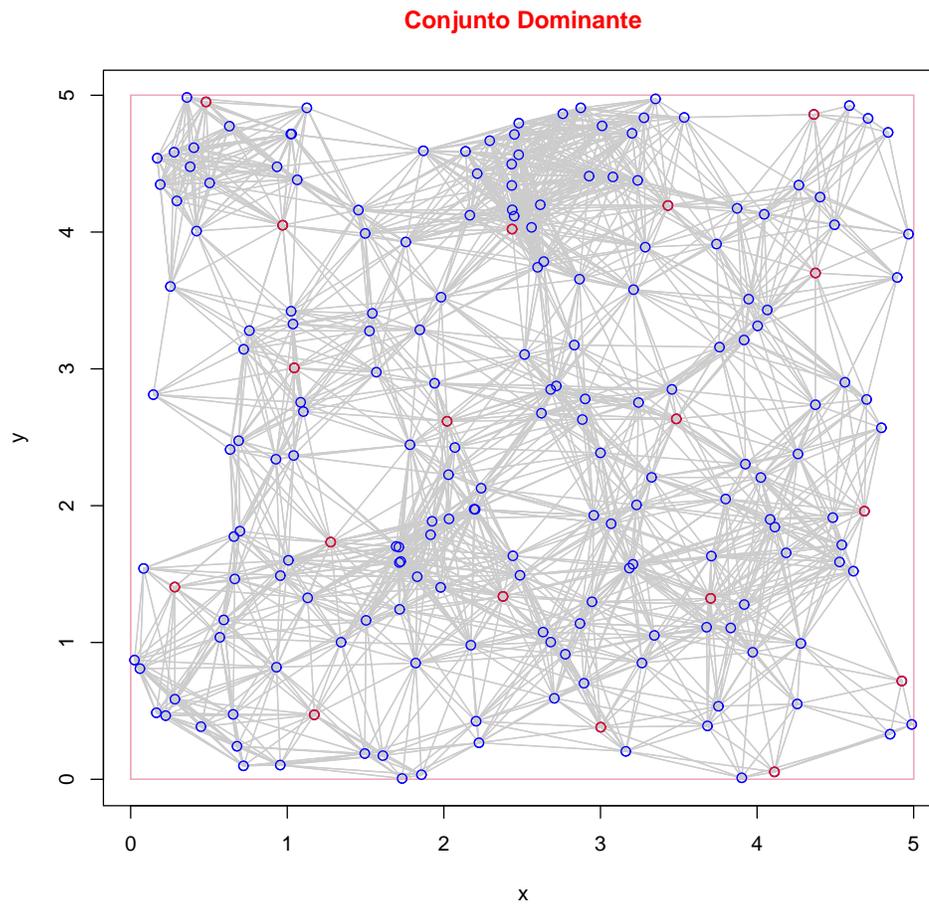


Figura A.34: Resultado del algoritmo glotón para una red de tamaño 200 y grid 5.

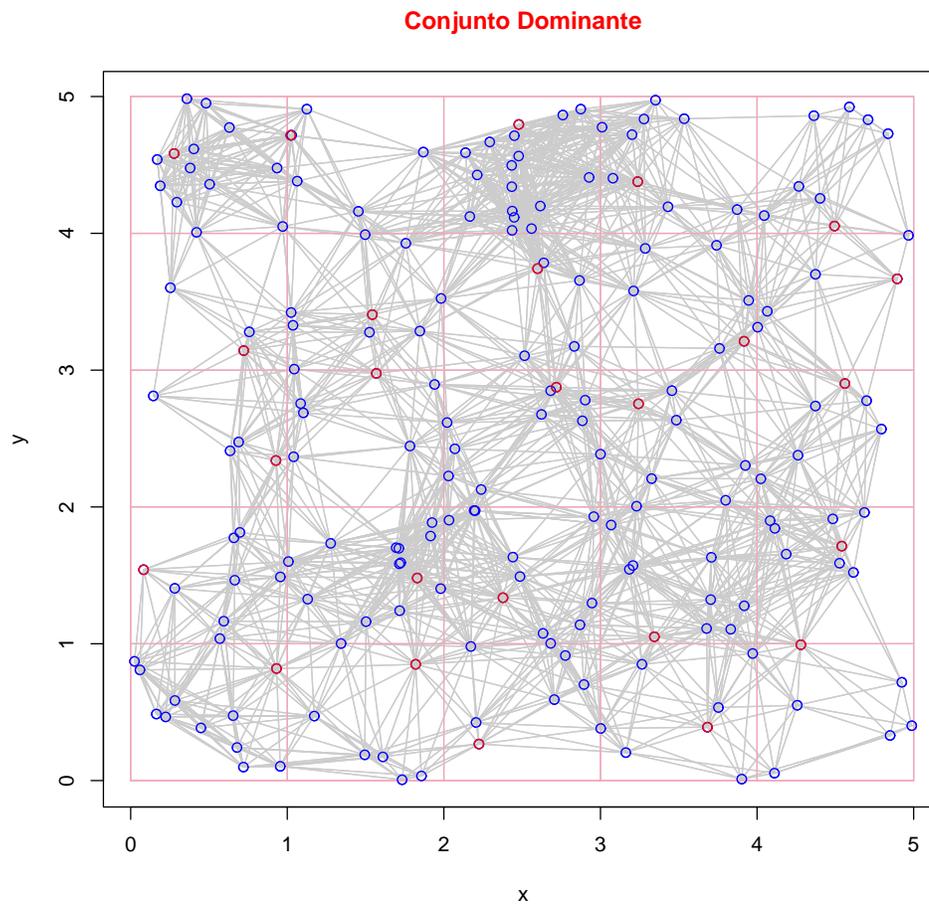


Figura A.35: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión cuadrada y aristas de  $1cm$ .

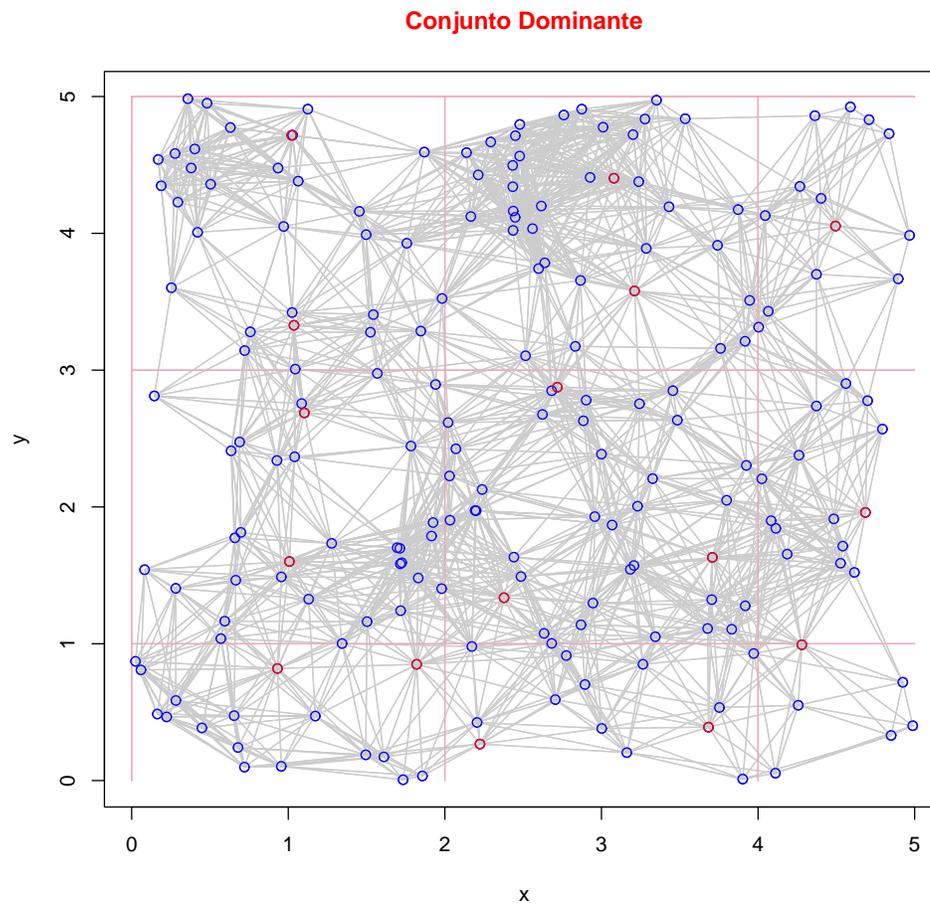


Figura A.36: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión cuadrada y aristas de 2cm.

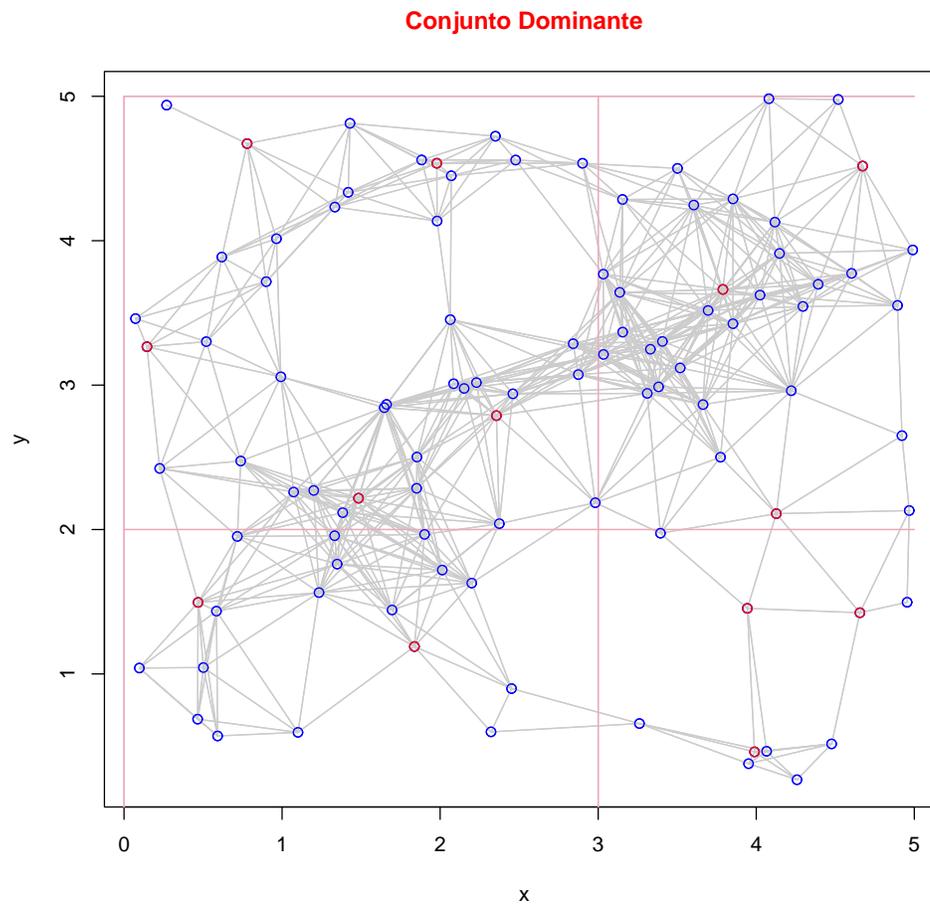


Figura A.37: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión cuadrada y aristas de 3cm.

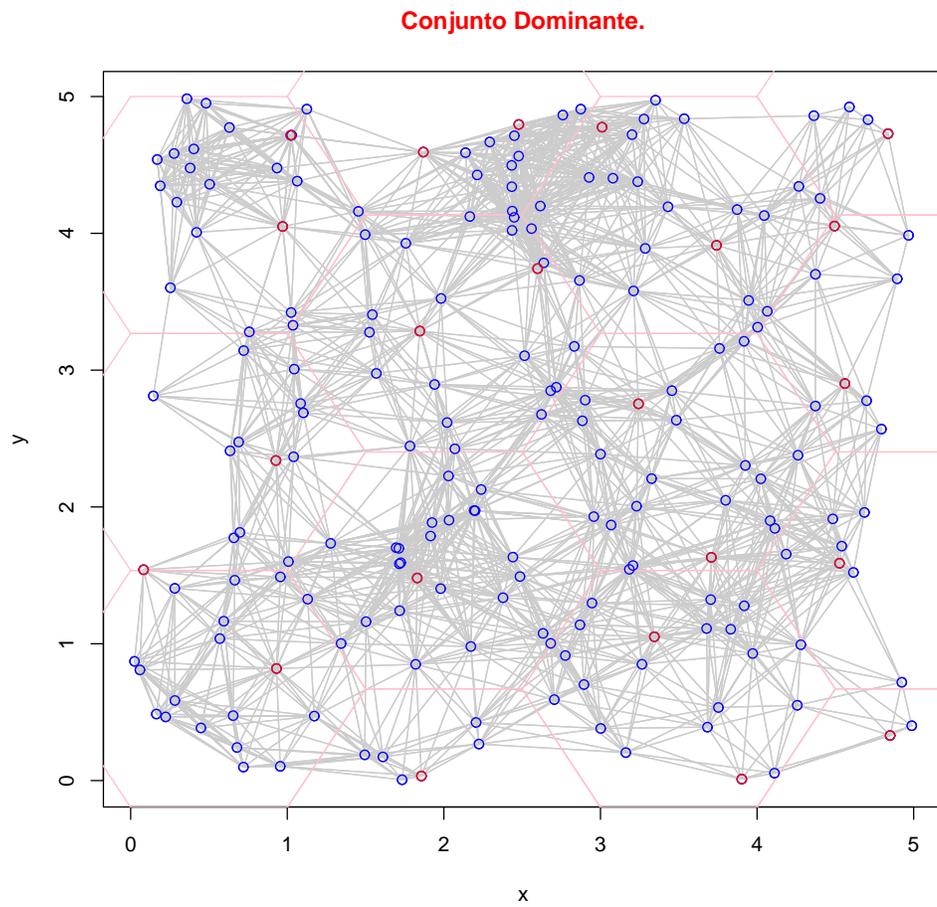


Figura A.38: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión hexagonal y aristas de  $1cm$ .

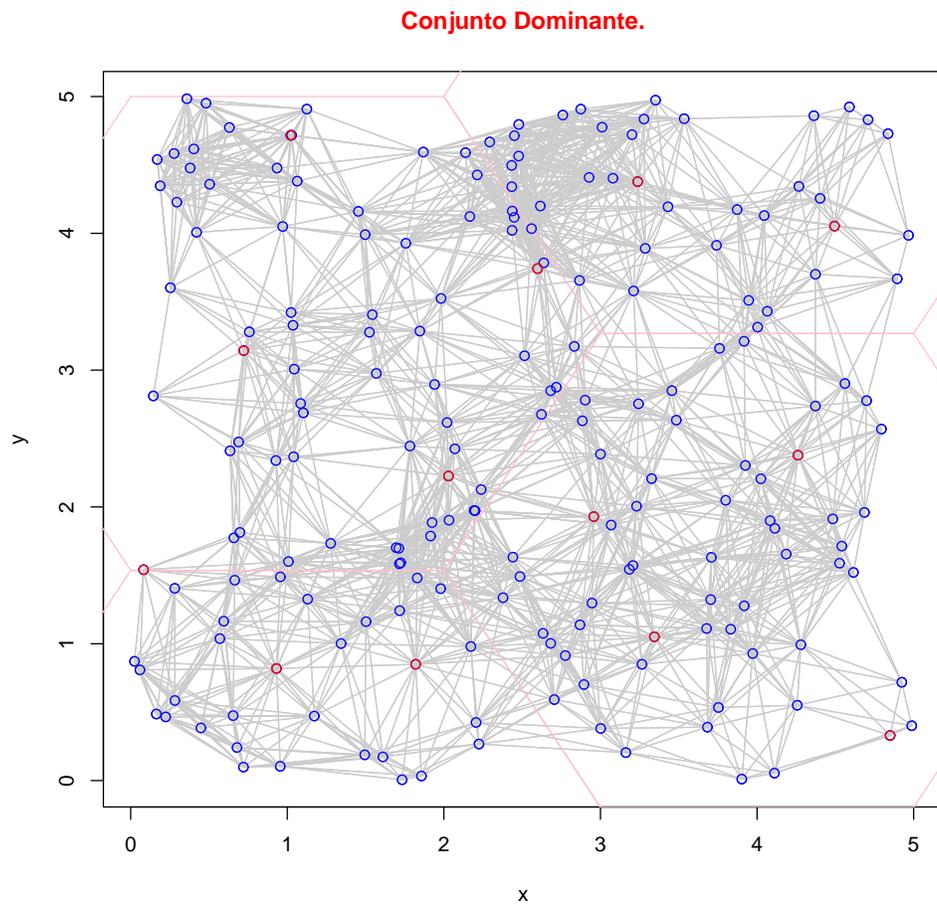


Figura A.39: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión hexagonal y aristas de 2cm.

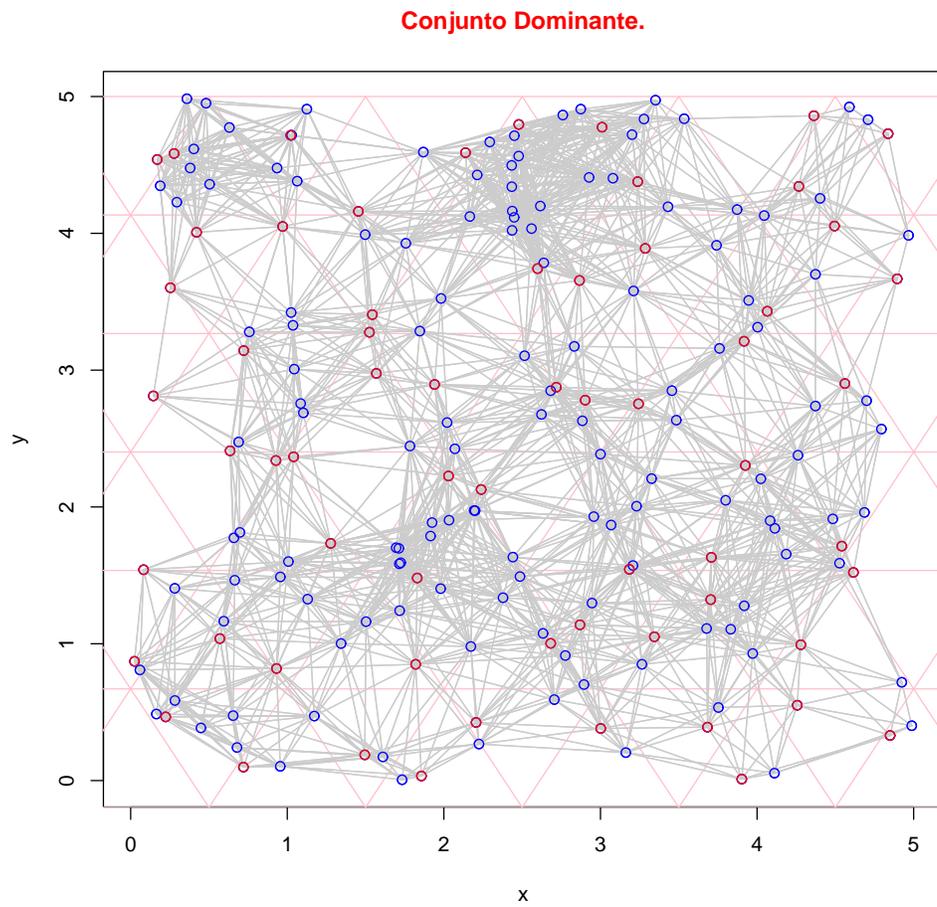


Figura A.40: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión triangular y aristas de  $1cm$ .

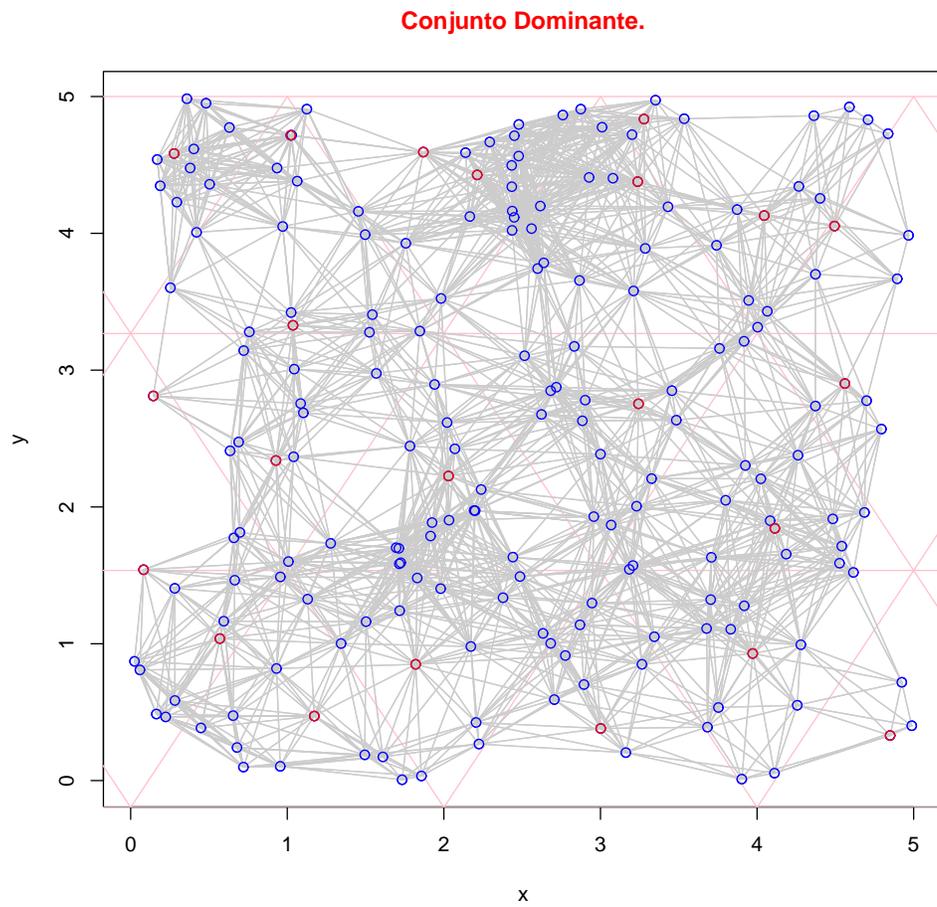


Figura A.41: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión triangular y aristas de  $2cm$ .

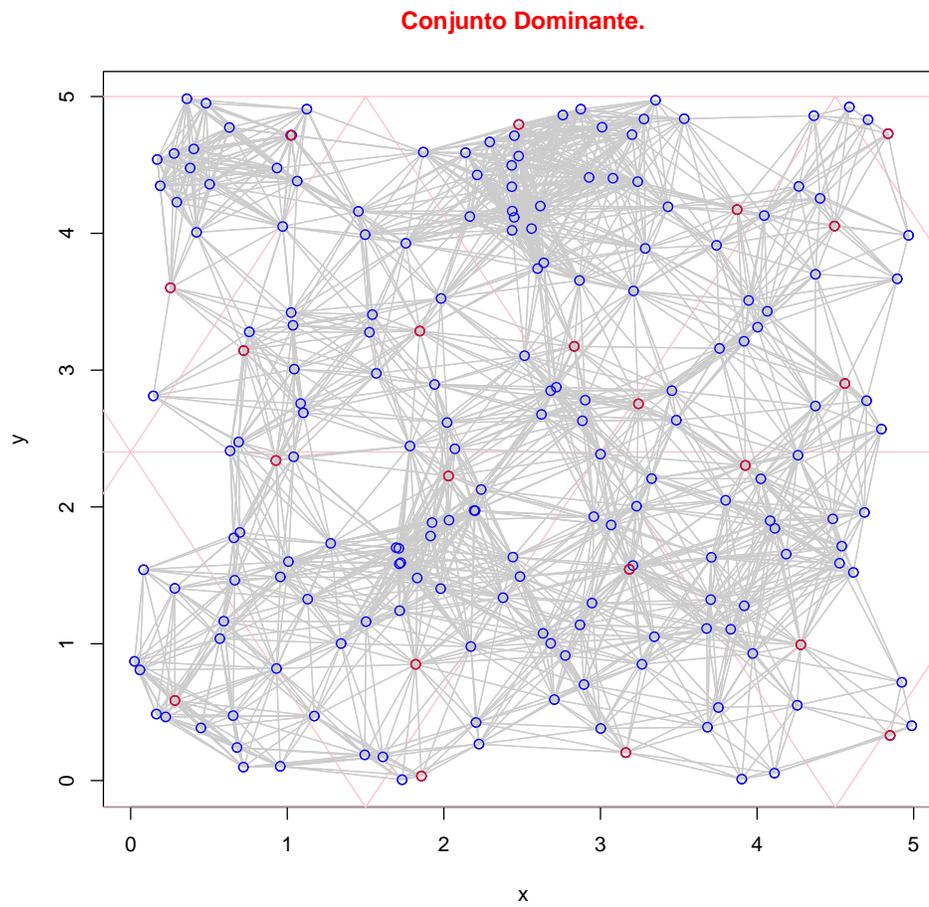


Figura A.42: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión triangular y aristas de 3cm.

