

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

-----o0o-----



Lê Đình Thanh

**HỖ TRỢ ĐỊNH VỊ VÀ NÂNG CAO HIỆU NĂNG
ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ
CHO CÁC MẠNG CẢM BIẾN KHÔNG DÂY**

LUẬN ÁN TIẾN SỸ CÔNG NGHỆ THÔNG TIN

Hà Nội - 2014

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ
-----oOo-----



Lê Đình Thanh

**HỖ TRỢ ĐỊNH VỊ VÀ NÂNG CAO HIỆU NĂNG
ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ
CHO CÁC MẠNG CẢM BIẾN KHÔNG DÂY**

Chuyên ngành: Truyền Dữ liệu và Mạng Máy tính
Mã số: 62.48.15.01

LUẬN ÁN TIẾN SỸ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC:

1. PGS. TS. HỒ THUẦN
2. TS. NGUYỄN ĐẠI THỌ

Hà Nội - 2014

LỜI CẢM ƠN

Nghiên cứu sinh xin được bày tỏ lòng biết ơn tới các thầy hướng dẫn khoa học của mình là PGS. TS. Hồ Thuận và TS. Nguyễn Đại Thọ. Những khích lệ và chỉ dẫn tận tình của các thầy đã giúp nghiên cứu sinh hoàn thành luận án này. Nghiên cứu sinh cũng xin cảm ơn GS. Stefan Funke đã cho nghiên cứu sinh những gợi ý hữu ích ban đầu về lựa chọn đề tài nghiên cứu.

Nghiên cứu sinh xin cảm ơn lãnh đạo Trường Đại học Công nghệ, ĐHQGHN đã tạo môi trường và điều kiện nghiên cứu tốt, hỗ trợ tài chính giúp nghiên cứu sinh tham dự một số hội nghị quốc tế. Đồng thời, nghiên cứu sinh cũng xin được cảm ơn các thầy, cô Bộ môn Mạng và Truyền thông Máy tính, các thầy, cô Khoa Công nghệ Thông tin và Trường Đại học Công nghệ đã hỗ trợ nghiên cứu sinh trong quá trình nghiên cứu và bảo vệ luận án.

LỜI CAM ĐOAN

Tôi xin cam đoan luận án “*Hỗ trợ định vị và nâng cao hiệu năng định tuyến dựa trên thông tin vị trí cho các mạng cảm biến không dây*” là do tôi thực hiện dưới sự hướng dẫn của PGS. TS. Hồ Thuận và TS. Nguyễn Đại Thọ, và không chứa bất kỳ nội dung nào được sao chép từ các công trình đã được người khác công bố. Các tài liệu trích dẫn là trung thực và được chỉ rõ nguồn gốc.

Tôi xin hoàn toàn chịu trách nhiệm về lời cam đoan trên.

Hà Nội, ngày 15 tháng 4 năm 2014.

Lê Đình Thanh

MỤC LỤC

LỜI CẢM ƠN.....	1
LỜI CAM ĐOAN	2
DANH MỤC CÁC THUẬT NGỮ, KÝ HIỆU VÀ TỪ VIẾT TẮT	5
DANH MỤC CÁC BẢNG.....	8
DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ.....	9
CHƯƠNG 1. MỞ ĐẦU.....	11
1.1 Mạng cảm biến không dây	11
1.2 Một vài ứng dụng điển hình của mạng cảm biến không dây	12
1.3 Định tuyến và định vị trong mạng cảm biến không dây	13
1.4 Vấn đề được giải quyết và mục tiêu của luận án.....	16
1.5 Nội dung luận án	19
1.6 Đóng góp của luận án	20
CHƯƠNG 2. TỔNG QUAN VỀ ĐỊNH VỊ VÀ ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ	23
2.1 Định vị	23
2.2 Phát hiện biên	26
2.3 Định tuyến dựa trên thông tin vị trí	28
2.3.1 Dịch vụ thông tin vị trí	30
2.3.2 Chuyển tiếp dựa trên thông tin vị trí.....	32
2.3.3 Cực tiểu địa phương.....	34
2.3.4 Giảm thiểu và tránh cực tiểu địa phương	34
2.3.5 Khôi phục sau cực tiểu địa phương	39
2.4 Thảo luận.....	42
CHƯƠNG 3. HỖ TRỢ ĐỊNH VỊ VỚI PHÁT HIỆN BIÊN DỰA TRÊN KẾT NỐI.....	45
3.1 Tìm biên dựa trên kết nối.....	45
3.1.1 Trực quan và heuristic	45
3.1.2 Thuật toán.....	47
3.1.3 Đáp ứng với thay đổi mạng.....	50
3.2 Phân tích và thử nghiệm	50
3.3 So sánh với các thuật toán hiện có	52
3.4 Thảo luận.....	54
CHƯƠNG 4. TỐI ƯU HÓA ĐƯỜNG ĐI TRONG ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ	56

4.1 Đặt vấn đề	56
4.2 Mô tả giao thức.....	59
4.2.1 Bảng định tuyến	60
4.2.2 Vùng khả áp dụng của phân tử định tuyến.....	61
4.2.3 Chuyển tiếp có chỉ dẫn.....	62
4.2.4 Định tuyến và cập nhật bảng định tuyến.....	63
4.3 Ưu điểm	66
4.4 Phân tích và so sánh với các giao thức khác	68
4.5 Mô phỏng	69
4.5.1 Tỷ lệ kéo dài độ dài đường đi.....	73
4.5.2 Trễ đầu cuối – đầu cuối	75
4.5.3 Tỷ lệ chuyển gói thành công.....	76
4.5.4 Chi phí truyền thông	77
4.5.5 Lựa chọn số chặng được ghi	71
4.6 Thảo luận.....	78
CHƯƠNG 5. ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ SỬ DỤNG CẠNH	
TRANH KẾT HỢP	80
5.1 Mô tả giao thức.....	82
5.1.1 Cạnh tranh kết hợp	82
5.1.2 Vùng cạnh tranh và hàm trễ.....	83
5.1.3 Hành vi của các nút	85
5.2 Phân tích và mô phỏng.....	89
5.2.1 Tỷ lệ chuyển gói tin thành công.....	91
5.2.2 Phụ tải truyền thông.....	91
5.2.3 Độ trễ đầu cuối – đầu cuối.....	92
5.2.4 Tỷ lệ gói tin trùng lặp.....	93
5.3 Thảo luận.....	93
KẾT LUẬN	95
DANH MỤC CÁC CÔNG TRÌNH KHOA HỌC CỦA TÁC GIẢ LIÊN QUAN ĐẾN	
LUẬN ÁN.....	97
TÀI LIỆU THAM KHẢO	98
PHỤ LỤC	108
Phụ lục 1. Ước lượng khoảng cách và góc	108
Phụ lục 2. Cơ sở toán học cho định vị theo khoảng cách	111

DANH MỤC CÁC THUẬT NGỮ, KÝ HIỆU VÀ TỪ VIẾT TẮT

<i>Thuật ngữ tiếng Anh</i>	<i>Viết tắt</i>	<i>Thuật ngữ tiếng Việt tương đương</i>
2-hop Neighbourhood Graph	<i>2NG</i>	Đồ thị vùng lân cận 2 chặng
Aggressive Area	<i>AA</i>	Vùng cạnh tranh quyết liệt
Angulation		Định vị theo góc
Aggressive contention		Cạnh tranh quyết liệt
Applicable area		Vùng khả áp dụng
Beacon Path/Shortcut		Đường tắt
Behavior Based Tagging	<i>BBT</i>	Đánh dấu dựa vào hành vi
Boundary node		Nút biên
Boundary detouring		Đi theo biên
Communication hole		Vùng trống truyền
Compass forwarding		Chuyển tiếp theo góc
Connectivity-based		Dựa trên kết nối
Contention		Cạnh tranh
Distance-based forwarding		Chuyển tiếp theo khoảng cách
Face routing		Định tuyến trên mặt
Geographic Forwarding	<i>GF</i>	Chuyển tiếp dựa trên thông tin vị trí
Geographic routing		Định tuyến dựa trên thông tin vị trí
Global Positioning System	<i>GPS</i>	Hệ thống định vị toàn cầu
Guided forwarding		Chuyển tiếp có chỉ dẫn

Greedy forwarding		Chuyển tiếp tham lam
Greedy with Path Optimization Routing	<i>GPOR</i>	Định tuyến tham lam với tối ưu hóa đường đi
Hole Announcement	<i>HA</i>	Gói tin thông báo vùng trống
Hole Boundary Detection	<i>HBD</i>	Gói tin phát hiện biên vùng trống
Hull tree		Cây bao
Hybrid Contention-Based Geographic Routing	<i>HCGR</i>	Định tuyến dựa trên thông tin vị trí sử dụng cạnh tranh kết hợp
Inertia forwarding		Chuyển tiếp với quán tính
Lateration		Định vị theo khoảng cách
Local minimum		Cực tiểu địa phương
Location-based routing		Định tuyến dựa trên thông tin vị trí
Location service		Dịch vụ thông tin vị trí
Location server		Nút phục vụ vị trí
Multi-dementional Scaling	<i>MDS</i>	Co giãn đa chiều
Most Forwarding progress with Radius	<i>MFR</i>	Bước tiến dài nhất với bán kính
Neighbourhood Based Tagging	<i>NBT</i>	Đánh dấu dựa vào vùng lân cận
Non-aggressive Area	<i>NA</i>	Vùng cạnh tranh không quyết liệt
Non-aggressive contention		Cạnh tranh không quyết liệt
Proactive		Chủ động
Shortcut Creation	<i>SC</i>	Tạo đường tắt
Shortcut creation technique		Kỹ thuật tạo đường tắt
Range-based		Dựa trên khoảng

Random Progress Method	<i>RPM</i>	Phương pháp bước tiến ngẫu nhiên
Reactive		Thụ động
Recovery Routing	<i>RR</i>	Định tuyến khôi phục
Topological awareness	<i>TA</i>	Biết về topo
Topology-based		Dựa trên thông tin về topo mạng
Viewscope		Tầm vực
Wireless Sensor Networks	<i>WSN</i>	Mạng cảm biến không dây

DANH MỤC CÁC BẢNG

Bảng 2.1. So sánh các thuật toán định vị.	26
Bảng 2.2. So sánh các thuật toán phát hiện biên.	28
Bảng 2.3. Hành vi của mỗi nút cảm biến trong định tuyến dựa trên thông tin vị trí.	29
Bảng 2.4. So sánh các kỹ thuật chuyển tiếp dựa trên thông tin vị trí.	34
Bảng 2.5. So sánh các kỹ thuật chuyển tiếp dựa trên thông tin vị trí.	38
Bảng 2.6. So sánh các kỹ thuật khôi phục.	42
Bảng 3.1. Thuật toán phát hiện biên được đề xuất.	47
Bảng 3.2. Độ chính xác và độ hồi tưởng của thuật toán phát hiện biên được đề xuất qua thử nghiệm.	52
Bảng 4.1. Định dạng của các phân tử định tuyến.	61
Bảng 4.2. Hành vi của mỗi nút cảm biến trong GPOR.	65
Bảng 4.3. Cấu hình mạng với kích thước mạng khác nhau.	70
Bảng 4.4. Cấu hình mạng với số luồng lưu lượng khác nhau.	71
Bảng 5.1. Tiêu đề gói tin của HCGR.	86
Bảng 5.2. Giao thức HCGR, mã cho nút C.	86

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

Hình 1.1. Giải pháp được đề xuất.....	22
Hình 2.1. Hành vi của mỗi nút cảm biến trong định tuyến dựa trên thông tin vị trí.	31
Hình 2.2. Kỹ thuật quay.	41
Hình 3.1. Biên và vùng trống trong trường hợp liên tục.	46
Hình 3.2. Biên và vùng trống trong trường hợp rời rạc.	47
Hình 3.3. Minh họa thuật toán kiểm tra khả năng gần biên.	50
Hình 3.4. Một vài kết quả thử nghiệm thuật toán phát hiện biên được đề xuất.....	53
Hình 3.5. Phân hoạch các nút biên.	54
Hình 4.1. Một đoạn đường cong được thay thế bằng một đường tắt.	60
Hình 4.2. Vùng khả áp dụng của phần tử định tuyến.....	62
Hình 4.3. Hành vi của mỗi nút cảm biến trong GPOR.....	63
Hình 4.4. Phần tử định tuyến mới có hướng thoát khỏi vùng lõm trước vùng trống tốt hơn phần tử định tuyến đã có	64
Hình 4.5. Đường đi được tạo bởi gói dữ liệu và các gói SC kèm theo.....	67
Hình 4.6. Tối ưu hóa đường đi bởi các luồng lưu lượng đồng thời..	68
Hình 4.7. Tỷ lệ kéo dài độ dài đường đi với kích thước mạng khác nhau.	74
Hình 4.8. Tỷ lệ kéo dài độ dài đường đi với số luồng lưu lượng đồng thời khác nhau. ..	74
Hình 4.9. Trung bình trễ đầu cuối – đầu cuối với kích thước mạng khác nhau.....	75
Hình 4.10. Trung bình trễ đầu cuối – đầu cuối với số luồng lưu lượng đồng thời khác nhau.....	76
Hình 4.11. Tỷ lệ chuyển gói tin đến đích thành công với kích thước mạng khác nhau. ..	77
Hình 4.12. Tỷ lệ chuyển gói tin đến đích thành công với số lưu lượng đồng thời khác nhau.....	77
Hình 4.13. Tổng số phát tỏa với kích thước mạng khác nhau.	78
Hình 4.14. Tổng số phát tỏa với số luồng lưu lượng đồng thời khác nhau.	78

Hình 4.15. Tỷ lệ kéo dài độ dài đường đi với số nút được ghi khác nhau.....	72
Hình 4.16. Trung bình trễ đầu cuối – đầu cuối với số nút được ghi khác nhau.....	72
Hình 4.17. Tỷ lệ chuyển gói tin đến đích thành công với số nút được ghi khác nhau.	73
Hình 4.18. Tổng số phát tỏa với số nút được ghi khác nhau.	73
Hình 5.1. Các vùng cạnh tranh ở chế độ tham lam.	83
Hình 5.2. Các vùng cạnh tranh ở chế độ khôi phục.....	84
Hình 5.3. Tỷ lệ chuyển gói tin đến đích thành công của HCGR, ACGR và NCGR.....	91
Hình 5.4. Phụ tải truyền thông của HCGR, ACGR và NCGR.....	92
Hình 5.5. Độ trễ đầu cuối – đầu cuối của HCGR, ACGR và NCGR.....	92
Hình 5.6. Tỷ lệ gói tin trùng lặp của HCGR, ACGR và NCGR.	93

CHƯƠNG 1

MỞ ĐẦU

1.1 Mạng cảm biến không dây

Các thiết bị cảm biến (sensors) tạo liên kết giữa thế giới vật lý với thế giới số bằng việc thu nhận các hiện tượng của thế giới vật lý rồi chuyển đổi nó thành dạng có thể lưu trữ và xử lý được. Khi được tích hợp vào các hệ thống khác nhau, các thiết bị cảm biến đem lại nhiều lợi ích cho đời sống. Chúng cho phép nhiều ứng dụng mới như nhà thông minh, robot, hệ thống tự động hóa, ... Những tiến bộ trong phát triển các công nghệ VLSI (tích hợp phạm vi rất lớn), MEMS (hệ thống vi cơ điện tử), và truyền thông không dây đã giúp cho việc sử dụng các hệ thống thiết bị cảm biến phân tán ngày càng trở nên phổ biến. Các công nghệ tính toán cùng các công nghệ cảm biến tiên tiến cho phép sản xuất ra các thiết bị cảm biến có kích thước nhỏ, tiêu ít năng lượng và rẻ tiền. Ngoài ra, các hệ thống nhúng tiếp tục có nhiều ứng dụng trong nhiều lĩnh vực khác nhau. Mạng của hàng ngàn các nút cảm biến đã sẵn sàng được sử dụng để theo dõi các khu vực địa lý rộng lớn cho việc mô hình hóa và dự báo lũ lụt, điều khiển tưới tiêu và sử dụng hóa chất nhằm tăng năng suất mùa màng, giám sát hiện trường, phát hiện đột nhập, theo dõi hoạt động của núi lửa, ... Cùng với yêu cầu thu thập các thông số môi trường, phạm vi triển khai lớn, điều kiện khắc nghiệt, hay hạ tầng truyền thông có dây không sẵn sàng là những động lực dẫn đến nhu cầu sử dụng mạng của nhiều nút cảm biến có thể truyền thông không dây với nhau.

Từ góc độ kỹ thuật, mạng cảm biến không dây (Wireless Sensor Networks - WSN) bao gồm nhiều nút cảm biến, mỗi nút có bộ vi xử lý, bộ nhớ, bộ phận thu/phát tín hiệu không dây, một hoặc nhiều thiết bị cảm biến, nguồn năng lượng (pin) và có thể có cả bộ phận định vị. Nút cảm biến có nhiệm vụ thu nhận các tín hiệu vật lý từ môi trường xung

quanh. Tín hiệu vật lý thu nhận được được lượng hóa bằng bộ chuyển đổi tương tự-số rồi được chuyển vào bộ vi xử lý. Thông thường, mỗi thiết bị cảm biến chỉ đo được một tín hiệu vật lý như nhiệt độ, độ ẩm, áp suất, độ sáng, độ rung chuyển hay nồng độ khí CO₂, v.v. Để đo nhiều tín hiệu vật lý đồng thời, người ta thường tích hợp nhiều thiết bị cảm biến thành một bảng các thiết bị cảm biến. Bộ phận thu/phát tín hiệu không dây có nhiệm vụ điều chế và phát tín hiệu dưới dạng sóng không dây, đồng thời thu và giải điều chế tín hiệu. Các chuẩn công nghệ được sử dụng phổ biến cho WSN bao gồm IEEE 802.15.4 [105], ZigBee [111]. Bộ phận định vị, ví dụ thiết bị định vị GPS, cho biết vị trí (tọa độ) của nút cảm biến. Bộ vi xử lý có năng lực tính toán hạn chế. Tương tự, bộ nhớ có dung lượng cũng hạn chế. Pin có nhiệm vụ cung cấp điện cho nút hoạt động, có kích thước nhỏ và thường không được nạp điện bổ sung. Tất cả các thành phần kể trên cấu thành một máy tính siêu nhỏ với khả năng tính toán và truyền thông dữ liệu¹. Nhiều nút cảm biến được triển khai trên một khu vực tạo thành một *mạng tự hợp* của các nút cảm biến.

1.2 Một vài ứng dụng điển hình của mạng cảm biến không dây

Ứng dụng trong môi trường và nông nghiệp: Mạng cảm biến không dây được ứng dụng nhiều trong lĩnh vực môi trường. Một mạng cảm biến không dây có thể được triển khai để theo dõi và cảnh báo cháy rừng, theo dõi và cảnh báo cháy nổ ở kho bãi, đo hàm lượng khí CO₂ trong một khu vực kiểm định, thu nhập các thông số nhiệt độ, độ ẩm, áp suất không khí, tốc độ gió phục vụ cho dự báo thời tiết. Cũng có thể sử dụng mạng cảm biến không dây để theo dõi sự di cư của các loài chim, các loài động vật, các loài cá trong đại dương. Trong nông nghiệp, mạng cảm biến không dây được sử dụng để thiết kế hệ thống tưới tiêu tự động, theo dõi sự tăng trưởng của cây trồng, hoạt động của các loại côn trùng và thiên địch có tác động đến cây trồng, v.v. Corke và các cộng sự [19] cho chúng ta một cái nhìn khá toàn diện về ứng dụng của mạng cảm biến không dây trong lĩnh vực này.

Ứng dụng trong y tế và chăm sóc sức khỏe: Các thiết bị cảm biến y sinh có thể được cấy vào cơ thể bệnh nhân để theo dõi các cơn đau tim, các trận hen suyễn, ức chế thần

¹ Ngày nay có nhiều hãng cung cấp nền tảng phần cứng và/hoặc phần mềm mạng cảm biến không dây như Memsic [110], Libelium [109]. Wiki cung cấp cho chúng ta một danh sách khá nhiều các nền tảng mạng cảm biến không dây tại http://en.wikipedia.org/wiki/List_of_wireless_sensor_nodes.

kinh, sự phát triển của ung thư, nồng độ gluco trong máu, v.v. Các bác sỹ đặc biệt quan tâm đến ứng dụng của mạng cảm biến không dây để theo dõi các bệnh nhân hậu phẫu, các bệnh nhân cần được theo dõi liên tục. Một ứng dụng khá tiện ích của mạng cảm biến không dây là thực hiện chăm sóc sức khỏe tại nhà. Người tham gia chăm sóc sức khỏe tại nhà được gắn các thiết bị cảm biến lên cơ thể. Các thông số về sức khỏe được thu thập bởi các cảm biến sẽ được gửi đến bác sỹ qua mạng Internet. Bác sỹ, do vậy, có thể biết được tình trạng sức khỏe của các bệnh nhân của mình. Nhiều ứng dụng của mạng cảm biến không dây trong lĩnh vực y tế được mô tả tổng quan trong [74].

Ứng dụng trong quân sự và an ninh: Mạng cảm biến không dây ra đời với mục đích ban đầu phục vụ cho quân sự và an ninh. Chính vì vậy, các ứng dụng của mạng cảm biến không dây trong lĩnh vực này thực sự phong phú. Người quan tâm không khó để tìm thấy tên những ứng dụng này trong các bài báo tổng quan về ứng dụng của mạng cảm biến không dây, ví dụ như [11].

Cùng với sự gia tăng nhu cầu trong đời sống con người, ngày càng có nhiều các ứng dụng sử dụng mạng cảm biến không dây.

1.3 Định tuyến và định vị trong mạng cảm biến không dây

Mạng cảm biến không dây là một dạng của mạng tự hợp. Tuy nhiên, so với các mạng tự hợp khác như MANET [106] hay VANET [107], mạng cảm biến không dây có những đặc tính riêng sau dẫn đến những thuận lợi hoặc thách thức cho việc thiết kế các giao thức định tuyến cho mạng cảm biến không dây:

- *Tài nguyên (năng lực tính toán, bộ nhớ, pin) của mỗi nút hết sức hạn chế:* Đây là thách thức lớn khi thiết kế các giao thức định tuyến cho mạng cảm biến không dây. Những giao thức áp dụng được cho mạng cảm biến không dây phải yêu cầu tính toán cũng như lưu trữ rất ít tại mỗi nút. Đồng thời, những giao thức này phải giải quyết tốt vấn đề cân bằng tải để ít nút phải hoạt động nhiều hơn và sớm ngừng hoạt động do pin hết điện.

- *Số lượng nút được triển khai có thể rất lớn*: Có thể hàng nghìn nút được triển khai trên vùng rộng lớn. Đặc điểm này càng làm tăng yêu cầu tính toán cũng như lưu trữ rất ít tại mỗi nút.
- *Nút thay đổi chế độ thức và ngủ*: Để tiết kiệm điện của pin và kéo dài tuổi thọ các nút, một số nhà sản xuất cung cấp các nút có khả năng đi vào chế độ ngủ khi rỗi hay theo chu kỳ. Đây là một thách thức cho các giao thức định tuyến vì các liên kết giữa các nút hay bị thay đổi.
- *Nút ngừng hoạt động*: Nút có thể ngừng hoạt động do nhiều nguyên nhân như bị chìm trong bùn, nước, bị va đập, bị đốt cháy hay phổ biến nhất là hết điện. Các nút ngừng hoạt động sẽ tạo ra những vùng trống, tại đó không một lưu lượng nào có thể chuyển qua được.
- *Nút được bổ sung*: Các nút mới có thể được bổ sung để lấp các vùng trống do các nút đã ngừng hoạt động để lại hoặc để mở rộng khu vực triển khai.
- *Nút ít di chuyển*: Đây là một thuận lợi cho việc thiết kế các giao thức cho mạng cảm biến không dây.

Tiếp cận truyền thông cho định tuyến trong mạng tự hợp dựa trên thông tin về topo mạng (topology-based). Theo tiếp cận này, các nút khám phá (một phần hay toàn bộ) thông tin về topo mạng bằng việc trao đổi các thông báo điều khiển và sử dụng thông tin này để đưa ra các quyết định định tuyến sau này. Có hai phương pháp thực hiện tiếp cận định tuyến dựa trên topo là *chủ động* (proactive) và *thụ động* (reactive). Cũng có thể kết hợp hai phương pháp này trong cùng một giao thức và chúng ta có phương pháp định tuyến *lai* (hybrid).

Các giao thức định tuyến chủ động tính trước đường đi giữa mỗi cặp nút bất kỳ và lưu thông tin các đường đi đã được tính tại các nút theo một cấu trúc được gọi là bảng định tuyến. Khi có yêu cầu định tuyến, nút hiện tại mang gói tin sẽ tra cứu bảng định tuyến của nó để biết cần chuyển gói tin cho nút nào tiếp theo. Do việc tính trước các đường đi trên thông tin toàn cục, các giao thức định tuyến chủ động có thể cho các đường đi tối ưu.

Các ví dụ về định tuyến chủ động cho mạng tự hợp bao gồm DSDV [78], WRP [71], STAR [77], sử dụng tác tử di động [9], AIR [15], OLSR [40] và TBRPF [76].

Khác với định tuyến chủ động, các giao thức định tuyến thụ động không tính trước tất cả các đường đi mà chỉ tính các đường đi *khi có yêu cầu* (on-demand). Thông tin về các đường đi đã được tính có thể được lưu trong vùng đệm của các nút và chỉ có giá trị sử dụng trong khoảng thời gian nhất định. Khi có yêu cầu định tuyến, nút nguồn nhìn trong vùng đệm xem có đường đi đến nút đích hay không, nếu có, đường đi này sẽ được sử dụng, ngược lại nút nguồn phát động một lượt tìm đường bằng cách phát đi yêu cầu tìm đường. Yêu cầu tìm đường sẽ được phát tỏa trong mạng theo một cách thức nào đó, có thể là phát tràn (flooding), cho đến khi đường đi được tìm thấy. Thông tin về đường đi vừa tìm thấy sẽ được gửi ngược đến nút nguồn, lưu tại vùng đệm của các nút tham gia gửi ngược, và sau đó được sử dụng để định tuyến các gói tin. Các ví dụ về định tuyến thụ động bao gồm DSR [43, 44], AODV [79, 80], LBAR [35], preemptive routing [32], DLAR [57], và NCP-based [16].

Việc sử dụng kết hợp định tuyến chủ động và thụ động cũng đã được đề xuất. Các ví dụ về giao thức kết hợp chủ động và thụ động như ZRP [33], HSLS [85]. Theo cách thức này, mạng được chia nhỏ thành các vùng (zone), định tuyến giữa hai nút trong cùng vùng được thực hiện theo phương pháp chủ động, định tuyến giữa hai nút thuộc các vùng khác nhau được thực hiện theo phương pháp thụ động.

Các phương pháp định tuyến dựa trên thông tin topo yêu cầu các nút phải lưu trữ nhiều thông tin về các đường đi. Yêu cầu này vượt ngoài khả năng đáp ứng của các nút cảm biến. Ngoài ra, định tuyến dựa trên topo sử dụng nhiều gói tin điều khiển để tìm và duy trì các đường đi. Ngoài tác động làm giảm băng thông sẵn có cho dữ liệu, nhiều gói tin điều khiển tiêu hao nhiều điện năng của các nút và hệ quả là làm giảm tuổi thọ của các nút. Với các đặc điểm như phân tích ở trên, định tuyến dựa trên topo hầu như không áp dụng được cho mạng cảm biến không dây.

Những năm gần đây, một tiếp cận hoàn toàn khác cho vấn đề định tuyến cho mạng cảm biến không dây là sử dụng thông tin về vị trí/tọa độ của các nút làm thông tin dẫn

đường². Tiếp cận mới này có tên là *định tuyến dựa trên thông tin vị trí* (location-based hay geographic routing). Định tuyến dựa trên thông tin vị trí giả thiết mỗi nút biết về vị trí của nó bằng việc sử dụng hệ thống định vị như GPS hoặc bằng thuật toán *định vị* (localization) nào đó. Thông tin định tuyến mà mỗi nút phải duy trì chỉ là vị trí của các láng giềng. Thông tin này có thể được cập nhật một cách nhanh chóng và tiết kiệm. Do đó, định tuyến dựa trên thông tin vị trí phù hợp cho mạng cảm biến không dây, đặc biệt là các mạng có quy mô lớn.

Trong trường hợp các nút không được trang bị thiết bị định vị, vì lý do tài chính và hiệu quả sử dụng năng lượng, một thuật toán phân tán có thể được sử dụng để gán tọa độ cho các nút. Thuật toán như vậy được gọi là thuật toán định vị. Thuật toán định vị khai thác thông tin kết nối giữa các nút hoặc thông tin ước lượng khoảng cách hay góc giữa các nút. Tọa độ của các nút được xác định bằng thuật toán định vị giúp cho định tuyến dựa trên thông tin vị trí vẫn có thể được áp dụng trong mạng cảm biến không dây không được trang bị thiết bị định vị tại mỗi nút.

1.4 Vấn đề được giải quyết và mục tiêu của luận án

Định tuyến đơn phát dựa trên thông tin vị trí cho mạng cảm biến không dây đã được quan tâm nghiên cứu từ nhiều năm nay. Tuy nhiên, các giao thức hiện có còn tồn tại những hạn chế về hiệu năng như tìm đường đi dài, đi vòng và mất cân bằng tải, sử dụng gói tin chào hỏi làm tiêu tốn băng thông mạng, ... Do vậy, một trong những mục tiêu chính của luận án này là khắc phục các hạn chế kể trên của các giao thức hiện có. Nói cách khác, vấn đề thứ nhất được quan tâm giải quyết trong luận án này là nâng cao hiệu năng của định tuyến đơn phát dựa trên thông tin vị trí cho mạng cảm biến không dây.

Trong trường hợp các nút cảm biến không được trang bị thiết bị định vị, các thuật toán định vị là cần thiết để xác định tọa độ tương đối cho các nút, từ đó có thể áp dụng định tuyến dựa trên thông tin vị trí. Các thuật toán định vị hiện có dựa trên giải thiết biết các nút biên. Tuy nhiên, các thuật toán phát hiện biên (boundary detection) hiện có hoặc không hiệu quả hoặc chỉ làm việc tốt trên mạng có mật độ nút dày và phân bố đều. Do đó, vấn đề thứ hai được quan tâm giải quyết trong luận án này là phát hiện biên trong

² Tọa độ của các nút không chỉ có ý nghĩa trong định tuyến dựa trên thông tin vị trí mà còn được sử dụng trong nhiều ứng dụng khác nằm ngoài phạm vi quan tâm của luận án này.

mạng cảm biến không dây hiệu quả, có thể làm việc trên cả mạng có mật độ nút thưa và phân bố không đều.

Tóm lại, hai bài toán *định tuyến đơn phát dựa trên thông tin vị trí* và *phát hiện biên* trong mạng cảm biến không dây được quan tâm đồng thời. Bài toán phát hiện biên nhằm đến hỗ trợ cho định vị trong mạng cảm biến không dây. Để xác định rõ bài toán, các giả thiết sau đây được sử dụng:

- Mạng cảm biến không dây bao gồm nhiều nút được *phân bố ngẫu nhiên* trên một vùng triển khai được xem là *phẳng*. Số lượng nút được triển khai lớn và vùng triển khai rộng. Mạng được gọi là mạng 2D do vùng triển khai là một khu vực phẳng. Các kịch bản thực tế cho giả thiết này là các mạng cảm biến không dây được triển khai để theo dõi cháy rừng, để điều khiển tưới tiêu tự động, hay để theo dõi đột nhập của quân địch trên một bãi chiến trường, ...
- Các nút phát *sóng radio đẳng hướng*. Các liên kết được giả thiết là *đối xứng*. Kịch bản thực tế cho giả thiết này là sử dụng các nút cùng loại với anten đẳng hướng. Chúng ta cũng giả thiết rằng khoảng cách giữa các nút quyết định liệu có tồn tại liên kết giữa chúng hay không; dưới một ngưỡng khoảng cách nào đó, hai nút sẽ nằm trong vùng phủ sóng của nhau.
- Các nút *ít hoặc không di chuyển*. Mạng được xem là tĩnh. Giả thiết này hoàn toàn hợp lý vì hầu hết các mạng cảm biến được triển khai trong thực tế đều là các mạng tĩnh.

Phát biểu không hình thức của các bài toán như sau:

- *Hỗ trợ định vị với phát hiện biên*: Cho một mạng cảm biến không dây trong đó các nút cảm biến không có thông tin vị trí của chúng, hãy xác định những nút biên của mạng. Thông tin về các nút biên của mạng được sử dụng cho các thuật toán định vị.
- *Định tuyến đơn phát dựa trên thông tin vị trí*: Cho một nút nguồn bất kỳ và một nút đích, hãy tìm đường đi ngắn nhất hoặc gần ngắn nhất từ nút nguồn đến nút đích dựa trên thông tin vị trí của các nút.

Cụ thể, các vấn đề sau đây thuộc hai bài toán nêu trên được quan tâm giải quyết:

- **Hỗ trợ định vị với phát hiện biên dựa trên kết nối:** Nhiều thuật toán định vị đã được đề xuất, trong đó các thuật toán khả thi [27, 56, 81, 86] cho mạng cảm biến không dây phụ thuộc vào độ chính xác và hiệu quả của thuật toán phát hiện biên. Tuy nhiên, các thuật toán phát hiện biên đã có [5, 21, 22, 29, 53, 91] hoặc yêu cầu mạng có phân bố các nút đều và dày hoặc có chi phí truyền thông và tính toán cao do phải giải quyết các bài toán phức tạp là lựa chọn điểm mốc và phát tràn. Do vậy, *mục tiêu thứ nhất* của luận án là đề xuất một thuật toán phát hiện biên phục vụ cho định vị có chi phí truyền thông và tính toán thấp, có thể làm việc trên cả các mạng cảm biến có mật độ nút thưa và phân bố không đều.
- **Nâng cao hiệu năng định tuyến đơn phát dựa trên thông tin vị trí với tối ưu hóa đường đi:** Định tuyến dựa trên thông tin vị trí là tiếp cận tốt cho mạng cảm biến không dây do điều kiện hạn chế về tài nguyên của các nút mạng. Trong nhiều giao thức đã được đề xuất, định tuyến dựa trên thông tin vị trí kết hợp chuyển tiếp tham lam [24] và kỹ thuật khôi phục đi theo biên [20] là giải pháp hiệu quả và khả thi. Tuy nhiên, định tuyến theo phương pháp này có hai yếu điểm chính. Thứ nhất, các đường đi dọc theo biên thường dài và không tối ưu. Thứ hai, nhiều đường đi dọc theo biên dẫn đến lưu lượng quá tải cho các nút biên. Điều này không chỉ dẫn đến tắc nghẽn tại biên khi có nhiều luồng lưu lượng đồng thời mà còn làm giảm nhanh tuổi thọ của các nút biên dẫn đến khoét rộng hơn các vùng trống. Nhằm khắc phục các yếu điểm trên, nhiều kỹ thuật tối ưu hóa đường đi đã được đưa ra [51, 68, 69], trong đó *tạo và sử dụng đường tắt* [51] là kỹ thuật có chi phí thấp, có thể áp dụng với nhiều giao thức định tuyến dựa trên thông tin vị trí. Tuy nhiên, kỹ thuật tạo đường tắt đã có có thể gặp thất bại trong việc tạo đường tắt, cần nhiều thời gian để xây dựng thành công một đường tắt dẫn đến nhiều gói tin không được hưởng lợi từ đường tắt. Ngoài ra, kỹ thuật tối ưu hóa đường đi đã có chỉ được nghiên cứu cho kịch bản có một nút đích. Xuất phát từ thực tế này, *mục tiêu thứ hai* được thực hiện trong luận án là đề xuất một giao thức tối ưu hóa đường đi có thể tạo nhanh và khai thác hiệu quả các đường tắt, có thể áp dụng cho kịch bản có nhiều nút đích, và do vậy có thể áp dụng để nâng cao hiệu năng của định tuyến đơn phát dựa trên thông tin vị trí.