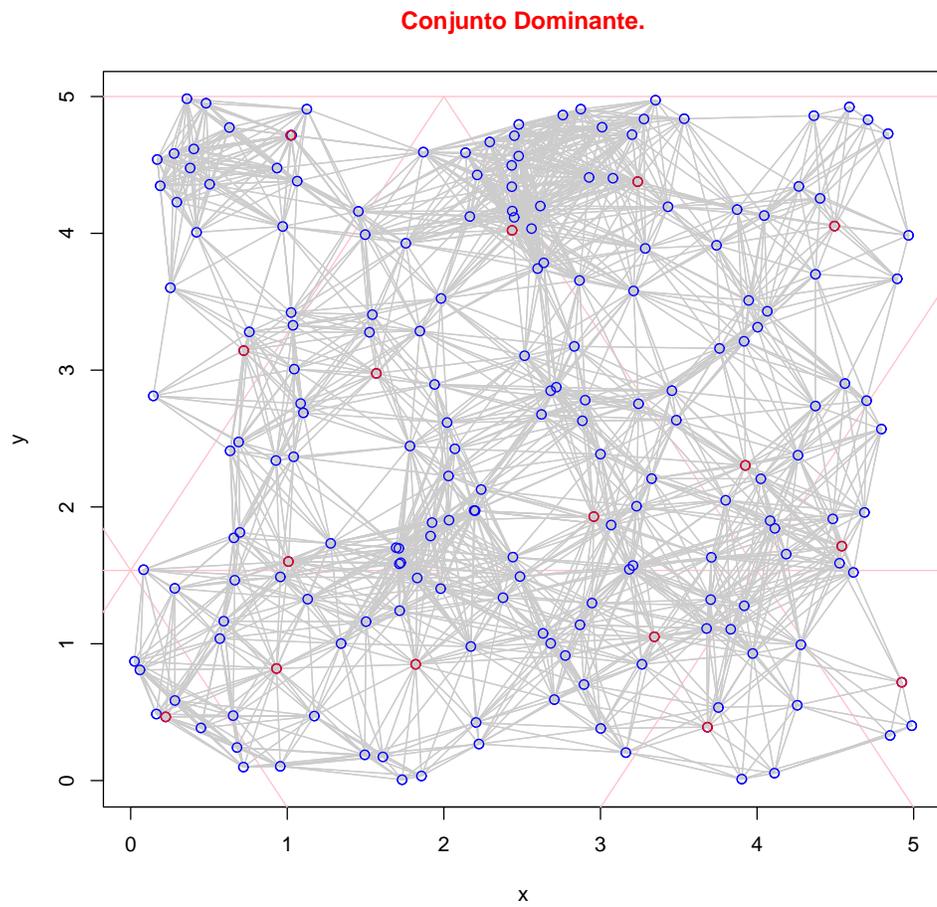


Figura A.43: Resultado de la heurística para una red de tamaño 200 y grid 5, con subdivisión triangular y aristas de 4cm.

## A.2.2. Grid de tamaño seis

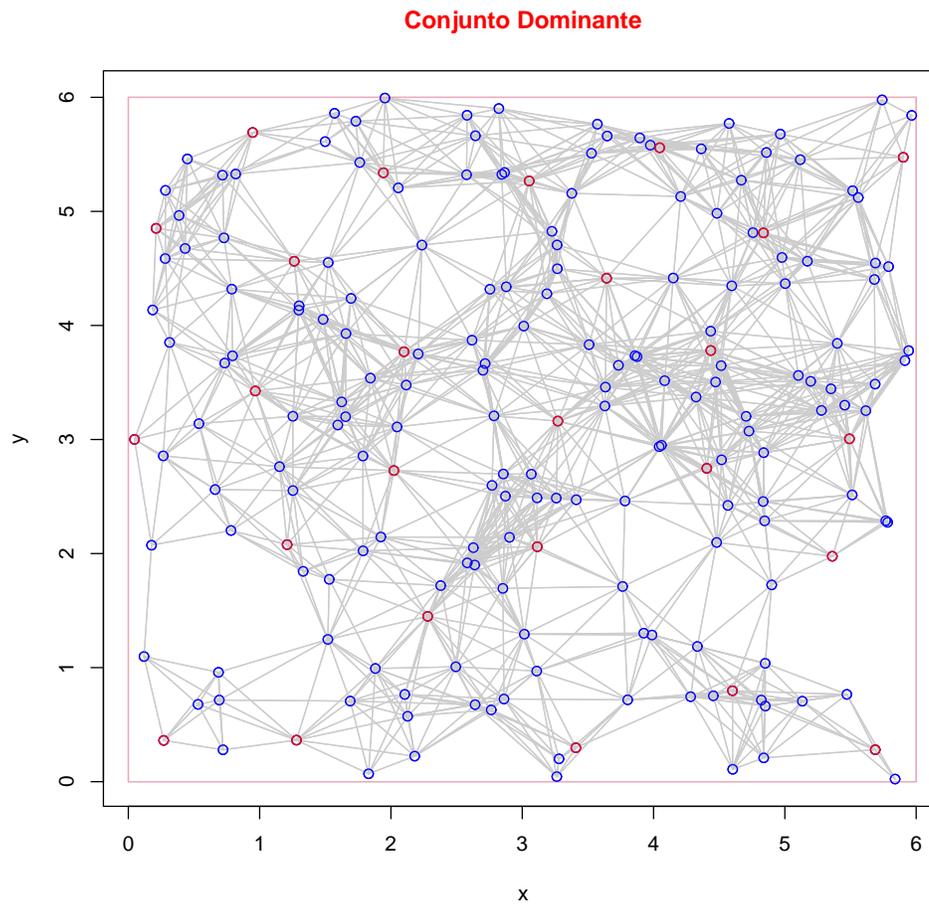


Figura A.44: Resultado del algoritmo glotón para una red de tamaño 200 y grid 6.

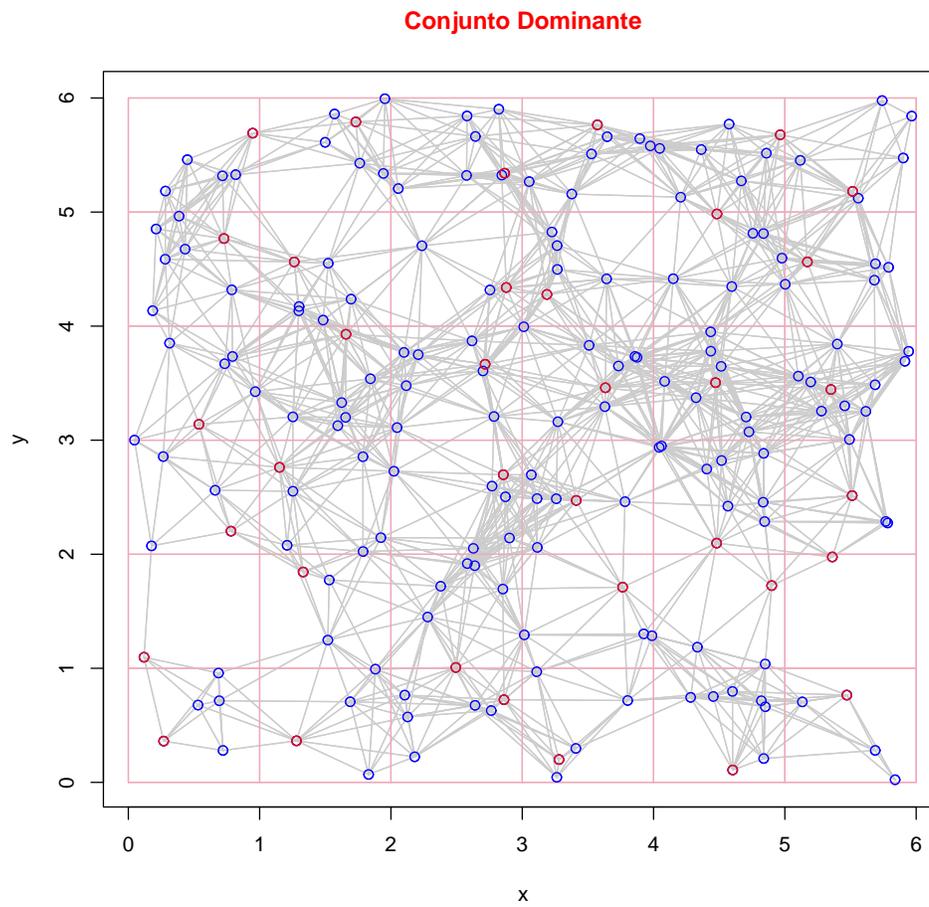


Figura A.45: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión cuadrada y aristas de  $1cm$ .

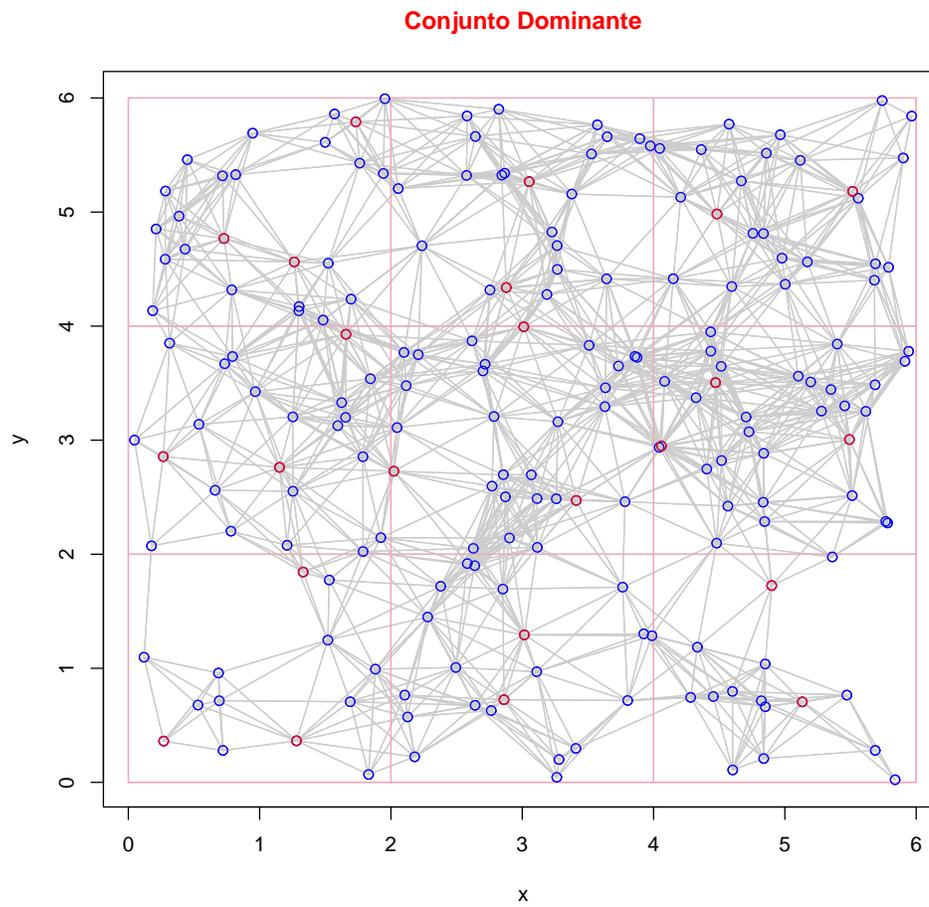


Figura A.46: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión cuadrada y aristas de  $2\text{cm}$ .

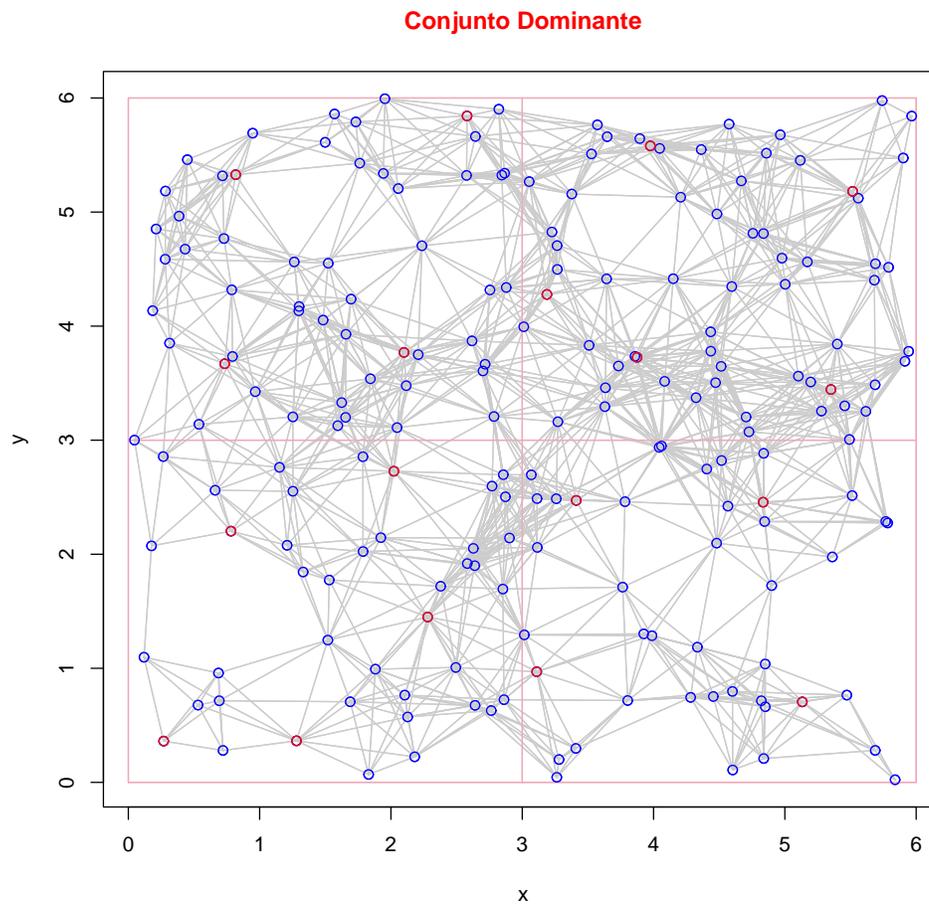


Figura A.47: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión cuadrada y aristas de 3cm.

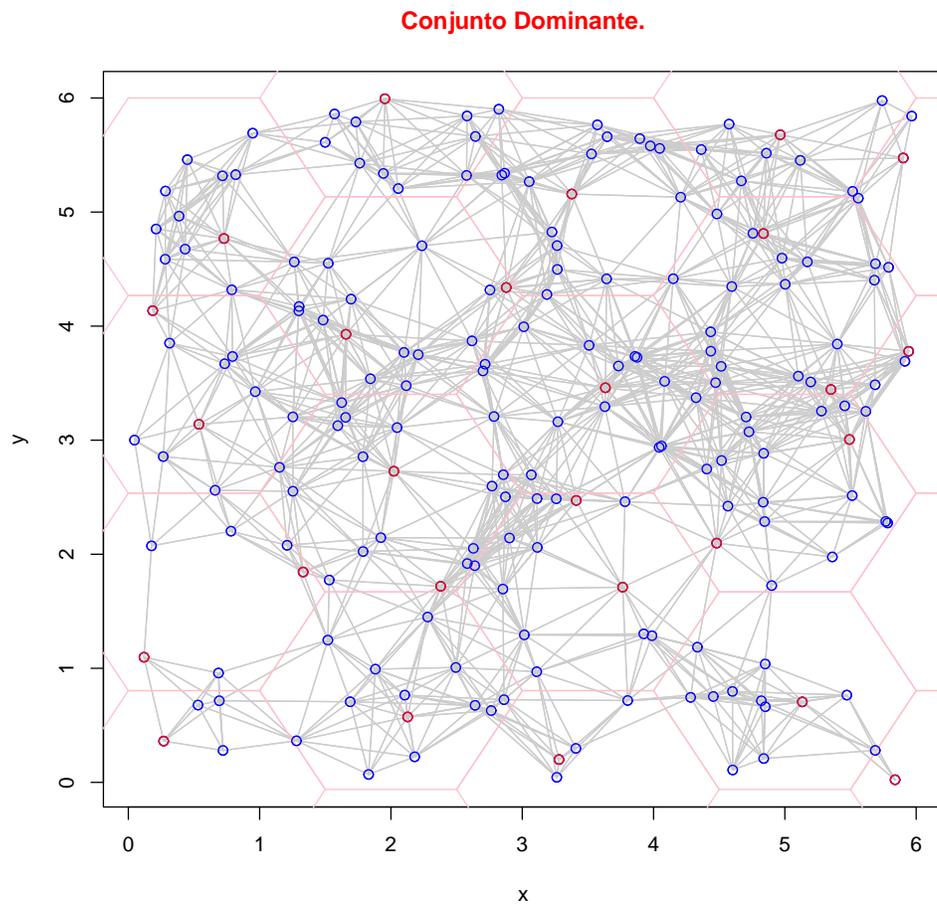


Figura A.48: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión hexagonal y aristas de  $1cm$ .

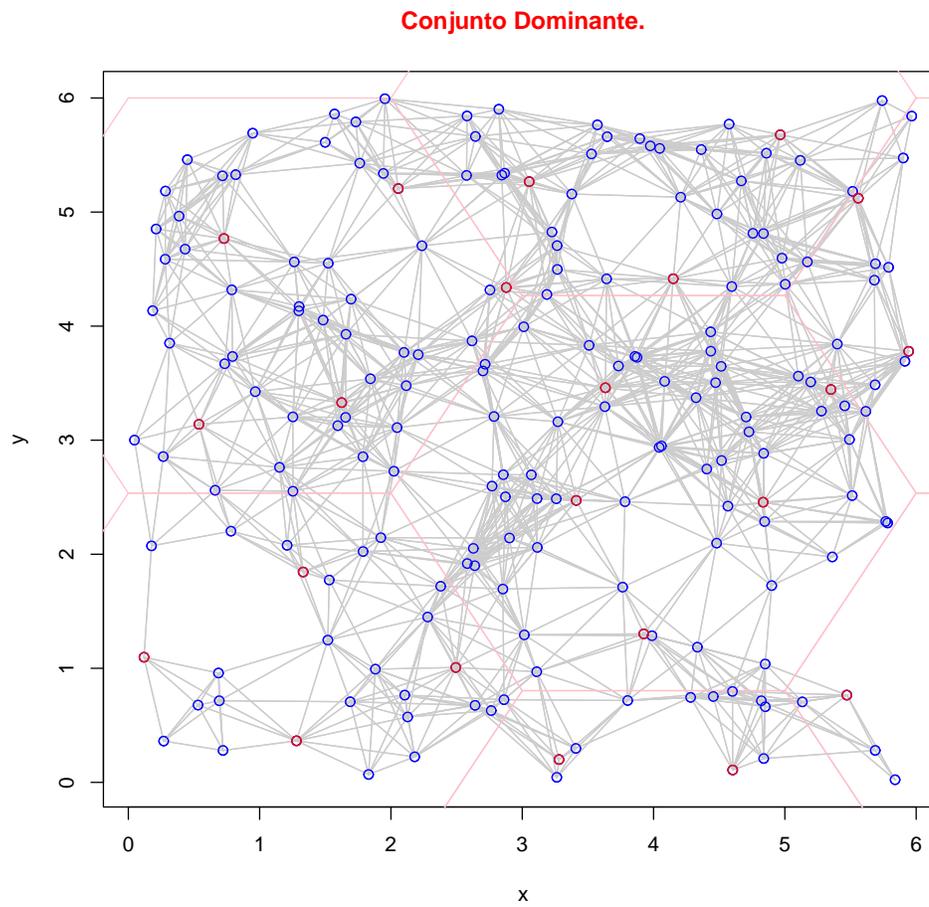


Figura A.49: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión hexagonal y aristas de  $2cm$ .

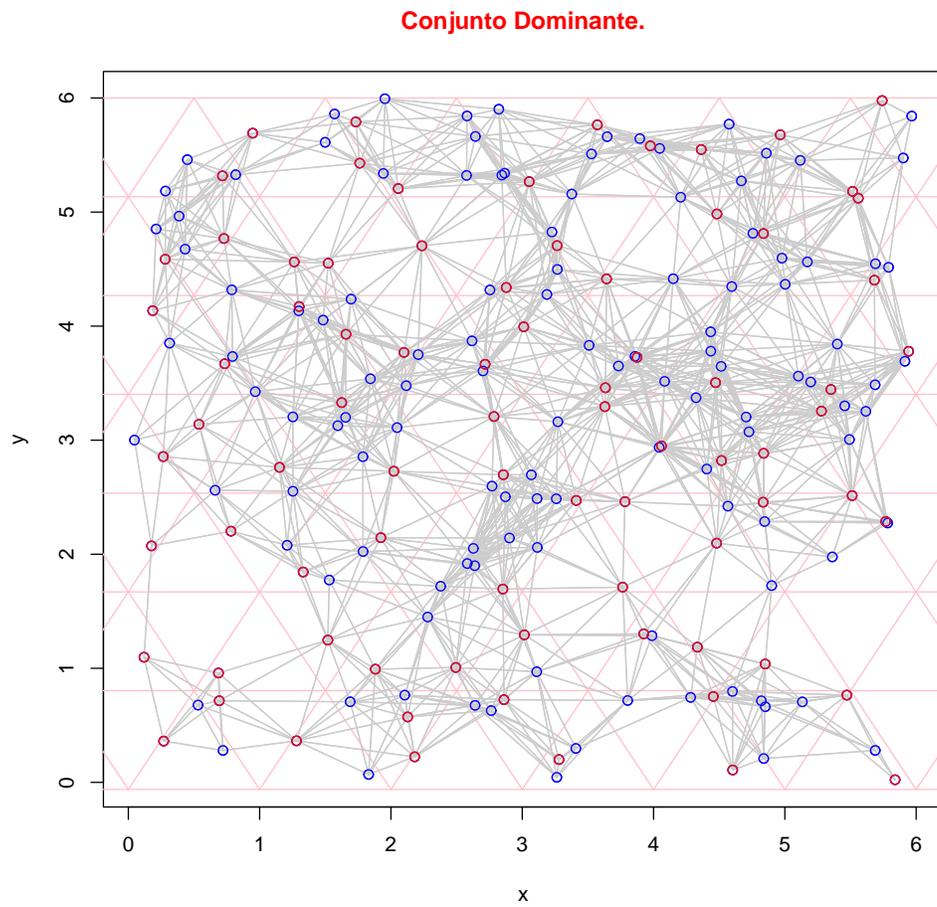


Figura A.50: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión triangular y aristas de  $1cm$ .

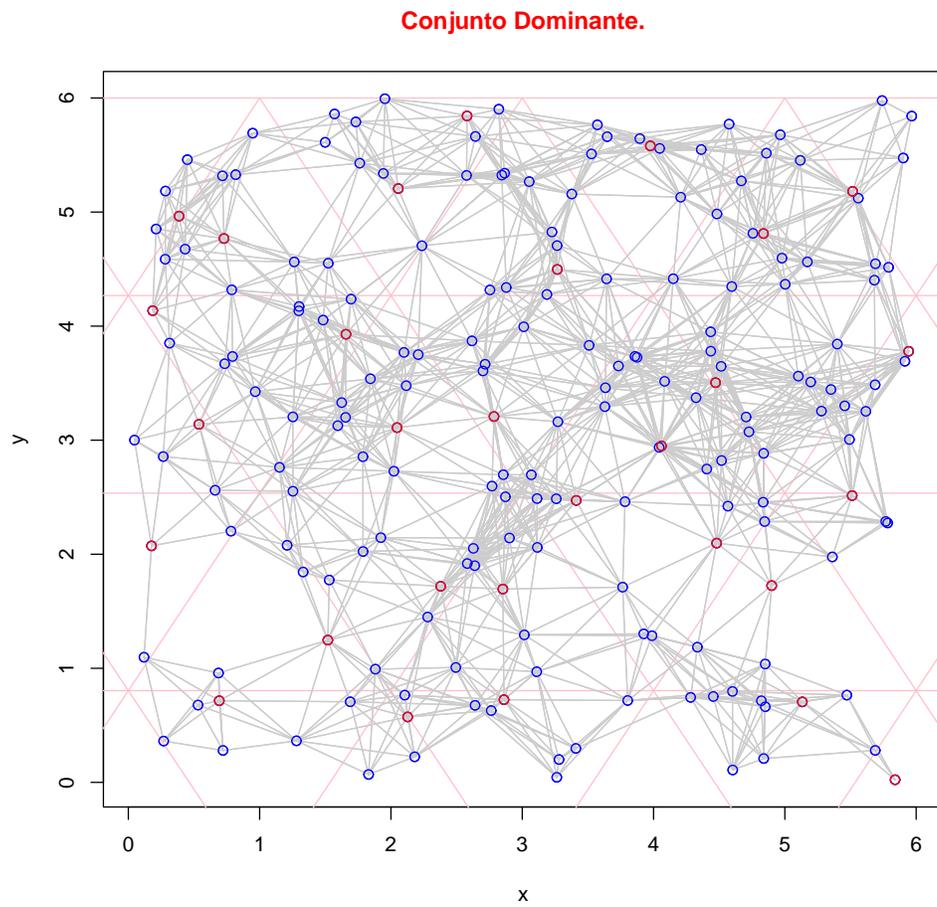


Figura A.51: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión triangular y aristas de 2cm.