- **Nâng cao hiệu năng định tuyến đơn phát dựa trên thông tin vị trí sử dụng cạnh tranh kết hợp:** Các gói tin chào hỏi được sử dụng trong các giao thức định tuyến dựa trên thông tin vị trí nhằm duy trì thông tin về vị trí của các nút láng giềng. Những gói tin này không chỉ chiếm dụng nhiều băng thông mạng mà còn tiêu thụ nhiều năng lượng, do vậy làm giảm tuổi thọ của các nút. Gần đây, nhiều giao thức định tuyến không sử dụng gói tin chào hỏi đã được đề xuất. Các giao thức này sử dụng cơ chế cạnh tranh để lựa chọn nút chuyển tiếp tiếp theo. Hai hình thức cạnh tranh đã được đề xuất là cạnh tranh quyết liệt) [13] và không quyết liệt [12]. Mỗi hình thức cạnh tranh có những ưu điểm và hạn chế riêng. Ưu điểm chính của hình thức cạnh tranh quyết liệt là nó cho thời gian trễ thấp. Ngoài ra, hình thức cạnh tranh này không sử dụng bất kỳ gói tin điều khiển nào, do vậy tiết kiệm được băng thông mạng cũng như năng lượng của các nút. Tuy nhiên, hạn chế của hình thức cạnh tranh quyết liệt bao gồm việc tạo ra các gói tin trùng lặp. Khác với cạnh tranh quyết liệt, cạnh tranh không quyết liệt không tạo ra gói tin trùng lặp. Tuy nhiên, cạnh tranh không quyết liệt sử dụng các gói tin điều khiển và có trễ lớn hơn. Do vậy, *mục tiêu thứ ba* của luận án là sử dụng đồng thời cả hai hình thức cạnh tranh, được gọi là cạnh tranh kết hợp, nhằm kế thừa các ưu điểm của cả hai hình thức cạnh tranh quyết liệt và không quyết liệt.

1.5 Nội dung luận án

Luận án được trình bày trong năm chương với nội dung từng chương như sau:

- *Chương 1* đặt vấn đề, phát biểu bài toán và mục tiêu của luận án, tóm tắt nội dung và những đóng góp chính của luận án.
- *Chương 2* trình bày kết quả nghiên cứu tổng quan về định vị và định tuyến đơn phát dựa trên thông tin vị trí. Các bài toán định vị, phát hiện biên, và định tuyến đơn phát dựa trên thông tin vị trí được phát biểu. Các công trình liên quan đến các bài toán nói trên được khảo sát và trình bày tóm tắt. Chương 2 kết thúc với một số nhận xét về những giải pháp đã có.
- *Chương 3* trình bày một thuật toán phát hiện biên dựa trên kết nối được đề xuất. Mục đích thiết kế, nội dung thuật toán, đánh giá và so sánh hiệu năng của thuật

toán được đề xuất với những thuật toán đã có, cùng với hướng phát triển lần lượt được trình bày.

- *Chương 4* trình bày một giao thức được đề xuất nhằm tối ưu hóa đường đi trong định tuyến đơn phát dựa trên thông tin vị trí. Kỹ thuật tối ưu hóa đường đi giúp nâng cao hiệu năng của giao thức định tuyến đơn phát một cách rõ rệt và được kiểm chứng qua mô phỏng. Mục đích thiết kế giao thức, nội dung giao thức, đánh giá hiệu năng của giao thức, hướng nghiên cứu cải tiến giao thức lần lượt được trình bày.
- *Chương 5* trình bày một giao thức khác được đề xuất nhằm nâng cao hiệu năng định tuyến đơn phát dựa trên thông tin vị trí bằng việc sử dụng cạnh tranh kết hợp không sử dụng gói tin chào hỏi. Mục đích thiết kế giao thức, nội dung giao thức, đánh giá hiệu năng của giao thức, hướng nghiên cứu cải tiến giao thức lần lượt được trình bày.
- Luận án kết thúc với phần kết luận tổng kết các kết quả chính của luận án và giới thiệu một số hướng nghiên cứu mở rộng tiếp theo.

Ngoài nội dung chính được trình bày trong các chương kể trên, những nội dung tham khảo hoặc mở rộng được trình bày trong hai phụ lục:

- *Phụ lục 1* trình bày các phương pháp ước lượng khoảng cách và ước lượng góc được áp dụng trong định vị.
- *Phụ lục 2* trình bày cơ sở toán học của phương pháp định vị theo đa khoảng cách.

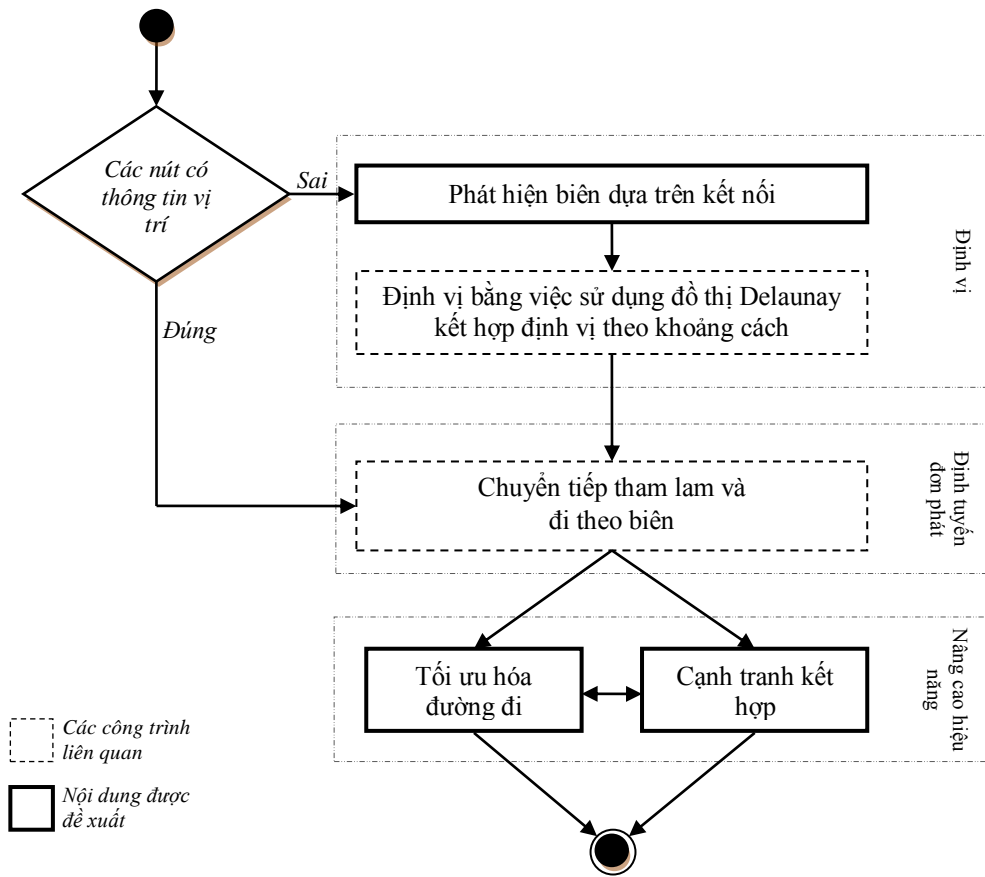
1.6 Đóng góp của luận án

Những đóng góp chính của luận án bao gồm:

- Đề xuất một thuật toán phát hiện biên dựa trên kết nối có độ phức tạp tính toán và truyền thông thấp, có thể làm việc tốt trên cả các mạng cảm biến có mật độ thấp. Theo thuật toán này, mỗi nút đánh giá đồ thị vùng lân cận 2 chặng của nó để quyết định nó có nằm gần biên hay không. Một nút nằm gần biên khi và chỉ khi đồ thị vùng lân cận 2 chặng của nó không tạo thành một cái vành. Việc xây dựng và đánh giá đồ thị vùng lân cận 2 chặng đơn giản và ít tốn kém.

- Đề xuất một giao thức tối ưu hóa đường đi có tên Greedy with Path Optimization Routing (GPOR) cho mạng cảm biến không dây. Theo giao thức này, các đường đi ban đầu được tìm bằng việc áp dụng chuyển tiếp tham lam và kỹ thuật đi theo biên, tiếp đó các đường tắt được tạo và sử dụng nhằm rút ngắn các đường đi, đồng thời tránh cực tiểu địa phương. Các đường đi được rút ngắn và đẩy ra xa biên, do vậy giảm tải cho các nút biên và đạt cân bằng tải tốt hơn. Các phần tử định tuyến có thể áp dụng cho một vùng đích thay vì chỉ một nút đích như các giao thức đã có.
- Đề xuất một giao thức định tuyến dựa trên thông tin vị trí không sử dụng gói tin chào hỏi có tên là Hybrid Contention-Based Geographic Routing (HCGR) sử dụng đồng thời hai hình thức cạnh tranh, được gọi là cạnh tranh kết hợp. Cạnh tranh quyết liệt được sử dụng trước. Nếu cạnh tranh quyết liệt thành công, cạnh tranh không quyết liệt sẽ bị hủy bỏ. Ngược lại, tức cạnh tranh quyết liệt thất bại, cạnh tranh không quyết liệt sẽ được dùng để chuyển tiếp gói tin. Do đó, HCGR có thể làm cực đại tỉ lệ chuyển gói thành công trong khi giữ được độ phức tạp thông báo và trễ ở mức tương đối thấp. Ngoài cạnh tranh kết hợp, một biến thể của kỹ thuật đi theo biên sử dụng cạnh tranh kết hợp cũng được đề xuất.

Hình 1.1 thể hiện trực quan về các bài toán được quan tâm cùng những đề xuất đã được đưa ra nhằm giải quyết các bài toán này.



Hình 1.1. Giải pháp được đề xuất.

CHƯƠNG 2

TỔNG QUAN VỀ ĐỊNH VỊ VÀ ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ

Chương này trình bày tổng quan các vấn đề định vị và định tuyến đơn phát dựa trên thông tin vị trí trong mạng cảm biến không dây, tóm tắt nội dung và đánh giá các thuật toán và giao thức đã có nhằm làm cơ sở cho các đề xuất được trình bày ở các chương tiếp theo. Những nội dung sau đây được trình bày trong chương này:

- Vấn đề định vị.
- Tổng quan về các thuật toán định vị.
- Vấn đề phát hiện biên.
- Tổng quan về các thuật toán phát hiện biên.
- Tổng quan về định tuyến đơn phát dựa trên thông tin vị trí.
- Nhận xét về các thuật toán đã có.

2.1 Định vị

Trong định tuyến dựa trên thông tin vị trí, vị trí hay tọa độ của các nút là thông tin thiết yếu. Thông tin này có thể nhận được bằng việc sử dụng các hệ thống định vị như GPS. Tuy nhiên, việc sử dụng các hệ thống định vị sẽ tốn kém về mặt kinh tế. Ngoài ra, các thiết bị định vị sẽ tiêu tốn nhiều năng lượng dẫn đến làm giảm tuổi thọ của các nút cảm biến. Trong một số môi trường, ví dụ trong nhà, các thiết bị định vị không phát huy tác dụng. Một vấn đề nữa là sự thiếu thuận tiện trong việc trang bị thiết bị định vị cho các nút cảm biến có kích thước nhỏ. Với những hạn chế kể trên của hệ thống định vị, một

phương pháp khác kinh tế và hiệu quả hơn được áp dụng để xác định tọa độ của các nút là sử dụng các thuật toán định vị. Không hoặc một số ít các nút, còn gọi là điểm neo, được trang bị thiết bị định vị và do đó biết tọa độ của chúng, tọa độ của những nút còn lại sẽ được tính toán bằng thuật toán định vị.

Các thuật toán định vị đã được đề xuất được chia thành hai lớp chính là *dựa trên khoảng* (range-based) và *dựa trên kết nối* (connectivity-based), tùy theo thông tin về ước lượng khoảng hay thông tin về kết nối giữa các nút được sử dụng để xác định tọa độ của các nút. Có thể tiếp tục chia định vị dựa trên khoảng thành hai lớp nhỏ hơn là *định vị theo khoảng cách* (lateration) và *định vị theo góc* (angulation) tùy theo thông tin về khoảng cách hay thông tin về góc được sử dụng³.

Với định vị theo khoảng cách [61, 97], trường hợp đơn giản nhất là một nút có thông tin chính xác về khoảng cách từ nó đến ba điểm neo không thẳng hàng. Sử dụng các khoảng cách và vị trí các điểm neo, vị trí của một nút là ở giao điểm của ba đường tròn có tâm ở ba điểm neo. Vấn đề là trong thực tế các ước lượng khoảng cách thường không chính xác dẫn đến ba đường tròn không giao nhau tại một điểm. Để khắc phục điều này, ước lượng khoảng cách đến nhiều hơn ba điểm neo được sử dụng, dẫn đến bài toán *định vị theo đa khoảng cách* (multilateration)⁴.

Định vị theo góc [72] khai thác tính chất nếu biết hai đỉnh, độ dài hai cạnh hoặc độ lớn hai góc của một tam giác thì sẽ biết được vị trí của điểm thứ ba là giao điểm của hai cạnh còn lại. Ước lượng không chính xác cũng có thể xảy ra và có thể khắc phục bằng đa ước lượng.

Để có thông tin ước lượng khoảng ngoại trừ RSSI [61], các phần cứng phụ trợ được yêu cầu. Các phần cứng phụ trợ này dẫn đến các hạn chế như thiết bị định vị. Hơn nữa, độ chính xác trong ước lượng do các thiết bị mang lại có thể bị ảnh hưởng do tác động của môi trường [67]. Ngoài ra, để thông tin ước lượng khoảng có ích, phải có một số lượng đủ lớn các điểm neo được phân bố đồng đều. Nhìn chung, định vị dựa trên khoảng

³ Ước lượng khoảng cách và góc được trình bày trong Phụ lục 1.

⁴ Cơ sở toán học cho định vị theo khoảng cách được trình bày trong Phụ lục 2.

có thể áp dụng cho các hệ thống khác nhưng không phù hợp cho mạng cảm biến không dây.

Khác với định vị dựa trên khoảng, định vị dựa trên kết nối chỉ khai thác thông tin kết nối giữa các nút và không cần các điểm neo. Do vậy, định vị dựa trên kết nối hiệu quả về mặt kinh tế và không bị ảnh hưởng bởi nhiễu [36]. Việc sử dụng các điểm neo giúp cung cấp tọa độ tuyệt đối, tức tọa độ đúng với mọi đối tượng và được đặt trong một khung tham chiếu như GPS. Tuy nhiên, với định vị dựa trên thông tin vị trí, một hệ tọa độ tương đối – trong đó các nút có tọa độ tương đối đối chiếu với các nút khác - là đủ. Từ những phân tích ở trên, luận án này chỉ tập trung vào định vị dựa trên kết nối không sử dụng điểm neo.

Định vị dựa trên kết nối không sử dụng điểm neo đã được quan tâm giải quyết từ nhiều năm nay. Shang và các cộng sự sử dụng cơ giãn đa chiều (MDS) [86]. Dựa trên thông tin kết nối giữa các nút, một thuật toán đường đi ngắn nhất giữa mọi cặp đỉnh được sử dụng để tính ma trận khoảng cách của đồ thị mạng. Ma trận này được dùng làm đầu vào cho MDS, MDS sẽ tính vị trí các nút trong không gian đa chiều rồi chiếu vào không gian hai chiều. Một cách trực quan, MDS cố gắng kéo giãn mạng theo mọi hướng. Nếu khoảng cách giữa các nút được biết chính xác, MDS sẽ cho kết quả định vị chính xác. Tuy nhiên, vì khoảng cách giữa các nút chỉ được ước lượng thô bằng số chặng trên đường đi ngắn nhất giữa chúng và ước lượng thô có thể bị sai lệch rất nhiều so với khoảng cách thực tế do xuất hiện các vùng trống trong mạng, MDS có thể cho kết quả không tốt. Ngoài ra, MDS là thuật toán tập trung nên không thể chạy được trên các nút cảm biến với tài nguyên hạn chế.

Trong phương pháp mạng cao su Rubberband [27, 81], đầu tiên các nút biên được cố định vào một hình chữ nhật. Tiếp theo mỗi nút không thuộc biên v lặp đi lặp lại cập nhật tọa độ của nó (x_v, y_v) là trung bình tọa độ của các láng giềng. Nếu biên ngoài của mạng cảm biến không được phát hiện đúng, hình dạng của mạng có thể bị biến dạng nghiêm trọng. Ngoài ra, độ phức tạp thông báo và độ phức tạp tính toán của thuật toán này cao do mỗi nút phải lặp lại nhiều lần gửi thông tin vị trí cho các láng giềng và tính lại vị trí của mình.

Gần đây Lederer và các cộng sự đề xuất một thuật toán sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách [56]. Đầu tiên, một số nút biên được chọn làm các điểm mốc (landmarks). Tiếp theo, biểu đồ Voronoi và đồ thị Delaunay được xây dựng dựa trên các điểm mốc này. Dựa trên đồ thị Delaunay, các điểm mốc được định vị trước. Khi các điểm mốc đã được định vị, các nút còn lại có thể được định vị một cách dễ dàng bằng định vị theo khoảng cách với khoảng cách đến các điểm mốc được ước lượng bằng số chẵn. Thuật toán định vị này đã được đánh giá cho kết quả chính xác và khả thi cho các mạng cảm biến không dây [56]. Nhược điểm của thuật toán này là có độ phức tạp tính toán và thông báo cao. Bảng 2.1 tóm tắt những điểm khác nhau chính giữa các thuật toán định vị được khảo sát.

Bảng 2.1. So sánh các thuật toán định vị.

<i>Nhóm</i>	<i>Thuật toán định vị</i>	<i>Phân tán</i>	<i>Yêu cầu có các điểm neo</i>	<i>Yêu cầu phần cứng phụ trợ</i>	<i>Yêu cầu biết nút biên</i>	<i>Khả năng biến dạng</i>	<i>Độ phức tạp tính toán</i>	<i>Độ phức tạp thông báo</i>
Dựa trên khoảng	Định vị theo khoảng cách [61, 97]	✓	✓	✓		Thấp	Thấp	Thấp
	Định vị theo góc [72]	✓	✓	✓		Thấp	Thấp	Thấp
Dựa trên kết nối	Co giãn đa chiều [86]				✓	Cao	Cao	Cao
	Màng cao su [27, 81]	✓			✓	Cao	Cao	Cao
	Sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách [56]	✓			✓	Thấp	Cao	Cao

2.2 Phát hiện biên

Vùng trống truyền thông (communication hole) là vùng không được phủ sóng bởi các nút cảm biến. Nút biên (boundary node) là nút nằm cạnh vùng trống truyền thông, tức có vùng phủ sóng tiếp giáp với vùng trống truyền thông. Phát hiện biên, hay phát hiện các nút biên, là một bài toán có nhiều ứng dụng như phát hiện đối tượng đột nhập, khoanh

vùng khu vực có bất thường. Trong luận án này, ứng dụng của phát hiện biên trong định vị dựa trên kết nối được quan tâm: Phát hiện biên là công đoạn đầu trong các thuật toán định vị dựa trên kết nối đã được trình bày ở trên. Do vậy, luận án tập trung vào phát hiện biên dựa trên kết nối. Các phương pháp phát hiện biên dựa trên kết nối đã có được chia thành hai lớp là *thống kê* và *dựa trên kết nối*.

Trong phương pháp thống kê, dữ liệu thống kê được sử dụng nhiều nhất trong phát hiện biên là bậc của nút. Fekete và các cộng sự [21] đã đề xuất một kỹ thuật phát hiện biên sử dụng các ngưỡng để phân biệt bậc của các nút biên và bậc của các nút trong. Một nút được xem là ở biên nếu bậc của nó nhỏ hơn một ngưỡng cho trước. Để xử lý các hình dạng khác nhau của biên, các ngưỡng được chọn một cách động. Bi và các cộng sự [5] đề xuất một phương pháp khác phát hiện các nút biên bằng việc so sánh tương phản bậc của một nút với bậc của các nút cách xa hai chặng. Thông thường, bậc của nút biên nhỏ hơn bậc của các nút cách nó hai chặng. Một dữ liệu thống kê khác đã được sử dụng để phát hiện biên là mức trung tâm (centrality) của nút, hay số đường đi ngắn nhất với độ dài giới hạn đi qua nút [22]. Tương tự bậc của nút, mức trung tâm của các nút trong thường cao hơn mức trung tâm của các nút biên. Để có độ chính xác cao, các phương pháp thống kê yêu cầu các nút được phân bố đều với mật độ tương đối cao. Yêu cầu này làm cho các phương pháp thống kê ít được sử dụng trong thực tế.

Khác với phương pháp thống kê, phát hiện biên dựa trên kết nối khai thác thông tin topo của mạng. Ghrist và các cộng sự [29] đã đề xuất một thuật toán tập trung phát hiện biên sử dụng khái niệm tính tương đồng. Kroller cùng các cộng sự đề xuất một thuật toán khác khai thác các cấu trúc tổ hợp được gọi là hoa và chu kỳ gia tăng [53]. Thuật toán này có chi phí khá cao. Wang và các cộng sự phát triển một thuật toán khác [91] cho phát hiện biên dựa trên topo. Thuật toán này xây dựng cây đường đi ngắn nhất và xác định các nút liền kề có nút tổ tiên chung bé nhất ở xa. Hiện tượng hai nút liền kề nhưng có tổ tiên chung bé nhất ở xa chứng tỏ tổ tiên chung bé nhất của chúng nằm ở biên của vùng trống và hai đường đi đến hai nút liền kề dọc theo biên theo hai hướng khác nhau. Thuật toán này không hiệu quả vì hai nút liền kề bất kỳ phải xác định tổ tiên chung bé nhất. Như đã được chỉ ra trong [26], các thuật toán trong [29, 53, 91] chỉ áp dụng cho các mạng có phân bố đều và dày. Funke sử dụng khái niệm đường đồng mức để phát hiện các nút gần

biên [26]. Vấn đề của thuật toán này là việc lựa chọn các điểm mốc có độ phức tạp như thuật toán bầu thủ lĩnh [3]. Với bốn điểm mốc toàn cục và nhiều điểm mốc cục bộ trên các đường đồng mức, thời gian để lựa chọn các điểm mốc là rất lớn. Ngoài ra, thuật toán thực hiện nhiều phát tràn dẫn đến chi phí cao cả về thông báo và thời gian.

Bảng 2.2. tóm tắt những điểm khác nhau chính giữa các thuật toán phát hiện biên được khảo sát.

Bảng 2.2. So sánh các thuật toán phát hiện biên.

<i>Nhóm</i>	<i>Thuật toán phát hiện biên</i>	<i>Phân tán</i>	<i>Yêu cầu mạng có phân bố đều và dày</i>	<i>Độ phức tạp tính toán</i>	<i>Độ phức tạp thông báo</i>
Thống kê	Phân ngưỡng bậc [21]	✓	✓	Thấp	Thấp
	So sánh tương phản bậc [5]	✓	✓	Thấp	Thấp
	Phân ngưỡng mức trung tâm [22]	✓	✓	Trung bình	Trung bình
Dựa trên topo	Tính tương đồng [29]		✓	Trung bình	Trung bình
	Hoa và chu kỳ gia tăng [53]	✓	✓	Trung bình	Trung bình
	Tổ tiên chung bé nhất [91]	✓	✓	Cao	Cao
	Đường đồng mức [26]	✓		Cao	Cao

2.3 Định tuyến dựa trên thông tin vị trí

Các phương pháp định tuyến dựa trên thông tin topo yêu cầu các nút phải lưu trữ nhiều thông tin về các đường đi. Yêu cầu này vượt ngoài khả năng đáp ứng của các nút cảm biến. Ngoài ra, định tuyến dựa trên topo sử dụng nhiều gói tin điều khiển để tìm và duy trì các đường đi. Ngoài tác động làm giảm băng thông sẵn có cho dữ liệu, nhiều gói tin điều khiển tiêu hao nhiều điện năng của các nút và hệ quả là làm giảm tuổi thọ của các nút. Phương pháp định tuyến thụ động còn có nhược điểm nữa là có thời gian trễ lớn và cho ra các đường đi không tối ưu. Với các đặc điểm như phân tích ở trên, định tuyến dựa trên topo hầu như không áp dụng được cho mạng cảm biến không dây.

Những năm gần đây, một tiếp cận hoàn toàn khác cho vấn đề định tuyến cho mạng cảm biến không dây là sử dụng thông tin về vị trí của các nút làm thông tin dẫn đường.

Tiếp cận mới này có tên là định tuyến *dựa trên thông tin vị trí* (location-based hay geographic routing). Định tuyến dựa trên thông tin vị trí giả thiết mỗi nút biết về vị trí của nó bằng việc sử dụng hệ thống định vị như GPS hoặc bằng kỹ thuật định vị nào đó. Ngoài ra, định tuyến cần sử dụng một giao thức khác, được gọi là *dịch vụ thông tin vị trí* (location service) (xem Mục 2.3.1), để xác định vị trí của nút đích. Trước khi thực hiện giao thức định tuyến, nút nguồn xác định vị trí của nút đích thông qua triệu gọi dịch vụ thông tin vị trí. Thông tin về vị trí nút đích, sau đó, được gắn vào tiêu đề mỗi gói tin cần chuyển đi và được sử dụng làm thông tin dẫn đường. Các nút không phải duy trì các đường đi cũng như các trạng thái định tuyến.

Bảng 2.3. Hành vi của mỗi nút cảm biến trong định tuyến dựa trên thông tin vị trí.

PROCEDURE Xử lý gói tin

1: Nếu tôi là nút đích của gói tin,

1.1: Chuyển gói tin lên tầng trên (giao vận)

2: Ngược lại, tôi không phải là nút đích của gói tin,

2.1: Nếu gói tin đang ở chế độ tham lam,

2.1.1: Sử dụng chuyển tiếp dựa trên thông tin vị trí (GF) để chuyển gói tin

2.1.1.1: Nếu chọn được láng giềng để chuyển gói tin theo GF,

2.1.1.1.1: Chuyển gói tin cho láng giềng được chọn

2.1.1.2: Nếu không chọn được láng giềng để chuyển gói tin theo GF

2.1.1.2.1: Ghi vị trí của tôi vào tiêu đề gói tin như là cực tiểu địa phương cuối

2.1.1.2.2: Đặt gói tin về chế độ khôi phục rồi thực hiện bước 2.2.2.1

2.2: Ngược lại, gói tin đang ở chế độ khôi phục,

2.2.1: Nếu tôi gần đích hơn cực tiểu địa phương cuối cùng mà gói tin đi qua

2.2.1.1: Chuyển gói tin về chế độ tham lam rồi quay lại bước 2.1.1.

2.2.2: Ngược lại, không thể đưa gói tin về chế độ tham lam,

2.2.2.1: Áp dụng định tuyến khôi phục để chuyển gói tin. Nếu áp dụng định tuyến khôi phục không thành công thì loại bỏ gói tin.

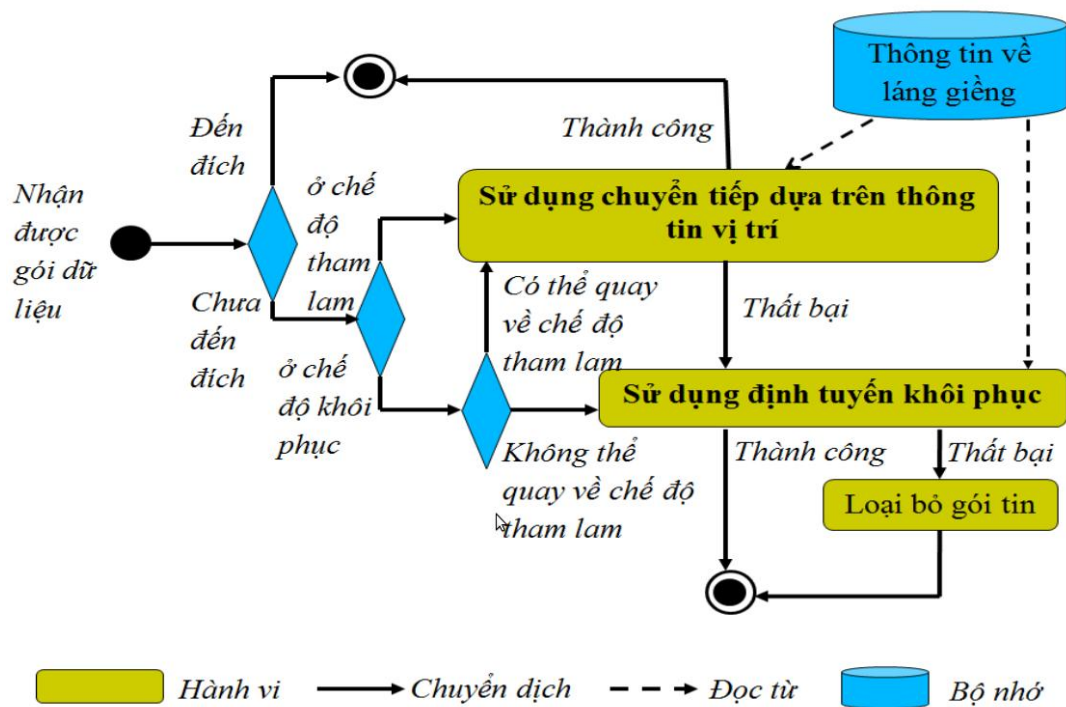
Định tuyến dựa trên thông tin vị trí sử dụng kết hợp *chuyển tiếp dựa trên thông tin vị trí* (geographic forwarding - GF) và *định tuyến khôi phục* (recovery routing - RR) để định tuyến gói tin. Chuyển tiếp dựa trên thông tin vị trí là kỹ thuật chuyển gói tin từ nút này đến nút khác *gần* đích hơn. Độ gần đích của một nút có thể đo bằng khoảng cách từ nút đó đến nút đích hoặc cũng có thể đo dựa trên những tiêu chí khác như sẽ được trình bày trong phần tiếp theo. Nút không có láng giềng gần đích hơn được gọi là *cực tiểu địa phương* (local minimum). Cực tiểu địa phương xuất hiện ở biên của những *vùng trống* (hole), hay vùng không có nút cảm biến, và biên của mạng. Để mỗi nút nhận được gói tin biết nên sử dụng chuyển tiếp dựa trên thông tin vị trí hay định tuyến khôi phục cho gói tin nhận được, một thông tin chỉ dẫn được gọi là *chế độ* (mode) định tuyến được ghi trong tiêu đề của gói tin. Chế độ định tuyến có thể là *tham lam* (greedy) hoặc *khôi phục* (recovery). Nút nguồn thiết lập chế độ tham lam, hay chế độ sử dụng chuyển tiếp dựa trên thông tin vị trí, cho gói tin. Khi nhận được gói tin ở chế độ tham lam, cực tiểu địa phương sẽ thay đổi gói tin sang chế độ khôi phục, hay chế độ sử dụng định tuyến khôi phục, đồng thời ghi thông tin về vị trí của nó vào tiêu đề gói tin ở trường cực tiểu địa phương gặp cuối cùng. Khi nhận được gói tin ở chế độ khôi phục, nút *gần* đích hơn cực tiểu địa phương gặp cuối cùng khôi phục gói tin về chế độ tham lam. Ở những tình huống còn lại, nút nhận được gói tin không thay đổi chế độ định tuyến của gói tin mà chỉ chuyển tiếp gói tin bằng việc áp dụng chuyển tiếp dựa trên thông tin vị trí hay định tuyến khôi phục tùy theo chế độ định tuyến hiện tại của gói tin. Hình 2.1 minh họa hành vi của các nút dưới dạng sơ đồ khối với mô tả bằng lời được trình bày trong Bảng 2.3.

Dịch vụ thông tin vị trí, chuyển tiếp dựa trên thông tin vị trí cùng một số kỹ thuật khôi phục hay được sử dụng được trình bày tổng quan trong các mục nhỏ tiếp sau.

2.3.1 Dịch vụ thông tin vị trí

Dịch vụ thông tin vị trí được sử dụng như dịch vụ nền có chức năng cung cấp thông tin về vị trí của các nút cho các giao thức khác. Bởi vậy, dịch vụ thông tin vị trí thường được nghiên cứu như một bài toán độc lập. Một dịch vụ thông tin vị trí bao gồm hai thành phần thực hiện hai chức năng khác nhau là cập nhật vị trí và yêu cầu vị trí. Thành phần cập nhật vị trí thực hiện phát tán thông tin vị trí hiện tại của mỗi nút T đến một số nút khác trên mạng. Mỗi nút có nhiệm vụ tiếp nhận và duy trì thông tin về vị trí của nút T

được gọi là một nút *phục vụ vị trí* (location server) của T . Mỗi khi có nhu cầu thông tin về vị trí của T , S phát đi một yêu cầu vị trí. Yêu cầu này được định tuyến đến một trong các nút phục vụ vị trí của T , nút phục vụ vị trí của T , sau đó, sẽ trả lời S với thông tin vị trí của T do nó nắm bắt được. Trong trường hợp tất cả các nút đều là nút phục vụ vị trí của T , thành phần yêu cầu vị trí không phải làm gì nhưng thành phần cập nhật phải thực hiện phát tràn thông tin vị trí của T đến tất cả các nút trong mạng. Dịch vụ thông tin vị trí được sử dụng bởi DREAM [4] nằm ở thái cực này. Ở một thái cực khác, trường hợp chỉ mình T là nút phục vụ thông tin vị trí của nó, thành phần cập nhật thông tin vị trí không làm gì nhưng thành phần yêu cầu vị trí sẽ phải phát tràn yêu cầu trong mạng cho đến khi yêu cầu đến được nút T . RLS [48] là một đơn cử cho dịch vụ thông tin vị trí ở thái cực thứ hai.



Hình 2.1. Hành vi của mỗi nút cảm biến trong định tuyến dựa trên thông tin vị trí.

Một dịch vụ thông tin vị trí hiệu quả phải thích ứng nhanh với thay đổi topo và tránh được các phát tràn bằng việc lựa chọn các nút phục vụ vị trí được phân bổ theo một quy luật nhất định để cả thao tác cập nhật vị trí và thao tác yêu cầu vị trí đều được thực hiện

nhanh chóng và hiệu quả. Phương pháp được sử dụng phổ biến là chia vùng chứa các nút thành những ô (cell) nhỏ cố định, sau đó sử dụng hàm băm để xác định những ô tại đó các nút sẽ phục vụ vị trí cho nút T [30, 50, 59, 98]. Mặc dù tỏ ra hiệu quả hơn các phương pháp sử dụng phát tràn, các phương pháp chia ô và sử dụng hàm băm vẫn còn nhiều điểm cần khắc phục như xác định số lượng và phân bố các nút phục vụ vị trí phù hợp và hiệu quả, thay đổi các nút phục vụ vị trí khi nút di chuyển, xử lý các ô trống (ô không có nút nào), các ô thiếu cập nhật (ô có các nút phục vụ vị trí chưa được cập nhật thông tin khi nút được phục vụ đã di chuyển) và các nút phục vụ vị trí hết hạn phục vụ.

2.3.2 Chuyển tiếp dựa trên thông tin vị trí

Như đã được giới thiệu, chuyển tiếp dựa trên thông tin vị trí là chiến lược sử dụng thông tin vị trí, thường là của các nút láng giềng và nút đích, để chuyển gói tin đến nút láng giềng có độ đo, ví dụ khoảng cách đến đích, tốt hơn và tốt nhất. Nhiều độ đo khác nhau đã được sử dụng trong chuyển tiếp dựa trên thông tin vị trí, dẫn đến nhiều kỹ thuật chuyển tiếp dựa trên thông tin vị trí.

Kỹ thuật chuyển tiếp dựa trên thông tin vị trí được sử dụng rộng rãi nhất, vì tính đơn giản của nó, là *chuyển tiếp tham lam* (greedy forwarding) hay *chuyển tiếp theo khoảng cách* (distance-based forwarding) [24]. Trong chuyển tiếp này, khoảng cách đến đích được sử dụng làm độ đo khi lựa chọn nút kế tiếp. Cụ thể, nút hiện tại chọn láng giềng gần đích nhất và gần đích hơn nó làm nút kế tiếp. Nếu nút hiện tại không có láng giềng gần đích hơn, chuyển tiếp tham lam sẽ thất bại. Rất nhiều giao thức định tuyến được đề xuất sử dụng chuyển tiếp tham lam [8, 20, 46]. Chuyển tiếp tham lam có ưu điểm là đơn giản, hiệu quả và không tạo vòng lặp định tuyến nhưng có yếu điểm là dễ thất bại khi gặp các vùng trống [47]. Nếu chuyển tiếp tham lam được sử dụng thành công bởi tất cả các nút trên đường đi từ nguồn đến đích, đường đi sẽ gần với đường tối ưu. Nelson và các cộng sự đề xuất một biến thể của chuyển tiếp tham lam là Random Progress Method (RPM) [73]. RPM cũng sử dụng khoảng cách đến nút đích làm độ đo nhưng cho phép nút hiện tại chọn ngẫu nhiên một láng giềng gần đích hơn thay vì phải chọn láng giềng gần đích nhất. Như vậy, có thể xem RPM là chuyển tiếp tham lam được nói lỏng. Sự nói lỏng này mang đến cho RPM ưu điểm là phân tải tốt hơn nhưng cũng đem đến yếu điểm là tạo

đường đi dài hơn. Chưa có một minh chứng nào chứng minh RPM ít thất bại hơn chuyển tiếp tham lam.