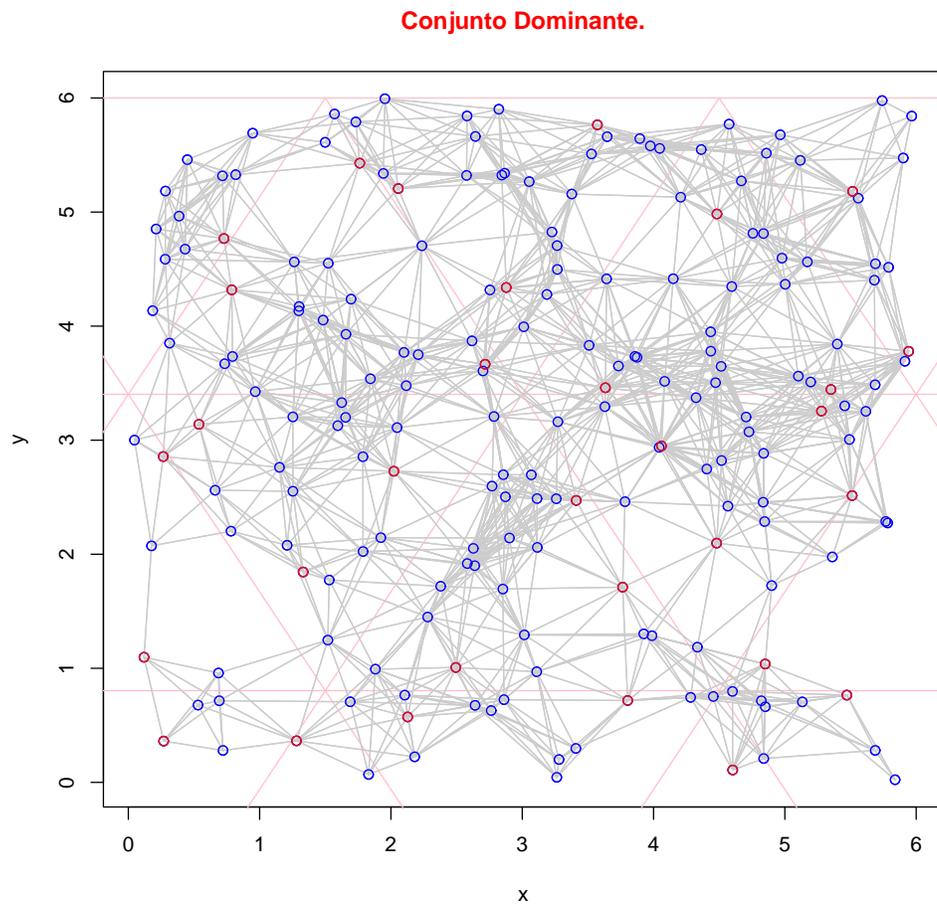


Figura A.52: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión triangular y aristas de  $3cm$ .

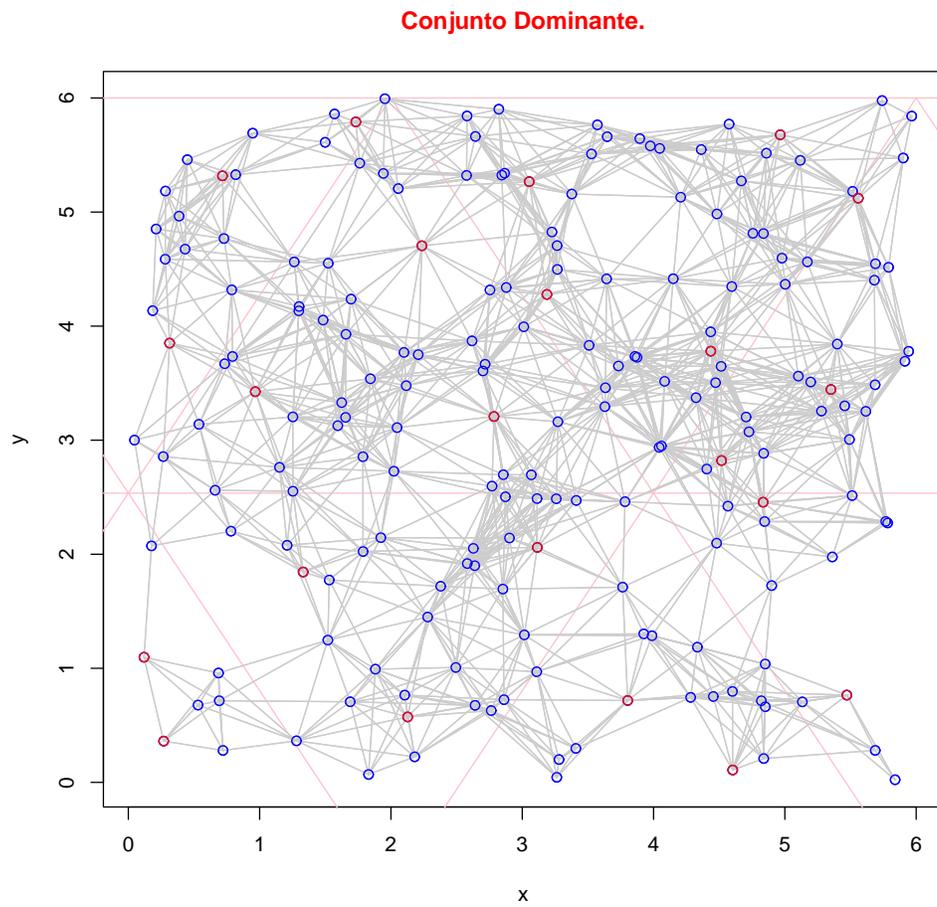


Figura A.53: Resultado de la heurística para una red de tamaño 200 y grid 6, con subdivisión triangular y aristas de  $4cm$ .

## A.2.3. Grid de tamaño siete

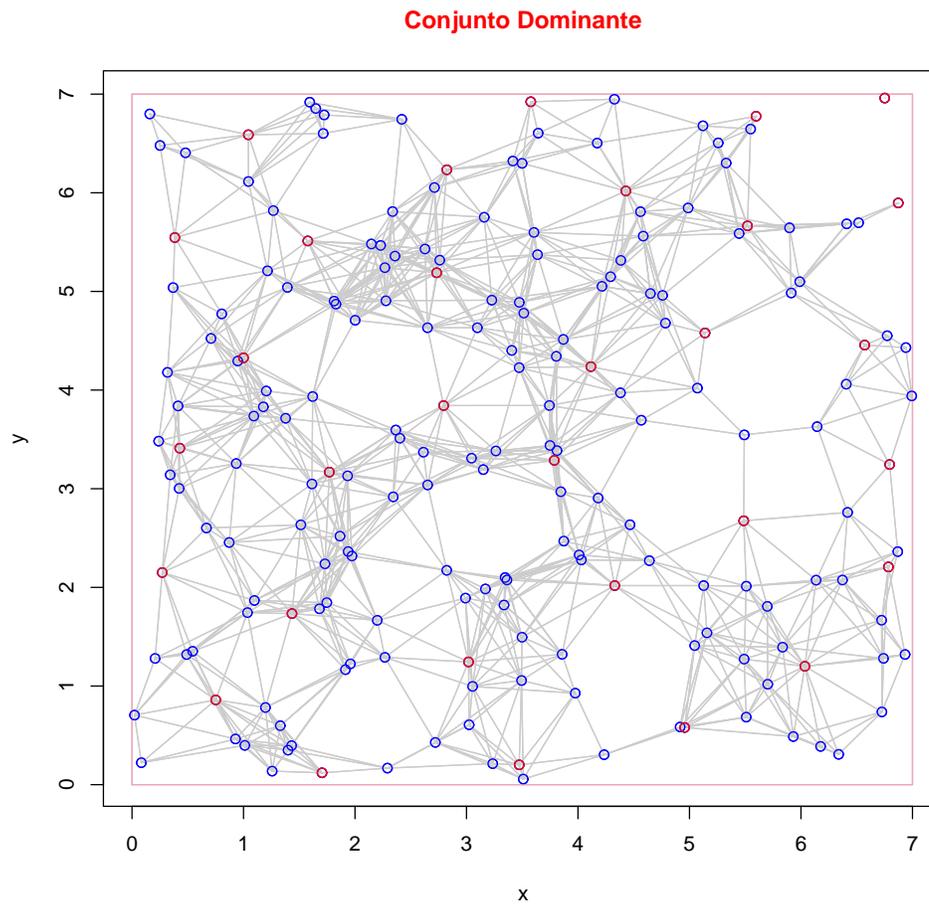


Figura A.54: Resultado del algoritmo glotón para una red de tamaño 200 y grid 7.

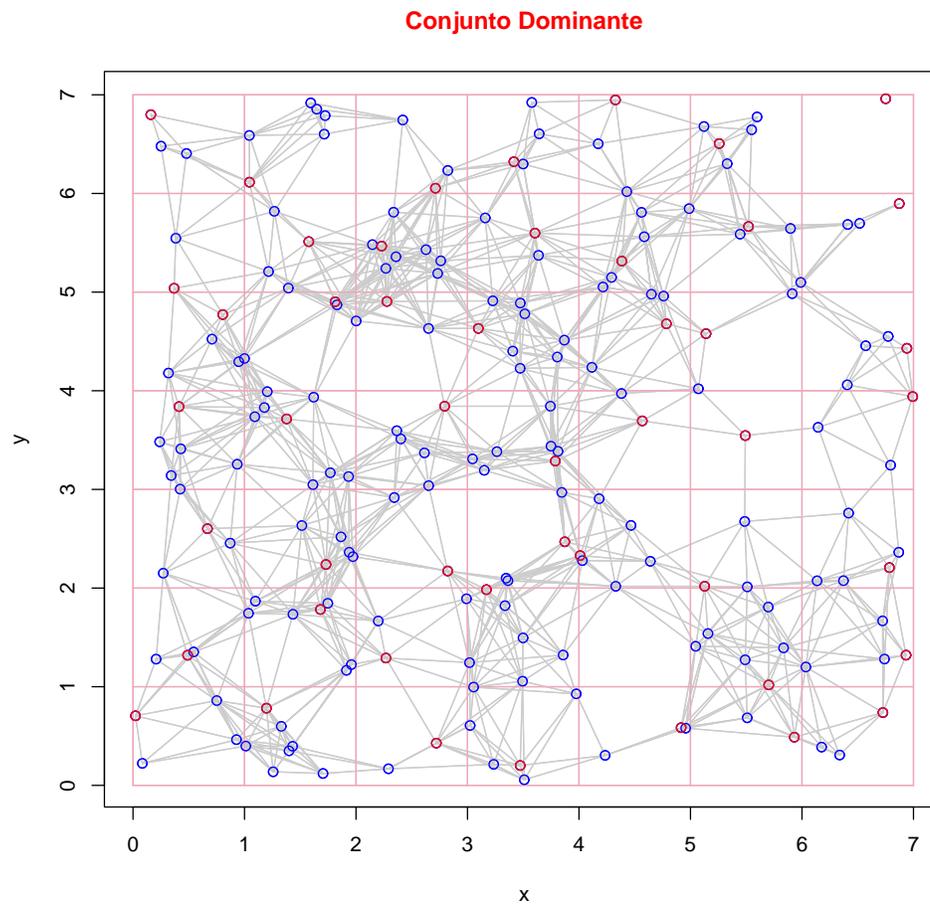


Figura A.55: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión cuadrada y aristas de  $1cm$ .

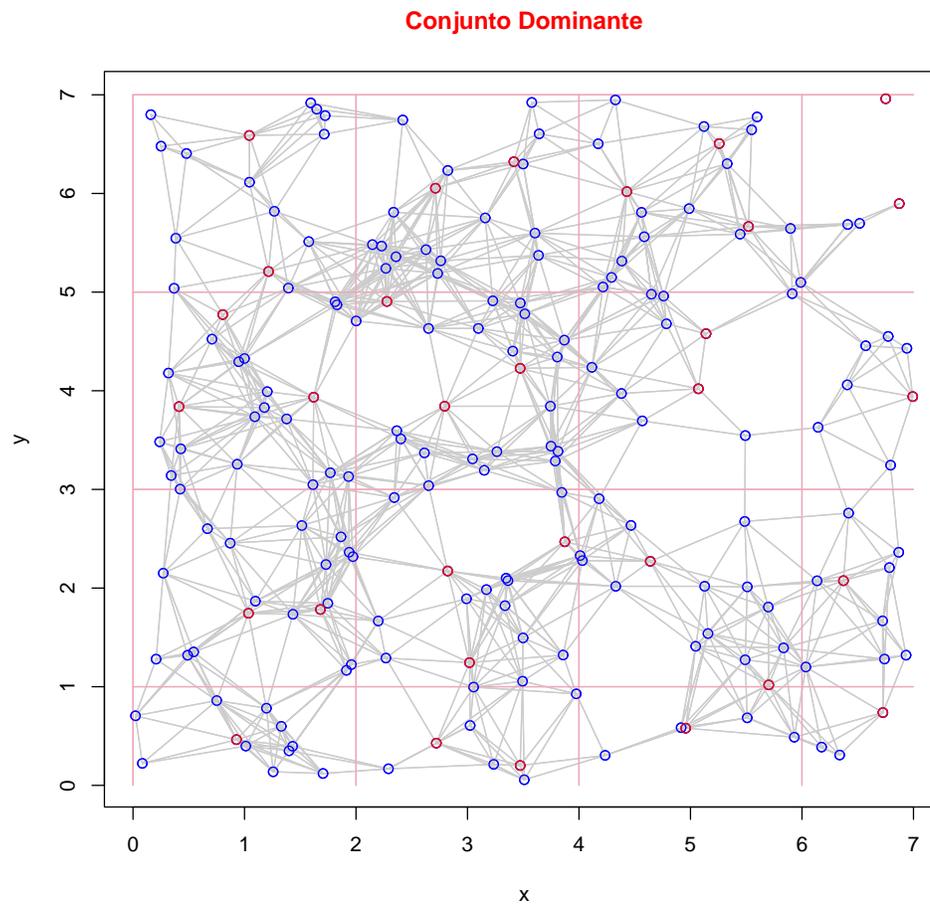


Figura A.56: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión cuadrada y aristas de  $2\text{cm}$ .

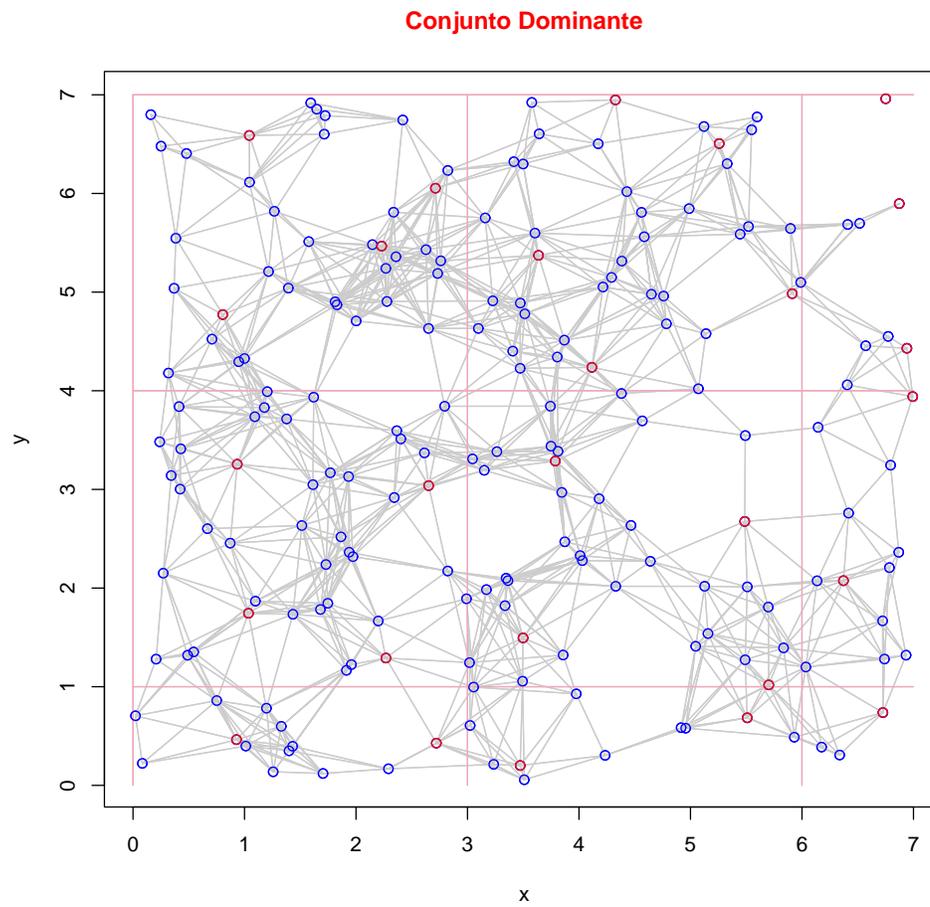


Figura A.57: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión cuadrada y aristas de 3cm.

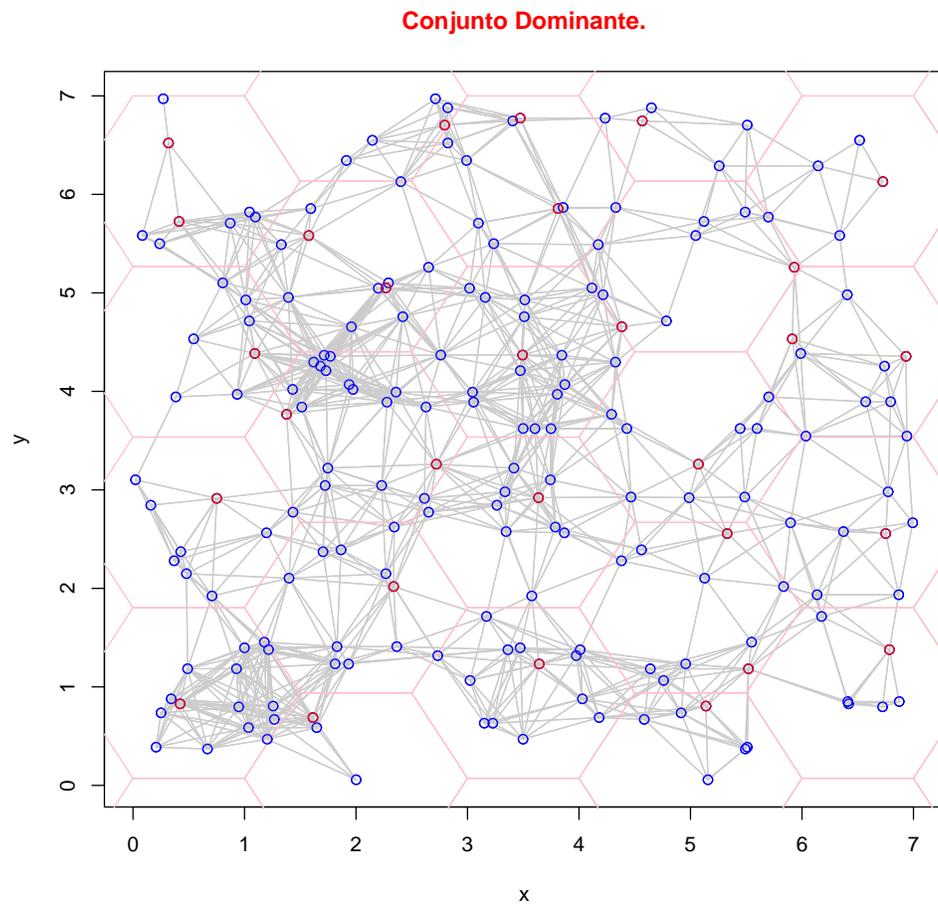


Figura A.58: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión hexagonal y aristas de 1cm.

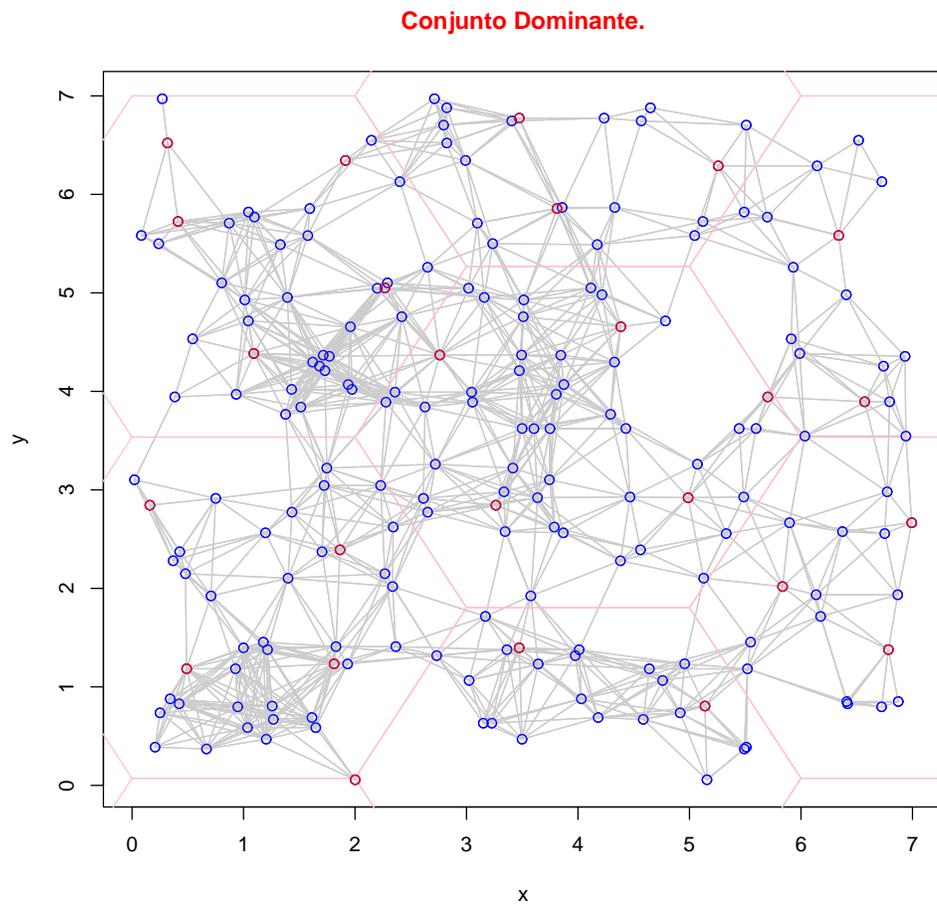


Figura A.59: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión hexagonal y aristas de 2cm.

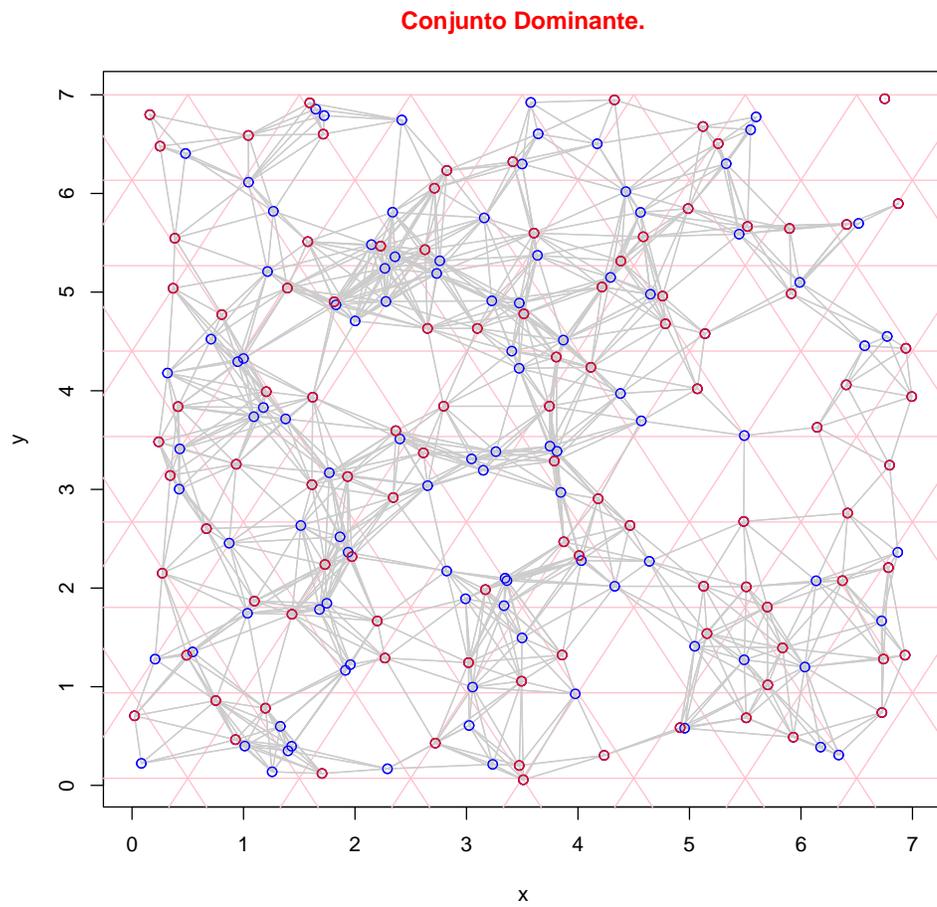


Figura A.60: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión triangular y aristas de  $1cm$ .