Chuyển tiếp theo góc (compass forwarding) [6, 7, 52] cũng yêu cầu thông tin vị trí của các láng giềng và nút đích như chuyển tiếp tham lam nhưng sử dụng góc được tạo bởi hai vectơ, một từ nút hiện tại đến đích và một từ nút hiện tại đến láng giềng, làm độ đo. Láng giềng có góc bé nhất sẽ được chọn làm nút kế tiếp. So với chuyển tiếp tham lam, chuyển tiếp theo góc ít thất bại do cực tiểu địa phương nhưng có thể tạo ra vòng lặp định tuyến. Sự khác biệt này có được do chuyển tiếp theo góc không ràng buộc nút kế tiếp phải gần đích hơn. Chuyển tiếp theo góc chỉ thất bại khi nút hiện tại không có láng giềng nào. Khi nút hiện tại không có láng giềng gần đích hơn, gói tin được tạm thời đi xa đích. Sau đó, khi gói tin được chuyển tiếp gần đích hơn, vòng lặp định tuyến có thể được tạo ra. Chuyển tiếp theo góc bị chặn (bounded compass forwarding) [18], hay chuyển tiếp theo góc chỉ áp dụng cho các láng giềng gần đích hơn, đã được đề xuất nhằm tránh vòng lặp định tuyến. Dĩ nhiên, chuyển tiếp theo góc bị chặn dễ thất bại trước cực tiểu địa phương nhiều hơn chuyển tiếp theo góc không bị chặn.

Chuyển tiếp theo bước tiến (Most Forwarding progress with Radius - MFR) [88] sử dụng độ đo là độ dài hình chiếu của đoạn thẳng nối nút hiện tại đến láng giềng lên đường thẳng đi qua nút hiện tại và nút đích. Láng giềng gần đích hơn và có độ đo lớn nhất được chọn làm nút kế tiếp.

Chuyển tiếp với quán tính (inertia forwarding) [75] là kỹ thuật chuyển tiếp dựa trên thông tin vị trí kết hợp chuyển tiếp tham lam và di chuyển quán tính. Trong kỹ thuật này, nếu quan niệm gói tin là một vật di chuyển thì di chuyển của nó là do tác dụng của lực hút của nút đích cùng với lực quán tính. Di chuyển dưới lực hút của đích đưa gói tin tiến dần về đích trong khi di chuyển theo quán tính có thể giúp gói tin di chuyển theo biên của vùng trống để thoát vùng trống.

Bảng 2.4 tóm tắt những điểm khác nhau chính giữa các kỹ thuật chuyển tiếp dựa trên thông tin vị trí.

2.3.3 Cực tiểu địa phương

Chuyển tiếp dựa trên thông tin vị trí thất bại khi nút hiện tại không thể chọn nút láng giềng làm nút chuyển tiếp tiếp theo. Những nút tại đó không áp dụng thành công chuyển tiếp dựa trên thông tin vị trí được gọi là cực tiểu địa phương. Một cách trực quan, cực tiểu địa phương nằm ở biên của vùng trống và biên của mạng.

Cực tiểu địa phương là vấn đề không thể tránh khỏi đối với chuyển tiếp dựa trên thông tin vị trí. Phương pháp giảm thiểu và khôi phục sau cực tiểu địa phương đã là đối tượng nghiên cứu chính của định tuyến dựa trên thông tin vị trí. Các mục tiếp sau trình bày tổng quan các kết quả nghiên cứu về vấn đề này.

Bảng 2.4. So sánh các kỹ thuật chuyển tiếp dựa trên thông tin vị trí.

<i>Kỹ thuật chuyển tiếp</i>	<i>Độ đo được sử dụng</i>	<i>Tạo vòng lặp định tuyến</i>	<i>Khả năng thất bại do cực tiểu địa phương</i>
Tham lam [24, 73]	Khoảng cách	Không	Cao
Theo góc [6, 7, 52]	Góc	Có	Trung bình
Theo bước tiến [88]	Bước tiến	Không	Cao
Với quán tính [75]	Khoảng cách và quán tính	Có	Thấp

2.3.4 Giảm thiểu và tránh cực tiểu địa phương

Kỹ thuật giảm thiểu và tránh cực tiểu địa phương được đề xuất sớm nhất là *sử dụng thông tin vùng lân cận*. Lim và các cộng sự [62] đề xuất sử dụng thông tin về phân mạng bao gồm các nút cách xa nút hiện tại không quá k chặng cùng các liên kết của giữa các nút này. Các tác giả gọi phân mạng này là *tầm vực* (viewscope). Mỗi nút lưu thông tin vị trí của tất cả các nút nằm trong tầm vực của nó, đồng thời lưu các đường đi đến các nút này; để chuyển tiếp gói tin, nút hiện tại lựa chọn một nút nằm trong tầm vực của nó gần nút đích nhất rồi gửi gói tin cho nút liền kề trên đường đi đến nút được chọn. Chiến lược được đề xuất bởi Lim và các cộng sự có thể tránh được các vùng trống có kích thước nhỏ nằm trong tầm vực của mỗi nút. Để tránh cực tiểu địa phương tại các vùng trống có kích thước lớn hơn, các nút phải có tầm vực lớn hơn, tức phải lưu nhiều thông tin hơn và mất nhiều chi phí hơn cho duy trì các bảng định tuyến. Dĩ nhiên nếu tầm vực quá lớn, giao thức của Lim và các cộng sự trở thành định tuyến chủ động dựa trên topo thông

thường. Cũng với tư tưởng sử dụng thông tin vùng lân cận, F. Xi và các cộng sự đã đưa ra khái niệm biết về topo (topological awareness – TA) với mục đích nắm thông tin về vùng lân cận rộng [95]. TA bao gồm hai phần: mịn và thô. Phần mịn cung cấp thông tin về vị trí các nút ở gần trong khi phần thô cung cấp thông tin mang tính bao quát về phân bố các nút ở xa. Với sự có mặt của TA, nút hiện tại chọn nút thuộc TA và gần đích nhất làm điểm neo, tiếp theo chọn nút thuộc phần mịn của TA gần điểm neo nhất làm điểm đích trung gian, rồi sử dụng chuyển tiếp tham lam để chuyển gói tin hướng tới điểm đích trung gian. Những ưu điểm của sử dụng TA bao gồm nâng cao tỷ lệ chuyển gói tin đến đích thành công và giảm độ dài các đường đi. Tuy nhiên, việc duy trì TA khá tốn kém. Với những ưu điểm và hạn chế của TA như vậy, F. Xi và các cộng sự khuyến cáo sử dụng giao thức của họ cho các mạng tĩnh, hay topo ít khi thay đổi. Trường hợp khác đơn giản hơn, L. C. Wu và các cộng sự đề xuất giao thức sử dụng vùng lân cận bao gồm các nút cách xa nút hiện tại không quá 2 chặng [94] và phân loại các nút thành nút đỏ và nút trắng. Mỗi nút xem nó như tâm của mặt phẳng trong hệ tọa độ Đề-các và tự đánh dấu nó là nút đỏ nếu ở mỗi phần tư mặt phẳng có ít nhất một nút thuộc vùng lân cận được xét, hoặc tự đánh dấu nó là nút trắng nếu ngược lại. Một nút đỏ sẽ không trở thành cực tiểu địa phương trong khi một nút trắng có thể. Nút hiện tại chọn nút đỏ thuộc vùng lân cận được xét và gần đích nhất làm nút kế tiếp (hoặc đích tạm thời). Nếu không có nút đỏ nào thuộc vùng lân cận được xét, nút hiện tại sử dụng chuyển tiếp tham lam để chuyển gói tin. Giải pháp của L. C. Wu và các cộng sự có tác dụng giảm cực tiểu địa phương nhưng còn mang tính thô sơ và hiệu quả giảm thiểu cực tiểu địa phương chưa cao.

Một kỹ thuật tránh cực tiểu địa phương khác đã được đề xuất là *mô hình hóa và cảnh báo vùng trống*. Theo kỹ thuật này, các vùng trống được phát hiện và mô hình hóa bằng các đối tượng hình học như hình tròn [101], eclipse [60, 89, 100], đa giác lồi [87], chứa toàn bộ vùng trống. Giao thức BOUNDHOLE [20] được sử dụng để phát hiện các vùng trống. Cụ thể, một gói tin phát hiện biên vùng trống (Hole Boundary Detection-HBD) được sinh ra bởi một nút biên và được chuyển tiếp dọc theo biên của vùng trống cho đến khi quay về nút đã sinh ra nó. Thông tin vị trí các nút biên lần lượt được ghi trong thân của gói HBD. Khi nhận lại gói HBD, nút đã sinh ra nó có thông tin đầy đủ về vùng trống và tiến hành mô hình hóa vùng trống bằng một đối tượng hình học chứa toàn bộ vùng trống ở bên trong và sử dụng kết quả mô hình hóa như biểu diễn của vùng trống. Tiếp đó,

thông tin biểu diễn vùng trống được quảng bá đến tất cả các nút bên trong vùng biểu diễn bằng gói tin thông báo vùng trống (Hole Announcement-HA). Một số giao thức, ví dụ [60], còn quảng bá thông tin biểu diễn vùng trống đến tất cả các nút cách biên của vùng trống không quá h chặng, trong đó h là một tham số. Với sự có mặt của thông tin về vùng trống, các điểm neo, hay đích trung gian, được tính toán và sử dụng trong chuyển tiếp gói tin. Cụ thể, các nút sẽ xem điểm neo như đích tạm thời của gói tin và cố gắng chuyển gói tin bằng chuyển tiếp dựa trên thông tin vị trí đến đích tạm thời đó thay vì đến đích thực sự của gói tin. Điểm neo có tác dụng chỉ đường để gói tin không rơi vào cực tiểu địa phương, tức tránh được các vùng trống. Điểm neo có thể được tính toán một lần bởi nút đầu tiên có thông tin về vùng trống nhận được gói tin [89, 100] và được ghi vào tiêu đề của gói tin. Nút gần điểm neo nhất sẽ xóa thông tin về điểm neo khi nhận được gói tin. Sau đó, gói tin được định tuyến đến đích của nó như thông thường. Theo một phương pháp khác linh động và hiệu quả hơn, mỗi nút tự tính điểm neo cho riêng nó [61]. Như vậy, điểm neo liên tục thay đổi và có tác dụng chỉ ra đường tốt hơn. Cách tính điểm neo của một số giao thức được mô tả như sau: Trong [101], vùng trống được biểu diễn bằng hình tròn, điểm neo là giao điểm của hai đường tiếp tuyến của hình tròn, một tiếp tuyến đi qua điểm nguồn, tiếp tuyến còn lại đi qua điểm đích của gói tin. Trong [100] và [89], vùng trống được biểu diễn bằng hình eclipse, điểm neo là một điểm nằm trên tiếp tuyến của eclipse có tiếp điểm là nút hiện tại nhận gói tin. Trong [60], vùng trống cũng được biểu diễn bằng hình eclipse và được quảng bá rộng hơn đến cả các nút nằm ngoài eclipse, điểm neo được tính là giao điểm của hai tiếp tuyến của eclipse, một tiếp tuyến đi qua nút hiện tại, tiếp tuyến còn lại đi qua điểm đích của gói tin. Trong [87], vùng trống được biểu diễn bằng đa giác lồi, điểm neo được chọn từ bốn điểm có tọa độ (min_x, min_y) , (min_x, max_y) , (max_x, min_y) và (max_x, max_y) trong đó min_x , min_y , max_x và max_y là tọa độ x nhỏ nhất, tọa độ y nhỏ nhất, tọa độ x lớn nhất và tọa độ y lớn nhất của tất cả các nút biên của vùng trống.

Sử dụng tọa độ ảo thay vì tọa độ thật nhằm giảm thiểu cực tiểu địa phương là kỹ thuật mang tính sáng tạo. Ảo hóa tọa độ có tác dụng thay đổi hình dạng, cụ thể là lồi hóa, vùng trống do vậy giảm được các cực tiểu địa phương. Để tăng tỷ lệ chuyển gói tin đến đích thành công của chuyển tiếp dựa trên thông tin vị trí, Liu và các cộng sự đã đề xuất Aligned Virtual Coordinate System (AVCS) [63]. Tọa độ ảo của mỗi nút được tính dựa

trên số chặng từ nút đó đến các nút tham chiếu. AVCS đạt được mục đích tăng tỷ lệ chuyển gói tin đến đích thành công của chuyển tiếp dựa trên thông tin vị trí nhưng việc thiết lập AVCS hết sức tốn kém do các nút tham chiếu phải phát tràn các gói tin điều khiển trong mạng. Cũng với mục đích nâng cao tỷ lệ chuyển gói tin đến đích thành công của chuyển tiếp dựa trên thông tin vị trí, một tọa độ ảo đa chiều đã được đề xuất [103]. Theo kỹ thuật này, một số nút trong mạng được chọn ngẫu nhiên làm các điểm mốc. Tọa độ ảo của mỗi nút, được gọi là HopID, là bộ các số chặng ít nhất từ nút đó đến các điểm mốc. HopID giúp tránh cực tiểu địa phương hiệu quả nhưng việc thiết lập nó còn tốn kém hơn thiết lập AVCS. Vận dụng định lý về biến đổi đồ thị phẳng⁵ của W. Tutte [90], Brandes và các cộng sự đã đề xuất một giao thức xây dựng tọa độ ảo [10]. Theo giao thức này, đồ thị phẳng Gabriel G_{GG} được xây dựng, các nút biên được phát hiện và đặt cố định trên một vòng tròn bao G_{GG} , nhiều vòng lặp được thực hiện để tính tọa độ ảo của các nút trong (tại mỗi vòng lặp, giá trị mỗi chiều thuộc tọa độ của từng nút trong được tính bằng trung bình cộng các giá trị cùng chiều của tọa độ các nút láng giềng). Ngoài việc gán tọa độ ảo cho các nút, giao thức cũng tạo ra những liên kết ảo (đại diện cho một đường đi) giữa các nút. Sử dụng tọa độ ảo được xây dựng, các tác giả đã đề xuất các phương pháp chuyển tiếp đảm bảo (sẽ đưa gói tin đến đích nếu tồn tại đường đi đến nó). Ưu điểm của tọa độ ảo này là có thể áp dụng các phương pháp chuyển tiếp đơn giản, đảm bảo và tạo ra đường đi ngắn hơn so với các phương pháp chuyển tiếp trên tọa độ thật. Tuy nhiên, hạn chế của giải pháp này là việc tính tọa độ ảo có độ phức tạp lớn, cần nhiều thời gian để hoàn thành. Trong [2], các nút được định vị lại bằng việc thêm một chiều mới vào tọa độ. Bằng thủ tục xử lý cục bộ, các nút gần đoạn lõm của biên được phát hiện và chiều mới thêm trong tọa độ của chúng được nâng lên trong khi giá trị chiều mới thêm trong tọa độ của các nút còn lại được giữ nguyên ở 0. Với việc sử dụng tọa độ mới này, chuyển tiếp tham lam cho tỷ lệ thành công cao hơn, nghĩa là giảm được các cực tiểu địa phương. Các tác giả cũng đã đề xuất thủ tục đi theo biên của vùng trống sử dụng hệ tọa độ mới cho mục đích khôi phục. Trong [99], tọa độ ảo được tính đơn giản là điểm giữa của tất cả các láng giềng, tức giá trị mỗi chiều là trung bình giá trị cùng chiều của tất cả các láng giềng. Do tính đơn giản và hiệu quả trong việc tính tọa độ ảo, các tác giả đã đề xuất tọa độ ảo đa cấp, hay tọa độ ảo của tọa độ ảo. Chuyển tiếp tham lam và MFR sử dụng tọa độ

⁵ “Fixing the nodes of a face of a planar embedded, 3-connected graph onto the corners of a convex polygon C and setting the remaining nodes to the barycenter of their neighbors, yields a planar embedding” [86].

ảo đa cấp làm giảm thiểu nhiều cực tiểu địa phương trong khi không tạo vòng lặp định tuyến. Một tiếp cận khác trong xây dựng tọa độ ảo cho các nút mạng là sử dụng biến đổi hình học. Với quan sát cực tiểu địa phương không xuất hiện ở biên của hình tròn, Sarkar và các cộng sự [83] đã phát triển một giao thức gán tọa độ ảo cho các nút sao cho tất cả các biên, kể cả biên ngoài của mạng và biên của các vùng trống, sau ánh xạ đều là các đường tròn. Đầu tiên, một lưới tam giác với mắt lưới là các nút và đường lưới là các liên kết giữa các nút được tạo theo giao thức [28]. Quan niệm lưới tam giác vừa tạo như một mặt với độ cong bằng 0, dựa trên lý thuyết hình học vi phân, các tác giả đã sử dụng một ánh xạ bảo toàn góc để ánh xạ bề mặt vừa tạo lên một mặt tròn với tất cả các biên được biến đổi thành đường tròn. Các tác giả đã sử dụng dòng Ricci [34] trong hình học vi phân với giao thức tính dòng Ricci rời rạc được đề xuất trong [42] có cơ sở chứng minh trong [17] để tạo ánh xạ bảo toàn góc nói trên. Thành công của các tác giả là toàn bộ các nút mạng được gán tọa độ ảo và chuyển tiếp tham lam không bao giờ thất bại trên tọa độ ảo được tạo ra. Tuy nhiên, độ phức tạp trong tính toán tọa độ ảo còn khá lớn. Ngoài ra, khi có sự thay đổi về topo mạng, tọa độ ảo cần phải được tính lại.

Bảng 2.5 tóm tắt những đặc điểm chính của các kỹ thuật tránh cực tiểu địa phương đã được trình bày và phân tích ở trên.

Bảng 2.5. So sánh các kỹ thuật chuyển tiếp dựa trên thông tin vị trí.

<i>Kỹ thuật tránh cực tiểu địa phương</i>	<i>Yêu cầu lưu trữ</i>	<i>Độ phức tạp thông báo</i>	<i>Độ phức tạp tính toán</i>	<i>Thời gian hội tụ</i>	<i>Khả năng tránh cực tiểu địa phương</i>
Sử dụng thông tin vùng lân cận [62, 94, 95]	Trung bình hoặc lớn	Trung bình hoặc cao	Trung bình hoặc cao	Trung bình hoặc chậm	Một phần
Mô hình hóa và cảnh báo vùng trống [60, 87, 89, 100, 101]	Trung bình	Cao	Cao	Chậm	Hoàn toàn
Sử dụng tọa độ ảo [2, 10, 17, 34, 42, 63, 83, 99, 103]	Trung bình	Trung bình hoặc cao	Trung bình hoặc cao	Trung bình hoặc chậm	Một phần

2.3.5 Khôi phục sau cực tiểu địa phương

Khi gặp cực tiểu địa phương, gói tin được chuyển sang chế độ khôi phục, vị trí của cực tiểu địa phương được ghi vào tiêu đề của gói tin. Ở chế độ khôi phục, gói tin được chuyển bằng định tuyến khôi phục.

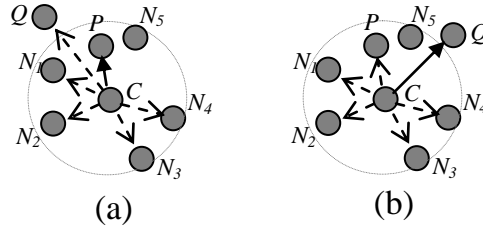
Kỹ thuật đơn giản nhất được sử dụng cho gói tin ở chế độ khôi phục là *phát tràn*. Giao thức trong [88] quảng bá gói tin sau khi gói tin gặp cực tiểu địa phương. Trong [41], khi gói tin gặp cực tiểu địa phương, một thủ tục tìm đường bằng tìm kiếm theo chiều sâu được sử dụng để tìm đường đi đến đích. Thông tin về đường đi được lưu trong vùng đệm ở các nút và được sử dụng cho các gói tin khác nữa.

Kỹ thuật *quay lui* cũng đã được đề xuất cho mục đích khôi phục. Trong [31], nếu gói tin gặp cực tiểu địa phương, cực tiểu địa phương sẽ phát một thông báo “I am a hole-node” đến tất cả láng giềng của nó, rồi gửi gói tin đến láng giềng gần đích nhất. Khi nhận được thông báo trên, nút nhận đánh dấu nút gửi là một nút thuộc vùng trống; nếu tất cả các láng giềng đã trở thành nút thuộc vùng trống nút nhận cũng tự đánh dấu nó thuộc vùng trống. Thủ tục này được lặp lại cho đến khi gói tin được chuyển đến đích hoặc bị loại bỏ do hết thời gian sống.

Một kỹ thuật khôi phục phổ biến và được nghiên cứu nhiều là *định tuyến trên mặt* (face routing). Theo kỹ thuật này, các nút đồng thuận không sử dụng một số liên kết trong chuyển tiếp gói tin ở chế độ khôi phục sao cho những liên kết còn lại cùng các nút tạo thành một đồ thị phẳng. Quá trình tính toán hoàn toàn mang tính cục bộ, do vậy có chi phí không đáng kể. Khi chuyển tiếp tham lam thất bại, gói tin được chuyển theo các cạnh của đồ thị phẳng bằng kỹ thuật duyệt đồ thị. Nhiều giao thức đã được đề xuất sử dụng kỹ thuật này [46, 49, 54, 55, 58, 65]. Nhược điểm chính của kỹ thuật này là các phương pháp tính đồ thị phẳng hiện có có thể tạo đồ thị phẳng không chính xác khi các nút có phạm vi phát sóng không thuận nhất, tức có kích thước vùng phủ sóng khác nhau hoặc vùng phủ sóng không phải là hình tròn, do vậy dẫn đến định tuyến thất bại.

Với quan sát rằng cực tiểu địa phương chỉ xuất hiện ở biên của các vùng trống và biên của mạng, một phương pháp khôi phục đã được đề xuất là chuyển gói tin *đi theo biên* (boundary detouring) [20, 64, 70]. Ưu điểm của kỹ thuật này là tính đơn giản, khả mở

rộng, hiệu quả và quan trọng là tính thực tiễn hay khả áp dụng. Hạn chế chính của kỹ thuật này là đặt nhiều lưu lượng hơn trên các nút biên dẫn đến tắc nghẽn tại biên và giảm tuổi thọ của các nút biên hay khoét rộng các vùng trống. Ngoài ra, các đường đi được tạo ra theo kỹ thuật đi theo biên thường không tối ưu. Do tính thực tiễn, đơn giản và hiệu quả, kỹ thuật đi theo biên đã được sử dụng trong nhiều giao thức. Do vậy, phần này trình bày chi tiết kỹ thuật này: BOUNDHOLE [20] là một kỹ thuật đi theo biên hiệu quả. Khi gói tin gặp cực tiểu địa phương, BOUNDHOLE sử dụng *kỹ thuật quay* (sweeping) để chuyển gói tin dọc theo biên. Kỹ thuật quay được mô tả như sau. Gọi C là nút hiện tại, P là nút liền trước (theo trình tự nhận gói tin) C , và Q là nút liền trước P . Nếu P là láng giềng duy nhất của C thì P được chọn làm nút tiếp theo (nghĩa là quay lui). Ngược lại, gọi $N_1, N_2, N_3 \dots$ là các láng giềng khác của C . Thủ tục chọn láng giềng làm nút chuyển tiếp tiếp theo như sau. Nếu Q bên phải \overrightarrow{CP} thì đặt $\vec{R} = \overrightarrow{CQ}$, ngược lại đặt $\vec{R} = \overrightarrow{CP}$. Tiếp theo, quay \vec{R} theo chiều kim đồng hồ. Láng giềng đầu tiên được gặp bởi \vec{R} sẽ được chọn làm nút tiếp theo. Hình 2.2 cho chúng ta minh họa về kỹ thuật quay. Trong ví dụ thứ nhất ở Hình 2.2(a), Q ở bên trái \overrightarrow{CP} , \vec{R} được đặt bằng \overrightarrow{CP} và N_5 được chọn làm nút tiếp theo. Trong ví dụ thứ hai ở Hình 2.2(b), Q ở bên phải \overrightarrow{CP} , \vec{R} được đặt bằng \overrightarrow{CQ} , và N_4 được chọn làm nút tiếp theo. BOUNDHOLE là kỹ thuật đơn giản, dễ mở rộng và hiệu quả. Nó không yêu cầu thêm thông tin và tính toán so với chuyển tiếp tham lam. Ưu điểm nữa của BOUNDHOLE là nó cho tỷ lệ chuyển gói tin đến đích thành công cao ngay cả trong các mạng thưa. Kỹ thuật này có thể chuyển gói tin qua vùng trống có hình thù bất kỳ. GRIC [74] là một kỹ thuật đi dọc biên khác. Nó sử dụng chuyển tiếp với quán tính khi gói tin ở chế độ tham lam. Ở chế độ khôi phục, gói tin cũng được chuyển đi bằng chuyển tiếp với quán tính nhưng áp dụng quy tắc *tay phải* (right-hand rule) được đề xuất trong [38] để đi dọc theo biên của vùng trống. GRIC tuân thủ quy tắc tay phải bằng chuyển tiếp với quán tính ngược, hay chuyển tiếp với quán tính với hướng lý tưởng quay ngược. Nhờ vậy, GRIC có thể chuyển gói tin dọc theo biên của các vùng trống đơn giản có đường biên lồi với chi phí thấp.



Hình 2.2. Kỹ thuật quay. C có năm láng giềng khác P là N_1, N_2, N_3, N_4 và N_5 . (a) N_5 được chọn làm nút tiếp theo khi Q ở bên trái \overline{CP} . (b) N_4 được chọn làm nút tiếp theo khi Q ở bên phải \overline{CP} .

Sử dụng cây bao lồi là kỹ thuật khác nữa được dùng cho mục đích khôi phục. Trong [58], các tác giả đã sử dụng cây bao trùm và đề xuất giao thức xây dựng cho mỗi nút v một bao lồi (convex hull), tức đa giác lồi bao tất cả các nút thuộc cây con có gốc là v . Việc tính toán bao lồi được thực hiện lần lượt ở các nút từ lá đến gốc của cây bao trùm. Khi một nút có đủ thông tin về các bao lồi của các nút con, nó dễ dàng tính bao lồi cho nó rồi gửi thông tin bao lồi vừa tính được cho nút cha (nếu không phải là nút gốc). Cây bao trùm với thông tin bao lồi ở các nút được các tác giả gọi là *cây bao* (hull tree). Cây bao được sử dụng trong khôi phục sau cực tiểu địa phương như sau. Ở chế độ khôi phục, gói tin được chuyển tiếp theo các cạnh của cây bao trùm. Nút hiện tại sẽ xét xem nút đích có nằm trong bao lồi của của cây con nào hay không. Nếu có, nó chuyển gói tin xuống cây con này. Nếu tất cả bao lồi của các cây con đều không chứa nút đích, nút hiện tại chuyển gói tin lên nút cha nếu nó không phải là nút gốc hoặc kết luận gói tin không thể đến được đích nếu nút hiện tại là nút gốc. Kết quả mô phỏng của các tác giả cho thấy giao thức sử dụng cây bao cho đường đi ngắn hơn sử dụng định tuyến trên mặt. Một vấn đề trong giải pháp của các tác giả là sự chồng nhau giữa các bao lồi. Nếu nút đích nằm trong các bao lồi của nhiều nút con, giao thức phải duyệt lần lượt từng cây con này để tìm nút đích. L. Zhang và các cộng sự đã đề xuất giao thức tạo cây bao [102] để các bao lồi ít chồng nhau hơn, đồng thời những liên kết có băng thông lớn hơn cũng như những liên kết giữa các nút còn nhiều năng lượng hơn được chọn nhiều hơn. Theo giao thức này, cây bao được tạo từ việc kết hợp các cây bao nhỏ hơn, xuất phát mỗi nút là một cây bao. Mỗi cây bao được “chứa” trong một cụm (cluster) với đầu cụm CH (cluster header) quản lý các thông tin về tổng băng thông của các liên kết giữa các nút thuộc cụm (gọi tắt là băng thông cụm), tổng năng lượng của các nút thuộc cụm (gọi tắt là năng lượng cụm),

tâm của cụm (có tọa độ là trung bình tọa độ các nút thuộc cụm), các cụm láng giềng (hai cụm được gọi là láng giềng của nhau nếu tồn tại ít nhất một nút thuộc cụm này là láng giềng của nút thuộc cụm còn lại). Khoảng cách giữa hai cụm được định nghĩa là khoảng cách giữa hai tâm của chúng. Tiếp đó, lực hấp dẫn (gravity) giữa hai cụm được xác định bằng $(\text{tổng băng thông hai cụm}) \times (\text{tổng năng lượng hai cụm}) / (\text{bình phương khoảng cách giữa hai cụm})$. Mỗi cụm sẽ được kết hợp với một cụm láng giềng có lực hấp dẫn lớn nhất để tạo thành một cụm mới lớn hơn. Việc kết hợp hai cụm bao gồm chọn liên kết giữa hai nút thuộc hai cụm có băng thông lớn nhất để nối hai cây bao thành một cây bao, tính giá trị các thông số cho cụm mới, và thống nhất một trong hai đầu cụm sẽ trở thành nút thường để đầu cụm còn lại trở thành đầu của cụm mới. Kết quả mô phỏng của các tác giả cho thấy cây bao được tạo ra có rất ít các bao lỗi chồng nhau, các đường đi được tạo ra ngắn hơn, đồng thời trung bình năng lượng và băng thông các đường đi cao hơn.

Bảng 2.6 tóm tắt những điểm khác nhau chính giữa các chiến lược khôi phục được khảo sát.

Bảng 2.6. So sánh các kỹ thuật khôi phục.

<i>Kỹ thuật khôi phục</i>	<i>Chiến lược</i>	<i>Yêu cầu duy trì thông tin phụ trợ</i>	<i>Độ phức tạp thông báo</i>	<i>Độ phức tạp tính toán</i>	<i>Độ dài đường đi/Trễ đầu cuối – đầu cuối</i>
Phát tràn [41, 88]	Đa đường		Cao	Thấp	Nhỏ
Quay lui [31]	Đơn đường		Trung bình	Thấp	Lớn
Định tuyến trên mặt [46, 49, 54, 55, 58].	Đơn đường	Đồ thị phẳng	Trung bình	Trung bình	Trung bình
Đi theo biên [20]	Đơn đường		Thấp	Thấp	Trung bình
Sử dụng cây bao lỗi [58, 102]	Đơn đường	Cây bao	Cao	Cao	Trung bình

2.4 Thảo luận

Định tuyến dựa trên thông tin vị trí là tiếp cận tốt cho mạng cảm biến không dây do điều kiện hạn chế về tài nguyên của các nút mạng. Trong nhiều giao thức đã được đề xuất,

định tuyến dựa trên thông tin vị trí kết hợp chuyển tiếp tham lam [24] và kỹ thuật khôi phục đi theo biên [20] là giải pháp hiệu quả và khả thi. Tuy nhiên, định tuyến theo phương pháp này có hai yếu điểm chính. Thứ nhất, các đường đi dọc theo biên thường dài và không tối ưu. Thứ hai, nhiều đường đi dọc theo biên dẫn đến lưu lượng quá tải cho các nút biên. Điều này không chỉ dẫn đến tắc nghẽn tại biên khi có nhiều luồng lưu lượng đồng thời mà còn làm giảm nhanh tuổi thọ của các nút biên dẫn đến khoét rộng hơn các vùng trống.

Các phương pháp giảm thiểu và tránh cực tiểu địa phương giúp hạn chế những thất bại của chuyển tiếp tham lam nhưng yêu cầu các nút phải thực thi một giao thức tiên định tuyến nhằm xây dựng thông tin định tuyến hoặc thay đổi tọa độ của mỗi nút. Giao thức tiên định tuyến sử dụng các gói tin điều khiển, yêu cầu lưu trữ tại mỗi nút cảm biến và cần thời gian để hoàn thành, do đó làm giảm tính hấp dẫn của định tuyến dựa trên thông tin vị trí.

Để có thể áp dụng định tuyến dựa trên thông tin vị trí, mỗi nút cảm biến phải biết vị trí của nó. Sử dụng thiết bị định vị để có thông tin vị trí của các nút không phải là giải pháp được ưa chuộng cho mạng cảm biến không dây, đặc biệt là các mạng có quy mô lớn, do các vấn đề về chi phí, tiêu thụ năng lượng, kích thước nút cảm biến, ... Do vậy, các thuật toán định vị đã được nghiên cứu nhằm thay thế cho sử dụng thiết bị định vị. Trong nhiều thuật toán định vị đã được đề xuất, định vị dựa trên kết nối sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách [56] là giải pháp hiệu quả và cho kết quả tốt. Để thuật toán này hoạt động, một thuật toán phát hiện biên được yêu cầu chạy trước nhằm phát hiện các nút biên. Tuy nhiên, các thuật toán phát hiện biên đã được đề xuất hoặc có chi phí cao hoặc cho kết quả không tốt và chỉ hoạt động trên các mạng có mật độ nút cao.

Từ kết quả khảo sát, phân tích và so sánh các thuật toán như đã được trình bày ở trên, giải pháp cho vấn đề định vị được lựa chọn là sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách của Lederer và các cộng sự [56]. Với định tuyến đơn phát dựa trên thông tin vị trí, giải pháp được lựa chọn là kết hợp chuyển tiếp tham lam [24] và kỹ thuật

khôi phục đi theo biên của Fang và các cộng sự [20]. Để hoàn thiện giải pháp được đề xuất, các vấn đề sau sẽ được nghiên cứu giải quyết (ở các chương tiếp theo):

- Phát hiện biên dựa trên kết nối nhằm hỗ trợ giải pháp định vị đã được lựa chọn.
- Nâng cao hiệu năng, khắc phục những hạn chế còn tồn tại của giải pháp định tuyến dựa trên thông tin vị trí đã được lựa chọn.

CHƯƠNG 3

HỖ TRỢ ĐỊNH VỊ VỚI PHÁT HIỆN BIÊN DỰA TRÊN KẾT NỐI

Định vị dựa trên kết nối, sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách [56], là phương pháp định vị hiệu quả và khả thi cho mạng cảm biến không dây có quy mô lớn. Phát hiện biên dựa trên kết nối là công đoạn đầu trong định vị dựa trên kết nối. Tuy nhiên, như đã được phân tích trong Mục 2.2 của Chương 2, cho đến nay chưa có thuật toán phát hiện biên dựa trên kết nối nào có tính khả thi và hiệu quả cao.

Trong chương này, một thuật toán phát hiện biên dựa trên kết nối được đề xuất. Thuật toán này có độ phức tạp tính toán và truyền thông thấp, có thể làm việc tốt trên cả các mạng cảm biến có mật độ thấp. Theo thuật toán này, mỗi nút đánh giá đồ thị vùng lân cận 2 chặng của nó để quyết định nó có nằm gần biên hay không. Một nút nằm gần biên khi và chỉ khi đồ thị vùng lân cận 2 chặng của nó không tạo thành một cái vành. Việc xây dựng và đánh giá đồ thị vùng lân cận 2 chặng là đơn giản và ít tốn kém.

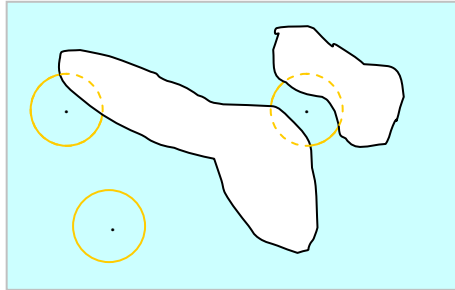
Phần đầu của chương này trình bày thuật toán được đề xuất. Tiếp theo, thuật toán có thể đáp ứng nhanh như thế nào với thay đổi mạng được phân tích. Hiệu năng của thuật toán được đánh giá và so sánh với các thuật toán đã có. Cuối cùng, một số thảo luận và hướng phát triển mở rộng thuật toán được đề cập ở cuối chương này.

3.1 Tìm biên dựa trên kết nối

3.1.1 Trực quan và heuristic

Xét khu vực $R \subseteq \mathbb{R}^2$ với một số vùng trống trong nó. Với mỗi điểm $p \in R$, xét đường tròn $c(p, r)$, được gọi là đường tròn của p , có tâm tại p và bán kính r , trong đó r là một số

thực. Nếu p gần biên, tức có điểm thuộc biên cách p không quá r , thì $c(p, r)$ bị cắt thành các cung *liền nét* và *đứt nét* xen kẽ nhau. Cung liền nét và cung đứt nét là các khái niệm được đưa ra: cung liền nét bao gồm các điểm không thuộc vùng trống; cung đứt nét bao gồm các điểm thuộc vùng trống hoặc điểm nằm ngoài khu vực đang xét. Quan sát trên được minh họa trong Hình 3.1.

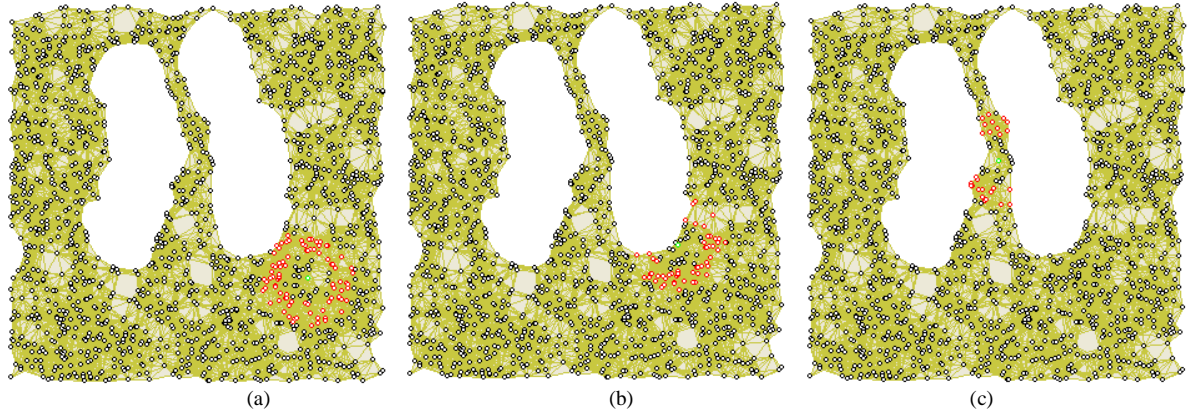


Hình 3.1. Biên và vùng trống trong trường hợp liên tục. Một khu vực với hai vùng trống và ba điểm được đánh giá. Đường tròn có tâm tại điểm gần biên bị chia cắt thành các cung *liền nét* và *đứt nét* xen kẽ nhau.

Từ quan sát trên, trong mạng cảm biến không dây, với mỗi nút p , xét đồ thị được tạo bởi các nút cách p hai chặng và liên kết giữa chúng. Đồ thị này được gọi là đồ thị vùng lân cận 2 chặng của p , viết tắt là 2NG (*2-hop neighbourhood graph*). Về mặt trực quan, nếu p không nằm gần biên thì 2NG của nó tạo thành một cái “vành” (“ring”) (nghĩa là có hình dạng tựa một cái vành); ngược lại 2NG của p bao gồm một hoặc nhiều “mảnh vỡ” của một cái vành. Điều này được minh họa trong Hình 3.2.

Trực quan được mô tả ở trên dẫn đến một heuristic mô phỏng trường hợp liên tục trên mặt phẳng: *p gần biên nếu và chỉ nếu 2NG của nó không tạo thành một cái vành.*

Tính chất quan sát được ở trên của 2NG cũng đúng cho các đồ thị vùng lân cận k chặng (được tạo bởi các nút cách xa nút khảo sát k chặng và liên kết giữa các nút này) với $k > 2$, nhưng không đúng cho đồ thị vùng lân cận 1 chặng. Tuy nhiên, những nút tương đối xa biên cũng có thể có đồ thị vùng lân cận k chặng không tạo thành cái vành nếu $k > 2$.



Hình 3.2. Biên và vùng trống trong trường hợp rời rạc. Một nút với 2NG của nó. Các cạnh biểu diễn liên kết giữa các nút. Nút màu đỏ là nút thuộc 2NG của nút màu xanh lá cây. (a) Nút không gần biên; (b) Nút gần một biên; (c) Nút gần hai biên.

3.1.2 Thuật toán

Heuristic được đưa ra ở trên dẫn đến một thuật toán đơn giản nhưng hiệu quả để phát hiện các nút gần biên, như được mô tả trong Bảng 3.1. Mỗi nút p khám phá tất cả các láng giềng của nó, tạo danh sách các láng giềng ($MYNB$) rồi gửi danh sách các láng giềng đến tất cả các nút cách nó hai chặng. Nếu nhận được danh sách láng giềng từ các nút cách p hai chặng, p tạo 2NG của nó, gọi là $G2$, dựa trên các danh sách các láng giềng đã nhận được. Tiếp theo, p kiểm tra $G2$ có tạo thành một cái vành hay không bằng cách gọi hàm $IsRing(G2)$. Nếu $G2$ là một cái vành, nghĩa là $IsRing(G2)$ trả về *true*, thì đặt $nearBoundaries = false$ (p không gần biên), ngược lại đặt $nearBoundaries = true$ (p gần biên).

Bảng 3.1. Thuật toán phát hiện biên được đề xuất, mã cho nút p .

Vào: Không.

Ra: $nearBoundaries = true$ nếu p gần biên, $nearBoundaries = false$ nếu p không gần biên.

Khởi tạo:

$nearBoundaries \leftarrow false$

$MYNB \leftarrow \{\text{tất cả các láng giềng của } p\}$

$V \leftarrow \{\}$

$PE \leftarrow \{\}$

broadcast(p.id, MYNB, 1)

onReceive(id, NB, hop):

if *hop* = 1 then *broadcast(id, nb, 2)*

else if *id* $\notin V$ and *id* $\notin MYNB$ and *id* $\neq p.id$ then

$V \leftarrow V \cup \{id\}$

foreach *v* $\in NB$

$PE \leftarrow PE \cup \{<id, v>\}$

$E \leftarrow \{\}$

foreach $<v1, v2> \in PE$

if $v1 \in V$ and $v2 \in V$ then

$E \leftarrow E \cup \{<v1, v2>\}$

$G2 \leftarrow \langle V, E \rangle$

if *IsRing*($G2$) then *nearBoundaries* $\leftarrow false$

else *nearBoundaries* $\leftarrow true$

***IsRing*($G2 \equiv \langle V2, E2 \rangle$):**

if not *IsConnected*($G2$) then return *false*

else

$t \leftarrow anyElementOf(V2)$

$V2_0 \leftarrow \{t\}$

$V2_1 \leftarrow \{v \in V2 \mid \langle v, t \rangle \in E2\}$

$V2_2 \leftarrow \{v \in V2 \mid v \neq t \text{ and } v \notin V2_1 \text{ and } \exists u \in V2_1: \langle u, v \rangle \in E2\}$

$E2_2 \leftarrow \{\langle u, v \rangle \in E2 \mid u \in V2_2 \text{ and } v \in V2_2\}$

$g_2 \leftarrow \langle V2_2, E2_2 \rangle$

if $V2_2 = \{\}$ or *IsConnected*(g_2) then return *false*

else

$V'2 \leftarrow V2 - V2_0 - V2_1$

$E'2 \leftarrow \{\langle u, v \rangle \in E2 \mid u \in V'2 \text{ and } v \in V'2\}$

$G'2 \leftarrow \langle V'2, E'2 \rangle$

if not *IsConnected*($G'2$) then return *false*

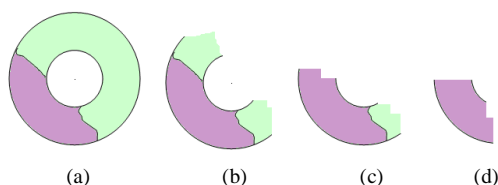
else return *true*

IsConnected($G \equiv \langle V, E \rangle$):

```
 $k \leftarrow \text{anyElementOf}(V)$   
 $L \leftarrow [k]$   
 $V \leftarrow V - \{k\}$   
while  $L \neq []$   
     $l \leftarrow L[0]$   
     $L \leftarrow L[1:]$   
    foreach  $m \in V$   
        if  $\langle l, m \rangle \in E$  then  
             $V \leftarrow V - \{m\}$   
             $L \leftarrow L + [m]$   
if  $V \neq \{\}$  then return false  
else return true
```

Trong thuật toán được đề xuất, đồ thị vùng lân cận 2 chặng G_2 được lưu trữ và $IsRing(G_2)$ được thực thi cục bộ tại mỗi nút. Sau hai vòng không đồng bộ, mỗi nút P biết topo của vùng lân cận bao gồm các nút cách xa P không quá hai chặng. G_2 được trích ra từ topo này. Hàm $IsRing(G_2)$ được gọi để kiểm tra xem G_2 có là một cái “vành” hay không. Để kiểm tra xem đồ thị vùng lân cận 2 chặng G_2 có tạo thành một cái vành hay không, đầu tiên tính liên thông của nó được kiểm tra bằng việc áp dụng thuật toán loang màu (gọi hàm $IsConnected(G_2)$): chọn một nút bất kỳ k thuộc G_2 , tô màu cho k và tất cả các nút thuộc G_2 liên thông với k . Nếu sau khi áp dụng thuật toán loang màu vẫn còn nút thuộc G_2 không được tô thì G_2 không liên thông; ngược lại G_2 liên thông. Rõ ràng rằng G_2 không thể là một cái vành nếu nó không liên thông. Trong trường hợp G_2 liên thông, G_2 có thể là một cái vành hoặc chỉ là một mảnh vỡ của cái vành. Để biết G_2 là cái vành hay chỉ là một mảnh vỡ, chúng ta cắt loại bỏ một phần của G_2 bao gồm nút bất kỳ t , các láng giềng của nó (V_{2_1}) và các láng giềng của láng giềng của nó (V_{2_2}), rồi kiểm tra các mảnh còn lại. Nếu chúng ta không thể cắt một phần đủ lớn (G_2 liên thông hoặc không có nút nào), thì G_2 đích thực không phải là một cái vành. Ngược lại, nếu chúng ta còn hai

mảnh (G_2 không liên thông), thì G_2 cũng không phải là một cái vành. G_2 là một cái vành chỉ khi chúng ta có duy nhất một mảnh còn lại vừa khớp với mảnh bị cắt và loại bỏ. Trực quan của việc cắt loại bỏ một mảnh và kiểm tra các mảnh còn lại của G_2 được minh họa trong Hình 3.3.



Hình 3.3. Minh họa thuật toán kiểm tra khả năng gần biên. Một cái vành hoặc một mảnh vỡ của cái vành sau khi được cắt và loại bỏ một phần. Phần được cắt và loại bỏ có màu tím. Các mảnh còn lại có màu xanh. (a) Một mảnh còn lại vừa với mảnh bị cắt và loại bỏ. (b) Hai mảnh còn lại. (c), (d) Không thể cắt và loại bỏ mảnh đủ lớn.

3.1.3 Đáp ứng với thay đổi mạng

Mỗi nút gửi lại danh sách láng giềng của nó khi có thay đổi đáng kể trong danh sách này, tạo và kiểm tra lại 2NG của nó mỗi khi nhận được một danh sách láng giềng mới từ nút cách xa hai chặng. Theo cách này, thuật toán được đề xuất có thể đáp ứng nhanh và hiệu quả với thay đổi mạng.

3.2 Phân tích và thử nghiệm

Gọi $adeg$ là bậc (số láng giềng) trung bình của các nút. Mỗi thông báo danh sách láng giềng có trung bình $adeg+1$ định danh (định danh thứ nhất là định danh nút tạo và gửi thông báo, các định danh còn lại là định danh của các láng giềng). Mỗi thông báo được phát tỏa một lần đến tất cả các láng giềng. Tại mỗi láng giềng, nó được phát tỏa một lần nữa. Như vậy, mỗi thông báo được phát tỏa trung bình $adeg+1$ lần. Mỗi nút chỉ tạo và gửi một thông báo danh sách láng giềng. Như vậy, tổng số thông báo được sử dụng cho một lần hội tụ tối đa là $N*(adeg+1)$, trong đó N là số nút trong mạng. Từ phân tích ở trên chúng ta thấy rằng mật độ nút càng thưa thì số thông báo phải sử dụng càng ít và kích thước các thông báo càng nhỏ, tức độ phức tạp truyền thông càng thấp. Tuy nhiên, nếu mật độ nút quá thưa thì thuật toán có thể cho kết quả không chính xác như được phân tích ở phía sau.

Về độ phức tạp tính toán, mỗi nút chỉ cần gọi hàm *IsRing* một lần khi nhận được một thông báo từ nút cách xa hai chặng. Mỗi nút có trung bình $adeg^2$ (đơn giản hóa cách tính) các nút cách xa hai chặng nên có thể thực hiện hàm *IsRing* $adeg^2$ lần cho mỗi lần hội tụ. Độ phức tạp của hàm *IsRing* là tuyến tính và nhỏ.

Về lưu trữ, mỗi nút phải lưu trung bình $adeg^2$ danh sách láng giềng, mỗi danh sách có trung bình $adeg+1$ định danh như đã được phân tích ở trên.

Thuật toán phát hiện biên được đề xuất có độ phức tạp thời gian là hằng số. Sau hai vòng không đồng bộ, mỗi nút có đầy đủ thông tin topo vùng lân cận hai chặng, không còn thông báo nào được gửi hay chuyên tiếp, và các nút hoàn thành quá trình tính toán.

Để kiểm chứng tính đúng của thuật toán, một chương trình thử nghiệm được phát triển cho phép tạo phân bố nút một cách ngẫu nhiên và cung cấp nhiều công cụ để quan sát mạng, tạo vùng trống, cũng như quan sát và đánh giá kết quả thu được. Thuật toán được đề xuất đã được thử nghiệm qua nhiều thể hiện mạng khác nhau. Phương pháp đánh giá tính đúng của thuật toán phát hiện biên được đề xuất trong [66] được áp dụng. *Nút biên thực* là nút nằm cạnh vùng trống truyền thông, tức là nút có vùng phủ sóng tiếp giáp với vùng trống truyền thông. Tính đúng của thuật toán phát hiện biên được đo bằng độ chính xác (precision) và độ hồi tưởng (recall). Các độ đo này được định nghĩa như sau: Gọi B_T là tập các nút biên thực. Ký hiệu B_A là tập các nút biên được phát hiện theo thuật toán. Độ hồi tưởng là tỉ lệ số nút biên thực được phát hiện, được tính theo công thức

$$recall = \frac{|B_T \cap B_A|}{|B_T|}$$

Độ chính xác là tỉ lệ số nút được phát hiện là nút biên thực, được tính theo công thức

$$precision = \frac{|B_T \cap B_A|}{|B_A|}$$

Các cấu hình mạng với mật độ nút khác nhau đã được thử nghiệm để đo độ chính xác và độ hồi tưởng của thuật toán đề xuất. Mỗi cấu hình mạng với một mật độ nút được thử

nghiệm trên một tập gồm 30 thể hiện mạng khác nhau. Kết quả độ chính xác và độ hồi tưởng được lấy trung bình trên mỗi tập, được thể hiện trong Bảng 3.2.

Bảng 3.2. Độ chính xác và độ hồi tưởng của thuật toán phát hiện biên được đề xuất qua thử nghiệm.

<i>Bậc trung bình của mạng</i>	<i>recall (%)</i>	<i>precision (%)</i>
7	98.16	18.89
10	98.57	24.74
13	98.72	30.04
16	98.79	32.87
19	99.28	35.28
21	99.33	37.53

Kết quả thử nghiệm cho thấy độ chính xác giảm nhanh khi mật độ nút giảm, trong khi độ hồi tưởng ít bị ảnh hưởng bởi mật độ nút. Kết quả này dễ giải thích vì 2NG của nút biên thực là cái vành bị vỡ bất kể mật độ nút cao hay thấp trong khi 2NG của một nút gần nút biên thực có xác suất trở thành cái vành bị vỡ cao hơn khi mật độ nút thấp hơn.

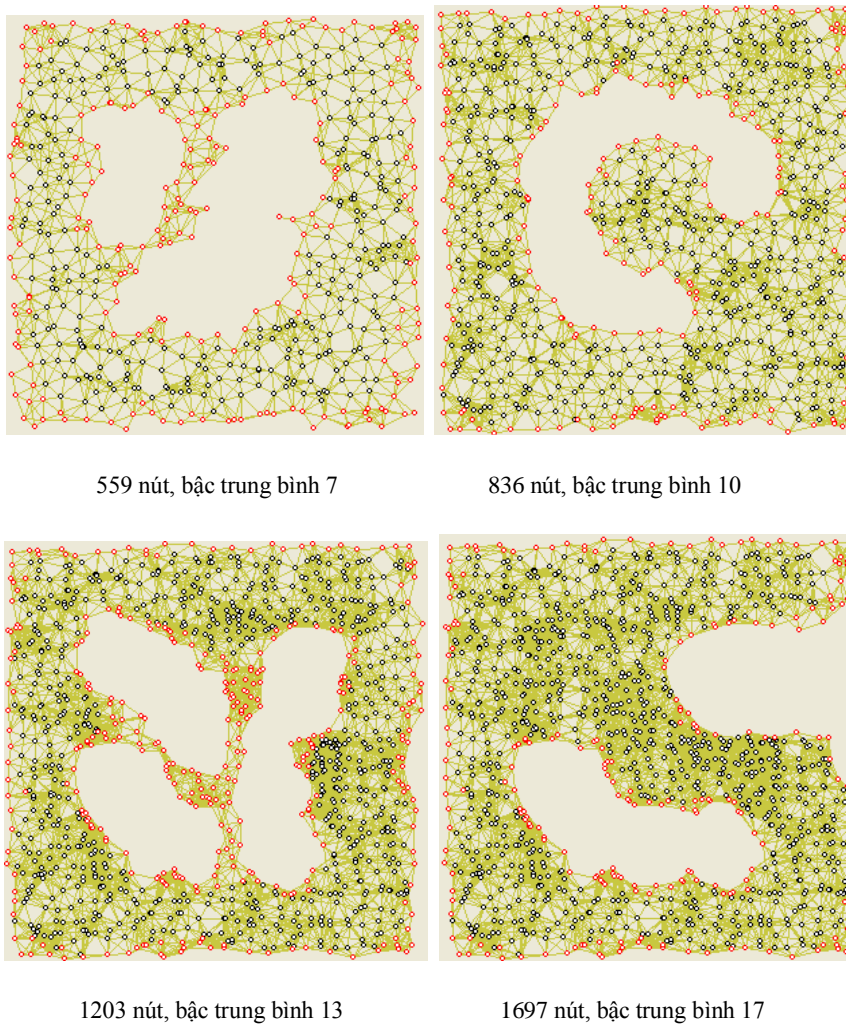
Để đánh giá tác động của mật độ nút đến khả năng phát hiện nút biên của thuật toán, các thể hiện mạng với mật độ nút khác nhau đã được sử dụng. Kết quả thử nghiệm cho thấy thuật toán được đề xuất làm việc tốt khi bậc trung bình 7 trở lên.

Để đánh giá tính ổn định của thuật toán được đề xuất, các thể hiện mạng với nhiều vùng trống có hình dạng phức tạp, cả lồi và lõm, gần nhau đã được sử dụng. Kết quả thử nghiệm cho thấy các nút biên được phát hiện chính xác. Một vài ví dụ về kết quả mô phỏng cho thấy tính ổn định của thuật toán được đề xuất được cho trong Hình 3.4.

3.3 So sánh với các thuật toán hiện có

Như đã được giới thiệu trong Mục 2.2, tới nay chỉ các thuật toán phát hiện biên trong [26, 29, 53, 91] thuộc nhóm dựa trên kết nối. Phân tích trong [26] cho thấy các thuật toán trong [29, 53, 91] chỉ áp dụng cho các mạng có phân bố đều và dày, kém hiệu quả hơn thuật toán trong [26]. Thuật toán trong [26] tồn kém vì phải giải quyết hai bài toán phức tạp là lựa chọn điểm mốc và phát tràn. Trong ngữ cảnh phân tán, bài toán lựa chọn điểm

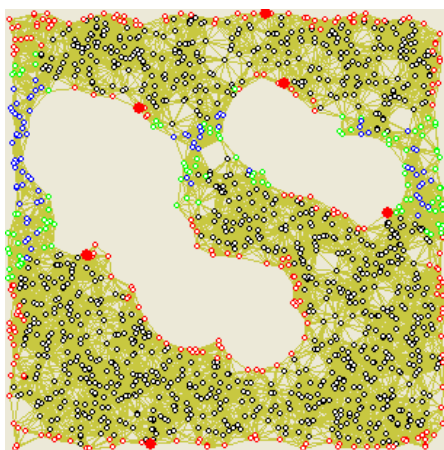
mốc chính là bài toán bầu thủ lĩnh. Thuật toán bầu thủ lĩnh có độ phức tạp thời gian $O(D)$ và độ phức tạp thông báo $O(adeq + N \log N)$ [3], với D là đường kính mạng. Thuật toán phát tràn có độ phức tạp thời gian $O(D)$ và độ phức tạp thông báo $O(N*adeq)$ [3]. Như vậy, thuật toán trong [26] có độ phức tạp thời gian và độ phức tạp thông báo cao hơn thuật toán được đề xuất. Kết quả thử nghiệm cho thấy độ chính xác và độ hồi tưởng của thuật toán trong [26] là khá thấp [66]. Ngoài ra, thuật toán trong [26] không làm việc tốt với mạng có mật độ nút dưới 18 [26]. Ngược lại, thuật toán phát hiện biên được đề xuất trong chương này có thể làm việc tốt trên các mạng có mật độ nút từ 7 trở lên.



Hình 3.4. Một vài kết quả thử nghiệm thuật toán phát hiện biên được đề xuất.

3.4 Thảo luận

Có thể chia các nút biên thành hai lớp là SB và MB . Lớp thứ nhất bao gồm các nút gần đúng một biên trong khi lớp thứ hai bao gồm các nút gần nhiều hơn một biên. Nhớ lại rằng trong thuật toán được đề xuất, mỗi nút kiểm tra $2NG$ của nó để xác định xem nó có gần biên hay không: Nếu $2NG$ của nó bao gồm (các) mảnh vỡ của một cái vành, thì nút thuộc biên. Quan sát kỹ hơn cho thấy *một nút thuộc SB nếu $2NG$ của nó chỉ bao gồm một mảnh vỡ (xem Hình 3.2-b), và thuộc MB nếu $2NG$ của nó bao gồm nhiều hơn một mảnh vỡ (xem Hình 3.2-c)*. Với quan sát này, chúng ta có thể tạo một thay đổi nhỏ trong thuật toán được đề xuất để xác định nút biên thuộc SB hay MB mà không tốn thêm bất kỳ chi phí nào. Ngoài ra, chúng ta cũng có thể tiếp tục phân nhỏ SB thành SB_1 và SB_2 , trong đó SB_2 bao gồm các nút thuộc SB có láng giềng thuộc MB . Điều này có thể được thực hiện đơn giản bằng việc cho mỗi nút MB phát tỏa một thông báo đến các láng giềng. Hình 3.5 minh họa các khái niệm MB , SB , SB_1 và SB_2 : Nút màu xanh lục thuộc MB , nút màu đỏ thuộc SB_1 và nút màu xanh lá cây thuộc SB_2 .



Hình 3.5. Phân hoạch các nút biên: MB bao gồm các nút màu xanh lục, SB_1 bao gồm các nút màu đỏ, và SB_2 bao gồm các nút màu xanh lá cây.

Các nút biên, cùng với liên kết giữa chúng, tạo thành một đồ thị tựa bản đồ giao thông: Mỗi thành phần liên thông thuộc SB_1 như một con đường trong khi mỗi thành phần liên thông MB hay SB_2 như một ngã rẽ nối các con đường với nhau. Đồ thị này có thể sẽ hữu ích trong việc định vị và khai thác các tính chất hình học ẩn trong topo mạng.

Thuật toán được đề xuất đã được phân tích và đánh giá độ phức tạp một cách chặt chẽ. Tuy nhiên, tính đúng của nó mới chỉ được kiểm chứng bằng mô phỏng. Do vậy, một hướng nghiên cứu trong tương lai có thể là đề xuất mô hình để có thể chứng minh tính đúng của thuật toán bằng lý thuyết.

Một vấn đề nữa liên quan đến thuật toán được đề xuất sẽ được tiếp tục nghiên cứu trong tương lai là phân biệt biên trong (của vùng trống) và biên ngoài (của vùng triển khai). Các vùng trống trong mạng cảm biến thường được tạo khi các nút trong một khu vực lớn bị hỏng do các sự cố như thiên tai, cháy nổ. Do đó, các ứng dụng theo dõi môi trường cần có khả năng khoanh vùng khu vực xảy ra sự cố, tức là các vùng trống. Trong các ứng dụng giám sát, chúng ta muốn biết kẻ đột nhập đã rời khỏi khu vực theo dõi hay chỉ “ẩn náu” trong các vùng trống đã được tạo ra. Bản đồ giao thông ở trên có thể phát huy tác dụng khi cần phân biệt biên trong và biên ngoài.

Thuật toán phát hiện biên được đề xuất cũng có thể được mở rộng để áp dụng cho các mạng cảm biến trong không gian ba chiều. Một cách trực quan, trong không gian ba chiều, đồ thị vùng lân cận 2 chặng của nút không gần biên tạo thành một “quả cầu rỗng”, tức quả cầu chỉ có lớp vỏ hình cầu nhưng rỗng bên trong, trong khi đồ thị vùng lân cận 2 chặng của nút gần biên bao gồm một hoặc nhiều mảnh vỡ của quả cầu rỗng.

Tóm lại, thuật toán phát hiện biên dựa trên kết nối được đề xuất là thuật toán đơn giản nhưng hiệu quả, là ứng viên tốt được sử dụng làm công đoạn đầu cho định vị trong mạng cảm biến không dây.

CHƯƠNG 4

TỐI ƯU HÓA ĐƯỜNG ĐI TRONG ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ

Trong chương này, giao thức tối ưu hóa đường đi có tên Greedy with Path Optimization Routing (GPOR) được đề xuất cho định tuyến đơn phát dựa trên thông tin vị trí trong mạng cảm biến không dây. Theo giao thức được đề xuất, các đường đi ban đầu được tìm bằng việc áp dụng chuyển tiếp tham lam và kỹ thuật đi theo biên, tiếp đó các đường tắt được tạo và sử dụng nhằm rút ngắn các đường đi, đồng thời tránh cực tiểu địa phương. Các đường đi được tối ưu hóa, nghĩa là được rút ngắn để chúng gần và tốt nhất là trở thành đường đi ngắn nhất. Đồng thời, các đường đi được đẩy ra xa biên, do vậy giảm tải cho các nút biên và đạt cân bằng tải tốt hơn. Giao thức tạo các phần tử định tuyến có thể áp dụng cho một vùng đích thay vì cho một nút đích như các giao thức thường thấy.

4.1 Đặt vấn đề

Như đã được thảo luận trong Chương 2, định tuyến đơn phát dựa trên thông tin vị trí kết hợp chuyển tiếp tham lam và kỹ thuật đi theo biên là một giải pháp đơn giản và ít tốn kém. Đi theo biên là kỹ thuật khôi phục hiệu quả. Tuy nhiên, định tuyến theo phương pháp này có hai yếu điểm chính. Thứ nhất, các đường đi dọc theo biên thường dài và không tối ưu. Thứ hai, nhiều đường đi dọc theo biên dẫn đến lưu lượng quá tải cho các nút biên. Điều này không chỉ dẫn đến tắc nghẽn tại biên khi có nhiều luồng lưu lượng đồng thời mà còn làm giảm nhanh tuổi thọ của các nút biên dẫn đến khoét rộng hơn các vùng trống.

Nhằm khắc phục các yếu điểm trên, nhiều kỹ thuật tối ưu hóa đường đi đã được đưa ra. Có thể chia các kỹ thuật này thành hai lớp là *đánh dấu nút* và *tạo đường tắt*.

Đánh dấu nút

Moraru và các cộng sự đã đề xuất hai kỹ thuật tối ưu hóa đường đi theo phương pháp đánh dấu nút là *Đánh dấu dựa vào hành vi* (Behavior Based Tagging - BBT) [68] và *Đánh dấu dựa vào vùng lân cận* (Neighbourhood Based Tagging - NBT) [69]. Trong đánh dấu dựa vào hành vi [68], mỗi nút đánh dấu nó có là tối ưu đối với đích hay không. Ban đầu, tất cả các nút đều đánh dấu chúng là tối ưu. Mỗi nút đếm số lần nó chuyển tiếp gói tin ở chế độ tham lam và số lần nó chuyển tiếp gói tin ở chế độ khôi phục. Nếu tỉ lệ giữa số lần chuyển tiếp gói tin ở chế độ khôi phục và số lần chuyển tiếp gói tin ở chế độ tham lam vượt quá một ngưỡng được định trước, nút đánh dấu nó là không tối ưu. Những nút đã đánh dấu không tối ưu sẽ không tham gia chuyển tiếp tham lam. Do vậy, các gói tin sau sẽ không được đưa vào vùng các nút không tối ưu được hình thành trước các cực tiểu địa phương. Trong đánh dấu dựa vào vùng lân cận [69], một nút được xem là tối ưu nếu nó có ít nhất một láng giềng tối ưu gần đích hơn. Khi chuyển tiếp gói tin ở chế độ tham lam, nút hiện tại xem xét các láng giềng tối ưu trước. Nếu có láng giềng tối ưu gần đích hơn, láng giềng tối ưu gần đích nhất sẽ được chọn làm nút chuyển tiếp tiếp theo. Ngược lại, nút hiện tại xét đến các láng giềng không tối ưu, nếu có láng giềng gần đích hơn thì láng giềng gần đích nhất được chọn làm nút chuyển tiếp tiếp theo. Như đã được chứng minh trong [69], NBT có tính ổn định trong khi BBT có thể ước lượng sai các nút không tối ưu.

Các kỹ thuật BBT và NBT cần nhiều thời gian để hình thành vùng bao gồm các nút không tối ưu, do đó nhiều gói tin sẽ không được hưởng lợi từ việc tối ưu đường đi. Một hạn chế nữa của các kỹ thuật đánh dấu này là chúng chỉ có thể được áp dụng cùng với định tuyến trên mặt.

Trong BBT và NBT, vùng các nút không tối ưu được hình thành một cách thụ động. Nói cách khác, việc xác định nút không tối ưu chỉ được thực hiện sau khi có lưu lượng đi qua nút. Một nút không có lưu lượng đi qua sẽ không bao giờ đánh dấu nó là không tối ưu cho dù nút đó có là cực tiểu địa phương chẳng nữa.

Một phương pháp đánh dấu khác, có tên là Dead-End, đã được đề xuất trong [39]. Trong Dead-End, các nút chủ động phát hiện nó có tối ưu hay không rồi đánh dấu nó.

Cũng như trong NBT, một nút được xem là tối ưu nếu nó có ít nhất một láng giềng tối ưu gần đích hơn. Vùng bao gồm các nút không tối ưu được gọi là dead-end. Cực tiểu địa phương là nút không tối ưu đầu tiên được phát hiện. Nó quảng bá tính không tối ưu của nó đến tất cả láng giềng. Khi nhận được thông báo không tối ưu, nút nhận đánh giá tính tối ưu của nó rồi quảng bá thông báo không tối ưu nếu nó trở nên không tối ưu. Thủ tục trên được lặp lại ở những nút khác cho đến khi tất cả các nút không tối ưu được phát hiện. Các nút tối ưu có láng giềng không tối ưu quảng bá thông báo “lối thoát”. Thông báo lối thoát được phát tràn toàn khu vực dead-end để mỗi nút không tối ưu có thể chọn một lối thoát cho mình. Giống như BBT, các láng giềng không tối ưu không được sử dụng trong chuyển tiếp tham lam. Không giống BBT và NBT, Dead-End chuyển tiếp gói tin ở chế độ khôi phục về các lối thoát để gói tin có thể thoát khỏi dead-end. Dead-End không phụ thuộc vào đồ thị phẳng. Thời gian hội tụ của nó đã được cho thấy ngắn hơn nhiều so với thời gian hội tụ của BBT và NBT.

Directional Dead-End, một mở rộng của Dead-End, được đề xuất để áp dụng cho mẫu lưu lượng tất-cả-đến-tất-cả. Phạm vi phát sóng của nút được chia thành nhiều hướng/cung. Tính tối ưu của nút được định nghĩa cho từng hướng. Cụ thể, một nút không tối ưu đối với hướng có đích nếu tất cả các láng giềng gần đích hơn đều không tối ưu đối với hướng này. Đầu tiên, nút hiện tại tính hướng d có đích, rồi tính lại tính tối ưu của nó đối với hướng d . Nếu nút hiện tại tối ưu đối với hướng d , nó chọn láng giềng tối ưu gần đích nhất. Ngược lại, nghĩa là nút hiện tại không tối ưu đối với hướng d , nó gửi gói tin đến lối thoát đối với hướng d . Việc ước lượng và sử dụng hướng trong Directional Dead-End không nhất quán với nhau. Một ví dụ cho thấy tính không nhất quán này được mô tả như sau. Gọi L là nút đầu tiên đánh dấu nó không tối ưu đối với hướng d đến đích. Quá trình ước lượng cho hướng d bắt đầu từ L và tỏa ra các nút khác. Gọi P là nút có hướng t đến đích và tham gia quá trình này. P ước lượng hướng d của nó. Hướng t đến đích của P không được ước lượng, và vẫn ở trạng thái tối ưu. Tiếp đó, Q , một láng giềng của P , tối ưu hóa đối với hướng đến đích và xa đích hơn P , chọn P vì P gần đích hơn và có hướng t đến đích tối ưu. Do đó, hướng d của P được ước lượng nhưng không được sử dụng trong khi hướng t của nó được sử dụng nhưng không được ước lượng.

Tạo đường tắt

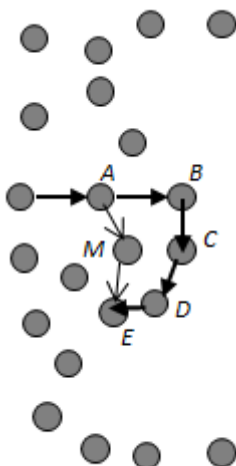
Tạo đường tắt là kỹ thuật tối ưu hóa đường đi khác đã được đề xuất. Đường đi tắt không chỉ có tác dụng tránh cực tiểu địa phương mà còn rút ngắn được đường đi và tăng cân bằng tải. Trong Beacon Path [51], khi gói tin ở chế độ khôi phục được chuyển về chế độ tham lam, một gói điều khiển được sinh ra và gửi theo một đường tắt vuông góc với đường thẳng hướng đến đích và chắn trước vùng trống. Các gói dữ liệu tiếp sau sẽ được chuyển theo đường tắt khi đường tắt được xây dựng thành công. Beacon Path có thể áp dụng với cả định tuyến trên mặt và đi theo biên. Nó đã được chứng minh là hiệu quả hơn các phương pháp đánh dấu. Tuy nhiên, Beacon Path có thể thất bại trong việc tạo đường tắt do có các vùng trống cắt đứt đường tắt. Thêm nữa, việc tạo đường tắt cần thời gian đáng kể nên nhiều gói tin trong luồng vẫn đi theo đường cũ.

Các phương pháp đánh dấu mất nhiều thời gian để hội tụ, phương pháp tạo đường tắt Beacon Path còn những hạn chế về khả năng tạo đường tắt và khai thác đường tắt. Xuất phát từ phân tích trên, một mục tiêu được thực hiện trong luận án là đề xuất một giao thức tối ưu hóa đường đi có thể tạo và khai thác hiệu quả các đường tắt, có thể áp dụng cho kịch bản có nhiều nút đích, và do vậy có thể áp dụng để nâng cao hiệu năng của định tuyến đơn phát dựa trên thông tin vị trí.