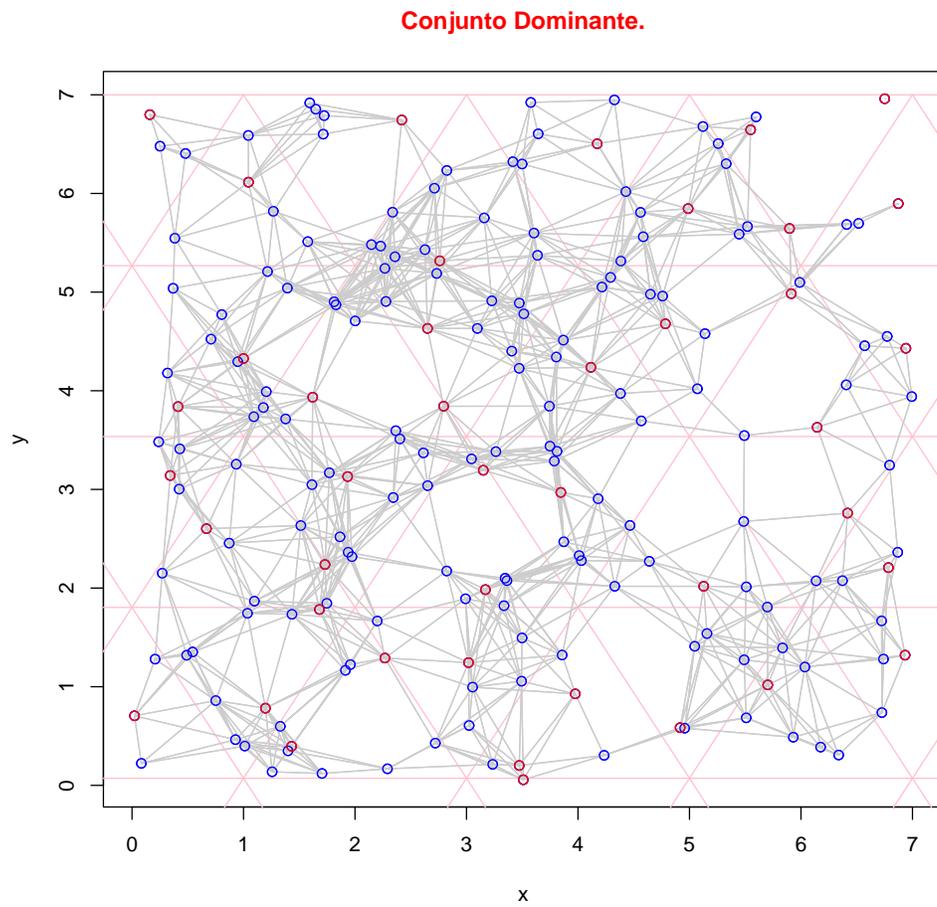


Figura A.61: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión triangular y aristas de  $2cm$ .

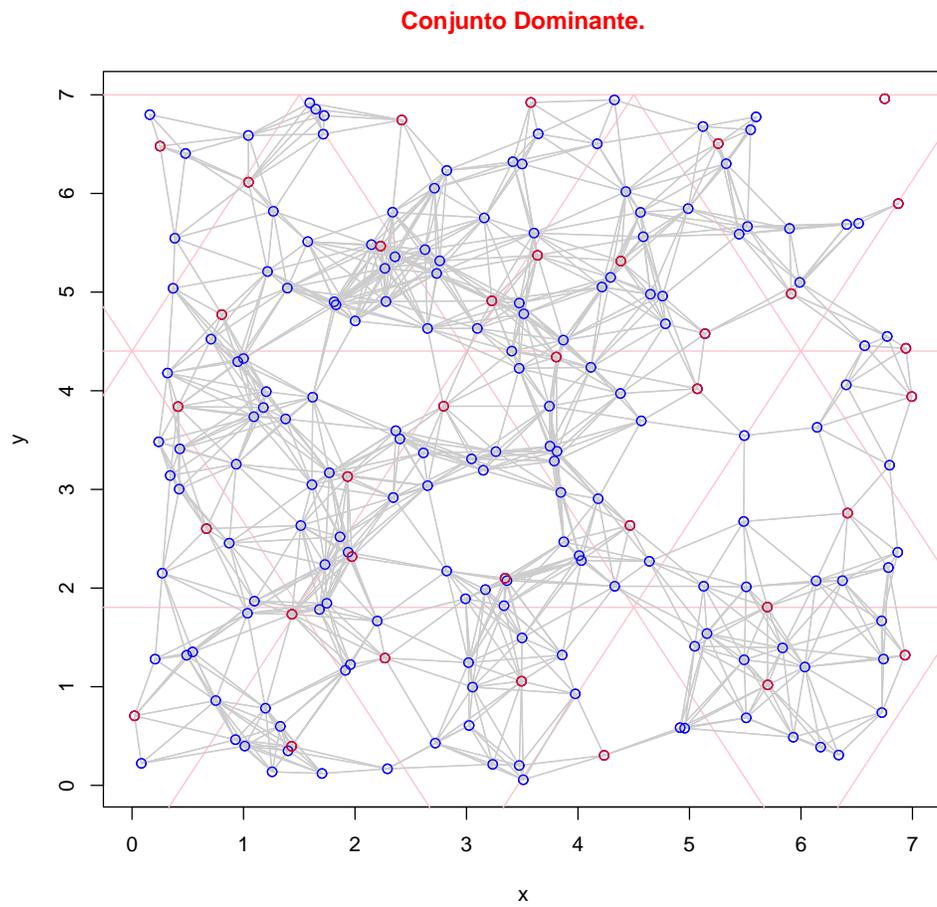


Figura A.62: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión triangular y aristas de 3cm.

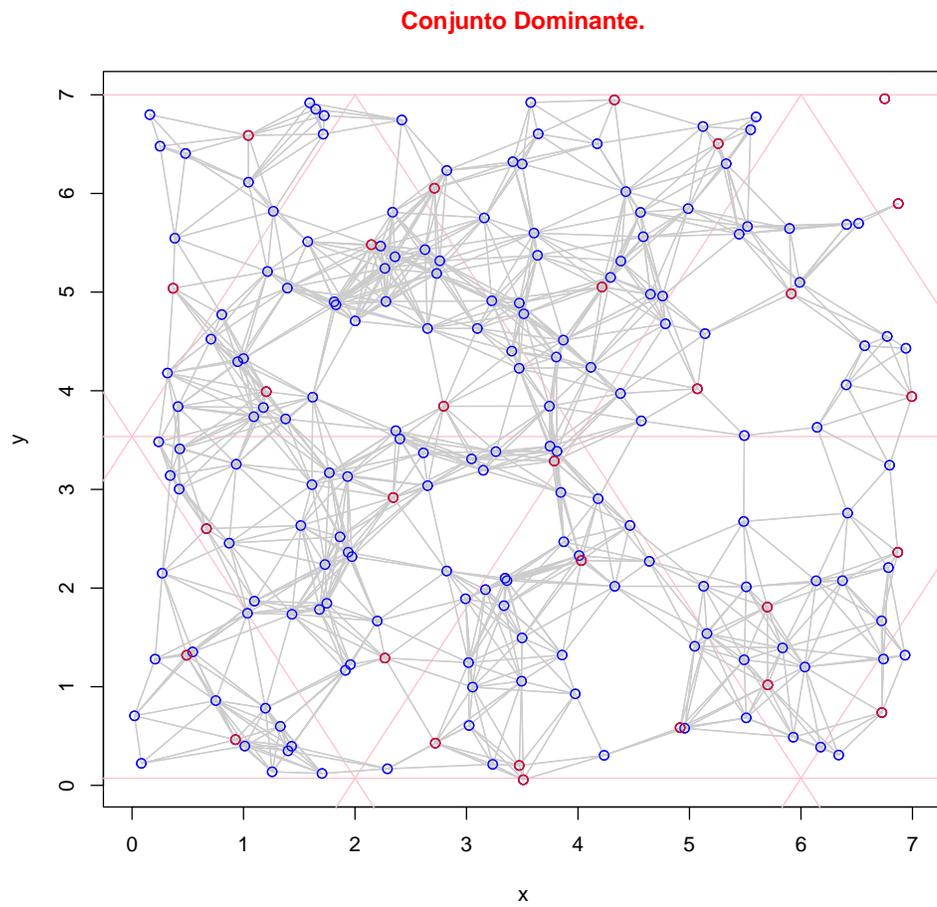


Figura A.63: Resultado de la heurística para una red de tamaño 200 y grid 7, con subdivisión triangular y aristas de  $4cm$ .

### A.3. Redes de tamaño quinientos

#### A.3.1. Grid de tamaño diez

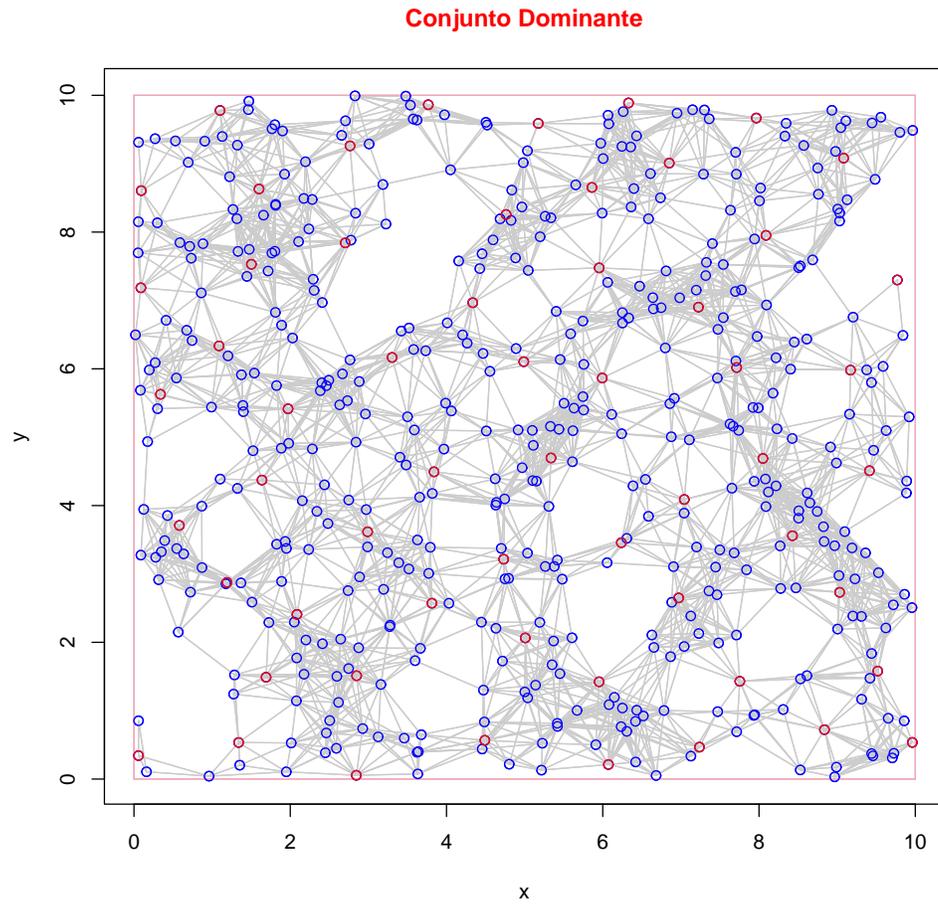


Figura A.64: Resultado del algoritmo glotón para una red de tamaño 500 y grid 10.

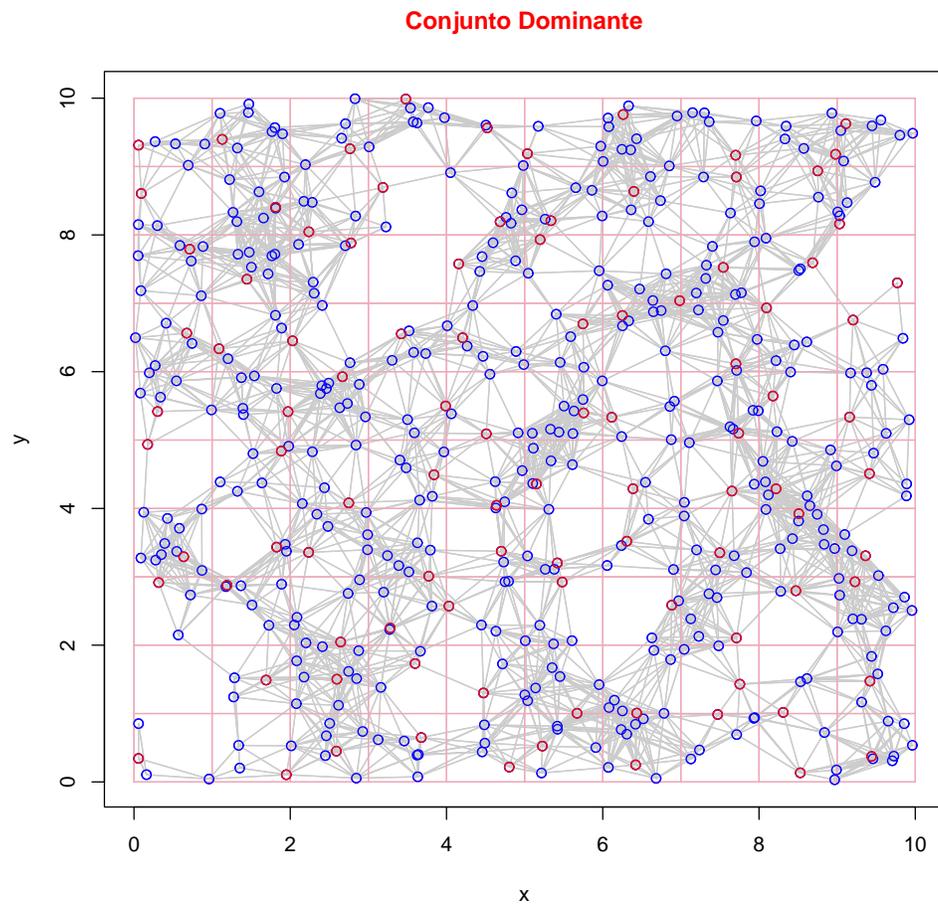


Figura A.65: Resultado de la heurística para una red de tamaño 500 y grid 10, con subdivisión cuadrada y aristas de  $1cm$ .

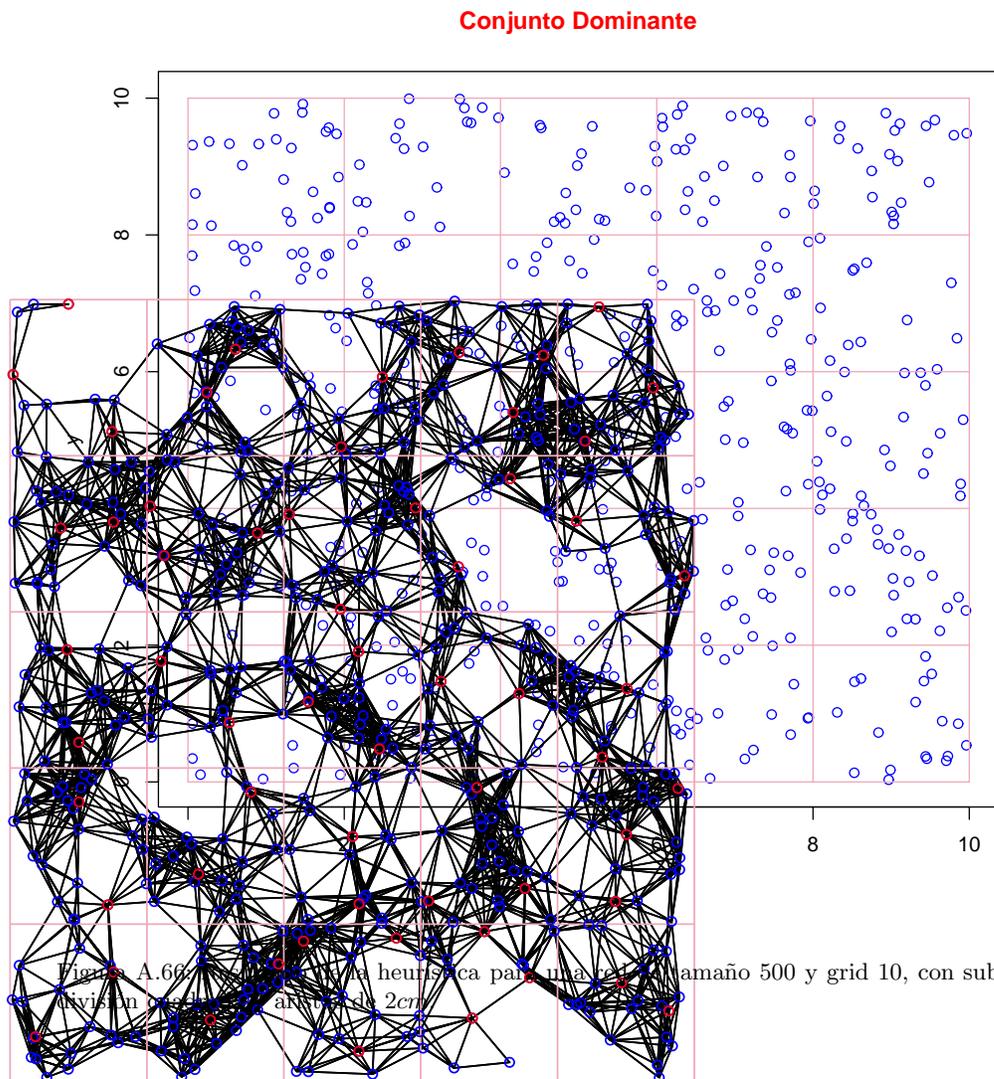


Figura A.66: Red generada por una heurística para un tamaño 500 y grid 10, con sub-  
 división en  $2 \times 2$  arcos de  $2cm$ .

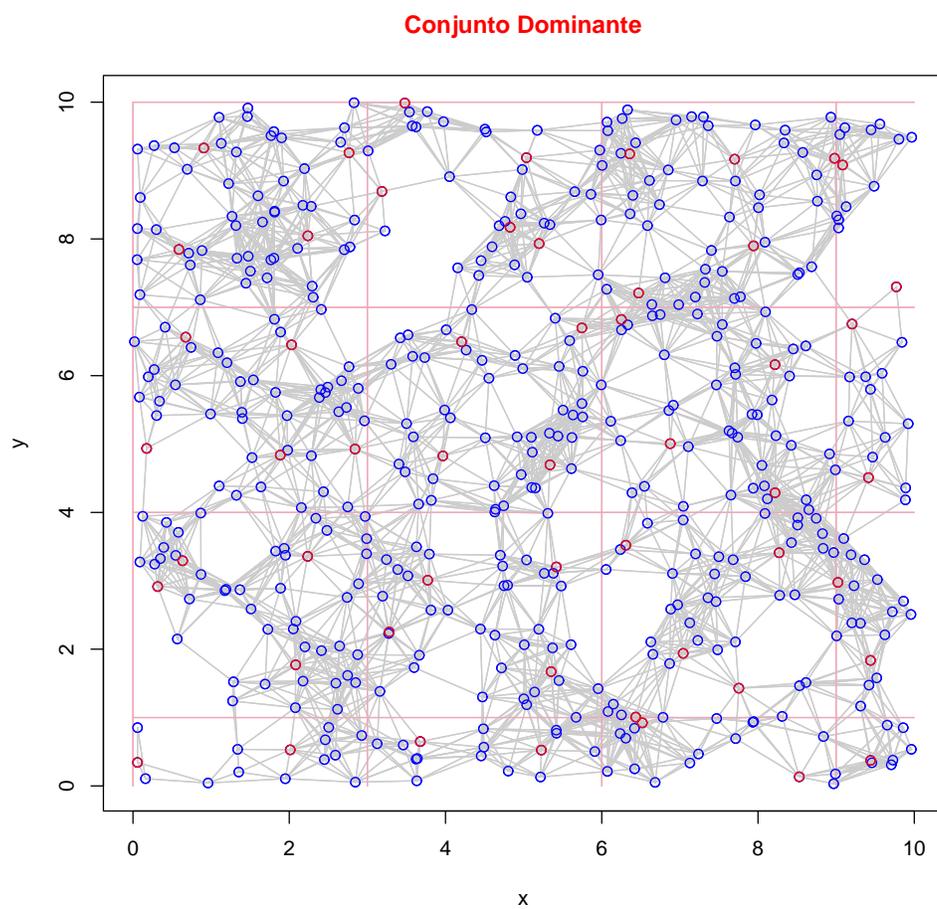


Figura A.67: Resultado de la heurística para una red de tamaño 500 y grid 10, con subdivisión cuadrada y aristas de 3cm.

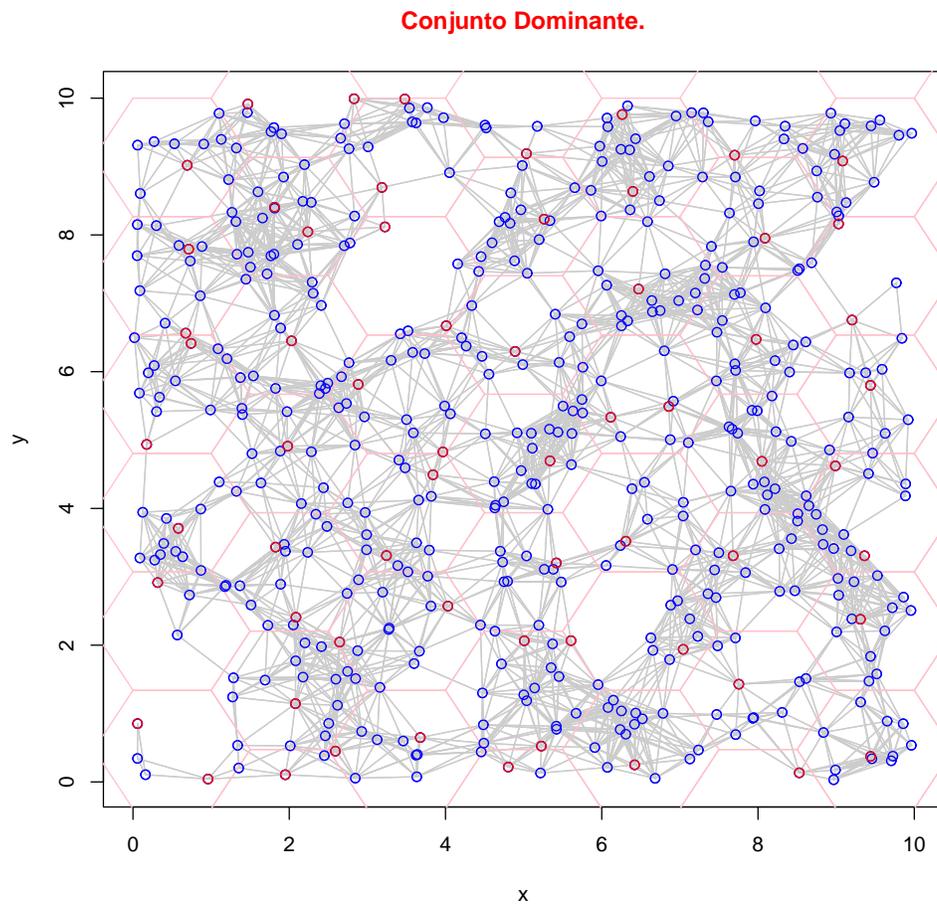


Figura A.68: Resultado de la heurística para una red de tamaño 500 y grid 10, con subdivisión hexagonal y aristas de  $1cm$ .

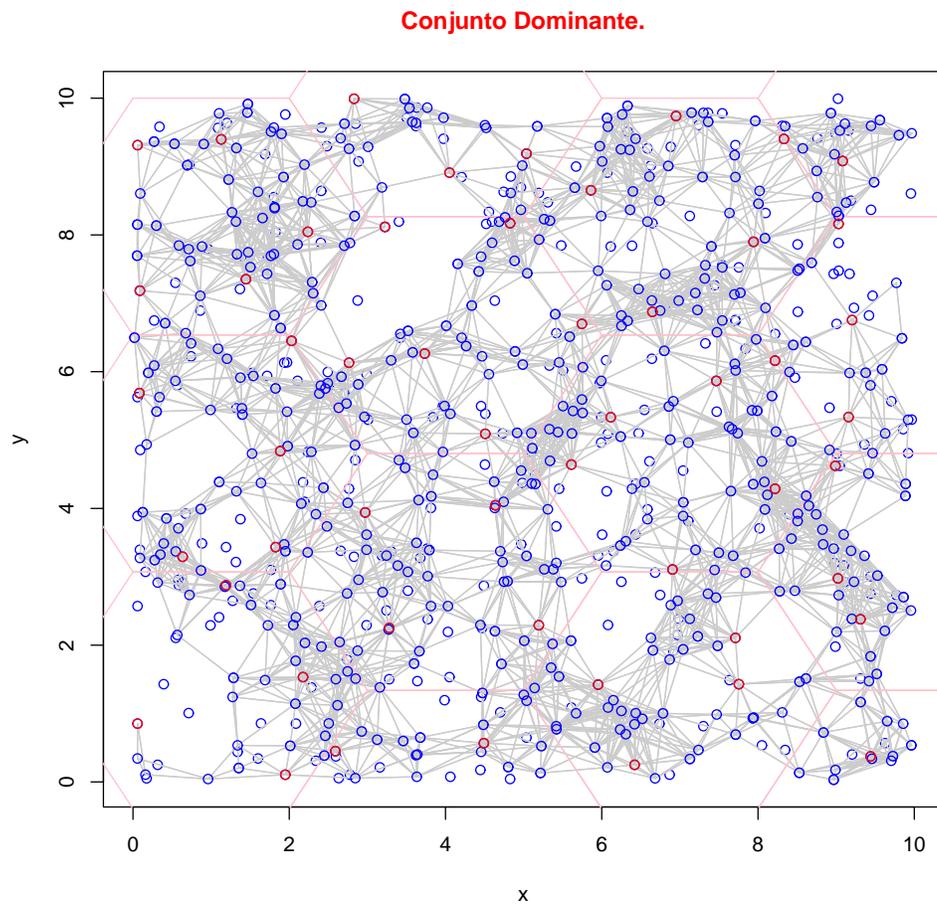


Figura A.69: Resultado de la heurística para una red de tamaño 500 y grid 10, con subdivisión hexagonal y aristas de 2cm.