4.2 Mô tả giao thức

Một kỹ thuật tối ưu hóa đường đi theo phương pháp tạo đường tắt được đề xuất. Các đường đi ban đầu được khám phá nhờ sử dụng chuyển tiếp tham lam và đi theo biên. Tiếp đó, các đường tắt được tạo và thay thế cho các đoạn đường cong dài trong đường đi (xem Hình 4.1). Khi một nút phát hiện gói tin vừa đi qua một đoạn cong, tức nó là đầu cuối của đoạn cong đó, nó tạo ra gói tin tạo đường tắt SC và SC được chuyển đến đầu kia của đoạn cong bằng chuyển tiếp tham lam. Đường đi của SC tạo thành một đường tắt. Thời gian để tạo một đường tắt là ngắn, do đó đường tắt được khai thác ngay bởi các gói dữ liệu tiếp sau trong luồng. Hơn nữa, việc tạo đường tắt có chủ định giảm thất bại do vùng trống cắt đứt đường tắt khi chỉ tạo các đường tắt nối giữa hai nút đã biết không quá xa nhau. Để có thể áp dụng đường tắt cho nhiều luồng lưu lượng, *vùng khả áp dụng* (applicable area) của các phần tử đường tắt (chặng thuộc đường tắt) được đề xuất. Nói

cách khác, một phần tử đường tắt có thể áp dụng cho bất kỳ đích nào nằm trong vùng khả áp dụng của nó.



Hình 4.1. Một đoạn đường cong $A-B-C-D-E$ được thay thế bằng một đường tắt $A-M-E$.

GPOR sử dụng *chuyển tiếp có chỉ dẫn* (guided forwarding) dựa trên bảng định tuyến trước khi sử dụng chuyển tiếp tham lam. Các phần tử từ nhiều bảng định tuyến kết hợp lại tạo ra các đường tắt và được cập nhật bởi *kỹ thuật tạo đường tắt* (shortcut creation technique). Các mục tiếp theo lần lượt trình bày định dạng của bảng định tuyến, chuyển tiếp có chỉ dẫn, giao thức chuyển gói tin, kỹ thuật tạo đường tắt và những ưu điểm của GPOR.

4.2.1 Bảng định tuyến

Mỗi nút duy trì một bảng định tuyến với các phần tử có định dạng được mô tả trong Bảng 4.1. Một phần tử định tuyến được tạo khi một nút chuyển gói tin đến đích. Trường *pos* của nó được đặt giá trị là vị trí của đích, trường *next* được đặt giá trị là định danh nút chuyển tiếp tiếp theo, trường *lastlm* được đặt giá trị là vị trí của cực tiểu cuối cùng mà gói tin đi qua. Lưu ý rằng không giống định tuyến dựa trên topo, mỗi phần tử định tuyến của GPOR không cho một đích cụ thể, mà cho bất kỳ đích nào nằm trong vùng khả áp dụng của nó. Vùng khả áp dụng của phần tử định tuyến được mô tả trong mục tiếp theo.

Bảng 4.1. Định dạng của các phần tử định tuyến.

<i>Trường</i>	<i>Miêu tả</i>
<i>pos</i>	Vị trí đích
<i>next</i>	Định danh của láng giềng được chọn làm nút tiếp theo nếu phần tử định tuyến được sử dụng
<i>lastlm</i>	Vị trí của cực tiểu địa phương cuối

4.2.2 Vùng khả áp dụng của phần tử định tuyến

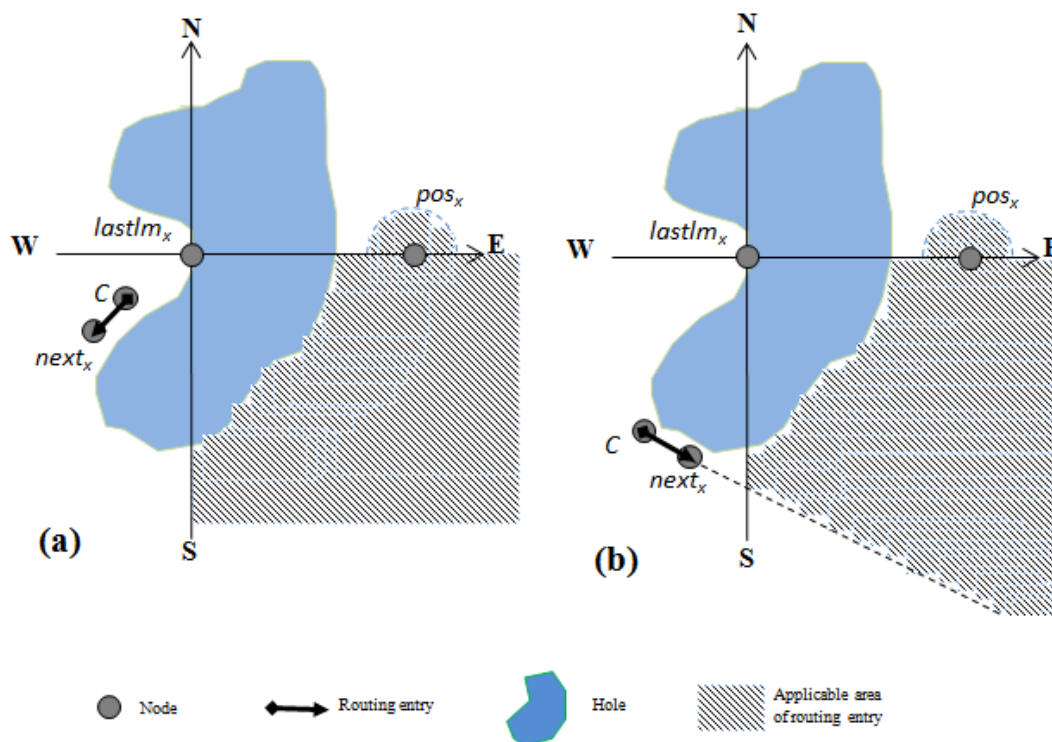
Với phần tử định tuyến $\langle pos_x, next_x, lastlm_x \rangle$, chúng ta định nghĩa một hệ tọa độ Đề-các có gốc tọa độ tại $lastlm_x$, trục hoành hướng từ $lastlm_x$ đến pos_x . Quan niệm trục hoành của hệ tọa độ vừa được định nghĩa có hướng từ Tây (W) sang Đông (E), mặt phẳng được chia thành bốn phần tư lần lượt là Đông-Bắc (NE), Đông-Nam (ES), Tây-Nam (SW) và Tây-Bắc (WN). Phần tử định tuyến $\langle pos_x, next_x, lastlm_x \rangle$ của nút C có thể áp dụng cho đích D nếu ít nhất một trong các điều kiện sau được thỏa mãn:

- *Điều kiện 1:* Khoảng cách từ D đến pos_x không lớn hơn r , trong đó r là bán kính vùng phủ sóng của mỗi nút.
- *Điều kiện 2:* D nằm trong phần tư ES, C nằm trong các phần tư SW hoặc WN, và $\overrightarrow{Cnext_x}$ chỉ hướng Tây, Tây-Bắc, Tây-Nam hoặc Đông-Bắc hoặc D nằm bên trái của $\overrightarrow{Cnext_x}$ nếu $\overrightarrow{Cnext_x}$ chỉ hướng Đông-Nam.⁶

Ý nghĩa của Điều kiện 1 rõ ràng và dễ hiểu. Gói tin có thể được chuyển theo một đường tốt đến pos_x , sau đó được chuyển trực tiếp từ pos_x đến D . Ý nghĩa của Điều kiện 2 được mô tả trực quan như sau. Đích D nằm trong phần tư ES, tức là phía sau vùng trống, trong khi nút hiện tại C thuộc khu vực phía Tây, tức là trước vùng trống. Do vậy, $\overrightarrow{Cnext_x}$ chỉ hướng Tây, Tây-Bắc, Tây-Nam hoặc Đông-Bắc cho biết gói tin đang trong vùng lổm và cần thoát khỏi khu vực này. Ngược lại, $\overrightarrow{Cnext_x}$ chỉ hướng Đông-Nam cho biết gói tin vừa thoát khỏi vùng lổm, tuy nhiên D nằm bên trái của nếu $\overrightarrow{Cnext_x}$ cho biết vùng trống vẫn còn chặn đường đi thẳng từ nút hiện tại đến đích. Trong cả hai tình huống trên, gói tin nên đi theo đường đi tốt của các gói trước nhằm tránh và đi qua vùng trống.

⁶ Giả thiết quy tắc bàn tay phải (chiều kim đồng hồ) được áp dụng trong định tuyến khôi phục.

Hình 4.2 cho chúng ta hai ví dụ về vùng khả áp dụng của phần tử định tuyến. Trong ví dụ thứ nhất, vectơ $\overrightarrow{Cnext_x}$ chỉ hướng Tây-Nam, vùng khả áp dụng của phần tử định tuyến bao gồm phần tử ES và các điểm cách pos_x không quá r . Trong ví dụ thứ hai, $\overrightarrow{Cnext_x}$ chỉ hướng Đông-Nam, vùng khả áp dụng của phần tử định tuyến bao gồm các điểm thuộc phần tử ES và bên trái $\overrightarrow{Cnext_x}$ cùng các điểm cách pos_x không quá r .



Hình 4.2. Vùng khả áp dụng của phần tử định tuyến $\langle pos_x, next_x, lastlm_x \rangle$ của nút C :

(a) $\overrightarrow{Cnext_x}$ chỉ hướng Tây-Nam; (b) $\overrightarrow{Cnext_x}$ chỉ hướng Đông-Nam.

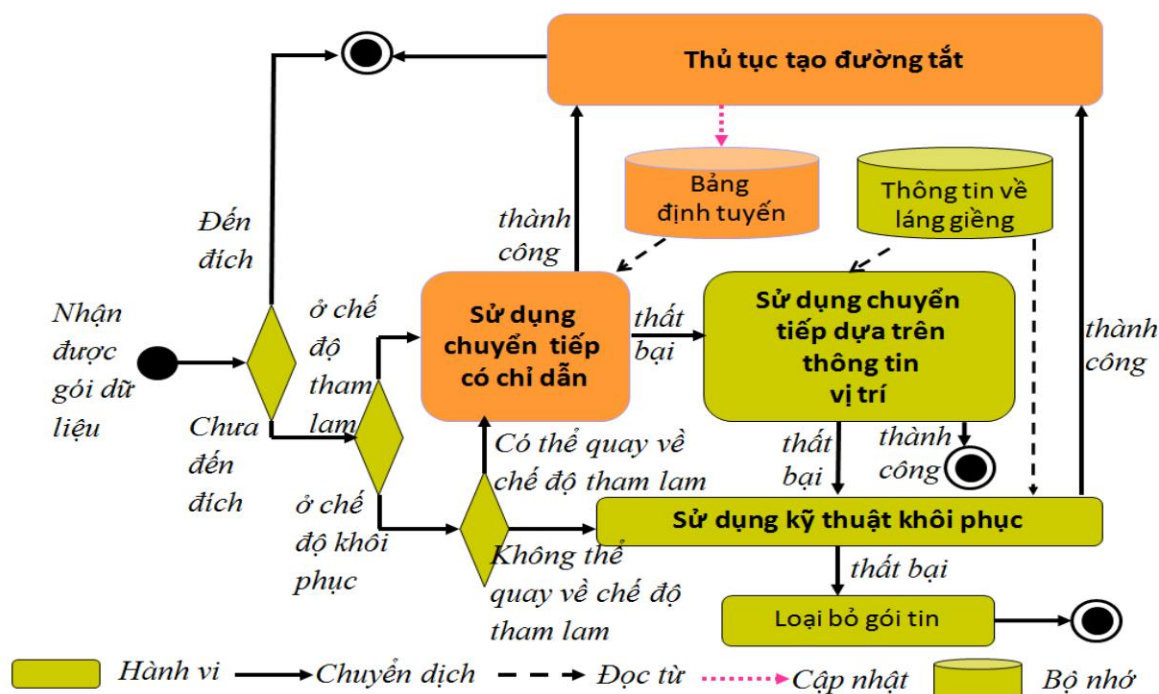
4.2.3 Chuyển tiếp có chỉ dẫn

Chuyển tiếp có chỉ dẫn (guided forwarding) được mô tả như sau: Nút hiện tại tìm trong bảng định tuyến của nó phần tử định tuyến có thể áp dụng cho đích của gói tin và có pos gần đích của gói tin nhất. Tiếp theo, nó chuyển gói tin cho láng giềng có định danh $next$ của phần tử được chọn.

Trong các mục tiếp theo, chuyên tiếp có chỉ dẫn được sử dụng trong định tuyến kết hợp chuyển tiếp tham lam và kỹ thuật đi theo biên, việc duy trì các bảng định tuyến sẽ lần lượt được trình bày.

4.2.4 Định tuyến và cập nhật bảng định tuyến

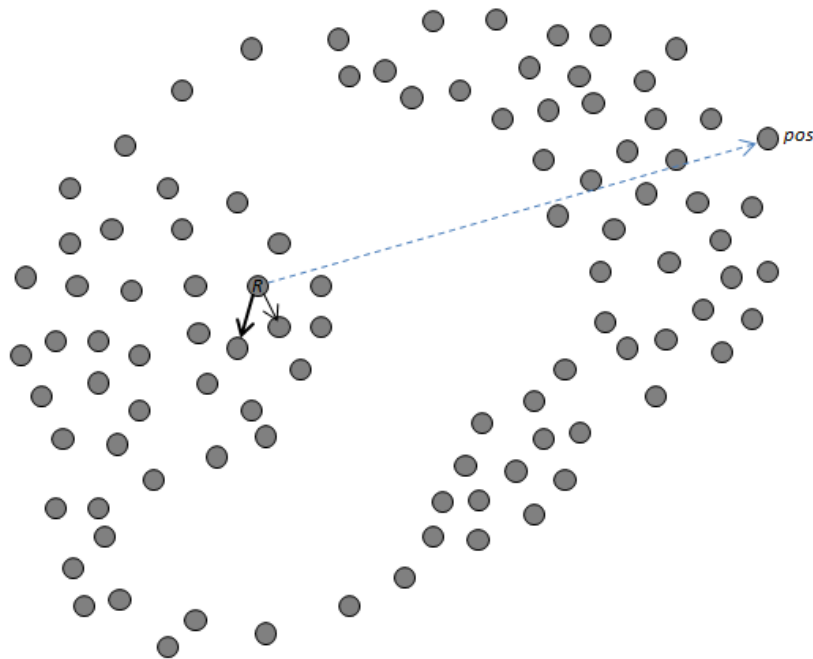
Giống như các giao thức định tuyến dựa trên thông tin vị trí khác, GPOR (xem Hình 4.3 và Bảng 4.2) có hai chế độ chuyển tiếp gói tin là tham lam và khôi phục. Khác với các giao thức định tuyến dựa trên thông tin vị trí khác, GPOR chuyển gói tin ở chế độ tham lam bằng chuyển tiếp có chỉ dẫn trước, nếu chuyển tiếp có chỉ dẫn thất bại gói tin mới được chuyển bằng chuyển tiếp tham lam. Kỹ thuật đi theo biên được sử dụng cho gói tin ở chế độ khôi phục. Gói tin được sinh ra ở chế độ tham lam, được chuyển sang chế độ khôi phục tại cực tiểu địa phương tại đó chuyển tiếp có chỉ dẫn thất bại, và được chuyển về chế độ tham lam tại nút gần hơn cực tiểu địa phương cuối tại đó chuyển tiếp có chỉ dẫn thất bại.



Hình 4.3. Hành vi của mỗi nút cảm biến trong GPOR.

Kỹ thuật tạo đường tắt được sử dụng để cập nhật các bảng định tuyến. Theo kỹ thuật này, k nút cuối P_k, P_{k-1}, \dots, P_1 và cực tiểu cuối cùng $lastlm$ mà gói tin đi qua được ghi trong tiêu đề của gói tin, đồng thời các gói tin tạo đường tắt SC (shortcut creation) được

sử dụng. Sau khi chuyển gói tin có đích là D tới láng giềng N bằng kỹ thuật đi theo biên hay chuyển tiếp có chỉ dẫn, nút hiện tại kiểm tra liệu đoạn $pseg$ được tạo bởi k nút vừa đi qua có là một đoạn cong hay không. Nếu đúng như vậy, nút hiện tại (1) thêm phần tử $\langle D.pos, N, lastlm \rangle$ vào bảng định tuyến của nó, (2) tạo một gói SC $\langle SC, P_k, D.pos, lastlm \rangle$ có đích là đầu kia của đoạn cong (P_k), rồi gửi gói SC đến đích của nó bằng chuyển tiếp tham lam. Tính cong của đoạn $pseg$ có thể dễ dàng được xác định bằng cách kiểm tra liệu nút đầu tiên trên đường chuyển tiếp tham lam từ nút hiện tại đến P_k không phải là nút liền trước (P_1) và tất cả các nút $P_i, i = 1, \dots, k-1$, đều nằm bên phải của $\overrightarrow{CP_k}$. Gói SC được chuyển đến đích của nó chỉ bằng chuyển tiếp tham lam. Khi nhận được gói tin điều khiển $\langle SC, T, pos, lastlm \rangle$ từ láng giềng Q , nút nhận được gói SC, R , hoặc thêm phần tử $\langle pos, Q, lastlm \rangle$ vào bảng định tuyến của nó nếu trong bảng định tuyến không có phần tử có pos , hoặc thay thế phần tử có pos bằng phần tử $\langle pos, Q, lastlm \rangle$ nếu vectơ \overrightarrow{Rpos} sẽ gặp vectơ \overrightarrow{RO} trước vectơ \overrightarrow{RQ} khi quay \overrightarrow{Rpos} theo chiều kim đồng hồ, trong đó O là nút có định danh $next$ của phần tử định tuyến cũ được thay thế. Việc thay thế phần tử định tuyến có pos đảm bảo rằng \overrightarrow{Rnext} được quay theo chiều kim đồng hồ xuất phát từ \overrightarrow{Rpos} . Ý nghĩa của nguyên tắc thay thế này là \overrightarrow{Rnext} ngày càng cho hướng tốt hơn để gói tin thoát khỏi vùng lổm trước vùng trống. Hình 4.4 minh họa trực quan ý nghĩa này.



Hình 4.4. Phần tử định tuyến mới (mũi tên nét đậm) có hướng thoát khỏi vùng lổm trước vùng trống tốt hơn phần tử định tuyến đã có (mũi tên nét mảnh).

Để tiết kiệm bộ nhớ, mỗi nút chỉ lưu trữ nhiều nhất một phần tử định tuyến cho một vị trí đích. Ngoài ra, phần tử định tuyến cũ có thể bị loại bỏ nếu không còn đủ bộ nhớ để lưu trữ.

Bảng 4.2. Hành vi của mỗi nút cảm biến trong GPOR.

Khi nhận được gói dữ liệu:

Nếu tôi là đích của gói tin

Gửi gói tin cho tầng trên (giao vận)

Ngược lại

Cập nhật danh sách k chặng cuối P_k, P_{k-1}, \dots, P_1 mà gói tin vừa đi qua trong tiêu đề của gói tin.

Nếu gói tin ở chế độ tham lam

Gọi **Guided-Greedy**

Ngược lại

Gọi **Recovery**

Khi nhận được gói tạo đường tắt $\langle SC, T, pos, lastlm \rangle$ từ láng giềng Q :

Thêm hoặc cập nhật $\langle pos, Q, lastlm \rangle$ vào bảng định tuyến.

Chuyển tiếp gói SC đến T bằng chuyển tiếp tham lam.

Guided-Greedy:

Nếu có phần tử trong bảng định tuyến có thể áp dụng cho đích của gói tin

Loại bỏ tất cả các phần tử cho $D.pos$ rồi thêm $\langle D.pos, N, lastlm \rangle$ vào bảng định tuyến, trong đó D là đích của gói tin và N là nút kế tiếp được xác định bởi phần tử định tuyến được sử dụng.

Chuyển gói tin đến N

Nếu **Is-A-Contour-Segment**(P_k, P_{k-1}, \dots, P_1)

Gửi $\langle SC, P_k, D.pos, lastlm \rangle$ đến P_k .

Ngược lại

Chuyển gói tin bằng chuyển tiếp tham lam.

Nếu chuyển tiếp tham lam thất bại

Ghi tọa độ của tôi vào tiêu đề gói tin xem như cực tiểu địa phương cuối

Đặt gói tin vào chế độ khôi phục

Gọi **Recovery**

Recovery:

Nếu tôi gần đích hơn cực tiểu địa phương cuối mà gói tin đi qua

Đặt gói tin về chế độ tham lam

Gọi **Guided-Greedy**

Ngược lại

Loại bỏ tất cả các phần tử định tuyến cho $D.pos$ rồi thêm $\langle D.pos, N, lastlm \rangle$ vào bảng định tuyến, trong đó D là đích của gói tin và N là nút tiếp theo được chọn bởi kỹ thuật khôi phục.

Chuyển gói tin cho N .

Nếu **Is-A-Contour-Segment**(P_k, P_{k-1}, \dots, P_1)

Gửi $\langle SC, P_k, D.pos, lastlm \rangle$ đến P_k .

Is-A-Contour-Segment(P_k, P_{k-1}, \dots, P_1):

Nếu không có láng giềng gần P_k hơn

return **false**

Nếu P_1 là láng giềng gần P_k nhất

return **false**

Với mỗi $P_i, i = 1, \dots, k-1$

Nếu P_i không nằm bên phải của $\overrightarrow{CP_k}$

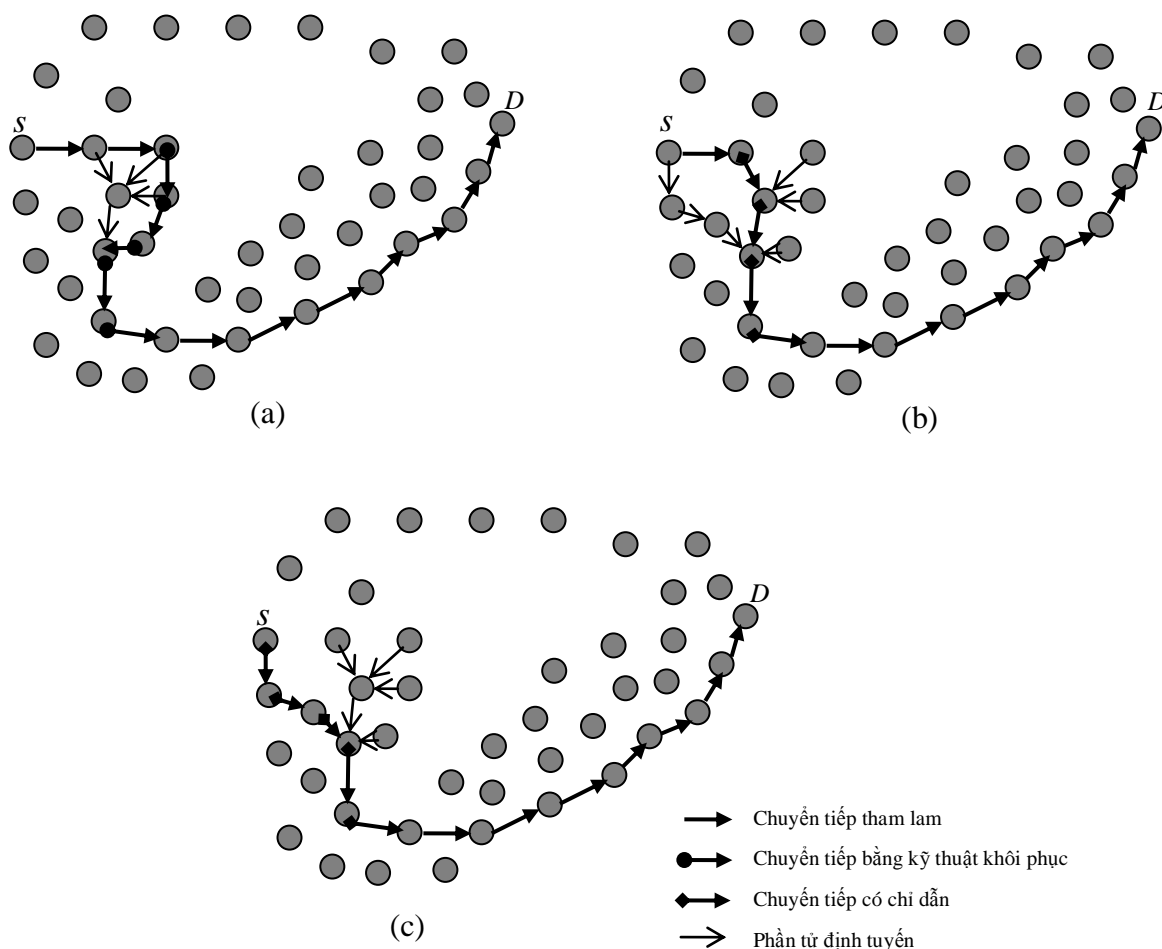
return **false**

return **true**

4.3 Ưu điểm

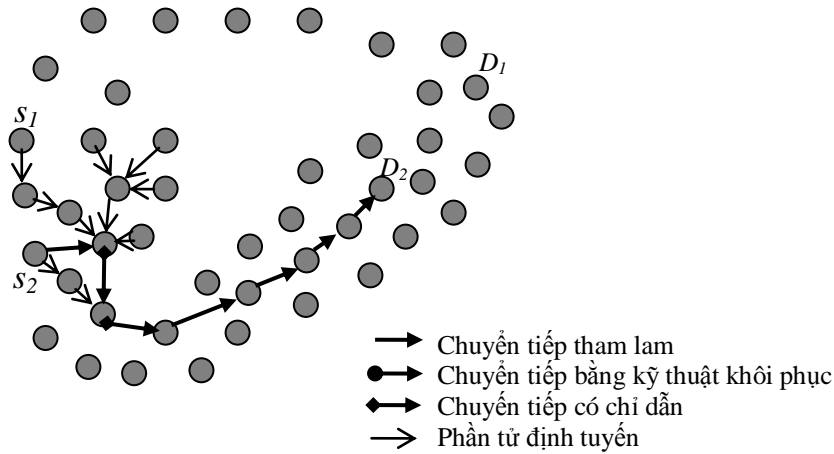
Đường đi được rút ngắn (tối ưu hóa) là ưu điểm chính của GPOR. Tối ưu hóa đường đi không chỉ thu được từ các luồng lưu lượng đơn lẻ mà còn từ nhiều luồng lưu lượng đồng thời. Hình 4.5 cho một ví dụ về tối ưu hóa đường đi bởi một luồng lưu lượng. Trong ví

dụ này, đường đi gần tối ưu được tạo khi ba gói dữ liệu và các gói điều khiển SC tương ứng hoàn thành hành trình của chúng.



Hình 4.5. Đường đi được tạo bởi (a) một gói dữ liệu và các gói SC kèm theo, (b) hai gói dữ liệu và các gói SC kèm theo, (c) ba gói dữ liệu và các gói SC kèm theo.

Ở ví dụ trong Hình 4.6, hai luồng lưu lượng đồng thời được thiết lập, một luồng từ S_1 đến D_1 , luồng còn lại từ S_2 đến D_2 . Vùng khả áp dụng của các phần tử định tuyến phát huy tác dụng. Phần tử định tuyến áp dụng được cho D_1 cũng áp dụng được cho D_2 . Do vậy, đường đi từ S_2 đến D_2 có thể tránh được cực tiểu địa phương và được rút ngắn ngay từ đầu.



Hình 4.6. Tối ưu hóa đường đi và tránh cực tiểu địa phương bởi các luồng lưu lượng đồng thời. Bằng việc áp dụng các phần tử định tuyến cho D_1 , đường đi từ S_2 đến D_2 tránh cực tiểu địa phương thành công.

Ưu điểm thứ hai của GPOR là nó cho khả năng tránh cực tiểu địa phương. Điều này được minh họa ở ví dụ trong Hình 4.6. Trong ví dụ này, mọi gói tin có đích gần D_1 hoặc D_2 đi qua các nút có băng định tuyến không rộng sẽ tránh được cực tiểu địa phương. Ưu điểm thứ ba của GPOR là khả năng phân tải. Các đường đi dần dần được đẩy ra xa biên do vậy giảm tải cho các nút biên. Nhờ các ưu điểm thứ nhất và thứ ba, ưu điểm thứ tư của GPOR là cho trễ đầu cuối – đầu cuối thấp hơn. Nếu kỹ thuật tối ưu hóa đường đi không được áp dụng, không những đường đi dài mà lưu lượng còn tập trung nhiều ở các nút biên dẫn đến các gói tin phải xếp hàng lâu tại các nút này. Một số nút biên quá tải có thể loại bỏ gói tin. Do đó, ưu điểm thứ năm của GPOR chính là làm tăng tỉ lệ chuyển gói tin đến đích thành công.

4.4 Phân tích và so sánh với các giao thức khác

Giả sử $p = [V_1, V_2, \dots, V_n]$ là một đường đi từ nguồn đến đích được tìm ra bởi GPOR. Dễ dàng nhận thấy từ giao thức GPOR là nếu p không chứa đoạn cong (theo thủ tục *Is-A-Contour-Segment*) nào thì giao thức không gọi thủ tục tạo đường tắt, do đó không tạo các gói tin điều khiển SCP, và p không thay đổi. Nếu p chỉ chứa các đoạn cong không tồn tại đường đi theo chuyển tiếp tham lam giữa hai đầu đoạn cong thì thủ tục tạo đường tắt thất bại, p không thay đổi, trong khi một lượng nhỏ các gói SCP vẫn được tạo và chuyển tiếp.

Ngược lại, nếu $c = [V_i, V_{i+1}, \dots, V_{i+k-1}]$ là đoạn cong đầu tiên được phát hiện trong p mà tồn tại đường đi theo chuyển tiếp tham lam từ V_{i+k-1} đến V_i , thủ tục tạo đường tắt thành công và c sẽ được thay thế bởi đường tắt có ít chặng hơn nó, tức p được rút ngắn thành đường đi p' với $len(p') < len(p)$, trong đó $len(.)$ là độ dài đường đi. Do đường đi không thể rút ngắn được mãi nên mạng sẽ trở về trạng thái ổn định, hay trạng thái mà các đường đi từ nguồn đến đích không thay đổi nữa. Có thể sử dụng chứng minh tương tự như trong [51] để tính được tỷ lệ độ dài được đi được rút ngắn bởi GPOR.

Các giao thức trong [39, 68] không mất chi phí nhưng mất nhiều thời gian để tạo ra vùng các nút không tối ưu. Trong khi vùng các nút không tối ưu đang được hình thành, rất nhiều gói dữ liệu đã đi vào vùng này và theo những đường đi dài không hiệu quả. Ngoài ra, như đã được chỉ ra trong [39, 68], các kỹ thuật đánh dấu này chỉ có thể được áp dụng cùng với định tuyến trên mặt. Phương thức tạo đường tắt trong [51] có thể áp dụng với cả định tuyến trên mặt và đi theo biên nhưng dễ thất bại do các vùng trống làm gián đoạn đường đi tham lam của gói tin tạo đường tắt. Khi thủ tục tạo đường tắt thất bại, đường đi không được rút ngắn và tất cả các gói tin đều phải đi theo đường này [51]. Một vấn đề nữa là đường tắt chỉ phát huy tác dụng khi nó được xây dựng hoàn chỉnh, tuy nhiên, thời gian xây dựng đường tắt tương đối dài, dẫn đến nhiều gói tin vẫn đi theo đường cũ không hiệu quả. Ngoài ra, đường tắt được tạo theo [51] chỉ áp dụng được cho một đích đơn lẻ. Ngược lại, giao thức tối ưu hóa được đề xuất có thể áp dụng với cả định tuyến trên mặt và đi theo biên, ít thất bại trong việc tạo ra các đường tắt; các đường tắt được xây dựng nhanh và sớm phát huy tác dụng, có thể áp dụng cho nhiều đích khác nhau.

4.5 Mô phỏng

GPOR đã được cài đặt và tiến hành mô phỏng với bộ mô phỏng mạng mã nguồn mở ns-2 [108]. BOUNDHOLE [20] được sử dụng để tìm các đường đi ban đầu. Rất nhiều mô phỏng đã được thực hiện. Hiệu năng của GPOR được đánh giá ở bốn độ đo chịu tác động nhiều nhất của việc tạo và sử dụng đường tắt: tỷ lệ kéo dài độ dài đường đi, trễ đầu cuối – đầu cuối, tỷ lệ chuyển gói tin đến đích thành công, và độ phức tạp thông báo. Tỷ lệ kéo dài độ dài đường đi là tỷ lệ giữa trung bình độ dài các đường đi của các gói tin đến đích thành công trên độ dài trung bình của các đường đi tối ưu giữa nguồn và đích của các gói

tin này (ns-2 cung cấp cho chúng ta thông tin về đường đi tối ưu giữa mọi cặp nút). Trễ đầu cuối – đầu cuối được ước lượng bằng trung bình số đơn vị thời gian để gói tin đi từ nguồn đến đích. Độ phức tạp thông báo được ước lượng bằng tổng số phát tỏa cả các gói điều khiển và gói dữ liệu (trừ các gói chào hỏi). Mặc dù độ phức tạp thông báo thông thường được đánh giá bằng số (bits của) thông báo điều khiển, tổng số phát tỏa được chủ định sử dụng vì nó phản ánh sát mức tiêu thụ năng lượng ở các nút. Để ước lượng độ phức tạp thông báo dễ dàng, chúng ta giả thiết kích thước các gói tin là bằng nhau, mặc dù kích thước gói điều khiển rất nhỏ so với gói dữ liệu. Tiếp đó, kết quả mô phỏng của GPOR được so sánh với BOUNDHOLE. Lưu ý rằng GPOR có thể sử dụng kỹ thuật khôi phục khác thay vì đi theo biên. Mục đích của mô phỏng là ước lượng xem kỹ thuật tối ưu hóa đường đi cải thiện hiệu năng của giao thức định tuyến cơ bản (kết hợp chuyển tiếp tham lam và định tuyến khôi phục) đến mức nào. Do vậy, chúng ta chỉ cần so sánh hiệu năng của GPOR với hiệu năng của BOUNDHOLE.

Bảng 4.3. Cấu hình mạng với kích thước vùng triển khai mạng khác nhau.

<i>Số nút</i>	<i>Kích thước vùng triển khai</i>	<i>Mật độ nút</i>	<i>Số luồng CBR</i>
125	750 m × 1500 m	1 node / 9000 m ²	30
167	1000 m × 1500 m	1 node / 9000 m ²	30
208	1250 m × 1500 m	1 node / 9000 m ²	30
250	1500 m × 1500 m	1 node / 9000 m ²	30
292	1725 m × 1500 m	1 node / 9000 m ²	30
333	2000 m × 1500 m	1 node / 9000 m ²	30

Nhằm đạt được các mục tiêu đánh giá hiệu năng, các mô phỏng được tiến hành trên các cấu hình mạng với số luồng lưu lượng đồng thời thay đổi và kích thước mạng thay đổi. Kích thước mạng có ảnh hưởng đến hiệu quả tạo và sử dụng đường tắt vì mạng càng rộng, đường đi càng dài càng có nhiều cơ hội để tạo và sử dụng đường tắt. Các thể hiện mạng với số luồng lưu lượng đồng thời thay đổi được sử dụng nhằm đánh giá tác dụng của vùng khả áp dụng của các phần tử định tuyến. Các nút có bán kính phát sóng 250m

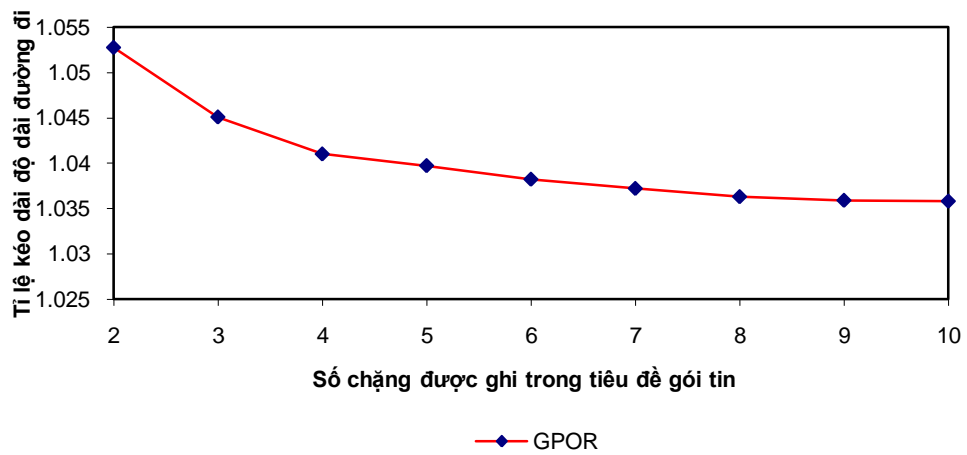
và chu kỳ phát gói tin chào hỏi là 1 giây mô phỏng. UDP được sử dụng ở tầng giao vận. Công nghệ không dây IEEE 802.11 WaveLAN được sử dụng ở tầng MAC. Ăngten đẳng hướng (omni) với mô hình lan truyền tín hiệu Two-Ray Ground được sử dụng cho mô hình truyền tín hiệu không dây. Mỗi mô phỏng kéo dài 1500 giây mô phỏng và sử dụng các lưu lượng CBR các gói tin kích thước 512 bytes với tốc độ 5 Kbps. Mỗi tập các mô phỏng (xác định bởi một kích thước mạng và một số lượng các lưu lượng) bao gồm 6 mô phỏng. Giá trị trung bình trên tập mô phỏng này được lấy và phân tích. Bảng 4.3 và 4.4 tóm tắt các cấu hình mạng của các mô phỏng.