## A.3.2. Grid de tamaño quince

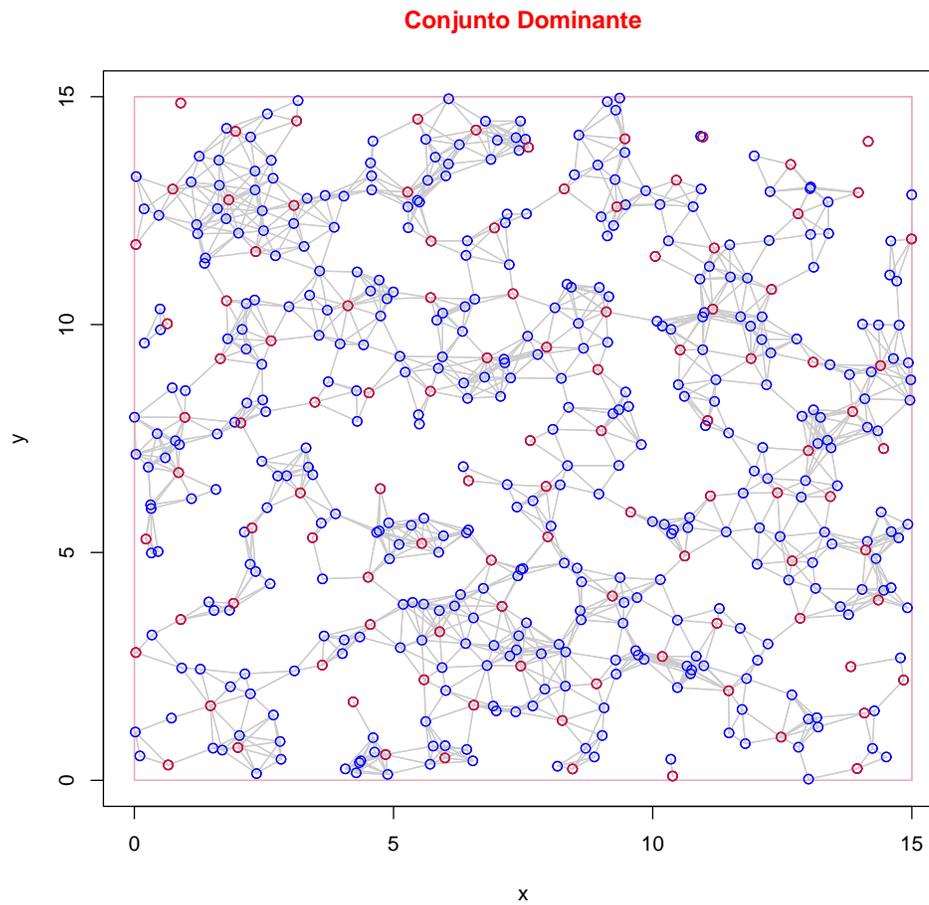


Figura A.70: Resultado del algoritmo glotón para una red de tamaño 500 y grid 15.

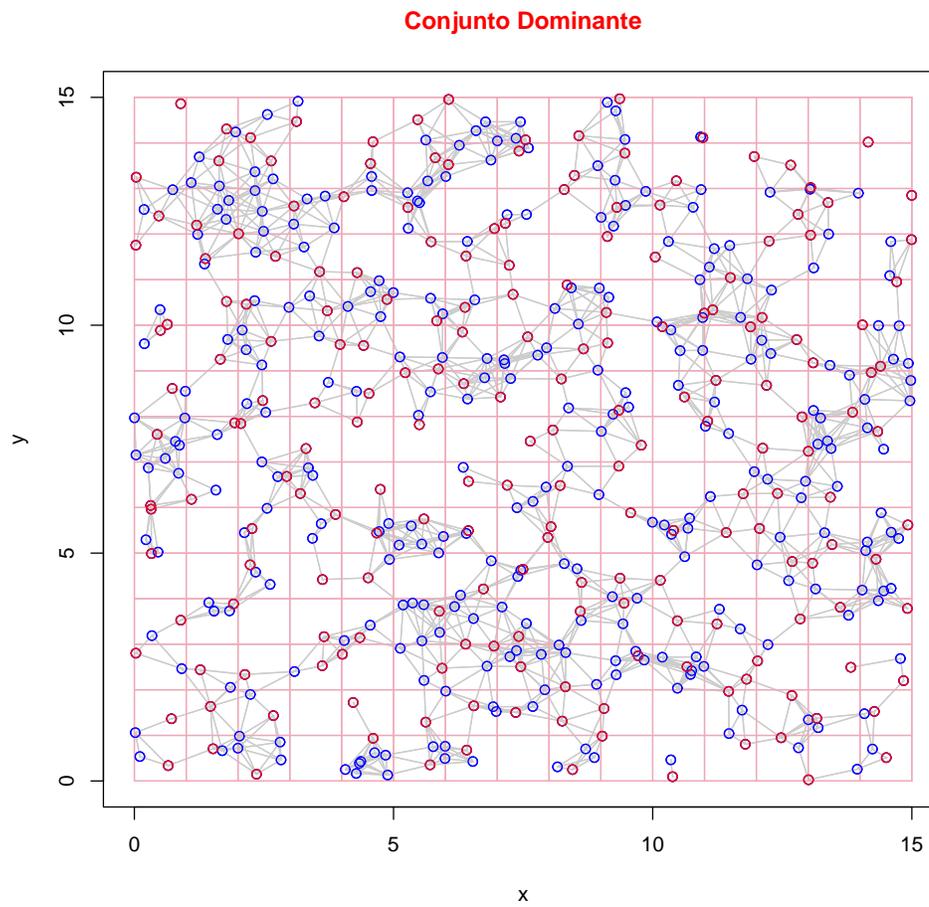


Figura A.71: Resultado de la heurística para una red de tamaño 500 y grid 15, con subdivisión cuadrada y aristas de 1cm.

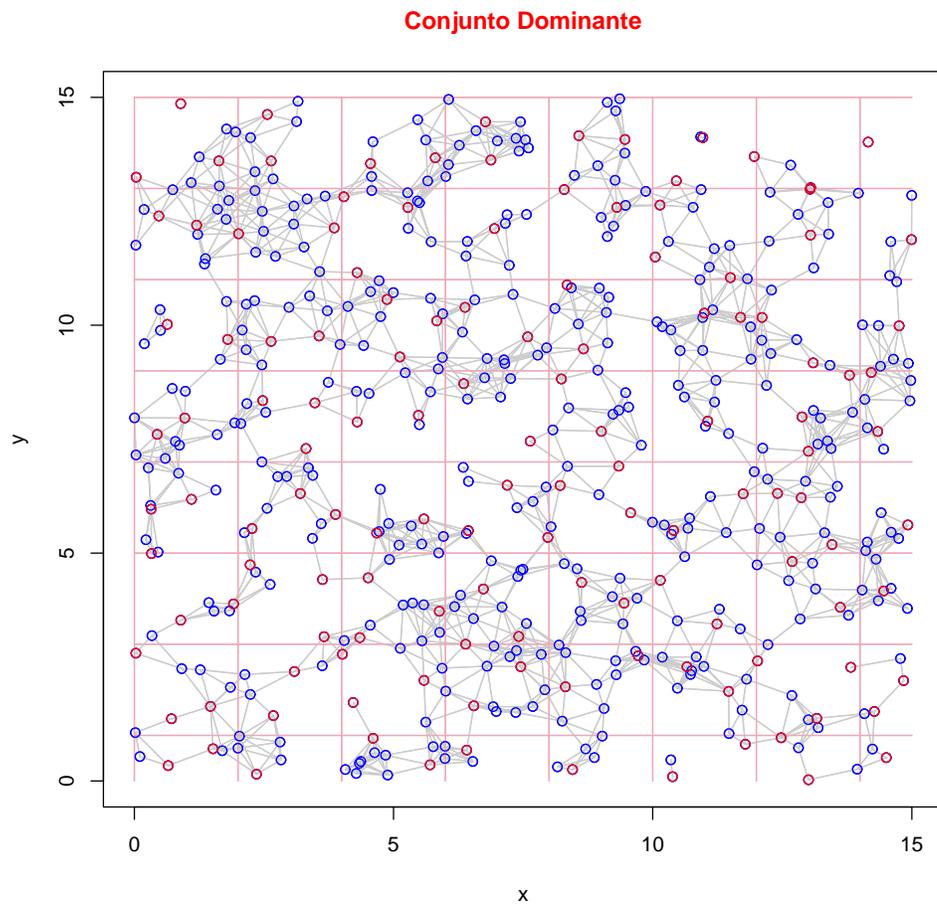


Figura A.72: Resultado de la heurística para una red de tamaño 500 y grid 15, con subdivisión cuadrada y aristas de 2cm.

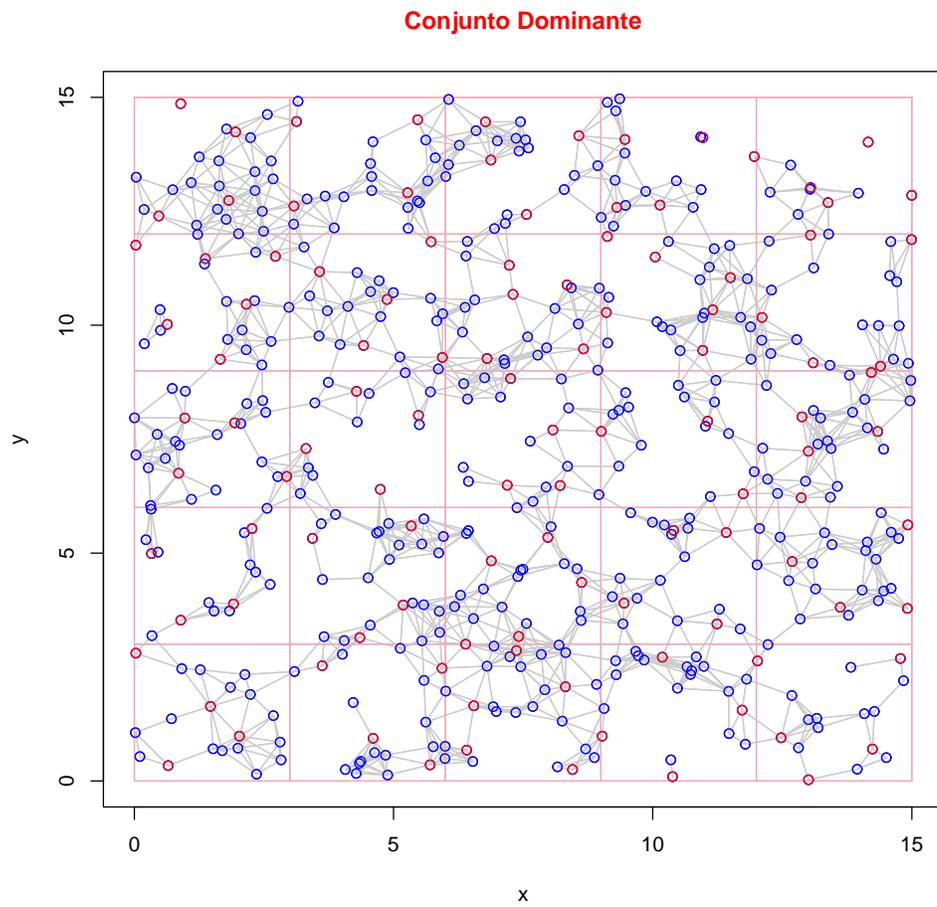


Figura A.73: Resultado de la heurística para una red de tamaño 500 y grid 15, con subdivisión cuadrada y aristas de 3cm.

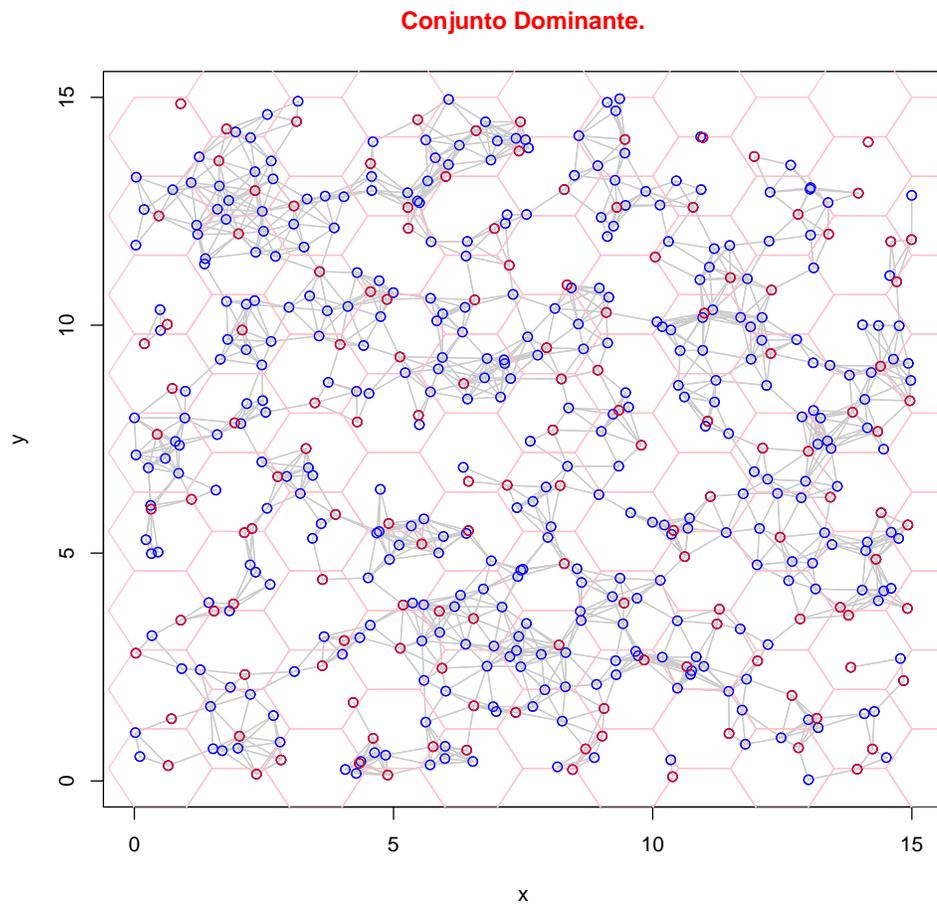


Figura A.74: Resultado de la heurística para una red de tamaño 500 y grid 15, con subdivisión hexagonal y aristas de 1cm.

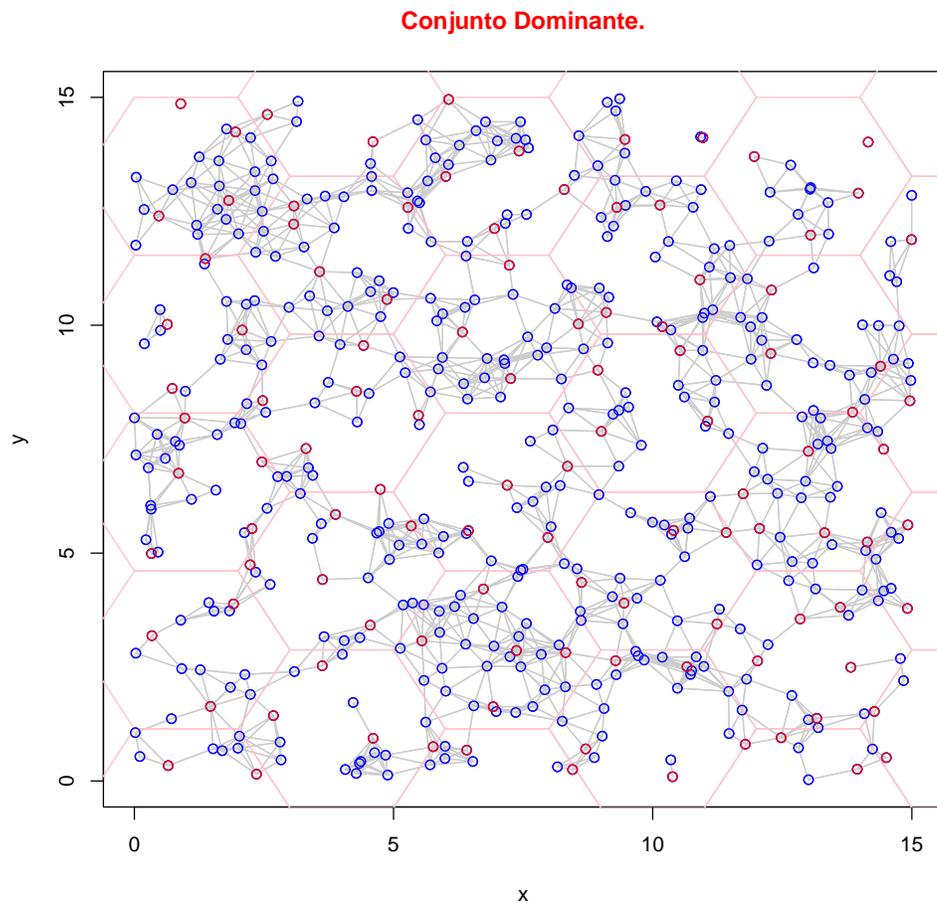


Figura A.75: Resultado de la heurística para una red de tamaño 500 y grid 15, con subdivisión hexagonal y aristas de 2cm.

## A.3.3. Grid de tamaño veinte

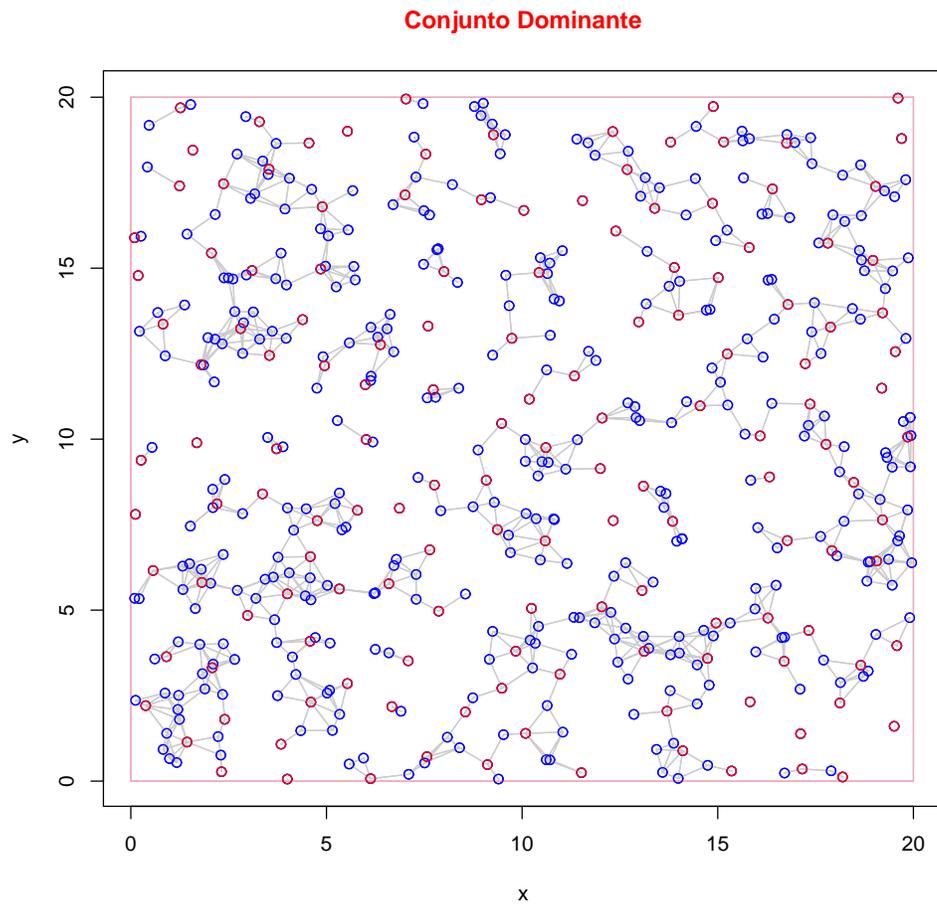


Figura A.76: Resultado del algoritmo glotón para una red de tamaño 500 y grid 20.

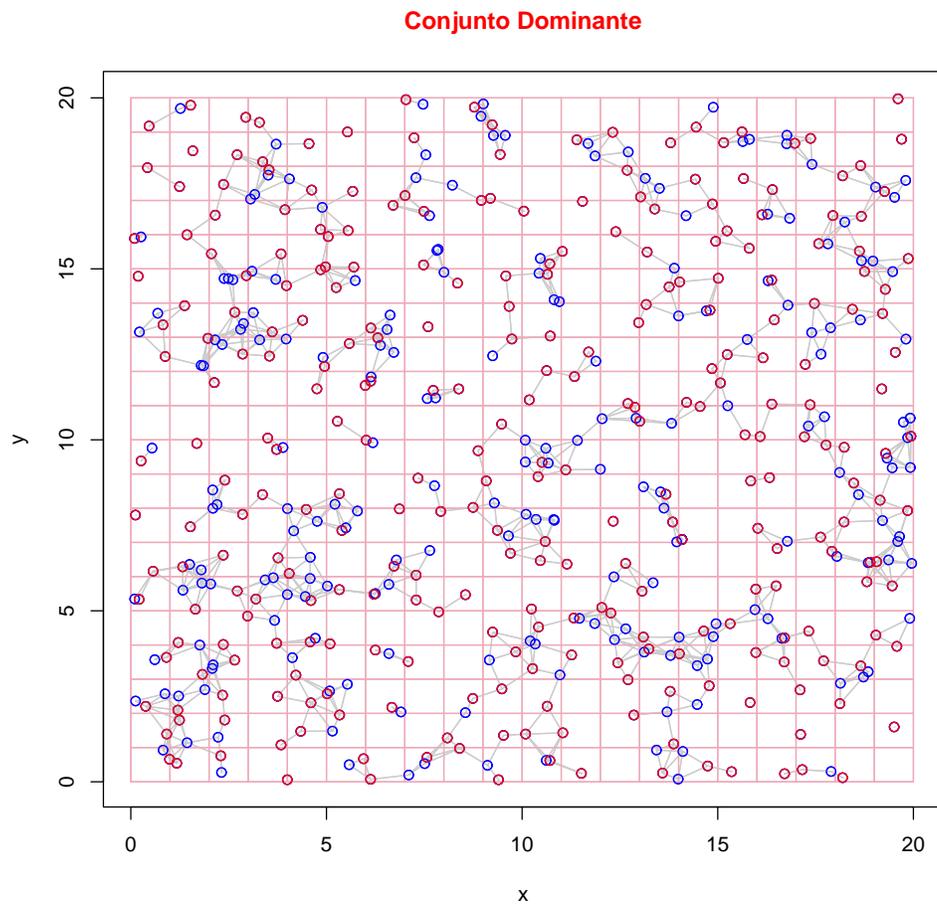


Figura A.77: Resultado de la heurística para una red de tamaño 500 y grid 20, con subdivisión cuadrada y aristas de 1cm.

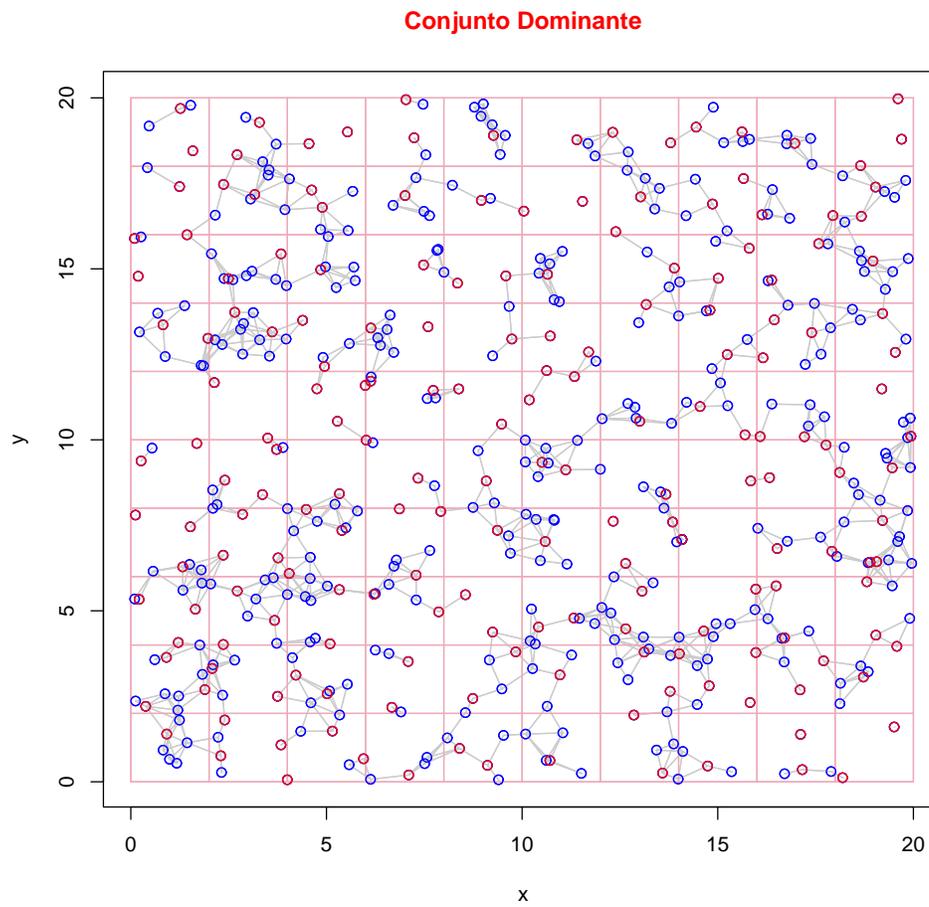


Figura A.78: Resultado de la heurística para una red de tamaño 500 y grid 20, con subdivisión cuadrada y aristas de  $2\text{cm}$ .

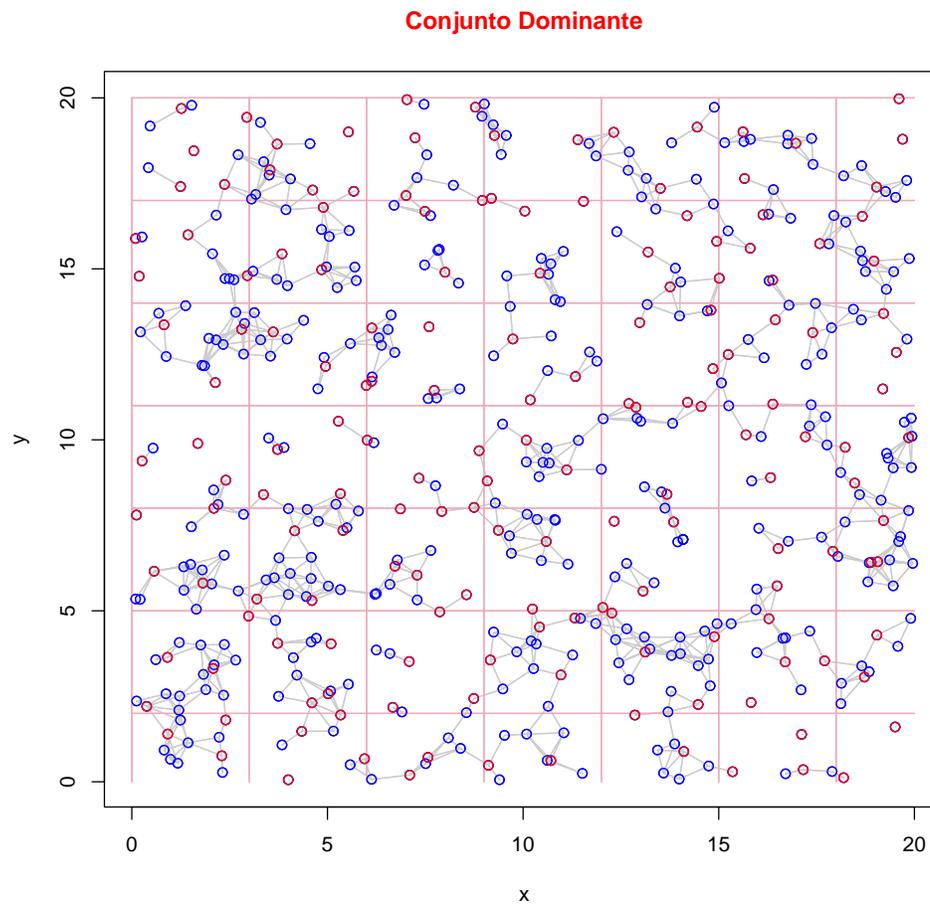


Figura A.79: Resultado de la heurística para una red de tamaño 500 y grid 20, con subdivisión cuadrada y aristas de  $3cm$ .

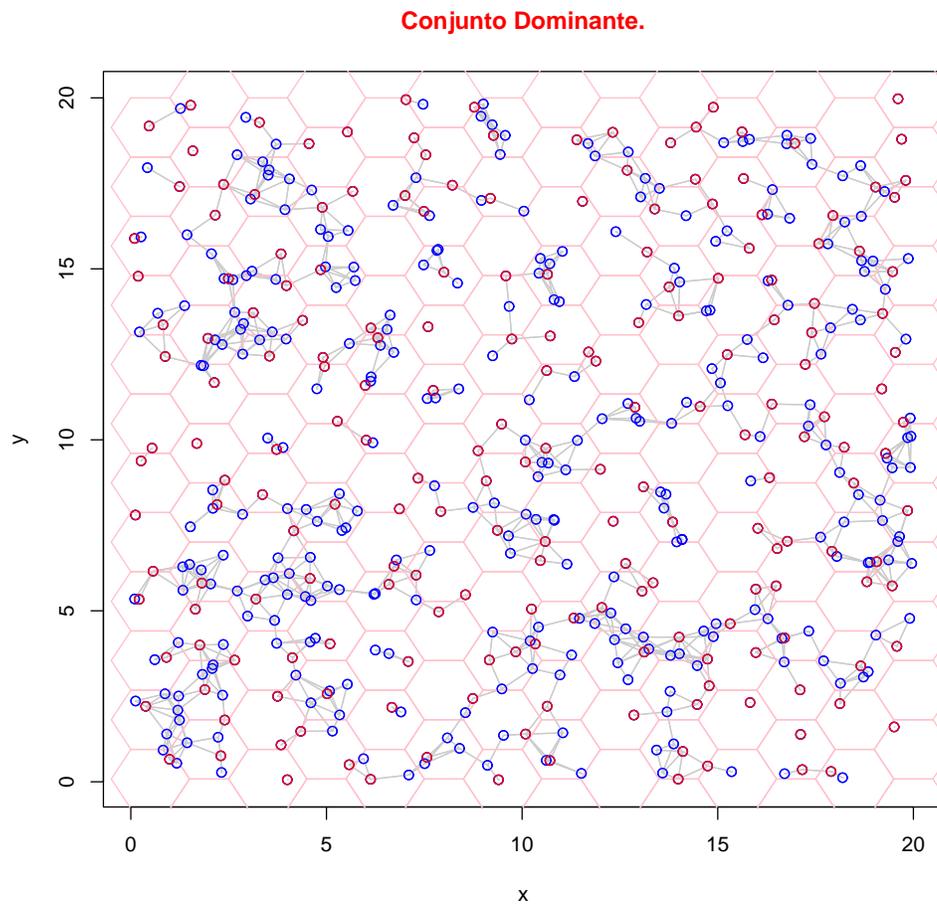


Figura A.80: Resultado de la heurística para una red de tamaño 500 y grid 20, con subdivisión hexagonal y aristas de  $1cm$ .

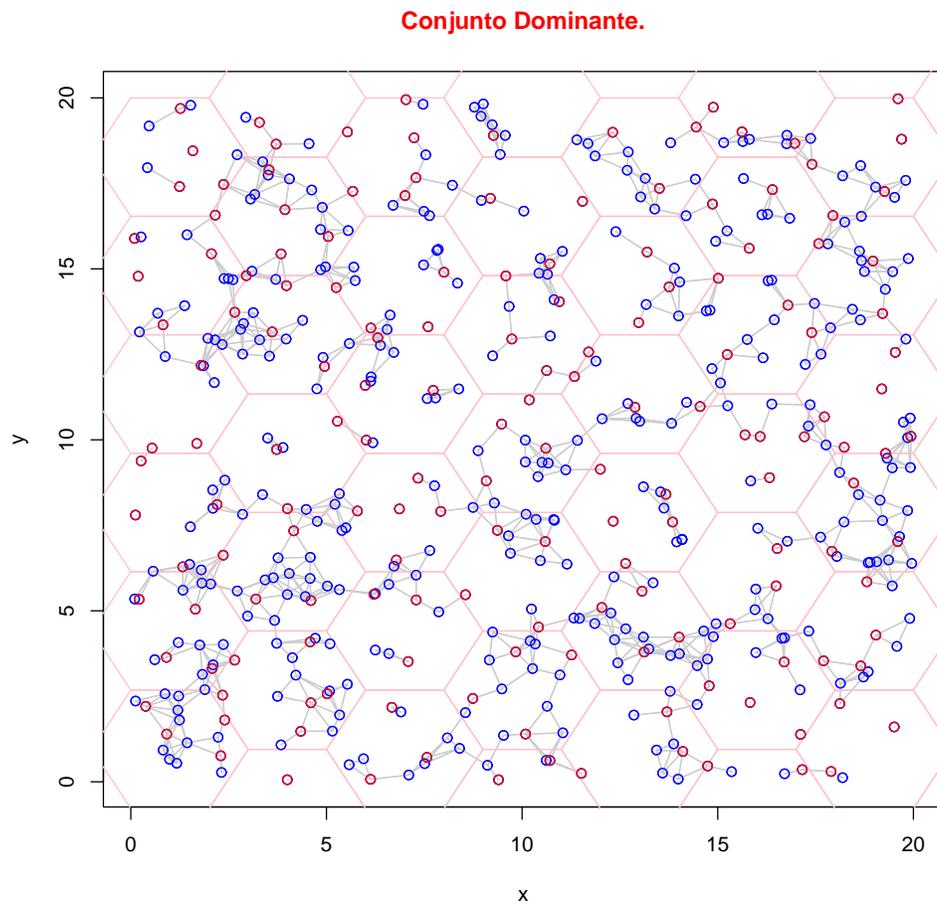


Figura A.81: Resultado de la heurística para una red de tamaño 500 y grid 20, con subdivisión hexagonal y aristas de 2cm.

## A.4. Redes de tamaño ochocientos

### A.4.1. Grid de tamaño diez

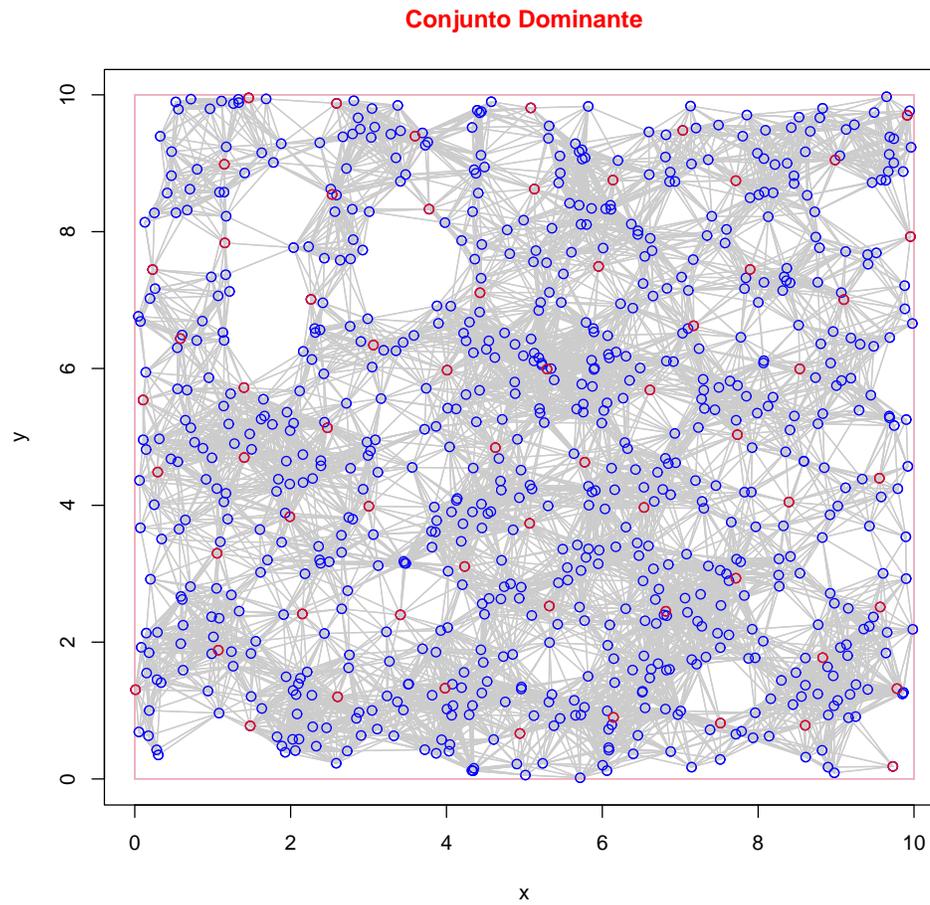


Figura A.82: Resultado del algoritmo glotón para una red de tamaño 800 y grid 10.

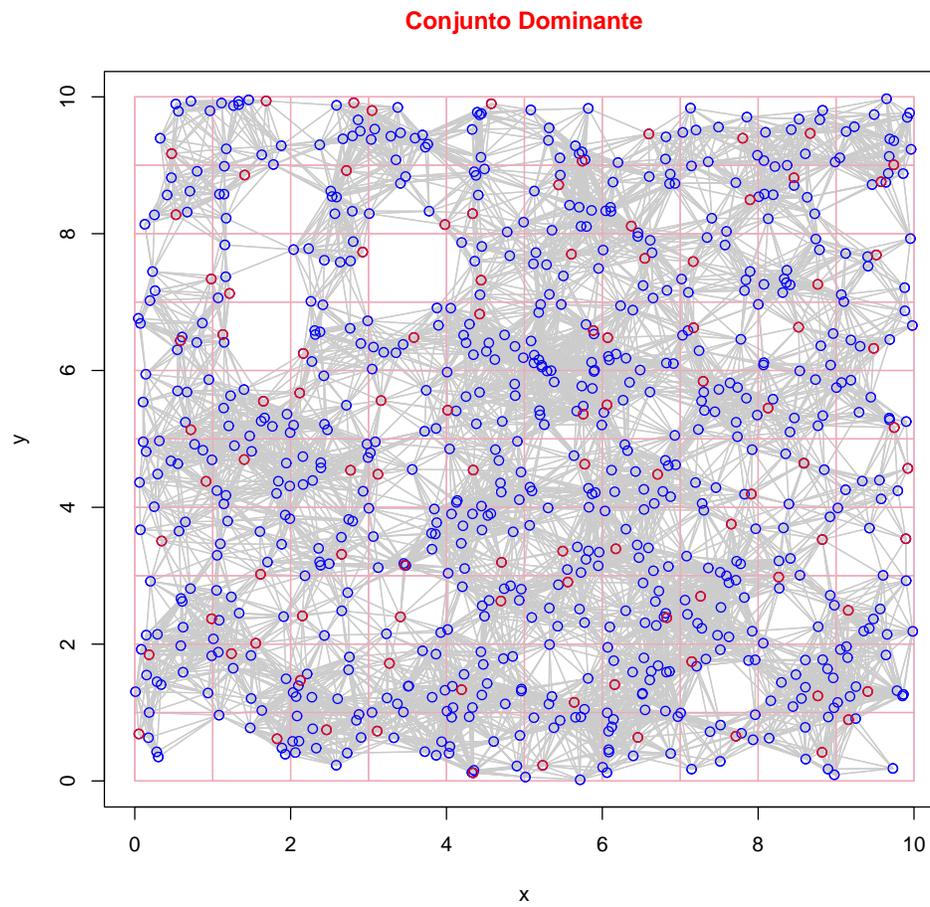


Figura A.83: Resultado de la heurística para una red de tamaño 800 y grid 10, con subdivisión cuadrada y aristas de 1cm.

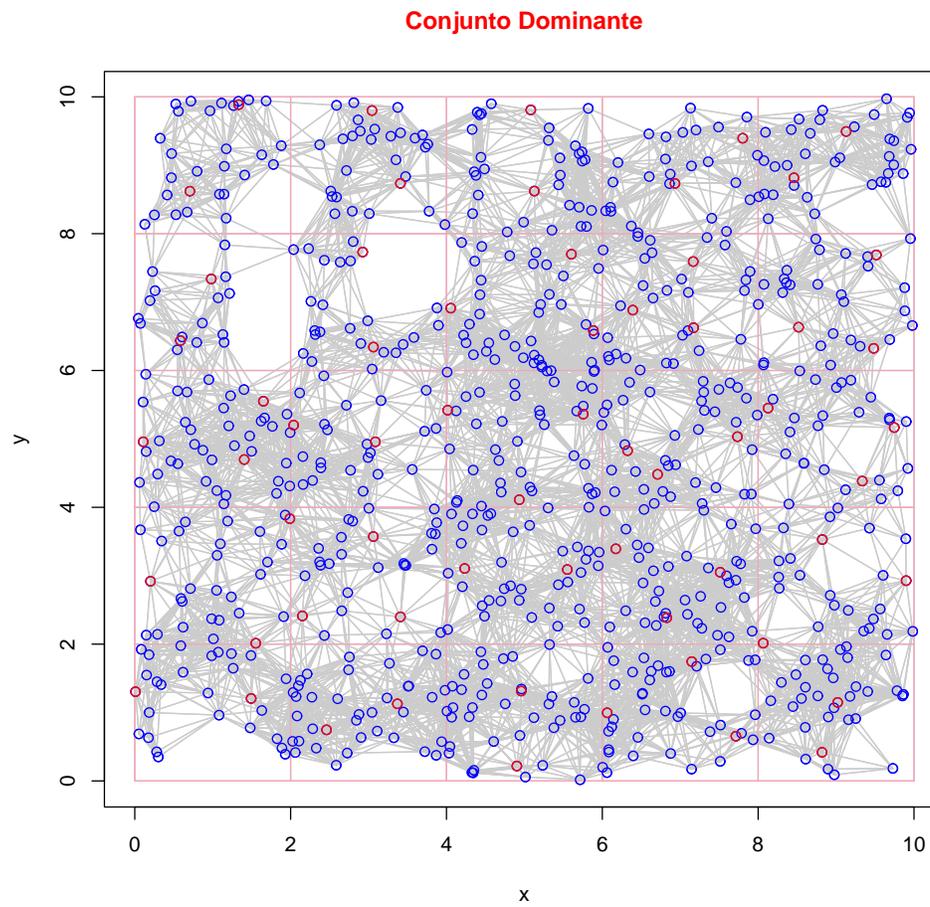


Figura A.84: Resultado de la heurística para una red de tamaño 800 y grid 10, con subdivisión cuadrada y aristas de  $2\text{cm}$ .

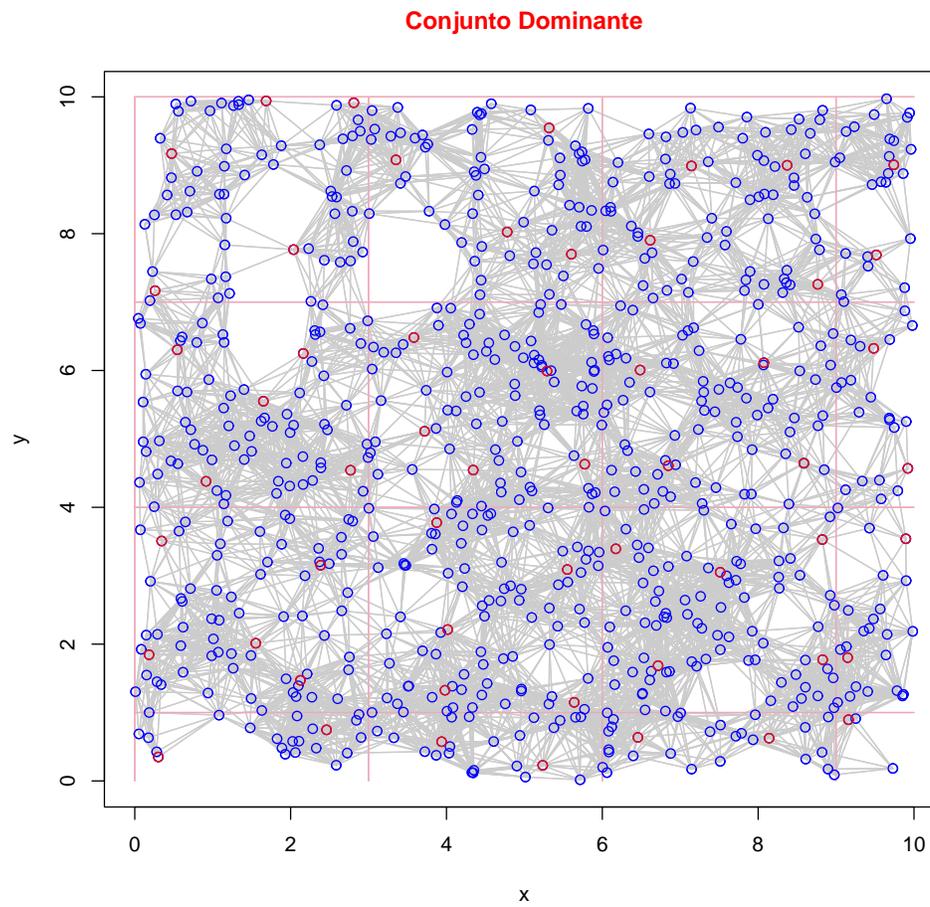


Figura A.85: Resultado de la heurística para una red de tamaño 800 y grid 10, con subdivisión cuadrada y aristas de 3cm.

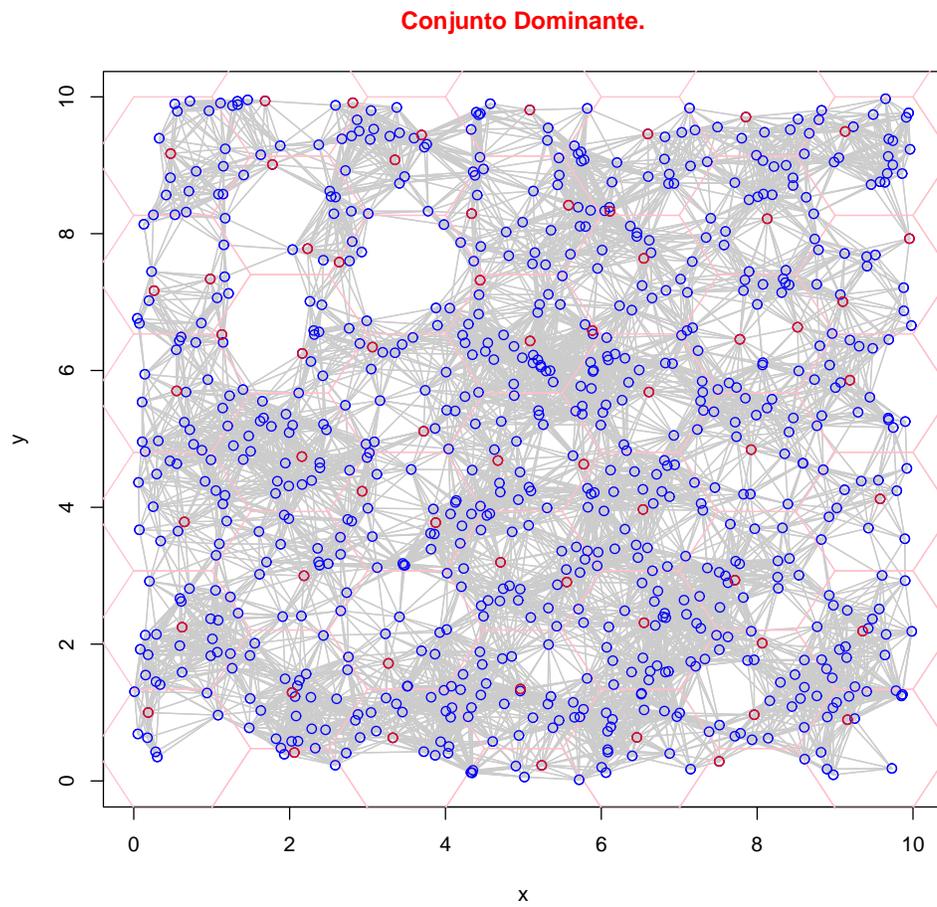


Figura A.86: Resultado de la heurística para una red de tamaño 800 y grid 10, con subdivisión hexagonal y aristas de 1cm.