Bảng 4.4. Cấu hình mạng với số luồng lưu lượng khác nhau.

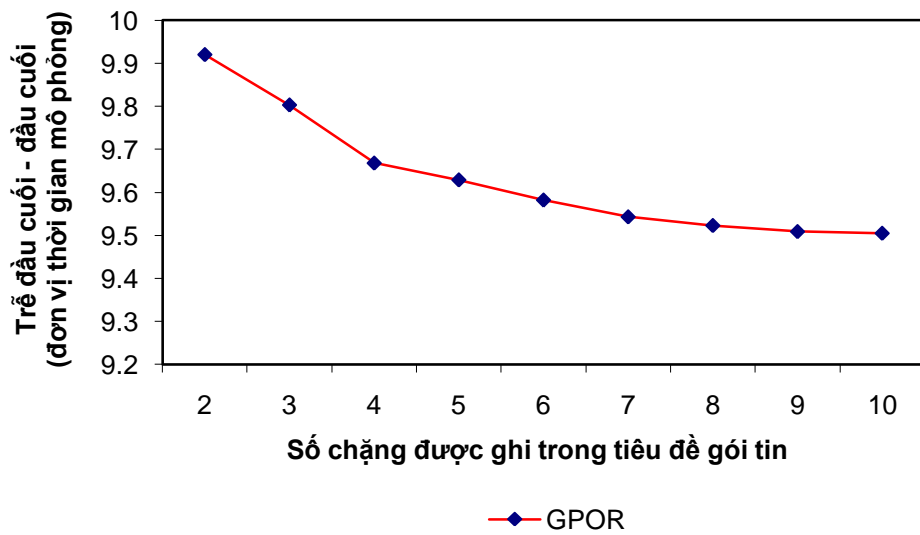
<i>Số nút</i>	<i>Kích thước vùng triển khai</i>	<i>Mật độ nút</i>	<i>Số luồng CBR</i>
250	1500 m × 1500 m	1 node / 9000 m ²	20
250	1500 m × 1500 m	1 node / 9000 m ²	30
250	1500 m × 1500 m	1 node / 9000 m ²	40
250	1500 m × 1500 m	1 node / 9000 m ²	50
250	1500 m × 1500 m	1 node / 9000 m ²	60
250	1500 m × 1500 m	1 node / 9000 m ²	70

4.5.1 Lựa chọn số chặng được ghi

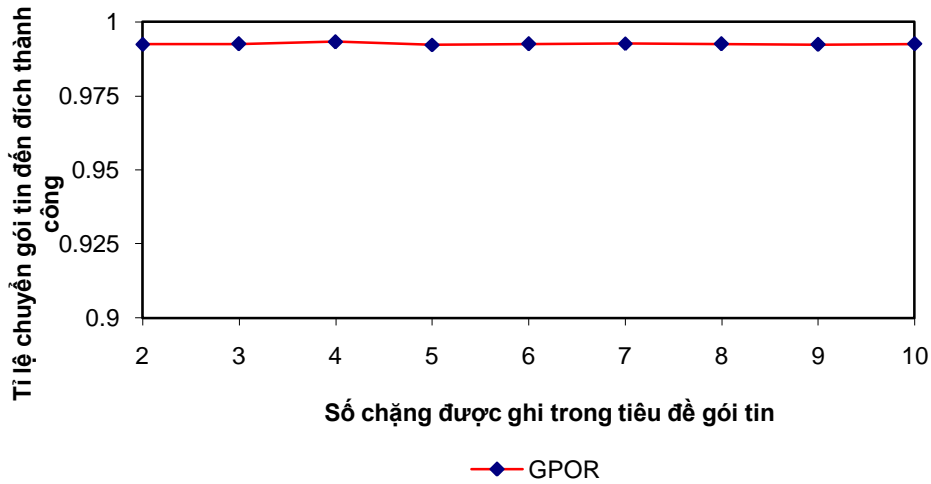
Để đặt tham số k , là số chặng được ghi trong tiêu đề gói tin, tức là độ dài các đoạn cong được phát hiện và thay thế, một cách tối ưu, các mô phỏng với k thay đổi đã được tiến hành. Cụ thể, k được thay đổi từ 2 đến 10 với các mô phỏng trên cấu hình mạng gồm 250 nút, 30 lưu lượng CBR và kích thước mạng 1500m × 1500 m. Kết quả mô phỏng cho thấy tỷ lệ kéo dài độ dài đường đi (Hình 4.7) và trễ đầu cuối – đầu cuối (Hình 4.8) giảm nhanh khi k tăng từ 2 đến 5, và giảm chậm khi k tiếp tục tăng. Kết quả mô phỏng cũng cho thấy số chặng được ghi k ít ảnh hưởng đến tỷ lệ chuyển gói tin đến đích thành công (Hình 4.9), nhưng chi phí truyền thông tăng lên đáng kể khi k tăng (Hình 4.10). Từ kết quả mô phỏng, giá trị sử dụng cho k nên là 4, 5 và 6.



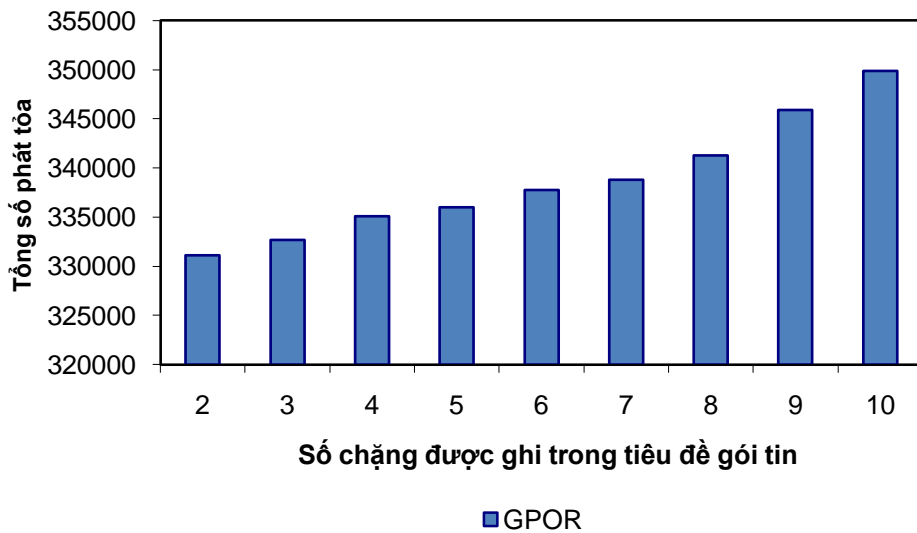
Hình 4.7. Tỷ lệ kéo dài độ dài đường đi với số chặng được ghi khác nhau.



Hình 4.8. Trung bình trễ đầu cuối – đầu cuối với số chặng được ghi khác nhau.



Hình 4.9. Tỷ lệ chuyển gói tin đến đích thành công với số chặng được ghi khác nhau.

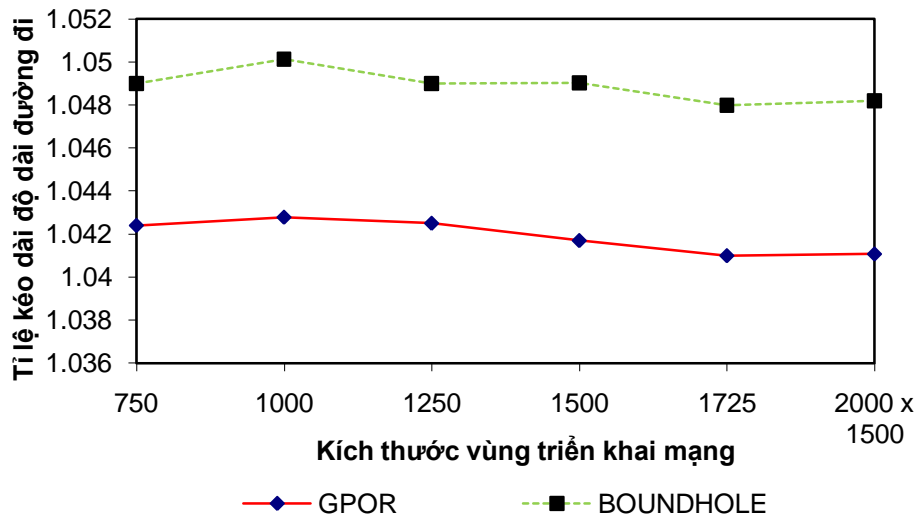


Hình 4.10. Tổng số phát tủa với số chặng được ghi khác nhau.

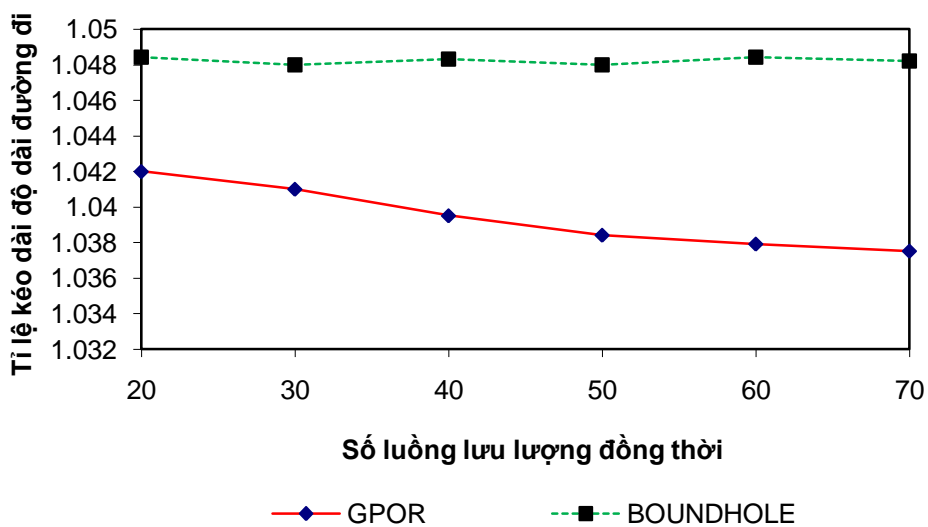
4.5.2 Tỷ lệ kéo dài độ dài đường đi

Kết quả mô phỏng được thể hiện trong các Hình 4.11 và 4.12 cho thấy đường đi được rút ngắn như thế nào khi áp dụng kỹ thuật tối ưu hóa đường đi được đề xuất, tức GPOR tạo đường đi ngắn hơn đường đi của BOUNDHOLE như thế nào. Đường đi của GPOR ngắn hơn vì chúng sử dụng các đường tắt. Trong GPOR, khi số lưu lượng đồng thời tăng lên,

tỷ lệ rút ngắn độ dài đường đi thu được càng cao. Kết quả mô phỏng này chứng tỏ tác dụng của vùng khả áp dụng của các phần tử định tuyến.



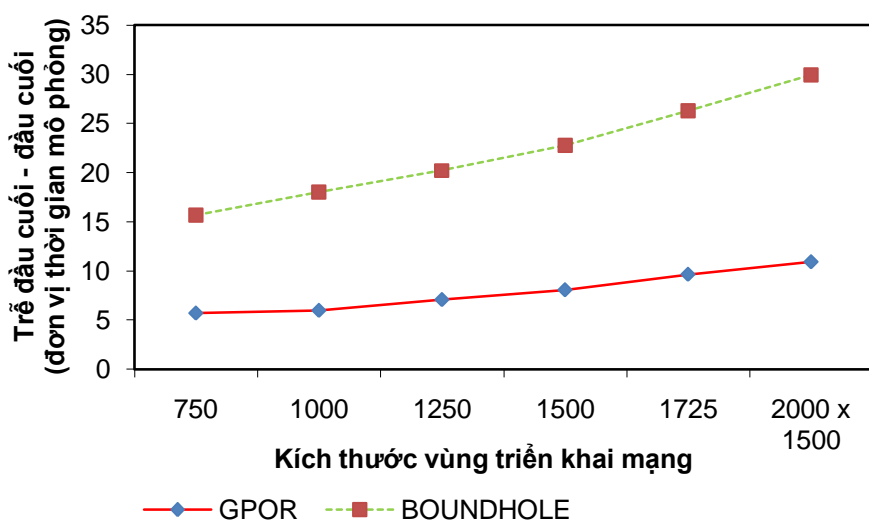
Hình 4.11. Tỷ lệ kéo dài độ dài đường đi với kích thước mạng khác nhau ($k = 4$).



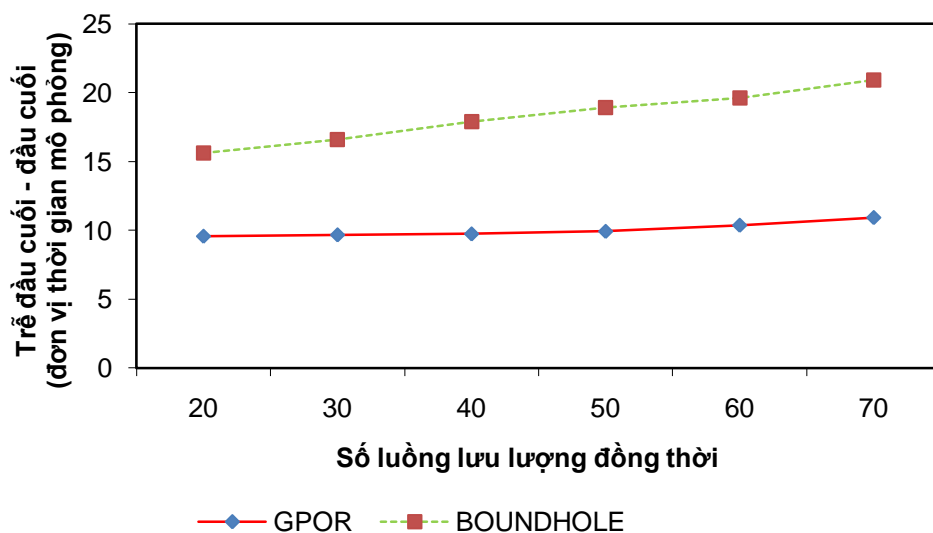
Hình 4.12. Tỷ lệ kéo dài độ dài đường đi với số luồng lưu lượng đồng thời khác nhau ($k = 4$).

4.5.3 Trễ đầu cuối – đầu cuối

Một cải thiện hiệu năng quan trọng mà kỹ thuật tối ưu hóa đường đi tạo ra cho định tuyến dựa trên thông tin vị trí là rút ngắn một cách đáng kể trễ đầu cuối – đầu cuối, như kết quả mô phỏng được thể hiện trong các Hình 4.13 và 4.14. Vì các đường đi được rút ngắn cùng với ít tắc nghẽn hơn tại các nút biên, trễ đầu cuối – đầu cuối được giảm đáng kể. Trễ đầu cuối–đầu cuối của GPOR tăng chậm hơn trễ đầu cuối–đầu cuối được tạo bởi BOUNDHOLE khi kích thước mạng tăng hoặc số luồng lưu lượng đồng thời tăng. Thực tế là khi kích thước mạng tăng hoặc số luồng lưu lượng đồng thời tăng, cơ hội để rút ngắn các đường đi tăng, do đó giảm trễ được nhiều hơn.



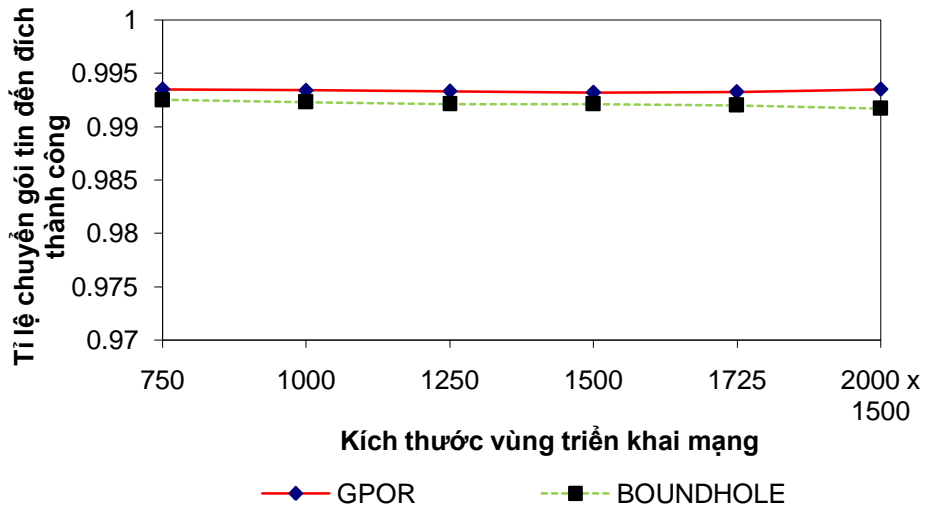
Hình 4.13. Trung bình trễ đầu cuối – đầu cuối với kích thước mạng khác nhau ($k = 4$).



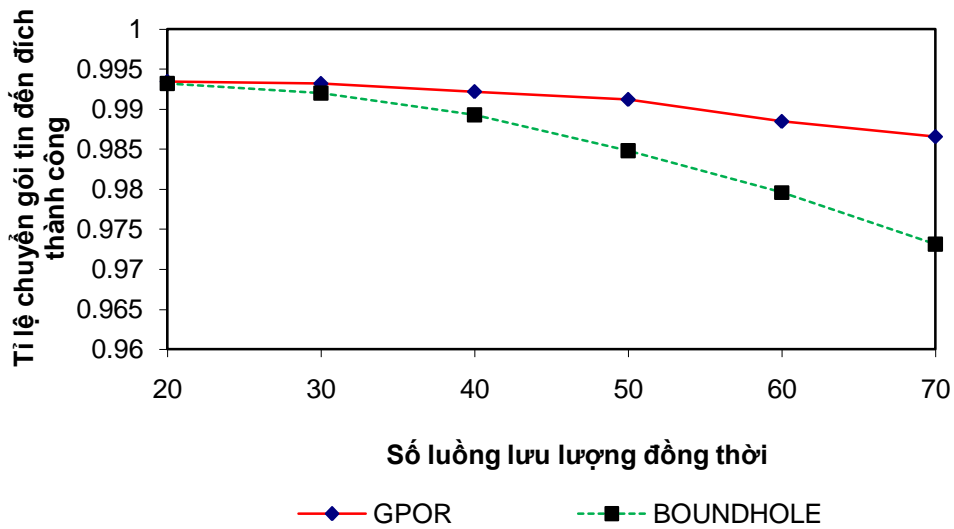
Hình 4.14. Trung bình trễ đầu cuối – đầu cuối với số luồng lưu lượng đồng thời khác nhau ($k = 4$).

4.5.4 Tỷ lệ chuyển gói thành công

Kết quả mô phỏng được thể hiện trong Hình 4.15 cho thấy cả BOUNDHOLE và GPOR cho tỷ lệ chuyển gói tin đến đích thành công ổn định khi kích thước mạng thay đổi. Hình 4.16 thể hiện tỷ lệ chuyển gói đến đích thành công của GPOR và BOUNDHOLE với số lưu lượng đồng thời thay đổi. Kết quả mô phỏng cho thấy GPOR có tỷ lệ chuyển gói đến đích thành công cao hơn so với BOUNDHOLE. Kết quả này được giải thích như sau. GPOR phân bổ lưu lượng đều đến cả các nút xa biên trong khi BOUNDHOLE đặt quá nhiều lưu lượng lên các nút biên. Do vậy, nhiều gói tin hơn có thể bị loại bỏ bởi BOUNDHOLE do tắc nghẽn xảy ra.



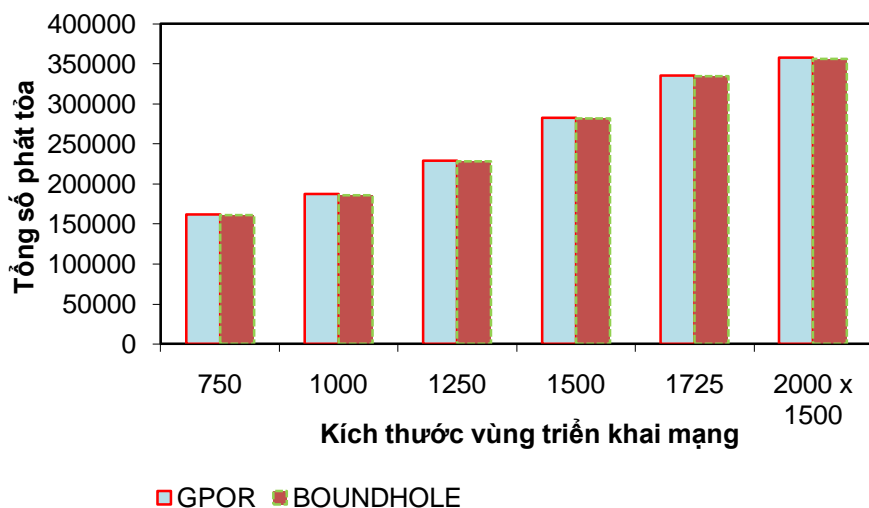
Hình 4.15. Tỷ lệ chuyển gói tin đến đích thành công với kích thước mạng khác nhau ($k = 4$).



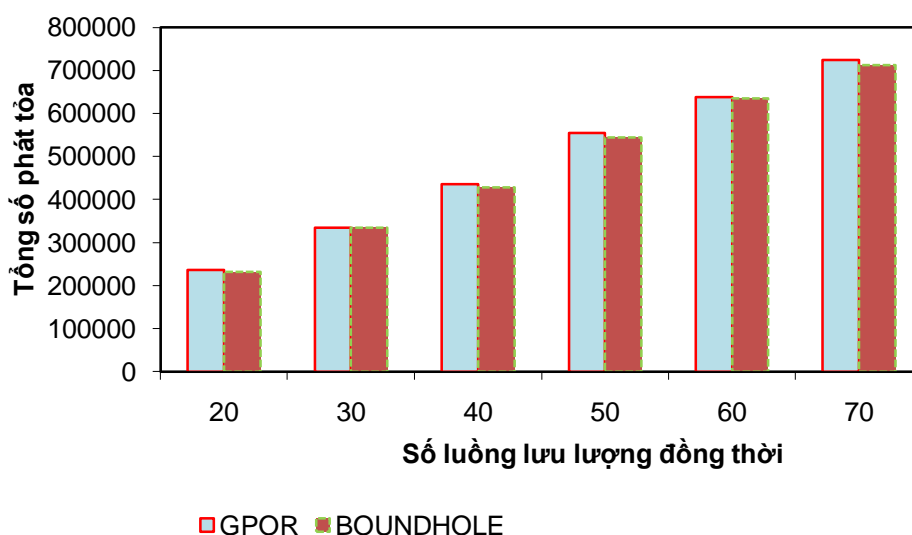
Hình 4.16. Tỷ lệ chuyển gói tin đến đích thành công với số lưu lượng đồng thời khác nhau ($k = 4$).

4.5.5 Chi phí truyền thông

Kết quả mô phỏng, được thể hiện trong Hình 4.17 và Hình 4.18, cho thấy GPOR sử dụng nhiều phát tỏa hơn BOUNDHOLE nhưng chênh lệch không đáng kể.



Hình 4.17. Tổng số phát tỏa với kích thước mạng khác nhau ($k = 4$).



Hình 4.18. Tổng số phát tỏa với số luồng lưu lượng đồng thời khác nhau ($k = 4$).

4.6 Thảo luận

GPOR, một giao thức tối ưu hóa đường đi và tránh cực tiểu địa phương cho định tuyến dựa trên thông tin vị trí đã được đề xuất. Trong khi vùng khả áp dụng của các phần tử định tuyến cho khả năng khai thác hiệu quả bảng định tuyến, kỹ thuật tạo đường tắt xây dựng các bảng định tuyến theo các luồng lưu lượng, do vậy đường đi được rút ngắn liên

tục. Việc đưa hai yếu tố này vào định tuyến dựa trên thông tin vị trí dẫn đến giao thức định tuyến mới và tốt hơn.

Gần đây, nhiều công trình nghiên cứu về định tuyến dựa trên thông tin vị trí cho mạng cảm biến trong không gian ba chiều đã được đề xuất [23, 96]. Giao thức tối ưu hóa đường đi được đề xuất trong luận án này có thể được mở rộng để áp dụng trong thế giới ba chiều. Ví dụ, giao thức trong [23] có thể được sử dụng để tìm các đường đi ban đầu và kỹ thuật tạo đường tắt có thể được sử dụng để tạo các đường ngắn hơn. Vùng khả áp dụng của các phần tử định tuyến được mở rộng thành *không gian khả áp dụng* để phù hợp với không gian ba chiều. Không gian khả áp dụng đơn giản nhất là hình cầu bán kính r có tâm tại pos_x . Nhiều nghiên cứu hơn nữa có thể được thực hiện để có không gian khả áp dụng tốt hơn.

CHƯƠNG 5

ĐỊNH TUYẾN DỰA TRÊN THÔNG TIN VỊ TRÍ SỬ DỤNG CẠNH TRANH KẾT HỢP

Các gói tin chào hỏi được sử dụng trong các giao thức định tuyến dựa trên thông tin vị trí nhằm duy trì thông tin về vị trí của các nút láng giềng. Những gói tin này không chỉ chiếm dụng nhiều băng thông mạng mà còn tiêu thụ nhiều năng lượng, do vậy làm giảm tuổi thọ của các nút.

Gần đây, nhiều giao thức định tuyến không sử dụng gói tin chào hỏi đã được đề xuất. Các giao thức này sử dụng cơ chế cạnh tranh (contention) để lựa chọn nút chuyển tiếp tiếp theo. Một cách cụ thể, thay vì chờ được chỉ định làm nút chuyển tiếp tiếp theo một cách thụ động, các nút láng giềng phải tham gia một đợt cạnh tranh mà *nút thắng cuộc sẽ trở thành nút chuyển tiếp tiếp theo*. Cho đến nay, có hai hình thức cạnh tranh đã được đề xuất là cạnh tranh quyết liệt (aggressive) và không quyết liệt (non-aggressive). Hai hình thức cạnh tranh này được mô tả ngắn gọn như sau:

Cạnh tranh quyết liệt (Aggressive Contention) [1, 13, 25, 37, 93]: (1) Nút hiện tại phát động đợt cạnh tranh bằng việc phát tủa gói dữ liệu DATA. (2) Khi nhận được gói DATA, mỗi nút láng giềng trong vùng cạnh tranh – một vùng con trong phạm vi phát sóng của nút hiện tại – đặt bộ định thời của nó với một giá trị phụ thuộc vào vị trí của nó, vị trí của nút hiện tại và vị trí của nút đích. Giá trị đặt cho bộ định thời được tính bởi các hàm trễ (delay function). (3) Nút láng giềng có bộ định thời hết hạn trở thành nút thắng cuộc. (4) Các nút láng giềng nhận được gói DATA được phát bởi nút thắng cuộc từ bỏ đợt cạnh tranh, hay trở thành nút thua cuộc.

Ưu điểm chính của hình thức cạnh tranh này là nó cho thời gian trễ thấp. Ngoài ra, hình thức cạnh tranh này không sử dụng bất kỳ gói tin điều khiển nào, do vậy tiết kiệm được băng thông mạng cũng như năng lượng của các nút. Tuy nhiên, hạn chế của hình thức cạnh tranh này bao gồm việc tạo ra các gói tin trùng lặp khi một nút không nhận được gói DATA từ nút thắng cuộc và cũng tuyên bố thắng cuộc. Để giảm thiểu số lượng gói tin trùng lặp, chúng ta phải thu hẹp vùng cạnh tranh sao cho tất cả các nút trong vùng cạnh tranh đều có thể liên lạc được với nhau. Rõ ràng, việc thu hẹp vùng cạnh tranh làm giảm tỉ lệ thành công khi chọn nút chuyển tiếp tiếp theo, do vậy làm giảm tỉ lệ chuyển gói tin đến đích thành công. Ảnh hưởng của vùng cạnh tranh đối với tỉ lệ chuyển gói tin đến đích thành công và trễ đã được nghiên cứu trong [14].

Cạnh tranh không quyết liệt (Non-aggressive Contention) [12, 45, 82, 84, 92, 104]:
(1) Nút hiện tại phát động đợt cạnh tranh bằng việc phát tủa gói REQUEST. (2) Khi nhận được gói REQUEST, mỗi nút lắng giềng trong vùng cạnh tranh đặt bộ định thời của nó với một giá trị được tính bởi hàm trễ. (3) Nút lắng giềng có bộ định thời hết hạn phát tủa gói tin RESPONSE. (4) Các nút lắng giềng nhận được gói RESPONSE từ bỏ đợt cạnh tranh. (5) Nút hiện tại xác định nút lắng giềng có hồi âm bằng gói RESPONSE sớm nhất là nút thắng cuộc và công bố nút thắng cuộc bằng cách phát tủa gói tin SELECTION. Trong hình thức cạnh tranh này, có thể gói REQUEST hoặc SELECTION là gói dữ liệu cần chuyển tiếp.

Khác với cạnh tranh quyết liệt, cạnh tranh không quyết liệt không tạo ra gói tin trùng lặp. Tuy nhiên, hình thức cạnh tranh này sử dụng các gói tin điều khiển và có trễ lớn hơn.

Trong chương này, giao thức định tuyến dựa trên thông tin vị trí không sử dụng gói tin chào hỏi có tên là Hybrid Contention-Based Geographic Routing (HCGR) được đề xuất. HCGR sử dụng đồng thời cả hai hình thức cạnh tranh, được gọi là cạnh tranh kết hợp. Mục đích của việc cạnh tranh kết hợp là để kế thừa các ưu điểm của cả hai hình thức cạnh tranh quyết liệt và không quyết liệt. Ngoài ra, một biến thể của kỹ thuật đi theo biên trong [20] sử dụng cạnh tranh kết hợp cũng được đề xuất. Hiệu năng của HCGR được đánh giá và so sánh với hiệu năng của các hình thức cạnh tranh đơn lẻ thông qua mô phỏng. Nghiên cứu về hiệu năng cho thấy cạnh tranh kết hợp tối ưu được tỉ lệ chuyển gói

tin thành công trong khi có độ phức tạp thông báo thấp và thời gian trễ đầu cuối – đầu cuối thấp.

5.1 Mô tả giao thức

Các gói dữ liệu được chuyển tiếp ở hai chế độ: tham lam và khôi phục. Thông tin về vị trí của nút đích, ba nút vừa đi qua và cực tiểu địa phương cuối cùng được ghi trong tiêu đề của gói tin. Mỗi gói tin được sinh ra ở chế độ tham lam, được chuyển về chế độ khôi phục khi gặp cực tiểu địa phương, và được đặt lại chế độ tham lam tại nút gần đích hơn cực tiểu địa phương cuối. Thông tin về vị trí của ba nút vừa đi qua được sử dụng ở chế độ khôi phục giúp gói tin đi dọc biên của vùng trống.

5.1.1 Cạnh tranh kết hợp

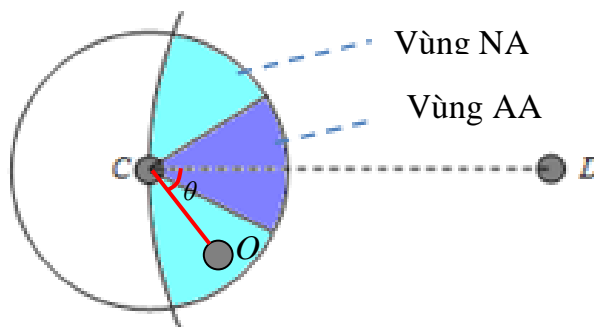
Hai hình thức cạnh tranh, quyết liệt và không quyết liệt, được sử dụng đồng thời. Vùng cạnh tranh được chia thành hai vùng con: Aggressive Area (AA) và Non-aggressive Area (NA). Các nút trong vùng AA tuân theo hình thức cạnh tranh quyết liệt trong khi các nút trong vùng NA tuân theo hình thức cạnh tranh không quyết liệt. Mục tiêu của cạnh tranh kết hợp là đạt tỉ lệ chuyển gói thành công cao như cạnh tranh không quyết liệt trong khi có độ phức tạp thông báo và trễ thấp hơn. Để đạt mục tiêu về tỉ lệ chuyển gói thành công, vùng cạnh tranh được thiết kế rộng nhất có thể, tức vùng cạnh tranh sẽ bao gồm tất cả các điểm thuộc phạm vi phát sóng của nút hiện tại và ở gần đích hơn so với nút hiện tại nếu gói tin ở chế độ tham lam, hoặc bao gồm toàn bộ vùng phủ sóng của nút hiện tại nếu gói tin ở chế độ khôi phục. Để đạt các mục tiêu độ phức tạp thông báo và trễ thấp, vùng cạnh tranh AA được thiết kế rộng nhất có thể (có diện tích lớn nhất trong đó các nút còn nghe được lẫn nhau), và cạnh tranh quyết liệt được ưu tiên cao hơn so với cạnh tranh không quyết liệt, nghĩa là cạnh tranh quyết liệt sẽ diễn ra trước cạnh tranh không quyết liệt. Cạnh tranh không quyết liệt đóng vai trò *dự phòng*, thực hiện chuyển gói tin khi cạnh tranh quyết liệt thất bại, và *không làm nhiễu* cạnh tranh quyết liệt. Để thực hiện điều này, các vùng cạnh tranh và hàm trễ được thiết kế sao cho *không có nút nào trong vùng AA có thời gian trễ lớn hơn thời gian trễ của nút trong vùng NA*, và nút trong vùng NA sẽ từ bỏ cạnh tranh nếu nó nghe dữ liệu được phát lại từ một nút trong vùng NA.

Tóm lại, hai nguyên tắc thực hiện cạnh tranh kết hợp là:

- Nguyên tắc 1: Thiết kế vùng cạnh tranh quyết liệt có diện tích lớn nhất trong đó các nút còn nghe được lẫn nhau; phần còn lại của vùng cạnh tranh được dùng cho cạnh tranh không quyết liệt.
- Nguyên tắc 2: Thiết kế hàm trễ sao cho không có nút nào trong vùng cạnh tranh quyết liệt có thời gian trễ lớn hơn thời gian trễ của nút trong vùng cạnh tranh không quyết liệt.

5.1.2 Vùng cạnh tranh và hàm trễ

Giả thiết các nút có cùng bán kính phát sóng r và tham số chung T_{max} được sử dụng trong các hàm trễ. T_{max} là thời gian tối đa mà nút tham gia cạnh tranh có thể đợi từ khi nó nhận được gói tin cho đến khi nó có hành động đầu tiên. Nếu gói dữ liệu ở chế độ tham lam, vùng cạnh tranh bao gồm tất cả các điểm thuộc phạm vi phát sóng của nút hiện tại và ở gần đích hơn so với nút đích với bán kính r . Lưu ý rằng 60° là góc lớn nhất trong đó các nút có thể nghe lẫn nhau, do đó cạnh tranh quyết liệt không tạo các gói tin trùng lặp. Vùng NA bao gồm các điểm còn lại của vùng cạnh tranh. Các vùng cạnh tranh được chỉ ra trong Hình 5.1.