## A.4.2. Grid de tamaño veinte

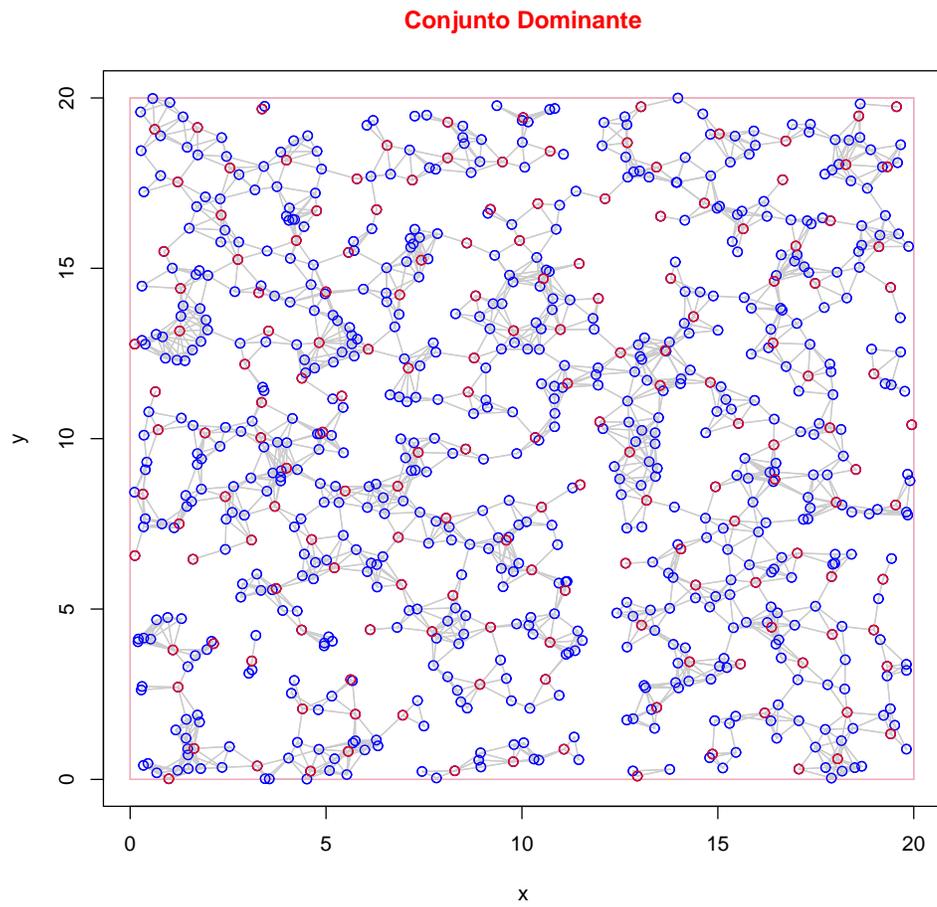


Figura A.87: Resultado del algoritmo glotón para una red de tamaño 800 y grid 20.

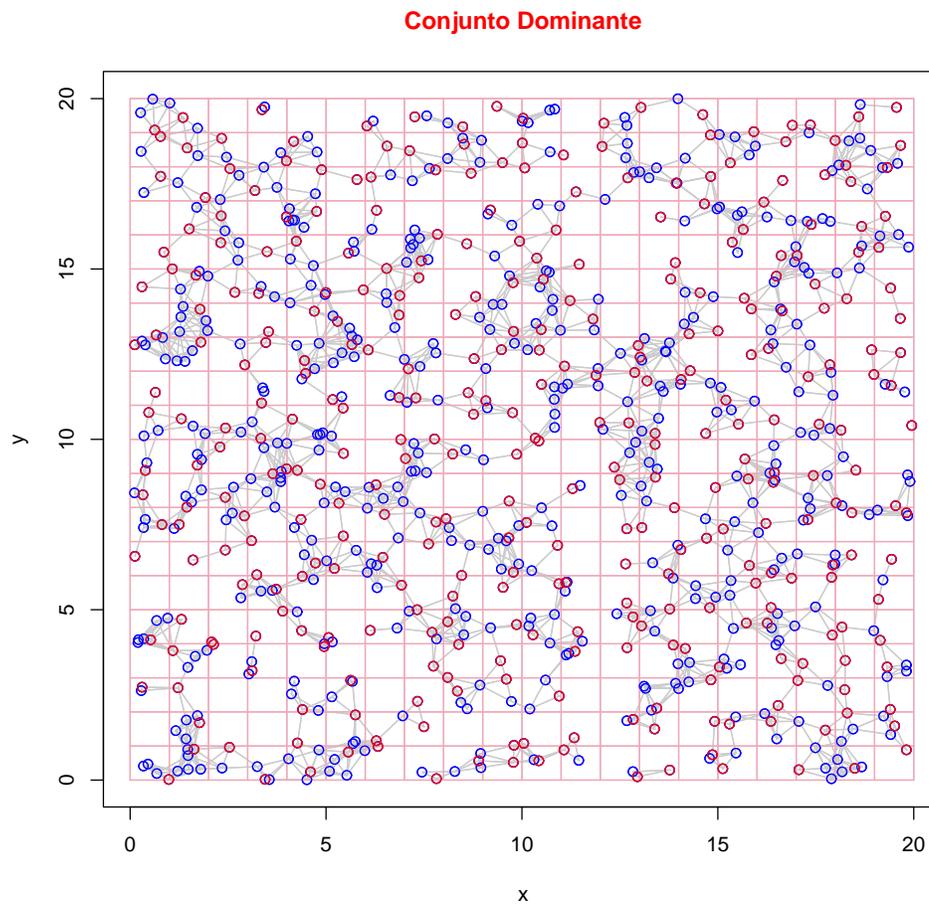


Figura A.88: Resultado de la heurística para una red de tamaño 800 y grid 20, con subdivisión cuadrada y aristas de 1cm.

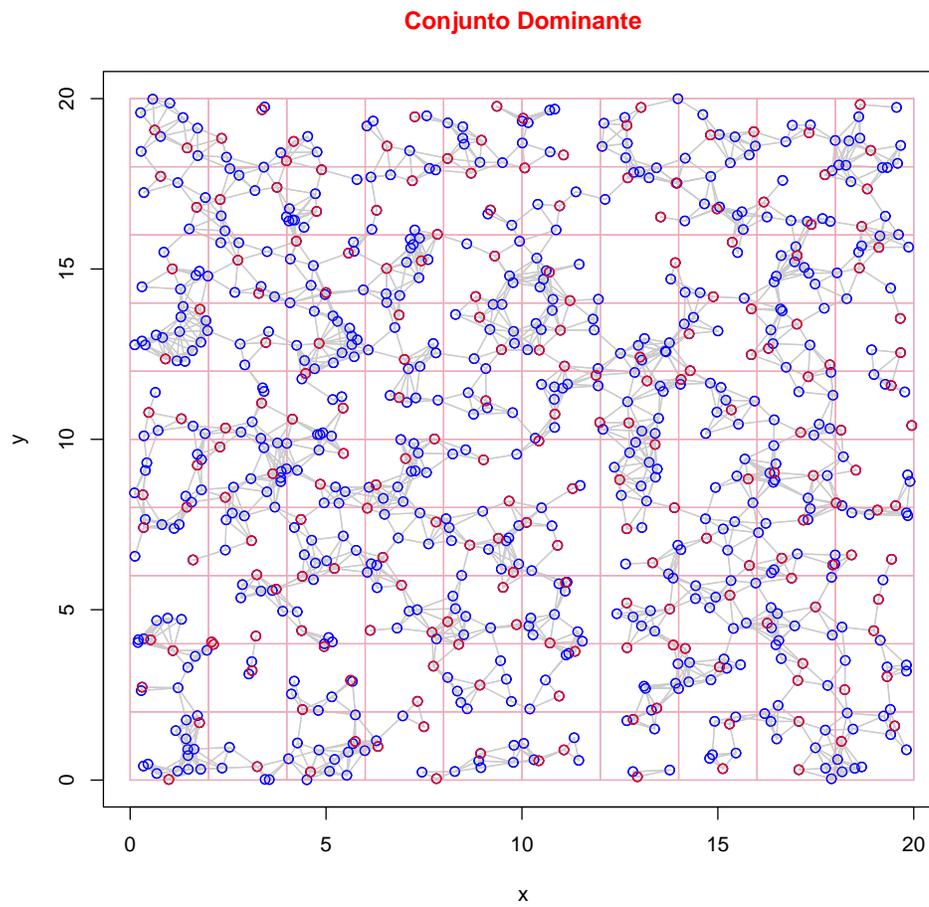


Figura A.89: Resultado de la heurística para una red de tamaño 800 y grid 20, con subdivisión cuadrada y aristas de 2cm.

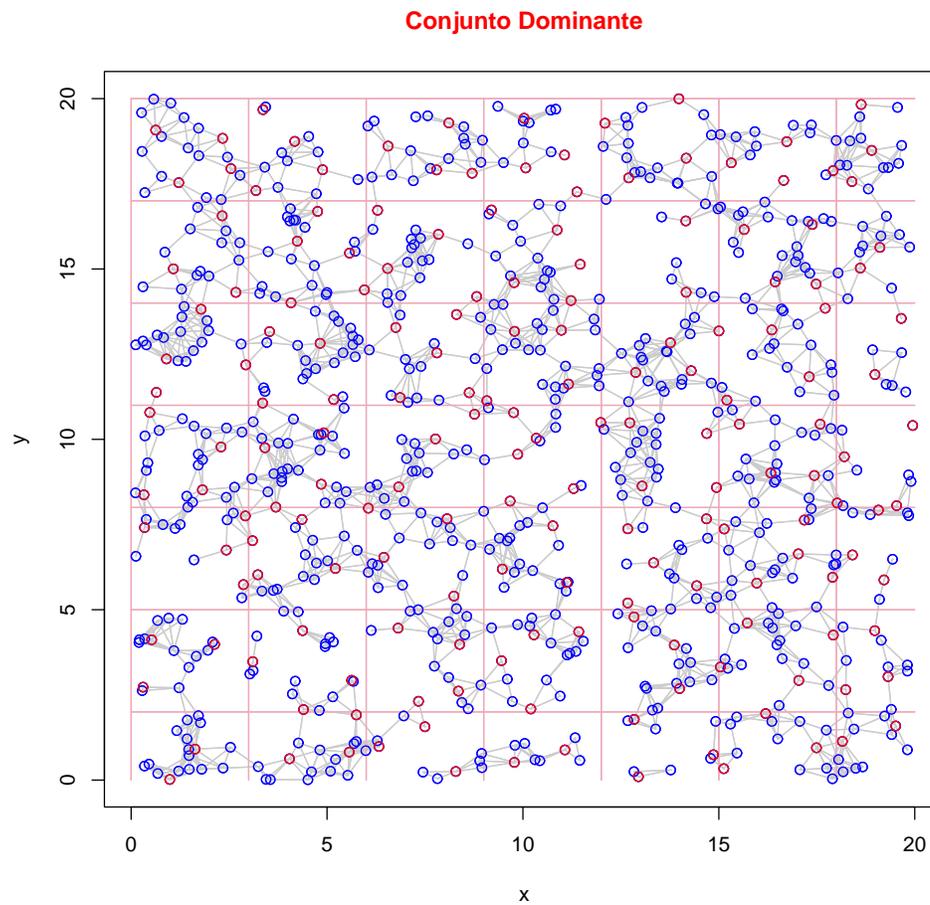


Figura A.90: Resultado de la heurística para una red de tamaño 800 y grid 20, con subdivisión cuadrada y aristas de 3cm.

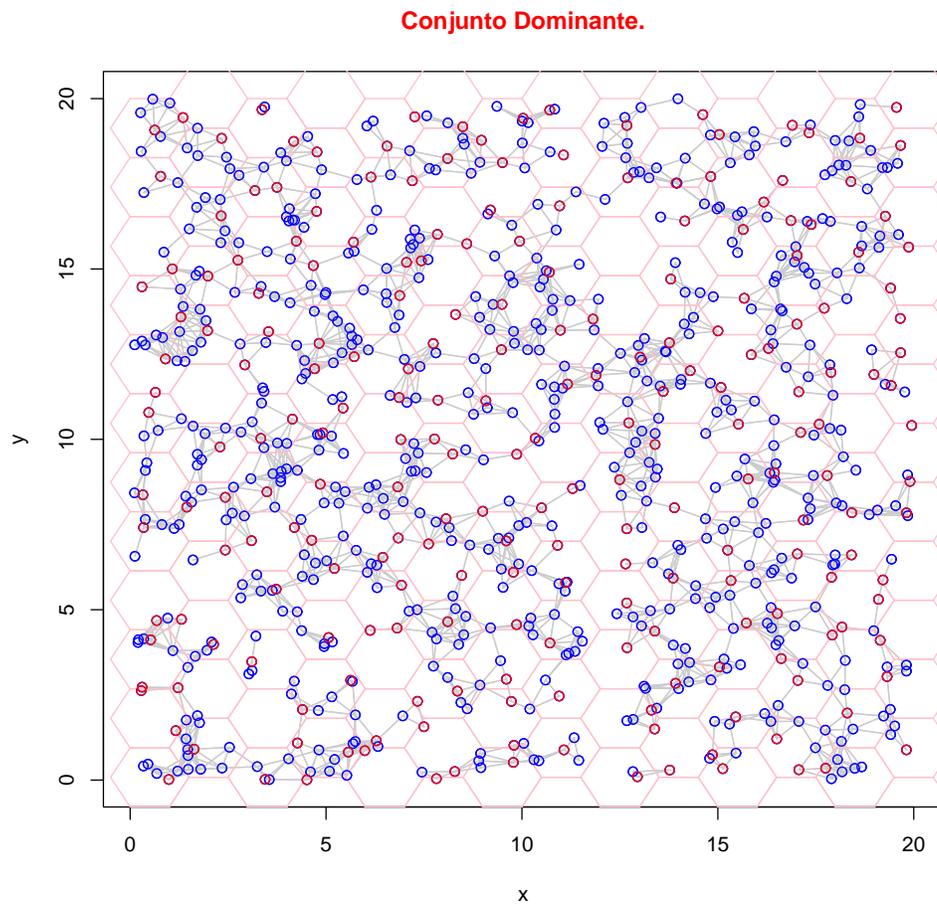


Figura A.91: Resultado de la heurística para una red de tamaño 800 y grid 20, con subdivisión hexagonal y aristas de 1cm.

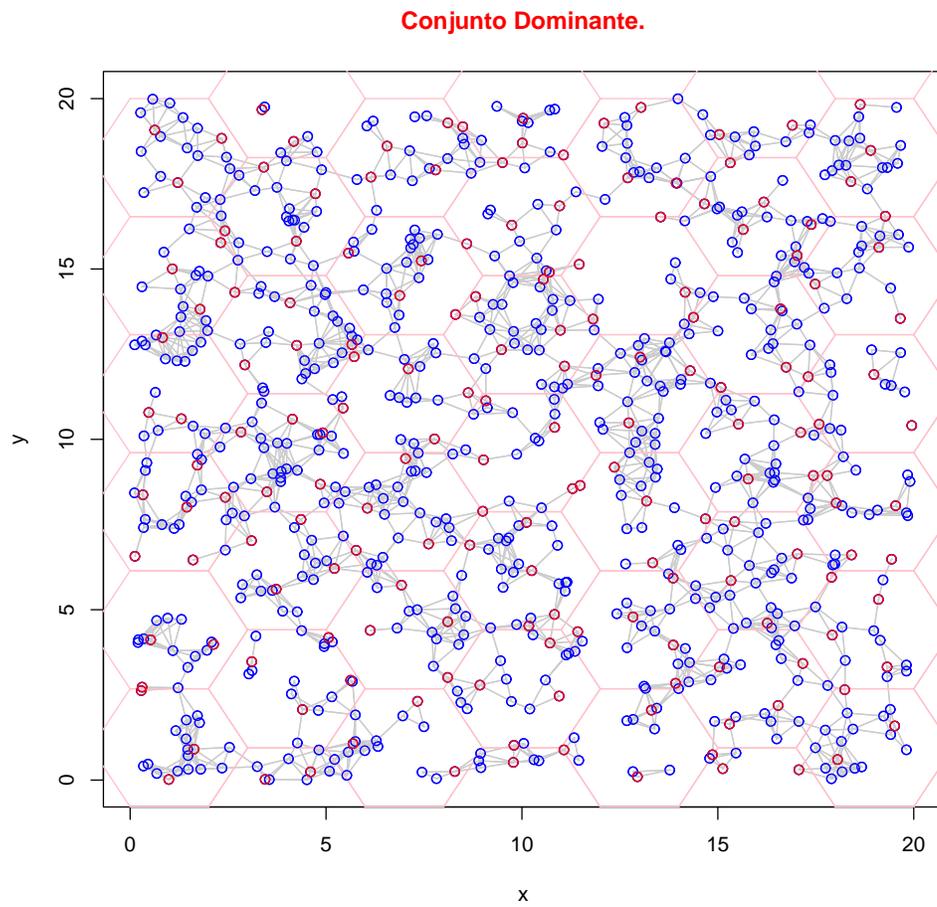


Figura A.92: Resultado de la heurística para una red de tamaño 800 y grid 20, con subdivisión hexagonal y aristas de 2cm.

## A.4.3. Grid de tamaño treinta

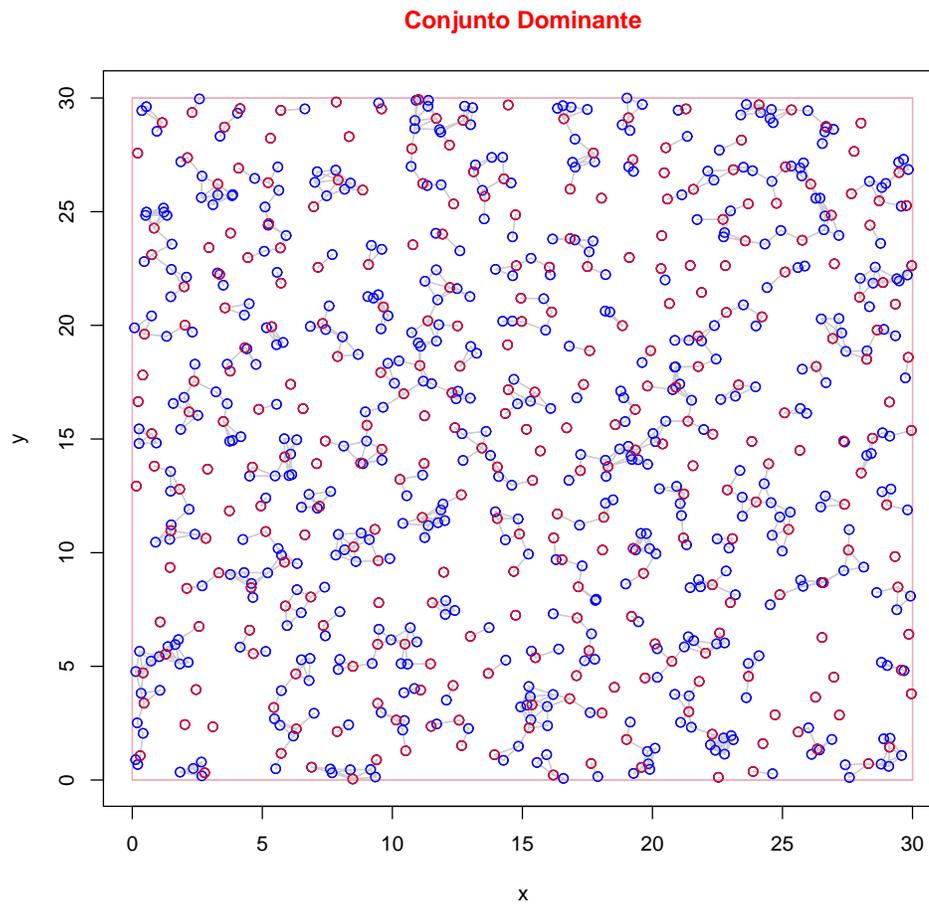


Figura A.93: Resultado del algoritmo glotón para una red de tamaño 800 y grid 30.

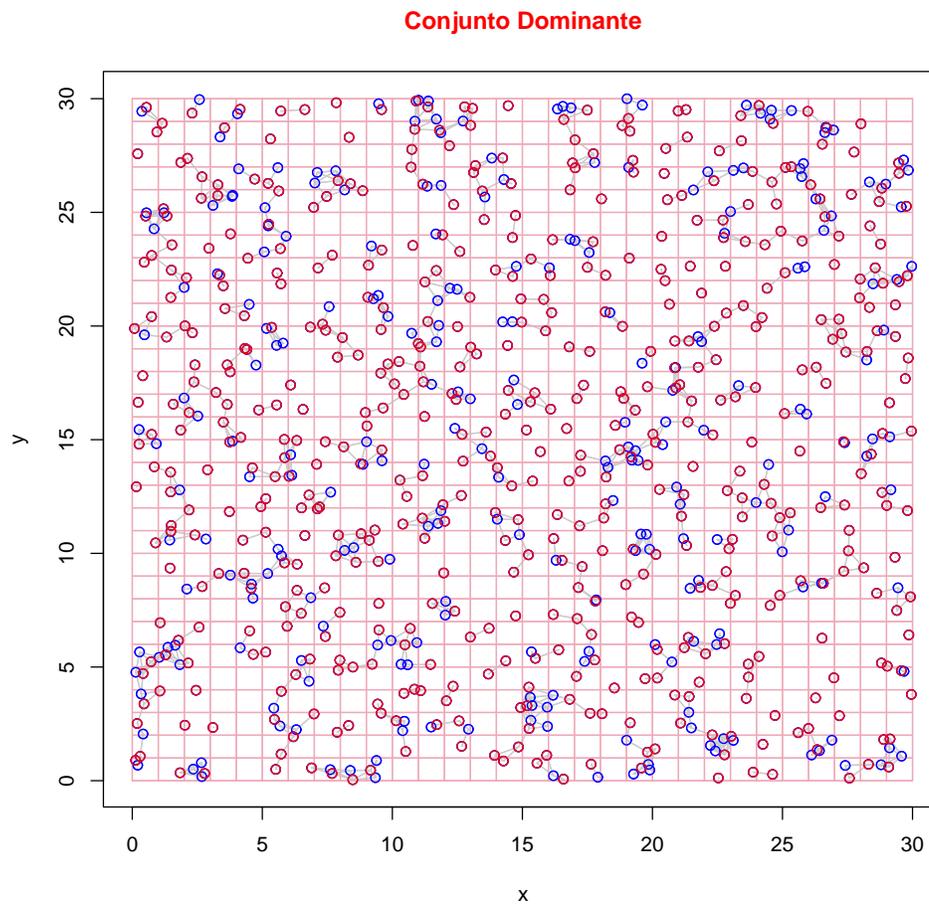


Figura A.94: Resultado de la heurística para una red de tamaño 800 y grid 30, con subdivisión cuadrada y aristas de 1cm.

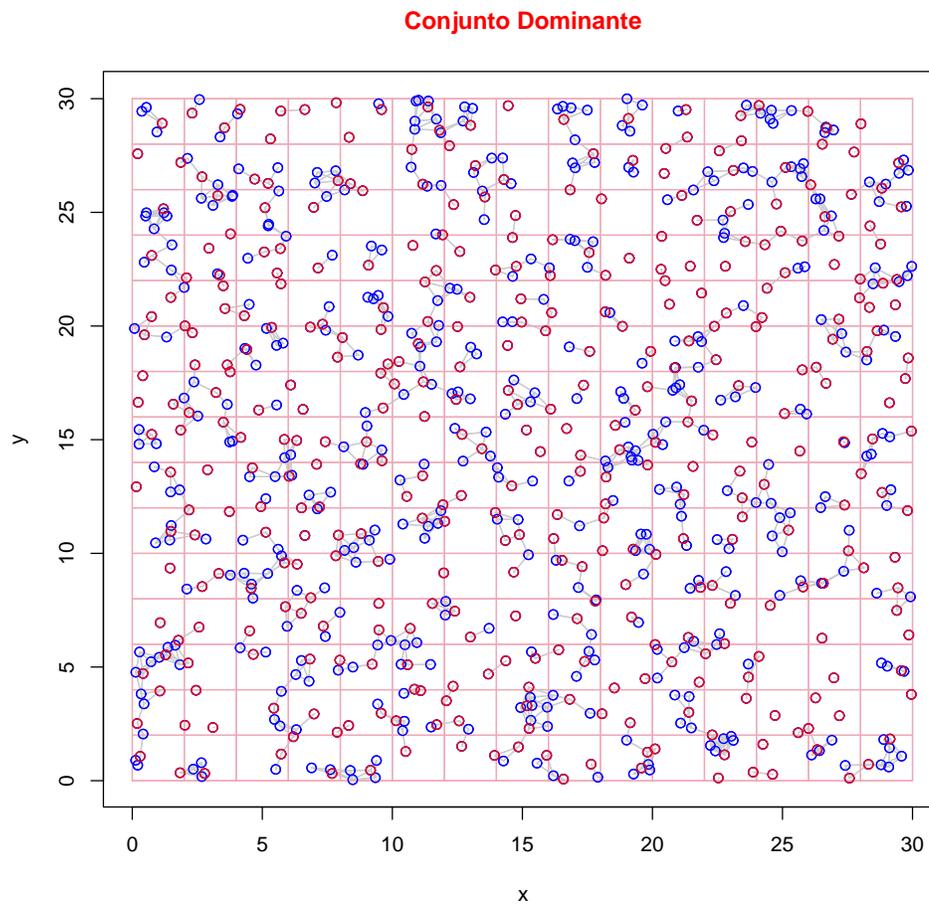


Figura A.95: Resultado de la heurística para una red de tamaño 800 y grid 30, con subdivisión cuadrada y aristas de 2cm.