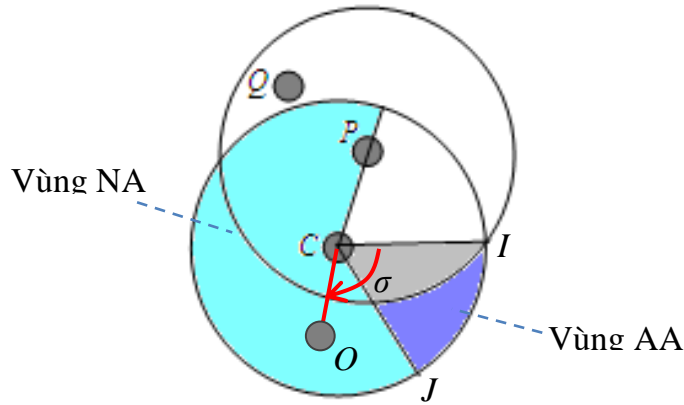
Hình 5.1. Các vùng cạnh tranh ở chế độ tham lam.

Với các vùng cạnh tranh này, để không có nút nào trong vùng AA có thời gian trễ lớn hơn thời gian trễ của nút trong vùng NA, ý tưởng chuyển tiếp theo góc bị chặn [18] được vận dụng. Cụ thể, để nút có góc nhỏ hơn sẽ có thời gian trễ nhỏ hơn, hàm trễ đơn giản nhất được đề xuất là

$$gct = \left(\frac{\theta}{360^\circ} \right) T_{max} \quad (5.1)$$

trong đó θ là góc \widehat{OCD} với C, D, O lần lượt là nút hiện tại, nút đích, nút láng giềng tham gia cạnh tranh và sở hữu bộ định thời.

Các vùng cạnh tranh cho gói dữ liệu ở chế độ khôi phục được mô tả như sau. Ký hiệu C là nút hiện tại, P và Q là hai nút trước đó mà gói dữ liệu vừa đi qua. Không mất tính tổng quát, giả sử Q nằm bên trái \overrightarrow{CP} , các vùng cạnh tranh được chỉ ra trong Hình 5.2. Ký hiệu (C, r) là đường tròn có tâm ở C và bán kính r . Gọi I là giao điểm bên phải \overrightarrow{CP} của hai đường tròn (C, r) và (P, r) . Vùng AA bao gồm các điểm thuộc góc $60^\circ \widehat{ICJ}$, với J là điểm bên phải \overrightarrow{CI} . Vùng NA bao gồm các điểm còn lại trong vùng phát sóng của C .



Hình 5.2. Các vùng cạnh tranh ở chế độ khôi phục.

Hàm trễ cho các bộ định thời ở chế độ khôi phục là

$$rct = \left(\frac{\sigma}{360^\circ} \right) T_{max} \quad (5.2)$$

trong đó σ là góc được tạo bởi \overrightarrow{CI} và \overrightarrow{CO} với O là nút láng giềng tham gia cạnh tranh và sở hữu bộ định thời. Lưu ý từ công thức (5.2) rằng giá trị hàm trễ tỉ lệ thuận với độ lớn góc được tạo khi quay \overrightarrow{CP} theo chiều kim đồng hồ cho đến khi nó gặp nút sở hữu bộ định thời. Do đó, hàm trễ thực hiện vai trò của kỹ thuật quay được sử dụng trong BOUNDHOLE [20]. Nói cách khác, hàm trễ cùng các vùng cạnh tranh được thiết kế trong chế độ khôi phục như trên tạo nên phiên bản không sử dụng gói tin chào hỏi dựa trên cạnh tranh kết hợp của BOUNDHOLE. Cũng lưu ý từ công thức (5.2) rằng không có nút nào trong vùng NA có thời gian trễ thấp hơn thời gian trễ của nút trong vùng AA.

Điều đó có nghĩa hai tính chất dự phòng và không làm nhiễu của cạnh tranh kết hợp được đảm bảo.

5.1.3 Hành vi của các nút

Nút hiện tại phát quảng bá gói DATA, đồng thời lưu một bản sao của gói tin và đặt một bộ định thời theo dõi (monitoring timer) cho gói tin với giá trị T_{max} . Nếu gói DATA ở chế độ khôi phục, trước khi phát gói tin nút hiện tại kiểm tra xem nó có gần đích hơn cực tiểu địa phương cuối mà gói tin đi qua hay không, nếu đúng như vậy, nó đặt gói tin về chế độ tham lam. Trong thời gian theo dõi, nếu nút hiện tại nhận lại được gói DATA hoặc nhận được gói RESPONSE, nó dừng bộ định thời theo dõi cho gói tin và xóa gói tin khỏi bộ đệm (do gói tin đã được chuyển tiếp thành công); nếu nút hiện tại nhận được gói RESPONSE đầu tiên mà trước đó chưa nhận lại được gói DATA, nó phát gói SELECTION với trường nút được chọn làm nút chuyển tiếp tiếp theo là định danh của nút đã gửi gói RESPONSE. Ngược lại, khi hết thời gian theo dõi, nếu không nhận lại được gói DATA và gói RESPONSE, nút hiện tại sẽ ứng xử tùy thuộc vào chế độ hiện tại của gói DATA. Cụ thể, nếu gói DATA đang ở chế độ khôi phục, nút hiện tại sẽ loại bỏ gói tin; ngược lại, nút hiện tại ghi thông tin vị trí của nó vào tiêu đề gói tin như cực tiểu địa phương cuối, đặt gói tin về chế độ khôi phục rồi lặp lại thủ tục trên.

Khi nhận được gói DATA, mỗi nút láng giềng không trong vùng cạnh tranh loại bỏ gói tin; nút trong vùng cạnh tranh lưu gói tin và đặt một bộ định thời cạnh tranh (contention timer) cho gói tin với giá trị gct hoặc rct tùy theo gói tin đang ở chế độ tham lam hay chế độ khôi phục, tương ứng. Khi bộ định thời cạnh tranh kết thúc, nút trong vùng AA trở thành nút thắng cuộc và phát gói DATA, nút trong vùng NA phát gói RESPONSE và đặt bộ định thời chờ đợi (waiting timer) với giá trị T_{max} để đợi được chọn làm nút chuyển tiếp tiếp theo. Các nút láng giềng nhận được gói DATA hoặc RESPONSE từ nút láng giềng khác sẽ dừng bộ định thời cạnh tranh và loại bỏ gói DATA đang được lưu. Khi bộ định thời chờ đợi hết hạn mà không nhận được gói SELECTION, nút láng giềng loại bỏ gói DATA.

Khi nhận được gói SELECTION, nút láng giềng dừng bộ định thời cạnh tranh (nếu chưa hết hạn) và bộ định thời chờ đợi (nếu có), phát gói DATA nếu nó được chỉ định là

nút chuyển tiếp tiếp theo hoặc loại bỏ gói DATA nếu nó không được chọn làm nút chuyển tiếp tiếp theo.

Nếu nút đích nhận được gói DATA, nó phát gói FINISH ngay lập tức. Khi nhận được gói FINISH, nút hiện tại dừng bộ định thời theo dõi, loại bỏ gói DATA đang lưu; các nút láng giềng dừng bộ định thời cạnh tranh và loại bỏ gói DATA đang lưu cho cạnh tranh.

Giao thức HCGR được mô tả hình thức trong Bảng 5.2 với cấu trúc gói tin được mô tả trong Bảng 5.1. Hàm *onReceive(...)* được gọi mỗi khi nút nhận được một gói tin mới. Hàm *onTimerExpired(...)* được kích hoạt khi có một bộ định thời hết hạn. Các hàm còn lại được gọi trong khi thực hiện *onReceive(...)* hoặc *onTimerExpired(...)*.

Bảng 5.1. Tiêu đề gói tin của HCGR.

Tên trường	Mô tả
<i>cid</i>	Định danh của cạnh tranh, có thể là kết hợp của định danh nút phát động cạnh tranh và số thứ tự được sinh bởi nút đó.
<i>launcher</i>	Định danh của nút phát động cạnh tranh.
<i>type</i>	Kiểu gói tin, có thể là DATA, RESPONSE, SELECTION hoặc FINISH.
<i>mode</i>	Chế độ chuyển tiếp gói dữ liệu, có thể là GREEDY hoặc RECOVERY.
<i>resp</i>	cid của gói DATA khác đang được lưu và đặt trữ cho việc cạnh tranh.
<i>sel</i>	Định danh của nút được chọn làm nút chuyển tiếp tiếp theo
<i>prev_{0,1,2}</i>	Vị trí ba nút mà gói DATA vừa đi qua
<i>lastlm</i>	Vị trí cực tiểu địa phương cuối mà gói DATA đi qua
<i>source</i>	Định danh nút nguồn của gói tin
<i>sink</i>	Định danh nút đích của gói tin
<i>sink-position</i>	Vị trí nút đích
<i>sender</i>	Định danh nút gửi

Bảng 5.2. Giao thức HCGR, mã cho nút C.

onReceive(p):

If *p.type* = DATA then

If *p.source* = *hostID()* then


```

    p.mode ← GREEDY; p.resp ← NULL; p.prev0 ← p.prev1 ← p.prev2 ←
    hostPosition();
    p.sink-position ← locservice(p.sink); LaunchContention(p);
Else
    cp ← lookup(p.resp);
    If cp <> NULL then
        If cp.launcher = hostID() then stopMonitoringTimer(cp.cid);
        Else stopContentionTimer(cp.cid);
        delete(cp); discard(p);
    Else
        If hostID() = p.sink then
            fp ← createFinishPacket(p.cid); broadcast(fp); sendUpper(p);
        Else If contentionArea(p) = NULL then discard(p);
        Else setContenttionTimer(p); cache(p);
Else If p.type = RESPONSE then
    cp ← lookup(p.resp);
    If cp <> NULL then
        If cp.launcher = hostID() then
            stopMonitoringTimer(cp.cid);
            sp ← createSelectionPacket(p.cid, p.sender); broadcast(sp);
        Else stopContentionTimer(cp.cid);
        delete(cp);
    discard(p);
Else If p.type = SELECTION then
    cp ← lookup(p.resp);
    If cp <> NULL then
        stopContentionTimer(cp.cid); stopWaitingTimer(cp.cid);
        If p.sel = hostID() then LaunchContention(cp);
        Else delete(cp);
    discard(p);
Else If p.type = FINISH then
    cp ← lookup(p.resp);
    If cp <> NULL then
        If cp.launcher = hostID() then stopMonitoringTimer(cp.cid);
        Else stopContentionTimer(cp.cid);
        delete(cp);
    discard(p);

```

onTimerExpired(p, timerType):

If *timerType* = MONITORING **then**

If *p.mode* = GREEDY **then**

p.mode \leftarrow RECOVERY; *p.lastlm* \leftarrow *hostPosition()*; *LaunchContention(p)*;

Else *delete(p)*;

Else If *timerType* = CONTENTION **then**

ca \leftarrow *contentionArea(p)*;

If *ca* = AA **then**

p.resp \leftarrow *p.cid*; *LaunchContention(p)*;

Else // *ca* = NA

rp \leftarrow *createResponsePacket(p.cid)*; *broadcast(rp)*; *setWaitingTimer(p)*;

Else // *timerType* = WAITING

delete(p);

LaunchContention(p):

If *p.mode* = RECOVERY **and** *distance(hostPosition(), p.sink-position) < distance(p.lastlm, p.sink-position)* **then** *p.mode* \leftarrow GREEDY;

p.launcher \leftarrow *hostID()*; *p.cid* \leftarrow *newContentionID()*;

p.prev₂ \leftarrow *p.prev₁*; *p.prev₁* \leftarrow *p.prev₀*; *p.prev₀* \leftarrow *hostPosition()*;

broadcast(p); *cache(p)*; *setMonitoringTimer(p)* ;

broadcast(p): Gửi quảng bá *p* đến tất cả các nút láng giềng

cache(p): Lưu *p* vào bộ nhớ đệm

contentionArea(p): Trả về vùng cạnh tranh (AA, NA, hoặc NULL) mà nút thuộc vào

createFinishPacket(pcid): Tạo gói FINISH cho cạnh tranh có định danh là *pcid*

createResponsePacket(pcid): Tạo gói RESPONSE cho cạnh tranh có định danh là *pcid*

createSelectionPacket(pcid, N): Tạo gói SELECTION, chọn *N* làm nút chuyển tiếp tiếp theo, cho cạnh tranh có định danh là *pcid*

delete(p): Xóa *p* từ bộ nhớ đệm

discard(p): Loại bỏ *p*

distance(A, B): Khoảng cách Euclid từ A đến B

hostID(): Trả về định danh của nút đang chạy chương trình

hostPosition(): Trả về tọa độ của nút đang chạy chương trình

locservice(N): Triệu gọi dịch vụ thông tin vị trí để biết vị trí của nút *N*

lookup(*pcid*): Trả về gói tin có *cid* là *pcid* đang được lưu trong bộ nhớ đệm

newContentionID(): Tạo một định danh cho một cạnh tranh mới

sendUpper(*p*): Gửi *p* lên tầng trên của ngăn xếp các giao thức

setContentionTimer(*p*): Tạo một bộ định thời cạnh tranh cho gói tin *p*, đặt giá trị cho bộ định thời được tạo là *gct* (1) nếu *p.mode* = GREEDY, hoặc *rct* (2) nếu *p.mode* = RECOVERY. *p.prev₂*, *p.prev₁*, *p.prev₀* được sử dụng để tính *rct* (2). *p.prev₀* được sử dụng để tính *gct* (1)

setMonitoringTimer(*p*): Tạo một bộ định thời theo dõi cho gói tin *p*, đặt giá trị cho bộ định thời được tạo là T_{\max}

setWaitingTimer(*p*): Tạo một bộ định thời chờ đợi cho gói tin *p*, đặt giá trị cho bộ định thời được tạo là T_{\max}

stopContentionTimer(*pcid*): Dừng bộ định thời cạnh tranh cho gói tin có *cid* là *pcid*

stopMonitoringTimer(*pcid*): Dừng bộ định thời theo dõi cho gói tin có *cid* là *pcid*

stopWaitingTimer(*pcid*): Dừng bộ định thời chờ đợi cho gói tin có *cid* là *pcid*

5.2 Phân tích và mô phỏng

Ký hiệu ACGR là giao thức cạnh tranh quyết liệt, tức là một biến thể của HCGR không sử dụng cạnh tranh không quyết liệt. Ngược lại, ký hiệu NCGR là giao thức cạnh tranh không quyết liệt, tức là một biến thể của HCGR chỉ sử dụng cạnh tranh không quyết liệt trên toàn bộ vùng cạnh tranh và không sử dụng cạnh tranh quyết liệt. ACGR chính là một biến thể của BLR [37] sử dụng góc 60° làm vùng cạnh tranh và hàm trễ theo góc. NCGR chính là một biến thể của SELECT-n-PROTEST [45] không sử dụng gói tin PROTEST.

HCGR và NCGR sử dụng các vùng cạnh tranh và hàm trễ như nhau, do vậy trong cùng cấu hình mạng, chúng sẽ tìm ra các đường đi như nhau. Nếu bỏ qua tắc nghẽn (hoặc giả thiết không có tắc nghẽn, nếu lưu lượng thấp) HCGR và NCGR sẽ cho cùng tỷ lệ chuyển gói tin đến đích thành công. Dễ dàng nhận thấy NCGR sử dụng nhiều thông báo điều khiển hơn HCGR do có những trường hợp HCGR không cần sử dụng các thông báo hồi đáp (RESPONSE) và thông báo lựa chọn (SELECTION), trong khi NCGR luôn luôn phải sử dụng các thông báo này. Điều đó có nghĩa NCGR có chi phí truyền thông cao hơn HCGR. Điều đó cũng có nghĩa khả năng tắc nghẽn xảy ra đối với NCGR sẽ cao hơn đối với HCGR, dẫn đến tỷ lệ chuyển gói tin đến đích thành công của NCGR bị giảm sút và thấp hơn so với tỷ lệ chuyển gói tin đến đích thành công của HCGR. Mặt khác, NCGR

cũng có thể có trễ cao hơn trễ của HCGR do có những tình huống gói tin được HCGR chuyển tiếp ngay trong khi NCGR còn phải gửi thông báo hồi đáp và chờ nhận thông báo lựa chọn. Hạn chế duy nhất của HCGR so với NCGR là cạnh tranh kết hợp có thể tạo ra các gói tin trùng lặp trong khi NCGR không tạo ra các gói tin như vậy.

ACGR sử dụng cùng hàm trễ với HCGR nhưng sử dụng vùng cạnh tranh là vùng con của vùng cạnh tranh được sử dụng bởi HCGR. Do đó, trong cùng cấu hình mạng, nếu ACGR tìm được một đường đi thì HCGR cũng tìm được đúng đường đi đó. Tuy nhiên, trong những tình huống ACGR thất bại do không có nút nào nằm trong vùng cạnh tranh quyết liệt, HCGR vẫn có thể thành công nếu có nút nằm trong vùng cạnh tranh không quyết liệt. Điều này dẫn đến tỷ lệ chuyển gói tin đến đích thành công của HCGR sẽ cao hơn so với tỷ lệ chuyển gói tin đến đích thành công của ACGR. Khả năng tạo gói tin trùng lặp của HCGR và ACGR là như nhau do các giao thức này sử dụng cùng hàm trễ và vùng cạnh tranh cho cạnh tranh quyết liệt. Ở những tình huống ACGR thành công, một vài gói tin điều khiển có thể được sử dụng bởi HCGR. Ở những tình huống ACGR thất bại, cạnh tranh không quyết liệt được thực hiện bởi HCGR. Do vậy, HCGR có thêm chi phí truyền thông so với ACGR.

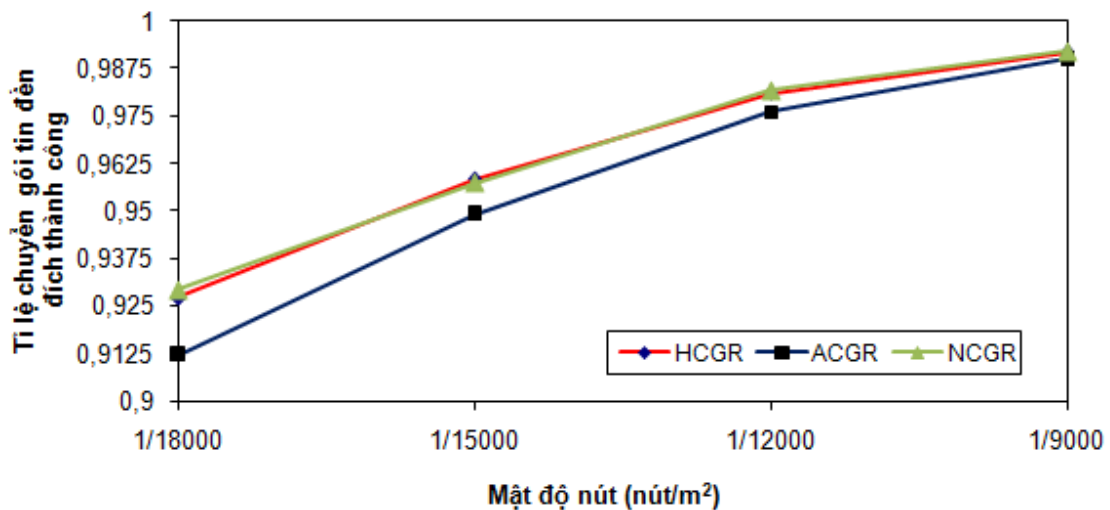
Những phân tích ở trên về hiệu năng của các giao thức HCGR, ACGR và NCGR đã được kiểm nghiệm qua mô phỏng. HCGR, ACGR và NCGR đã được cài đặt và đánh giá hiệu năng trong bộ mô phỏng mạng ns-2 [108]. Tiếp đó, chúng tôi đã thực hiện nhiều kịch bản mô phỏng. Các thông số đo hiệu năng được sử dụng bao gồm tỷ lệ chuyển gói tin thành công, phụ tải truyền thông, tỷ lệ gói tin trùng lặp và độ trễ đầu cuối – đầu cuối. Tỷ lệ chuyển gói tin thành công được đo bằng tổng số gói tin nhận được tại các nút đích chia cho tổng số gói tin được phát ra ở các nút nguồn. Phụ tải truyền thông được tính bằng tổng số các gói điều khiển đã được gửi trong mạng trong thời gian mô phỏng. Tỷ lệ gói tin trùng lặp được đo bằng tổng số gói tin trùng lặp chia cho tổng số gói tin nhận được tại các nút đích. Độ trễ đầu cuối – đầu cuối được đo bằng tổng độ trễ từ nguồn tới đích của các gói tin chia cho tổng số gói tin tới đích.

Nhằm đạt được các mục tiêu trong mô phỏng, các cấu hình có mật độ nút khác nhau đã được sử dụng. Vùng triển khai mạng là hình chữ nhật có kích thước $3750 \times 600 \text{ m}^2$.

Các nút có bán kính phát sóng 250 m, được bố trí vào vùng triển khai theo phân bố đồng xác suất. Mỗi mô phỏng kéo dài 900 giây (s), sử dụng 20 CBR với tốc độ phát 2 Kbps các gói tin kích thước 64-byte. Giá trị T_{max} được đặt là 10 mili-giây (ms). UDP được sử dụng ở tầng giao vận. Công nghệ không dây IEEE 802.11 WaveLAN được sử dụng ở tầng MAC. Ăngten đẳng hướng (omni) với mô hình lan truyền tín hiệu Two-Ray Ground được sử dụng cho mô hình truyền tín hiệu không dây. Kết quả mô phỏng cho thấy HCGR có tỷ lệ chuyển gói tin thành công cao bằng NCGR nhưng có phụ tải truyền thông cũng như độ trễ đầu cuối – đầu cuối thấp hơn NCGR.

5.2.1 Tỷ lệ chuyển gói tin thành công

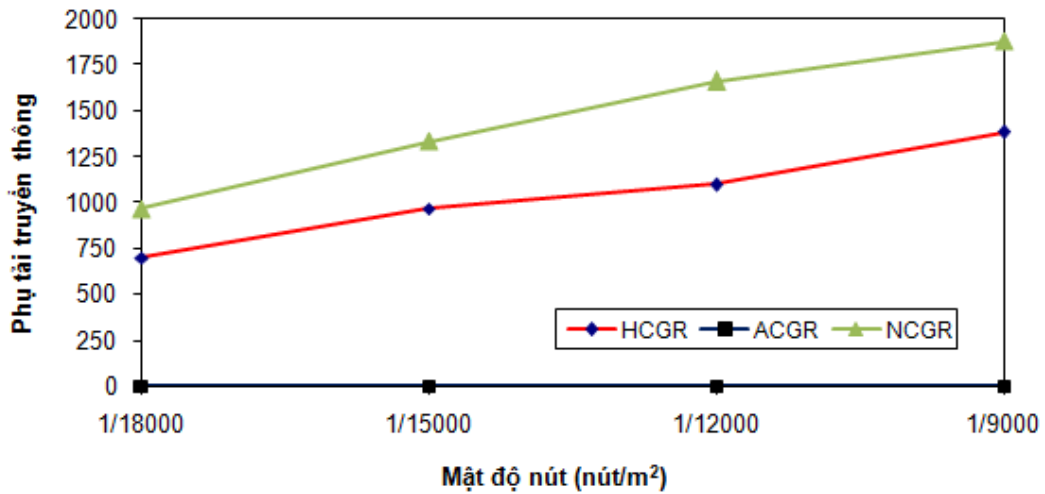
Hình 5.3 thể hiện kết quả mô phỏng về tỉ lệ chuyển gói tin đến đích thành công của các giao thức. HCGR và NCGR có cùng tỉ lệ chuyển gói tin đến đích thành công. ACGR có tỉ lệ chuyển gói tin đến đích thành công thấp hơn và tỉ lệ này giảm nhanh hơn khi mật độ nút giảm. Nguyên nhân của kết quả này là HCGR và NCGR sử dụng toàn bộ vùng cạnh tranh trong khi ACGR chỉ sử dụng vùng AA.



Hình 5.3. Tỉ lệ chuyển gói tin đến đích thành công của HCGR, ACGR và NCGR.

5.2.2 Phụ tải truyền thông

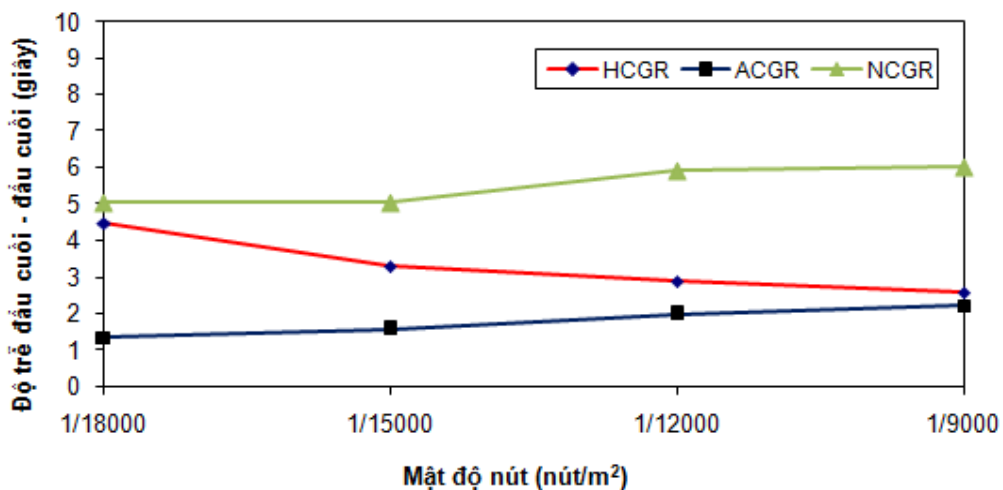
Kết quả mô phỏng khẳng định rằng HCGR sử dụng ít thông báo điều khiển hơn NCGR (Hình 5.4). Nguyên nhân là do cạnh tranh quyết liệt không sử dụng gói tin điều khiển.



Hình 5.4. Phụ tải truyền thông của HCGR, ACGR và NCGR.

5.2.3 Độ trễ đầu cuối – đầu cuối

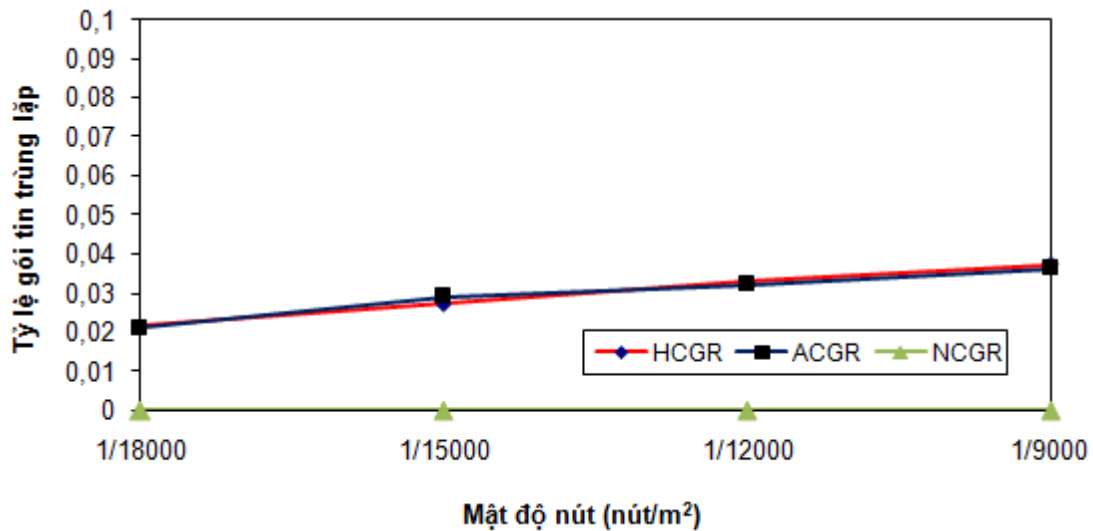
Kết quả mô phỏng cho thấy độ trễ đầu cuối – đầu cuối của HCGR hội tụ về độ trễ đầu cuối – đầu cuối của ACGR khi mật độ nút tăng lên (Hình 5.5). Điều này được giải thích là mạng có mật độ nút càng cao càng có ít vùng trống, do vậy cạnh tranh quyết liệt càng ít thất bại hơn; khi mật độ nút đủ cao, cạnh tranh quyết liệt không thất bại và HCGR sẽ có độ trễ đầu cuối – đầu cuối tương đương độ trễ đầu cuối – đầu cuối của ACGR. Khi mật độ nút thấp, cạnh tranh quyết liệt thường xuyên thất bại và độ trễ đầu cuối – đầu cuối của HCGR tiệm cận về độ trễ đầu cuối – đầu cuối của NCGR.



Hình 5.5. Độ trễ đầu cuối – đầu cuối của HCGR, ACGR và NCGR.

5.2.4 Tỷ lệ gói tin trùng lặp

Kết quả mô phỏng cho thấy HCGR và ACGR tạo ra cùng tỷ lệ gói tin trùng lặp trong khi NCGR không tạo ra gói tin trùng lặp (Hình 5.6). Kết quả này dễ giải thích vì cạnh tranh không quyết liệt không tạo ra gói tin trùng lặp.



Hình 5.6. Tỷ lệ gói tin trùng lặp của HCGR, ACGR và NCGR.

5.3 Thảo luận

Như đã được phân tích trong Mục 5.1.1, ràng buộc trong thiết kế các hàm trễ là “ không có nút nào trong vùng AA có thời gian trễ lớn hơn thời gian trễ của nút trong vùng NA”. Một cách thiết kế các hàm trễ thỏa mãn ràng buộc này là chia T_{\max} thành hai khoảng T_1 và T_2 , T_1 trước T_2 , các nút trong vùng AA sẽ tham gia cạnh tranh trong T_1 , sau đó các nút trong vùng NA sẽ tham gia cạnh tranh trong T_2 . Theo cách này, các vùng cạnh tranh AA và NA có thể được thiết kế tự do và hàm trễ cho các nút trong vùng này có thể khác hàm trễ cho các nút trong vùng còn lại. Sự tự do trong thiết kế vùng cạnh tranh và hàm trễ sẽ dẫn đến nhiều vùng cạnh tranh và hàm trễ khác nhau. Do vậy, một hướng mở cho phát triển giao thức này là thiết kế và lựa chọn các vùng cạnh tranh và hàm trễ tối ưu.

Tóm lại, trong chương này, chúng tôi đã đề xuất HCGR, một thuật toán kết hợp hai hình thức cạnh tranh là cạnh tranh quyết liệt và cạnh tranh không quyết liệt trong định tuyến dựa trên thông tin vị trí không sử dụng gói tin chào hỏi. Hai nguyên tắc kết hợp đã được đề xuất. Nguyên tắc thứ nhất là phân chia vùng cạnh tranh sao cho vùng cạnh tranh quyết liệt rộng nhất có thể. Thực hiện nguyên tắc này giúp hạn chế tối đa sử dụng các gói

tin điều khiển trong khi tạo ít các gói tin trùng lặp . Nguyên tắc thứ hai là thiết kế hàm trễ sao cho không có nút nào trong vùng cạnh tranh quyết liệt có thời gian trễ lớn hơn thời gian trễ của nút trong vùng cạnh tranh không quyết liệt. Thực hiện nguyên tắc này, nghĩa là ưu tiên cạnh tranh quyết liệt, giúp hạn chế sử dụng các gói tin điều khiển và rút ngắn trễ đầu cuối – đầu cuối. Mọi vùng cạnh tranh và hàm trễ cụ thể tuân thủ hai nguyên tắc được đề xuất đều có thể được sử dụng cho cạnh tranh kết hợp. Kết quả mô phỏng khẳng định, so với cạnh tranh không quyết liệt, cạnh tranh kết hợp có cùng tỉ lệ chuyển gói tin đến đích thành công nhưng có độ trễ đầu cuối – đầu cuối thấp hơn, và có phụ tải truyền thông báo thấp hơn.

KẾT LUẬN

Một giải pháp tổng thể cho định vị và định tuyến đơn phát dựa trên thông tin vị trí cho mạng cảm biến không dây là mục tiêu của luận án này. Từ việc khảo sát và phân tích các công trình liên quan, một giải pháp định vị và định tuyến đã được lựa chọn (Chương 2).

Giải pháp định vị hiệu quả và khả thi được lựa chọn là sử dụng đồ thị Delaunay kết hợp định vị theo khoảng cách [56]. Giải pháp định vị này cần một giải pháp phát hiện biên hiệu quả. Để đáp ứng yêu cầu đó, một thuật toán phát hiện biên hiệu quả dựa trên kết nối đã được đề xuất (Chương 3).

Với định tuyến đơn phát dựa trên thông tin vị trí, giải pháp hiệu quả và khả thi được đề cử là kết hợp chuyển tiếp tham lam [24] với đi theo biên [20]. Tuy nhiên, định tuyến theo phương pháp này còn hai yếu điểm chính. Thứ nhất, các đường đi dọc theo biên thường dài và không tối ưu. Thứ hai, nhiều đường đi dọc theo biên dẫn đến lưu lượng quá tải cho các nút biên. Điều này không chỉ dẫn đến tắc nghẽn tại biên khi có nhiều luồng lưu lượng đồng thời mà còn làm giảm nhanh tuổi thọ của các nút biên dẫn đến khoét rộng hơn các vùng trống. Để khắc phục các yếu điểm trên, một giao thức tối ưu hóa đường đi bằng cách tạo đường tắt đã được đề xuất (Chương 4). Trong khi vùng khả áp dụng của các phân tử định tuyến cho khả năng khai thác hiệu quả bảng định tuyến, kỹ thuật tạo đường tắt xây dựng các bảng định tuyến theo các luồng lưu lượng, do vậy đường đi được rút ngắn liên tục. Việc đưa hai yếu tố này vào định tuyến dựa trên thông tin vị trí dẫn đến giao thức định tuyến mới và tốt hơn.

Một vấn đề nữa trong định tuyến dựa trên thông tin vị trí là việc sử dụng các gói tin chào hỏi nhằm duy trì thông tin vị trí của các nút láng giềng. Những gói tin này không chỉ chiếm dụng nhiều băng thông mạng mà còn tiêu thụ nhiều năng lượng, do vậy làm giảm tuổi thọ của các nút. Nhằm loại bỏ các gói tin chào hỏi, nhiều giao thức định tuyến không sử dụng gói tin chào hỏi đã được đề xuất. Các giao thức này sử dụng cơ chế cạnh tranh để lựa chọn nút chuyển tiếp tiếp theo. Các hình thức cạnh tranh đơn lẻ, quyết liệt và

không quyết liệt, có những ưu điểm và nhược điểm riêng. Do vậy, một giao thức cạnh tranh kết hợp kế thừa được các ưu điểm từ cả hai hình thức cạnh tranh đơn lẻ đã được đề xuất (Chương 5).

Dĩ nhiên, có thể sử dụng đồng thời cả giao thức tối ưu hóa đường đi và cạnh tranh kết hợp được đề xuất trong cùng một giao thức. Nói cách khác, có thể cài đặt giao thức tối ưu hóa đường đi không sử dụng gói tin chào hỏi sử dụng cạnh tranh kết hợp. Tuy nhiên, sử dụng đồng thời như vậy chỉ là việc làm đơn giản và không được thực hiện trong luận án này. Với những giao thức được lựa chọn và đề xuất như trên, một giải pháp tổng thể cho cho định vị và định tuyến đơn phát dựa trên thông tin vị trí cho mạng cảm biến không dây đã được hoàn thiện.

Những vấn đề mở rộng đã được thảo luận ở cuối các Chương 3, 4, và 5 sẽ được tiếp tục nghiên cứu nhằm phát triển giải pháp được đưa ra ngày càng hoàn thiện và hiệu quả hơn.

Bên cạnh những kết quả đạt được, chắc chắn luận án không tránh khỏi những thiếu sót. Nghiên cứu sinh rất mong nhận được nhiều góp ý hữu ích của các thầy, cô và bạn đọc.

DANH MỤC CÁC CÔNG TRÌNH KHOA HỌC CỦA TÁC GIẢ LIÊN QUAN ĐẾN LUẬN ÁN

1. Thanh Le Dinh (2009), “Topological boundary detection in wireless sensor networks”, *International Journal of Information Processing Systems* 5(3), pp. 145-150.
2. Thanh Le Dinh, Dai Tho Nguyen (2010), “Greedy geographic routing with path optimization in wireless sensor networks”, *Proceedings of the 2010 IEEE-RIVF International Conference on Computing and Communications Technologies*, pp.148-153.
3. Thanh Le Dinh, Dai Tho Nguyen and Ho Thuan (2011), “Hybrid contention-based geographic routing in wireless sensor networks”, *Proceedings of the 2nd International Symposium on Information and Communications Technologies*, pp. 86-91.
4. Le Dinh Thanh, Ho Thuan, Nguyen Dai Tho (2013), “More efficient path optimization for greedy geographic routing in wireless sensor networks”, *Tạp chí Khoa học Trường Đại học Sư phạm Hà Nội* 58, tr. 150-156.