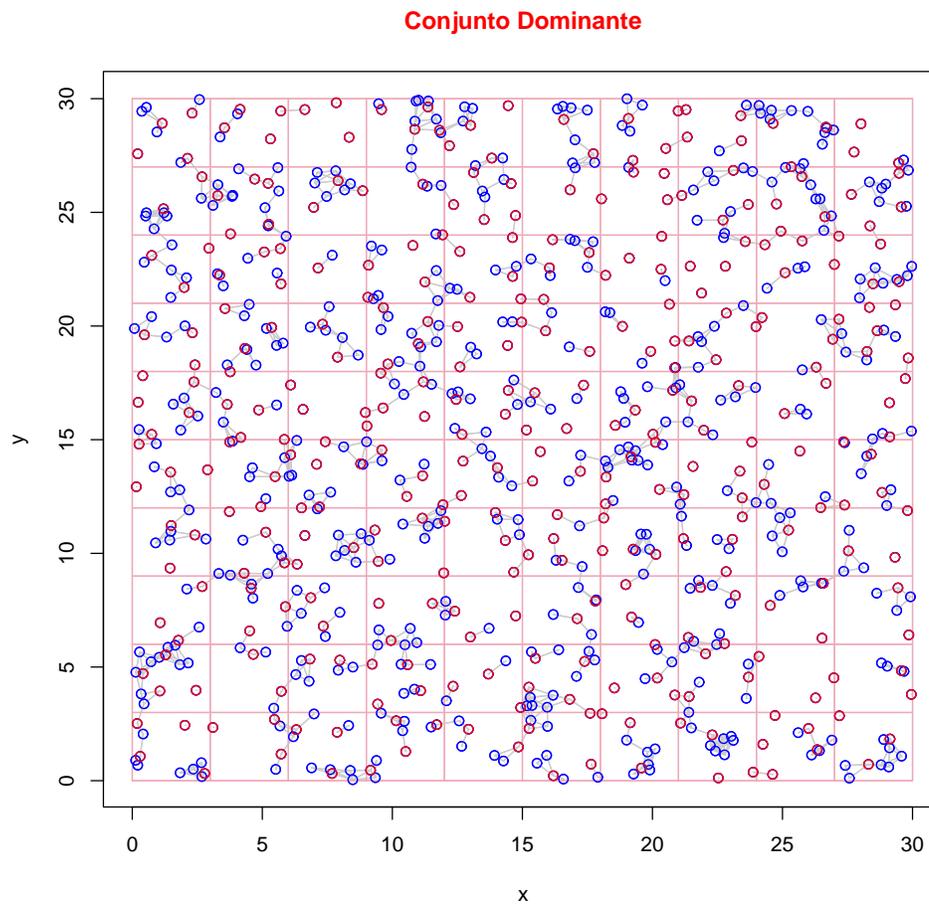


Figura A.96: Resultado de la heurística para una red de tamaño 800 y grid 30, con subdivisión cuadrada y aristas de  $3\text{cm}$ .

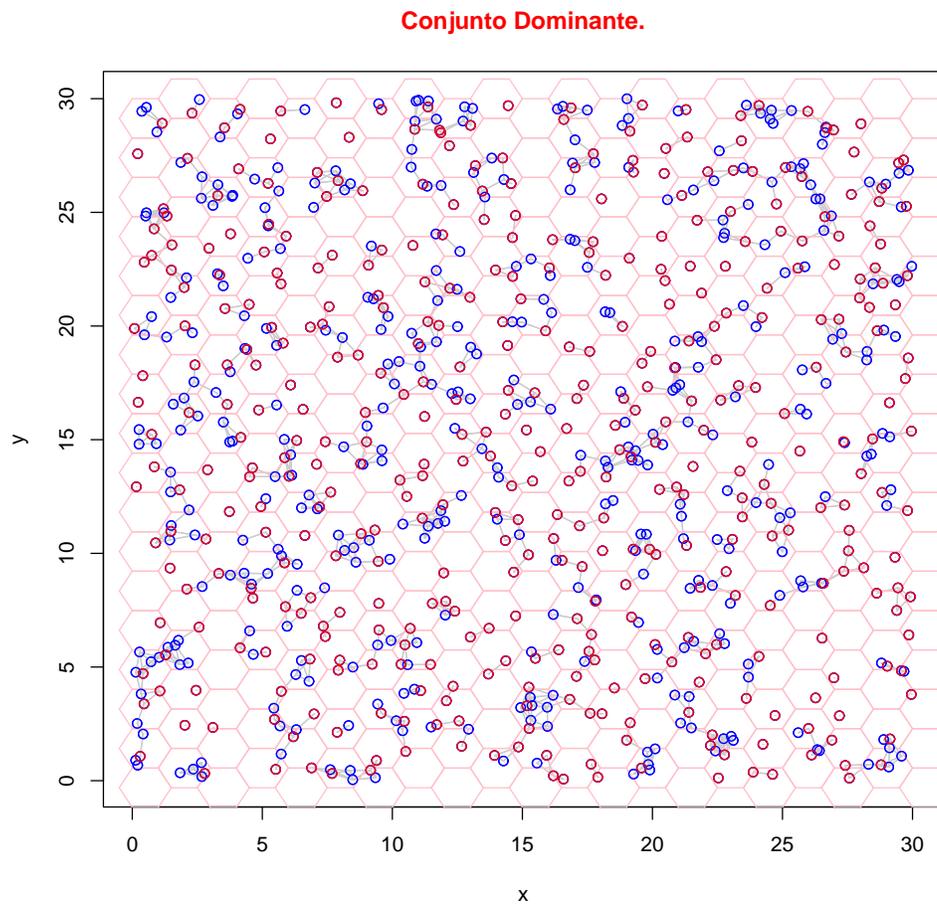


Figura A.97: Resultado de la heurística para una red de tamaño 800 y grid 30, con subdivisión hexagonal y aristas de 1cm.

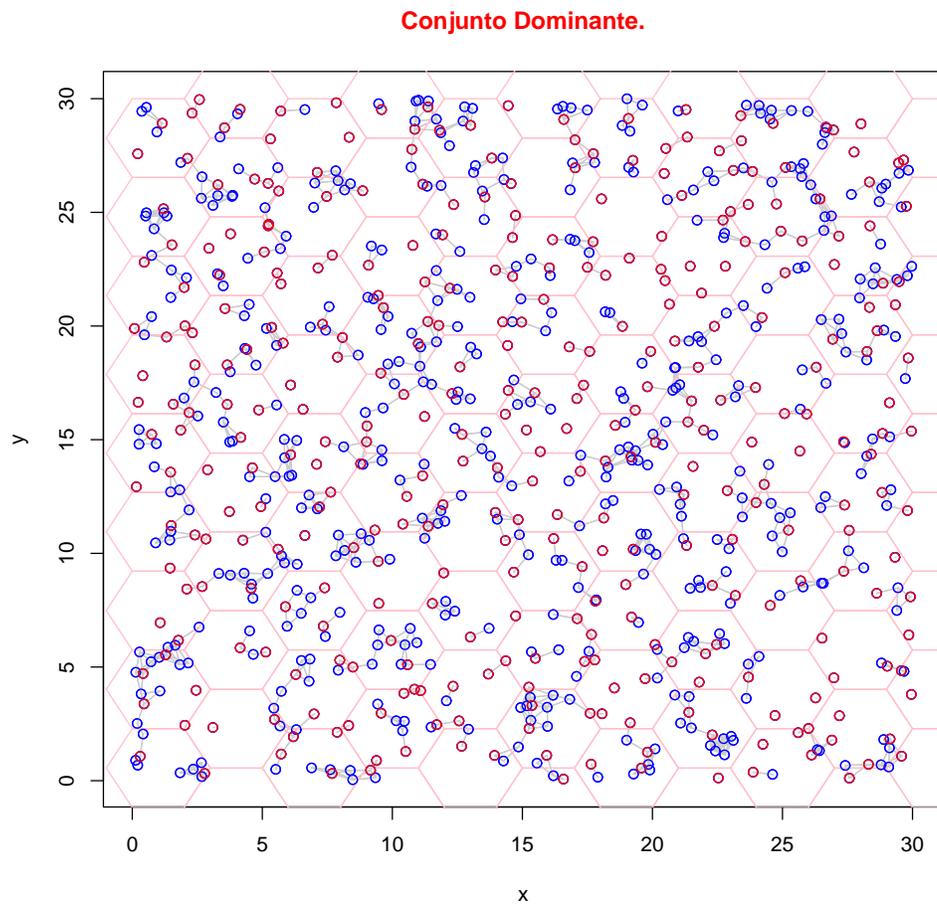


Figura A.98: Resultado de la heurística para una red de tamaño 800 y grid 30, con subdivisión hexagonal y aristas de 2cm.

# Referencias

- [1] Prosenjit Bose, Pat Morin, Ivan Stojmenovic, and Jorge Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7(6):609–616, 2001.
- [2] Magnus Frodigh, Per Johansson, and Peter Larsson. Wireless ad hoc networking—the art of networking without a network. *Ericsson Review*, 4:248–263, 2000.
- [3] N. Abramson. The ALOHA system—another alternative for computer communication. In *Fall Joint Computer Conference, AFIPS Press.*, volume 37, pages 281–285, 1970.
- [4] John Jubin and Janet D. Tornow. The DARPA packet radio network protocols. In *Proceedings of the IEEE*, volume 75, pages 21–32, 1987.
- [5] (05/2006) <http://grouper.ieee.org/groups/802/11/>.
- [6] Mobile Ad hoc Networks Working Group. Internet Engineering Task Force. (04/2006) <http://www.ietf.org/html.charters/manet-charter.html>.
- [7] S. Corson and J. Macker. RFC 2501 mobile ad hoc networking(MANET): Routing protocol performance issues and evaluation considerations. Technical report, January 1999. (04/2006) <http://www.rfc-archive.org/getrfc.php?rfc=2501>.
- [8] D. Goodman. *The Evolution of Untethered Communications*. National Academy Press, Washington DC, December 1997. <http://www.nap.edu/readingroom/books/evolution/index.html>.
- [9] Luca Benini and Giovanni De Micheli. System-level power optimization: techniques and tools. *ACM Transactions on Design Automation of Electronic Systems*, 5(2):115–192, 2000.
- [10] Robert A. Powers. Batteries for low power electronics. In *Proceedings of the IEEE*, pages 687–693, april 1995.

- [11] Lidong Zhou and Zygmunt J. Haas. Securing ad hoc networks. *Network IEEE*, 13(6):24–30, 1999.
- [12] Frank Stajano and Ross Anderson. The resurrecting duckling: security issues for ad-hoc wireless networks. In *Proceedings of the 7th International Workshop of Security Protocols*, 1999.
- [13] Bluetooth Special Interest Group (SIG). (04/2006) <https://www.bluetooth.org/>.
- [14] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38:393–422.
- [15] Stephan Preuss and Clemens Cap. Overview of spontaneous networking—evolving concepts and technologies.
- [16] L. Feeney, B. Ahlgren, and A. Westerlund. Spontaneous networking: an application-oriented approach to ad hoc networking. *IEEE Communications Magazine*, June 2001.
- [17] Ioan Chisalita and Nahid Shahmehri. A peer-to-peer approach to vehicular communication for the support of traffic safety applications. In *Proceedings of the IEEE 5th International Conference on Intelligent Transportation Systems*, pages 336–341, 2002.
- [18] Xue Yang, Jie Liu, Feng Zhao, and Nitin H. Vaidya. A vehicle-to-vehicle communication protocol for cooperative collision warning. In *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems(MobiQuitous)*, pages 114–123, 2004.
- [19] Holger Fler and Martin Mauve. A comparison of routing strategies for vehicular ad-hoc networks.
- [20] Hannes Hartenstein, Bernd Bochow, André Ebner, Matthias Lott, Markus Radimirsch, and Dieter Vollmer. Position-aware ad hoc wireless networks for inter-vehicle communications: the Flenet project. In *Proceedings of the 2nd ACM International Symposium on Mobile ad hoc Networking and Computing*, pages 259–262, 2001.
- [21] S. Xu and T. Saadawi. Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks? *IEEE Communications Magazine*, 39(6):130–137, 2001.
- [22] L. Kleinrock and F. A. Tobagi. Packet switching in radio channels: Part I—carrier sensing multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, COM-23:1400–1416, December 1975.

- [23] R. M. Metcalfe and D. R. Boggs. Ethernet: Distributed packet switching for local computer networks. *Communications of the Associations for Computing Machinery*, 19(7):395–404, July 1976.
- [24] S. Lam. A carrier sense multiple access protocol for local networks. *Computer Networks*, 4:21–32, 1980.
- [25] R. Rom and M. Sidi. *Multiple Access Protocols: Performance and Analysis*. Springer-Verlag, 1990.
- [26] P. Karn. MACA—a new channed access method for packet radio. In *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, pages 134–140, 1990.
- [27] V. Bharghavan, A. Demets, S. Shenker, and L. Zhang. MACAW: A media access protocol for wireless LAN’s. In *Proceedings of ACM SIGCOMM ’94 Conference*, pages 212–225, 1994.
- [28] F. Talucci and M. Gerla. MACA-BI: A wireless MAC protocol for high speed ad hoc networking. In *Proceedings of ICUPC ’97*, 1997.
- [29] Zygmunt J. Haas and Jing Deng. Dual Busy Tone Multiple Access (DBTMA)—A multiple access control scheme for ad hoc networks. *IEEE Transactions on Communications*, 50(6):975–985, 2002.
- [30] A. Colvin. CSMA with collision avoidance. *Computer Communications*, 6(5):227–235, October 1983.
- [31] I. F. Akyildiz, J. McNair, L. C. Martorell, R. Puigianer, and Y. Yesha. Medium access control protocols for multimedia traffic in wireless networks. *Network IEEE*, 13(4):39–47, 1999.
- [32] Samba Sesay, Zongkai Yang, and Jianhua He. A survey on mobile ad hoc wireless networks. *Information Technology Journal*, 3(2):168–175, 2004.
- [33] Martin Mauve, Jörg Widmer, and Hannes Hartenstein. A survey on position-based routing in mobile ad hoc networks. *IEEE Network*, 15(6):30–39, 2001.
- [34] James F. Kurose and Keith W. Ross. *Computer Networking: A top-down approach featuring the internet*. Addison Wesley, 2001.
- [35] Per Johansson, Tony Larsson, and Nicklas Hedman. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing Networking*, pages 195–206, 1999.

- [36] David B Johnson and David A Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.
- [37] Vincent D. Park and M. Scott Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM (3)*, pages 1405–1413, 1997.
- [38] C. Perkins. Ad-hoc on-demand distance vector routing, 1997.
- [39] B. Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global Positioning System: Theory and Practice*. Four edition, 1994.
- [40] B. W. Parkinson and J. J. Spilker. Global positioning system: Theory and applications. *Progress in Astronautics and Aeronautics*, 163, 1996.
- [41] S. Capkun, M. Hamdi, and J-P Hubaux. Gps-free positioning in mobile ad-hoc networks. In *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, pages 100–110, January 2001.
- [42] Niculescu and Nath Badri. Ad hoc positioning system (APS) using AOA. In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 3, pages 1734–1743, April 2003.
- [43] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, August 2001.
- [44] E. M. Royer and Toh Chai-Keong. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 6(2):46–55, April 1999.
- [45] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, 2005.
- [46] J. Al-Karaki and A. E. Kamal. Routing techniques in wireless sensor networks: a survey. *Wireless Communications IEEE*, 11(6):6–28, December 2004.
- [47] Ivan Stojmenovic, Anand Prakash Ruhil, and D. K. Lobiyal. Voronoi diagram and convex hull based geocasting and routing in wireless networks. *Wireless Communications and Mobile Computing*, 6:247–258, 2006.
- [48] Das B. and Bharghavan V. Routing in ad-hoc networks using minimum connected dominating sets. In *Proceedings of the IEEE International Conference on Communications, 1997*, volume 1, pages 376–380, 1997.

- [49] Jie Wu, Ming Gao, and Ivan Stojmenovic. On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks. In *Proceedings of the 2001 International Conference on Parallel Processing*, volume 1, page 346, 2001.
- [50] Jie Wu and Hailan Li. A dominating-set-based routing scheme in ad hoc wireless networks. *Telecommunication Systems*, 18(1-3):13–36, september 2001.
- [51] Samba Sesay, Zongkai Yang, and Jianhua He. A survey on mobile ad hoc wireless networks. *Information Technology Journal*, 3(2):168–175, 2004.
- [52] A. Spyropoulos and C. S. Raghavendra. An energy efficient communications in ad hoc networks using directional antennas. In *Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, volume 1, pages 220–228, 2002.
- [53] R. Ramanathan, J. Redi, C. Santiva nez, D. Wiggins, and S. Polit. Ad hoc networking with directional antennas: a complete system solution. *IEEE Journal on Selected Areas in Communications*, 23(3):496–506, March 2005.
- [54] S. Agarwal, R. H. Katz, S. V. Krishnamurthy, and S. K. Dao. Distributed power control in ad-hoc wireless networks. In *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 2, pages F59–F66, 2001.
- [55] Vitas Kawadia and P. R. Kumar. Power control and clustering in ad hoc networks. In *Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, volume 1, pages 459–469, 2003.
- [56] Ye Wei, J. Heidemann, and D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. In *Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, volume 3, pages 1567–1576, 2002.
- [57] Chunlong Guo, Lizhi Charlie Zhong, and J. M. Rabaey. Low power distributed MAC for ad hoc sensor radio networks. In *Proceedings of the Global Telecommunications Conference (GLOBECOM '01)*, volume 5, pages 2944–2948, 2001.
- [58] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless Networks*, 8(5):481–494, September 2002.

- [59] Chai-Keong Toh. Associativity-based routing for ad hoc mobile networks. *Wireless Personal Communications*, 4(2):130–139, March 1997.
- [60] C. K. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE Communications Magazine*, 39(6):138–147, Jun 2001.
- [61] Suresh Singh, Mike Woo, and C. S. Ragavendra. Power aware routing in mobile ad hoc networks. In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pages 181–190, 1998.
- [62] R. C. Shah and J. M. Rabaey. Energy aware routing for low energy ad hoc sensor networks. In *Wireless Communications and Networking Conference (WCNC2002)*, volume 1, pages 350–355, March 2002.
- [63] Robin Kravets and P. Krishnan. Application driven power management for mobile communication. *Wireless Networks*, 6(4):263–277, July 2000.
- [64] S. Singh and C. S. Raghavendra. PAMAS: Power aware multi-access protocol with signaling for ad hoc networks. *ACM SIGCOMM Computer Communications Review*, 28(3):5–26, July 1998.
- [65] Y. Xu, J. Heidemann, and D. Estrin. Geography informed energy conservation for ad hoc routing. In *Proceedings of the 7th ACM/IEEE International Conference on Mobile Computing and Networking*, July 2001.
- [66] B. Chen, K. Jamieson, R. Morris, and H. Balakrishnan. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *Proceedings of the 7th ACM/IEEE International Conference on Mobile Computing and Networking*, July 2001.
- [67] Jae-Hwan Chang and Leandros Tassiulas. Energy conserving routing in wireless ad hoc networks. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, volume 1, pages 22–31, 2000.
- [68] B. Prabhakar, E. Uysal Biyikoglu, and A. El Gamal. Energy efficient transmission over a wireless link via lazy packet scheduling. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 386–394, 2001.
- [69] Design challenges for energy-constrained ad hoc wireless networks. *IEEE Wireless Communications*, 9(4):8–27, August 2002.

- [70] Crawley, Nair, Rajagopalan, and Sandick. A framework for QoS-based routing in the internet. RFC 2386, Network Working Group, August 1998.  
(05/2006) <http://rfc.sunsite.dk/rfc/rfc2386.html>.
- [71] Z Wang. *Internet QoS: Architectures and Mechanisms for Quality of Service*. 2001.
- [72] B Braden, D Clark, and S Shenker. RFC14ZZ integrated service in the internet architecture: An overview. Technical report, September 1993.  
(05/2006) <https://dspace.mit.edu/bitstream/1721.1/1542/1/isia.pdf>.
- [73] Zheng Wang and J. Crowcroft. Quality-of-service routing for supporting multimedia applications. *IEEE Journal on Selected Areas in Communications*, 14(7):1228–1234.
- [74] R. Braden, D. Clark, and S. Shenker. RFC1633–integrated services in the Internet architecture: an overview. Technical report, 1994.
- [75] S. Blake, D. Black, M. Carlson, E. Davies, and Z. Wang. RFC2475—an architecture for differentiated services. Technical report, 1998.
- [76] Indu Mahadevan and Krishna M. Sivalingam. Quality of service architectures for wireless networks: Intserv and diffserv models. In *Proceedings of the Four International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN '99)*, pages 420–425, 1999.
- [77] Hannan Xiao, W. K. G. Seah, A. Lo, and K. C. Chua. A flexible quality of service model for mobile ad-hoc networks. In *51st Vehicular Technology Conference Proceedings*, volume 1, pages 445–449, 2000.
- [78] Kui Wu and Janelle Harms. QoS support in mobile ad hoc networks. *Crossing Boundaries—An Interdisciplinary Journal*, 1(1):92–106, 2001.
- [79] Satyabrata Chakrabarti and Amitabh Mishra. QoS issues in ad hoc wireless networks. *IEEE Communications Magazine*, 39(2):142–148, 2001.
- [80] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, 7(5):16–27, october 2000.
- [81] Chenxu Zhu and M. Scott Corson. QoS routing for mobile ad hoc networks. In *Proceedings of the Twenty-first Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002)*, volume 2, pages 958–967, 2002.

- [82] Chunhung Richard Lin and Jain-Shing Liu. QoS routing in ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1426–1438, 1999.
- [83] Shigang Chen and Nahrstedt. Distributed quality-of-service routing in ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1488–1505, August 1999.
- [84] Qi Xue and Aura Ganz. Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks. *Journal of Parallel and Distributed Computing*, 63:154–165, 2003.
- [85] Mehdi Behzad, Gary Chartrand, and Linda Lesniak-Foster. *Graphs and Digraphs*. Wadsworth International Group.
- [86] Leonhard Euler. Elementa doctrinae solidorum—demostratio nonnullarum insignium proprietatum, quibus solida hedris planis inclusa sunt praedita. *Novi comment acad. sc. imp. Petropol*, 4:1752–1753.
- [87] Evangelos Kranakis, Harvinder Singh, and Jorge Urrutia. Compass routing on geometric networks. In *Proceedings of the 11th Canadian Conference on Computational Geometry*, CCCG-99, pages 51–54, August 1999.
- [88] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. 1976.
- [89] A. Okabe, B. Boots, and K. Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. 1992.
- [90] Stephen A. Cook. The complexity of theorem-proving procedures. In *Third Annual ACM Symposium on Theory of Computing*.
- [91] Alan Gibbons. *Algorithmic Graph Theory*.
- [92] Haynes, Hedetniemi, and Slater. *Domination in Graphs—Advanced Topics*. CRC, 1998.
- [93] Haynes, Hedetniemi, and Slater. *Fundamentals of Domination in Graphs*. CRC, 1998.
- [94] V. Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operation Research*, 4(3):233–235, 1979.
- [95] Fabian Kuhn. *The price of locality: exploring the complexity of distributed coordination primitives*. PhD thesis, Swiss Federal Institute of Technology Zurich, 2005.

- [96] U. Feige. A threshold of  $\ln n$  for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.
- [97] Jie Wu and Hailan Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, volume 1, pages 7–14, 1999.
- [98] Peng-Jun Wan, Khaled M. Alzoubi, and Ophir Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. *Mobile Networks and Applications*, 9(2):141–149, April 2004.
- [99] Lujun Jia, Rajmohan Rajaraman, and Torsten Suel. An efficient distributed algorithm for constructing small dominating sets. *Distributed Computing*, 15(4):193–205, December 2002.
- [100] Alzoubi, Peng-Jun Wan, and Frieder. New distributed algorithm for connected dominating set in wireless ad hoc networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, volume 1, pages 3849–3855, January 2002.
- [101] Fabian Kuhn and Roger Wattenhofer. Constant-time distributed dominating set approximation. *Distributed Computing*, 17(4):303–310, May 2005.
- [102] Yu Wang and Xiang-Yang Li. Geometric spanners for wireless ad hoc networks. In *Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems*, volume 1, 2002.
- [103] E. Chaves, S. Dobrev, E. Kranakis, J. Opatrny, L. Stacho, and J. Urrutia. Local coloring of the vertices of planar spanners in unit disk graphs. *To appear*.
- [104] Richard Kershner. The number of circles covering a set. *American Journal of Mathematics*, 61(3):665–671, July 1939.
- [105] N. Oler. An inequality in the geometry of numbers. *Acta Math.*, 105:19–48, 1961.
- [106] J. H. Folkman and R. L. Graham. A packing inequality for compact convex subsets of the plane. *Canad. Math. Bull.*, 12:745–752, 1969.
- [107] Graham, Knuth, and Patashnik. *Concrete Mathematics*. Addison Wesley, 1989.
- [108] Madhav V. Marathe, H. Breu, Harry B. Hunt III, S. S. Ravi, and Daniel J. Rosenkrantz. Geometry based heuristics for unit disk graphs. LACES 68Q-94-22.

- [109] The R project for statistical computing.  
(06/2006)<http://www.r-project.org/>.