Danh mục này bao gồm 04 công trình.

TÀI LIỆU THAM KHẢO

- [1] I. Amadou, F. Valois (2010), "Pizza forwarding: A beaconless routing protocol designed for realistic radio assumptions" *Proc. of the 4th International Conf. on Sensor Technologies and Applications*, pp. 495-500.
- [2] N. Arad, Y. Shavitt (2009), "Minimizing recovery state in geographic ad hoc routing", *IEEE Transactions on Mobile Computing* 8(2), pp. 203-217.
- [3] H. Attiya and J. Welch (2004), *Distributed Computing: Fundamentals, Simulations and Advanced Topics, Second Edition*, John Wiley & Sons, Hoboken, New Jersey.
- [4] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B. A. Woodward (1998), "A distance routing effect algorithm for mobility," *Proceedings of IEEE/ACM MobiCom*, pp. 76-84.
- [5] K. Bi, K. Tu, N. Gu, W. L. Dong and X.Liu (2006), "Topological hole detection in sensor networks with cooperative neighbors", *Proceedings of the International Conference on Systems and Networks Communications*, pp. 31-35.
- [6] P. Bose, P. Morin (1999), "Online routing in triangulations", *Proc. of 10th International Symposium on Algorithms and Computation*, pp. 113-122.
- [7] P. Bose, A. Brodnik, S. Carlsson, E. D. Demaine, R. Fleischer, A. Lopez-Ortiz, P. Morin, J. I. Munro (2000), "Online routing in convex subdivisions", *Proc. of the International Symposium on Algorithms and Computation*, pp. 47-59.
- [8] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia (2001), "Routing with guaranteed delivery in ad hoc wireless networks", *Wireless Networks* 7(6), pp. 609-616.
- [9] M. Bui, S. K. Das, A.K. Datta, D. T. Nguyen (2001), "Randomized mobile agent based routing in wireless networks", *Internat. J. Found. Comput. Sci.* 12 (3), pp. 365-384.
- [10] U. Brandes, D. Fleischer (2007), "Geographic routing on improved coordinates", *Proc. of the 11th International Conference on Information Visualization*, pp. 263-270.
- [11] Carlos F. García-Hernández, Pablo H. Ibargüengoytia-González, Joaquín García-Hernández, and Jesús A. Pérez-Díaz (2007), "Wireless Sensor Networks and Applications: a Survey", *IJCSNS Int 264 ernational Journal of Computer Science and Network Security* 7(3), pp. 264-273.

- [12] M. Chaula, N. Goel, K. Kalaichelvan, A. Nayak, and I. Stojmenovic (2006), "Beaconless position based routing with guaranteed delivery for wireless ad-hoc and sensor networks", *Proc. of 1st IFIP Int. Conf. on Ad-Hoc Networking*, pp. 61-70.
- [13] D. Chen, J. Deng and P. K. Varshney (2005), "On the forwarding area of contention-based geographic forwarding for ad hoc and sensor networks", *Proc. IEEE SECON*, pp.130-141.
- [14] D. Chen, J. Deng and P. K. Varshney (2007), "Selection of a forwarding area for contention-based geographic forwarding in wireless multi-hop networks," *IEEE Trans. On Vehicular Technology* 56(5), pp. 3111-3122.
- [15] W. Choi, S. K. Das (2003), "Design and performance analysis of a proxy-based indirect routing scheme in ad hoc wireless networks", *Mobile Networks Appl.* 8(5), pp. 499-515.
- [16] W. Choi, S. K. Das, J. Cao, A. K. Datta (2005), "Randomized dynamic route maintenance for adaptive routing in multihop mobile ad hoc networks", *J. Parallel Distrib. Comput.* 65, pp. 107-123.
- [17] B. Chow and F. Luo (2003), "Combinatorial ricci flows on surfaces", *Journal of Differential Geometry* 63(1), pp. 97-129.
- [18] Colin J. Lemmon, Phillip Musumeci (2008), "Boundary mapping and boundary state routing (BSR) in ad hoc networks," *IEEE Transactions on Mobile Computing* 7 (1), pp. 127-139.
- [19] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, and D. Moore (2010), "Environmental Wireless Sensor Networks", *Proceedings of the IEEE, Special Issue on Emerging Sensor Network Applications* 98 (11), pp. 1903-1917.
- [20] Q. Fang, J. Gao, and L. Guibas (2006), "Locating and bypassing routing holes in sensor networks", *Mobile Networks and Applications* 11(2), pp. 187-200.
- [21] S. P. Fekete, M. Kaufmann, A. Kroller and K. Lehmann (2005), "A New Approach for Boundary Recognition in Geometric Sensor Networks", *Proceedings of the 17th Canadian Conference on Computational Geometry*, pp. 82-85.
- [22] S. P. Fekete, A. Kraoller, D. P. Sterer, S. Fischer and C. Buschmann (2004), "Neighborhood-Based Topology Recognition in Sensor Networks", *Proceedings of the ALGOSENSORS*, pp. 123-136.
- [23] R. Flury and R. Wattenhofer (2008), "Randomized 3D Geographic Routing," *Proceedings of IEEE Infocom*, pp. 834-842.

- [24] G. G. Finn (1987), *Routing and addressing problems in large metropolitan-scale internetwork*, Tech. Rep. ISI/RR-87-180, Information Sciences Institute, University of Southern California, California.
- [25] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, H. Hartenstein (2003), “Contention-based forwarding for mobile ad-hoc networks”, *Ad Hoc Netw. J.* 1(4), pp. 351-369.
- [26] Stefan Funke (2005), “Topological hole detection in wireless sensor networks and its applications”, *Proceedings of the DIALM-POMC 2005 joint workshop on Foundations of Mobile Computing*, pp. 44-53.
- [27] S. Funke and N. Milosavljević (2007), “Network sketching or: “how much geometry hides in connectivity? - part II”, *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 958–967.
- [28] J. Gao, L. J. Guibas, J. Hershberger, L. Zhang, and A. Zhu (2005), “Geometric spanners for routing in mobile networks”, *IEEE Journal on Selected Areas in Communications Special Issue on Wireless Ad Hoc Networks* 23(1), pp. 174–185.
- [29] R. Ghrist, A. Muhammad (2005), “Coverage and hole-detection in sensor networks via homology”, *Proceedings of the IPSN'05*, pp. 254–260.
- [30] S. Giordano, M. Hamdi (1999), *Mobility management: The virtual home region*, Technical Report SSC/1999/037, EPFL-ICA, Switzerland.
- [31] Z. Guping, W. Yu (2009), “Advance detour strategy for geographic routing in wireless sensor networks”, *Proceedings of the 2009 International Forum on Information Technology and Applications*, pp. 296-299.
- [32] T. Goff, N. B. Abu-Ghazaleh, D. S. Phatak, R. Kahvecioglu (2001), “Preemptive routing in ad hoc networks”, *Proceedings of Seventh ACM MobiCom*, pp. 43-52.
- [33] Z. Haas, M. Pearlman (2001), “The performance of query control schemes for the zone routing protocol”, *IEEE/ACM Transactions on Networking* 9(4), pp. 427-438.
- [34] R. S. Hamilton (1982), “Three manifolds with positive ricci curvature”, *Journal of Differential Geometry* 17, pp. 255–306.
- [35] H. Hassanein, A. Zhou (2000), “Routing with load balancing in wireless ad hoc networks”, *Proceedings of ACM Workshop on Modeling, Analysis and Simulation of Wireless and Mobile System*, pp. 89-96.

- [36] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher (2003), “Range-free localization schemes for large scale sensor networks”, *Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking (ACM Mobicom)*, pp. 81-95.
- [37] M. Heissenbüttel, T. Braun, T. Bernoulli, M. Wälchli (2004), “BLR: Beacon-less routing algorithm for mobile ad-hoc networks”, *Computer Communications* 27(11), pp. 1076-1086.
- [38] A. Hemmerling (1989), *Labyrinth problems: Labyrinth-searching abilities of automata*, B. G. Teubner, Leipzig.
- [39] F. Huc, A. Jarry, P. Leone, L. Moraru, S. Nikolettseas and J. Rolim (2009), “Early obstacle detection and avoidance for all to all traffic pattern in wireless sensor networks,” *Proceedings of the ALGOSENSORS 2009*, pp. 102–115.
- [40] P. Jacquet, P. Muhlethaler, A. Qayyum, L. Viennot, T. Clausen (2001), “Optimized link state routing (OLSR)”, *IETF Internet Draft, draft-ietf-manet-olsr-04.txt*.
- [41] R. Jain, A. Puri, R. Sengupta (2001), “Geographical routing using partial information in wireless ad hoc networks”, *IEEE Personal Communications*, pp. 48-57.
- [42] M. Jin, J. Kim, F. Luo, and X. Gu (2008), “Discrete surface ricci flow”, *IEEE Transaction on Visualization and Computer Graphics* 14(5), pp. 1030–1043.
- [43] D. Johnson, D. Maltz (1996), *Dynamic source routing in ad hoc wireless networks*, T. Imielinski and H. Korth, editors, Mobile Computing, chapter 5, Kluwer Academic.
- [44] D. B. Johnson, D. A. Maltz, Y. Hu, J. G. Jetcheva (2002), “The dynamic source routing protocol for mobile ad hoc networks (DSR),” *IETF Internet Draft, draft-ietf-manet-dsr-07.txt*.
- [45] H. Kalosha, A. Nayak, S. Rührup, and I. Stojmenovic (2008), “Select-and-protest-based beaconless georouting with guaranteed delivery in wireless sensor networks”, *Proc. of the 27th IEEE International Conf. on Computer Communications, Joint Conf. of IEEE Computer and Communications Societies (INFOCOM’08)*, pp. 346-350.
- [46] B. Karp and H.T. Kung (2000), “GPSR: Greedy perimeter stateless routing for wireless sensor networks”, *Proc. of Mobicom*, pp. 243-254.
- [47] B. Karp (2001), *Challenges in geographic routing: Sparse networks, obstacles, and traffic provisioning*, DIMACS Workshop on Pervasive Networking, Piscataway, NJ.

- [48] M. Käsemann, H. Fübler, H. Hartenstein, M. Mauve (2002), *A reactive location service for mobile ad hoc networks*, Technical Report TR-02-014, Department of Computer Science, University of Mannheim, Germany.
- [49] Y. J. Kim, R. Govindan, B. Karp, and S. Shenker (2005), “Geographic routing made practical”, *Proc. of Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*, pp. 217-230.
- [50] W. Kieb, H. Fübler, J. Widmer, M. Mauve (2004), “Hierarchical location service for mobile ad hoc networks,” *ACM SIGMOBILE Mobile Computing and Communications Review (MC2R)* 8 (4), pp. 47-58.
- [51] A. Koutsopoulos, S. Nikolettseas, J. D. P. Rolim (2009), “Near-optimal data propagation by efficiently advertising obstacle boundaries”, *Proc. of the 6th ACM Symposium on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*, pp. 15-22.
- [52] E. Kranakis, H Singh, and J. Urrutia (1989), “Compass routing on geometric networks”, *Proc. of the 11th Canadian Conference on Computational Geometry*, pp. 51-54.
- [53] A. Kroller, S. P. Fekete, D. Pfisterer, and S. Fischer (2006), “Deterministic boundary recognition and topology extraction for large sensor networks”, *Proceedings of the SODA'06*, pp. 1000-1009.
- [54] F. Kuhn, R.Wattenhofer, A. Zollinger (2002), “Asymptotically optimal geometric mobile ad-hoc routing”, *Proc. of the 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 24-33.
- [55] F. Kuhn, R.Wattenhofer, Y. Zhang, and A. Zollinger (2003), “Geometric ad-hoc routing: Of theory and practice”, *Proc. of PODC 2003*, pp. 63-72.
- [56] S. Lederer, Y. Wang, and J. Gao (2008), “Connectivity-based localization of large scale sensor networks with complex shape”, *Proc. of the 27th Annual IEEE Conference on Computer Communications (INFOCOM'08)*, pp. 789–797.
- [57] S. J. Lee, M. Gerla (2001), “Dynamic load-aware routing in ad hoc networks”, *Proceedings of 10th IEEE International Conference on Communication*, pp. 3206-3210.
- [58] B. Leong, B. Liskov, R. Morris (2006), “Geographic routing without planarization”, *Proc. of the 3rd Symposium on Networked Systems Design & Implementation*, pp. 339-352.

- [59] J. Li, J. Jannotti, D. S. J. DeCouto, D. R. Karger, R. Morris (2000), “A scalable location service for geographic ad hoc routing”, *Proceedings of the Sixth Annual ACM/IEEE MobiCom*, pp. 120-130.
- [60] P. Li, G. Wang and J. Wu, H. C. Yang (2009), “Hole reshaping routing in large-scale mobile ad-hoc networks”, *Proc. of the 28th IEEE Conference on Global Telecommunications*, pp. 1738-1743.
- [61] X. Li, H. Shi and Y. Shang (2005), “A Sorted RSSI Quantization Based Algorithm for Sensor Network Localization”, *Proceedings of the 11th International Conference on Parallel and Distributed Systems*, pp. 557-563.
- [62] M. Lim, A. Greenhalgh, J. Chesterfield, and J. Crowcroft (2005), “Landmark guided forwarding”, *Proc. of the IEEE International Conference on Network Protocols*, pp. 169-178.
- [63] K. Liu, N. Abu-Ghazaleh (2006), “Aligned virtual coordinates for greedy routing in WSNs”, *Proc. 2006 IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pp. 377–386.
- [64] W. J. Liu, K.T. Feng (2009), “Greedy routing with anti-void traversal for wireless sensor networks”, *IEEE Transactions on Mobile Computing* 8(7), pp. 910–922.
- [65] A. Loch, H. Frey, M. Hollick (2014), “Curve-based planar graph routing with guaranteed delivery in multihop wireless networks”, *Pervasive and Mobile Computing* 11, pp. 70-85.
- [66] K. Luthy, E. Grant, N. Deshpande, T. C. Henderson (2012), “Perimeter detection in wireless sensor networks”, *Robotics and Autonomous Systems* 60, pp. 266–277.
- [67] G. Q. Mao, B. Fidan, and B. D. O. Anderson (2007), “Wireless sensor network localization techniques”, *The International Journal of Computer and Telecommunications Networking Computer Networks* 51(10), pp. 2529-2553.
- [68] L. Moraru, P. Leone, S. Nikolettseas, J. D. P. Rolim (2007), “Near optimal geographic routing with obstacle avoidance in wireless sensor networks by fast-converging trust-based algorithms”, *Proc. of the 3rd ACM workshop on QoS and security for wireless and mobile networks*, pp. 31-38.
- [69] L. Moraru, P. Leone, S. Nikolettseas and J. Rolim (2008), “Path quality detection algorithms for near optimal geographic routing with obstacles”, *Wirel. Commun. Mob. Comput.*, pp. 1-13.

- [70] A. Mostefaoui, M. Melkemi, A. Boukerche (2012), “Routing Through Holes in Wireless Sensor Networks”, *Proc. of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 395-402.
- [71] S. Murthy, J. J. Garcia-Luna-Aceves (1996), “An efficient routing protocol for wireless networks”, *ACM/Baltzer Mobile Networks and Applications* 1(2), pp. 183-197.
- [72] A. Nasipuri and K. Li (2002), “A directionality based location discovery scheme for wireless sensor networks”, *Proc. of 1st ACM International Workshop on Sensor Networks and Applications*, pp. 105-111.
- [73] R. Nelson, L. Kleinrock (1984), “The spatial capacity of a slotted oloha multihop packet radio network with capture”, *IEEE Transactions on Communications* 3, pp. 684-694.
- [74] Paulo Neves, Michal Stachyra, Joel Rodrigues (2008), “Application of wireless sensor networks to healthcare promotion”, *Journal of Communications Software and Systems* 4(3), pp. 1845-6421.
- [75] S. Nikolettseas and O. Powell (2007), “Simple and efficient geographic routing around obstacles for wireless sensor networks”, *Proc. of the 6th Workshop on Efficient and Experimental Algorithms, LNCS, Springer-Verlag*, pp. 161-174.
- [76] R. Ogier, M. Lewis, F. Templin (2003), “Topology dissemination based on reverse-path forwarding (TBRPF)”, *IETF Internet Draft, draft-ietf-manet-tbrpf-09.txt*.
- [77] C. E. Perkins (2000), *Ad hoc networking*, Addison-Wesley, Reading, MA.
- [78] C. E. Perkins, P. Bhagwat (1994), “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers”, *Proceedings of ACM SIGCOMM*, pp. 234-244.
- [79] C. E. Perkins, E. M. Royer (1999), “Ad hoc on-demand distance vector”, *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100.
- [80] C. E. Perkins, E. Belding-Royer, S. R. Das (2003), “Ad hoc on-demand distance vector (AODV) routing”, <http://www.ietf.org/rfc/rfc3561.txt>, *RFC 3561*.
- [81] A. Rao, C. Papadimitriou, S. Shenker, I. Stoica (2003), “Geographic routing without location information”, *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, pp. 96–108.

- [82] S. Ruehrup, I. Stojmenovic (2013), "Optimizing Communication Overhead while Reducing Path Length in Beaconless Georouting with Guaranteed Delivery for Wireless Sensor Networks", *IEEE Transactions on Computers* 62 (12), pp. 2440 – 2453.
- [83] R. Sarkar, X. Yin, J. Gao, F. Luo, and X. D. Gu (2009), "Greedy routing with guaranteed delivery using ricci flows", *Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*, pp. 121 - 132.
- [84] J. A. Sanchez, R. Marin-Perez and P. M. Ruiz (2007), "BOSS: Beacon-less on demand strategy for geographic routing in wireless sensor networks", *Proc. of 4th IEEE MASS*, pp. 1-10.
- [85] C. Santivanez, R. Ramanathan, I. Stavrakakis (2001), "Making link-state routing scale for ad hoc networks", *Proceedings of ACM MobiHoc*, pp. 22-32.
- [86] Y. Shang, W. Ruml, Y. Zhang and M. P. J. Fromherz (2003), "Localization from mere connectivity", *Proceedings of the 4th ACM International Symposium on Mobile Ad hoc Networking & Computing*, pp. 201-212.
- [87] Inyoung Shin, Ngoc Duy Pham, and Hyunseung Choo (2009), "Virtual convex polygon based hole boundary detection and time delay based hole detour scheme in WSNs", *Human Interface, Part I, HCII 2009, LNCS 5617*, pp. 619–627.
- [88] I. Stojmenovic, X. Lin (2001), "Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks", *IEEE Trans. on Parallel and Distributed Systems* 12, pp. 1023-1032.
- [89] Tian, Y., Yu, F., Choi, Y., Park, S., Lee, E., Jin, M., Kim, S.H. (2008), "Energy-efficient data dissemination protocol for detouring routing holes in wireless sensor networks", *Proc. of IEEE International Conference on Communications*, pp. 2322–2326.
- [90] W. Tutte (1963), "How to draw a graph", *Proc. London Math. Soc.* 13(3), pp. 743–768.
- [91] Y. Wang, J. Gao and J. S. B. Mitchell (2006), "Boundary recognition in sensor networks by topological methods", *Proceedings of the MobiCom '06*, pp. 122-133.
- [92] M. Watanabe, H. Higaki (2007), "No-beacon GEDIR: Location-based ad-hoc routing with less communication overhead", *Proc. of the International Conf. on Information Technology*, pp. 48-55.

- [93] M. Witt, V. Turau (2005), “BGR: Blind geographic routing for sensor networks”, *Proc. of 3rd Intl. Workshop on Intelligent Solutions in Embedded Systems*, pp. 51-61.
- [94] L. C. Wu, W. B. Li, W. C. Kuo (2010), “Detour Routing Protocol for Geographic Sensor Networks,” *Proc. of the 2010 International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 505-210.
- [95] Feng Xi, Zhong Liu (2009), “Small world topology-aware geographic routing in wireless sensor networks”, *Proc. of the 2009 International Conference on Communications and Mobile Computing*, pp. 116-120.
- [96] Su Xia, Xiaotian Yin, Hongyi Wu, Miao Jin, Xianfeng David Gu (2011), “Deterministic greedy routing with guaranteed delivery in 3D wireless sensor networks”, *Proceeding MobiHoc '11 Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, doi>10.1145/2107502.2107504.
- [97] P. Xing, H. Yu and Y. Zhang (2005), “An assisting localization method for wireless sensor networks”, *Proceedings of the Second International Conference on Mobile Technology, Applications and Systems*, pp. 1-6.
- [98] Y. Xue, B. Li, K. Nahrstedt (2001), “A scable location management scheme in mobile ad hoc networks”, *Proc. of the IEEE Conference on Local Computer Networks*, pp. 102-111.
- [99] J. You, Q. Han, D. Lieckfeldt, J. Salzmann, D. Timmermann (2010), “Virtual position based geographic routing for wireless sensor networks”, *Computer Communications* 33, pp. 1255–1265.
- [100] Yu, F., Lee, E., Choi, Y., Park, S., Lee, D., Tian, Y., Kim, S.H. (2007), “A modeling for hole problem in wireless sensor networks”, *Proc. of International Wireless Communications and Mobile Computing Conference*, pp. 370–375.
- [101] Fucai Yu, Younghwan Choi, Soochang Park, Euisin Lee, Ye Tian, Minsuk Jin, and Sang-Ha Kim (2008), “Anchor node based virtual modeling of holes in wireless sensor networks”, *Proc. IEEE International Conference on Communications*, pp. 3120 – 3124.
- [102] L. Zhang, D. Li, A. Lim (2010), “Energy-efficient traffic-aware detour trees for geographical routing”, *International Journal of Computer Networks & Communications* 2(1), pp. 154-168.
- [103] Y. Zhao, Y. Chen, B. Li, Q. Zhang (2007), “Hop ID: a virtual coordinate-based routing for sparse mobile ad hoc networks”, *IEEE Trans. on Mobile Computing* 6(9), pp. 1075–1089.

- [104] M. Zorzi (2004), “A new contention-based mac protocol for geographic forwarding in ad hoc and sensor networks”, Proc. of IEEE Conf. on Communications, pp. 3481-3485.
- [105] http://en.wikipedia.org/wiki/IEEE_802.15.4
- [106] http://en.wikipedia.org/wiki/Mobile_ad_hoc_network
- [107] <http://en.wikipedia.org/wiki/VANET>
- [108] <http://www.isi.edu/nsnam>
- [109] <http://www.libelium.com>
- [110] <http://www.memsic.com>
- [111] <http://www.zigbee.org>

PHỤ LỤC

Phụ lục 1. Ước lượng khoảng cách và góc

Để sử dụng định vị theo khoảng cách, ước lượng khoảng cách đến các điểm neo là bắt buộc. Quá trình ước lượng theo khoảng cách sử dụng các tiện ích đã có của nút không dây, cụ thể là thiết bị truyền thông radio. Các đặc tính của truyền thông không dây phần nào được quyết định bởi khoảng cách giữa nút gửi và nút nhận, và nếu ước lượng được các đặc tính ở nút nhận thì chúng có thể được dùng để ước lượng khoảng cách. Đặc tính quan trọng nhất là độ mạnh tín hiệu nhận được RSSI, thời điểm đến ToA, chênh lệch thời điểm đến TDoA.

Độ mạnh tín hiệu nhận được (Received Signal Strength Indicator - RSSI) [61]

Giả sử đã biết năng lượng phát là P_{tx} , mô hình suy giảm tín hiệu, và hệ số suy giảm α , độ mạnh tín hiệu nhận được P_{rcvd} được sử dụng để ước lượng khoảng cách d trong công thức suy giảm như sau

$$P_{rcvd} = c \frac{P_{tx}}{d^\alpha} \Leftrightarrow d = \sqrt[\alpha]{\frac{c P_{tx}}{P_{rcvd}}}$$

Cách ước lượng này hấp dẫn vì không yêu cầu thêm bất kỳ phần cứng nào cũng như không yêu cầu thêm chi phí truyền thông. Tuy nhiên, hạn chế của cách ước lượng này là giá trị của RSSI không phải là một hằng số mà rất dao động, thậm chí cả khi nút gửi và nút nhận không di chuyển. Điều này do các hiệu ứng như suy giảm và di động của môi trường. Ở mức độ nào đó, hiệu ứng này có thể được hạn chế bằng các ước lượng lặp và lọc bỏ các giá trị không đúng bằng các kỹ thuật thống kê. Ngoài ra, các bộ thu phát đơn giản và rẻ tiền có thể cho các giá trị RSSI khác nhau đối với cùng độ mạnh tín hiệu thực tế; tương tự năng lượng phát thực sự của các bộ thu phát này cho thấy sự khác biệt từ năng lượng chủ định. Vấn đề thứ ba là sự có mặt của các vật cản cùng với suy giảm đa đường. Ở đây, suy giảm tín hiệu dọc đường gián tiếp lớn hơn suy giảm tín hiệu dọc

đường trực tiếp, dẫn đến ước lượng khoảng cách lớn hơn thực tế. Vì đây là một vấn đề có tính cấu trúc, nó không thể được giải quyết bằng nhiều ước lượng lặp.

Thời điểm đến (Time of Arrival - ToA) [97]

Thời điểm đến (còn được gọi là “thời gian bay”) khai thác quan hệ giữa khoảng cách và thời gian phát khi biết tốc độ lan truyền. Giả sử các nút gửi và nút nhận biết thời điểm bắt đầu phát – ví dụ nhờ một xung cao tần, ToA có thể được dùng để tính thời gian lan truyền, và do vậy tính được khoảng cách. Để đỡ cho nút nhận khỏi nhiệm vụ này, nút nhận có thể trả về “xung ước lượng” ở một thời điểm xác định; tiếp đó nút gửi chỉ cần ước lượng thời gian quay vòng với giả thiết kênh truyền đối xứng.

Tùy thuộc vào môi trường truyền dẫn được sử dụng, thời điểm đến yêu cầu đồng hồ có độ phân giải lớn để có kết quả với độ chính xác chấp nhận được. Với các sóng âm, các yêu cầu về độ phân giải là khiêm tốn; nhưng yêu cầu này khó đối với lan truyền sóng radio.

Chênh lệch thời điểm đến (Time Difference of Arrival - TDoA) [97]

Để khắc phục nhược điểm phải có sự đồng bộ rõ ràng, phương pháp chênh lệch thời điểm đến (TDoA) sử dụng đồng bộ không rõ ràng bằng việc cung cấp thông tin bắt đầu phát cho nút nhận. Điều này có thể được thực hiện nếu hai môi trường truyền dẫn với hai tốc độ lan truyền rất khác nhau được sử dụng – ví dụ các sóng radio lan truyền với tốc độ ánh sáng và siêu âm với một chênh lệch về tốc độ khoảng sáu bậc cường độ. Do vậy, khi một nút gửi bắt đầu phát một sóng siêu âm và một sóng radio đồng thời, nút nhận có thể tính được chênh lệch thời điểm đến giữa sóng radio và sóng siêu âm, không cần biết thời gian lan truyền của sóng radio.

Hạn chế hiển nhiên của tiếp cận này là cần sử dụng hai loại thiết bị thu phát ở mỗi nút. Ngược lại, ưu điểm của tiếp cận này là độ chính xác tốt hơn đáng kể so sánh với tiếp cận dựa vào RSSI.

Góc đến (Angle of Arrival - AoA) [97]

Thay vì ước lượng khoảng cách giữa các nút, góc đến có thể được ước lượng. Một góc như vậy có thể là góc của đường nối một điểm neo và một nút chưa biết vị trí so với một hướng tham chiếu (“0° bắc”). Nó cũng có thể là góc giữa hai đường nối như vậy nếu không có hướng tham chiếu được biết bởi tất cả các nút.

Một tiếp cận truyền thống trong ước lượng góc là sử dụng ăngten có hướng (ăngten chỉ gửi và nhận tín hiệu theo một hướng xác định), quay quanh trục của chúng, tương tự một trạm radar hoặc hải đăng quen thuộc. Nó làm cho ước lượng góc về khái niệm là đơn giản, nhưng các thiết bị như thế khá không phù hợp cho nút cảm biến; chúng có thể dùng cho các điểm neo thuộc hạ tầng.

Một kỹ thuật khác là khai thác tốc độ lan truyền của các dạng sóng. Với nhiều ăngten được nối vào một thiết bị với phân cách biết trước và tính được chênh lệch thời điểm đến giữa các ăngten, ta có thể tính được hướng đến của sóng. Phân cách giữa các ăngten nhỏ hơn yêu cầu độ chính xác cao hơn trong chênh lệch thời gian, dẫn đến yêu cầu định thời không ngừng cho các nút cảm biến kích thước nhỏ.

Phụ lục 2. Cơ sở toán học cho định vị theo khoảng cách

Giải pháp với ba điểm neo và các giá trị khoảng cách đúng

Giả sử có ba điểm neo với vị trí biết trước (x_i, y_i) , $i = 1, \dots, 3$, một nút chưa biết vị trí (x_u, y_u) , và giá trị khoảng cách lý tưởng r_i , $i=1, \dots, 3$. Từ định lý Pytago, một hệ ba phương trình sau:

$$(x_i - x_u)^2 + (y_i - y_u)^2 = r_i^2 \text{ với } i = 1, \dots, 3 \quad (\text{p2.1})$$

Để giải hệ phương trình này, ta biến đổi hệ này thành các phương trình tuyến tính theo x_u và y_u . Để làm như vậy, ta cần loại bỏ các toán hạng x_u^2 và y_u^2 bằng cách trừ phương trình thứ nhất và thứ hai cho phương trình thứ ba, được các phương trình

$$(x_1 - x_u)^2 - (x_3 - x_u)^2 + (y_1 - y_u)^2 - (y_3 - y_u)^2 = r_1^2 - r_3^2 \quad (\text{p2.2})$$

$$(x_2 - x_u)^2 - (x_3 - x_u)^2 + (y_2 - y_u)^2 - (y_3 - y_u)^2 = r_2^2 - r_3^2 \quad (\text{p2.3})$$

Tương đương

$$2(x_3 - x_1)x_u + 2(y_3 - y_1)y_u = (r_1^2 - r_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \quad (\text{p2.4})$$

$$2(y_3 - x_2)x_u + 2(y_3 - y_2)y_u = (r_2^2 - r_3^2) - (x_2^2 - x_3^2) - (y_2^2 - y_3^2) \quad (\text{p2.5})$$

Viết lại (p2.4) và (p2.5) dưới dạng ma trận

$$2 \begin{bmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{bmatrix} \begin{bmatrix} x_u \\ y_u \end{bmatrix} = \begin{bmatrix} (r_1^2 - r_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \\ (r_2^2 - r_3^2) - (x_2^2 - x_3^2) - (y_2^2 - y_3^2) \end{bmatrix} \quad (\text{p2.6})$$

Ví dụ 2.1: Giả sử $(x_1, y_1) = (2, 1)$, $(x_2, y_2) = (5, 4)$, và $(x_3, y_3) = (8, 2)$ – với các khoảng cách giữa các điểm neo đến nút chưa biết thông tin vị trí là $r_1 = \sqrt{10}$, $r_2 = 2$, $r_3 = 3$. Công thức p2.6 trở thành

$$2 \begin{bmatrix} 6 & 1 \\ 3 & -2 \end{bmatrix} \begin{bmatrix} x_u \\ y_u \end{bmatrix} = \begin{bmatrix} 64 \\ 22 \end{bmatrix} \quad (\text{p2.7})$$

cho kết quả $x_u = 5, y_u = 2$.

Xử lý lỗi khoảng cách

Thách thức thực sự trong định vị theo ba khoảng cách là khi các ước lượng khoảng cách không chính xác, chỉ ước lượng \tilde{r} với sai số biết trước ε . Giải hệ phương trình ở trên với $\tilde{r}_i = r_i + \varepsilon_i$ sẽ không cho kết quả (x_u, y_u) đúng.

Một cách trực quan, giải pháp cho vấn đề này là sử dụng nhiều hơn ba điểm neo và sử dụng các ước lượng dư để tính lỗi trong mỗi ước lượng. Về mặt toán học, nó biến đổi hệ phương trình ở trên thành hệ phương trình không xác định, viết dưới dạng ma trận như sau

$$2 \begin{bmatrix} x_n - x_1 & y_n - y_1 \\ \vdots & \vdots \\ x_n - x_{n-1} & y_n - y_{n-1} \end{bmatrix} \begin{bmatrix} x_u \\ y_u \end{bmatrix} = \begin{bmatrix} (r_1^2 - r_n^2) - (x_1^2 - x_n^2) - (y_1^2 - y_n^2) \\ \vdots \\ (r_{n-1}^2 - r_n^2) - (x_{n-1}^2 - x_n^2) - (y_{n-1}^2 - y_n^2) \end{bmatrix} \quad (\text{p2.8})$$

Với hệ phương trình không xác định này, lời giải có thể tối thiểu trung bình bình phương lỗi, nghĩa là tìm (x_u, y_u) để $\|Ax - b\|_2$, nhỏ nhất, trong đó $0.5A$ là ma trận bên trái (một ma trận $(n-1) \times 2$), $x = (x_u, y_u)$, và b là ma trận bên phải (một vectơ $(n-1)$ chiều) trong công thức (p2.8). Vì $\|\cdot\|_2$ – chuẩn 2 (căn bậc hai của tổng các bình phương các thành phần của vectơ) nhỏ nhất, trung bình lỗi sẽ nhỏ nhất.

Để tìm nghiệm cho bài toán tối thiểu hóa này, xét bình phương chuẩn 2 ở trên. Với vectơ v , $\|v\|_2^2 = v^T v$. Do vậy,

$$\|Ax - b\|_2^2 = (Ax - b)^T (Ax - b) = x^T A^T Ax - 2x^T A^T b + b^T b \quad (\text{p2.9})$$

Tối thiểu hóa biểu thức này tương đương tối thiểu hóa trung bình bình phương lỗi. Xem biểu thức này như một hàm của x , gradient của nó được đặt bằng 0.

$$2A^T Ax - 2A^T b = 0 \Leftrightarrow A^T Ax = A^T b \quad (\text{p2.10})$$

Công thức (p2.10) được gọi là phương trình chính tắc cho bài toán bình phương tối thiểu tuyến tính. Có nhiều lời giải cho bài toán này, ví dụ phương pháp Cholesky hoặc QR

(bằng cách thay thế $A = QR$, Q là một ma trận trực giao và R là một ma trận tam giác trên), với chi phí khác nhau và tính ổn định khác nhau.

Ví dụ 2.2: Để minh họa khái niệm này, hãy xem ví dụ trước, giả thiết ba điểm neo chỉ có thông tin vị trí ước lượng không chính xác $\tilde{r}_1=5$, $\tilde{r}_2=1$, $\tilde{r}_3=4$. Giải hệ phương trình (p2.7) cho vị trí không chính xác (5.2, 4.8) với khoảng cách $\sqrt{(5.2 - 5)^2 + (4.2 - 2)^2} \approx 2.2$ giữa vị trí ước lượng và vị trí đúng.

Thêm các điểm neo ở $(x_4, y_4) = (3, 1)$, $(x_5, y_5) = (7, 5)$, $(x_6, y_6) = (2, 8)$, và $(x_7, y_7) = (4, 6)$ với các ước lượng khoảng cách lần lượt là $\tilde{r}_4 = 2$, $\tilde{r}_5 = 3$, $\tilde{r}_6 = 7$, $\tilde{r}_7 = 4$ sẽ cải thiện ước lượng. Ma trận A và vectơ b là

$$A = \begin{bmatrix} 2 & 5 \\ -1 & 2 \\ -4 & 4 \\ 1 & 5 \\ -3 & 1 \\ 2 & -2 \end{bmatrix} \quad b = \begin{bmatrix} 56 \\ -4 \\ -16 \\ 30 \\ -29 \\ 17 \end{bmatrix} \quad (\text{p2.11})$$

Giải $A^T A x = A^T b$ được $x = (5.5, 2.7)$, với lỗi khoảng cách $\sqrt{(5.5 - 5)^2 + (2.7 - 2)^2} \approx 0.86$.